Toward Abstractive Text Summarization

Elaheh ShafieiBavani

A thesis in fulfillment of the requirements for the degree of

Doctor of Philosophy

UNSW
AUSTRALIA

School of Computer Science and Engineering

Faculty of Engineering

The University of New South Wales

August 2018

THE UNIVERSITY OF NEW SOUTH WALES
Thesis/Dissertation Sheet

Surname or Family name: **ShafieiBavani**

First name: **Elaheh** Other name/s: **Ella Shafiei**

Abreviation for degree as given in the University calendar: **PhD**

School: **School of Computer Science and Engineering** Faculty: **Faculty of Engineering**

Title: Toward Abstractive Text Summarization

Abstract 350 words maximum

Automatic text summarization is the process of automatically creating a compressed version of a given text. Content reduction can be addressed by extraction or abstraction. Extractive methods select a subset of most salient parts of the source text for inclusion in the summary. In contrast, abstractive methods build an internal semantic representation to create a more human-like summary. The majority of summarizers are designed to be extractive due to the complex nature of abstraction. This thesis moves toward abstractive text summarization, and makes this task: (i) more adaptable to a wide range of applications; (ii) more dynamic to different sources and types of text; and (iii) better evaluated using semantic representations.

To make it more adaptable, we propose a word graph-based multi-sentence compression approach for improving both informativity and grammaticality of summaries, which shows 44% error reduction over state-of-the-art systems. Then, we discuss adapting this approach into query-focused multi-document summarization, focusing on semantic similarities between the input query and source texts. This approach satisfies the query-biased relevance, information novelty and richness criteria.

To make this task more dynamic, we appraise the coverage of knowledge sources for the purpose of abstractive text summarization, and found a decline in performance of summarizers that only rely on specific terminologies. Our approach integrates general and domain-specific lexicons for incorporating textual semantic similarities, and bridging the knowledge and language gaps in domain-specific summarizers.

To fairly evaluate abstractive summaries including lexical variations and paraphrasing, we propose an approach based on both lexical and semantic similarities, which highly correlates with human judgments. Furthermore, we present an approach to evaluate summaries on test sets where model summaries are not available. Our hypothesis is that comparing semantic representations of the input and summary content leads to a more accurate evaluation. We exploit the compositional capabilities of corpus-based and lexical resource-based word embeddings for predicting the summary content quality. The experiment results support our proposal to use semantic representations for model-based and model-free evaluation of summaries.

FOR OFFICE USE ONLY	Date of completion of requirements for Award

Originality Statement

'I hereby declare that this submission is my own work and to the best of my knowledge it contains no materials previously published or written by another person, or substantial proportions of material which have been accepted for the award of any other degree or diploma at UNSW or any other educational institution, except where due acknowledgement is made in the thesis. Any contribution made to the research by others, with whom I have worked at UNSW or elsewhere, is explicitly acknowledged in the thesis. I also declare that the intellectual content of this thesis is the product of my own work, except to the extent that assistance from others in the project's design and conception or in style, presentation and linguistic expression is acknowledged.'

Elaheh ShafieiBavani
January 30, 2019

INCLUSION OF PUBLICATIONS STATEMENT

UNSW is supportive of candidates publishing their research results during their candidature as detailed in the UNSW Thesis Examination Procedure.

Publications can be used in their thesis in lieu of a Chapter if:

- The student contributed greater than 50% of the content in the publication and is the "primary author", ie., the student was responsible primarily for the planning, execution and preparation of the work for publication
- The student has approval to include the publication in their thesis in lieu of a Chapter from their supervisor and Postgraduate Coordinator.
- The publication is not subject to any obligations or contractual agreements with a third party that would constrain its inclusion in the thesis

Please indicate whether this thesis contains published material or not.

☐ *This thesis contains no publications, either published or submitted for publication (if this box is checked, you may delete all the material on page 2)*

☑ *Some of the work described in this thesis has been published and it has been documented in the relevant Chapters with acknowledgement (if this box is checked, you may delete all the material on page 2)*

☐ *This thesis has publications (either published or submitted for publication) incorporated into it in lieu of a chapter and the details are presented below*

CANDIDATE'S DECLARATION

I declare that:

- I have complied with the Thesis Examination Procedure
- where I have used a publication in lieu of a Chapter, the listed publication(s) below meet(s) the requirements to be included in the thesis.

Name	Signature	Date (dd/mm/yy)
Elaheh ShafieiBavani		29/08/2018

Postgraduate Coordinator's Declaration (to be filled in where publications are used in lieu of Chapters)

I declare that:

- the information below is accurate
- where listed publication(s) have been used in lieu of Chapter(s), their use complies with the Thesis Examination Procedure
- the minimum requirements for the format of the thesis have been met.

PGC's Name	PGC's Signature	Date (dd/mm/yy)

Copyright Statement

Elaheh ShafieiBavani
January 30, 2019

Authenticity Statement

Elaheh ShafieiBavani
January 30, 2019

Abstract

Automatic text summarization is the process of automatically creating a compressed version of a given text. Content reduction can be addressed by extraction or abstraction. Extractive methods select a subset of most salient parts of the source text for inclusion in the summary. In contrast, abstractive methods build an internal semantic representation to create a more human-like summary. The majority of summarizers are designed to be extractive due to the complex nature of abstraction. This thesis moves toward abstractive text summarization, and makes this task: (i) more adaptable to a wide range of applications; (ii) more dynamic to different sources and types of text; and (iii) better evaluated using semantic representations.

To make it more adaptable, we propose a word graph-based multi-sentence compression approach for improving both informativity and grammaticality of summaries, which shows 44% error reduction over state-of-the-art systems. Then, we discuss adapting this approach into query-focused multi-document summarization, focusing on semantic similarities between the input query and source texts. This approach satisfies the query-biased relevance, information novelty and richness criteria.

To make this task more dynamic, we appraise the coverage of knowledge sources for the purpose of abstractive text summarization, and found a decline in performance of summarizers that only rely on specific terminologies. Our approach integrates general and domain-specific lexicons for incorporating textual semantic similarities, and bridging the knowledge and language gaps in domain-specific summarizers.

To fairly evaluate abstractive summaries including lexical variations and paraphrasing, we propose an approach based on both lexical and semantic similarities, which highly correlates with human judgments. Furthermore, we present an approach to evaluate summaries on test sets where model summaries are not available. Our hypothesis is that comparing semantic representations of the input and summary content leads to a more accurate evaluation. We exploit the compositional capabilities of corpus-based and lexical resource-based word embeddings for predicting the summary content quality. The experiment results support our proposal to use semantic representations for model-based and model-free evaluation of summaries.

Acknowledgements

Firstly, I would like to offer my sincere gratitude to my knowledgeable and friendly supervisor, Raymond Wong and my experienced co-supervisor, Fang Chen. I am grateful for their great patience, continuous support, professional integrity and insightful feedback on my academic and even personal development. I am thankful to Alan Blair, Fethi Rabhi, and Helen Paik, my panel members, who have always provided me with helpful and practical suggestions on my research and future career. Many thanks also goes to Philip Cohen and Bin Li for being my thesis examiners and providing valuable suggestions and comments to improve this thesis. I do appreciate all kind support and encouragement I received from Salil Kanhere, the Postgraduate Research Coordinator of the School of Computer Science and Engineering.

My sincere thanks also goes to my Bachelor's thesis supervisor and Master's thesis co-supervisor, Mostafa Fakhrahmd, for introducing me to Natural Language Processing, and encouraging me to do a PhD in this area.

I would like to extend my gratitude to my former manager, Hamidreza Javid, for supporting and motivating me to approach my professional goals.

I am grateful for the Australian Government Research Training Program (AGRTP) Scholarship, International Postgraduate Research Scholarship (IPRS), and the financial and travel supports I received from the University of New South Wales, NICTA, and CSIRO's Data61 during my research.

I wish to express my sincerest thanks, love and deep appreciation to my family. I appreciate all the love and support I have ever received from my parents Parviz and Iran, and my Mohammad who happens to be my dearest friend.

Finally, I would like to appreciate all motivation and kind support I received from my beloved brother, who taught me to never give up on my dreams. May he live on in our hearts forever.

Abbreviations

AESOP Automatically Evaluating Summaries of Peers

AMR Abstract Meaning Representation

BE Basic Elements

CNN Convolutional Neural Network

CRF Conditional Random Fields

CW Chinese Whispers

DUC Document Understanding Conference

EBM Evidence-based Medicine

IDF Inverse Document Frequency

ILP Integer Linear Programming

IR Information Retrieval

JSD Jensen Shannon Divergence

KLD Kullback-Leibler Divergence

LDA Latent Dirichlet Allocation

LSA Latent Semantic Analysis

LSTM Long-Short-Term Memory

MMR Maximum Marginal Relevance

MSC Multi-sentence Compression

MWE Multiword Expressions

NER Named Entity Recognition

NLP Natural Language Processing

OOV Out-of-vocabulary

POS-LM POS-based Language Model

POS Part-Of-Speech

PPDB Paraphrase Database

PPR Personalized PageRank

RNN Recurrent Neural Networks

ROUGE Recall-Oriented Understudy for Gisting Evaluation

RST Rhetorical Structure Theory

SCU Summary Content Units

SOR Strength of Recommendation

SVM Support Vector Machine

SVR Support Vector Regression

TAC Text Analysis Conference

TF Term Frequency

UMLS Unified Medical Language System

WSD Word Sense Disambiguation

Contents

1 Introduction **1**

1.1 Background and Motivation 1

1.1.1 Shortcomings of Extractive Summarization 2

1.2 Aim and Scope 5

1.2.1 Word Graph-based Multi-sentence Compression 6

1.2.2 Query-focused Multi-document Summarization 7

1.2.3 Domain-specific Multi-document Summarization 8

1.2.4 Model-based Semantic Evaluation of Summaries 10

1.2.5 Model-free Summary Content Evaluation 11

1.3 Contributions 11

1.4 Thesis Outline 14

2 Literature Review **17**

2.1 Automatic Text Summarization 18

2.1.1 Various Types of Text Summarization 18

2.1.2 The Conventional Framework 22

2.1.3 Recent Advances 28

2.2 Beyond Sentence Extraction 31

2.2.1 Compressive Summarization 31
2.2.2 Toward Full Abstraction 36
2.2.3 Toward End-to-end Abstractive Summarization 41
2.2.4 Toward Abstraction in Specific Domains and Genres 43
2.3 Automatic Evaluation of Text Summarization 47
2.3.1 Model-based Summarization Evaluation 51
2.3.2 Model-free Summarization Evaluation 53
2.4 Applications 54
2.5 Summary 55

3 Word Graph-based Multi-sentence Compression 57
3.1 Proposed Approach 59
3.1.1 Word Graph Construction for MSC 59
3.1.2 Merging and Mapping Strategies 60
3.1.3 Re-ranking Strategies 67
3.2 Data Preparation 71
3.3 Experiments 73
3.3.1 Evaluation Metrics 73
3.3.2 Experiment Results 74
3.4 Summary 78

4 Query-focused Multi-document Summarization 79
4.1 Proposed Approach 80
4.1.1 Capturing Semantic Similarities 80
4.1.2 Semantic Similarities at Sentence Level 82

4.1.3 Semantic Disambiguation 83
4.1.4 Query-biased Relevance 84
4.1.5 Graph-based Clustering 86
4.1.6 Query-biased Information Novelty 88
4.1.7 Query-biased Information Richness 91
4.2 Data 92
4.3 Experiments 94
4.3.1 Evaluation Metrics 94
4.3.2 Experiment Results 95
4.4 Summary 101

5 Domain-specific Multi-document Summarization 103

5.1 Data 104
5.1.1 UMLS 105
5.1.2 WordNet 106
5.1.3 UMLS vs. WordNet 106
5.1.4 EBM Corpus 107
5.2 Preprocessing 109
5.2.1 Biomedical Domain Peculiarities 109
5.2.2 Preprocessing Steps 111
5.3 Proposed Approach 112
5.3.1 Semantic Similarities on UMLS 113
5.3.2 UMLS-based Semantic Disambiguation 115
5.3.3 Semantic Similarities at Sentence Level 116

5.3.4 Sentence Pruning and Clustering . . . 118
5.3.5 Abstractive Summarization of Medical Evidence . . . 119
5.4 Experiments . . . 123
5.4.1 Evaluation Metrics . . . 123
5.4.2 Experiment Results . . . 123
5.5 Summary . . . 130

6 Model-based Semantic Evaluation of Summaries 131
6.1 Proposed Approach . . . 132
6.1.1 Graph-theoretic Summary Evaluation . . . 132
6.1.2 OOV Handling . . . 136
6.1.3 Multiple Levels of Evaluation . . . 137
6.2 Experiments . . . 139
6.2.1 Data and Meta-evaluation . . . 139
6.2.2 Experiment Results . . . 141
6.2.3 Significance Test . . . 145
6.2.4 Exploring Scaling Factor . . . 148
6.3 Summary . . . 149

7 Model-free Summary Content Evaluation 150
7.1 Data and Evaluation Metrics . . . 151
7.2 Proposed Approach . . . 153
7.2.1 Distributional Semantic Similarity . . . 154
7.2.2 Topical Relevance . . . 157
7.2.3 Query Relevance . . . 158

7.2.4 Coherence 158
7.2.5 Novelty 158
7.2.6 Feature Combination with SVR 159
7.3 Experiments and Results 160
7.3.1 Error Analysis 164
7.3.2 Evaluation on the Test Set 165
7.4 Summary 166

8 Conclusion **167**
8.1 Summary of Chapters and Contributions 168
8.2 Future Work 174
8.2.1 Short-term Extensions 174
8.2.2 Future Directions 177

Bibliography **180**

List of Figures

2.1 Example of sequence-to-sequence learning with neural networks . . . 41

2.2 Example of a pyramid with SCUs identified and marked 51

3.1 Overview of the proposed word graph-based MSC approach 58

3.2 Example of detecting, merging, and mapping multi-word expressions . 62

3.3 Example of Synonym Mapping 63

3.4 The generated word graph and a salient path 64

3.5 The effectiveness of ROUGE and BLEU 77

3.6 The impacts of the improvements separately 78

4.1 Overview of the proposed news summarization approach 80

4.2 Partial view of the Similarity Graph: sentence-to-query and sentence-to-sentence similarity edges are depicted as solid and dashed lines, respectively; dotted lines reveal the loose sentence-to-sentence relations. 85

4.3 Example of our multi-sentence compression graph. Thick edges indicate the salient path, where PIDs define the order of nodes. 90

4.4 Example of using Link Grammar Parser for the syntactic analysis of a sample generated sentence; (a) and (b) demonstrate two found complete linkages with no p.p. violations. The constituent trees (c) and (d) correspond to linkages (a) and (b), respectively. 98

4.5 Syntactic analysis of a number of 600 generated summary sentences using Link Grammar Parser 100

4.6 Exploring scaling factor μ in Equation 3.8 on the development set . . 101

5.1 Overview of the proposed framework 104

5.2 Example of difference between WordNet and UMLS 107

5.3 Screenshot of Mapping the term *cold* using MetaMap 116

5.4 Example of the Constructed Word Graph. Thick edges indicate salient paths. 120

5.5 Example of Biomedical Synonym Mapping 121

5.6 Average scores by ROUGE metrics over the EBM corpus 125

5.7 Exploring *Scaling Factor 1* (η in Equation 5.3), and *Scaling Factor 2* (μ in Equation 3.8) on the development set. 128

5.8 Example of using Link Grammar Parser for the syntactic analysis of a sample generated sentence *"A treatment strategy for chronic daily headaches is medication withdrawal."* 129

5.9 Syntactic analysis of a number of 600 generated summary sentences using Link Grammar Parser 129

6.1 Comparing PPR vectors between $n\text{-}gram_m$ (unigrams and bigrams in a model summary text) and a peer summary text 136

6.2 Correlation of the best-performing AESOP metrics with the manual metric of Pyramid using the correlation metrics of Pearson r, Spearman ρ, and Kendall τ on the TAC 2011 AESOP dataset 142

6.3 Correlation of the best-performing AESOP metrics with the manual metric of Responsiveness using the correlation metrics of Pearson r, Spearman ρ, and Kendall τ on the TAC 2011 AESOP dataset 143

6.4 Correlation of the best-performing AESOP metrics with the manual metric of Readability using the correlation metrics of Pearson r, Spearman ρ, and Kendall τ on the TAC 2011 AESOP dataset 144

6.5 Exploring scaling factor β on the TAC 2010 AESOP dataset 148

7.1 The weighted centroid embedding of text $T = \{t_1, t_2, ..., t_n\}$ 156

List of Tables

1.1 Extractive Summarization vs. Abstractive Summarization 3

3.1 Information about the constructed dataset for MSC 72

3.2 Points scale defined in the agreement between raters 74

3.3 Manual Evaluation: Average scores over normal and diverse clusters, along with the estimated compression rates 75

3.4 Automatic Evaluation: Average scores over normal and diverse clusters 76

3.5 The impacts of the improvements separately 77

4.1 Part of the summary generated by human, an extractive system, and our summarization approach (Proposed-Abs) for topic D0626H (DUC 2006). Greyed out parts in extractive summary, are query-irrelevant phrases, such as temporal information or source of the news, and also redundant parts which have been automatically removed in the Proposed-Abs. 93

4.2 Information about the utilized DUC datasets 94

4.3 Evaluation on DUC 2005 Dataset 96

4.4 Evaluation on DUC 2006 Dataset 96

4.5 Evaluation on DUC 2007 Dataset 96

4.6 Standard deviation of ROUGE scores for the summaries generated by Proposed-Abs across DUC 2005, DUC 2006, and DUC 2007 datasets 97

5.1 Information about the EBM Corpus 109

5.2 Example of query-focused multi-document summarization, showing the question, the bottom-line summary and two of the source abstracts. 111

5.3 MetaMap mapping for the sentence " *There is no evidence of increased risk for major bleeding as a result of falls in hospitalized patients taking warfarin.*" 114

5.4 Example of Clustering Potential of the Utilized Corpus 119

5.5 Average scores by ROUGE metrics over the EBM corpus 125

5.6 Example of different summaries: Gold summary; Proposed-Abs summary; and LexRank summary. 126

5.7 Standard deviation of ROUGE scores for the summaries generated by Proposed-Abs 127

6.1 Correlation results ($p < 0.05$) with the manual metric of Pyramid using the correlation metrics of Pearson r, Spearman ρ, and Kendall τ. The best correlations are specified in bold, and the underlined scores show the top correlations in the TAC AESOP 2011. 145

6.2 Correlation results ($p < 0.05$) with the manual metric of Responsiveness using the correlation metrics of Pearson r, Spearman ρ, and Kendall τ. The best correlations are specified in bold, and the underlined scores show the top correlations in the TAC AESOP 2011. . 146

6.3 Correlation results ($p < 0.05$) with the manual metric of Readability using the correlation metrics of Pearson r, Spearman ρ, and Kendall τ. The best correlations are specified in bold, and the underlined scores show the top correlations in the TAC AESOP 2011. 147

7.1 Input-summary evaluation on the query focused and update summarization tasks from TAC 2008 data: MACRO level Spearman correlations, all results are significant ($p < 0.05$). 161

7.2 Input-summary evaluation on the query focused and update summarization tasks from TAC 2008 data: MICRO level percentage of inputs with significant correlations ($p < 0.05$). 163

7.3 Error analysis: Overlap between ROUGE-2 and SVR predictions for the best system in a pair (TAC 2008, 1,653 pairs). The gold-standard judgment for a better system is computed using pyramid. 164

7.4 Input-summary evaluation on the query focused and update summarization tasks from TAC 2009 data: MACRO level Spearman correlations, all results are significant ($p < 0.05$). 165

7.5 Input-summary evaluation on the query focused and update summarization tasks from TAC 2009 data: MICRO level percentage of inputs with significant correlations ($p < 0.05$). 166

Publications

Large portions of Chapter 3 have appeared in the following papers:

- **E. ShafieiBavani**, M. Ebrahimi, R. Wong, F. Chen, "An Efficient Approach for Multi-Sentene Compression" , In *Proceedings of Machine Learning Research (PMLR): Asian Conference on Machine Learning (ACML 2016), volume 63, pages 414-429, 2016*

- **E. ShafieiBavani**, M. Ebrahimi, R. Wong, F. Chen, "On Improving Informativity and Grammaticality for Multi-Sentence Compression", *arXiv preprint arXiv:1605.02150* (2016)

Large portions of Chapter 4 have appeared in the following paper:

- **E. ShafieiBavani**, M. Ebrahimi, R. Wong, F. Chen, "A Query-based Summarization Service from Multiple News Sources", In *Proceedings of the 2016 IEEE International Conference on Services Computing (SCC 2016)*, pages 42-49, IEEE, 2016 **(Best Student Paper)** (CORE Rank: A)

Large portions of Chapter 5 have appeared in the following paper:

- **E. ShafieiBavani**, M. Ebrahimi, R. Wong, F. Chen , "Appraising UMLS Coverage for Summarizing Medical Evidence", In *Proceedings of the 26th International Conference on Computational Linguistics (COLING 2016)*, pages 513-524, ACL, 2016 (CORE Rank: A)

Large portions of Chapter 6 have appeared in the following papers:

- **E. ShafieiBavani**, M. Ebrahimi, R. Wong, F. Chen, "A Graph-theoretic Summary Evaluation for ROUGE", In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: short paper (EMNLP 2018)*, pages 762-767, ACL, 2018 (CORE Rank: A)

- **E. ShafieiBavani**, M. Ebrahimi, R. Wong, F. Chen, "A Semantically Motivated Approach to Compute ROUGE Scores", *arXiv preprint arXiv:1710.07441* (2017)

Large portions of Chapter 7 have appeared in the following paper:

- **E. ShafieiBavani**, M. Ebrahimi, R. Wong, F. Chen , "Summarization Evaluation in the Absence of Human Model Summaries Using the Compositionality of Word Embeddings", In *Proceedings of the 27th International Conference on Computational Linguistics (COLING 2018)*, pages 905-914, ACL, 2018 (CORE Rank: A)

Other Co-authored Publications

- M. Ebrahimi, **E. ShafieiBavani**, R. Wong, F. Chen, "A Unified Neural Network Model for Geolocating Twitter Users", In *Proceedings of the 22nd SIGNLL Conference on Computational Natural Language Learning (CoNLL 2018)*, pages 42-53, ACL, 2018 (CORE Rank: A)

- M. Ebrahimi, **E. ShafieiBavani**, R. Wong, F. Chen, "Leveraging Local Interactions for Geolocating Social Media Users", In *Proceedings of the 22nd Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD 2018)*, pages 803-815, Springer, 2018 (CORE Rank: A)

- M. Ebrahimi, **E. ShafieiBavani**, R. Wong, F. Chen, "Twitter User Geolocation by Filtering of Highly Mentioned Users", *Journal of the Association for Information Science and Technology (JASIST)*, volume 69(7), pages 879-889, Wiley Online Library, 2018, (ERA Rank: A*)

- M. Ebrahimi, **E. ShafieiBavani**, R. Wong, F. Chen , "Exploring Celebrities on Inferring User Geolocation in Twitter", In *Proceedings of the 21st Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD 2017)*, pages 395-406, Springer, 2017 (CORE Rank: A)

- M. Ebrahimi, **E. ShafieiBavani**, R. Wong, S. Fong, J. Fiaidhi, "An Adaptive Meta-heuristic Search for the Internet of Things" , *Journal of Future Generation Computer Systems (FGCS)*, volume 76, pages 486-494, 2017 (ERA Rank: A)

- A. Hamzehi, M. Ebrahimi, **E. ShafieiBavani**, R. Wong , F. Chen, "Scalable Sentiment Analysis for Microblogs based on Semantic Scoring", In *Proceedings of the 12th IEEE International Conference on Services Computing (SCC 2015)*, pages 271-278, IEEE, 2015 (CORE Rank: A)

- M. Ebrahimi, **E. ShafieiBavani**, R. Wong, Ch. Chi, "A New Meta- Heuristic Approach for Efficient Search in the Internet of Things", In *Proceedings of the 12th IEEE International Conference on Services Computing (SCC 2015)*, pages 264-270, IEEE, 2015 (CORE Rank: A)

Chapter 1

Introduction

This thesis is about abstractive text summarization - *a content reduction technique that generates new summary sentences representing the gist of the content of the source document(s).* In this chapter, we briefly introduce automatic text summarization, and its main categories. We then move toward abstractive text summarization and its importance while addressing the shortcomings of extractive level. Finally, we explain the scope and contributions of this thesis, and finish the chapter by introducing the upcoming chapters.

1.1 Background and Motivation

The explosive growth of Internet content requires development of automatic techniques to present information to users in an effective way. For example, a user may search the Internet news (e.g., Google News[1]) to survey a particular topic of interest. Multiple topic-relevant results are usually returned. It is time consuming and

[1]http://news.google.com/

tedious to read through all these results.

Text summarization has been introduced as one of the techniques of Natural Language Processing (NLP), with the aim of producing a short summary from one (single document summarization) or multiple (multi-document summarization) text documents by condensing, selecting, or generating the important information (Jones, 1999). A good summary should cover the most salient information while being coherent, non-redundant and grammatically readable. Text summarization approaches can be categorized into two main groups:

- **Extractive**, which selects a subset of most salient parts of the source text, and concatenates them for inclusion in the final summary. It is important that the selected parts are different enough to avoid redundancy in the summary.
- **Abstractive**, which generates new summary sentences representing the gist of the content of the source text. Abstractive methods build an internal semantic representation to create a more human-like summary using natural language generation techniques.

Table 1.1 shows a simple example for comparing extractive and abstractive summaries of a given source text.

1.1.1 Shortcomings of Extractive Summarization

Consider the four sentences in Examples 1.1.1 to 1.1.4 which were manually extracted from four related news articles about the car crash in New Jersey on Saturday, May 23, 2015:

Example 1.1.1. US mathematician John Nash, whose life story was turned into the Oscar-winning film A Beautiful Mind, has died in a car crash.

Source Text: Digital is the wires, but digital intelligence, or artificial intelligence as some people call it, is about much more than that. This next decade is about how you combine those and become a cognitive business. It's the dawn of a new era.
Extractive Summary: Digital is the wires, but digital intelligence, or artificial intelligence as some people call it, is about much more than that.
Abstractive Summary: The dawn of a new era is about how you combine digital and artificial intelligence to become a cognitive business.

Table 1.1: Extractive Summarization vs. Abstractive Summarization

Example 1.1.2. John Nash, the 86-year-old mathematician who inspired 'A Beautiful Mind', died in a car crash along with his wife Alicia, 82, the BBC reports.

Example 1.1.3. Nash, whose struggle with schizophrenia was chronicled in the 2001 movie 'A Beautiful Mind', died in a car crash in New Jersey on Saturday, May 23, 2015, the police said.

Example 1.1.4. The Nobel-winning mathematician whose pioneering work in game theory and torturous tussle with severe mental illness became the subject of A Beautiful Mind, died in a car crash in New Jersey.

The above sentences are all about a Nobel-winning mathematician, John Nash, who died in a car crash with his wife. However, each of the sentences contains bits of information which other sentences lack. For example, it is only Example 1.1.1 that tells the movie 'A Beautiful Mind' inspired by Nash's life is an Oscar-winning movie; likewise, it is only Example 1.1.2 which points out that John Nash died in 86 along with his wife, Alicia; only Example 1.1.3 says that John Nash struggled with schizophrenia, and mentions the date of the accident; Example 1.1.4 is also the only sentence that tells us John Nash was a Nobel-winning mathematician and a pioneer in game theory and torturous tussle.

Given the above sentences, an extractive summarizer is highly likely to rank them equally, and select each of them as a summary of the event. Therefore, the extractive summarization system faces the following challenges:

(i) *trade-off between non-redundancy and completeness* - If the summary includes a sentence, some important information is put away. Concatenating two or more sentences also makes the summary redundant.

(ii) *trade-off between information relevance and summary length constraints* - some sentences contain information irrelevant to the main event. For example, 'the BBC reports' in Example 1.1.2, or 'the police said' in Example 1.1.3 could be eliminated due to summary length constraints. Similarly, 'a pioneer in game theory and torturous tussle' in Example 1.1.4 could be omitted, given that it is further explanation to the 'Noble-winning' adjective, and its omission does not have any impact on the summary main point. Including some information like 'Nash's schizophrenia inspired the Oscar-winning film A Beautiful Mind' in the summary is not also necessary considering the length limitation.

Thereupon, an ideal summarizer should generate novel sentences from the source text such that these sentences incorporate important content from several parts of the source text and exclude irrelevant information. Hence, an ideal summary for the above examples would be:

Example 1.1.5. 86-year-old schizophrenic mathematician and Nobel Prize winner, John Nash, died in a car crash with his wife, in New Jersey on May 23, 2015.

Although extractive approaches are unable to produce non-redundant and complete summaries at the same time (Filippova and Strube, 2008), the majority of existing summarizers are still extractive (Wang et al., 2016). Hence, we are motivated to make contributions toward abstractive text summarization. To be abstractive and

more human-like, a summarizer should interpret the source text, construct its semantic representation, and make necessary inferences to generate a summary from the representation (Jones, 1999). Therefore, abstractive summarization is much more complex than extractive level, and faces the challenges of *Natural Language Understanding* to select the truly important content, and *Natural Language Generation* to generate summary sentences.

1.2 Aim and Scope

Since there still exists huge quality gap between automatic and human-written summaries, we need good summarizers that consider all semantically important information described in the source documents. This emphasizes the necessity of proposing summarization approaches in the abstractive level across all types of text and domains that can be applied to a wide range of applications. We also need effective evaluation metrics to assess the newly generated summaries. Moving toward abstractive text summarization, we aim to make this task more adaptable to a wide range of applications, more dynamic to different sources and types of text, and better evaluated using semantic representations. To achieve these goals, we focus on the following tasks in this thesis and will introduce them in this section:

- Word Graph-based Multi-sentence Compression
- Query-focused Multi-document Summarization
- Domain-specific Multi-document Summarization
- Model-based Semantic Evaluation of Summaries
- Model-free Summary Content Evaluation

1.2.1 Word Graph-based Multi-sentence Compression

Collaborating compression and summarization has recently been known as a promising step toward abstractive summarization (Li et al., 2013). The reason is its ability to remove insignificant sentence constituents and make room for more salient information. Hence, our first focus is on Multi-sentence Compression (MSC), that refers to *the method of mapping a collection of related sentences to a sentence shorter than the average length of the input sentences, while retaining the most important information that conveys the gist of the content, and still remaining grammatically correct* (Jing, 2000; Boudin and Morin, 2013). MSC is one of the challenging tasks in NLP that has recently attracted increasing interest. This is mostly because of its potential use in various applications such as guided microblog summarization, opinion summarization, newswire summarization, text simplification for mobile devices and so on.

A standard way to generate summaries in this task usually consists of the following steps: (i) ranking sentences by their importance; (ii) clustering them by their similarity; and (iii) selecting a sentence from the top ranked clusters (Wang et al., 2008b). Most of the MSC approaches, e.g., (Filippova and Strube, 2008; Elsner and Santhanam, 2011) rely on syntactic parsers to produce grammatical compressions. As an alternative, some recent work in the field (Filippova, 2010b; Boudin and Morin, 2013) is based on word graphs, which only require a Part-Of-Speech (POS) tagger and a list of stopwords[2]. These approaches simply rely on words of the sentences and efficient dynamic programming. They take advantage of the redundancy among a set of related sentences to generate informative and grammatical summaries.

[2]Stopwords are frequent words which carry little or no standalone semantic content, like determiners.

Although the proposed approach by Filippova (2010b) introduces an elegant word graph to MSC, approximately half of their generated sentences are missing important information about the set of related sentences (Boudin and Morin, 2013). Therefore, Boudin and Morin (2013) enhanced their work and produced more informative sentences by maximizing the range of topics they cover. However, they reported that grammaticality scores are decreased, since their re-ranking algorithm produces longer compressions to ameliorate informativity. Thereupon, grammaticality might be sacrificed while enhancing informativity and vice versa.

In this thesis, we intend to tackle the main difficulty of the above mentioned MSC approaches, which is to simultaneously improve both informativity and grammaticality of the compressed sentences.

1.2.2 Query-focused Multi-document Summarization

As mentioned in Section 1.1, text summarization can be performed in two levels of extractive and abstractive. A summary can either be *query-focused* (biased to a user query), or *generic* (conveying the document gist). Recently, query-focused multi-document summarization has shown potential use in a number of information analysis applications including open-ended question answering, recommender systems, and summarization of search engine results (Wang et al., 2016).

One of the known challenges of Query-focused Multi-Document Summarization is to extract the most non-redundant query-relevant parts of the documents (Mohamed and Oussalah, 2015). This involves the ability to understand the underlying semantic relatedness of query and document sentences. The problem with current statistical methods is that they fail to capture the semantic similarities when comparing a sentence with a user query. Hence, there is often a conflict between the extracted sentences and users' requirements (Abdi et al., 2015).

In this thesis, we utilize a distributional lexico-semantic model to quantify the semantic similarities between linguistic items. We also investigate the applicability of multi-sentence compression to enhance query-focused multi-news summarization from extractive to abstractive level.

1.2.3 Domain-specific Multi-document Summarization

Over the past two decades, clinical guidelines urged practitioners to move towards evidence-based medicine, which is formally defined as *'conscientious and judicious use of current best evidence in making decisions about the care of individual patients'* (Sackett et al., 1996). Evidence-based medical practice heavily relies on research evidence, rather than intuition, unsystematic clinical experience, or pathologic rationale (Group et al., 1992). However, searching through and evaluating primary medical literature is extremely time consuming (Sarker et al., 2015). Even targeted searches tend to return a large set of relevant documents, and not summaries or answers to the queries.

Even though the problem of information overload and the advantages of summarization are critical in the biomedical domain, the majority of summarizers are designed to be general-purpose. They usually work with a simple representation of the summary comprising of information that can be directly extracted from the document, such as terms, phrases, or sentences (Mihalcea and Tarau, 2004). However, the study by Fiszman et al. (2004) has demonstrated the benefits of summarization based on richer representations that makes use of domain-specific knowledge sources. These approaches represent the documents using concepts instead of words, and may also be enriched by using semantic associations among concepts (e.g., synonymy, hypernymy, etc.) (Plaza et al., 2011).

While a query is asked in the field of biomedicine, one of the main challenges is

to understand the underlying semantic relatedness of the query and document sentences, and consequently extract the most non-redundant, query-relevant parts from the documents. Documents in biomedicine are very different from documents in other fields, and include very different document types (e.g., patient records, web documents, scientific papers, etc.) (Plaza et al., 2011). Therefore, particular characteristics and the type of biomedical documents are required to be exploited by the summarization systems. To this end, promising domain specific NLP techniques have been efficiently employed to release a repository of biomedical vocabularies named the Unified Medical Language System (UMLS[3]) developed by the U.S. National Library of Medicine (Bodenreider, 2004). UMLS is a very rich source of information in medical and biological domain. Therefore, most existing biomedical summarizers utilize UMLS as a large lexical and semantic medical ontology. However, this ontology does not provide a full coverage of non-medical concepts, terms, and relations included in general-purpose thesauri such as WordNet[4] (Huang et al., 2009). Moreover, utilizing WordNet to complement the UMLS coverage is challenging due to their different structures, natures, terms, and sizes.

This challenge has motivated us to provide a deeper analysis of biomedical texts by keeping an eye on the biomedical peculiarities. Given a clinical query and a set of relevant medical evidence, our aim is to generate a fluent, well-organized, and compact summary that answers the query. The quality of biomedical summaries is also enhanced by appraising the applicability of both general-purpose (WordNet), and domain-specific (UMLS) knowledge sources for concept discrimination.

[3]http://www.nlm.nih.gov/research/umls/

[4]http://wordnet.princeton.edu/

1.2.4 Model-based Semantic Evaluation of Summaries

Quantifying the quality of summaries is an important and necessary task in the field of automatic text summarization. Among the metrics proposed for this task (Hovy et al., 2006; Tratz and Hovy, 2008; Giannakopoulos et al., 2008), ROUGE (Lin, 2004) is the first and still most widely used one (Graham et al., 2015). This metric measures the concordance of system-generated (peer) summaries and human-generated reference (model) summaries by determining n-grams, word sequences, and word pair matches. ROUGE assumes that a peer summary is of high quality if it shares many words or phrases with a model summary. However, different terminology may be used to refer to the same concepts and hence relying only on lexical overlaps may underrate content quality scores. For clarity, consider the following sentences:

(i) *They strolled around the city.*

(ii) *They took a walk to explore the town.*

These sentences are semantically similar, but lexically different. If one of them is included in a model summary, while a peer summary contains another one, ROUGE or other surface based evaluation metrics cannot capture their similarity due to the minimal lexical overlap.

In this thesis, we help ROUGE with identifying the semantic similarities of linguistic items, and consequently tackling the main problem of its bias towards lexical similarities.

1.2.5 Model-free Summary Content Evaluation

Current summary evaluation methods like manual and automated pyramid (Passonneau et al., 2005, 2013) and well-established ROUGE (Lin, 2004) heavily rely on multiple human-generated model summaries to assess the quality of system-generated summaries. This evaluation paradigm falls short on non-standard test sets where model summaries are not available. According to the quantitative analysis by Louis and Nenkova (2009a) and Singh and Jin (2016), evaluating summaries by their comparison with the input obtains good correlations with manual evaluations. Therefore, identifying a suitable input-summary similarity metric will provide a means for model-free evaluation of summaries.

In this thesis, we hypothesize that comparing semantic representations of the input and summary content will lead to a more accurate input-summary evaluation. Hence, we explore the effectiveness of compositionality of word embeddings in developing a model-free automatic metric to evaluate summary content quality.

1.3 Contributions

This section outlines the contributions of this thesis.

- We propose an effective word graph-based MSC approach to tackle the issue that most of the state-of-the-art MSC approaches are confronted with: i.e., improving both informativity and grammaticality at the same time. The contributions of this abstractive summarization approach can be summarized as follows: (i) we exploit Multiword Expressions (MWEs) from the given sentences and merge their words, constructing each MWE into a specific node in the word graph to reduce the ambiguity of mapping upcoming words, so

that well-organized and more informative compressions can be produced; (ii) we take advantage of the concept of synonymy in two ways: firstly, we replace a merged MWE with its *one*-word synonym if available, and secondly, we use the synonyms of an upcoming single word to find the most proper nodes for mapping; (iii) we train a 7-gram POS-based language model (POS-LM) to re-rank the k-shortest obtained paths in the word graph, and produce well-structured and more grammatical compressions, with an improved compression ratio; (iv) we construct a dataset made of clusters of English newswire sentences for MSC evaluation. To our knowledge, this approach presents the first attempt to use MWEs, synonymy and POS-LM for improving the quality of word graph-based MSC. The observed improvements on informativity and grammaticality of the generated compressions show an up to 44% error reduction over the state-of-the-art MSC systems.

- We present an effective query-focused multi-document summarization approach for newswire. Given a query and a set of news documents, our approach generates a well-organized and informative summary that answers the query through the following steps: (i) performing iterative random walks over WordNet to capture semantic similarities between sentences in the source text and the input query; (ii) sentence pruning to filter out less query-relevant sentences; (iii) clustering the relevant sentences using a graph-based clustering algorithm; (iv) abstractive summarization of the clusters through an MSC word graph, which considers the important key-phrases, along with the grammatical structure of the generated summaries. This query-focused multi-document summarizer has satisfied query-biased relevance, biased information novelty, and biased information richness. The effectiveness of our approach is demonstrated by conducting a set of experiments over the popular summarization benchmark datasets.

- We propose an effective approach for summarizing biomedical texts. Our approach generates a query-biased abstractive summary from a set of related biomedical abstracts mainly utilizing WordNet and UMLS to capture semantic similarities between sentences and the input query. The experiment results achieved using automatic evaluation over an evidence-based medicine corpus reveal that our approach outperforms the two competitive systems. We have tackled the main issue faced by state-of-the-art biomedical summarizers (i.e., decline in summarization performance due to the poor UMLS coverage of general concepts in the documents to be summarized) (Plaza et al., 2011). This issue is addressed by using WordNet to represent the layman knowledge, and UMLS to represent the professional knowledge. We believe that this approach can bridge the knowledge and language gaps in biomedical summarizers.

- We propose a graph-based approach (ROUGE-G) to overcome the limitation of high lexical dependency in ROUGE. Considering senses[5] instead of words, we leverage repetitive random walks on WordNet (Fellbaum, 1998) as a semantic network. We disambiguate each word into its intended sense, and obtain the probability distribution of each sense over all senses in the network. Weights in this distribution denote the relevance of the corresponding senses. At each iteration, we measure the semantic similarity by looking at the path taken by the random walker, and weighting the overlaps between a pair of ranked vectors. Our approach computes semantic similarity scores between n-grams, along with their match counts, to perform both semantic and lexical comparisons of peer and model summaries. The experiment results over standard evaluation datasets indicate that the variants of the proposed metric significantly outperform their corresponding variants of ROUGE. Beyond enhancing the evaluation prowess of ROUGE, due to its lexico-semantic analysis of summaries, we believe that our approach has the potential to expand the

[5]A sense is a symbolic form for meaning representation.

applicability of ROUGE to abstractive summarization.

- We present a new summary evaluation approach that does not require human model summaries. Our approach exploits the compositional capabilities of corpus-based and lexical resource-based word embeddings to develop the features reflecting coverage, diversity, informativeness, and coherence of summaries. The features are then used to train a learning model for predicting the summary content quality in the absence of gold models. We evaluate the proposed metric in replicating the human assigned scores for summarization systems and summaries on data from query-focused and update summarization tasks in benchmark summarization datasets. The experiment results show that our feature combination provides reliable estimates of summary content quality when model summaries are not available.

1.4 Thesis Outline

The rest of this thesis is organized as follows:

Chapter 2: This chapter provides a definition of automatic text summarization, its main categories and conventional framework. Then, we move toward abstractive summarization and explain its specific characteristics. Furthermore, we review the computational linguistic literature toward full abstraction, end-to-end and specific-domain abstraction, and discuss previously proposed approaches to perform automatic summarization evaluation along with their strengths and shortcomings. Previous studies on developing applications for text summarization are finally surveyed. Meanwhile, the literature of every proposed approach in this thesis is provided. This literature consists of previously proposed methods to perform abstractive summarization through multi-sentence compression, query-focused multi-document sum-

marization, domain-specific summarization, and finally summarization evaluation with and without human model summaries.

Chapter 3: This chapter details the first attempt (ShafieiBavani et al., 2016b,c) to simultaneously generate informative and grammatical compressions using a word graph representation, consists of three main components: (i) a merging strategy based on Multiword Expressions; (ii) a mapping strategy based on the notion of synonymy; (iii) a re-ranking strategy based on POS-based Language Model to identify the most grammatical compression candidates. We demonstrate the effectiveness of this approach with respect to automatic and manual evaluations over the standard dataset we made of clusters of English newswire sentences.

Chapter 4: In this chapter, we propose a query-focused abstractive summarization approach to summarize multiple news documents (ShafieiBavani et al., 2016d) through the following steps: (i) capturing semantic similarities between sentences and the input query; (ii) filtering less query-relevant sentences; (iii) clustering the query-relevant sentences; (iv) multi-sentence compression. The proposed approach makes use of the relations provided in WordNet to measure the semantic similarities. We additionally investigate the applicability of the MSC word graph proposed in Chapter 3 for this task. Finally, we report experiment results and compare this approach with state-of-the-art methods.

Chapter 5: This chapter presents the first attempt (ShafieiBavani et al., 2016a) to appraise the coverage of knowledge sources for domain-specific summarization. We provide the intuition behind this work and focus on biomedical domain peculiarities. We explain our approach in integrating the general and domain-specific lexicons for incorporating textual semantic similarities to summarize clinical texts. For this purpose, our approach is adapted to the proposed query-focused multi-document summarization framework in Chapter 4. Finally, we provide the experiment results achieved by comparing our work with the competitive baselines.

Chapter 6: This chapter provides insights into the shortcomings of existing evaluation metrics for text summarization. We introduce our new evaluation metric namely ROUGE-G (ShafieiBavani et al., 2018a, 2017), which investigates summaries considering their underlying semantics. The impact of capturing lexical and semantic similarities to compute ROUGE scores is also studied in this chapter. Finally, we conduct a set of experiments over benchmark summarization evaluation datasets to compare ROUGE-G variants with their corresponding variants of ROUGE.

Chapter 7: In this chapter, our proposed approach (ShafieiBavani et al., 2018b) incorporates the word embedding models trained on the Google News corpus and the WordNet lexical resource to compare centroid vectors of the input and summary. For this purpose, we design multiple features to train a learning model for predicting summary content quality in the absence of model summaries. To demonstrate the effectiveness of our approach, we have conducted a set of experiments on data from query-focused and update summarization tasks. The reliability of our metric is also studied conducting an error analysis.

Chapter 8: This chapter summarizes the contributions and outcomes of this thesis. We will also present possible future research directions, and finish the chapter by discussing untouched, but interesting topics in this research area.

Chapter 2

Literature Review

Numerous approaches for automatic text summarization have been developed to date. While (Das and Martins, 2007; Nenkova et al., 2011; Nenkova and McKeown, 2012; Gambhir and Gupta, 2017; Yao et al., 2017) provided comprehensive view of the field of automatic text summarization, significant progress has recently been made from traditional extractive level to more abstractive summarization. This chapter reviews recent progress made for automatic text summarization and the literature on abstractive text summarization.

First, in Section 2.1, we provide a detailed overview of automatic text summarization, its various types, the conventional framework, and the recent advances in this task which are mostly extractive. In Section 2.2, we move beyond sentence extraction, and introduce compressive summarization. In the same section, we review recent progress made toward fully and end-to-end abstractive summarization. We then discuss abstraction in different domains and genres. Model-based and model-free summarization evaluation are also explored in Section 2.3. Next, Section 2.4 surveys previous studies on developing applications for text summarization. Finally, the literature review discussed in this chapter is summarized in Section 2.5.

2.1 Automatic Text Summarization

Information overload is an increasing problem due to the fast growth of Internet content. Hence, need of text summarization has emerged to alleviate this problem. It is time consuming and exhausting to read through this large amount of texts. Moreover, many important and interesting documents might be skipped. Therefore, robust automatic text summarization systems are required. Text summarization is the process of automatically creating a compressed version of a given text while considering the following main objectives (Huang et al., 2010): (i) information coverage; (ii) information significance; (iii) information redundancy; and (iv) text cohesion.

Automatic text summarization was firstly introduced in the late 1950's (Luhn, 1958), and moved forward in the 1960s and 1970s (Edmundson, 1969; Skorokhod et al., 1972). In the late 1990s, this task attracted strong interest, which was reflected in text summarization competitions (e.g., DUC[1] and TAC[2]) organized annually since 2001, a textbook about text summarization by Mani (2001), an edited collection (Mani and Maybury, 1999) and a special issue of the Computational Linguistics journal (Radev et al., 2002).

2.1.1 Various Types of Text Summarization

Considering number of documents, single and multi-document summarizations are the two important categories of summarization (Zajic et al., 2008; Fattah and Ren, 2009). Summary is generated from a single document in *single-document summarization* whereas in *multi-document summarization*, many documents are used for

[1]Document Understanding Conference (DUC) (http://duc.nist.gov/) in the period 2001-2007

[2]Text Analysis Conference (TAC) since 2008 (http://www.nist.gov/tac/)

generating a summary. Generally, the task of multi-document summarization is more difficult than the task of single-document summarization. Redundancy is one of the biggest problems in summarizing multiple documents. Some approaches tackled this issue by selecting only the relevant new sentences while measuring their similarity to the rest of sentences (Sarkar, 2010). Maximal Marginal Relevance (MMR) is also another approach suggested by Carbonell and Goldstein (1998) to reduce redundancy. In (Tao et al., 2008; Wan, 2008; Wang et al., 2008a,b, 2009, 2011), different approaches have been proposed to present the best-performing multi-document summarizer.

Furthermore, a summary can either be *query-focused*[3] (biased to a user query) (Ouyang et al., 2011; Abdi et al., 2015; Sarker et al., 2016; ShafieiBavani et al., 2016a), or *generic* (conveying the document gist) (Gong and Liu, 2001; Wan, 2008). In later DUC/TAC evaluation tasks, query-focused multi-document summarization and guided summarization are starting to receive more attention. They differ from generic summarization because of a provided query sentence that describes the specific information need, and thereby guides the summarization process (Yao et al., 2017).

In another distinction, content reduction can be addressed by selection and/or by generalization of what is important in the source (Jones, 1999). Consequently, two common categories of *Extractive* and *Abstractive* are defined in the text summarization literature (as discussed in Section 1.1). Most text summarization systems produce summaries from extracted sentences. The general approach in extractive summarization is as follows: (i) ranking sentences (based on their importance) from a given set of related source texts; (ii) selecting the top-ranked ones to make a summary of a desired length - the selected important sentences should also be different

[3]Topic-focused or user-focused summaries are the other names for query-focused summaries.

enough to avoid redundancy in the summary (Carbonell and Goldstein, 1998); (iii) performing post-processing steps (sentence ordering, sentence compression or simplification) to improve the coherence of the produced summary. Naturally, sentences selected from different documents are unlikely to produce a coherent summary when combined together. This can be observed in the poor ratings of the linguistic quality in the DUC and TAC competitions (Filippova, 2010a). According to a series of psychological experiments, the way humans summarize is very different from the extractive strategy (Kintsch and Van Dijk, 1978). To be abstractive and more human-like, an automatic summarization system should interpret the input text, construct its (symbolic) representation, make necessary inferences and only then generate a summary from the representation (Jones, 1999). Text interpretation and generation on the level required for truly abstractive text summarization is not possible yet (Filippova, 2010a). Consequently, the absolute majority of existing text summarization systems are purely extractive (Jones, 2007).

Summarization task can also be of two types: *supervised* or *unsupervised* (Mani and Maybury, 1999; Fattah and Ren, 2009; Riedhammer et al., 2010). In supervised methods, training data is required to select important content from the source, and large amount of labeled or annotated data is required for learning. These methods are addressed at the sentence level as a two-class classification problem, where positive samples are sentences belonging to the summary, and negative samples are the ones not existing in the summary (Song et al., 2011; Chali and Hasan, 2012b). Some popular classification methods like Support Vector Machine (SVM) (Ouyang et al., 2011) and neural networks (Fattah and Ren, 2009) are usually utilized for sentence classification. In contrast, unsupervised methods do not require any training data. Hence, they are suitable for any newly observed data without any advanced modifications. They apply heuristic rules to extract highly relevant sentences and generate a summary using clustering techniques (Fattah and Ren, 2009).

Based on the style of output, there are also two types of *indicative* and *informative* summaries. Indicative summaries indicate what the source text is about. They give information about the topic of the text. Informative summaries give the whole information in elaborated form while covering the topics. One more type of summary is *critical evaluation abstracts*. These summaries contain opinions, reviews, recommendations, feedbacks, etc.. Considering the domain of source texts, a summarization approach can either be *generic* when it works for summarizing general texts like newswire, or *domain-specific* when it is limited to more specific domains like biomedical.

On the basis of language, there are three kinds of summaries: *multi-lingual*, *mono-lingual* and *cross-lingual* summaries. Currently, most summarization research settings are monolingual (Yao et al., 2017). In mono-lingual summarization, language of source and target text is the same. A few exceptions have tried to explore the multilingual summarization setting, in which the source text is in a number of languages like English, French, Persian and summary is also generated in these languages (Litvak and Last, 2013). Litvak and Last (2013) proposed an approach to train an extractive single-document text summarizer called MUltilingual Sentence Extractor (MUSE). Their approach uses a genetic algorithm to find the best linear combination of a rich set of language-independent sentence scoring metrics. The last related but different setting is cross-language summarization, where the source and target languages are different.

Another common type is *personalized summary*, which contains the specific information that the user desires. These systems need to keep an eye on the user's profile and select the important content for generating the summary. In *update summaries*, it is considered that users have the basic information about the topic and require only the current updates regarding the topic. Summarization is also required to help users better digest the large amounts of highly redundant opinions expressed

on the web (Ganesan et al., 2010). In *sentiment-based summaries* or *opinion mining*, opinions are initially detected and classified on the basis of subjectivity (whether the sentence is subjective or objective), and also on the basis of polarity (positive, negative or neutral) (Pang et al., 2008).

Although text summarization methods vary considerably, most of them share one important property: *they should produce a summary which is shorter and/or more informative than the source text.* Thereupon, this task is challenging due to the issues like redundancy, temporal dimension, co-reference, sentence ordering, etc.. In this thesis, we focus on proposing abstractive summarization approaches (Chapter 3) for query-focused multi-document summarization of newswire (Chapter 4) and biomedical (Chapter 5) domains.

2.1.2 The Conventional Framework

Earlier research in the last decade is dominated by extractive summarization approaches, with a few of them also including other sentence-level operations such as sentence compression or reordering as a post-processing step after sentence extraction (Yao et al., 2017). The conventional framework for this task can be explained with the following key steps:

- **Sentence Scoring:** Each sentence is assigned a score which indicates its importance. Summarization aims at preserving the most important information via extracting the most important sentences.
- **Sentence Selection:** The best combination of important sentences is selected to form a summary of the desired length. Content coherence and redundancy should be considered in this step.

- **Sentence Reformulation:** The extracted sentences should sometimes be modified or paraphrased to produce clear, coherent and concise summaries.

The distinctions among these steps are sometimes vague, as some of them are implicitly considered or integrated into the other components. In the following, we briefly discuss previous studies on these steps.

Sentence Scoring

Sentence scoring gives answer to the question *"which sentences are important enough to be selected as summary sentences?"*

To answer this question, earlier unsupervised approaches mostly rely on *frequency* and *centrality*. The assumption behind frequency-driven approaches is that the most important information appears more frequently in the text documents. For instance, the earlier probabilistic system namely SUMBASIC (Vanderwende et al., 2007) was driven by word probability estimation, assigning each sentence a weight equal to the average probability of the content words in the sentence. More powerful usages include log-likelihood ratio test for identifying topic signature words that are highly descriptive of the input texts (Lin and Hovy, 2000). In earlier coverage-based models, the important concepts were those with high document frequency (Gillick et al., 2008).

In some approaches, similar sentences to other sentences are considered as central with the assumption of their carrying the most central ideas of the source documents. This assumption forms the basis of graph-based summarization approaches, typically adapted from link analysis algorithms in network analysis. Both TextRank (Mihalcea and Tarau, 2004) and LexRank (Erkan and Radev, 2004) utilized the PageRank algorithm in a weighted graph of words or sentences, with edge weights

defined using literal or more semantic-driven similarities. In centroid-based summarization (Radev et al., 2004), a pseudo-sentence of the document called centroid is constructed, consisting of words with TF-IDF[4] scores above a predefined threshold. The score of each sentence is defined by summing the scores based on different features including cosine similarity of the sentence with the centroid.

Probabilistic topic models based on co-occurrences have also been exploited in summarization. For example, the HIERSUM model (Haghighi and Vanderwende, 2009) was presented based on hierarchical Latent Dirichlet Allocation (hLDA) to represent content specificity as a hierarchy of topic vocabulary distributions. A later work (Celikyilmaz and Hakkani-Tur, 2010) also utilized a hLDA-style model to devise a sentence-level probabilistic topic model and a hybrid learning algorithm for extracting salient features of sentences. All these approaches focus on selecting the most frequent information from the source text. However, this strategy does not work well in noisy texts with significant amounts of redundant and unimportant texts.

To date, extractive summarization benefits from various machine learning techniques, which learn to extract sentences. Given sentences with labeled importance scores, it is straightforward to train regression models for importance prediction (Galanis et al., 2012; Hong and Nenkova, 2014; Ouyang et al., 2011) or learning to rank models to train a model that is capable of assigning high rank to the most important sentences (Metzler and Kanungo, 2008; Shen and Li, 2011; Wang et al., 2013). To model possible inter-sentence dependency rather than predicting the importance score of each sentence individually, text summarization can also be treated as a sequence labeling problem, with latent labels indicating whether to extract the

[4]The TF-IDF weighting scheme is a well-known concept in information retrieval that uses the Term Frequency (TF) in the document for each term and a complementary weight for each term which penalizes terms found in many documents in the collection by using the Inverse Document Frequency (IDF), i.e., the inverse of the number of documents that contain the term, as weights.

sentence into the summary or not. As a result, hidden Markov models (Conroy and O'leary, 2001), Conditional Random Fields (CRF) (Shen et al., 2007) and structural SVMs (Li et al., 2009) have all been applied in such settings. All these approaches extract indicative features including sentence position, named entities, similarity or distance to query and content word frequency.

Supervised approaches rely on labeled training data. A typical way to construct labeled data for training is to set ROUGE - a 'de facto' standard automatic evaluation metric for summarization - or its variants and approximations as prediction target for sentence scoring. This treatment has become more theoretically clarified in a very recent study (Peyrard and Eckle-Kohler, 2016).

In query-focused summarization, similarity between the content of query and sentences in the source text is typically considered. These scores can either be used in similarity-based approaches or act as features for importance prediction (Ouyang et al., 2011). Supervised approaches have achieved more significant improvements for sentence scoring in query-focused summarization (Wang et al., 2013).

Sentence Selection

Considering the achieved importance scores in the previous step, a typical strategy for this step is directly selecting the high ranked sentences. However, redundancy should be removed specially in multi-document summarization to avoid considering the whole relevant sentences as important ones. One of the most popular approaches for sentence selection is MMR (Carbonell and Goldstein, 1998). It satisfies the relevant novelty criterion by measuring relevance and novelty independently and providing their linear combination as marginal relevance. In this context, a sentence has high marginal relevance if it is both relevant to the query and contains minimal similarity to previously selected sentences. It is defined as an objective function in

order to add sentence k to set S, ($k \notin S$):

$$\lambda Sim_1(s_k, q) - (1 - \lambda) \max_{i \in S} Sim_2(s_i, s_k) \tag{2.1}$$

where $Sim_1(s_k, q)$ measures the similarity between sentence s_k and a query q, and $Sim_2(s_i, s_k)$ measures the similarity between sentences s_i and s_k. Sim_2 can be the same as Sim_l or a different metric, and $\lambda \in [0, 1]$ is a trade-off coefficient.

In probabilistic approaches (Haghighi and Vanderwende, 2009; Vanderwende et al., 2007), sentences are selected in order to minimize the Kullback-Leibler Divergence (KLD) between the probability distributions of words in the summary and the input. Since finding the smallest KL divergence is computationally intractable, greedy selection is often used. Instead of greedily adding sentences to form a summary, sentence scoring and selection can also be formulated as global optimization (e.g., Integer Linear Programming (ILP)) (Gillick et al., 2008; McDonald, 2007). The objective is usually to maximize coverage of consistency constraints between the selected sentences and sub-sentential units, and a knapsack constraint to limit the summary total length. For example, the aim of the classic concept-based ILP approaches for summarization (Gillick et al., 2008), is to maximize the total weights of the concepts (implemented as bigrams) included in the summary. The assumption here is that frequently appeared bigrams will mostly contain important concepts. However, this strategy is not useful in the face of significant amounts of redundant and unimportant texts. As a remedy, supervised learning may better predict which sentences are more important and should be kept in the final summary. The relation between the concepts and sentences serves as the constraints. This ILP framework

is formally represented as:

$$
\begin{aligned}
& max \sum_i w_i c_i \\
& \text{subject to, } \textstyle\sum_j l_j s_j < L, \\
& \sum_j s_j o_{ij} \geq c_i, \forall i, \\
& s_j o_{ij} \leq c_i, \forall i, j, \\
& c_i \in \{0, 1\}, \forall i, \\
& s_j \in \{0, 1\}, \forall j
\end{aligned}
\tag{2.2}
$$

where c_i and s_j are binary variables that respectively indicate the presence of a concept and a sentence in the summary. w_i is the weight of concept i, and o_{ij} means the occurrence of concept i in sentence j. A concept can be selected only if it is present in at least one selected sentence and a sentence can be selected only if all concepts it contains are selected.

Sentence Reformulation

Most of the previous summarizers extract sentences and just leave them as they are. Systems targeting more practical usages also include additional operations as an additional step following sentence selection.

Sentences extracted from source documents usually contain unnecessary or redundant information, which makes them less suitable to be directly used as summary sentences. A popular solution is to pipeline sentence extraction and rule-based compression. More sophisticated operations may also be used to enhance compactness and informativeness, like paraphrasing and sentence fusion (Barzilay and McKeown,

2005). Due to the immatureness of current natural language generation techniques, some of these operations may hurt readability of the final summary. As a result, very few progress in terms of sentence rewriting has been made in fully abstractive summarization in earlier work.

Meanwhile, the order in which information is presented to the reader critically influences the quality of a summary. In a single document, summary information can be presented by preserving the order in the source document (Radev et al., 2004). However, extracted sentences do not always retain their precedence orders in manually written summaries. Reordering is a more significant issue for multi-document summarization as summary sentences are from multiple unaligned sources. Classic reordering approaches include inferring order from weighted sentence graph (Cohen et al., 1998) or perform a chronological ordering algorithm (Barzilay and Elhadad, 2002) that sorts sentences based on timestamp and position.

2.1.3 Recent Advances

Many studies have tried to improve automatic text summarization from the aspects that have not been explicitly considered in traditional approaches. For instance, some studies focused on extracting more certain sentences (Wan and Zhang, 2014) or utilizing timeline information in order to enhance summarization (Ng et al., 2014). Some others focused on integrating the power of different summarization approaches, and promoting weighted consensus (Wang and Li, 2012), directly performing supervised aggregation (Pei et al., 2012), or re-ranking outputs from different baseline approaches (Hong et al., 2015).

There are also a few recent studies that focused on improving graph-based summarization. Li and Li (2014) integrated topic models into graph ranking to utilize relations between topics and sentences. Parveen and Strube (2015) employed a bi-

partite graph connecting sentences and topics to represent a document and applied the HITS algorithm to calculate importance. Graph-based topical coherence can be naturally introduced in graph-based frameworks. Coherence scores can be derived from node degrees in sentence-entity bipartite graphs to be integrated in an ILP objective function (Parveen et al., 2015). Meanwhile, using rich syntactic/semantic information to derive frequent sub-patterns for similarity calculations may also improve the performance of graph ranking models (Yan and Wan, 2014).

Semi-supervised machine learning approaches, such as using reinforcement learning (Rioux et al., 2014) or learning to search (Kedzie et al., 2016), have recently been adapted to summarization tasks. They showed great potential by defining proper reward functions. Such approaches can directly utilize relevant metrics like ROUGE during training for defining proper reward signals.

Representation learning based on neural networks with multiple layers has recently made significant progress in natural language processing. There is a lot of work that tries to model summarization tasks in neural network architectures, with fewer or no dependence on handcrafted features. Neural network approaches for automatic text summarization are mostly playing partial roles. For example, they act as a component like sentence scoring in traditional extractive frameworks. Deep Boltzmann machines have been adapted for text summarization to learn hierarchical concept representations, and to predict concept importance and select sentences accordingly (Liu et al., 2012b). In a few studies (Kobayashi et al., 2015; Kågebäck et al., 2014), similarity was measured based on distributed representations, using the sum of trained word embeddings to represent sentences or documents. Convolutional Neural Network (CNN) architectures have been designed for sentence modeling and selection (Cao et al., 2015b; Yin and Pei, 2015), and used as sentence scoring components for extractive summarization. A later work (Cao et al., 2016b) also used convolutional sentence embeddings to model sentence-level attentive be-

haviors. They utilized a layered neural network to learn a ranking model based on both query relevance and sentence salience. Sentence ranking models can also be trained using recursive neural networks, formulating scoring as a hierarchical bottom-up regression (Cao et al., 2015a). It has recently been shown effective to use even the simplest form of neural network (i.e., generic multilayer perceptron) to directly predict the relative importance of a sentence, given a set of selected sentences, considering both importance and redundancy (Ren et al., 2016).

Meanwhile, a couple of unsupervised approaches have been proposed. They have mostly been outperformed by supervised approaches, even with the small amount of available training data. Zhang et al. (2015) utilized the density peaks clustering algorithm (Rodriguez and Laio, 2014) for scoring representativeness and diversity, yielding relatively strong ROUGE results as an unsupervised framework. Currently, the OCCAMS system (Davis et al., 2012) gives empirically the best performance in unsupervised approaches on standard DUC datasets. It first derives the term weights via Latent Semantic Analysis (LSA) and then selects sentences that cover the maximum combined weights. Another recently explored idea is data reconstruction (He et al., 2012), based on an assumption that a good summary may consist of sentences that can best reconstruct the source text. This idea has been extended in numbers of follow-up studies (Li et al., 2015c; Liu et al., 2015b; Ma et al., 2016; Yao et al., 2015). However, they failed to achieve convincing performance according to their experiment results on standard multi-document DUC datasets. The reported results are inferior to OCCAMS and far less comparable to the state-of-the-art supervised approaches. Data reconstruction approaches encourage summaries to cover information as much as possible, while in practice good summaries should only cover a small portion of the source text.

2.2 Beyond Sentence Extraction

One of the main issues that most of the extractive approaches suffer from is *information redundancy*. Furthermore, there still exists a huge gap between system-generated summaries and human-written summaries. For single-document summarization in particular, the well-known Lead baseline (i.e., extracting the first sentences of a document), has already been close to the 99% percentile of the ROUGE score distribution over all possible extractive summaries for newswire and scientific domains (Ceylan et al., 2010). This shows that it is difficult to significantly improve over the Lead system on standard benchmarks (e.g., see standard DUC/TAC evaluations). Similar percentile ranks have also been observed for the TextRank system (Mihalcea and Tarau, 2004). These results may suggest that making further improvements based on only sentence extraction will be considerably difficult.

Abstractive summarization is generally considered to be much more difficult, involving sophisticated techniques for meaning representation, content organization, surface realization, etc.. Hence, there has been a surge of interest in recent years on compressive text summarization that tries to compress source texts to form a summary, as an intermediate step toward abstractive summarization.

2.2.1 Compressive Summarization

Compressive summarization receives increasing attention in recent years, since it offers a viable step toward abstractive text summarization. This type of summarization removes less important sentence components and makes room for more salient information. Hence, compressive summaries often contain more information than summaries resulted by sentence extraction. Two general strategies have been used for compressive summarization: (i) *pipelining*, where sentence extraction is fol-

lowed by sentence compression[5] (Lin, 2003; Wang et al., 2016; Zajic et al., 2006); and (ii) *joint compression and summarization*, which has been shown to achieve promising performance but is computationally much more expensive. Chali and Hasan (2012a) have studied the effectiveness of sentence compression under the ILP framework for query-focused summarization. They utilized a comprehensive set of query-related, importance-oriented, and various sentence similarity measures to define the relevance and redundancy constraints. Their experiment results showed that performing joint compression and extraction via optimizing a combined objective function outperforms pipeline approaches.

Sentence compression in an unsupervised fashion was traditionally performed based on frequency-driven scores and tree-trimming rules, or in a supervised fashion using external sentence compression datasets. Some approaches use summarization data for training compression models. Li et al. (2013) utilized data annotated based on word importance derived from manually written summaries to train a CRF model for sentence compression. They showed that including sentences with such guided compression in ILP models improves over including sentences with generic compression. For sentence compression based on trimming constituent trees, the reference label for every node in the tree can also be obtained automatically from the bottom to the top of the tree (Li et al., 2014). In a pipeline framework where sentences are firstly compressed via trimming expanded constituent trees using the learned model, the system achieves similar ROUGE scores but better linguistic quality on TAC data.

Some other approaches combine multiple scoring models with the guidance of summarization data. Wang et al. (2013) explored the role of supervised sentence compression approaches to query-focused multi-document summarization. The compres-

[5]Sentence compression aims at producing a condensed and grammatically correct sentence.

sion scoring function in their approach is constructed to incorporate scores from their proposed tree-based compression, query relevance, significance and redundancy, by tunning combination weights on development data. Their system shows statistically significant improvements over pure extraction-based approaches. They achieved the state-of-the-art results on query-focused DUC datasets (DUC 2006 and 2007), in terms of both ROUGE and pyramid scores, along with reasonably good manual evaluation scores.

Currently, the most popular supervised approach to compressive summarization is to perform multi-task learning or jointly learn an extraction model and a compression model in the same framework. Berg-Kirkpatrick et al. (2011) proposed an approach to score the candidate summaries according to a combined linear model of extractive sentence selection and compression. They trained a model using a margin-based objective to capture the final summary quality. Since the search space is very large, they initially performed sentence filtering to reduce the number of candidates for more practical approximation.

By growing the scale of problem in joint extraction and compression settings, various alternatives to ILP have been studied. A recently proposed framework has enabled independent decoding for compression while dealing with knapsack constraint separately, based on alternating direction dual decomposition (Almeida and Martins, 2013). The authors proposed multi-task learning to train compressive summarizers, using auxiliary data for extractive summarization and sentence compression. Their framework achieved high ROUGE scores while consuming running time as short as extractive systems. Another approximate inference strategy is to cast the original ILP into graph cuts (Qian and Liu, 2013). The authors modified the objective function with super-modular binary quadratic functions to eliminate subtree deletion constraints and relax the length constraint using Lagrangian relaxation. The relaxed objective function is bounded by the super-modular binary quadratic programming

problem which can be approximately solved using graph max-flow/min-cut. Morita et al. (2013) tried to produce compressive summarization by extracting a set of dependency subtrees in the document cluster, under the budgeted submodularity framework, with dependency constraints to guarantee readability. They proposed an efficient greedy algorithm for approximate inference with performance guarantee, calling a dynamic programming procedure for subtree extraction.

Compressive summarization for a single document can integrate discourse-level compression, which may lead to more coherent compressed sentences. A natural way is to consider both the syntactic dependency tree for words and discourse dependency tree between sentences (rhetorical structures) as a nested tree structure. This nested tree-trimming problem can then be formulated as combinatorial optimization (Kikuchi et al., 2014) to generate compressive summaries using ILP or dynamic programming procedure (Nishino et al., 2015). Durrett et al. (2016) tried to combine discourse-level compression based on the Rhetorical Structure Theory (RST) tree and syntactic compression based on constituent trees.

Applicability of sentence compression to multi-document compressive summarization has motivated some approaches to use Multi-sentence Compression (MSC) for this task. MSC refers to the method of mapping a collection of related sentences to a sentence shorter than the average length of the input sentences, while retaining the most important information that conveys the gist of the content, and still remaining grammatically correct (Jing, 2000). This idea was introduced by Barzilay and McKeown (2005), who developed a multi-document summarizer which represents each sentence as a dependency tree. Their approach aligns and combines these trees for sentence fusion.

State-of-the-art approaches that utilized MSC for compressive summarization are generally divided into supervised (McDonald, 2006; Galley and McKeown, 2007) and unsupervised groups (Clarke and Lapata, 2007). MSC methods traditionally

use syntactic parsers to generate grammatical compressions, and fall into two categories (based on their implementations): (i) *tree-based* approaches, which create a compressed sentence by making edits to the syntactic tree of the original sentence (McDonald, 2006; Galley and McKeown, 2007; Filippova and Strube, 2008; Elsner and Santhanam, 2011); (ii) *sentence-based* approaches, which generates strings directly (Clarke and Lapata, 2007).

As an alternative, word graph-based approaches that only require a POS tagger have recently been used in different tasks, such as guided microblog summarization (Sharifi et al., 2010b), opinion summarization (Ganesan et al., 2010) and newswire summarization (Filippova, 2010b; Boudin and Morin, 2013; Tzouridis et al., 2014). In these approaches, a directed word graph is constructed in which nodes represent words while edges between two nodes represent adjacency relations between words in a sentence. Hence, the task of sentence compression is performed by finding the k-shortest paths in the word graph. In this regard, Filippova (2010b) has introduced an elegant word graph-based multi-sentence compression approach that relies on the redundancy among the set of related sentences.

Several studies have used their simple and effective approach as the first step to generate a list of the N shortest paths (Boudin and Morin, 2013; Tzouridis et al., 2014). They have relied on different re-ranking strategies to analyze the candidates and select the best compression. Another work (Tzouridis et al., 2014) proposed a structured learning-based approach. Instead of applying heuristics as Filippova (2010b), they adapted the decoding process to the data by parameterizing a shortest path algorithm. They devised a structural SVM to learn the shortest path in possibly high dimensional joint feature spaces and proposed a generalized, loss-augmented decoding algorithm that is solved exactly by ILP in polynomial time. As reported by Boudin and Morin (2013), some important information is missed from 48% to 60% of the generated sentences in the approach by Filippova (2010b). Thereupon,

they proposed an additional re-ranking scheme to identify compressions that contain keyphrases using the TextRank algorithm (Mihalcea and Tarau, 2004). However, they reported that grammaticality might be sacrificed to improve informativity in their work.

In this thesis (Chapter 3), we utilize MultiWord Expressions (MWE) and synonym words in sentences to significantly enhance the traditional word graph, and improve informativity (ShafieiBavani et al., 2016b,c). Components of an MWE are treated as a single unit to improve the effectiveness of re-ranking steps in Information Retrieval (IR) systems (Acosta et al., 2011). Herein, we identify MWEs, merge their components, and replace them with their available *one*-word synonyms, if applicable. These strategies help to construct an improved word graph and enhance the informativity of the compression candidates. Then, we re-rank the generated compression candidates with TextRank algorithm (Mihalcea and Tarau, 2004) to favor the compressions containing important keyphrases. We also trained a 7-gram POS-based Language Model (POS-LM) that selects the most grammatical compression candidates to improve the grammaticality. POS-LMs were traditionally used for speech recognition problems (Heeman, 1998) and statistical machine translation systems (Koehn et al., 2008; Monz, 2011; Popović, 2012) to capture syntactic information.

In all, compressive systems are currently producing competitive results with syntactic and discourse constraints directly guiding the results toward being concise and coherent. They are achieving a good trade-off between content compactness and readability (Yao et al., 2017).

2.2.2 Toward Full Abstraction

Fully abstractive summarization aims to understand the input and generate the summary from scratch, usually including paraphrasing and lexical variations. This field

of research involves multiple subproblems including: simplification, paraphrasing, merging or fusion. Due to the inherent difficulty and complexity of full abstraction, current research in abstractive text summarization is mostly restricted to one or a few of the subproblems (Yao et al., 2017). It is also less active compared with compressive summarization where we have a boost in system performance by merely considering compressions.

Woodsend and Lapata (2012) proposed a model for multi-document summarization that attempts to cover many different aspects of the task such as content selection, surface realization, paraphrasing, and stylistic conventions. These aspects are learned separately using specific predictors, but are optimized jointly using an ILP to generate the output summary. For document summarization that involves paraphrasing and fusing multiple sentences simultaneously, other than grammar-based rewriting, a typical approach is to merge information contained in sub-sentence-level units. For instance, one can cluster sentences, build word graphs and generate (shortest) paths from each cluster to produce candidates for making up a summary (Banerjee et al., 2015; Filippova, 2010b). More sophisticated treatments can also be built on syntactic or semantic analysis. One may build sentences via merging consistent noun and verb phrases (Bing et al., 2015) or linearizing graph-based semantic units derived from semantic formalisms such as Abstract Meaning Representation (AMR) (Liu et al., 2015a).

There also exist psychologically motivated studies (Fang and Teufel, 2014) trying to implement cognitive human comprehension models based on propositions extracted from a source sentence. Their model gets around the problem of identifying concepts in text by applying co-reference resolution, Named Entity Recognition (NER), and semantic similarity detection, implemented as a two-step competition. The current systems have mostly been evaluated on over-specific datasets and heavily relied on various components including parsing, co-reference resolution, distributional seman-

tics, lexical chains (Fang and Teufel, 2016) and natural language generation from semantic graphs (Fang et al., 2016).

In order to better guide alignment and merging processes, supervised learning-based approaches have been investigated (Elsner and Santhanam, 2011; Thadani and McKeown, 2013). A later work (Cheung and Penn, 2014) expanded the sentence fusion process with external resources beyond the input sentences. They presented *sentence enhancement* as a novel technique for text-to-text generation in abstractive summarization. Sentence enhancement increases the range of possible summary sentences by allowing the combination of dependency subtrees of any sentence from the source text.

Abstractive summarization has also been studied in information extraction perspective. For example, IE-informed metrics are useful to re-rank the output of high performing baseline summarization systems (Ji et al., 2013). In the context of guided summarization, preliminary full abstraction has been achieved by extracting templates using predefined rules for various types of events (Genest and Lapalme, 2012; Saggion, 2013).

A large part of existing work in abstractive summary generation is actually limited to more specific domains, where fixed templates or rules are manually crafted for generating the sentences. For example, Ganesan et al. (2010) proposed an abstract summarization approach for product reviews, where graph-based algorithms can be designed to merge reviews with similar textual content. Sentence realization templates were utilized to ensure grammaticality (Gerani et al., 2014). Meanwhile, instead of generating a summary consist of multiple sentences, Alfonseca et al. (2013) and Pighin et al. (2014) focused on generating a headline sentence for a news article. The authors first clustered or learned the event templates from a large number of news articles and then filled the entities into appropriate templates to form the headline. Headline generation has also become a test bed for modern neural abstractive

generation.

Existing research on query-focused multi-document summarization largely relies on extractive approaches, where systems usually take a set of documents as input and select the top relevant sentences for inclusion in the final summary (Wang et al., 2016). A wide range of methods have been employed for this task. In unsupervised methods, sentence importance can be estimated by calculating topic signature words (Conroy et al., 2006; Lin and Hovy, 2000), and combining query similarity and document centrality within a graph-based model (Otterbacher et al., 2005). In (Davis et al., 2012), term weights are learned by LSA, and sentences that cover the maximum combined weights are selected by a greedy algorithm. Supervised approaches have mainly focused on applying discriminative learning for ranking sentences (Fuentes et al., 2007). For instance, the work proposed by Lin and Bilmes (2011) used a class of submodular functions to reward the diversity of the summaries and select sentences greedily.

In this thesis (Chapter 4), we propose an unsupervised approach (ShafieiBavani et al., 2016d), to query-focused multi-document summarization. To our knowledge, this is the first time that abstraction finds its way to this task. Our approach provides an abstractive summary of a set of news documents with respect to a user query. In the field of query-sensitive summarization, qualified summary sentences should mainly meet the following typical requirements: *query-biased relevance* (Otterbacher et al., 2005; Shen and Li, 2011), *biased information novelty*, and *biased information richness* (Wan et al., 2007). Our proposed summarization approach is also organized to cover these three criteria. To be query-biased relevant, summary sentences must overlap with the query in terms of topical content. Query-biased information novelty denotes that summary sentences need to be unique, as well as responding to the demands of the query. Finally, to acquire query-biased information richness, summary sentences should include as much important information as

possible with respect to both the set of sentences and the query. In the following, we briefly introduce some related summarization methods that explicitly take these requirements into consideration.

In the approach presented in (Otterbacher et al., 2005), query-biased sentence relevance is acquired by opting for the trade-off of the sentence's initial relevance to the query and its similarities to other sentences in the document cluster. In another work (Li et al., 2009), summarization is treated as a supervised sentence ranking process, where coverage, balance and novelty properties are incorporated. However, its focus is on generic summarization rather than query-biased. Explicit definitions of biased information richness and novelty are given by the approach proposed by Wan et al. (2007). Meanwhile, a manifold-ranking process is proposed to compute biased information richness, and a greedy algorithm similar to (Zhang et al., 2005) is also applied to reduce information redundancy in the summaries.

Recently, the method presented in (Yin et al., 2012) (RelationListwise) has considered all three properties of novelty, coverage and balance. This approach integrated sentence relation information with list-wise learning to rank and automatically learn feature weights. RelationListwise outperforms all the aforementioned approaches. However, a common characteristic of existing methods in acquiring novelty or balance properties lies in the area of extractive summarization. Our proposed approach effectively satisfies these requirements through an abstractive summarization framework, which results in high-quality summaries, by involving underlying disambiguated textual semantic similarities. We also meet a fourth criterion (i.e., grammaticality), which ensures the grammatical structure of newly generated summary sentences.

2.2.3 Toward End-to-end Abstractive Summarization

Recently, end-to-end training with encoder-decoder neural networks (Sutskever et al., 2014) have achieved huge success in machine translation systems, which brings potential applications for abstractive text summarization. In these approaches, source texts are encoded in an encoder network and then passed to a decoder network to produce the desired output. This architecture is typically implemented using basic building blocks such as Recurrent Neural Networks (RNN) with gated units and attention weighting. Hence, they can be adapted to different sequence-to-sequence tasks like machine translation and text rewriting.

The inputs are typically just raw texts, making the whole system free from heavy manual feature engineering. Figure 2.1 (Sutskever et al., 2014) depicts an instance model based on two Long-Short-Term Memory (LSTM) (Hochreiter and Schmidhuber, 1997) recurrent networks as encoder and decoder, used for rewriting the input text into a more concise output. This model reads an input sentence ABC and produces WXYZ as the output sentence. The model stops making predictions after outputting the end-of-sentence token. The LSTM also reads the input sentence in reverse, because doing so introduces many short term dependencies in the data that make the optimization problem much easier.

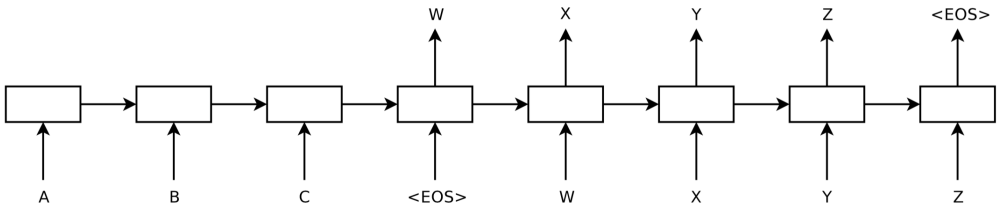


Figure 2.1: Example of sequence-to-sequence learning with neural networks

This line of research was started by Rush et al. (2015) under the term *sentence summarization*, and misleadingly called *text summarization* in some follow-up work. However, it is in fact a *sentence simplification* approach working on short text inputs

such as microblogs, tweets or single sentences. Therefore the applications are mainly in microblog summarization, sentence simplification and headline generation.

Relevant advances typically contribute to improve sequence-to-sequence learning and attention-based RNN encoder-decoder structures (Chopra et al., 2016; Nallapati et al., 2016). For example, since many words in a simplified sentence are retained from the original input sentence, a copying mechanism has been shown to be useful (Gu et al., 2016; Gulcehre et al., 2016; Nallapati et al., 2016; See et al., 2017). This strategy allows a word to be generated by directly copying an input word rather than producing from the hidden state. However, direct optimization of ROUGE via reinforcement learning has been shown to be more effective than likelihood optimization for the decoder generation (Ayana et al., 2016; Ranzato et al., 2015). For the task of sentence simplification, there usually exists a predefined length constraint. Since it is difficult to pose hard constraints on decoder generation, a recent work by Kikuchi et al. (2016) proposed a couple of methods for controlling the output sequence length for neural encoder-decoder models. These solutions are including direct truncation of the generated sequence, discarding out-of-range generations in the decoding beam, and directly embedding length information in the LSTM units.

Unfortunately, it is still a long way to go to adapt such architectures to text summarization (Yao et al., 2017). Encoding for generic texts still lacks satisfactory solutions. This hampers the applicability of sequence-to-sequence approaches. Currently, there are few approaches to generic text summarization under end-to-end neural architectures. To challenge the problem of longer inputs, hierarchical encoding and multiple levels of attention have been developed (Cheng and Lapata, 2016; Nallapati et al., 2016). However, recent proposals of architectural designs have not yet achieved competitive performance for fully abstractive summarization.

Another less noticed drawback in the field of neural sentence simplification is evaluating the quality and performance with the ROUGE metrics. However, there exists

no proof that the quality of simplification correlates well with ROUGE on sentence-level output (Yao et al., 2017). A research by Toutanova et al. (2016) introduced a manually created multi-reference dataset for abstractive compression of sentences or short paragraphs. Empirical evaluation on the dataset shows the importance of multiple references as well as suitable units in order to make automatic metrics more reliable. In a very recent study by Cao et al. (2018), a sequence-to-sequence framework to conduct template re-ranking and template-aware summary generation simultaneously has been proposed. They improved the readability and stability of sequence-to-sequence summarization systems. Nevertheless, sequence-to-sequence frameworks have been shown to be effective for some specific genres with short outputs like generating abstracts for opinions and arguments (Wang and Ling, 2016).

2.2.4 Toward Abstraction in Specific Domains and Genres

Traditional summarization approaches are mostly generic, which is based on standard benchmarks collected from news data. However, there exist various types of different tasks, domains and genres that need summarization. For example, microblog data may come in massively large scale, consisting multiple items that repeatedly and redundantly describe the same event. Texts are usually informal and contain huge amount of noise. Information might be time-variant, while user needs are diverse. Extractive approaches are predominant on tweet summarization. These approaches were firstly used for streams following simple and structured events like sports and games (Chakrabarti and Punera, 2011; Nichols et al., 2012; Takamura et al., 2011). In particular, Chakrabarti and Punera (2011) utilized temporal structural properties by designing modified hidden Markov models to automatically learn differences in language models of sub-events. Date selection is also important in timeline summarization (Tran et al., 2015).

More abstractive studies have started from the Phrase Reinforcement algorithm (Sharifi et al., 2010a) which extracts frequently used sequences of words and was firstly applied in summarizing topic streams. Subsequent research has focused on improving word graphs using dependency parses (Judd and Kalita, 2013), sequential summarization over evolving topics (Gao et al., 2013) or having online stream data as input (Olariu, 2014). Due to the specific properties of microblogs, personalization and social context can also be introduced in the model to enhance the performance of twitter summarization (Hu et al., 2012; Li and Cardie, 2014; Liu et al., 2012a; Yang et al., 2011) or leverage both social factors and content quality (Duan et al., 2012; Zhao et al., 2013). There also exists research that studied summarizing the repost structures of popular tweets (Li et al., 2015a), leveraging both the content of repost messages and different reposting relations between followers. A related task is indicative tweet generation with the aim of generating indicative tweets that contain a link to an external web page. There has been some work within extractive frameworks (Lloret and Palomar, 2013). However, it has recently been shown that word extraction is rather limited for this task (Sidhaye and Cheung, 2015).

Summarizing spoken data or transcripts poses the extreme challenge of noise and redundancy. In addition to information coverage, special treatments are required to extend beyond utterance extraction. For meetings (Oya et al., 2014; Wang and Cardie, 2013) and conversations (Trione et al., 2016), more compact and more abstractive generations are needed. However, unlike generic summarization, they usually have relatively fixed patterns and procedures. This makes template extraction and information fusion slightly easier and more feasible. Conventional frameworks for this task consist of template extraction from the training set and template filling.

Opinion summarization is the task of producing a summary while preserving the sentiment of the text. Therefore, there is a trade-off between summarization and opinion mining or sentiment analysis. Submodular functions or modifications can

be developed to address the conflicting requirements, balancing the coverage of both topics and polarities (Jayanth et al., 2015; Wang et al., 2014). Product review summarization can also be implemented via ILP, based on phrase selection. This can optimize both popularity and descriptiveness of phrases (Yu et al., 2016). Additional information for reviews such as review helpfulness ratings has also been proven to be useful to guide review summarization (Xiong and Litman, 2014).

Meanwhile, abstractive approach has been shown to be more appropriate for summarizing evaluative text (Carenini et al., 2013; Di Fabbrizio et al., 2014). In particular, a graph-based method (Ganesan et al., 2010) has been explored to produce ultra-concise opinion summaries. To improve fluency for abstraction, Carenini et al. (2013) tried to generate well-formed grammatical abstracts that explain the distribution of opinions over the entity and its features. Di Fabbrizio et al. (2014) proposed a hybrid abstractive/extractive sentiment summarizer to select salient quotes from the input reviews and embed them into the abstractive summary to provide evidence for the aggregate positive or negative opinions. End-to-end encoder-decoder RNNs have also shown effectiveness in producing short, abstractive summaries for opinions (Wang and Ling, 2016). For longer reviews, it is feasible to perform discourse parsing and aggregate discourse units in a graph. Review summarization can then sequentially perform subgraph selection and template-based generation (Gerani et al., 2014).

Recent research focus has drifted to domain-specific summarization techniques that utilize the available knowledge specific to the domain of text. For example, automatic summarization research on medical text generally attempts to utilize the various sources of codified medical knowledge and ontologies (Sarker et al., 2013). Among the research work in this area, Marshall et al. (2015) proposed an approach to automatically assess the risk of bias for clinical trials, and extracted specific study characteristics from trial abstracts (Summerscales, 2013). Many studies have

explored the obstacles associated with evidence-based medicine practice in the absence of pre-existing systematic reviews (Coumou and Meijman, 2006). When primary care physicians seek answers to clinical problems, the time required to search, evaluate, and synthesize evidence has been known as a critical factor (Sarker et al., 2016). Literature review and analysis may take a long time. For example, it takes more than 30 minutes on average for a practitioner to find and extract evidence (Hersh et al., 2002).

Numerous IR approaches have already been proposed to address the search-related needs of practitioners (Hanbury, 2012). However, post-retrieval techniques (Sarker et al., 2016) to perform query-oriented summarization are still scarce. The complicated nature of biomedical texts and the limited amount of suitable annotated data for the task of summarization are the main reasons that raise various difficulties in progress (Athenikos and Han, 2010; Sarker et al., 2016). To overcome the lack of incorporation of specific information, Unified Medical Language System (UMLS) came to play, and has proven to be a useful knowledge source for summarization in biomedical domain (Reeve et al., 2007). However, a decline is found in the performance of the summarizers that only utilize UMLS as their source of knowledge. The reason is that UMLS is less likely to cover all concepts included in the source text (Plaza et al., 2011).

To compensate for this deficiency, a question-oriented extractive system for biomedical multi-document summarization (Shi et al., 2007), utilized WordNet as a general-purpose lexicon to capture the concepts not covered by UMLS. They constructed a graph containing ontological concepts (general ones from WordNet, and specific ones from UMLS), named entities, and noun phrases. Our work (ShafieiBavani et al., 2016a) explained in Chapter 5 differs in intent, and explores the utility of graph representations of both WordNet and UMLS lexicons for incorporating underlying textual semantic similarities as the main basis of an abstractive biomedical summa-

rizer. Analysis via ROUGE metrics shows that using WordNet as a general purpose lexicon helps to capture the concepts not covered by the UMLS Metathesaurus, and hence significantly increases the summarization performance.

Besides the aforementioned studies, there also exists research on summarizing scientific articles (Cohan and Goharian, 2015), emails (Loza et al., 2014), community question answering (Chan et al., 2012), student responses (Luo and Litman, 2015), movie scripts (Gorinski and Lapata, 2015), entity descriptions in knowledge graphs (Cheng et al., 2015) and source codes descriptions (Iyer et al., 2016). Different scenarios pose variously different requirements and objectives on summarization systems.

2.3 Automatic Evaluation of Text Summarization

Assessing the quality of summaries is an important and necessary task in the field of automatic text summarization. Traditionally, this task involves a human assessment of various quality criteria (e.g., coherence, conciseness, grammaticality, informativity and readability) (Mani, 2001). Therefore, manual evaluation requires a lot of time and expertise in the field of given texts. To tackle this issue, automatic evaluation metrics come into play. This advent opens a new door to meta-evaluation (i.e., evaluation of evaluation metrics (Ellouze et al., 2013)). On the importance of meta-evaluation and its impact on summarization research, Text Analysis Conference (TAC[6]) provides the task of Automatically Evaluating Summaries of Peers (AESOP) to assess the correlation of evaluation metrics with human judgments.

Jones and Galliers (1996) introduced two types of evaluation methods: (i) *intrinsic*; and (ii) *extrinsic*. Intrinsic evaluation assesses the coherence and the informativeness

[6]http://www.nist.gov/tac/

of a summary, whereas extrinsic evaluation assesses the utility of summaries in a given application context like relevance assessment, reading comprehension, etc.. Most of the metrics proposed in the literature focused on intrinsic evaluation (Lloret et al., 2018). In intrinsic evaluation of automatic summarization, it is common to distinguish reference or *model* summaries from system-generated or *peer* summaries. Model summaries are those summaries that will be considered correct, and normally refer to those summaries generated manually by humans. Peer summaries are the summaries to be evaluated that usually have been automatically produced.

Among the metrics proposed for the task of summarization evaluation (Hovy et al., 2006; Tratz and Hovy, 2008; Giannakopoulos et al., 2008), Recall-Oriented Understudy for Gisting Evaluation (ROUGE) metric (Lin, 2004) has frequently been proven to correlate very well with human judgments (Lin and Och, 2004; Owczarzak and Dang, 2011; Over and Yen, 2004). ROUGE includes a large number of distinct variants, including four methods of n-gram counting (ROUGE-N; S; W; L). These metrics work based on the comparison of n-grams between the peer summary and human-written model summaries. The most commonly used ROUGE-N is an n-gram-based metric with the recall-oriented score, the precision-oriented score and the F-measure score as follows:

$$\text{ROUGE-N}_{recall} = \frac{\sum_{S \in \{ModelSummaries\}} \sum_{gram_n \in S} Count_{match}(gram_n)}{\sum_{S \in \{ModelSummaries\}} \sum_{gram_n \in S} Count(gram_n)} \tag{2.3}$$

$$\text{ROUGE-N}_{precision} = \frac{\sum_{S \in \{ModelSummaries\}} \sum_{gram_n \in S} Count_{match}(gram_n)}{\sum_{S \in \{PeerSummaries\}} \sum_{gram_n \in S} Count(gram_n)} \tag{2.4}$$

$$\text{ROUGE-N}_{F-measure} = 2 \times \frac{\text{ROUGE-N}_{recall} \times \text{ROUGE-N}_{precision}}{\text{ROUGE-N}_{precision} + \text{ROUGE-N}_{recall}} \tag{2.5}$$

In the summarization literature, a few of these variants are often chosen arbitrarily to assess the quality of summarization approaches. ROUGE-1, ROUGE-2, and ROUGE-SU4 are reported to have a strong correlation with human assessments, and are frequently used to evaluate summaries (Lin and Och, 2004; Owczarzak and Dang, 2011; Over and Yen, 2004). ROUGE-1 and 2, respectively calculate unigram and bigram co-occurrence statistics. ROUGE-SU4 measures co-occurring bigrams with maximum skip distance 4. It is noteworthy that ROUGE-2 and SU4 have been defined as baseline systems in TAC summarization tasks. Although ROUGE is a popular evaluation metric, improving the current evaluation metrics is still an open research area.

Other commonly used evaluation metrics also exist. Many of these efforts are analyzed and gathered in the surveys provided by (Steinberger and Ježek, 2012; Yao et al., 2017; Lloret et al., 2018). Herein, we try to briefly review the most significant ones. Since DUC 2005, the Pyramid metric (Passonneau et al., 2005) was introduced as one of the principal metrics for evaluating summaries in the TAC conference. This metric was created under the assumption that no single best model summary exists. The main idea is to create a gold-standard based on a comparison between human-written summaries in terms of Summary Content Units (SCUs). From a set of model summaries, similar sentences are manually identified. From these similar sentences, SCUs are generated and ranked in a pyramid model. The pyramid model has n levels, where n is the number of model summaries. The levels are labeled in ascending order from 1 to n. SCUs are ranked in the pyramid according to their occurrence in the model summaries. The resulting set of SCUs is what is called a "Pyramid". For instance, if a SCU occurs in 3 of the 4 model summaries then this SCU will be placed in the 3rd level of the pyramid. A peer summary that has X

SCUs against n model summaries is evaluated using:

$$max = \sum_{i-j+1}^{n} i \times |T_i| \times (X - \sum_{i=j+1}^{n} |T_i|) \tag{2.6}$$

where $j = max_i(\sum_{t=i}^{n} |T_t| \geq X), T_x$ is the tier at level x.

Accordingly, an SCU has a higher weight if it is used more frequently by human model summaries. Consequently, a summary covering SCUs with higher weights will have a higher pyramid score.

For example, if there are four model summaries, a SCU appearing in all summaries is considered as one of the most important concepts and would receive a weight of 4. A SCU appearing in only one model summary would be considered as less important, and would receive a weight of 1. A pyramid is formed because the tiers descend with the SCUs assigned the highest weight at the top, and the SCUs with the lowest weight appearing in the bottommost tiers (Lloret et al., 2018). The fewest SCUs would appear in the topmost tier since fewer concepts would be present in all model summaries (Figure 2.2).

Soon after, Hovy et al. (2006) proposed a metric based on comparison of basic syntactic units, so called Basic Elements (BE), between peer and model summaries. This metric, namely BE-HM was specified as one of the baselines in the TAC summarization tasks. Among participating systems in these tasks from 2009 to 2011, AutoSummENG (Giannakopoulos et al., 2008) was reported as one of the top systems. This graph-based metric (DemokritosGR), compares the graph representations of peer and model summaries.

Summarization evaluation suffers from the following problems: (i) the current evaluation metrics do not work properly in some cases (e.g., evaluating abstractive

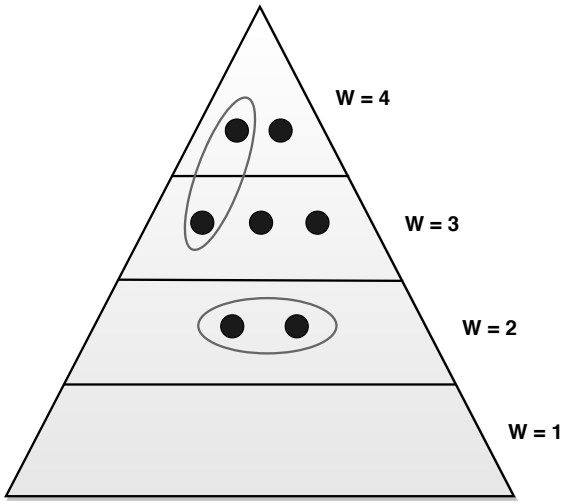


Figure 2.2: Example of a pyramid with SCUs identified and marked

summaries); (ii) the current evaluation metrics heavily rely on human judgments. For DUC or TAC conferences, human judges are asked to rate various aspects of the system summaries, e.g., grammaticality, non-redundancy, clarity or coherence. To date, none of these aspects can be properly modeled by automatic approaches. In this section, we intend to investigate these issues for ROUGE.

2.3.1 Model-based Summarization Evaluation

Surface-based evaluation metrics work well as long as a surface-based summary (i.e., extractive) is to be assessed. Difficulties arise while evaluating abstractive summaries including terminology variations and paraphrasing. For example, consider the following two phrases (Ng and Abrecht, 2015):

(i) *It is raining heavily.*

(ii) *It is pouring.*

These sentences are semantically similar, but lexically different. If we are performing

a lexical string match, as ROUGE does, there is nothing in common between the terms "raining", "heavily", and "pouring". For better clarity, if one of them is included in a model summary, while a peer summary contains another one, ROUGE or other surface based evaluation metrics cannot capture their similarity due to the minimal lexical overlap.

Recently, some evaluation metrics have studied the effectiveness of word semantic similarity to evaluate summaries including terminology variations and paraphrasing (i.e., abstractive summaries) Turian et al. (2010); Baroni et al. (2014). For instance, an automated variant of the Pyramid metric has used distributional semantics to map text content within peer summaries to SCUs (Passonneau et al., 2013). However, the SCUs still need to be manually identified. A more recent metric, ROUGE-WE (Ng and Abrecht, 2015), has enhanced ROUGE by incorporating the use of a variant of word embeddings, called word2vec (Mikolov et al., 2013b). However, a good performance for Word2vec is usually obtained upon tuning different configurations of this model on a large number of different datasets (Baroni et al., 2014).

In this thesis (Chapter 6), we propose a graph-based approach, ROUGE-G (ShafieiBavani et al., 2018a, 2017), adopted into ROUGE to evaluate summaries based on both lexical and semantic similarities. Our approach helps ROUGE with identifying the semantic similarities of linguistic items, and consequently tackling the main problem of its bias towards lexical similarities. Experiment results over TAC AESOP datasets show that exploiting the lexico-semantic similarity of the words used in summaries would significantly help ROUGE correlate better with human judgments. Beyond enhancing the evaluation prowess of ROUGE, due to its lexico-semantic analysis of summaries, our approach has the potential to expand the applicability of ROUGE to abstractive summarization.

2.3.2 Model-free Summarization Evaluation

Proposals for developing automatic summary evaluation methods (Ellouze et al., 2013; Ng and Abrecht, 2015; ShafieiBavani et al., 2018a) have been put forward in the past. However, these methods are not applicable on non-standard test sets where model summaries are not available. Herein, we try to briefly review the most significant approaches that have addressed this issue. Donaway et al. (2000) proposed an alternative to model-based evaluation where a comparison of the input text with a summary can clarify how good the summary is. A summary that has higher similarity with the input text can be considered better than one with lower similarity. Radev et al. (2003) performed an automated ranking of the test documents using a search engine scenario. Their approach was motivated by the assumption that the distribution of terms in a good summary is similar to the distribution of terms in the input document.

With the same intuition, Louis and Nenkova (2009a) and Louis and Nenkova (2013) introduced an evaluation system (SIMetrix) that comprises multiple features to determine the quality of a summary. Their focus was on computing divergences between the probability distributions of words in the input and summary. Jensen Shannon Divergence (JSD) and feature regression turned out to be their best metrics. Louis and Nenkova (2009b) also presented a similar evaluation approach utilizing a collection of large number of system summaries in place of model summaries. Saggion et al. (2010) and Cabrera-Diego and Torres-Moreno (2017) proposed follow-up work to SIMetrix to assess the usefulness of divergences for multilingual summarization evaluation, and the applicability of multiple divergences for evaluating summaries.

Alternatively, we assume that the way of representing the input and summary is a key factor in high performance prediction of manual metrics (Chapter 7). Ac-

cordingly, we present an approach (ShafieiBavani et al., 2018b) which exploits the compositional capabilities of corpus-based and lexical resource-based word embeddings to develop the features reflecting coverage, diversity, informativeness, and coherence of summaries. The features are then used to train a learning model for predicting the summary content quality in the absence of gold models. We evaluate the proposed metric in replicating the human assigned scores for summarization systems and summaries on data from query-focused and update summarization tasks in TAC 2008 and 2009. The results show that our feature combination provides reliable estimates of summary content quality when model summaries are not available.

2.4 Applications

Automatic text summarization presents a significant problem to NLP applications. In this section, we review some of the most recent ones. Among them, much research explores new applications of classic text summarization techniques. For instance, traditional summarization framework including sentence scoring and selection has been applied in new scenarios such as automatically generating presentation slides for scientific papers (Hu and Wan, 2015), and automatically constructing sport news from live commentary scripts (Zhang et al., 2016). More crafted content selection and organization have even enlightened the possibility to automatically compose poetry (Yan et al., 2013). There also exist studies for generating topically relevant event chronicles, mainly consisting of event detection module followed by learning-to-rank extractive summarization to select salient events and construct the final chronicle (Ge et al., 2015).

Summarization techniques have also been used to help interpreting predictions from neural networks, which are commonly treated as black boxes that make predictions without explicitly readable justifications. For example, it is useful to extract or

generate short rationales to explain why a neural network model predicts certain sentiment classes for a paragraph of user-generated reviews. Sentences generated for such scenarios should be concise, coherent, and sufficient for making the same predictions without referring to the full passage of review (Lei et al., 2016).

There exists another kind of high-level text summarization application that tries to produce a summary of huge topic hierarchies. Bairi et al. (2015) have recently studied this task to summarize topics over a massive topic hierarchies (a huge directed acyclic graph) such that the produced summary for a set of topics represents the objects in the collection. The representation is characterized through various classes of monotone submodular functions with learned mixture weights capturing coverage, similarity, diversity, specificity, clarity, relevance and fidelity of the topics.

These applications mainly tend to apply extractive summarization. The reason is that in case the critical information is mostly centered in particular parts of the source text and the parts themselves are non-redundant and compact, extractive summarization might help by simply extracting those parts. However, there are usually redundancy in the whole text. When necessary information exists in several parts of the text and simply copying and aggregating them does not make sense, we need an abstractive summarizer that will allow us to write new summarized content from the aggregated information. An ideal abstractive summarizer will always produce more coherent and polished summaries than extractive level. Given that, in this thesis, we will present our attempts toward abstractive text summarization.

2.5 Summary

In this chapter, we provided a detailed overview of automatic text summarization, the various types along with a description of the conventional framework for this

task. Then, we reviewed previous studies beyond sentence extraction and toward abstractive text summarization. Finally, we explored the main issues discussed in automatic evaluation of summaries, and surveyed the recent applications of text summarization techniques.

In a nutshell, due to the limitations of extractive summarization, many attempts have recently been made toward abstractive text summarization. As an intermediate step, compressive summarization that integrates sentence compression and extraction has attracted much attention. Although compressive summarization can produce more concise summaries compared to extractive approaches, they are not as flexible as fully abstractive approaches. Research in non-extractive summarization is still at the beginning, and current fully abstractive summarization approaches cannot always ensure grammatical summaries.

This thesis aims to propose approaches toward abstractive text summarization to make this task more adaptable to a wide range of applications, more dynamic to different sources and types of texts, and better evaluated using semantic representations. Chapter 3 proposes an abstractive summarizer based on multi-sentence compression. Chapters 4 and 5 propose two approaches to perform abstractive summarization in two different domains and genres. Chapters 6 and 7 will also discuss how to semantically evaluate summaries in the presence and absence of human model summaries. Finally, Chapter 8 concludes the thesis and discusses possible future directions.

Chapter 3

Word Graph-based Multi-sentence Compression

As discussed in Section 1.2.1, word graph-based approaches have recently been proposed and become popular in Multi-sentence Compression (MSC). Their key assumption is that redundancy among a set of related sentences provides a reliable way to generate informative and grammatical sentences.

In this chapter, we present an effective approach to enhance the word graph-based MSC and tackle the issue that most of the state-of-the-art MSC approaches are confronted with: i.e., improving both informativity and grammaticality at the same time. Figure 3.1 depicts the overview of the approach.

In Section 3.1, we first explain how to construct a word graph for MSC. Then, we describe the details of our approach that consists of three main components: (i) a merging strategy based on Multiword Expressions; (ii) a mapping strategy based on synonymy between words; (iii) a re-ranking strategy based on POS-based Language Model to identify the best compression candidate in terms of grammatical structure.

Data preparation is discussed in Section 3.2, and Section 3.3 provides the experiment results. Finally, Section 3.4 summarizes the chapter.

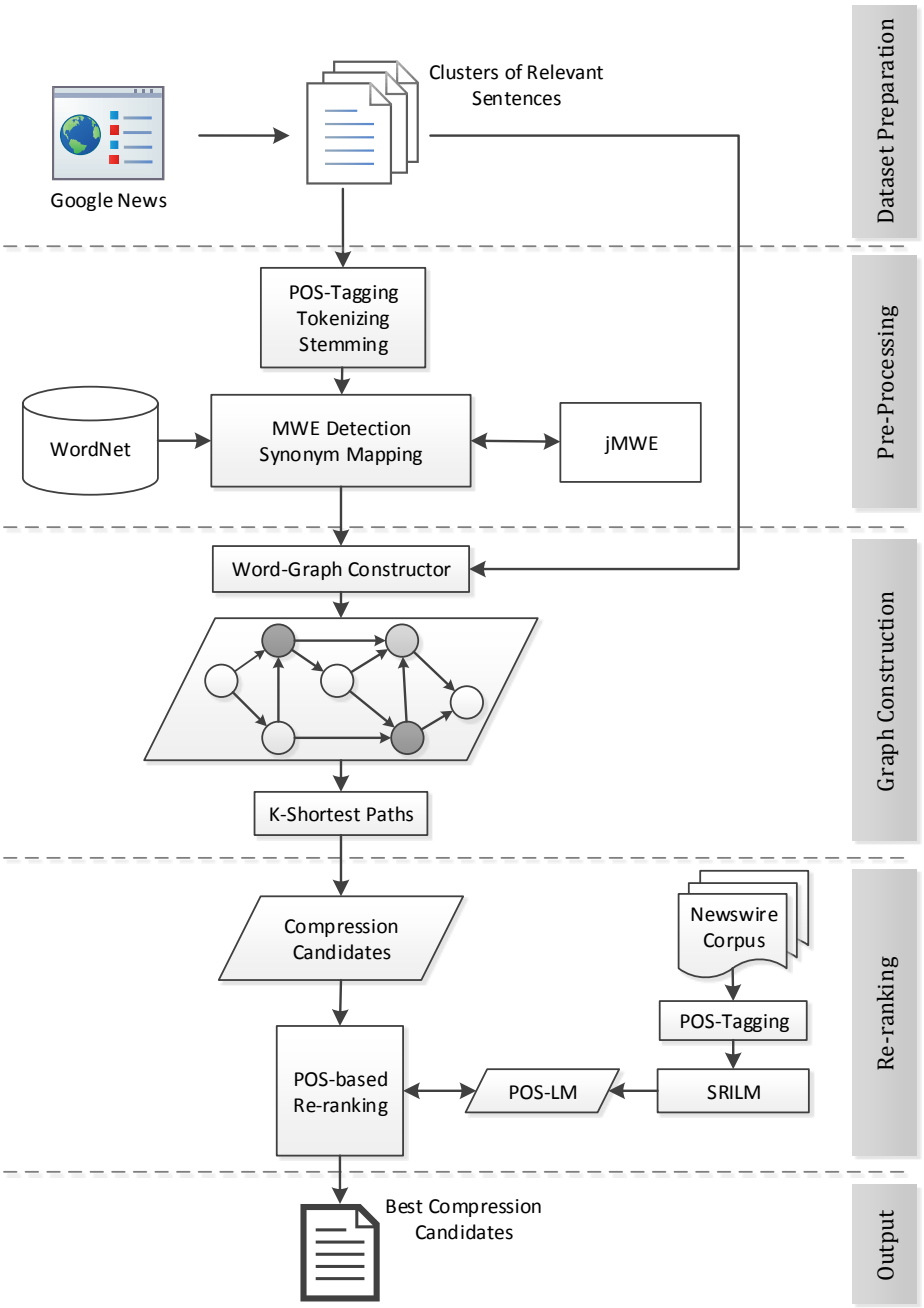


Figure 3.1: Overview of the proposed word graph-based MSC approach

3.1 Proposed Approach

3.1.1 Word Graph Construction for MSC

Consider a set of related sentences $S = \{s_1, s_2, ..., s_n\}$, a traditional word graph is constructed by iteratively adding sentences to it. This directed graph is an ordered pair $G = (V, E)$ comprising of a set of vertices or words together with a set of directed edges which shows the adjacency between corresponding nodes (Filippova, 2010b; Boudin and Morin, 2013). The graph is firstly constructed by adding the first sentence and displays words in a sentence as a sequence of connected nodes. The first word is the start node and the last one is the end node. Words are added to the graph in three steps of the following order:

(i) non-stopwords for which no candidate exists in the graph; or for which an unambiguous mapping is possible (i.e., there is only one node in the graph that refers to the same word/POS pair).

(ii) non-stopwords for which there are either several possible candidates in the graph; or for which they occur more than once in the sentence.

(iii) stopwords for which we use the stopword list included in NLTK[1] extended with temporal nouns such as 'yesterday', 'Friday', and etc..

All MSC approaches aim at producing condensed sentences that inherit the most important information from the original content while remaining syntactically correct. However, gaining these goals at the same time remains still difficult. As a remedy, we believe that a better resolution to construct an improved word graph can be

[1]The Natural Language Toolkit, or more commonly NLTK, is a suite of libraries and programs for computational linguistics written in the Python programming language; available at http://nltk.org/

obtained by using more sophisticated preprocessing and re-ranking steps. Therefore, we focus on the notions of synonymy, MWE and POS-LM re-ranking, which dramatically raise the informativity and grammaticality of compression candidates. In the following, we describe the details of our proposed approach.

3.1.2 Merging and Mapping Strategies

Like many NLP applications, MSC will benefit from the identification of MWEs and the concept of synonymy; and even more so when lexical diversity arises in a collection of sentences. An MWE is a combination of words with lexical, syntactic or semantic idiosyncrasy (Sag et al., 2002; Baldwin and Kim, 2010). It is estimated that the number of MWEs in the lexicon of a native speaker of a language has the same order of magnitude as the number of single words (Jackendoff, 1997). Hence, explicit identification of MWEs has been shown to be useful in various NLP applications. Components of an MWE can be treated as a single unit to improve the effectiveness of re-ranking steps in IR systems (Acosta et al., 2011).

In this section, we identify MWEs, merge their components, and replace them with their available *one*-word synonyms, if applicable. These strategies help to construct an improved word graph and enhance the informativity of the compression candidates. For example, consider a sentence that includes an MWE (*kick the bucket*): *It would be a sad thing to kick the bucket without having been to Alaska.* To benefit from this MWE that has 3 components/words, we propose the merging strategy below.

Firstly, after tokenizing the sentence and stemming the words, we detect the MWE and its tuple POS with an MWE detector, and merge its components by hyphenation. This step has the advantage of reducing the ambiguity of mapping upcoming words onto the existing words with the same appearance in the graph. For example,

the word *kick* above has a different meaning and POS (as an MWE component) from the identical appearance word *kick* in isolation (in another sentence say *"they kick open the door and entered the room."*). This strategy can keep us from mapping these two *kick* together and retain the important meaning of the content. Herein, we use the jMWE[2] toolkit, which is a Java-based library for detecting MWEs.

Secondly, we use version 3.0 of WordNet (Miller, 1995) to obtain its available *one*-word synonym with an appropriate POS and replace the *n*-words MWE with a shorter synonym word. WordNet groups all synonyms into a SynSet - a synonym set. We only consider the most frequent *one*-word synonym in the WordNet that also appears in the other relevant sentences. If other relevant sentences contain none of the *one*-word synonyms, the most frequent one is selected directly from the WordNet to help condense the sentence. Three native speakers were asked to investigate all the synonym mappings performed in our approach, and specify whether each mapped synonym reflects the meaning of the original word in the sentence or not. Based on this evaluation, the average rate of correct synonym mappings is 88.21%. In case that no appropriate synonym is found for MWE, the merged MWE itself was used as a back-off. This can reduce the number of graph nodes and, consequently, the ambiguity for further false mappings of MWE components in the word graph. These steps are briefly depicted in Figure 3.2.

Furthermore, we use the concept of synonymy for mapping upcoming single words. For example, consider *n* different sentences containing words *glorious*, *splendid*, *festivity*, *celebration*, and *jubilation*. The former two words, and the latter three ones are synonyms of each other. Assume each sentence contains one of these possible combinations (i.e., glorious festivity, glorious celebration, glorious jubilation, splendid festivity, splendid celebration, and splendid jubilation).

[2]http://projects.csail.mit.edu/jmwe/

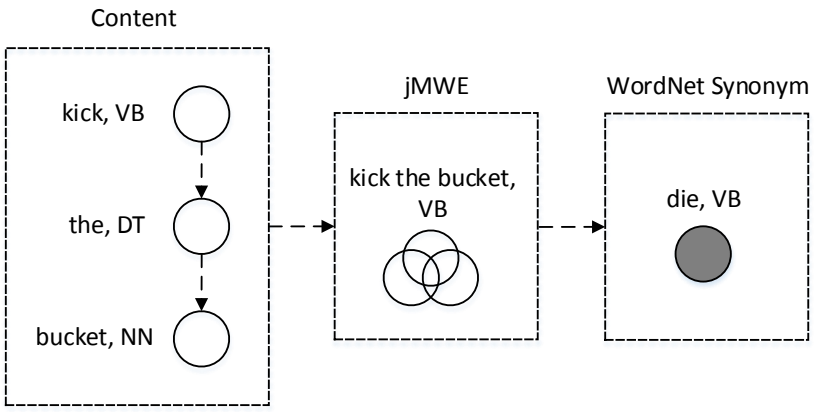


Figure 3.2: Example of detecting, merging, and mapping multi-word expressions

Without an appropriate mapping based on a notion of synonymy, these 3 nodes will be added to the word graph as separate nodes. With our approach, the word graph in this example is constructed with a single node containing a word as a representative of its synonyms from the other sentences. The weight of the obtained node is computed by summing the frequency scores from the other nodes as shown in Figure 3.3 for each pair of word/POS. The main purpose of this modification is three fold: (i) the ambiguity of mapping nodes is reduced; (ii) the number of total possible paths (compression candidates) is decreased; and (iii) the weight of frequent similar words with different appearances in the content is better reflected by the notion of synonymy.

Example 3.1.1. In this example, we will demonstrate how we use the preprocessing strategies to produce refined sentences, and generate an improved word graph. Among the underlined words, MWEs are put into bracket, and synonyms are identified by the same superscript notations. The word graph constructed for the following sentences are partially shown in Figure 3.4. Some nodes, edge weights and punctuations are omitted from the graph for more clarity.

- Teenage[a] boys are more interested[b] in [junk food][c] marketing and consume[d]

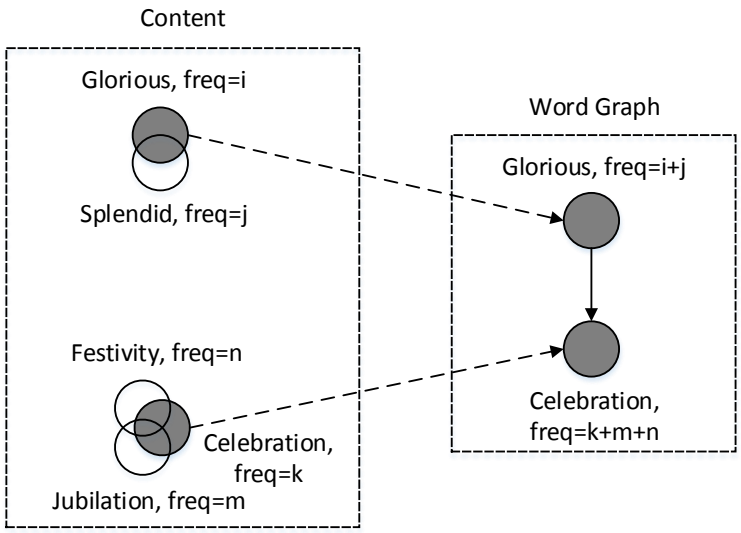


Figure 3.3: Example of Synonym Mapping

more [fast food]^{c} than girls.

- [Junk food]^{c} marketers find young^{a} boys more fascinated^{b} than girls, a survey released^{e} by the Cancer Council shows.
- Adolescent^{a} boys [use up]^{d} more [fast food]^{c} than girls, [according to] a new survey.
- The survey, published^{e} by the Cancer Council, observed teenage^{a} boys were regular consumers of [junk food]^{c}.

Where mapping in the graph is ambiguous (i.e., there are two or more nodes in the graph that refer to the same word/POS pair), we follow the instruction stated by Filippova (2010b): the immediate context (the preceding and following words in the sentence, and the neighboring nodes in the graph) or the frequency (i.e., the node which has words mapped to it) is used to select the best candidate node. A new node is created only if there is no suitable candidate to be mapped to, in the graph.

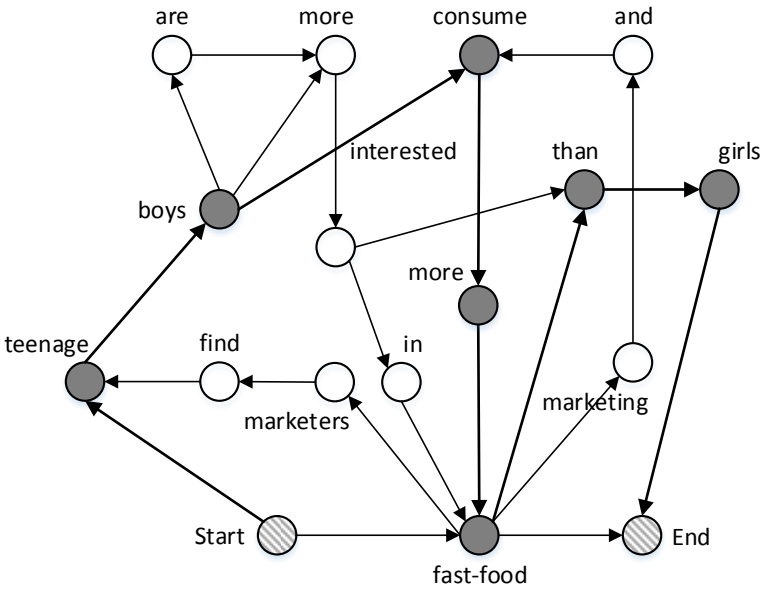


Figure 3.4: The generated word graph and a salient path

Edge weights are calculated using the weighting function defined in Equation 3.1, in which $w^{'}(e_{i,j})$ is given by Equation 3.2 (Filippova, 2010b).

$$w(e_{i,j}) = \frac{w^{'}(e_{i,j})}{freq(i) \times freq(j)} \tag{3.1}$$

$$w^{'}(e_{i,j}) = \frac{freq(i) + freq(j)}{\sum_{s \in S} diff(s,i,j)^{-1}} \tag{3.2}$$

where $freq(i)$ is the number of words mapped to the node i. The function $diff(s,i,j)$ refers to the distance between the offset positions of words i and j in sentence s.

Algorithm 1 presents the steps to build our proposed MSC word graph G(V, E). We start with a cluster of relevant sentences from a set of input newswire clusters. Each cluster is denoted as $S = \{s_i\}_{i=1}^{n}$ where s_i is a sentence containing POS annotations. *Line 4-5:* Each $s_i \in S$ is split into a set of tokens, where each token t_j consists

Algorithm 1 Proposed MSC Word Graph

Input: A cluster of relevant sentences: $S = \{s_i\}_{i=1}^{n}$
Output: $G = (V, E)$
for $i = 1$ to n **do**
    $t \leftarrow Tokenize(s_i)$
    $st \leftarrow Stemming(t)$
    $MWE\text{-}comp \leftarrow MWE\text{-}Detection(t, st)$
    $MWE\text{-}list \leftarrow Merge\text{-}MWE(MWE\text{-}comp)$
    $sentSize \leftarrow SizeOf(t)$
    for $j = 1$ to $sentSize$ **do**
        $LABEL \leftarrow t_j$
        $SID \leftarrow i$
        $PID \leftarrow j$
        $SameN \leftarrow getSameNodes(G, LABEL)$
        if $sizeOf(SameN) \geq 1$ **then**
            $v_j \leftarrow getBestSame(SameN)$
            $mapList_{v_j} \leftarrow mapList_{v_j} \cup (SID, PID)$
        else
            $SynN \leftarrow getSynonymNodes(G, LABEL)$
            if $sizeOf(SynN) \geq 1$ **then**
                $v_j \leftarrow getBestSyn(SynN)$
                $mapList_{v_j} \leftarrow mapList_{v_j} \cup (SID, PID)$
            esle if $t_j \in MWE\text{-}list$ **then**
                $WNSyn \leftarrow getBestWNSyn(LABEL)$
                $v_j \leftarrow creatNewNode(G, WNSyn)$
                $mapList_{v_j} \leftarrow (SID, PID)$
            esle
                $v_j \leftarrow creatNewNode(G, LABEL)$

28: $mapList_{v_j} \leftarrow (SID, PID)$
29: **end if**
30: **end if**
31: **if** *not existEdge*$(G, v_{j-1} \rightarrow v_j)$ **then**
32: *addEdge*$(v_{j-1} \rightarrow v_j, G)$
33: **end if**
34: **end for**
35: **end for**

of a word and its corresponding POS annotation (e.g.,*"boys:NN"*). The tokens are also stemmed into a set of stemmed words *st*. *Line 6-7:* For each sentence, MWE components, i.e., *MWE-comp*, are detected using the set of tokens t and stems *st*. Then, these MWE components are merged in each sentence, and kept in a list of *MWE-list*. *Line 10-12:* Each unique t_j will form a node v_j in the MSC graph, with t_j being the label. Since we only have one node per unique token, each node keeps track of all sentences that include its token. Hence, each node keeps a list of *sentence identifier* (SID) along with the *position of token* in that sentence (PID). Each node including a single word or a merged MWE will thus carry a *mapping list* (mapList) which is a list of {SID:PID} pairs representing the node's membership in a sentence.

Line 13-16: For mapping the token t_j, we first explore the graph to find the same node (i.e., node that refers to the same word/POS pair as t_j). If two or more same nodes are found, considering the aforementioned ambiguous mapping criteria in this section, the best candidate node is selected for mapping. Then the pair of (SID:PID) of t_j will be added to the mapping list of the selected node, i.e., $mapList_{v_j}$. *Line 18-21:* If no same node exists in the graph, then we look for the best synonym node in the graph (i.e., finding the most frequent synonym among the WordNet synsets that was earlier added to the graph). Again, the mapping list of the selected node, $mapList_{v_j}$ will be updated to include the pair of (SID:PID) of t_j. *Line 22-28:* If

none of the above conditions are satisfied, it is time to create a new node in the graph. However as explained earlier in this section, when t_j is MWE, we extract the best WordNet *one*-word synonym, and replace the n-words MWE with this shorter synonym word. Given that, a shorter content node will be added to the graph. *Line 31-33:* The original structure of a sentence is reordered with the use of directed edges.

A heuristic algorithm is then used to find the k-shortest paths from start to end node in the graph. Throughout our experiments, the appropriate value for k is 150. By re-ranking this number of shortest paths, most of the potentially good candidates are kept and a decline in performance is prevented. Paths shorter than eight words or do not contain a verb were filtered before re-ranking. The remaining paths are re-ranked and the path that has the lightest average edge weight is eventually considered as the best compression. Next, we explain our re-ranking approach to identify the most informative and grammatical compression candidates.

3.1.3 Re-ranking Strategies

In this section, we first re-rank the compression candidates based on the assumption that a word can recommend other co-occurring words, and the strength of the recommendation is recursively computed based on the importance of the words making the recommendation. For this purpose, we utilize TextRank (Mihalcea and Tarau, 2004) - a graph-based algorithm that takes into account edge weights. Accordingly, the score of a keyphrase k is computed by summing the salience of the words it contains, normalized with its $length + 1$ to favor longer n-grams according to Equation 3.3. Finally, the paths are re-ranked and the score of a compression candidate c is

given by Equation 3.4 (Boudin and Morin, 2013).

$$Score_{Key}(k) = \frac{\sum_{w \in k} TextRank(w)}{length(k) + 1} \tag{3.3}$$

$$Score_{Key}(c) = \frac{\sum_{i,j \in path(c)} w(e_{i,j})}{length(c) \times \sum_{k \in c} Score_{Key}(k)} \tag{3.4}$$

We then benefit from the fact that POS tags capture the syntactic roles of words in a sentence. We train a Part-Of-Speech Language Model (POS-LM) to assign a grammaticality score to each generated compression. A language model assigns a probability to a sequence of m words $P(w_1, ..., w_m)$ by means of a probability distribution. Hence, POS-LM describes the probability of a sequence of m POS tags $P(t_1, ..., t_m)$. Our hypothesis is that POS-LM helps in identifying the most grammatical sentence among the k-most informative compressions. This strategy shall improve the grammaticality of MSC, even when the grammatical structures of the input sentences are completely different. Word-based language models estimate the probability of a string of m words by Equation 3.5. Likewise, POS-LMs estimate the probability of string of m POS tags by Equation 3.6 (Monz, 2011).

$$p(w_1^m) \propto \prod_{i-1}^{m} p(w_i | w_{i-n+1}^{i-1}) \tag{3.5}$$

$$p(t_1^m) \propto \prod_{i-1}^{m} p(t_i | t_{i-n+1}^{i-1}) \tag{3.6}$$

where, n is the order of the language model, and w/t refers to the sub-sequence of words/tags from position i to j.

To build a POS-LM, we use the SRILM[3] toolkit with modified Kneser-Ney smoothing (Stolcke, 2002), and train the language model on our POS annotated corpus. SRILM collects n-gram statistics from all n-grams occurring in a corpus to build a single global language model. To train our POS-LM, we need a POS-annotated corpus. In this regard, we make use of the Stanford POS tagger (Toutanova et al., 2003) to annotate the AFE sections of LDC's Gigaword corpus (LDC2003T05) as a large newswire corpus (∼170M words). Then, we remove all words from the pairs of words/POS in the POS annotated corpus.

Although the vocabulary of a POS-LM, which is usually ranging between 40 and 100 tags, is much smaller than the vocabulary of a word-based language model, there is still a chance in some cases of unseen events. Since modified Kneser-Ney discounting appears to be the most efficient method in a systematic description and comparison of the usual smoothing methods (Goodman, 2001), we use this type of smoothing to help our language model. The compression candidates also need to be annotated with POS tags. Hence, the score of each compression is estimated by the language model, based on its sequence of POS tags. Since factors like POS tags, are less sparse than surface forms, it is possible to create a higher order language models for these factors. This may encourage more syntactically correct output. Thereupon, in our approach we use 7-gram language modeling based on part-of-speech tagging to re-rank the k-best compressions generated by the word graph.

To re-rank the obtained paths, our POS-LM gives the perplexity score ($Score_{LM}$) which is the geometric average of $1/probability$ of each sentence, normalized by the number of words. Hence, $Score_{LM}$ for each sequence of POS in the k-best

[3]SRILM is a toolkit for building and applying statistical language models, primarily for use in speech recognition, statistical tagging and segmentation, and machine translation; available at http://www.speech.sri.com/projects/srilm/

compressions is computed by:

$$Score_{LM}(c) = 10^{\frac{\log prob(c)}{|word|}} \tag{3.7}$$

where $prob(c)$ is the probability of compression candidate (C) including $|word|$ number of words, computed by the 7-gram POS-LM.

As the estimated scores for each cluster of sentences fall into different ranges, we make use of a unity-based normalization to bring the values of $Score_{Key}(c)$ in Equation 4, and the $Score_{LM}$ into the range [0, 1]. The score of each compression is finally given by:

$$Score_{final}(c) = \mu \times Score_{Key}(c) + (1 - \mu) \times Score_{LM}(c) \tag{3.8}$$

in which the scaling factor μ was optimized on development data in our experiments and has been set to 0.4, so as to reach the best re-ranking results.

To better understand how POS-LM is used, consider the following example:

Example 3.1.2. In this example, sentences have the same scores for informativity but are added into our re-ranking contest to be investigated based on their grammaticality. The corresponding POS sequences of these sentences are given to the trained language model to clarify which one is more grammatical. As expected, the winner of this contest is the second POS sequence, which has a better grammatical structure and gets a higher probability score from the POS-LM.

(i)	Boys	*more*	*consume*	fast	food	than	girls.
	NNS	*RBR*	*VBP*	JJ	NN	IN	NNS
		Wrong Pattern					

(ii)	Boys	*consume*	*more*	fast	food	than	girls.
	NNS	*VBP*	*JJR*	JJ	NN	IN	NNS

3.2 Data Preparation

Many attempts have been made to release various kinds of datasets and evaluation corpora for sentence compression and automatic summarization, such as the one introduced in (Clarke and Lapata, 2006). However, to our knowledge, there is no dataset available to evaluate MSC in an automatic way (Boudin and Morin, 2013). Since the prepared dataset in Boudin and Morin (2013) is also in French, we have followed the below instructions to construct a Standard English newswire dataset:

We have collected news articles in clusters on the Australian[4] and U.S.[5] editions of Google News over a period of five months from January to May 2015. Clusters composed of at least 15 news articles about one single news event, were manually extracted from different categories (i.e., Top Stories, World, Business, Technology, Entertainment, Science, Health, etc.). Leading sentences in news articles are known to provide a good summary of the article content and are used as a baseline in summarization (Dang, 2005). Hence, to obtain the sets of related sentences, we have extracted the first sentences from the articles in the cluster and removed duplicates.

The released dataset contains 568 sentences spread over 46 clusters (each is related to one single news event). The average number of sentences within each cluster is

[4]http://news.google.com.au/

[5]http://news.google.com/

12, with a minimum of 7 and a maximum of 24. Three native English speakers were also asked to meticulously read the sentences provided in the clusters, extract the most salient facts, summarize the set of sentences, and generate three model summaries for each cluster with as little new vocabulary as possible.

In practice, along with the clusters of sentences with similar lexical and grammatical structures (we refer to these clusters as *normal*), it is likely to have clusters of content-relevant sentences, but with different (non-redundant) appearance and grammatical structure (we consider these clusters as *diverse*). In fact, the denser a word graph is, the more edges interconnect with vertices and hence more paths pass through the same vertices. This results in low lexical and syntactical diversity, and vice versa (Tzouridis et al., 2014). The density of a word graph $G = (V, E)$ generated for each cluster is given by:

$$Density = \frac{|E|}{|V|(|V| - 1)} \tag{3.9}$$

Thereupon, we have identified 15 *diverse* clusters among the 46 clusters to demonstrate the effectiveness of our approach on the normal and diverse groups. Table 3.1 lists the properties of the evaluation dataset.

total #clusters	46
#normal clusters	31
#diverse clusters	15
total #sentences	568
avg #sentences/cluster	12
min #sentences/cluster	7
max #sentences/cluster	24

Table 3.1: Information about the constructed dataset for MSC

3.3 Experiments

3.3.1 Evaluation Metrics

We evaluate the proposed method over our constructed dataset (*normal* and *diverse* clusters) using automatic and manual evaluations. The quality of the generated compressions is assessed automatically through version 2.0[6] of ROUGE (Lin, 2004) and the version 13a[7] of BLEU (Papineni et al., 2002). These sets of metrics are typically used for evaluating automatic summarization and machine translation. They compare an automatically system-generated (peer) summary against human-generated (model) summaries.

For the manual investigation of the quality of the generated compressions, three native raters were asked to rate the grammaticality and informativity of the compressions based on the following points scale (Filippova, 2010b): *Grammaticality*: (i) if the compression is grammatically perfect → *point 2*; (ii) if the compression requires some minor editing → *point 1*; (iii) if the compression is ungrammatical → *point 0*. The lack of capitalization is ignored by the raters. *Informativity*: (i) if the compression conveys the gist of the content and is mostly similar to the human-produced summary → *point 2*; (ii) if the compression misses some important information → *point 1*; (iii) if the compression contains none of the important contents → *point 0* (Table 3.2). The k value for the agreement between raters falls into range ($0.4 \sim 0.6$) through Kappa's[8] evaluation metrics, which indicates that the strength of this agreement is moderate (Artstein and Poesio, 2008).

[6]http://kavita-ganesan.com/content/rouge-2.0/

[7]ftp://jaguar.ncsl.nist.gov/mt/resources/mteval-v13a.pl/

[8]Cohen's kappa coefficient (κ) measures inter-rater agreement for qualitative (categorical) items. It is a more robust measure than simple percent agreement, as κ takes into account the possibility of the agreement occurring by chance.

Feature	State of the Compression	Point		
		2	**1**	**0**
Grammaticality	grammatically perfect	✔		
	requires some minor editings		✔	
	ungrammatical			✔
Informativity	conveys the gist of the content	✔		
	misses some important information		✔	
	contains none of the important contents			✔

Table 3.2: Points scale defined in the agreement between raters

3.3.2 Experiment Results

For comparison purposes in our experiments, two existing approaches by Filippova (2010b) and Boudin and Morin (2013) are used as Baseline1 and Baseline2, respectively. To better understand the behavior of our system, we examined our test dataset, and made the following observations.

Manual Evaluation

Considering the results in Table 3.3, we observe a significant improvement in the average grammaticality and informativity scores along with the compression ratio (CompR) over the normal and diverse clusters. The informativity of Baseline1 is adversely influenced by missing important information about the set of related sentences (Boudin and Morin, 2013). However Baseline2 enhanced the informativity, the grammaticality scores are decreased due to the outputs of longer compressions. In our approach, the remarkable improvement in the grammaticality scores is due to the adding of the syntactic-based re-ranking step. Using this re-ranking method, the most grammatical sentences are picked among the k-best compression candidates. Furthermore, merging MWEs, replacing them with their available *one*-word

synonyms, and mapping words using synonymy all enhance the informativity scores, and help to generate a denser word graph instead of a sparse one. Given that, the value of the compression ratio (∼48%) is better than the best obtained compression ratio on the two baselines (50%).

Method	**Normal**		**Diverse**		**CompR**
	Info.	**Gram.**	**Info.**	**Gram.**	
Baseline1	1.44	1.67	1.17	1.19	50%
Baseline2	1.68	1.60	1.30	1.12	58%
Proposed	1.68	1.68	1.36	1.47	48%

Table 3.3: Manual Evaluation: Average scores over normal and diverse clusters, along with the estimated compression rates

Automatic Evaluation

The average performance of the baseline methods and the proposed approach over the normal and diverse clusters in terms of ROUGE and BLEU scores are also shown in Table 3.4. ROUGE measures the concordance of peer and model summaries by determining n-grams, word sequences, and word pair matches. Precision, Recall, and F-measure are used in ROUGE to perform comparisons over the summary scores. While precision and recall separately measure the correctness and coverage, we evaluate the compression candidates using F-measure that aggregates these two measures into a single value[9] for unigrams, bigrams, and SU4 (skip-bigrams with maximum gap length 4). The BLEU metric computes the scores for individual sentences; then averages these scores over the whole corpus for a final score. We use BLEU for 4-grams to evaluate the results.

To make the candidate and model summaries comparable, a process of manual

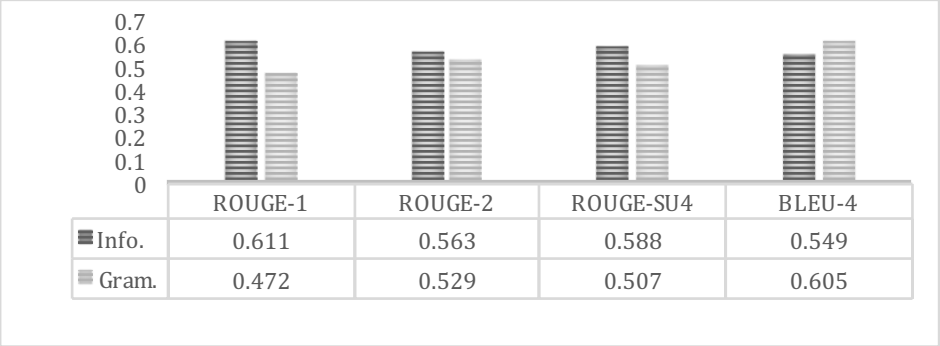
[9] $F\text{-}measure = 2 \times \frac{Precision \times Recall}{Precision + Recall}$

Metric	**Baseline1**	**Baseline2**	**Proposed**
ROUGE-1	0.4912	0.5093	0.5841
ROUGE-2	0.3050	0.3131	0.4284
ROUGE-SU4	0.2867	0.3002	0.3950
BLEU-4	0.4510	0.5144	0.6913

Table 3.4: Automatic Evaluation: Average scores over normal and diverse clusters

MWE detection is performed on the model summaries and the MWE components are merged by three native annotators. In details, automatic evaluation packages use WordNet to compare the synonyms in each candidate and model summaries. WordNet puts hyphenation on synonyms, e.g., kick-the-bucket, so annotators hyphenate MWEs in their summaries to be used in these packages. Then, the synonym properties are set in these packages to consider the synsets. Therefore, n-words MWEs are linked to their *one*-word synonyms in the candidate summary. The overall results support our hypothesis that using the POS-LM for re-ranking the compression candidates, results in more grammatical compressions, especially for diverse clusters. This issue is confirmed by 4-grams BLEU, which shows the grammaticality enhancement rather than the informativity. Meanwhile, we try to simultaneously improve the informativity by identifying and merging MWEs along with mapping the synonyms.

Furthermore, the effectiveness of ROUGE and BLEU is studied using the Pearson's correlation coefficient. We found that ROUGE shows a better correlation with informativity, while the BLEU correlates better with grammaticality. Overall, the results in Figure 3.5 show high correlation ($0.5 \sim 1.0$) between the automatic evaluation results and human ratings for both ROUGE and BLEU. The main reason may be the simulation of factors that humans usually consider for summarization, such as merging and mapping strategies, along with the syntactic criteria employed by POS-LM.



	ROUGE-1	ROUGE-2	ROUGE-SU4	BLEU-4
Info.	0.611	0.563	0.588	0.549
Gram.	0.472	0.529	0.507	0.605

Figure 3.5: The effectiveness of ROUGE and BLEU

Ablation Study

To investigate the impact of each improvement separately, we have also conducted separate experiments over the prepared dataset. The results are shown in Figure 3.6 and the related data are provided in Table 3.5. We observe that merging and mapping strategies significantly increase the informativity of the compressions. Hence, their computed scores by ROUGE are higher than the score of POS-LM. However, the combination of MWE merging and mapping gets a slightly lower score from ROUGE-SU4. One reason may be that usage of synonymy only for MWEs and ignoring other *one*-word synonym mappings causes a more diverse graph, which slightly decreases the informativity and grammaticality of the compressed sentences. Meanwhile, POS-LM gets better scores from BLEU-4, which indicates the grammaticality enhancement rather than the informativity.

Metric	**Synonymy**	**Merg/Map**	**POS-LM**	**All**
ROUGE-1	0.5659	0.5820	0.5381	0.5841
ROUGE-2	0.3723	0.4087	0.3599	0.4284
ROUGE-SU4	0.3508	0.3254	0.3629	0.3950
BLEU-4	0.5340	0.5601	0.6725	0.6913

Table 3.5: The impacts of the improvements separately

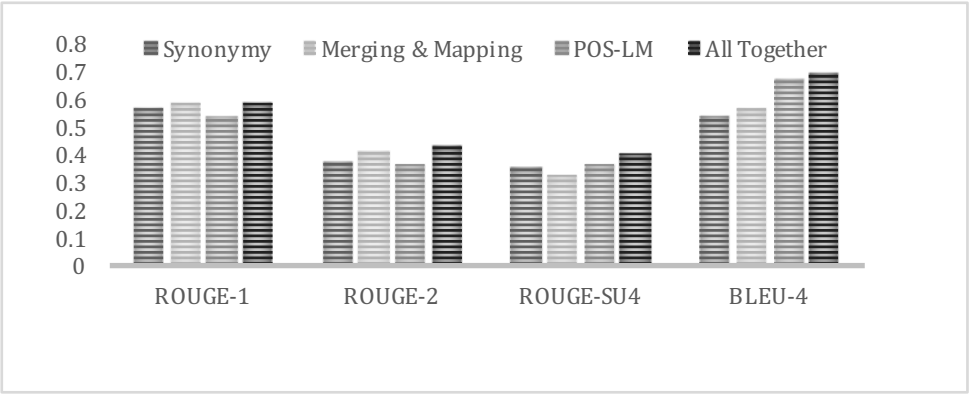


Figure 3.6: The impacts of the improvements separately

3.4 Summary

We have presented our attempt in using MWEs, Synonymy and POS-based language modeling to tackle one of the pain points of MSC, which is improving both informativity and grammaticality at the same time. By manual and automatic (ROUGE and BLEU) evaluations, experiments using the constructed English newswire dataset showed that our approach outperforms the competitive baselines. In particular, the proposed merging and mapping strategies, along with the grammar-enhanced POS-LM re-ranking method, ameliorate both informativity and grammaticality of the compressions, with an improved compression ratio. This approach can be used as an abstractive summarizer in a wide range of applications. To show this potential, we will use the proposed MSC word graph as a component for query-focused multi-document summarization in Chapter 4 and for domain-specific summarization in Chapter 5.

Chapter 4

Query-focused Multi-document Summarization

In Chapter 3, we proposed an enhanced word graph-based MSC approach that can be used as an abstractive summarizer in a wide range of applications. This chapter proposes a query-focused multi-document summarization approach for newswire. Herein, we utilize WordNet to measure the semantic relatedness between the input query and news sentences, while exploring the applicability of previously proposed MSC word graph as a component of the multi-document summarizer. Figure 4.1 provides an overview of the proposed approach.

In Section 4.1, we first explain why capturing semantic similarities between the query and sentences in the source text is important for this task. The captured semantic similarities over WordNet are then used for the relations at sentence level, and semantic disambiguation of words. Our approach to satisfy the query-biased relevance, biased information novelty and richness criteria is also introduced in the same section. We then discuss the utilized data and the conducted experiments in Sections 4.2 and 4.3. Finally, Section 4.4 concludes and summarizes this chapter.

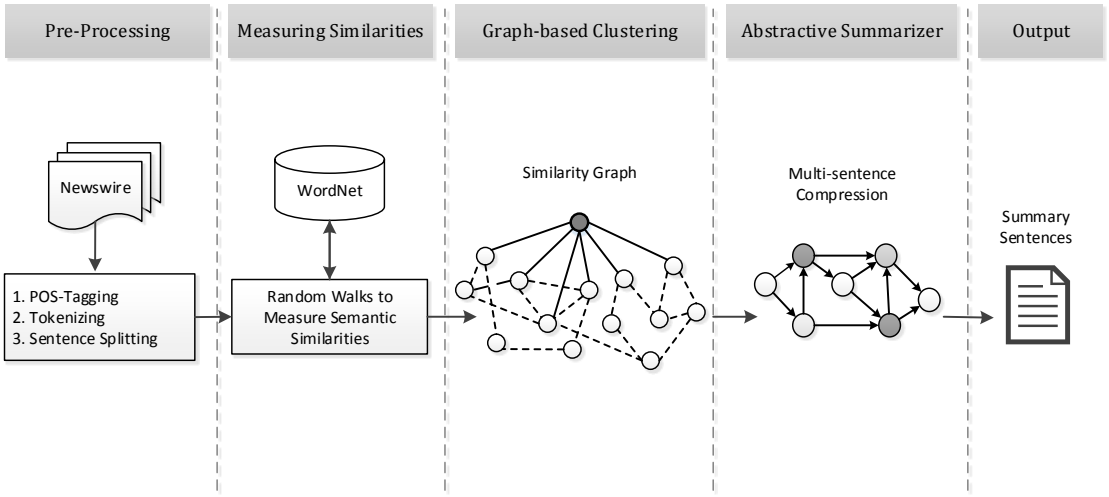


Figure 4.1: Overview of the proposed news summarization approach

4.1 Proposed Approach

4.1.1 Capturing Semantic Similarities

Quantifying semantic relationship between linguistic items lies at the core of many NLP applications. However, hard matching between words has long been an obstacle in identifying the relatedness of two sentences (Yin et al., 2012). For better clarity, consider the following example for general domain (Pilehvar and Navigli, 2015):

Example 4.1.1. General Domain:

a1. Officers fired.
a2. Several policemen terminated in corruption probe.

b1. Officers fired.
b2. Many injured during the police shooting incident.

Surface-based approaches that are merely based on string similarity cannot capture the relevancy between any of the above pairs of sentences because there exists no lexical overlap. In addition, a surface-based semantic similarity approach considers both a_1 and b_1 as being identical sentences, whereas different meanings of the verb *fire* are triggered in the two contexts.

In this section, we first compute sentence-to-sentence and sentence-to-query semantic similarities and any semantic ambiguity therein. To this end, we use WordNet 3.0 repository (Fellbaum, 1998) as our sense inventory. Next, we briefly explain our journey to capture semantic similarities between sentences. In our work, we treat a query as a long single sentence.

Many existing approaches to automatic text summarization rely on comparing the similarity of two sentences in some ways. In our approach, the main requirement for computing semantic similarities on WordNet is *Semantic Signature*, which is a multinomial distribution generated from repetitive random walks on WordNet (Pilehvar et al., 2013). To construct each semantic signature, an iterative method for calculating topic-sensitive PageRank has been used (Haveliwala, 2002). The key assumption is that repeated random walks beginning at a set of senses (seed nodes) in WordNet network can provide a frequency or multinomial distribution over all senses in WordNet. A higher probability will then be assigned to senses that are frequently visited from the seeds. Consider an adjacency matrix M for the WordNet network, where edges connect senses according to the relations defined in WordNet (e.g., synonymy, hypernymy[1], meronym[2] and etc.). A sense is further connected to all the other senses that appear in its disambiguated gloss.

[1] X is a hypernym of Y if X is the generalization of Y (e.g., amphibian is the hypernym of frog).

[2] X is a meronym of Y if X is a part or a member of Y (e.g., wheel is a meronym of car).

The probability distribution for the starting location of the random walker in the network is denoted by $\vec{s}^{(0)}$. Given the set of senses S in a sentence, the probability mass of $\vec{s}^{(0)}$ is uniformly distributed across the senses $s_i \in S$, with the mass for all $s_j \notin S$ set to zero. The PageRank vector is then computed as:

$$\vec{s}^{(t)} = (1 - \alpha)M\vec{s}^{(t-1)} + \alpha\vec{s}^{(0)} \tag{4.1}$$

where at each iteration, the random walker may jump to any node $s_i \in S$ with probability $\alpha/|S|$. Following the standard convention, the value of α is set to 0.15. The number of iterations is also set to 30, which is sufficient for the distribution to converge. The resulting probability vector $\vec{s}^{(t)}$ is the semantic signature of the sentence, as it has aggregated its senses' similarities over the entire graph. The UKB[3] off-the-shelf implementation of topic-sensitive PageRank has been used in this step.

4.1.2 Semantic Similarities at Sentence Level

For comparing pairs of signatures at sentence level, we use the Weighted Overlap algorithm (Pilehvar et al., 2013). This algorithm first sorts the two signatures according to their values and then harmonically weights the overlaps between them. The weighting process is such that differences in the highest ranks are penalized more than differences in the lower ranks (the first-ranked element has the highest rank). Finally, we calculate the similarity of two sentence signatures S_1 and S_2 using

[3]http://ixa2.si.ehu.es/ukb/

the following equation:

$$Sim(S_1, S_2) = \frac{\sum_{h \in H} (r_h(S_1) + r_h(S_2))^{-1}}{\sum_{i=1}^{|H|} (2i)^{-1}} \tag{4.2}$$

where H denotes the intersection of all senses with non-zero probability in both signatures (all non-zero dimensions) and $r_h(S_{Nj})$ denotes the rank of the dimension H in the sorted signature S_j, where rank 1 denotes the highest rank. The denominator is also used as a normalization factor that guarantees a maximum value of one. The minimum value is zero and occurs when there is no overlap between the two signatures, i.e., $|H| = 0$.

4.1.3 Semantic Disambiguation

In order to use a deeper modeling of linguistic items at the sense level, each word in a text has first to be analyzed and disambiguated into its intended sense. However, due to the inherent information shortage of sentences, traditional forms of Word Sense Disambiguation (WSD) are hard to use. Therefore, we make use of an alignment-based sense disambiguation algorithm (Pilehvar and Navigli, 2015). This algorithm leverages the content of the paired sentence in order to disambiguate each element.

Given two sentences, for each word type w_i in sentence S_1, the semantic alignment algorithm assigns w_i to the sense that has the maximal similarity score to any sense of the word types in the compared sentence S_2. For better clarity, let us perform the semantic disambiguation procedure for sentences in Example 4.1.1:

$P_{a1}.\{officer_n^3, fire_v^4\}$

$P_{a2}.\{policeman_n^1, terminate_v^4, corruption_n^6, probe_n^1\}$

where P_i denotes the corresponding set of senses of sentence i. w_p^i denotes the i-th sense of w in WordNet with part-of-speech p. We first align the senses of $fire_v$ to all the senses of all words in a_2, and compute the maximal similarity value of the sense $fire_v^4$, $(Sim(fire_v^4, terminate_v^4) = 1)$. Next, using the achieved disambiguated semantic similarity scores, sentences that are less relevant to the input query will be filtered out.

4.1.4 Query-biased Relevance

This section describes how our sentence pruning model is applied to satisfy the query-biased relevance criterion. To be query-biased relevant, summary sentences must overlap with the query in terms of topical content.

Sentence Pruning

Let $S = \{s_1, s_2, ..., s_n\}$, be a set of sentences, and $(S_{ij})_{i,j=1,...,N}$ be the similarity matrix in which each element indicates the similarity $S_{ij} \geq 0$ between two sentences S_i and S_j (similarity scores are already achieved in this chapter). We model this data as a weighted undirected graph (similarity graph) on which each node represents a sentence and the edge weight carries the similarity of two sentences. Hence, the input query and the document sentences are considered as nodes on the graph, where we consider two kinds of edge for each node: (i) sentence-to-sentence similarity edge; (ii) sentence-to-query similarity edge. The achieved similarity weight for each sentence-to-sentence and sentence-to-query relation is assigned to its corresponding edge in our similarity graph. This graph is partially depicted in Figure 4.2.

Considering the combination of sentence-to-sentence and sentence-to-query similarities, our model decides which sentences are relevant to the query, and should be

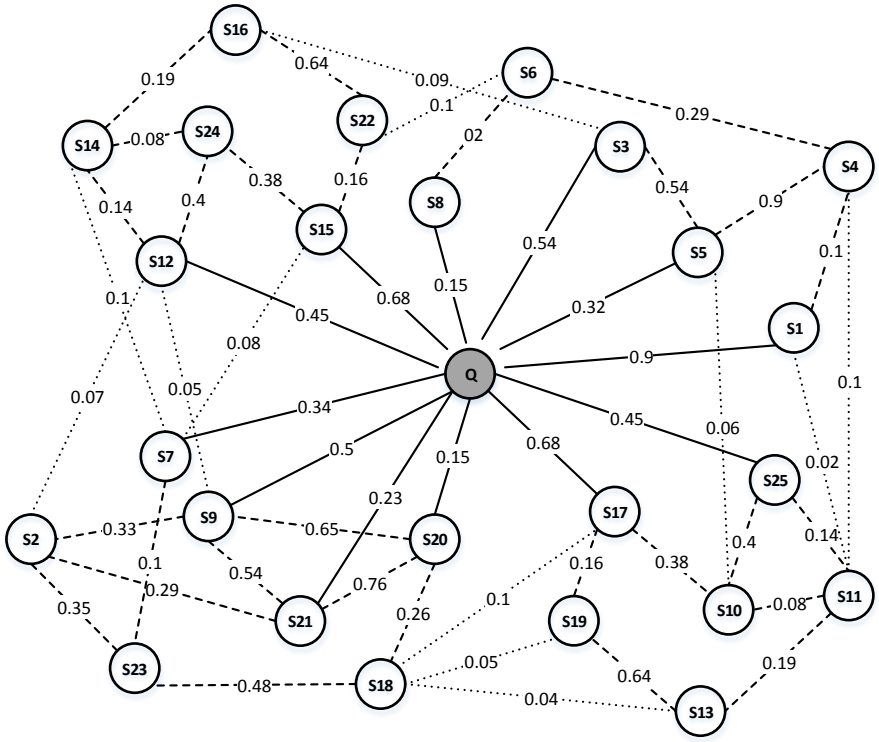


Figure 4.2: Partial view of the Similarity Graph: sentence-to-query and sentence-to-sentence similarity edges are depicted as solid and dashed lines, respectively; dotted lines reveal the loose sentence-to-sentence relations.

kept for the further clustering step. A sentence with a high sentence-to-query similarity score (direct query-biased sentence) is likely to include an answer to the query. Moreover, a sentence which may not be similar to the query, but still has a tight relation to a direct query-biased sentence, is also likely to include an answer. This idea is modeled by the following combinatorial equation (Otterbacher et al., 2005; Badrinath et al., 2011; Chali et al., 2011; Zhao et al., 2009):

$$C(S_i|Q) = \beta \times \frac{Sim(S_i, Q)}{\sum_{S_j \in D} Sim(S_j, Q)} + (1 - \beta) \times \sum_{S_k \in D} \frac{Sim(S_i, S_k)}{\sum_{S_j \in D} Sim(S_j, S_k)} \times C(S_k|Q) \tag{4.3}$$

where $C(S_i|Q)$ denotes the score of a sentence S_i given a query Q, which is determined as the sum of the similarity between the current sentence and the query, and the similarity between the current sentence and the other sentences in the document set. D also indicates the collection of sentences in the document set.

The weighting parameter $0 \leq \beta \leq 1$ is used to specify the relative contribution of two similarities: the similarity of a sentence to the query, and the similarity of a sentence to other sentences in the document set. The bigger the β, the heavier the weight for the sentence-to-query similarity. If $\beta = 0.5$, the sentence-to-sentence and the sentence-to-query similarity measures are assumed to be equally important. Previous experiments (Chali et al., 2011) lead us to choose 0.4 as the best value of β. The denominators in both terms are for normalization. $Sim(S_i, S_k)$ is the weight of the edge between two sentence nodes S_i and S_k. Likewise, $Sim(S_i, Q)$ is the weight of the edge connecting the sentence node S_i to the query node Q. Finally, sentences with $C \geq \gamma$ (with the best empirical value of 0.5 for γ) are picked among the set of sentences. This pruning step results in a subgraph comprising a set of the most query-relevant sentences to be clustered in the next step.

4.1.5 Graph-based Clustering

The clustering problem from a graph perspective, is formulated as partitioning the graph into clusters such that the edges in the same cluster have high weights and the edges between different clusters have low weights. Herein, we target hard clustering, where we partition nodes of the graph into non-overlapping clusters, i.e., let us partition S to a set of clusters $C = \{c_1, c_2, ..., c_n\}$ such that:

(i) $c_i \neq \phi \ \ for \ \ i \in \{1, ..., n\}$

(ii) $c_i \cap c_j = \phi \ \ for \ \ i, j \in \{1, ..., n\} \ \ and \ \ i \neq j$

(iii) $c_1 \cup ... \cup c_n = S$

We use a graph-based clustering algorithm namely Chinese Whispers (CW) (Biemann, 2006) for partitioning the curated subgraph. This clustering algorithm is a very basic, yet effective algorithm to partition the nodes of graphs in a bottom-up fashion. This algorithm is also a special case of Markov-Chain-Clustering (Van Dongen, 2000), but time-linear in the number of edges. Hence, the power of Chinese Whispers lies in its capability of handling very large graphs in reasonable time. Algorithm 2 shows the adopted clustering algorithm used in our work:

Algorithm 2 The Chinese Whispers Algorithm

Input: a graph $G = (S, E)$ to be clustered

Output: a clustering C of nodes in S

1: **For each** $s_i \in S$
2: $class(s_i) = i$
3: $C_i = \{s_i\}$
4: $C = \{c_i : i = 1, ..., |S|\}$
5: **repeat**
6: $C' = C$
7: **For each** $s_i \in S$, randomized order
8: $class(s_j) = \underset{c}{argmax} \sum_{\substack{\{s_i,s_j\}\in E(G) \\ s.t. class(s_i)=c}} Sim(s_i, s_j)$
9: **For each** i do $C_i = \{s_i \in S : class(s_i) = i\}$
10: $C = \{C_i : C_i \neq \phi\}$
11: **until** $C \neq C'$
12: **return** C

Line 1-4: First, a distinct class i is assigned to each node s_i, and a clustering C containing the singleton clusters c_i is created. *Line 5-11:* Then, a series of iterations is performed to merge the clusters. Specifically, at each iteration the algorithm analyzes each node s in random order and assigns it to the majority class among

those associated with its neighbors. In other words, it assigns each node s to the class c that maximizes the sum of the weights of the edges s_i, s_j incident on s_j such that c is the class of s_i, according to the following equation:

$$class(s_j) = \underset{c}{argmax} \sum_{\substack{\{s_i,s_j\}\in E(G) \\ s.t. class(s_i)=c}} Sim(s_i, s_j) \tag{4.4}$$

As soon as an iteration produces no change in the clustering (*Line 11*), the algorithm stops and outputs the final clustering (*Line 12*). The result of Chinese Whispers is a hard partitioning of the given graph into a number of clusters. It is also possible to obtain a soft partitioning in this algorithm. However, to keep the redundancy low, we prefer hard partitioning. Next, we build a word graph for each cluster, and utilize multi-sentence compression to generate abstractive summaries for the clusters.

4.1.6 Query-biased Information Novelty

A desired summary should have low information redundancy. However, in the previous section, a set of related and redundant sentences $S = \{s_1, s_2, ..., s_n\}$ have been collected in each cluster. In this section, we follow our word graph-based MSC approach proposed in Chapter 3 to build a word graph for each cluster, and satisfy the query-biased information novelty. This criterion denotes that summary sentences need to be unique, as well as responding to the demands of the query.

Abstractive Summarization of Newswire

Multi-sentence compression is a constrained form of abstractive summarization (Jing, 2000; Boudin and Morin, 2013), in which the task is replacing a collection of related sentences with a shorter sentence that captures the gist of what the related sentences

have in common, without sacrificing grammaticality. MSC is used for both summarization and text simplification. A standard way to generate summaries usually consists of the following steps: ranking sentences by their importance, clustering them by similarity, and selecting a sentence from the top ranked clusters (Wang et al., 2008b). MSC goes beyond this, composing new sentences from the clusters that may be shorter or more informative than any of their member sentences.

As already discussed in Section 3.1.1, a word graph is a directed graph $G = (V, E)$ comprising of a set of vertices or words together with a set of directed edges which shows the adjacency between corresponding nodes (Filippova, 2010b; Boudin and Morin, 2013). The graph is initiated with the first sentence, and displays its words as a sequence of connected nodes. Words are added in three steps: (i) non-stopwords for which no candidate exists in the graph; or for which an unambiguous mapping is possible; (ii) non-stopwords with multiple occurrences, or for which there are multiple possible mappings; (iii) stopwords for which we use the NLTK stopword list. The intuition behind the use of word graphs is that we can merge synonymous or redundant word nodes and use the graphs to generate more compressed sentences.

One limitation of word graphs is that they do not represent multi-word expressions, and hence MWEs lead to noise and information loss. After tokenization and stemming, we detect MWEs' components and their POS tags. We then merge the components that are associated with the same synset in WordNet 3.0 (Miller, 1995), and replace the MWEs with their available *one*-word synonyms. We only consider the most frequent *one*-word synonym in the WordNet that also appears in the other relevant sentences. If other relevant sentences contain none of the *one*-word synonyms, the most frequent one is selected.

Furthermore, we utilize WordNet synsets to find the synonym words in the content. For example, consider 3 different sentences containing words *bright*, *smart* and *brilliant*, which are synonyms of each other. Assume each sentence contains one of these

synonyms respectively. Usually, these three words will be different nodes in a word graph, but we merge the three nodes into a single node. The weight of the obtained node is computed by summing the frequency scores from the other nodes. In this way, the weight of frequent similar words with different appearances in the content is better reflected by the notion of synonymy. Hence, we make use of redundant parts to indicate the salient paths (Figure 4.3).

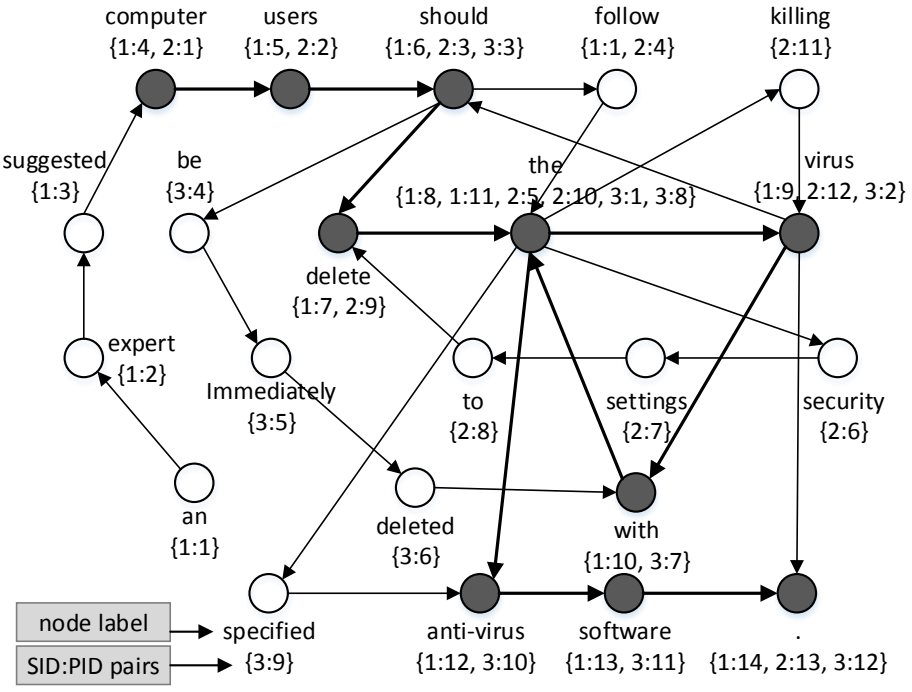


Figure 4.3: Example of our multi-sentence compression graph. Thick edges indicate the salient path, where PIDs define the order of nodes.

Edge weights are calculated using the weighting function defined in Equation 3.1. We finally use a heuristic algorithm (Boudin and Morin, 2013) to find the k-shortest

paths in the graph (with $k = 150$ throughout our experiments). In the following, we explain how to re-rank these summary candidates to favor more informative and grammatical compressions.

4.1.7 Query-biased Information Richness

Summary sentences should include as much important information as possible with respect to both the set of sentences and the input query in order to acquire query-biased information richness. To re-rank the compression candidates based on this criterion, important keyphrases have been exploited using the TextRank algorithm (Mihalcea and Tarau, 2004). The score of a keyphrase is computed by summing the salience of the words it contains, normalized with its length to favor longer n-grams (Equation 3.3). Compression candidates or paths in the word graph are then re-ranked based on the achieved scores for their keyphrases using Equation 3.4. This re-ranking step favors summary sentences conveying important information.

In a further re-ranking step, we benefit from the fact that part-of-speech tags capture the syntactic roles of words in a sentence. We use a part-of-speech based language model trained in Section 3.1.3 to assign a score to each generated summary in terms of grammatical structure (Equation 3.7). This grammar-enhanced language model helps in identifying the most grammatical sentence among the k-best compressions.

Considering the above-mentioned re-ranking scores, the normalized score of a compression candidate is given by Equation 3.8 with a different optimization settings. We will optimize the scaling factor μ on development set (Section 4.3.2) to reach the best final score. Finally, summary sentences are selected based on their sentence-to-query similarity scores, until the length constraint of query-focused multi-document summarization is reached (a 250-word summary).

To further illustrate our proposed approach in this chapter, let us consider a query to the Internet news regarding Kenya bombing. Part of the summary generated by a human, an extractive system, and our proposed abstractive summarizer (Proposed-Abs) are shown in Table 4.1 [4].

4.2 Data

We use the popular summarization benchmark datasets released by the Document Understanding Conferences[5] (DUC) that were held yearly from 2001 to 2007. In these conferences, different tasks were proposed over the years, taking into account new challenges and requirements for text summarization, also forcing systems to be dynamic and adaptable. For our experiments, we utilize DUC 2005[6], 2006[7], and DUC 2007[8]. Each of them consists of document sets and model human summaries. We have also employed OpenNLP[9] to detect and split the sentences, and Stanford POS tagger (Toutanova et al., 2003) for tokenizing and part-of-speech tagging of each sentence. Table 4.2 gives a short summary of these three datasets.

[4]Abstractive summarizers still cannot generate complete grammatical summaries. The Proposed-Abs Summary in this table is one of the most grammatical summaries generated by the proposed summarizer.

[5]http://www-nlpir.nist.gov/projects/duc/

[6]http://www-nlpir.nist.gov/projects/duc/duc2005/tasks.html/

[7]http://www-nlpir.nist.gov/projects/duc/duc2006/tasks.html/

[8]http://www-nlpir.nist.gov/projects/duc/duc2007/tasks.html/

[9]http://opennlp.sourceforge.net

Input Documents: Multiple News Sources ... 
Input Query: How were the bombings of the US embassies in Kenya and Tanzania conducted? What terrorist groups and individuals were responsible? How and where were the attacks planned?
A Human Produced Summary: Federal prosecutors in their investigation of the twin bombings of the American embassies in Kenya and Tanzania on August 7, 1998 in which more than 250 were killed, including two suicide bombers, have charged at least 14 men. All the indicted are members of Islamic extremist groups and are believed to have conspired with Saudi terrorist, Osama bin Laden. They include at least several Egyptians, ... After approximately three months of planning, the bombings were clearly well coordinated and precisely timed to coincide. Bombers destined for Kenya were dispatched through Somalia and Rwanda. ...
Extractive Summary: President Bill Clinton Thursday condemned terrorist bomb attacks at U.S. embassies in Kenya and Tanzania and vowed to find the bombers and bring them to justice. Local sources said the plan to bomb U.S. embassies in Kenya and Tanzania took three months to complete and bombers destined for Kenya were dispatched through Somali and Rwanda. Clinton met with top aides Wednesday in the White House to assess the situation following at U.S. embassies in Kenya and Tanzania which have killed more than 250 people and injured over 5,000, most of them Kenyans and Tanzanians. ...
Proposed-Abs Summary: On August 7, 1998, bombings of US Embassies in Kenya and Tanzania killed more than 250 people and over 5,000 injured. The other suspect in the bombing are members of the al-Qaeda founded by Saudi exile Osama bin Laden and Islamic Jihad Egyptian. The bomb originated in Middle East came by sea to Tanzania. Bomb U.S. embassies in Kenya and Tanzania took three months to complete and bombers destined for Kenya were through Somali and Rwanda. ...

Table 4.1: Part of the summary generated by human, an extractive system, and our summarization approach (Proposed-Abs) for topic D0626H (DUC 2006). Greyed out parts in extractive summary, are query-irrelevant phrases, such as temporal information or source of the news, and also redundant parts which have been automatically removed in the Proposed-Abs.

Features	DUC 2005	DUC 2006	DUC 2007
number of clusters	50	50	45
documents/clusters	25-50	25	25
summary length	250 words	250 words	250 words

Table 4.2: Information about the utilized DUC datasets

4.3 Experiments

4.3.1 Evaluation Metrics

The evaluation of automatically generated summaries is a critical issue due to the subjectivity in deciding what the evaluation criteria should be (Radev et al., 2003). The evaluation process may be performed manually, and require human judges to decide whether or not a summary is of good quality. Hence, manual evaluation is very costly and time-consuming. Besides, to objectively judge a summary has been proven difficult, as humans often disagree on what exactly makes a summary of good quality (Jones and Galliers, 1995). Thereupon, for our summarization approach, we automatically assess the generated summaries through ROUGE metrics Lin (2004).

ROUGE is a commonly used evaluation method to measure the summary quality by counting the overlapping units between system-generated peer summaries and human-written model/gold summaries. ROUGE measures the concordance of candidate and model summaries by determining n-grams, word sequences, and word pair matches. ROUGE metrics produce a value in [0, 1], where higher values are preferred, as they indicate a greater content overlap between the generated summary and model summaries. We use ROUGE F-measure for unigrams, bigrams, and SU4 (skip-bigrams with maximum gap length 4) to evaluate the generated summaries.

An important drawback of ROUGE metrics is that they use lexical matching instead

of semantic matching. Therefore, generated summaries that are worded differently but carry the same semantic information may be assigned different ROUGE scores (Plaza et al., 2011). In contrast, the main advantages of ROUGE are its simplicity and high correlation with human judgments (Lin, 2004).

4.3.2 Experiment Results

To investigate the effectiveness of our proposed approach for summarizing newswire, we conduct experiments using one of the competitive extractive query-focused multi-document summarizers - RelationListwise by Yin et al. (2012). This summarization system integrates relation information among all sentences into a list, and mainly considers some individual features: i.e., query-biased relevance, biased information novelty, and richness. Top three systems with the highest ROUGE scores that participated in DUC 2005 (S4, S15, S17), DUC 2006 (S12, S23, S24), and DUC 2007 (S4, S15, S29) are also compared with our system.

In an ablation study, the effectiveness of multi-sentence compression along with the re-ranking algorithms, is studied using the following experiments. We keep consistency for our algorithm framework except to omit the graph-based clustering and multi-sentence compression steps, and converting our abstractive summarization approach to the ranking-based extractive approach (Proposed-Ext). For more clarity, we have conducted only two first components of our approach, which are capturing sentence-to-sentence and sentence-to-query semantic similarities, and also sentence pruning step to achieve the most query-relevant sentences. Then we compare Proposed-Ext with our proposed abstractive approach (Proposed-Abs). Table 4.3, Table 4.4, and Table 4.5 show the experiment results on DUC 2005[10], DUC 2006, and DUC 2007, respectively.

[10]No experiment results on DUC 2005 were reported by RelationListwise

System	**Rouge-1**	**Rouge-2**	**Rouge-su4**
RelationListwise	-	-	-
S17	0.36933	0.07286	0.12937
S4	0.37584	0.07063	0.12868
S15	0.37656	0.07383	0.13248
Proposed-Ext	0.37106	0.07219	0.12963
Proposed-Abs	0.41980	0.08725	0.13941

Table 4.3: Evaluation on DUC 2005 Dataset

System	**Rouge-1**	**Rouge-2**	**Rouge-su4**
RelationListwise	0.43066	0.10852	0.16324
S23	0.40973	0.09785	0.16162
S12	0.41253	0.09633	0.16074
S24	0.41081	0.09957	0.15248
Proposed-Ext	0.41102	0.09722	0.16019
Proposed-Abs	0.44871	0.14208	0.17602

Table 4.4: Evaluation on DUC 2006 Dataset

System	**Rouge-1**	**Rouge-2**	**Rouge-su4**
RelationListwise	0.45852	0.13091	0.17824
S4	0.43603	0.11785	0.17162
S29	0.43159	0.12048	0.17374
S15	0.44481	0.12907	0.17748
Proposed-Ext	0.43564	0.12410	0.17391
Proposed-Abs	0.47641	0.15415	0.18753

Table 4.5: Evaluation on DUC 2007 Dataset

The statistics point out the superiority of our proposed approach to the compared systems on all evaluation metrics. Hence, the overall results support our hypothesis that query-focused abstractive summarization using the underlying textual semantic similarities while considering the grammatical structure of the generated summaries, results in more query-relevant, informative, and promising summaries. Besides,

the results achieved by Proposed-Ext on DUC benchmark datasets still show some improvements over some of the baseline systems. The reason might be capturing textual semantic similarities, which helps to select the most query-relevant sentences among the set of news documents.

Standard Deviation of Rouge Scores

Tables 4.3, 4.4, and 4.5 have demonstrated the average performance of the proposed summarization approach. Therefore, an important research question that immediately arises is how much the ROUGE scores differ across each of the datasets. To answer this question, the standard deviation of three variants of ROUGE scores for the summaries generated by Proposed-Abs are computed and reported in Table 4.6.

Dataset	**Rouge-1**	**Rouge-2**	**Rouge-su4**
DUC 2005	0.03017	0.00946	0.00483
DUC 2006	0.00952	0.01705	0.01008
DUC 2007	0.01263	0.01528	0.00773

Table 4.6: Standard deviation of ROUGE scores for the summaries generated by Proposed-Abs across DUC 2005, DUC 2006, and DUC 2007 datasets

Syntactic Analysis of the Generated Sentences

In this section, we employ version 5.3.7 of Link Grammar Parser[11] to analyze a random selected part of the generated summaries in terms of syntactic structure. An example of the parser analysis performed for a sample generated sentence *"Bombings of US Embassies in Kenya and Tanzania killed more than 250 people and over 5,000 injured."*, is shown in Figure 4.4.

[11]http://www.abisource.com/projects/link-grammar/

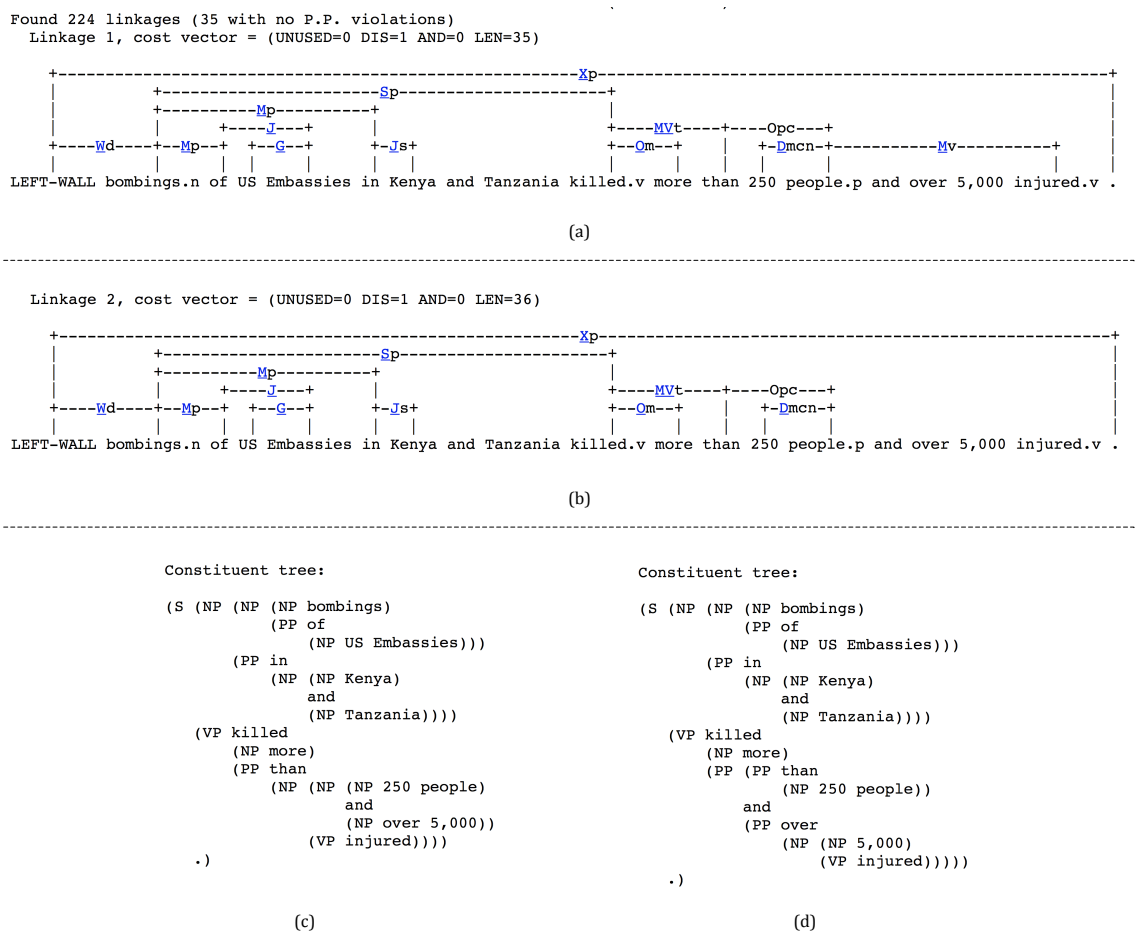


Figure 4.4: Example of using Link Grammar Parser for the syntactic analysis of a sample generated sentence; (a) and (b) demonstrate two found complete linkages with no p.p. violations. The constituent trees (c) and (d) correspond to linkages (a) and (b), respectively.

Link Grammar parser is a syntactic analyzer of English language developed at the Carnegie Mellon University (Sleator and Temperley, 1995). Having received a sentence, the system attributes it with a syntactic structure which consists of a set of marked links connecting the pairs of words. It includes approximately 60000 dictionary forms, and can skip a part of a sentence it cannot understand and define some structure for the rest of the sentence. It is capable of processing an unknown lexicon and doing reasonable assumptions about the syntactic category of unknown words based on the context and writing. The parser contains data about various names,

numerical expressions, and punctuation marks.

Link grammar differs from traditional dependency grammars by allowing cyclic relations between words. Therefore, for example, there can be links indicating both the head verb of a sentence, the head subject of the sentence, as well as a link between the subject and the verb. These three links thus form a cycle (a triangle, in this case). Cycles are useful in constraining what might otherwise be ambiguous parses, and can help "tighten up" the set of allowable parses of a sentence.

We have performed a random selection of generated summary sentences among the set of high ranked ones by our par-of-speech based language model, to syntactically analyze them using the Link Grammar, and consequently show the effectiveness of our grammar-enhanced re-ranking step. The parser gives a constituent representation of a sentence, labeling noun phrases, verb phrases, clauses, etc.. Hence, the constituent representation is derived from the linkage. The parser does not consider a sentence to be *grammatical*, just because it finds a valid linkage for that sentence. The linkage must satisfy a post-processing phase. The parser indicates the post-processing status with messages like *"Found 2 linkages (1 with no P.P. violations)"*. If all of the linkages at one stage have post processing violations, the parser continues looking for a satisfactory linkage in the next phase.

If there is more than one satisfactory linkage, the parser orders them according to certain simple heuristics. The cost vector determines the ordering used. This vector has three components. The first component (most significant in the ordering) is the total cost of all the usages of words in the linkage. The dictionary assigns different costs to the usages of a word; while most usages have cost nothing, some have non-zero cost. The second component has to do with the relative size of components combined by conjunctions. The third component is the total length of all links in the linkage. Figure 4.4 demonstrates this process for a sample generated sentence. For this example, the Link Grammar finds 224 complete linkages 35 of which with

no p.p violations, which indicates that this newly generated summary sentence is grammatically correct.

This analysis shows about 88% precision over the syntactic structure of summary sentences. In details, as shown in Figure 4.5, among 600 random-selected summary sentences (200 generated sentences of each analyzed DUC dataset), 528 sentences have been shown to have at least one complete linkage with no P.P violations, 54 sentences (9%) have a number of linkages but with some P.P violations, and finally, Link Grammar parser cannot find any linkage for 18 sentences (3%).

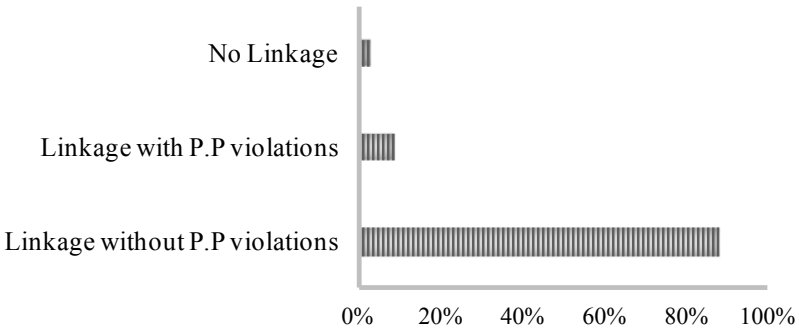


Figure 4.5: Syntactic analysis of a number of 600 generated summary sentences using Link Grammar Parser

Exploring Scaling Factor

In Section 4.1.7, a free parameter is discussed (μ in Equation 3.8). We randomly selected 30% of each DUC dataset as a development set to tune this parameter. Figure 4.6 shows the results obtained by ROUGE-1 F-Measure, using different values for μ. The best average result is observed while μ is between 0.4 and 0.5. Hence, we consider value of 0.45 to optimize μ. Performance deteriorates when the value of μ approaches 1.0 which indicates the system performance without any contribution of grammar-enhanced re-ranking step. Decreasing the weight of μ to zero causes the exclusion of keyphrase re-ranking step, and consequently ignorance of exploiting

important information. This demonstrates the importance of using both scores in appropriate re-ranking of the generated summary sentences.

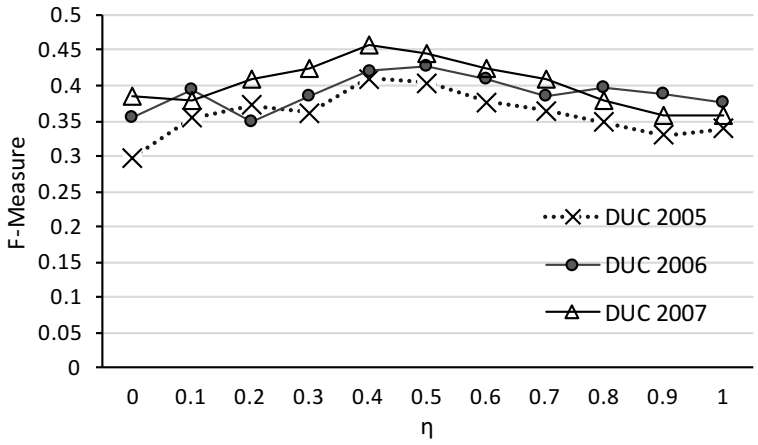


Figure 4.6: Exploring scaling factor μ in Equation 3.8 on the development set

4.4 Summary

In this chapter, we proposed an abstractive query-focused summarizer for newswire. Given a query and a set of news documents, our approach summarizes the documents to answer the query with the aim of satisfying query-biased relevance, biased information novelty, and biased information richness. For this purpose, sentence-to-sentence and sentence-to-query semantic similarities are captured by performing repetitive random walks over WordNet. Furthermore, less query-relevant sentences are filtered out, and a well-organized and informative summary is generated (through an MSC word graph) for each of the clusters of query-relevant sentences. This component considers the important keyphrases, along with the grammatical structure of the generated summaries. We also studied the importance of separate components in our approach by conducting a set of experiments in Section 4.3.2, where we used automatic evaluation metric over the DUC benchmark datasets. The overall

experiment results showed that our method outperforms the competitive baselines. The next chapter presents the first attempt to appraise the coverage of knowledge sources for domain-specific summarization, where we focus on the task of query-biased multi-document summarization for clinical texts.

Chapter 5

Domain-specific Multi-document Summarization

In this chapter, we intend to explore the coverage of general and domain specific knowledge sources for the purpose of abstractive summarization. Among the existing specific domains, we focus on the summarization of the vast body of medical evidence. The growth of content of medical evidence requires development of effective summarization techniques to provide required information to physicians and researchers. Given a clinical query and a set of relevant medical evidence, our aim is to generate a fluent, well-organized, and compact summary that answers the clinical query. This chapter contributes to enhance the quality of biomedical summaries by appraising the applicability of WordNet as a general-purpose lexicon to capture the concepts not covered by the UMLS Metathesaurus. Figure 5.1 provides an overview of the proposed approach. Herein, our approach is adopted into the summarization framework proposed in Chapter 4 to effectively summarize clinical text. We explore the utility of the graph representation of both general (WordNet) and domain-specific (UMLS) lexicons for incorporating underlying textual semantic

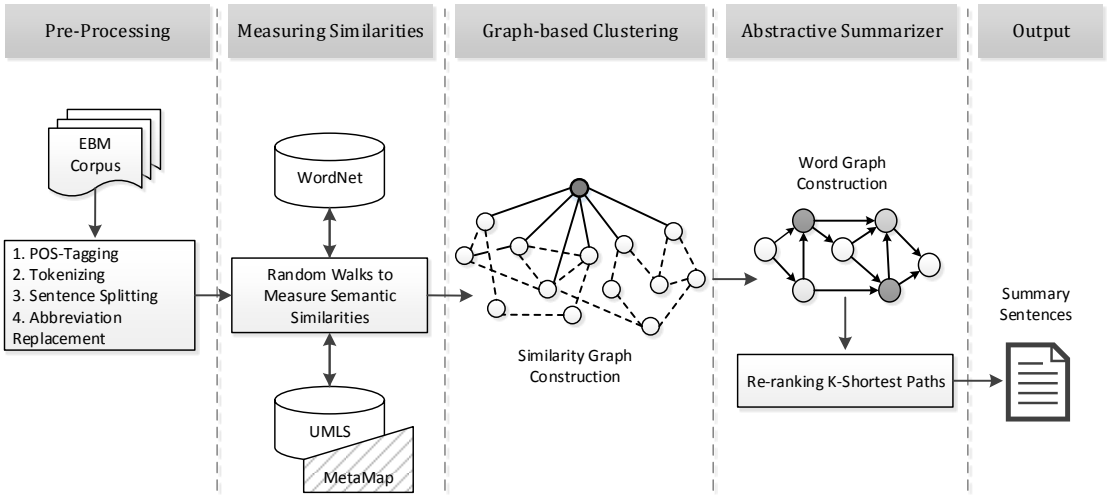


Figure 5.1: Overview of the proposed framework

similarities.

In Section 5.1, we discuss these resources, their distinctions, and the employed Evidence-based Medicine (EBM) corpus. Preprocessing step is explained in Section 5.2. We demonstrate the proposed approach in Section 5.3. Section 5.4 reports the evaluation metrics and the performed experiments. Finally, Section 5.5 concludes this chapter.

5.1 Data

In this chapter, we have utilized two knowledge sources: Unified Medical Language System (UMLS[1]) developed by the U.S. National Library of Medicine (Bodenreider, 2004), and WordNet[2] (Fellbaum, 1998) for concept discrimination. We have also employed the data provided in an EBM corpus by Mollá et al. (2015) to develop,

[1]http://www.nlm.nih.gov/research/umls/

[2]http://wordnet.princeton.edu/

test, and evaluate our summarization approach.

5.1.1 UMLS

UMLS (Bodenreider, 2004) is a database of biomedical vocabularies developed by the U.S. National Library of Medicine. In our approach, we have utilized version 2015AB of the UMLS Metathesaurus that contains more than 3.25 million concepts, and nearly 13 million unique concept names from over 190 source vocabularies. The three major components of UMLS are the Metathesaurus, Semantic Network and SPECIALIST Lexicon. This work focuses on the Metathesaurus which semi-automatically integrates information about biomedical and health-related concepts from various biomedical and clinical sources.

UMLS uses 12 different types of hierarchical and non-hierarchical relations between concepts. For instance, the hierarchical relations consist of the *parent/child* and *broader/narrower* (BR/NR) relations. We utilize version 2016 of MetaMap[3] program (Aronson, 2006) for mapping biomedical text to concepts in the UMLS Metathesaurus. MetaMap employs a knowledge-intensive approach that uses the SPECIALIST Lexicon in combination with lexical and syntactic analysis to identify noun phrases in text.

Matches between noun phrases and Metathesaurus concepts are computed by generating lexical variations and allowing partial matches between the phrase and the concept. The possible UMLS concepts are assigned scores based on the closeness of the match between the input noun phrase and the target concept. The highest scoring concepts and their semantic types are gradually returned.

[3]Developed by the U.S. National Library of Medicine; available at https://metamap.nlm.nih.gov/

5.1.2 WordNet

WordNet is a large general-purpose lexical database of English, which is often used in word sense discrimination. Words are grouped into sets of synonyms called synsets, each of which expressing a distinct concept. We have used WordNet 3.0 repository (Fellbaum, 1998) for the current study, that includes a total of 155,287 words organized in 117,659 concepts, which are linked by semantic and lexical relations.

5.1.3 UMLS vs. WordNet

Although WordNet includes a certain number of medical terms, UMLS is used extensively for medical text mining and retrieval. A study performed by Bodenreider (2004) shows that the concept overlap between WordNet and UMLS varies from 48% to 97%. This is because UMLS records the variability of the lexical forms encountered in the source vocabularies, while WordNet only records the canonical forms. WordNet and UMLS are also different in their graph structures. Therefore, there exists a huge discrepancy in granularity between WordNet and UMLS (Lu, 2015). For example, as shown in Figure 5.2, *malignant_tumor.n.01* is the parent of *cancer.n.01* in WordNet, but *malignant tumor* and *cancer* locate in the same concept *C0006826* (*malignant neoplasms*) in UMLS. In this example, WordNet has a finer granularity. However, UMLS possesses a finer granularity in some other cases.

While UMLS is a very rich source of information on medical and biological terms and concepts, it does not provide full coverage of non-medical concepts, terms and relations (Hogan, 2007; Burgun and Bodenreider, 2001; Huang et al., 2009; Mougin et al., 2006). In this chapter, we have utilized WordNet to represent the layman knowledge, and UMLS to represent the professional knowledge. Our goal is to capture sentence-to-query and sentence-to-sentence semantic similarities, and bridge

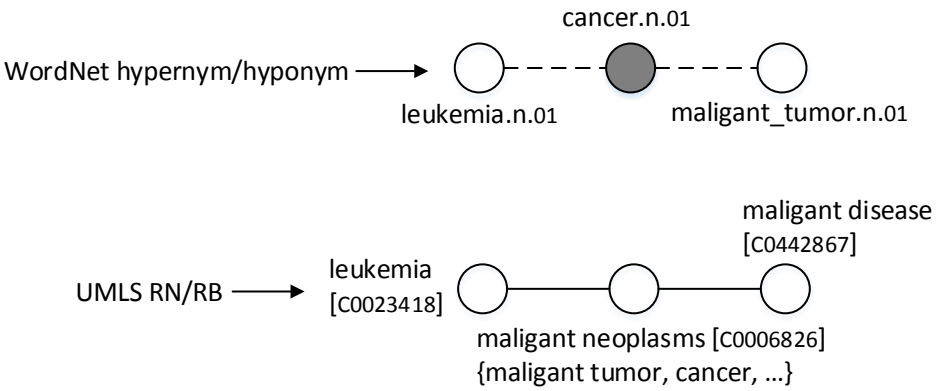


Figure 5.2: Example of difference between WordNet and UMLS

the knowledge and language gaps in biomedical summarizers.

5.1.4 EBM Corpus

At the time of writing this thesis, the corpus released by Mollá et al. (2015) is the only available corpus[4] for the task of evidence-based medicine text summarization. This corpus is sourced from the Clinical Inquiries section of the Journal of Family Practice[5]. Each article in this section of the journal (issued monthly) addresses a clinical question, and provides a systematic analysis of the best available medical evidence in response to the posed clinical query. For each question, this corpus contains the following information:

- **The URL of the clinical inquiry:** An address, from which the information has been sourced.

[4]http://sourceforge.net/projects/ebmsumcorpus/

[5]http://www.jfponline.com/articles/clinical-inquiries.html/

- **The question:** For example, "*What is the evaluation and treatment strategy for Raynaud's phenomenon?*".
- **The bottom-line evidence-based answer:** The answer may contain several parts, since a question may be answered according to distinct pieces of evidence. For each part, the corpus includes a short description of the answer, the Strength of Recommendation (SOR) grade of the evidence related to the answer, and a short description that explains the reasoning behind allocating such a SOR grade.
- **The answer justifications:** For each of the parts of the evidence-based answer, there is one or more justifications describing the actual findings reported in the research papers supporting the answer.
- **The references:** Each answer justification includes one or more references to the source research paper. Each reference includes the PubMed[6] ID and full abstract information as encoded in PubMed, if available.

This corpus consists of 456 clinical queries, with 1396 bottom-line, multi-document summaries (i.e., evidence-based answers). The total number of single-document evidence-based summaries is 3036, which are generated from 2908 unique articles. The corpus also contains XML versions of these articles, obtained from PubMed. We have utilized this corpus to develop and test our query-focused multi-document summarization framework. The bottom-line answers are used as the reference (gold) summaries. The question and all the abstracts associated with the bottom-line summary are also considered as the source texts. Table 5.1 lists the properties of this corpus, and Table 5.2 provides an example of query-focused multi-document summarization over this corpus.

[6]PubMed is a free search engine accessing primarily the MEDLINE database of references and abstracts on life sciences and biomedical topics, provided by the U.S. National Library of Medicine; available at https://www.ncbi.nlm.nih.gov/pubmed/

total #clinical queries	456
#bottom-line multi-document summaries	1396
#single-document evidence summaries	3036
total #unique articles	2908

Table 5.1: Information about the EBM Corpus

5.2 Preprocessing

5.2.1 Biomedical Domain Peculiarities

Biomedical texts exhibit certain unique attributes that must be taken into account in the development of a summarization system. First, medical information arises in a wide range of document types (Afantenos et al., 2005) like electronic health records, scientific articles, semi-structured databases, X-ray images and even videos. Each document type presents very distinct characteristics that should be considered in the summarization process. We focus on scientific articles, which are mainly composed of text. Having knowledge about the article layout can be exploited to improve the summaries that are generated automatically (Plaza et al., 2011). Second, the specific nature of biomedical terminology makes it difficult to automatically process biomedical information. Some of the discussed issues are as follows:

- **Synonyms:** The use of different terms to designate the same concept.
- **Homonyms:** The use of words/phrases with multiple meanings. For instance, the syntagms *coronary failure* and *heart attack* stand for the same concept, while the term *anaesthesia* may refer to either the *loss of sensation* or the procedure for *pain relief*.
- **Neologisms:** Newly coined words that are not likely to be found in a dictionary (e.g., the term *coumadinise* for the administration of coumadin).

Question: How should we manage a patient with a positive PPD and prior BCG vaccination?

Bottom-line answer (multi-document summary): A recently developed alternative is the interferon-gamma assay (QuantiFERON-TB Gold test), which may be used in place of, or in addition to, the PPD skin test for patients who are known to have received a BCG vaccine. [*PubMed IDs: 15059788, 16539718*]

Source text 1 [PMID: 15059788]: The tuberculin skin test for immunologic diagnosis of Mycobacterium tuberculosis infection has many limitations, including being confounded by bacillus Calmette-Gurin (BCG) vaccination or exposure to nontuberculous mycobacteria. M. tuberculosis-specific antigens that are absent from BCG and most nontuberculous mycobacteria have been identified. We examined the use of two of these antigens, CFP-10 and ESAT-6, in a whole blood IFN-gamma assay as a diagnostic test for tuberculosis in BCG-vaccinated individuals. Because of the lack of an accurate standard with which to compare new tests for M. tuberculosis infection, specificity of the whole blood IFN-gamma assay was estimated on the basis of data from people with no identified risk for M. tuberculosis exposure (216 BCG-vaccinated Japanese adults) and sensitivity was estimated on the basis of data from 118 patients with culture-confirmed M. tuberculosis infection who had received less than 1 week of treatment. Using a combination of CFP-10 and ESAT-6 responses, the specificity of the test for the low-risk group was 98.1% and the sensitivity for patients with M. tuberculosis infection was 89.0%. The results demonstrate that the whole blood IFN-gamma assay using CFP-10 and ESAT-6 was highly specific and sensitive for M. tuberculosis infection and was unaffected by BCG vaccination status.

Source text 2 [PMID: 16539718]: The whole-blood interferon-gamma release assay (IGRA) is recommended in some settings as an alternative to the tuberculin skin test (TST). Outcomes from field implementation of the IGRA for routine tuberculosis (TB) testing have not been reported. We evaluated feasibility, acceptability, and costs after 1.5 years of IGRA use in San Francisco under routine program conditions. Patients seen at six community clinics serving homeless, immigrant, or injection-drug user (IDU) populations were routinely offered IGRA (Quantiferon-TB). Per guidelines, we excluded patients who were 17 years old, HIV-infected, immunocompromised, or pregnant.

We reviewed medical records for IGRA results and completion of medical evaluation for TB, and at two clinics reviewed TB screening logs for instances of IGRA refusal or phlebotomy failure. Between November 1, 2003 and February 28, 2005, 4143 persons were evaluated by IGRA. 225(5%) specimens were not tested, and 89 (2%) were IGRA-indeterminate. Positive or negative IGRA results were available for 3829 (92%). Of 819 patients with positive IGRA results, 524 (64%) completed diagnostic evaluation within 30 days of their IGRA test date. Among 503 patients eligible for IGRA testing at two clinics, phlebotomy was refused by 33 (7%) and failed in 40 (8%). Including phlebotomy, laboratory, and personnel costs, IGRA use cost $33.67 per patient tested. IGRA implementation in a routine TB control program setting was feasible and acceptable among homeless, IDU, and immigrant patients in San Francisco, with results more frequently available than the historically described performance of TST. Laboratory-based diagnosis and surveillance for M. tuberculosis infection is now possible.

Table 5.2: Example of query-focused multi-document summarization, showing the question, the bottom-line summary and two of the source abstracts.

- **Elisions:** The omission of words or sounds in a word or phrase. For example, *white count* which is understood by physicians as the *count of white blood cells*.
- **Abbreviations:** A shortened form of a word or phrase. For example, the use of *OCT* to refer to *Optical Coherence Tomography*.

5.2.2 Preprocessing Steps

In our approach, if the abstract includes abbreviations, the abbreviations and their expansions are extracted. This information is then used to replace these shortened forms in the abstract body. For example, if the abbreviation defines *Autologous Bone Marrow Transplantation* as the expansion of *ABMT* for a particular abstract, this abbreviation would be replaced by *Autologous Bone Marrow Transplantation* anywhere else in the document body.

If the abstract contains abbreviations and acronyms, but without any definition, the software[7] for abbreviation definition recognition presented by Schwartz and Hearst (2002) is used. This software allows for the identification of abbreviations and their expansions in biomedical texts with an average precision of 95%. Abbreviations are then replaced by their expansions in the document body.

Furthermore, we have used the stopword list included in NLTK extended with the PubMed stopwords[8] to remove the generic terms (e.g., prepositions and pronouns), which are not useful in our summarization process. We have also employed OpenNLP[9] to detect and split the sentences, and Stanford POS tagger (Toutanova et al., 2003) for tokenizing and part-of-speech tagging of each sentence.

5.3 Proposed Approach

Many existing approaches to automatic summarization rely on comparing the similarity of two sentences in some ways. Most existing relatedness measures are based on knowledge sources such as concept hierarchies or ontologies. For general English text, research on measuring relatedness has relied on WordNet, a freely available database that can also be viewed as a semantic network. For clinical and biomedical vocabularies, they are compiled into UMLS, a large lexical and semantic database of medical terms maintained by the U.S. National Library of Medicine.

In Example 4.1.1, we discussed hard matching between words as an obstacle in identifying the relatedness of two sentences in *General Domain* (Yin et al., 2012). The following example illustrate such problem in *Biomedical Domain*:

[7]http://biotext.berkeley.edu/software.html/

[8]http://www.ncbi.nlm.nih.gov/books/NBK3827/table/pubmedhelp.T.stopwords/

[9]http://opennlp.sourceforge.net/

Example 5.3.1. Biomedical Domain:

(i) *Cerebrovascular diseases during pregnancy result from any of three major mechanisms: arterial infarction, haemorrhage or venous thrombosis.*

(ii) *Brain vascular disorders during gestation result from any of three major mechanisms: arterial infarction, haemorrhage or venous thrombosis.*

Because the two sentences present different terms, surface-based approaches are unable to make use of the fact that they have exactly the same meaning. We have solved this problem by leveraging a UMLS-based approach dealing with concepts instead of terms, and with semantic relations instead of lexical or syntactical ones.

In our approach, the main requirement for computing semantic similarities on WordNet and UMLS is *Semantic Signature*, which is a multinomial distribution generated from repeated random walks on WordNet (Pilehvar and Navigli, 2015). In Section 4.1.1, we utilized this concept to capture semantic similarities and disambiguate words on WordNet. Next, we explain how to capture semantic similarities on UMLS.

5.3.1 Semantic Similarities on UMLS

To construct each semantic signature on UMLS, we employ a graph-based algorithm to perform iterative random walks over the graph representation of the UMLS Metathesaurus. A variant of this algorithm has previously been utilized for query expansion (Martinez et al., 2014). The UMLS Metathesaurus contains a wide range of information about the relations between terms in the form of database tables. The MRREL table lists relations between concepts (i.e., *parent*, *can be qualified by*, and *related and possibly synonymous*) among others. Concepts in UMLS are considered as nodes (seeds), and the relations listed in the MRREL table as directed

edges. No weights are used for the relations that are extracted from the MRREL table.

We have used the MetaMap program to map each sentence to concepts from the UMLS Metathesaurus and semantic types from the UMLS Semantic Network. A broad range of concepts from very generic UMLS semantic types are discarded in this step for two reasons: (i) the generic concepts have already been considered in capturing WordNet-based semantic similarities; (ii) to reduce the size of UMLS graph, and consequently reduce the run time of iterative random walks. Following Plaza et al. (2011), these semantic types are defined as *quantitative concept, qualitative concept, temporal concept, functional concept, idea or concept, intellectual product, mental process, spatial concept,* and *language*. Therefore, only concepts of the rest of semantic types are considered for constructing semantic signatures. Table 5.3 provides an example of mapping a sentence by MetaMap.

Score	**Concept**	**Semantic Type**	**Considered**
862	No evidence of	Qualitative Concept	✘
593	Increase	Functional Concept	✘
593	Risk	Idea or Concept	✘
578	Major	Qualitative Concept	✘
744	Hemorrhage	Pathologic Function	✔
578	Result	Functional Concept	✘
578	Accidental Falls	Injury or Poisoning	✔
1000	Hospitalized Patients	Patient or Disabled Group	✔
966	Take	Health Care Activity	✔
1000	Warfarin	Pharmacologic Substance	✔

Table 5.3: MetaMap mapping for the sentence "*There is no evidence of increased risk for major bleeding as a result of falls in hospitalized patients taking warfarin.*"

Same as WordNet-based semantic signature, the UKB implementation of Personalized PageRank is utilized, but this time on UMLS. Consider an adjacency matrix X with all relations in MRREL, for the UMLS graph. The random walker starts in any

of the concepts included in the sentence, and follows at random one of the relations to another concept. With certain probability, the random walker would restart in any of the concepts, and continue its walk. Finally, the number of visits to each concept in the graph would give an indication of how related that concept is to the sentence terms. The result is a probability distribution over UMLS concepts. The higher the probability for a concept, the more related it is to the given sentence.

The probability distribution for the starting location of the random walker in the network is denoted by $\vec{u}^{(0)}$. Released the set of MetaMap concepts C in a sentence, the probability mass of $\vec{u}^{(0)}$ is uniformly distributed across the concepts $c_i \in C$, with the mass for all $c_i \notin C$ set to zero. The PageRank vector is then computed using the following equation:

$$\vec{u}^{(t)} = (1 - \beta)X\vec{u}^{(t-1)} + \beta\vec{u}^{(0)} \tag{5.1}$$

where at each iteration, the random walker may jump to any node $c_i \in C$ with probability $\beta/|C|$. Following the standard convention, the value of β is set to 0.15. The number of iterations is also set to 30, which is sufficient for the distribution to converge. The resulting probability vector $\vec{u}^{(t)}$ is the semantic signature of the sentence on UMLS, as it has aggregated its concepts' similarities over the entire graph.

5.3.2 UMLS-based Semantic Disambiguation

Using the built-in WSD module, MetaMap allows to disambiguate terms and return directly the relevant concept. For better clarity, we run MetaMap to find the UMLS concepts for the term *cold* (Figure 5.3). Normally, four concepts in MetaMap are assigned to this term. When WSD module is turned on, only one concept will be

returned by considering the terms included in the given sentence. Then, a uniform probability distribution is assigned to the concepts found in each sentence. The rest of the nodes are initialized to zero.

```
Processing 00000000.tx.1: cold

Phrase: cold
>>>>> Phrase
cold
<<<<< Phrase
>>>>> Variants
cold [noun] variants (n=1):
1: cold{[noun], 0=[]}

<<<<< Variants
>>>>> Candidates
Meta Candidates (Total=4; Excluded=0; Pruned=0; Remaining=4)
  1000   C0009264:Cold (Cold Temperature) [Natural Phenomenon or Process]
  1000   C0009443:COLD (Common Cold) [Disease or Syndrome]
  1000   C0041912:Cold (Upper Respiratory Infections) [Disease or Syndrome]
  1000   C0234192:Cold (Cold Sensation) [Physiologic Function]
<<<<< Candidates

>>>>> Mappings

Meta Mapping (1000):
  1000   C0234192:Cold (Cold Sensation) [Physiologic Function]

Meta Mapping (1000):
  1000   C0009264:Cold (Cold Temperature) [Natural Phenomenon or Process]

Meta Mapping (1000):
  1000   C0009443:COLD (Common Cold) [Disease or Syndrome]

Meta Mapping (1000):
  1000   C0041912:Cold (Upper Respiratory Infections) [Disease or Syndrome]
<<<<< Mappings
```

Figure 5.3: Screenshot of Mapping the term *cold* using MetaMap

5.3.3 Semantic Similarities at Sentence Level

For comparing pairs of semantic signatures at sentence level, we have used Weighted Overlap algorithm by Pilehvar and Navigli (2015). This algorithm first sorts the two signatures according to their values and then harmonically weights the overlaps

between them. The weighting process is such that differences in the highest ranks are penalized more than differences in lower ranks, while the first-ranked element has the highest rank. Using the knowledge source N (i.e., WordNet or UMLS), Weighted Overlap calculates the semantic similarity (Sim_N) of two sentence signatures S_{N1} and S_{N2} as:

$$Sim_N(S_{N1}, S_{N2}) = \frac{\sum_{h \in H} (r_h(S_{N1}) + r_h(S_{N2}))^{-1}}{\sum_{i=1}^{|H|} (2i)^{-1}} \tag{5.2}$$

where H denotes the intersection of all senses/concepts with non-zero probability (dimension) in both signatures, and $r_h(S_{Nj})$ denotes the rank of the dimension h in the sorted signature S_{Nj}, where rank 1 denotes the highest rank. The denominator is also used as a normalization factor that guarantees a maximum value of one. The minimum value is zero and occurs when there is no overlap between the two signatures, i.e., $|H| = 0$.

To estimate the final semantic similarity score between two sentences, we have conducted a set of experiments using the WordNet-based semantic similarities (Sim_W), and/or UMLS-based semantic similarities (Sim_U), and obtained the best result while using both scores with different weights according to the following equation:

$$Sim_{final}(S_1, S_2) = \eta \times Sim_U(S_{U1}, S_{U2}) + (1 - \eta) \times Sim_W(S_{W1}, S_{W2}) \tag{5.3}$$

where $Sim_U(S_{U1}, S_{U2})$ denotes the semantic similarity score between two sentence signatures on UMLS. The semantic similarity score between two sentence signatures on WordNet is also shown by $Sim_W(S_{W1}, S_{W2})$. Finally, the scaling factor η was optimized on development data in our experiments and set to 0.6 to reach the best result (Section 5.4).

Next, using the achieved semantic similarity score for each pair of sentences, sentences which are less or not relevant to the clinical query will be pruned, and remained sentences will be clustered according to their relevance to each other.

5.3.4 Sentence Pruning and Clustering

For keeping the most relevant sentences to the input query, sentences are modeled as a weighted undirected graph (similarity graph - explained completely in Section 4.1.4), in which each node represents a sentence and each edge weight scores the similarity of two sentences. Considering the combination of sentence-to-sentence and sentence-to-query similarities (Equation 4.3), we then prune the graph, passing on a subset of the input sentences to the subsequent clustering step.

The graph-based clustering algorithm we have used in this step, is CW (explained in Section 4.1.5) proposed by Biemann (2006). First, a distinct class is assigned to each node. Then, a series of iterations is performed aimed at merging the clusters. Specifically, it assigns each node to the class that maximizes the sum of the weights of the edges. As soon as the algorithm converges, producing no further merge operations, we output the final clustering.

Clustering Potential of the EBM Corpus

Each query in the corpus is accompanied with multiple candidate replies. Since each candidate reply is referred to a set of abstracts, their released corpus could be utilized for the task of clustering (Mollá et al., 2015). This ability is appreciated by an example shown in Table 5.4. However, we desire to consider a set of abstracts as a bag of sentences, pick the query-related sentences, and finally collect the relevant ones into a set of clusters. Hence, each cluster in our work is likely to include a set

of sentences from different clusters defined in the corpus. Hence, we do not use the clustering potential of corpus in our approach.

Question: What is the evaluation and treatment strategy for Raynaud's phenomenon?
Abstract IDs: 12814733, 12814733, 12324557, 11392916, 15865744, 10796398, 11508437
Resulting Clusters: Cluster1 → 12814733, 12814733, 12324557 Cluster2 → 11392916 Cluster3 → 15865744, 10796398, 11508437

Table 5.4: Example of Clustering Potential of the Utilized Corpus

Next, we build a MSC word graph for each cluster, and generate one sentence as an abstractive summary of each cluster.

5.3.5 Abstractive Summarization of Medical Evidence

As discussed in Section 3.1.1, a word graph is a directed graph constructed by iteratively adding sentences to it. Conducting this step not only removes the redundancy, but also makes use of redundant parts to indicate the salient paths. An example of the constructed MSC word graph for this task is depicted in Figure 5.4. Edge weights in the word graph are calculated using the weighting function defined by Filippova (2010b) (Equation 3.1).

Utilizing Synonymy

Similar to synonym mapping as explained in 3.1.2, in order to reduce the redundancy caused by existing synonyms in the sentences, we use the synsets in WordNet to identify synonym representative candidates. For example, consider n different sentences

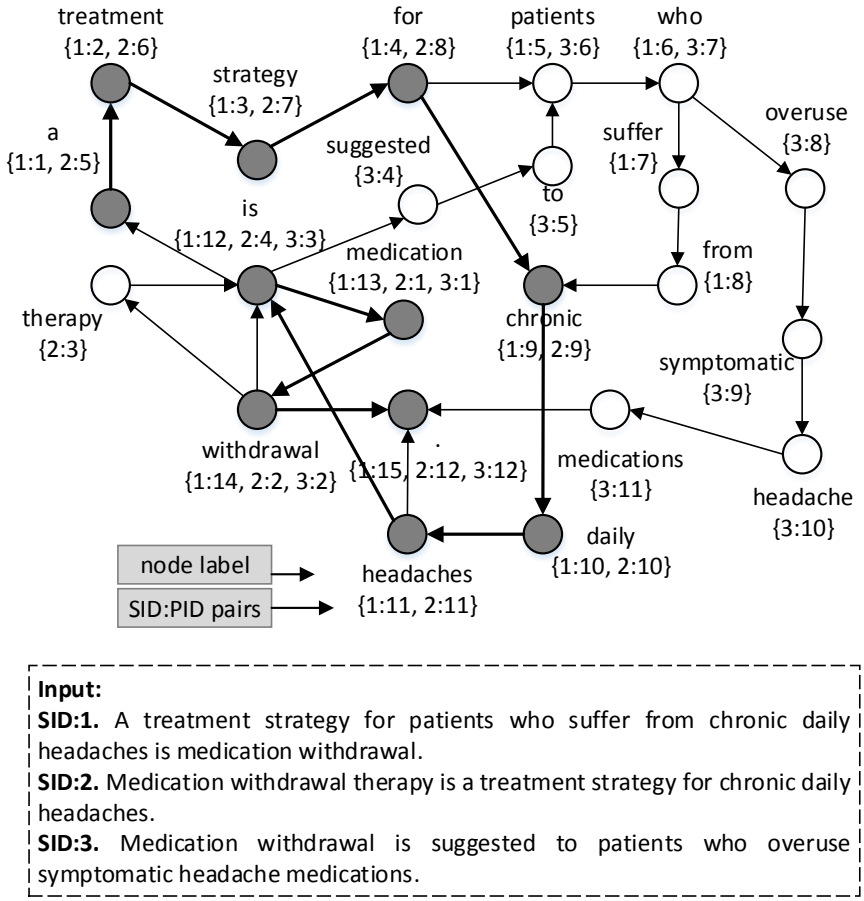


Figure 5.4: Example of the Constructed Word Graph. Thick edges indicate salient paths.

containing words *biliary*, *bilious*, *tumor*, *tumour*, and *neoplasm*. The first two words, and the latter three ones are synonyms of each other. Assume each sentence contains one of these possible combinations (i.e., biliary tumor, biliary neoplasm, biliary tumour, bilious tumor, bilious neoplasm, bilious tumour). Without an appropriate synonym mapping based on the notion of synonymy, these several synonym nodes will be added to the word graph as separate nodes. We consider their frequency to pick one of them as the representative of its synonyms from the other sentences. The weight of the obtained node is computed by summing the frequency scores from

the other nodes as shown in Figure 5.5.

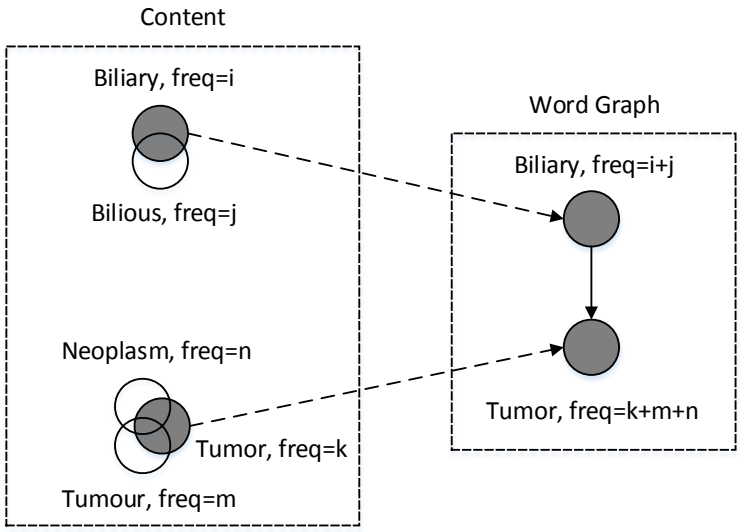


Figure 5.5: Example of Biomedical Synonym Mapping

The heuristic algorithm discussed in (Boudin and Morin, 2013) is then used to find the k-shortest paths ($k = 50$ throughout our experiments) from start to end node in the graph. Hence, most of the potentially good candidates are kept and a decline in performance is prevented. Paths shorter than eight words or do not contain a verb are filtered before re-ranking. The remaining paths are re-ranked and the path that has the lightest average edge weight is eventually considered as the best compression.

Keyphrase Extraction

Furthermore, we use keyphrase extraction based on TextRank algorithm (Mihalcea and Tarau, 2004) to re-rank the compression candidates (explained in Section 4.1.7). The score of a keyphrase is computed by summing the salience of the words it contains, normalized with its length to favor longer n-grams (Equation 3.3). Compression candidates or paths in the word graph are then re-ranked based on the

achieved scores for their keyphrases using Equation 3.4. This re-ranking step favors summary sentences conveying important information.

Ensuring the Syntactic Structure

Since our abstractive word graph generates new summary sentences, we need to ensure the grammatical structure of these newly constructed sentences. Hence, we benefit from the fact that POS tags capture the syntactic roles of words in a sentence. Herein, we train the POS-LM explained in Section 3.1.3 to assign a score to each generated biomedical summary in terms of grammatical structure (Equation 3.7). This grammar-enhanced language model helps in identifying the most grammatical sentence among the k-best compressions.

To build a POS-LM, we have employed the SRILM toolkit (Stolcke, 2002) to build a single global language model. To train the POS-LM, we use Stanford POS tagger to annotate a large part (∼100 M-words) of the BioMed Central full-text corpus for text mining research[10] that contains a large number (∼ 290914) of biomedical articles. Then, we remove all words from the pairs of words/POS in the POS annotated corpus. The candidate sentences also need to be annotated with POS tags. Hence, the score of each summary is estimated by the language model, based on its sequence of POS tags. Since factors like POS tags are less sparse than surface forms, we use a 7-gram POS-LM following Section 3.1.3.

The scaling factor μ was optimized on development data in our experiments and set to 0.4 (Section 5.4). A further syntactic analysis of the generated summaries is also explored in our experiments. Hence, the most grammatical candidate among the candidates contain the most important phrases has been selected as the summary for each cluster.

[10]http://old.biomedcentral.com/about/datamining/

All automatic summaries were generated by selecting sentences until the summary is 30% of the original document size (Plaza et al., 2011). This choice of the summary size is based on the well-accepted heuristic that a summary should be between 15% and 35% of the size of the source text. Considering this convention, we pick a number of three summary sentences (based on their sentence-to-query similarity scores) to answer the corresponding clinical query.

5.4 Experiments

5.4.1 Evaluation Metrics

We have used ROUGE (Lin, 2004) F-measure for unigrams, bigrams, and SU4 (skip-bigrams with maximum gap length 4) over the specialized EBM corpus to evaluate the generated summaries. The bottom-line answers in the EBM corpus have also been used as the model summaries.

5.4.2 Experiment Results

To investigate the effectiveness of our summarization approach for EBM, we compare our approach with *FastSum* (Schilder and Kondadadi, 2008), and a research prototype *LexRank* (Erkan and Radev, 2004). FastSum is a fast query-focused multi-document summarization system based only on word frequency features of topics, documents, and clusters. Each sentence is ranked based on a linear function of scores using a variety of frequency measures. A support vector regression is also used to learn weights of the features. LexRank is a topic-oriented generic summarizer that focuses on multi-document extractive text summarization, and extracts the information in the text that is related to the user specified topic. This prototype has

outperformed both centroid-based methods and other systems participating in DUC in most of the cases (Erkan and Radev, 2004). Comparison with LexRank will allow us to evaluate whether semantic information provides benefits over merely lexical information in graph-based summarization approaches.

In addition, we pick the first and last third sentences of each set of abstracts related to a clinical query, so called (*first part*, and *last part*). We also consider all sentences included in the abstracts related to a clinical query as *whole part*. Afterwards, included sentences in each of these three parts are considered as the input bag of sentences for the following baselines:

- **Head Baseline:** This baseline is used in a variety of summarization applications, specifically in the news summarization area. In our work, this baseline generates summaries by unintentionally selecting three sentences from the *first part*.
- **Random Baseline:** Randomly selects three sentences from the *whole part*.
- **Tail Baseline:** The last sentences in the medical abstracts usually provide conclusions. Hence, this has been used as a baseline for summarization of biomedical texts (Demner-Fushman and Lin, 2007). In our work, this baseline generates summaries by selecting three sentences at random from the *last part*.

Furthermore, the effectiveness of the abstractive summarizer of our approach along with the re-ranking algorithms is also studied using the following experiments. We keep consistency for our approach except to omit the graph-based clustering and the word graph, and converting our abstractive method to the ranking-based extractive approach (Proposed-Ext). For more clarity, we have conducted only two first components of our approach, which are capturing semantic WordNet and UMLS-based sentence-to-query and sentence-to-sentence similarities, and also sentence filtering

step to achieve the most query-relevant sentences. The average performance of the baseline systems and the proposed framework in terms of ROUGE scores are shown in Figure 5.6, and the data is provided in Table 5.5.

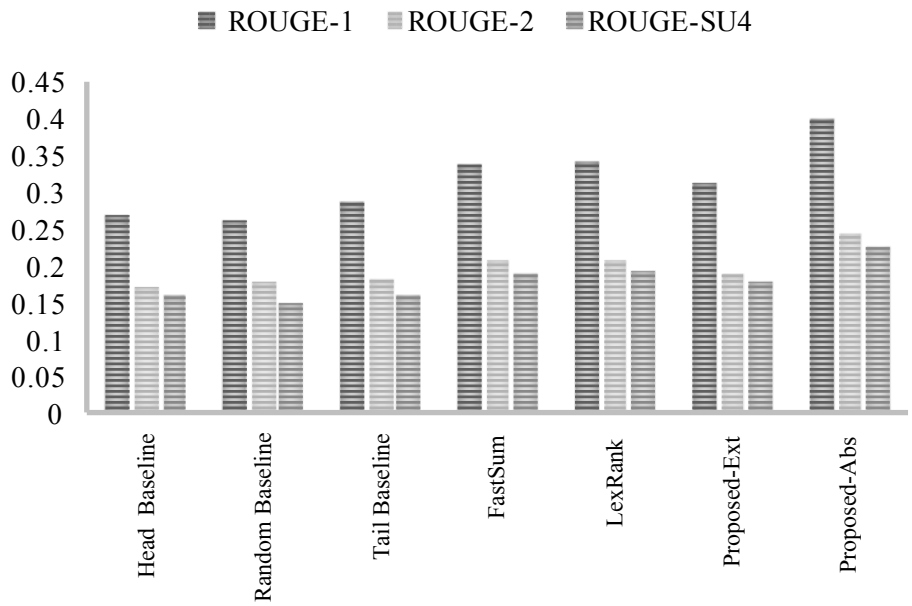


Figure 5.6: Average scores by ROUGE metrics over the EBM corpus

System	**Rouge-1**	**Rouge-2**	**Rouge-su4**
Head Baseline	0.2710	0.1723	0.1593
Random Baseline	0.2623	0.1801	0.1509
Tail Baseline	0.2866	0.1834	0.1607
FastSum	0.3382	0.2081	0.188
LexRank	0.3407	0.2069	0.1938
Proposed-Ext	0.3142	0.1911	0.1806
Proposed-Abs	**0.3985**	**0.2450**	**0.2259**

Table 5.5: Average scores by ROUGE metrics over the EBM corpus

The statistics point out the effectiveness of our abstractive approach over the compared systems on all evaluation metrics. Hence, the overall results support our hypothesis that query-based abstractive summarization using the underlying textual

semantic similarities based on both WordNet and UMLS knowledge sources results in significantly better performance. Besides, the results achieved by Proposed-Ext on the EBM corpus still show some improvements over some of the baseline systems. The main reason may be capturing both WordNet and UMLS-based semantic similarities, which help to select the most query-relevant sentences among the set of biomedical abstracts. Finally, considering the results obtained by *Tail Baseline*, we realize that the last part of each abstract is more likely to be included in the summary. Table 5.6 shows an example of a summary generated by human (gold), our abstractive framework (Proposed-Abs), and the extractive LexRank.

Question: Are major bleeding events from falls more likely in patients on warfarin?
Gold Summary: There is no evidence of increased risk for major bleeding as a result of falls in hospitalized patients taking warfarin. [*PubMed IDs: 7668955, 15638939*]
Proposed-Abs Summary One study found no difference in major bleeding complications between patients taking anticoagulation therapy with not taking. Criteria for taking warfarin were not reported. Prescribing warfarin for patients judged less likely to fall.
LexRank Summary No major hemorrhagic complications were seen following 131 falls in the anticoagulation group (93 patients) and 269 falls in the group not on anticoagulation (175 patients). The study was limited because most falls were from a seated position or partially controlled by an attendant. Major hemorrhage was defined as bruising or cuts requiring immediate attention from a physician.

Table 5.6: Example of different summaries: Gold summary; Proposed-Abs summary; and LexRank summary.

Standard Deviation of Rouge Scores

Since Table 5.5 shows the average results, an important research question that immediately arises is how much the ROUGE scores differ across the abstracts. Hence,

the standard deviation of different ROUGE scores for the summaries generated by Proposed-Abs are shown in Table 5.7.

Metric	**Rouge-1**	**Rouge-2**	**Rouge-su4**
Standard Deviation	0.02104	0.03250	0.03079

Table 5.7: Standard deviation of ROUGE scores for the summaries generated by Proposed-Abs

Exploring Scaling Factors

In our work, two free parameters are defined: *Scaling Factor 1* (η in Equation 5.3 - *measuring semantic similarities using WordNet and UMLS*), and *Scaling Factor 2* (μ in Equation 3.8 - *The final re-ranking score of each generated summary sentence*). We randomly selected 30% of the EBM corpus as a development set to tune these parameters. Figure 5.7 shows the results obtained by ROUGE-1 F-Measure, using different values for η and μ. The best results are obtained when $\eta = 0.6$, and $\mu = 0.4$. Performance deteriorates when the UMLS portion in measuring semantic similarities is less or more than 0.6. On the other hand, when contribution of *TextRank* score for each generated summary sentence is anything but 0.4, the performance gradually decreases. The lowest performance is obtained when *TextRank* score is ignored in re-ranking the generated summary sentences, and also when UMLS semantic signature occupies 0.9 of whole 1.0 value of final semantic similarity measure. This demonstrates the importance of using both WordNet and UMLS to capture the semantic similarities.

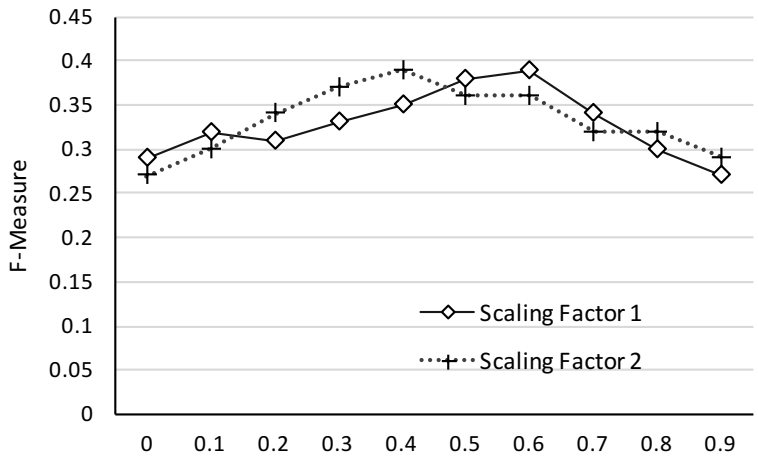


Figure 5.7: Exploring *Scaling Factor 1* (η in Equation 5.3), and *Scaling Factor 2* (μ in Equation 3.8) on the development set.

Syntactic Analysis of the Generated Sentences

Finally, we have analyzed a random selected part of the generated summaries in terms of syntactic structure, using version 5.3.7 of Link Grammar Parser (Figure 5.8). This parser (explained in Section 4.3) is a syntactic analyzer of English language developed at the Carnegie Mellon University (Sleator and Temperley, 1995).

The results show about 85% precision over the syntactic structure of summaries. In details, among 600 random-selected summary sentences, 512 sentences have been shown to have at least one complete linkage with no P.P violations, 64 sentences (11%) have a number of linkages but with some P.P violations, and finally, Link Grammar parser cannot find any linkage for 24 sentences (4%) (Figure 5.9).

```
Found 2 linkages (2 with no P.P. violations)
  Linkage 1, cost vector = (UNUSED=0 DIS=1 AND=0 LEN=21)

    +-------------------------------------------------Xp-------------------------------------------------+
    |                          +---------------------Ss--------------------+                             |
    +-----------Wd-----------+       +------------Jp------------+         |                             |
    |     +--------Dsu-------+       |       +---------A--------+         +---------Ost---------+      |
    |     |      +-----AN----+---Mp--+       |        +----A----+         |        +-----AN-----+      |
    |     |      |           |      |       |        |         |         |        |           |      |
LEFT-WALL a treatment.n strategy.n for.p chronic.a daily.a headaches.n is.v medication.n withdrawal.n .

 Constituent tree:

 (S (NP (NP A treatment strategy)
        (PP for
            (NP chronic daily headaches)))
    (VP is
        (NP medication withdrawal))
    .)


  Linkage 2, cost vector = (UNUSED=0 DIS=1 AND=0 LEN=21)

    +-------------------------------------------------Xp-------------------------------------------------+
    |                          +---------------------Ss--------------------+                             |
    +-----------Wd-----------+       +------------Jp------------+         |                             |
    |     +--------Dsu-------+       |       +---------A--------+         +---------Ost---------+      |
    |     |      +-----AN----+---Mp--+       |        +----AN---+         |        +-----AN-----+      |
    |     |      |           |      |       |        |         |         |        |           |      |
LEFT-WALL a treatment.n strategy.n for.p chronic.a daily.n headaches.n is.v medication.n withdrawal.n .

 Constituent tree:

 (S (NP (NP A treatment strategy)
        (PP for
            (NP chronic daily headaches)))
    (VP is
        (NP medication withdrawal))
    .)
```

Figure 5.8: Example of using Link Grammar Parser for the syntactic analysis of a sample generated sentence *"A treatment strategy for chronic daily headaches is medication withdrawal."*

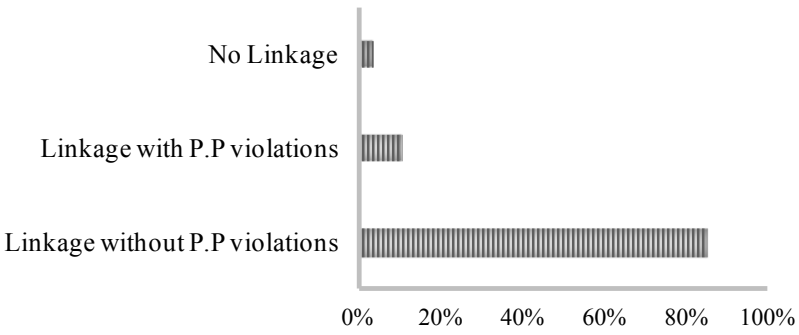


Figure 5.9: Syntactic analysis of a number of 600 generated summary sentences using Link Grammar Parser

5.5 Summary

We have presented the first attempt at integrating WordNet into UMLS for summarizing biomedical texts. Given a clinical query, our approach generates an informative and grammatical summary for a set of biomedical abstracts. We captured sentence-to-sentence and sentence-to-query semantic similarities using both WordNet and UMLS, and studied summarizing capability of the proposed approach in Chapter 4 for biomedical summarization. The experiment results over the evidence-based medicine corpus indicate that our approach outperforms the two competitive systems. Three different baselines for sentence selection have also been used, each aiming to construct a different type of summary according to the type of information in various parts of the source. It has been found that the last part of each abstract is more likely to be included in the summary.

Our approach has significantly satisfied query-biased relevance, biased information novelty and richness. We have tackled the main issue faced by the state-of-the-art biomedical summarizers (i.e., decline in summarization performance due to the poor UMLS coverage of non-medical concepts in the documents to be summarized) (Plaza et al., 2011). This issue is addressed by using WordNet to represent the layman knowledge, and UMLS to represent the professional knowledge. We believe that this approach can bridge the knowledge and language gaps in biomedical summarizers.

Chapter 6

Model-based Semantic Evaluation of Summaries

ROUGE[1] (Lin, 2004) is one of the first and most widely used evaluation metrics for text summarization. However, its assessment merely relies on surface similarities between peer and model summaries. Consequently, ROUGE is unable to fairly evaluate abstractive summaries including lexical variations and paraphrasing. In this chapter, we explore the effectiveness of lexical resource-based models to address this issue. To this end, we adopt a graph-based algorithm into ROUGE to capture the semantic similarities between peer and model summaries. Our semantically motivated approach computes ROUGE scores based on both lexical and semantic similarities. Our proposal is that exploiting the lexico-semantic similarity of the words used in summaries would significantly help ROUGE to correlate better with human judgments. The proposed approach is explained in Section 6.1. Section 6.2 reports the utilized data, the performed meta-evaluation, and the achieved results. Finally, Section 6.3 summarizes this chapter.

[1]Recall-Oriented Understudy for Gisting Evaluation

6.1 Proposed Approach

ROUGE assumes that a peer summary is of high quality if it shares many words or phrases with a model summary. However, different terminology may be used to refer to the same concepts and hence relying only on lexical overlaps may underrate content quality scores. To tackle this issue, our approach utilizes both semantic and lexical similarities between a peer and its corresponding model summary. This method also enables us to reward terms that are not lexically equivalent, but semantically related.

6.1.1 Graph-theoretic Summary Evaluation

Given a pair of peer and model summaries, we compute and compare Personalized PageRank (PPR) vectors at the following levels: (i) *sense level*, to disambiguate each word (having a set of senses); and (ii) *n-gram level*, to measure the semantic similarity. We compare the PPR vectors of each pair of n-grams using the following measures: (i) *Path-based:* considering the path that the random walker takes at each iteration to get to a particular node; (ii) *Rank and Weight:* weighting the overlaps between a pair of ranked PPR vectors. Next, we explain how a PPR vector is constructed for a sense or a set of senses, and how a similarity score is computed accordingly.

Vector Representation

The WordNet graph has edges of various types, with the main types being hypernymy and meronymy to connect nodes containing senses. However, we do not use these types, and consider an edge as an undirected semantic or lexical relation between two synsets. We have utilized the WordNet graph enriched by connecting a

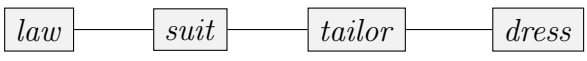
sense - irrespective of its part-of-speech (POS) - with all the other senses that appear in its disambiguated gloss Pilehvar and Navigli (2015). Dimension of the vector representation is the number of connected nodes in the graph. For better clarity, we consider the adjacency matrix A for our semantic graph, and perform iterative random walks beginning at a set of senses S on WordNet with the probability mass of $p^{(0)}(S)$, which is uniformly distributed across the senses $s_i \in S$, and the mass for all $s_i \notin S$ set to zero. This provides a frequency or multinomial distribution over all senses in WordNet, with a higher probability assigned to senses that are frequently visited. The PPR vector of S is given by:

$$p^{(k)}(S) = dAp^{(k-1)}(S) + (1-d)p^{(0)}(S) \tag{6.1}$$

where at each iteration, the random walker may follow one of the edges with probability d or jump back to any node $s_i \in S$ with probability $(1-d)/|S|$. Following the standard convention, the value of damping factor d is set to 0.85, and the number of iterations k is set to 20, which is sufficient for the distribution to converge. The resulting probability vector $p^{(k)}(S)$ is the PPR vector of the lexical item, as it has aggregated its senses' similarities over the entire graph.

Comparing Vectors

Conventional measures for comparing PPR vectors calculate the probability that a random walker meets a particular node after a specific number of iterations, which is potentially problematic (Rothe and Schütze, 2014). For example, consider the following connected nodes:



The PPR vectors of *suit* and *dress* have some weight on *tailor*, which is desirable. However, the PPR vector of *law* will also have a non-zero weight for *tailor*. Consequently, *law* and *dress* are spuriously similar because of the node *tailor*. To prevent this type of false similarity, the random walker needs to take into account the walking path to reach a particular node (Rothe and Schütze, 2014). We formalize this by defining the semantic similarity of two sets of nodes I and J as:

$$Sim_{sem}(I, J) = \sum_{x=0}^{k} c^x \times RW(p^{(x)}(I), p^{(x)}(J)) \tag{6.2}$$

where damping factor c was optimized on the TAC 2010 (Owczarzak and Dang, 2010) AESOP dataset, and set to 0.7 to ensure that early meetings are more valuable than later meetings. At each iteration x, we compare PPR vectors by ranking their dimensions (senses) based on their values, and weighting the overlaps between them (Equation 6.3). Hence, we weight the similarity such that differences in the highest ranks (most important senses in a vector) are penalized more than differences in lower ranks. This measure has proven to be superior to cosine similarity, Jensen-Shannon divergence, and Rank-Biased Overlap for comparing vectors (Pilehvar et al., 2013).

$$RW(Y, Z) = \begin{cases} \frac{\sum_{h \in H} (r_h(Y) + r_h(Z))^{-1}}{\sum_{i=1}^{|H|} (2i)^{-1}}, & \text{if } |H| > 0 \\ 0, & \text{otherwise} \end{cases} \tag{6.3}$$

where H denotes the intersection of all senses with non-zero probability in both vectors Y and Z. $r_h(Y)$ indicates the rank of sense h in vector Y, where rank 1 is the highest rank. The denominator is used as a normalization factor that guarantees a maximum value of one. The minimum value is zero and occurs when there is no overlap between the two vectors, i.e., $|H| = 0$.

Disambiguation of n-grams

Prior to measuring semantic similarities, each word in n-grams has to be analyzed and disambiguated into its intended sense. However, conventional word sense disambiguations are not applicable due to the lack of contextual information. Hence, we seek the semantic alignment that maximizes the similarity of the senses of the compared words. As an example (Pilehvar et al., 2013), consider two sentences of *"a1. Officers fired."* and *"a2. Several policemen terminated in corruption probe."*, the semantic alignment procedure has been performed as "P_{a1}. $officer_n^3, fire_v^4$", *and* "P_{a2}. $policeman_n^1, terminate_v^4, corruption_n^6, probe_n^1$". t_p^i denotes the i-th sense of a word t in WordNet with part-of-speech p. After alignment, among all possible pairings of all senses of $fire_v$ to all senses of all words in *a2*, the sense $fire_v^4$ (employment termination) obtains the maximal similarity value of $Sim_{sem}(fire_v^4, terminate_v^4) = 1$.

Therefore, ROUGE-G transforms the task of determining overlapping n-grams in ROUGE into that of computing the similarity of the best-matching sense pair across the two n-grams. It also enables the same n-grams to have different meanings when paired with different linguistic items. In the following, the generated PPR vectors for a pair of disambiguated n-gram in the model summary, and the peer summary text are compared to calculate their semantic similarities.

Model-summary against Peer Summary

Exploiting underlying semantic similarities between all n-gram pairs in the model and peer summary texts takes a lot of time and effort. To overcome this issue, we consider the peer summary text as a sense-tagged unit, and measure its semantic similarity against each n-gram in the model summary text. For better clarity, let us consider $MT = \{mt_1, mt_2, ..., mt_n\}$, and $PT = \{pt_1, pt_2, ..., pt_m\}$ as the sets of tokens of a model and a peer summary text, respectively. Figure 6.1 shows how

PPR vectors of unigrams and bigrams in a model summary text are compared to the PPR vector of the peer summary text.

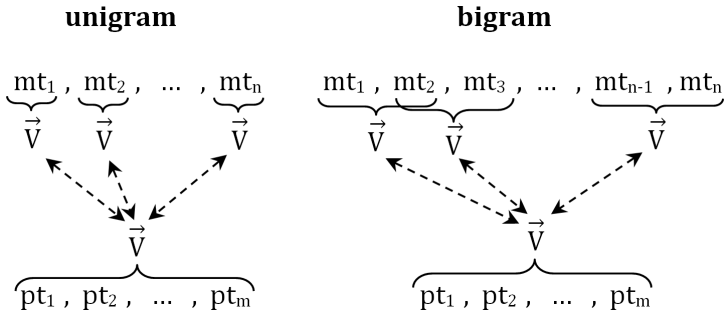


Figure 6.1: Comparing PPR vectors between $n\text{-}gram_m$ (unigrams and bigrams in a model summary text) and a peer summary text

Measuring semantic similarities and sense disambiguation have previously been explained in detail. We can list the steps as follows: (i) Generating PPR vectors for all senses of the n-grams; (ii) Comparing the PPR vectors to disambiguate the n-grams to a set of proper senses; (iii) Generating one PPR vector for each of the n-grams by initializing random walks from their disambiguated senses over WordNet; and finally (iv) Comparing the resulting PPR vectors to compute the semantic similarity between the n-grams. Treating the peer summary text as one unit not only reduces comparison time and increases the efficiency, but also provides a suitable number of content words which guarantees implicit word sense disambiguation, and semantic relationship derivation.

6.1.2 OOV Handling

Similarly to any other graph-based approach that maps words in a given textual item to their corresponding nodes in a semantic network, modeling n-grams through PPR vectors can suffer from the limited coverage of words. This means that only those words that are associated with some nodes in WordNet can be handled. Since Out-

of-vocabulary (OOV) words are the words that are not defined in the corresponding lexical resource, they will be ignored while generating PPR vectors. The reason is that they do not have an associated node in the WordNet graph for the random walk to be initialized from. Denying OOV words, such as infrequent named entities, acronyms or jargon, while increasing in a text, can be problematic when measuring semantic similarity of n-gram pairs. To take OOV words into consideration, we introduce new dimensions in the resulting PPR vector, one for each OOV term. Following Pilehvar and Navigli (2015), we finally set the associated weights of the new dimensions to 0.5 so as to guarantee their placement among the top dimensions in their vectors.

6.1.3 Multiple Levels of Evaluation

Most single automatic metrics use one level of evaluation (i.e., lexical, syntactic or semantic). A better approach is to assess the results while combining multiple levels of evaluation into one model (Ellouze et al., 2013). For better clarity, consider the following groups of sentences:

a1. Soldiers were killed.

a2. Soldiers were executed.

a3. Military personnel were executed for committed crimes.

b1. Soldiers were killed.

b2. Soldiers were murdered.

b3. Several servicemen were murdered by criminals.

Surface-based approaches that are merely based on string similarity cannot capture

the similarity between any of the above pairs of *a1* and *a3*, or *b1* and *b3* as there exists no lexical overlap. In addition, a surface-based semantic similarity approach considers both *a1* and *b1* as being identical sentences, whereas we know that different meanings of the verb "kill" are triggered in the two contexts. Although verbs "kill", "execute" and "murder" are close together in WordNet, *a2* and *b2* carry very different connotations. As a remedy, we need to transform words to senses and perform disambiguation by taking into account the context of the paired linguistic item, hence providing a deeper measure of similarity comparison. We combine lexical and semantic similarities to compute ROUGE-G-N (Equation 6.4). This approach can increase the chance of getting the evaluation results more correlated with human assessments.

$$\text{ROUGE-G-N} = \frac{\sum_{M \in \{ModelSums\}} \sum_{n\text{-}gram_m \in M, n\text{-}gram_p \in PeerSum} Sim_{LS}(n\text{-}gram_m, n\text{-}gram_p)}{\sum_{M \in \{ModelSums\}} \sum_{n\text{-}gram_m \in M} Count(n\text{-}gram_m)} \tag{6.4}$$

where n stands for the length of n-gram, and Sim_{LS} is the score of lexico-semantic similarity between a pair of n-grams in model summary ($n\text{-}gram_m$) and peer summary ($n\text{-}gram_p$).

To compute Sim_{LS}, we have conducted a set of experiments using lexical similarities, $Count_{match}(n\text{-}gram_m, n\text{-}gram_p)$, and/or semantic similarities, Sim_{sem} (Equation 6.2). The best correlation is obtained while using a linear combination of both

scores with different weights according to the following equation:

$$\begin{aligned} &Sim_{LS}(n\text{-}gram_m, n\text{-}gram_p) = \\ &\quad \beta \times Count_{match}(n\text{-}gram_m, n\text{-}gram_p) \\ &\quad + (1-\beta) \times Sim_{sem}(n\text{-}gram_m, n\text{-}gram_p) \end{aligned} \tag{6.5}$$

The scaling factor β was optimized on the TAC 2010 AESOP dataset (Owczarzak and Dang, 2010), and set to 0.5 to reach the best correlation with the manual metrics of Pyramid and Responsiveness. $Count_{match}(n\text{-}gram_m, n\text{-}gram_p)$ is the maximum number of the *n-gram* co-occurring in a peer summary and a set of model summaries.

6.2 Experiments

6.2.1 Data and Meta-evaluation

The only available datasets for the task of Summarization Evaluation are three AESOP datasets[2] provided by TAC 2009, 2010, and 2011. Among them, we optimize scaling factors using the TAC 2010 AESOP dataset, and evaluate ROUGE-G on the TAC 2011 (Owczarzak and Dang, 2011) AESOP dataset for two main reasons: (i) it is the only dataset on which evaluation metrics can be assessed for their ability to measure summary Readability; (ii) To be in line with the most recent work (ROUGE-WE) that has also been evaluated only on this dataset for measuring the Readability scores. This dataset consists of 44 topics, and two sets of 10 documents for each topic: set A for initial summaries; set B for update summaries. There are four human-crafted model summaries for each document set. A summary for each topic is generated by each of the 51 summarizers which participated in the main

[2]https://tac.nist.gov/data/

TAC summarization task. Source documents for summarization are taken from the New York Times, the Associated Press, and the Xinhua News Agency newswire.

Two different types of evaluation were tasked in TAC 2011 AESOP: *All Peers* and *No Models*. The former case assigns a score to each peer summary, including the model summaries. This evaluation is intended to focus on whether an automatic metric can distinguish between human and automatic summarizers. The latter assigns a score to each peer summary, excluding the model summaries. This case is intended to focus on how well an automatic metric is able to assess automatic summaries. Using model summaries as references, each automatic summary can be evaluated against all four references simultaneously. Since our aim is to evaluate the quality of automatic summaries, we make use of *No Models* evaluation.

The output of participating automatic metrics is tasked to be compared against human judges using three manual metrics of *Pyramid*, *Readability*, and *Responsiveness*. Hence, the outputs are scored based on their summary content, linguistic quality, and a combination of both, respectively. Prior to computing correlation of ROUGE-G variants with manual metrics, ROUGE-G scores have reliably been computed (95% confidence intervals) under ROUGE bootstrap resampling with the default number of sampling point (1000). Correlation of ROUGE-G evaluation scores with the human judgments is then assessed with three metrics of correlation: Pearson r; Spearman ρ; and Kendall τ.

The value of all measures is between -1 and 1 of which 1 or -1 indicates a strong relationship between the two measures. The closer the value is to zero, the weaker the relation between the two measures. 25 automatic metrics participated in the TAC 2011 AESOP task, three of which (i.e., ROUGE-2, ROUGE-SU4, and BE-HM) were used as baselines. In our experiments, the effectiveness of ROUGE-G is demonstrated by assessing its three variants (ROUGE-G-1, 2, and SU4) against their corresponding variants of ROUGE, and the other 23 AESOP participants. Note that ROUGE-1

was not among the participating metrics, but will be considered in our experiments. We compute scores using the default NIST settings for baselines in the TAC 2011 AESOP task (with stemming and keeping stopwords[3]).

6.2.2 Experiment Results

We evaluate three variants of ROUGE-G (i.e., ROUGE-G-1, 2, and SU4), against the top 13 best-performing metrics among the 23 metrics participated in AESOP 2011, the baselines (i.e., ROUGE-2, SU4, BE-HM), ROUGE-1, and the most recent related work (ROUGE-WE). Correlation results of the best-performing AESOP metrics with Pyramid, Responsiveness, and Readability scores to the correlation metrics of Pearson r, Spearman ρ, and Kendall τ are depicted in Figures 6.2, 6.3, and 6.4, respectively. The highest correlation results are highlighted for better clarity. To get a more complete picture of the usefulness of our proposal, it will be instructive to also compare it against the top metrics (C_S_IIITH3, DemokritosGR1, Catolicasc1) among the 23 metrics participated in TAC AESOP 2011, ROUGE, and ROUGE-WE with Pyramid, Responsiveness, and Readability.

To analyze the impact of preventing the above-mentioned false similarities, we also conduct an ablation study using the achieved semantic similarities of two sets of nodes without considering the walking path in computing the PPR vectors (ROUGE-G-NoPath). For this purpose, we exclude the walking path from Equation 6.2, and considering $Sim_{sem}(I, J) = RW(p^{(x)}(I), p^{(x)}(J)$. The results are provided in Tables 6.1, 6.2, and 6.3, respectively. Overall results support our proposal to consider semantics besides surface with ROUGE.

According to Table 6.1, ROUGE-G-2 achieves the best correlation with Pyramid,

[3]https://tac.nist.gov/2011/Summarization/AESOP.2011.guidelines.html/

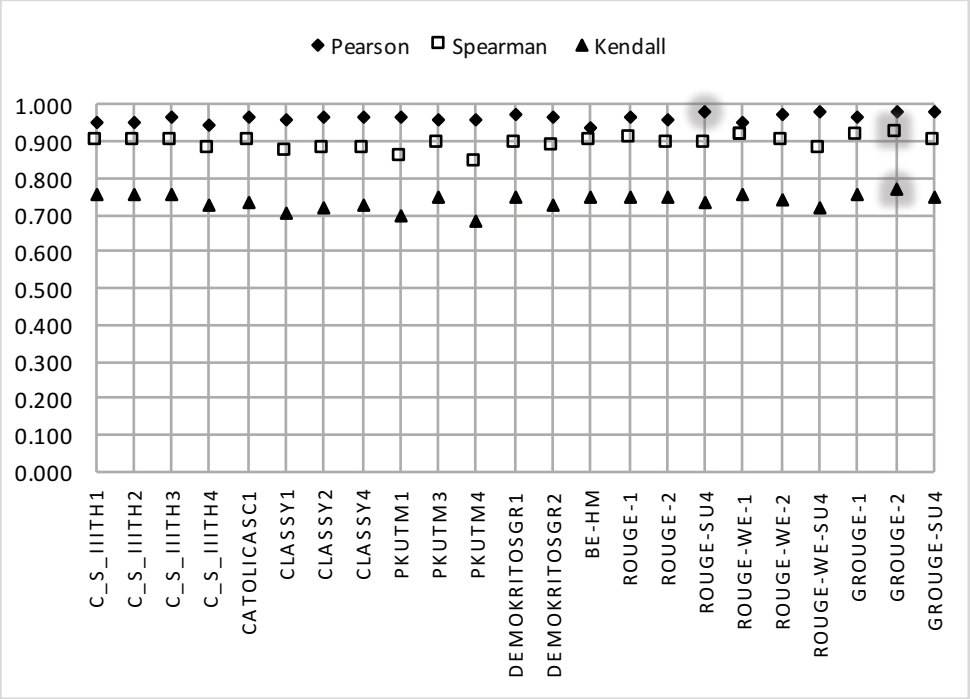


Figure 6.2: Correlation of the best-performing AESOP metrics with the manual metric of Pyramid using the correlation metrics of Pearson r, Spearman ρ, and Kendall τ on the TAC 2011 AESOP dataset

regarding all correlation metrics. Moreover, every ROUGE-G variant outperforms its corresponding ROUGE and ROUGE-WE variants, regardless of the correlation metric used. However, the only exception is ROUGE-SU4, which correlates slightly better with Pyramid when measuring with Pearson correlation. One possible reason is that Pyramid measures content similarity between peer and model summaries, while the variants of ROUGE-G favor semantics behind the content for measuring similarities. Since some of the semantics attached to the skipped words are lost in the construction of skip-bigrams, ROUGE-SU4 shows a better correlation comparing to ROUGE-G-SU4.

For Responsiveness, ROUGE-G-SU4 achieves the best correlation when measuring with Pearson (Table 6.2). We also observe that ROUGE-G-2 obtains the best correlation with Responsiveness while measuring with the Spearman and Kendall rank

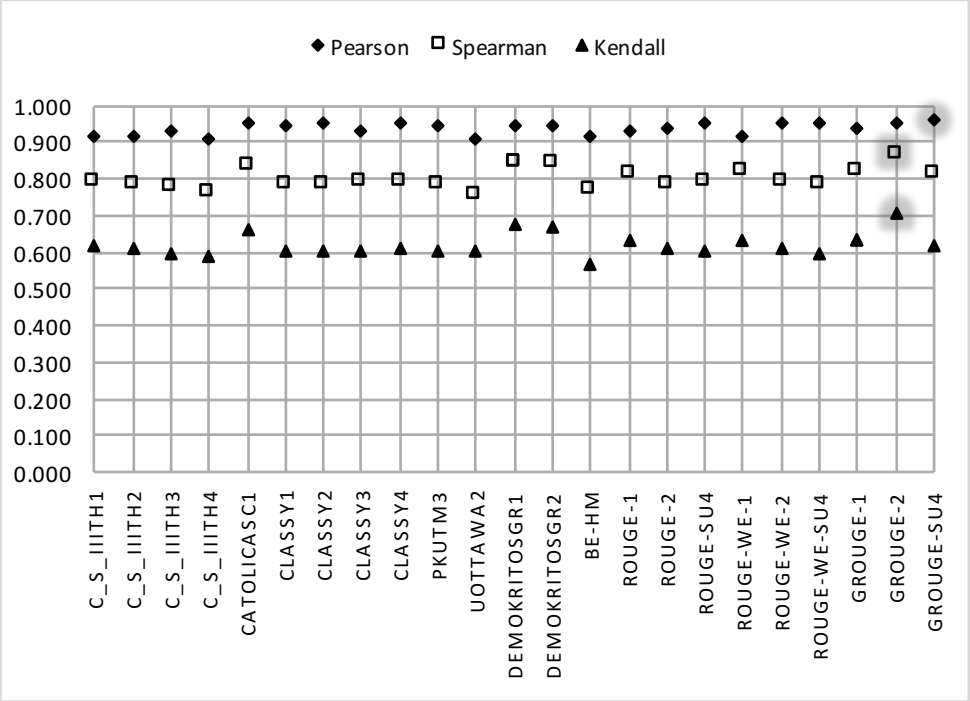


Figure 6.3: Correlation of the best-performing AESOP metrics with the manual metric of Responsiveness using the correlation metrics of Pearson r, Spearman ρ, and Kendall τ on the TAC 2011 AESOP dataset

correlations. The reason is that semantic interpretation of bigrams is easier, and that of contiguous bigrams is much more precise. We also see that every variant of ROUGE-G outperforms its corresponding ROUGE and ROUGE-WE variants. The key difference between the Pearson correlation and Spearman/Kendall rank correlation, is that the former assumes that the variables being tested are normally distributed, and linearly related to each other. The latter two measures are however non-parametric and make no assumptions about the distribution of the variables being tested. The assumption made by the Pearson correlation has been known too constraining (Ng and Abrecht, 2015), given that any two independent evaluation systems may not exhibit linearity.

The readability score reflects the fluency and structure of the summary, independently of content; and is based on grammaticality, structure, focus, coherence and

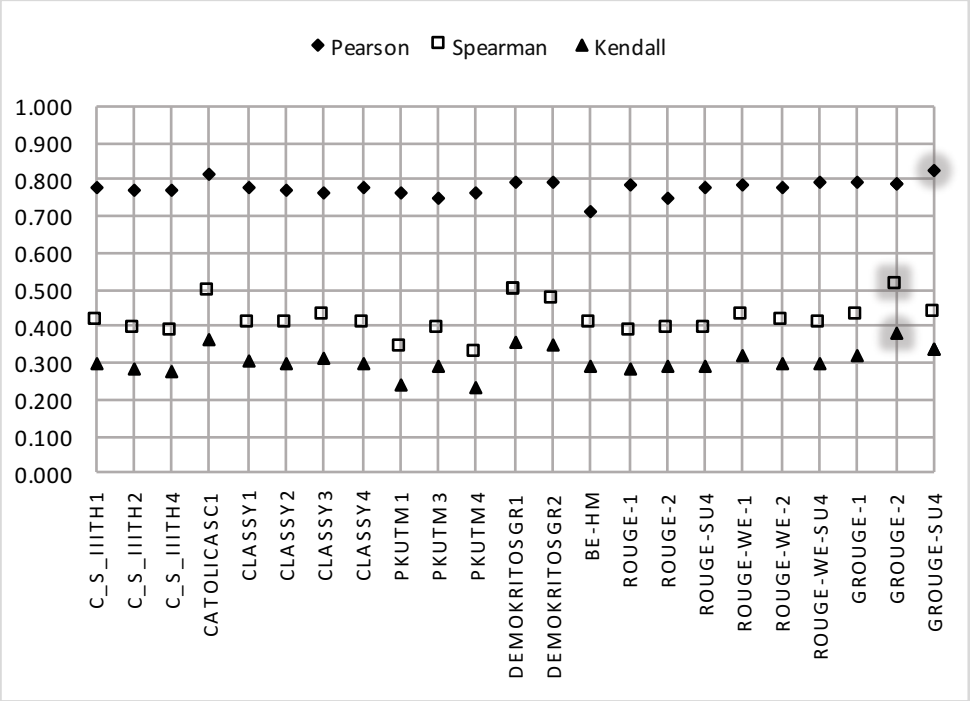


Figure 6.4: Correlation of the best-performing AESOP metrics with the manual metric of Readability using the correlation metrics of Pearson r, Spearman ρ, and Kendall τ on the TAC 2011 AESOP dataset

etc.. Although our main goal is not to improve the readability, ROUGE-G-SU4 and ROUGE-G-2 are observed to correlate very well with this metric when measured with the Pearson and Spearman/Kendall rank correlations, respectively (Table 6.3). Besides, every variant of ROUGE-G represents the best correlation results comparing to its corresponding variants of ROUGE and ROUGE-WE for all correlation metrics. This is likely due to considering word types and part-of-speech tagging while aligning and disambiguating n-grams. Part-of-speech features are shown by Feng et al. (2010) to be helpful in predicting linguistic quality.

Overall, considering Pyramid, Responsiveness, and Readability, and regardless of the correlation metric used, every ROUGE-G variant outperforms its corresponding ROUGE variant, with only one exception: ROUGE-SU4 correlates slightly better with Pyramid when measuring with Pearson correlation, to which possible reasons are

Metric	**Pearson**	**Spearman**	**Kendall**
C_S_IIITH3	0.965	<u>0.903</u>	<u>0.758</u>
DemokritosGR1	0.974	0.897	0.747
Catolicasc1	0.967	0.902	0.735
ROUGE-1	0.966	0.909	0.747
ROUGE-2	0.961	0.894	0.745
ROUGE-SU4	<u>0.981</u>	0.894	0.737
ROUGE-WE-1	0.949	0.914	0.753
ROUGE-WE-2	0.977	0.898	0.744
ROUGE-WE-SU4	0.978	0.881	0.720
ROUGE-G-1(NoPath)	0.968	0.916	0.758
ROUGE-G-2(NoPath)	0.979	0.921	0.768
ROUGE-G-SU4(NoPath)	0.980	0.901	0.747
ROUGE-G-1	0.971	0.915	0.758
ROUGE-G-2	**0.983**	**0.926**	**0.774**
ROUGE-G-SU4	0.979	0.898	0.741

Table 6.1: Correlation results ($p < 0.05$) with the manual metric of Pyramid using the correlation metrics of Pearson r, Spearman ρ, and Kendall τ. The best correlations are specified in bold, and the underlined scores show the top correlations in the TAC AESOP 2011.

discussed earlier. Looking at ROUGE-G-2 that is far superior than its corresponding variants while measuring with Spearman and Kendall rank correlations, supports our proposal to consider semantics besides surface with ROUGE. Furthermore, the superiority of every ROUGE-G variant to its corresponding variant without considering the path in WordNet, indicates the importance of taking into account the walking path taken by the random walker to reach a particular node in WordNet.

6.2.3 Significance Test

Evaluation of summarization metrics that depart from correlation with human judgment must include the ability of a metric/significance test combination to identify

Metric	**Pearson**	**Spearman**	**Kendall**
C_S_IIITH3	0.933	0.781	0.596
DemokritosGR1	0.947	<u>0.845</u>	<u>0.675</u>
Catolicasc1	0.950	0.837	0.666
ROUGE-1	0.935	0.818	0.633
ROUGE-2	0.942	0.790	0.610
ROUGE-SU4	<u>0.955</u>	0.790	0.602
ROUGE-WE-1	0.916	0.819	0.631
ROUGE-WE-2	0.953	0.797	0.615
ROUGE-WE-SU4	0.954	0.787	0.597
ROUGE-G-1(NoPath)	0.940	0.822	0.635
ROUGE-G-2(NoPath)	0.954	0.863	0.705
ROUGE-G-SU4(NoPath)	**0.958**	0.812	0.617
ROUGE-G-1	0.944	0.825	0.638
ROUGE-G-2	0.956	**0.869**	**0.713**
ROUGE-G-SU4	0.957	0.814	0.616

Table 6.2: Correlation results ($p < 0.05$) with the manual metric of Responsiveness using the correlation metrics of Pearson r, Spearman ρ, and Kendall τ. The best correlations are specified in bold, and the underlined scores show the top correlations in the TAC AESOP 2011.

a significant difference between the quality of human and system-generated summaries (Rankel et al., 2011). Since the large/small differences in competing correlations with human assessment are not an acceptable proof of superiority/inferiority in performance of one metric over another, prior to any conclusion in this regard, significance tests should be applied. Hence, to better clarify the effectiveness of ROUGE-G, we use pairwise Williams significance test[4] recommended by (Graham et al., 2015) for summarization evaluation.

Accordingly, evaluation of a given summarization metric, M_{new}, takes the form of quantifying three correlations: $r(M_{new}, H)$, that exists between the evaluation metric scores for summarization systems and corresponding human assessment scores;

[4]Also known as Hotelling-Williams

Metric	**Pearson**	**Spearman**	**Kendall**
C_S_IIITH3	0.731	0.358	0.242
DemokritosGR1	0.794	<u>0.497</u>	0.359
Catolicasc1	<u>0.819</u>	0.494	<u>0.366</u>
ROUGE-1	0.790	0.391	0.285
ROUGE-2	0.752	0.398	0.293
ROUGE-SU4	0.784	0.395	0.293
ROUGE-WE-1	0.785	0.431	0.322
ROUGE-WE-2	0.782	0.414	0.304
ROUGE-WE-SU4	0.793	0.407	0.302
ROUGE-G-1(NoPath)	0.793	0.433	0.326
ROUGE-G-2(NoPath)	0.787	0.513	0.384
ROUGE-G-SU4(NoPath)	**0.824**	0.440	0.334
ROUGE-G-1	0.791	0.434	0.330
ROUGE-G-2	0.790	**0.516**	**0.385**
ROUGE-G-SU4	0.823	0.445	0.334

Table 6.3: Correlation results ($p < 0.05$) with the manual metric of Readability using the correlation metrics of Pearson r, Spearman ρ, and Kendall τ. The best correlations are specified in bold, and the underlined scores show the top correlations in the TAC AESOP 2011.

$r(M_{base}, H)$, that stands for the correlation of baseline metrics with human judges; and the third correlation, between evaluation metric scores themselves, $r(M_{base}, M_{new})$. It can happen for a pair of competing metrics for which the correlation between metric scores is strong, that a small difference in competing correlations with human assessment is significant, while, for a different pair of metrics with a larger difference in correlation, the difference is not significant (Graham et al., 2015). Utilizing this significance test, the results show that all increases in correlations of ROUGE-G compared to ROUGE and ROUGE-WE variants in Tables 6.1, 6.2 and 6.3 are statistically significant ($p < 0.05$).

6.2.4 Exploring Scaling Factor

In this section, we have optimized scaling factor β in Equation 6.5, and obtained a balance between contributions of lexical and semantic similarity scores to calculate the lexico-semantic similarity. To this end, we make use of the TAC 2010 AESOP dataset. Figure 6.5 shows the correlation results by the variants of ROUGE-G with Pyramid (Pyr) and Responsiveness (Rsp) metrics measured by Pearson. The best results are observed when $\beta = 0.5$. Performance deteriorates when the value of β approaches 1.0 which indicates the ROUGE scores without any touch of semantic similarity. Decreasing the weight of β to zero causes the exclusion of lexical match counts, and consequently inappropriateness of the outcomes. This demonstrates the importance of using both lexical and semantic similarities to fairly judge the quality of summaries.

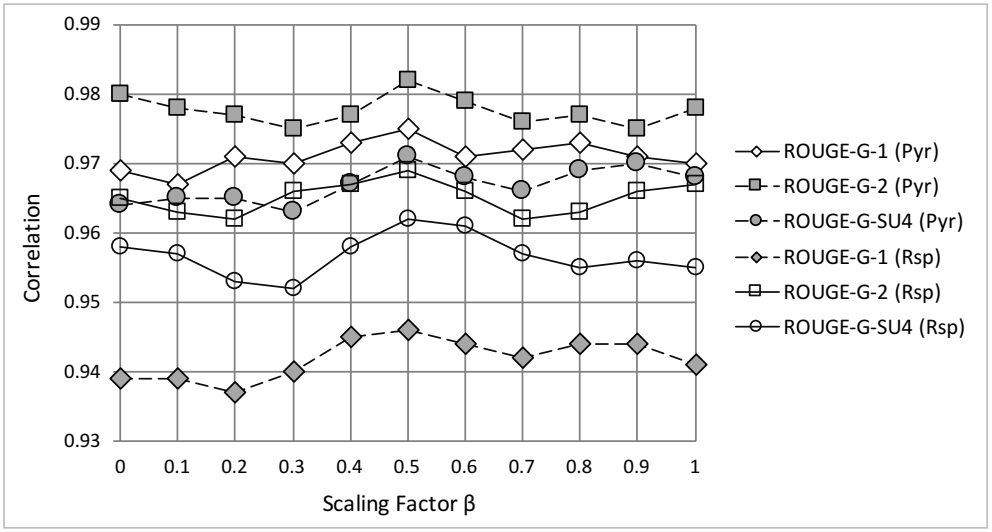


Figure 6.5: Exploring scaling factor β on the TAC 2010 AESOP dataset

6.3 Summary

We have proposed an effective approach (namely ROUGE-G) to overcome the limitation of high lexical dependency in ROUGE. Our approach leverages a sense-based representation to calculate PPR vectors for n-grams. The semantic similarity of n-grams are then computed using a formalization of *Path-based* and *Rank and Weight* measures. We finally improve on ROUGE by performing both semantic and lexical analysis of summaries. Evaluation is processed by comparing each n-gram in the model summary against the corresponding peer summary text. To this end, the PPR algorithm is employed, and all senses have been disambiguated before comparison. We have evaluated our approach with the following settings for computing and comparing PPR vectors: (i) *Path-based* with *Rank and Weight* measure (current setting); (ii) *Path-based* with *cosine similarity*; (iii) Excluding *path-based* measure and using *Rank and Weight* measure solely. The results show that the current setting performs better than the other two. Overall experiment results over the TAC AESOP datasets demonstrate that ROUGE-G achieves higher correlations with manual judgments in comparison with the well-established ROUGE.

Since this approach goes beyond the lexical surface and exploits the underlying semantics, we believe that it would work even better on more comprehensive texts such as a dataset provided for the evaluation of abstractive summaries. Therefore, our ongoing work includes constructing a standard dataset for assessing the automatic metrics specified to evaluate abstractive summaries. We also believe that this approach can open a door to the evaluation of automatic text simplification. The reason is that text simplification indicates the process of simplifying a text without losing its meaning, and this approach can capture the underlying meaning in a text, regardless of its surface. Hence, in future, we intend to adopt this approach with the aim of helping ROUGE to gain qualitative insights into the nature of text simplification.

Chapter 7

Model-free Summary Content Evaluation

Much previous research on summarization evaluation has focused on the model-based evaluation of summaries. Such studies typically target multiple human generated summaries to assess the quality of peer summaries. Their evaluation paradigm falls short on non-standard test sets where model summaries do not exist. This chapter describes a novel and significant attempt to evaluate summaries in the absence of human model summaries. Our proposed approach firstly focuses on exploiting the compositional capabilities of corpus-based and lexical resource-based word embeddings to develop multiple features. These features are then used to train a learning model for predicting the summary content quality.

We introduce each feature in Section 7.2, and discuss the experiments and results in Section 7.3. The findings of our error analysis are also explained in Section 7.3.1. Finally, we evaluate the trained model on the test set in Section 7.3.2.

7.1 Data and Evaluation Metrics

We carry out our experiments on the query-focused and update summarization tasks from TAC 2009 with 44 inputs as our test set, and from TAC 2008 with 48 inputs as the development set. These datasets consist of two sets of 10 news documents for each input: (i) set A for initial summaries; (ii) set B for update summaries. Both A and B are on the same general topic but B contains documents published later than those in A. The update summary of set B is created assuming that the user is aware of what exists in set A. There are also four human-crafted model summaries for each input in each document set. A maximum of 100 words summary that addresses the information required by the given query statement (consisting of a title and narrative) has been produced by each of the 53 and 58 automatic summarizers participated in TAC 2009 and 2008, respectively. An example query statement is shown here:

Title: Barack Obama

Narrative: Track the increase in Barack Obama's popularity, visibility, support, and activities.

Content and linguistic quality are two conventional factors in evaluation of summary quality. Herein, we focus on the problem of automatic evaluation of content quality. Hence, we assess the performance of our metrics in replicating manual correlations of pyramid and responsiveness. It is noteworthy that responsiveness incorporates at least some aspects of linguistic quality.

Pyramid

This evaluation method (Passonneau et al., 2005) is a content assessment measure which compares content units in a system summary to weighted content units in a set of model summaries. It uses multiple human models from which annotators identify semantically defined Summary Content Units (SCU). Each SCU is assigned a weight equal to the number of human model summaries that express that SCU. An ideal maximally informative summary would express a subset of the most highly weighted SCUs, with multiple maximally informative summaries being possible. The pyramid score for a system summary is equal to the ratio between the sum of weights of SCUs expressed in a summary (again identified manually) and the sum of weights of an ideal summary with the same number of SCUs. Four human summaries provided by NIST for each input and task were used for the pyramid evaluation at TAC.

Responsiveness

This is a measure of overall quality combining both content and linguistic quality. Summaries must present useful content in a structured fashion in order to better satisfy the user's need. Assessors directly assigned scores on a scale of 1 (poor) to 5 (very good) to each summary. These assessments are done without reference to any model summaries.

Linguistic Quality

This measure ranks summaries in a 5-point scale indicating how well a summary satisfied the factors of linguistic quality (i.e., grammaticality, non-redundancy, referential clarity, focus, structure and coherence). In our work, we do not evaluate linguistic quality.

7.2 Proposed Approach

We propose five classes of features to assess the quality of summary content in the absence of model summaries: (i) *Distributional Semantic Similarity*; (ii) *Topical Relevance*; (iii) *Query Relevance*; (iv) *Coherence*; and (v) *Novelty*. Before computing the features, all words in input documents, summaries, and queries are converted to lower case and stop-word filtered. We experiment with two variants of word embeddings as the basic building block to design our features:

Corpus-based Word Embeddings

We utilize the 300-dimensional embeddings for 3M words and phrases trained on Google News[1], a corpus of $\sim 10^{11}$ tokens, using word2vec CBOW (Mikolov et al., 2013a). Word2vec learns a vector representation for each word using a neural network language model. It also allows to learn complex semantic relationships using simple vectorial operators, such as $vec(king) - vec(man) + vec(woman) \approx vec(queen)$. Stemming is not performed to make the word embeddings discover the linguistic regularities of words with the same root.

Lexical Resource-based Word Embeddings

We use WordNet (Fellbaum, 1998) to measure the lexico-semantic similarity between the input and its summary. Since the constraints of WordNet lexical resource can be formalized as constraints on embeddings, we can use embeddings of non-word data types (i.e., senses). Specifically, we compute the embedding of a word by averaging the embeddings of its senses in WordNet. For example, the vector of the word *suit*

[1]https://code.google.com/p/word2vec/

is modeled as the average of a vector representing *lawsuit* and a vector representing *business suit*.

We obtain the sense embeddings using the pre-trained model by Rothe and Schütze (2015), that lives in the same vector space as the pre-trained word2vec by Mikolov et al. (2013a). Their model is an autoencoder neural-network that takes word embeddings and learns sense embeddings based on the following intuitions: (i) a word's embedding is the sum of the embeddings of its senses; and (ii) the senses related by WordNet relations (e.g., hypernymy, antonymy, similarity) have similar embeddings. Considering WordNet relations also helps to compute embeddings for senses in WordNet which are not in the word2vec vocabulary.

We further assume that the probability of a word sense is in proportion to its frequency in WordNet. Hence, the probability that a sense $\mathcal{S}_{ij}$ is the meaning of the word w_i, is the ratio of the frequency of that sense $freq(\mathcal{S}_{ij})$ to the total frequency of the word. If the frequency of a word sense is 0 in WordNet, we set it to 1. Finally, the embedding of word w_i is computed[2] as a weighted average of its senses $\mathcal{S}_{ij}, 1 \leq j \leq n$, where the weights represent the probability of senses:

$$\vec{w_i} = \frac{\sum_{\mathcal{S}_{ij} \in \mathcal{S}yn(w_i)} freq(\mathcal{S}_{ij}) \times \vec{\mathcal{S}}_{ij}}{n \sum_{\mathcal{S}_{ij} \in \mathcal{S}yn(w_i)} freq(\mathcal{S}_{ij})} \tag{7.1}$$

7.2.1 Distributional Semantic Similarity

A good summary must satisfy both *coverage* and *diversity* properties. For clarity, summary sentences should cover a sufficient non-redundant amount of information from the original input text. Diversity property is also fundamental especially for multi-document summarization. Moreover, one would expect good summaries to

[2]Words were stemmed before inferring their embeddings.

be characterized by low distance between probability distributions of words in the input and summary, and by high similarity with the input. Hence, we design this feature based on the geometric meaning of the centroid vector of a document using the compositional properties of the word embeddings (Mikolov et al., 2013a). The main idea is to give a distributed representation of words/senses in the input and its summary, and compare their centroid vectors to realize how much the summary content works as a pseudo-input and condenses the meaningful information of the input.

The centroid embedding $\vec{T}$ of a text $T = \{t_1, t_2, ..., t_n\}$, is the sum of the embeddings of tokens of T divided by the number of tokens n. Based on the problem, we can also assign a weight $\mathcal{W}$ to each token in T (Figure 7.1). Accordingly, the centroid embedding for each summary sentence $\vec{s_j}$ is computed by averaging the embeddings of all words comprising the sentence (Radev et al., 2004). Similarly, we construct a centroid vector for each document, $\vec{d_i}$, in the input document set. To better assess the *informativeness* of the summary content, we assign higher weights to specialized words in a document by considering the Inverse Document Frequency (IDF) scores of words:

$$\vec{d_i} = \frac{\sum_{w_j \in d_i} \vec{w_j} \times TF(w_j, d_i) \times IDF(w_j)}{n \sum_{w_j \in d_i} TF(w_j, d_i) \times IDF(w_j)} \tag{7.2}$$

where n is the number of words in document d_i, and $\vec{w_j}$ is the embedding of word w_j. $TF(w_j, d_i)$ stands for the term frequency of w_j in d_i. The IDF scores are computed on the whole document set.

Finally, we compare summary sentences and the input documents using the Word Mover's Distance (WMD) algorithm (Kusner et al., 2015). WMD measures the total distance the centroid embeddings of summary sentences and the input documents have to travel to become identical. Accordingly, we measure the dissimilarity degree

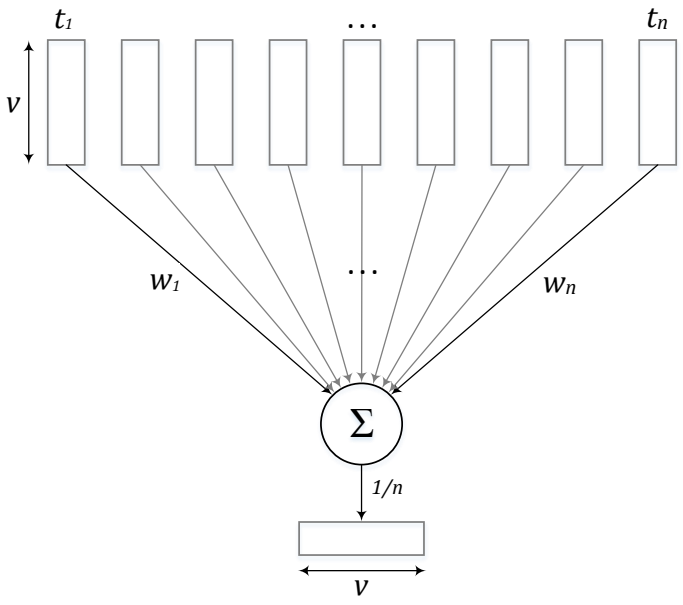


Figure 7.1: The weighted centroid embedding of text $T = \{t_1, t_2, ..., t_n\}$

between two sets of embedding vectors, $D = \{\vec{d}_1, ..., \vec{d}_n\}$ and $S = \{\vec{s}_1, ..., \vec{s}_m\}$, by calculating the minimum amount of summing up individual distances (travel costs) that centroid embeddings of the documents in D need to travel to reach the embeddings of sentences in S:

$$WMD(D, S) = \min_{F \geq 0} \sum_{\vec{d}_i \in D} \sum_{\vec{s}_j \in S} F_{\vec{d}_i \vec{s}_j} \times dist(\vec{d}_i, \vec{s}_j) \tag{7.3}$$

subject to,

$$\sum_{\vec{d}_i \in D} F_{\vec{d}_i \vec{s}_j} = \frac{1}{|S|}, \forall \vec{s}_j \in S, \sum_{\vec{s}_j \in S} F_{\vec{d}_i \vec{s}_j} = \frac{1}{|D|}, \forall \vec{d}_i \in D$$

where $F \in \mathbb{R}^{V \times V}$ with V as the vocabulary size, is a flow matrix which indicates how much probability mass should flow (or travel) from document centroid embedding $\vec{d}_i$ in set D to sentence embedding $\vec{s}_j$ in set S, and vice versa. $dist(\vec{d}_i, \vec{s}_j)$ denotes the individual distance (or travel cost) between $\vec{d}_i$ and $\vec{s}_j$: $dist(\vec{d}_i, \vec{s}_j) = \|\vec{d}_i - \vec{s}_j\|_2$.

7.2.2 Topical Relevance

Topic features serve as a basis for evaluating topical relevance of a summary to the input documents. Herein, we aim to find the distribution of the most probable topics embodied in the input document set, and their relevance to the summary sentences. To this end, we use Latent Dirichlet Allocation (LDA) algorithm (Blei et al., 2003; Arora and Ravindran, 2008) to determine the topics that characterize every document set. LDA is a generative model for documents to determine topic compositions of words and document mixtures of topics (represented by a probability distribution over topics), by assigning words to topics within documents. Hence, in the context of text modeling, the topic distribution provides an underlying semantic representation of the documents and can be useful in evaluating the summaries. Using weighted topic compositions, we measure the similarity of summary sentences with the most important topics identified in the document set.

We use Gibbs sampling (Griffiths, 2002) for inference in the topic model with concentration parameters $\alpha = 0.1$ and $\beta = 0.01$. We also set the number of topics $K = 10$ for each document set. Formally, each topic is defined as $\mathcal{T}_i = \{p_1, p_2, ..., p_n\}$, where p_j is the probability distribution of word w_j. We consider top $m = 30$ words and their probabilities to build a centroid as the representative of each topic. The embedding vector for word w_j is then multiplied with its normalized probability $\mathcal{P}_j$, and the weighted vectors are averaged to build a topic centroid representation:

$$\vec{\mathcal{T}_i} = \frac{1}{m}\sum_{j=1}^{m} \mathcal{P}_j \vec{w_j}, \quad where\ \mathcal{P}_j = \frac{p_j}{\sum_{i=1}^{m} p_i} \tag{7.4}$$

Finally, we use WMD to measure the dissimilarity degree between the centroid embeddings of summary sentences and those of the topics for evaluating topical relevance of the summary content.

7.2.3 Query Relevance

To measure the relevance degree of the summary content to the given query, we calculate the query embedding vector $\vec{Q}$ by averaging the embeddings of all words in the query narrative. Similarly, the centroid embedding vector for each summary $\vec{S}$ is also constructed. We further measure the cosine similarity between these vectors to formulate query relevance:

$$sim(\vec{S}, \vec{Q}) = \frac{\vec{S} \cdot \vec{Q}}{\| \vec{S} \| \| \vec{Q} \|} \tag{7.5}$$

7.2.4 Coherence

Coherence measures the degree to which a sequence of summary sentences represents a logical flow of thought. We compute the similarity between embeddings of adjacent summary sentences using cosine similarity. It results in $n-1$ comparisons for a summary of n sentences. While similarity between sentences is beneficial for coherence, very high similarity reflects redundancy in the summary. Given that, we combine the *mean* and *standard deviation* of the cosine similarity scores by training a simple linear regression model on our development set. In this way, we measure the trade-off between continuity and redundancy as the coherence feature.

7.2.5 Novelty

We would like our evaluation model to move beyond assessing initial summaries by giving a simple feature of Novelty to better evaluate update summaries. This feature rewards the update summary consisting of novel words that do not exist in initial document set D_A, but are semantically related to update document set

D_B. The relevancy of these words in update summary S_j, to the documents in set B, is measured using the cosine similarity between the embeddings of novel words and the centroid embedding of the whole document set B. We use the bag-of-words representation of the summary and the document sets while defining novel words. We finally measure the degree of novelty ($\mathcal{N}$) as:

$$\mathcal{N}(S_j) = \frac{1}{|S_j|} \sum_{w_i \in S_j | w_i \notin D_A} sim(\vec{w_i}, \vec{D_B}) \tag{7.6}$$

where $|S_j|$ is the total number of unique words in the update summary S_j. For S_j without any novel words, $\mathcal{N}(S_j) = 0$.

7.2.6 Feature Combination with SVR

We combine all the above features using a Support Vector Regression (SVR) model to predict the summary quality. We first transform the proposed features into a standard vector notation. Each summary S_i is represented by a feature vector $X = \{x_1, x_2, ..., x_n\}$ where n is the number of features. SVR model aims to learn a function $f : \mathbb{R}^n \mapsto \mathbb{R}$, which will be used to predict the content evaluation score for each summary $y \in \mathbb{R}$ given a feature vector $X \in \mathbb{R}^n$. In particular, given l training instances $(X_1, y_1), ..., (X_l, y_l)$, the SVR model is learnt by solving the following optimization problem (Vapnik, 1999); W is a vector of feature weights; ϕ is a function that maps feature vectors to a new vector space of higher dimensionality to allow non-linear functions to be learnt in the original space; $C > 0$ and $\epsilon > 0$ are given.

$$\min_{W,b,\xi,\xi^*} \frac{1}{2}\|W\|^2 + C\sum_{i=1}^{l} \xi_i + C\sum_{i=1}^{l} \xi_i^* \tag{7.7}$$

subject to (for $i = 1, ..., l$):

$$\begin{aligned} W^T \cdot \phi(X_i) + w_0 - y_i &\leq \epsilon + \xi_i \\ y_i - W^T \cdot \phi(X_i) - w_0 &\leq \epsilon + \xi_i^* \\ \xi_i &\geq 0 \\ \xi_i^* &\geq 0 \end{aligned}$$

The goal is to learn a linear (in the new space) function, whose prediction (value) $W^T \cdot \phi(X_i) + w_0$ for each training instance X_i will not to be farther than ϵ from the target (correct) value y_i. Since this is not always feasible, two slack variables ξ_i and ξ_i^* are used to measure the prediction's error above or below the target y_i. The objective (7.7) jointly minimizes the total prediction error and $\|W\|$, to avoid overfitting. The utilized SVR is implemented in Scikit-learn (Pedregosa et al., 2011). We use the default parameter settings, (*kernel='rbf', degree=3, gamma='auto', coef0=0.0, tol=0.001, C=1.0, epsilon=0.1, shrinking=True, cache_size=200, verbose=False, max_iter=-1*) without further optimization.

7.3 Experiments and Results

Reporting correlations with manual evaluation metrics is the norm for validating automatic metrics. We use the Spearman correlation metric to study the predictive power of our automatic features in replicating manual correlations of pyramid and responsiveness. Hence, we compare the rankings of systems against the human scores assigned to systems. The correlations achieved by the SIMetrix evaluation system (Louis and Nenkova, 2009a, 2013) are also included in our analysis. This system comprises multiple features to determine the quality of a summary, with a focus on computing divergences between the probability distributions of words in the input and summary. We consider Jensen Shannon Divergence (JSD) and feature

regression as the best metrics reported for SIMetrix. The correlations[3] are analyzed at two levels of granularity:

System Level (Macro)

The average score for a system is computed over the entire set of test inputs using both manual and automatic evaluations. The correlations between ranks assigned to systems by these average scores are indicative of the strength of our features to predict overall system rankings on the test set.

	Query - Macro		**Update - Macro**	
Features	**Pyr.**	**Resp.**	**Pyr.**	**Resp.**
Corpus-based Dist. Similarity	-0.887	-0.748	-0.833	-0.761
LexRes-based Dist. Similarity	-0.871	-0.723	-0.828	-0.755
Corpus-based Topical Relevance	-0.803	-0.696	-0.777	-0.720
LexRes-based Topical Relevance	-0.799	-0.705	-0.759	-0.735
LexRes-based Query Relevance	0.624	0.590	0.599	0.576
Corpus-based Query Relevance	0.615	0.547	0.613	0.576
LexRes-based Novelty	-	-	0.537	0.502
Corpus-based Novelty	-	-	0.530	0.500
Corpus-based Coherence	0.361	0.375	0.352	0.358
LexRes-based Coherence	0.353	0.362	0.349	0.358
Support Vector Regression	0.895	0.786	0.872	0.808
SIMetrix JS divergence	-0.880	-0.736	-0.827	-0.764
SIMetrix regression	0.867	0.705	0.789	0.605
ROUGE-1 recall (4 models)	0.859	0.806	0.912	0.865
ROUGE-2 recall (4 models)	0.905	0.873	0.941	0.884

Table 7.1: Input-summary evaluation on the query focused and update summarization tasks from TAC 2008 data: MACRO level Spearman correlations, all results are significant ($p < 0.05$).

[3]Significance values for the correlations are produced using the AS 89 algorithm (Best and Roberts, 1975).

Analyzing the macro level results on TAC 2008 (Table 7.1), we find that the variants of distributional similarity and the topical relevance features produce system rankings very similar to those produced by human. Other features, on the other hand, are less predictive of content quality. Distributional similarities also outperform SIMetrix, which proves the importance of semantic representation of the input and summary for comparison purposes in summary content evaluation. Overall, our feature regression obtains the best correlations with both types of manual scores, and even outperforms ROUGE-1 regarding pyramid for query-focused task. The usefulness of novelty feature is also reflected in high SVR correlation results for the update summarization task.

Overall ROUGE correlation is evidence that the model summaries provide information that is unlikely to ever be approximated by exploring the input alone. However, our features can provide reliable estimates of system quality when averaged over a set of test inputs. We also observe that corpus-based models mostly outperform their corresponding lexical resource-based models. A possible reason is the higher coverage of Google News word2vec model comparing to the WordNet-based sense embedding model. For example, some words like proper nouns (e.g., *'Barak Obama'*) are not covered in WordNet. However, replacing a word's embedding by the sum of the embeddings of its senses could generally improve the quality of embeddings (Rothe and Schütze, 2015). That is why our SVR performs well by leveraging WordNet senses for more precise word embeddings, and involving Google News to complement the WordNet coverage.

Input Level (Micro)

For each individual input, we compare the rankings for the system summaries using manual and automatic evaluations. Micro-level analysis highlights the ability of an evaluation metric to assess the quality of system summaries produced for a specific

input. This task is bound to be harder than system level predictions. For clarity, even with wrong prediction of rankings on a few inputs, the average scores (macro-level) for a system might not be affected.

	Query - Micro		**Update - Micro**	
Features	**Pyr.**	**Resp.**	**Pyr.**	**Resp.**
Corpus-based Dist. Similarity	75.0	72.9	77.1	70.8
LexRes-based Dist. Similarity	72.9	68.8	77.1	68.8
Corpus-based Topical Relevance	72.9	70.8	75.0	72.9
LexRes-based Topical Relevance	70.8	70.8	72.9	70.8
LexRes-based Query Relevance	58.3	58.3	62.5	56.3
Corpus-based Query Relevance	56.3	52.1	60.4	56.3
LexRes-based Novelty	-	-	54.2	50.0
Corpus-based Novelty	-	-	58.3	45.8
Corpus-based Coherence	37.5	39.6	41.7	35.4
LexRes-based Coherence	35.4	37.5	37.5	37.5
Support Vector Regression	79.2	75.0	87.5	77.1
SIMetrix JS divergence	72.9	72.9	85.4	75.0
SIMetrix regression	77.1	66.7	81.3	58.3
ROUGE-1 recall (4 models)	97.9	95.8	97.9	95.8
ROUGE-2 recall (4 models)	100	91.7	100	91.7

Table 7.2: Input-summary evaluation on the query focused and update summarization tasks from TAC 2008 data: MICRO level percentage of inputs with significant correlations ($p < 0.05$).

To be in line with SIMetrix, we report the percentage of inputs for which significant correlations were obtained (Table 7.2). We observe that feature combination with SVR gives the best results overall, similar to our findings for the macro level. The implication is that no single feature can reliably predict good content for a particular input. Moreover, our feature regression outperforms SIMetrix. This is because our approach depends not merely on the distribution of terms in the input, and therefore provides better representation for a set of documents each describing different opinion on a given issue. For example, our topical relevance feature gives a representative vector for every important aspect of the document set. However,

superiority of ROUGE performance to the rest of measures shows that model summaries generated for specific input would still give better indication of important information in the input.

7.3.1 Error Analysis

In this study, we aim to assess the reliability of our metric for evaluation in the absence of human model summaries, where ROUGE cannot be used. It is noteworthy that we do not intend to directly compare the performance of ROUGE with our metric. Thereupon, we provide an error analysis to understand if our SVR and ROUGE are making errors in ordering the same systems or whether their errors are different. Since at the macro level, the correlations between our regression and pyramid scores are close to those of ROUGE-2, we further analyze their errors. We considered pairs of systems and identified the better system in each pair according to the pyramid scores. Afterwards, we recorded how often ROUGE-2 and the SVR provided the correct judgment for the pairs as indicated by the pyramid evaluation. Table 7.3 provides the results for all 1,653 pairs of systems at the macro level.

	SVR correct	**SVR incorrect**
Rouge-2 correct	1,355(82.0%)	97(5.9%)
Rouge-2 incorrect	100(6.0%)	101(6.1%)

Table 7.3: Error analysis: Overlap between ROUGE-2 and SVR predictions for the best system in a pair (TAC 2008, 1,653 pairs). The gold-standard judgment for a better system is computed using pyramid.

A large majority (82%) of the same pairs are correctly predicted by both ROUGE and the SVR. Another 6% of the pairs are such that both metrics do not provide the correct judgment. Therefore, ROUGE and our SVR appear to agree on a large majority of the system pairs. There is a small percentage (12%) that is correctly

predicted by only one of the metrics.

7.3.2 Evaluation on the Test Set

Our SVR was trained on the TAC 2008 data with pyramid scores as the target. Herein, we evaluate this metric using the TAC 2009 data (Table 7.4 and 7.5). We report the correlations obtained by ROUGE-SU4 as the official baseline measure at TAC 2009 for comparison of automatic evaluation metrics. The results indicate that the correlations are lower than on our development set. This might be caused by the different characteristics of inputs in two year's data (Louis and Nenkova, 2013). However, the SVR is consistently predictive across two years, and outperforms SIMetrix.

	Query - Macro		**Update - Macro**	
Metric	**Pyr.**	**Resp.**	**Pyr.**	**Resp.**
Support Vector Regression	0.80	0.75	0.77	0.65
SIMetrix JS divergence	-0.74	-0.71	-0.72	-0.61
SIMetrix Regression	0.77	0.67	0.71	0.54
ROUGE-SU4 (4 models)	0.92	0.79	0.85	0.69

Table 7.4: Input-summary evaluation on the query focused and update summarization tasks from TAC 2009 data: MACRO level Spearman correlations, all results are significant ($p < 0.05$).

Overall results also show that correlations with pyramid scores are higher than those with responsiveness. The reason is that our features mainly evaluate summary content. Responsiveness judgments, on the other hand, are based on both content and linguistic quality. Nevertheless, our SVR performs better than SIMetrix in replicating responsiveness scores, perhaps because SVR considers coherence as a linguistic quality feature. Hence, a natural extension of our work would be considering more

	Query - Micro		Update - Micro	
Metric	**Pyr.**	**Resp.**	**Pyr.**	**Resp.**
Support Vector Regression	87.5	77.1	79.2	75.0
SIMetrix JS divergence	84.1	75.0	77.3	72.7
SIMetrix Regression	81.8	65.9	75.0	52.3
ROUGE-SU4 (4 models)	95.5	81.8	100	86.4

Table 7.5: Input-summary evaluation on the query focused and update summarization tasks from TAC 2009 data: MICRO level percentage of inputs with significant correlations ($p < 0.05$).

linguistic quality features along with content evaluations.

7.4 Summary

We have presented an effective model-free summary content evaluation approach that exploits the compositional properties of word and sense embeddings to develop a variety of features for input-summary comparisons. The experiment results show that the strength of different features varies considerably, and their combination provides reliable estimates of summary content quality when model summaries are not available. This lends further support to our proposal to use semantic representations of the input and summary contents for the model-free summary content evaluation.

Chapter 8

Conclusion

The goal of this thesis has been to investigate approaches toward abstractive text summarization, and to make this task: (i) more adaptable to a wide range of applications; (ii) more dynamic to different sources and types of texts; and (iii) better evaluated using semantic representations. To achieve these goals, we have focused on five studies: (i) enhancing word graph-based multi-sentence compression through merging, mapping, and re-ranking strategies; (ii) adapting the enhanced MSC word graph into query-focused multi-document summarization for newswire; (iii) integrating general and domain-specific knowledge sources for making this task more dynamic to summarize clinical texts; (iv) measuring both semantic and lexical similarities for computing ROUGE scores to fairly evaluate abstractive summaries; (v) predicting summary content quality in the absence of human model summaries using word and sense embeddings.

This chapter first summarizes the research questions, proposed methods and findings of each chapter of this thesis. Then, Section 8.2 discusses the limitations of our work and outlines future research directions.

8.1 Summary of Chapters and Contributions

In **Chapter 2**, we provided a detailed definition of the automatic text summarization, and discussed the objectives of this task including information coverage, significance, redundancy, and cohesion. Then, we introduced the most famous types of text summarization in addition to their individual characteristics. We explained the conventional framework for this task, reviewed the computational literature, and explained the challenges and approaches to extractive and abstractive summarization. In addition, we discussed recent efforts and progress made toward abstractive text summarization and evaluation.

A large part of this thesis focuses on multi-sentence compression, query-focused multi-document summarization, specific-domain summarization, and model-based and model-free evaluation of summaries. Therefore, we surveyed research on compressive summarization, full abstraction, abstraction in specific domains and genres, and automatic evaluation of text summarization. Our review showed that while many studies are still focusing on improving extractive summarization from various aspects, there is also a strong emerging favorite toward more abstractive text summarization, with compressive summarization being particularly popular as an intermediate step. Also much progress has been made in summarizing under various settings or genres of texts. Furthermore, considering semantic representations of texts can improve the reliability of evaluation metrics for this task.

At the end of Chapter 2, we explained the importance of automatic text summarization, in reviewing previous studies which investigate the integration of text summarization into NLP applications. This review shows that most previous research has chosen extractive summarization in the NLP applications. A possible reason is that abstractive text summarization is more complex than extractive level and needs more sophisticated techniques. However, when necessary information exists

in several parts of the text and simply copying and aggregating them does not make sense, we need an abstractive summarizer that will allow us to write new summarized content from the aggregated information. An ideal abstractive summarizer will always produce more coherent and polished summaries than extractive level.

In **Chapter 3**, we proposed an enhanced word graph-based multi-sentence compression approach. This approach tackles one of the pain points of MSC, which is improving both informativity and grammaticality at the same time, and helps to make MSC more effective and applicable to a wide range of applications. The proposed approach in this chapter comprises three strategies: (i) *a merging strategy* based on Multiword Expressions; (ii) *a mapping strategy* based on the notion of synonymy; and (iii) *a grammar-enhanced re-ranking strategy* based on POS tags.

Our proposed word graph is firstly constructed by adding the first sentence and displays words in a sentence as a sequence of connected nodes. The merging strategy is based on the assumption that identifying MWEs in the source text and merging their components can reduce the ambiguity of mapping upcoming words in the graph. The mapping strategy is also proposed for mapping upcoming single words using the concept of synonymy. The weight of the selected word as the representative is computed by summing the frequency scores from its synonyms. Given that, the number of total possible paths (compression candidates) is decreased, and the weight of frequent similar words with different appearances in the content is better reflected.

A heuristic algorithm is then used to find the k-shortest paths in the graph to make the compression candidates. These candidates are firstly re-ranked using the TextRank algorithm based on the assumption that a word can recommend other co-occurring words, and the strength of the recommendation is recursively computed based on the importance of the words making the recommendation. The next re-ranking strategy benefits from the fact that POS tags capture the syntactic roles of words in a sentence. For this purpose, we trained a 7-gram POS-based Language

Model to assign a grammaticality score to each generated compression. Finally, the path that has the lightest average edge weight is considered as the best compression.

For the evaluation purpose, we constructed a dataset made of clusters of English newswire sentences. Hence, we evaluated our approach using this dataset, by conducting manual and automatic (ROUGE and BLEU) evaluations. The experiment results showed that our approach is superior to the competitive baselines. The merging and mapping strategies, along with the grammar-enhanced POS-LM re-ranking method, enhanced both informativity and grammaticality of the compressions, with an improved compression ratio. We also investigated the strengths and weaknesses of each strategy through an ablation study. This approach can be used as an abstractive summarizer in a wide range of applications. We showed this potential by incorporating this MSC word graph as a component in the proposed approaches of the next two chapters.

Chapter 4 proposed a query-focused abstractive summarization approach to summarize multiple news documents. Given a query and a set of news documents, our approach summarizes the source documents to answer the query with the aim of satisfying query-biased relevance, biased information novelty, and biased information richness. For this purpose, sentence-to-sentence and sentence-to-query semantic similarities are captured by performing repetitive random walks over WordNet. An alignment-based sense disambiguation algorithm is also employed for leveraging the content of the paired sentence in order to disambiguate each element. Furthermore, less query-relevant sentences are filtered out through a similarity graph, and the remained sentences are clustered using a graph-based clustering algorithm. A well-organized and informative summary is finally generated for the clusters of query-relevant sentences. This component considers the important key-phrases, along with the grammatical structure of the generated summaries. We studied the importance of separate components in our approach by conducting a set of experiments,

where we used automatic evaluation metric over the DUC benchmark datasets. A further syntactic analysis is also performed using the link grammar parser. The overall experiment results showed that our method outperforms the state-of-the-art approaches.

In **Chapter 5**, we presented an effective approach to integrate WordNet into UMLS for abstractly summarizing biomedical texts. Given a clinical query, our approach generates an informative and grammatical summary for a set of biomedical abstracts. Keeping an eye on the biomedical peculiarities, we captured sentence-to-sentence and sentence-to-query semantic similarities using both WordNet and UMLS. Considering UMLS Metathesaurus as our semantic graph, concepts are represented as nodes, and the relations listed in the MRREL table as directed edges. Furthermore, we used the MetaMap program to map each sentence to concepts from the UMLS Metathesaurus and semantic types from the UMLS Semantic Network. Using the built-in WSD module, MetaMap allows to disambiguate terms and return directly the relevant concept. We also studied summarizing capability of the proposed abstractive summarization framework in Chapter 4 for biomedical summarization.

The experiment results over the specialized evidence-based medicine corpus indicate that our approach outperforms the two competitive systems. Moreover, we have conducted a set of experiments using the WordNet-based and/or UMLS-based semantic similarities, and obtained the best result while using both scores. This demonstrates the importance of using both WordNet and UMLS to capture the semantic similarities. Three different baselines for sentence selection have also been used, each aiming to construct a different type of summary according to the type of information in various parts of the source. It has been found that the last part of each abstract is more likely to be included in the summary. We have tackled the main issue faced by state-of-the-art biomedical summarizers (i.e., decline in summarization performance due to the poor UMLS coverage of non-medical concepts

in the documents to be summarized) (Plaza et al., 2011). This issue is addressed by using WordNet to represent the layman knowledge, and UMLS to represent the professional knowledge. This approach can bridge the knowledge and language gaps in biomedical summarizers.

Chapter 6 discussed an effective approach (ROUGE-G) to overcome the limitation of high lexical dependency in ROUGE. Our approach leverages a sense-based representation to calculate PPR vectors for n-grams. Given a pair of peer and model summaries, we compute and compare PPR vectors at the following levels: (i) *sense level*, to disambiguate each word; and (ii) *n-gram level*, to measure the semantic similarities. The PPR vectors of each pair of n-grams have been compared using the following measures: (i) *Path-based*, which considers the path that the random walker takes at each iteration to get to a particular node; (ii) *Rank and Weight*, which weights the overlaps between a pair of ranked PPR vectors. To take OOV words into consideration, we performed an OOV handling mechanism, where we introduced new dimensions in the resulting PPR vector, one for each OOV term.

We finally improved on ROUGE by performing both semantic and lexical analysis of summaries. Evaluation is processed by comparing each n-gram in the model summary against the corresponding peer summary text. Using the PPR algorithm, all senses have been disambiguated before comparison. Treating the peer summary text as one unit not only reduces comparison time and increases the efficiency, but also provides a suitable number of content words which guarantees implicit word sense disambiguation, and semantic relationship derivation. We have evaluated our approach with the following settings for computing and comparing PPR vectors: (i) *Path-based* with *Rank and Weight* measure (current setting); (ii) *Path-based* with *cosine similarity*; (iii) Excluding *path-based* measure and using *Rank and Weight* measure solely. The results showed that the current setting performs better than the other two. Overall experiment results over the TAC AESOP datasets showed

that ROUGE-G achieves higher correlations with manual judgments in comparison with the well-established ROUGE. This approach has the potential to expand the applicability of ROUGE to fairly evaluate abstractive summaries.

Finally, in **Chapter 7**, we proposed an effective model-free summary content evaluation approach that exploits the compositional properties of word and sense embeddings to develop a variety of features for input-summary comparisons. The proposed features to assess the quality of summary content in the absence of model summaries are: (i) *Distributional Semantic Similarity*; (ii) *Topical Relevance*; (iii) *Query Relevance*; (iv) *Coherence*; and (v) *Novelty*. We then provided a feature combination using an SVR model to predict the summary quality.

The effectiveness of our proposed approach is demonstrated by conducting a set of experiments at two levels of granularity: (i) *Macro/System Level*; and (ii) *Micro/Summary Level*, over the TAC summarization datasets. The experiment results showed that quantifying the indicators of content quality by taking advantage of compositional properties of the word and sense embeddings produces summary scores which accurately replicate human assessments. We also conducted an error analysis to assess the reliability of our metric for evaluation in the absence of human model summaries, where ROUGE cannot be used. This approach provided some insights into how semantic representations of the input and summary contents are useful for the model-free summary content evaluation. It is noteworthy that our approach complements but is not intended to replace existing model-based evaluation approaches, since their reliability and strength are important for high confidence evaluations.

8.2 Future Work

This thesis detailed semantically motivated approaches to: (i) multi-sentence compression; (ii) query-focused multi-document summarization; (iii) domain-specific multi-document summarization; and (iv) model-based and model-free summarization evaluation. This section first outlines potential short-term extensions to our proposed approaches in each chapter, and then returns to discuss untouched directions for future research.

8.2.1 Short-term Extensions

In Chapter 3, we proposed a multi-sentence compression approach for summarizing clusters of relevant sentences. This work demonstrates our attempt in using MWEs, Synonymy and POS-based language modeling to tackle one of the pain points of MSC, which is improving both informativity and grammaticality at the same time. We used version 3.0 of WordNet (Miller, 1995) to obtain available synonym words for replacing a merged MWE, and also for mapping upcoming single words in the word graph. However, WordNet has a limited coverage comparing to a larger semantic network like BabelNet[1] introduced by Navigli and Ponzetto (2010). Our intuition is that the use of BabelNet or words/phrases embeddings for capturing semantic relations between linguistic items can be a further extension to this work.

In Chapter 4, we proposed an abstractive query-focused summarizer for newswire. Given a query and a set of news documents, our approach summarizes the documents to answer the query with the aim of satisfying query-biased relevance, biased information novelty, and biased information richness. For satisfying the information

[1]BabelNet is automatically constructed by means of a methodology that integrates lexicographic and encyclopedic knowledge from WordNet and Wikipedia; available at http://lcl.uniroma1.it/babelnet/

richness criterion, we utilized the TextRank (Mihalcea and Tarau, 2004) algorithm (Section 4.1.7) to re-rank the summaries based on their keyphrases. We noticed that the most recent work (Mahata et al., 2018) has proposed an unsupervised technique (Key2Vec) that leverages phrase embeddings for ranking keyphrases extracted from scientific articles. At the time of writing this thesis, this approach is the first attempt in using multiword phrase embeddings for constructing thematic representation of a given document and to assign thematic weights to phrases for ranked keyphrase extraction. This motivates an extension by incorporating this methodology in our approach to capture semantic and syntactic similarities between textual units comprising of both single word and multiword phrases.

In Chapter 5, we have presented the first attempt at integrating WordNet into UMLS for summarizing biomedical texts. A medium sized corpus, which is the only available corpus for evidence-based biomedical summarization, was used in our experiments. Hence, for some features, there was not enough data available for the generation of statistics. For example, the corpus only contains a few samples for some of the question types, e.g., History and Device. Having a larger corpus would make the statistics associated with sparse data more reliable. Therefore, our ongoing work could be constructing a larger corpus for evidence-based medical summarization.

In Chapter 6, we have proposed a model-based summarization evaluation approach (ROUGE-G) to overcome the limitation of high lexical dependency in ROUGE. The idea is moving away from purely lexical summarization evaluation measures in order to fairly evaluate abstractive summaries including lexical variations and paraphrasing. In order to demonstrate the effectiveness of ROUGE-G to fairly evaluate abstractive summaries, we need to conduct experiments on a dataset composed of abstractive summaries. However, we evaluated our approach over the TAC 2011 AESOP dataset which is made of summaries that were generated mostly by extractive

systems. Since, there is not such dataset at the time of writing this thesis, we can continue building on this work by using model summaries, which are abstractive in nature, as a proxy. Model summaries are manually evaluated in terms of responsiveness and linguistic quality. Thereupon, it is possible to incorporate the *jackknifing* procedure in the scoring process in order to see whether our metric can differentiate between peer summaries (naturally extractive) vs. model summaries (naturally abstractive). In this procedure, each model summary should be evaluated four times, each time against a different subset of three human model summaries. The final score for the automatic summary will be the mean of the four scores. This process ensures a fair evaluation, since each human summary can only be evaluated against three (remaining) model summaries. Utilizing the Paraphrase Database (PPDB[2]) (Ganitkevitch et al., 2013) for evaluating could also be an ongoing direction.

In Chapter 7, we have presented a model-free summary content evaluation approach. This work exploits the compositional properties of word and sense embeddings to develop a variety of features for input-summary comparisons. We limited this work to considering distributional semantics and using their compositionality. Our ongoing work includes considering distributional and relational semantics together (Fried and Duh, 2014; Verga and McCallum, 2016; Rossiello, 2016) for different sentence representations, and using more complex neural language models (Le and Mikolov, 2014; Zhang and LeCun, 2015; Jozefowicz et al., 2016) for the comparison.

In another direction based on the experiment results in Section 7.3.2, we found that correlations with pyramid scores are higher than those with responsiveness. The reason is that our proposed features in this chapter mainly evaluate summary content, while responsiveness judgments are based on both content and linguistic

[2]PPDB is an automatically extracted database containing millions of paraphrases in 16 different languages. The goal of PPBD is to improve language processing by making systems more robust to language variability and unseen words; available at http://paraphrase.org/

quality. Nevertheless, we considered coherence as a linguistic quality feature, which led to superiority of our SVR to SIMetrix in replicating responsiveness scores. Hence, another extension of our work would be considering more linguistic quality features along with content evaluations.

8.2.2 Future Directions

Along with the proposed improvements above, there are some related areas that we did not touch on in this thesis. This section explains these areas.

Constructing Datasets for Abstractive Summarization

In Chapter 3, we constructed a dataset made of clusters of English newswire for multi-sentence compression. However, we did not focus on constructing large scale datasets or datasets containing lexical variations and paraphrasing for the purpose of abstractive text summarization and evaluation. The current standard datasets for text summarization tasks, particularly for multi-document summarization, are mostly of small scale. This hampers the progress of machine learning-based approaches to summarize, especially of non-English texts and domain-specific texts, that suffer more from a lack of data. Consequently, research in other domains and languages is limited. Constructing high-quality summarization datasets is an important future direction that will help improving this field. Cao et al. (2016a) and Hu et al. (2015) proposed preliminary approaches in collecting large scale data for producing short news summaries using microblogs. However, data preparation for other different genres is still an open research area. Utilizing external resources or additional background corpora can also temporarily help summarizers in capturing important information (Li et al., 2015a; Zopf et al., 2016a).

The necessity to prepare high-quality data appears more obviously in summarization evaluation, especially in evaluating abstractive summaries. This is a different issues than scale. For better clarity, current studies on abstractive summarization evaluation have to use data which is drawn from systems participating in the TAC summarization tasks, where there is a strong exhibited bias towards extractive summarizers (Ng and Abrecht, 2015). It will be helpful to enlarge this set of summaries to include output from summarizers which carry out substantial paraphrasing (Li et al., 2013; Ng et al., 2014; Liu et al., 2015a).

Summarizing Using User Interactions

Another untouched direction in this thesis is to involve user interactions in automatic text summarization. Certain level of personalization or user interaction is required due to different users demands. A user may modify the input queries based on the previous summaries received from the system. This idea has been studied as a query-chain summarization task (Baumel et al., 2014), where a series of relevant queries are considered, and an update summary is constructed for each query in the chain. In Chapter 7, we investigated the strength of our proposed features in model-free evaluation of update summaries. However, we mainly focused on initial summaries throughout the thesis. A system can also perform summarization in a hierarchical fashion. A user may click on a sentence from a global summary and get to see a more detailed, focused summarization for the point of that sentence (Christensen et al., 2014).

Summarizing Data Streams

Our third suggestion for future study is to summarize large-scale data streams. The main motivation for automatic text summarization is the explosion of information.

However, much current research focuses on summarization of news data using standard benchmark datasets, with a relatively small number of documents. Real data sometimes come in streams and may have different formats, including news texts and all kinds of user-generated contents (Ge et al., 2016; Olariu, 2014; Zopf et al., 2016b). Most proposed methods for generic summarization may not be trivially adaptable to large-scale streaming data with possible loss of either efficiency or effectiveness (Yao et al., 2017). To this end, more specific treatments are required to handle the challenges of event detection, dynamic modeling, contextual dependency, information fusion and credibility assessment.

Harnessing Deep Learning for Abstractive Summarization

Harnessing deep neural network models for abstractive summarization is also an important direction which is untouched in this thesis. Representation learning based on neural network architectures has proven to be useful in some natural language processing tasks that involve text rewriting, such as machine translation (Yao et al., 2017). Currently, some preliminary studies on text summarization have been made using end-to-end training (Nallapati et al., 2016). However, current naive RNN encoder-decoder structures fail to encode documents or longer and more structured texts compared to the current input sentences. A better hierarchical encoding and attention with multiple levels on both words and sentences (Li et al., 2015b) are perhaps required. We also need external memory units (Sukhbaatar et al., 2015) for storing distant but more significant information. Furthermore, explicitly designing latent variable structures to capture discourse relations between sentences (Ji et al., 2016) may help the document encoding process. Utilizing an intra-attention decoder and combined training objective could also be helpful in sequence-to-sequence tasks with long inputs and outputs (Paulus et al., 2017).

Bibliography

Asad Abdi, Norisma Idris, Rasim M Alguliyev, and Ramiz M Aliguliyev. Query-based multi-documents summarization using linguistic knowledge and content word expansion. *Soft Computing*, pages 1–17, 2015.

Otavio Costa Acosta, Aline Villavicencio, and Viviane P Moreira. Identification and treatment of multiword expressions applied to information retrieval. In *Proceedings of the Workshop on Multiword Expressions: from Parsing and Generation to the Real Worlds (MWE 2011)*, pages 101–109. ACL, 2011.

Stergos Afantenos, Vangelis Karkaletsis, and Panagiotis Stamatopoulos. Summarization from medical documents: a survey. *Artificial Intelligence in Medicine*, 33 (2):157–177, 2005.

Enrique Alfonseca, Daniele Pighin, and Guillermo Garrido. Heady: News headline abstraction through event pattern clustering. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (ACL 2013)*, volume 1, pages 1243–1253. ACL, 2013.

Miguel Almeida and Andre Martins. Fast and robust compressive summarization with dual decomposition and multi-task learning. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (ACL 2013)*, volume 1, pages 196–206. ACL, 2013.

Alan R Aronson. Metamap: Mapping text to the umls metathesaurus. *National Library of Medicine (NLM), National Institutes of Health (NIH), Bethesda, Maryland*, pages 1–26, 2006.

Rachit Arora and Balaraman Ravindran. Latent dirichlet allocation based multi-document summarization. In *Proceedings of the 2nd Workshop on Analytics for Noisy Unstructured Text Data (AND 2008)*, pages 91–97. ACM, 2008.

Ron Artstein and Massimo Poesio. Inter-coder agreement for computational linguistics. *Computational Linguistics*, 34(4):555–596, 2008.

Sofia J Athenikos and Hyoil Han. Biomedical question answering: A survey. *Computer Methods and Programs in Biomedicine*, 99(1):1–24, 2010.

Shiqi Shen Ayana, Zhiyuan Liu, and Maosong Sun. Neural headline generation with minimum risk training. *arXiv preprint arXiv:1604.01904*, 2016.

Rama Badrinath, Suresh Venkatasubramaniyan, and CE Veni Madhavan. Improving query focused summarization using look-ahead strategy. In *Proceedings of the 33rd European Conference on Information Retrieval (ECIR 2011)*, pages 641–652. Springer, 2011.

Ramakrishna Bairi, Rishabh Iyer, Ganesh Ramakrishnan, and Jeff Bilmes. Summarization of multi-document topic hierarchies using submodular mixtures. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (ACL-IJCNLP 2015)*, volume 1, pages 553–563. ACL, 2015.

Timothy Baldwin and Su Nam Kim. Multiword expressions. *Handbook of Natural Language Processing, Second Edition. Morgan and Claypool*, 2010.

Siddhartha Banerjee, Prasenjit Mitra, and Kazunari Sugiyama. Multi-document abstractive summarization using ilp based multi-sentence compression. In *Proceed-*

ings of the 28th International Joint Conference on Artifical Intelligence (IJCAI 2015), pages 1208–1214, 2015.

Marco Baroni, Georgiana Dinu, and Germán Kruszewski. Don't count, predict! a systematic comparison of context-counting vs. context-predicting semantic vectors. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (ACL 2014)*, pages 238–247, 2014.

Regina Barzilay and Noemie Elhadad. Inferring strategies for sentence ordering in multidocument news summarization. *Journal of Artificial Intelligence Research (JAIR)*, 17:35–55, 2002.

Regina Barzilay and Kathleen R McKeown. Sentence fusion for multidocument news summarization. *Computational Linguistics*, 31(3):297–328, 2005.

Tal Baumel, Raphael Cohen, and Michael Elhadad. Query-chain focused summarization. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (ACL 2014)*, volume 1, pages 913–922. ACL, 2014.

Taylor Berg-Kirkpatrick, Dan Gillick, and Dan Klein. Jointly learning to extract and compress. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics (ACL 2011)*, pages 481–490. ACL, 2011.

DJ Best and DE Roberts. Algorithm as 89: the upper tail probabilities of spearman's rho. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 24(3): 377–379, 1975.

Chris Biemann. Chinese whispers: an efficient graph clustering algorithm and its application to natural language processing problems. In *Proceedings of the 1st Workshop on Graph-Based Methods for Natural Language Processing (TextGraphs 2006)*, pages 73–80. ACL, 2006.

Lidong Bing, Piji Li, Yi Liao, Wai Lam, Weiwei Guo, and Rebecca Passonneau. Abstractive multi-document summarization via phrase selection and merging. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (ACL-IJCNLP 2015)*, volume 1, pages 1587–1597. ACL, 2015.

David M Blei, Andrew Y Ng, and Michael I Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research (JMLR)*, 3(Jan):993–1022, 2003.

Olivier Bodenreider. The unified medical language system (umls): integrating biomedical terminology. *Nucleic Acids Research*, 32(suppl 1):D267–D270, 2004.

Florian Boudin and Emmanuel Morin. Keyphrase extraction for n-best reranking in multi-sentence compression. In *Proceedings of the 2013 Annual Conference of the North American Chapter of the Association for Computational Linguistics - Human Language Technologies (NAACL HLT 2013)*. ACL, 2013.

Anita Burgun and Olivier Bodenreider. Comparing terms, concepts and semantic classes in wordnet and the unified medical language system. In *Proceedings of the NAACL2001 Workshop, WordNet and Other Lexical Resources: Applications, Extensions and Customizations*, pages 77–82. ACL, 2001.

Luis Adrián Cabrera-Diego and Juan-Manuel Torres-Moreno. Summtriver: A new trivergent model to evaluate summaries automatically without human references. *Data & Knowledge Engineering (DKE)*, 2017.

Ziqiang Cao, Furu Wei, Li Dong, Sujian Li, and Ming Zhou. Ranking with recursive neural networks and its application to multi-document summarization. In *Proceedings of the 29th Conference on Artificial Intelligence (AAAI 2015)*, pages 2153–2159, 2015a.

Ziqiang Cao, Furu Wei, Sujian Li, Wenjie Li, Ming Zhou, and WANG Houfeng. Learning summary prior representation for extractive summarization. In *Proceed-*

ings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (ACL-IJCNLP 2015), volume 2, pages 829–833. ACL, 2015b.

Ziqiang Cao, Chengyao Chen, Wenjie Li, Sujian Li, Furu Wei, and Ming Zhou. Tgsum: Build tweet guided multi-document summarization dataset. In *Proceedings of the 30th Conference on Artificial Intelligence (AAAI 2016)*, pages 2906–2912, 2016a.

Ziqiang Cao, Wenjie Li, Sujian Li, Furu Wei, and Yanran Li. Attsum: Joint learning of focusing and summarization with neural attention. *arXiv preprint arXiv:1604.00125*, 2016b.

Ziqiang Cao, Wenjie Li, Sujian Li, and Furu Wei. Retrieve, rerank and rewrite: Soft template based neural summarization. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL 2008)*, volume 1, pages 152–161. ACL, 2018.

Jaime Carbonell and Jade Goldstein. The use of mmr, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research & Development in Information Retrieval (SIGIR 1998)*, pages 335–336. ACM, 1998.

Giuseppe Carenini, Jackie Chi Kit Cheung, and Adam Pauls. Multi-document summarization of evaluative text. *Computational Intelligence*, 29(4):545–576, 2013.

Asli Celikyilmaz and Dilek Hakkani-Tur. A hybrid hierarchical model for multi-document summarization. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics (ACL 2010)*, pages 815–824. ACL, 2010.

Hakan Ceylan, Rada Mihalcea, Umut Özertem, Elena Lloret, and Manuel Palomar. Quantifying the limits and success of extractive summarization systems across

domains. In *Proceedings of the 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics - Human Language Technologies (NAACL HLT 2010)*, pages 903–911. ACL, 2010.

Deepayan Chakrabarti and Kunal Punera. Event summarization using tweets. In *Proceedings of the 5th International AAAI Conference on Weblogs and Social Media (ICWSM 2011)*, 2011.

Yllias Chali and Sadid A Hasan. On the effectiveness of using sentence compression models for query-focused multi-document summarization. *Proceedings of the 24th International Conference on Computational Linguistics (COLING 2012)*, pages 457–474, 2012a.

Yllias Chali and Sadid A Hasan. Query-focused multi-document summarization: Automatic data annotations and supervised learning approaches. *Natural Language Engineering (NLE)*, 18(1):109–145, 2012b.

Yllias Chali, Sadid A Hasan, and Shafiq R Joty. Improving graph-based random walks for complex question answering using syntactic, shallow semantic and extended string subsequence kernels. *Information Processing & Management*, 47(6): 843–855, 2011.

Wen Chan, Xiangdong Zhou, Wei Wang, and Tat-Seng Chua. Community answer summarization for multi-sentence question with group l 1 regularization. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (ACL 2012)*, pages 582–591. ACL, 2012.

Gong Cheng, Danyun Xu, and Yuzhong Qu. Summarizing entity descriptions for effective and efficient human-centered entity linking. In *Proceedings of the 24th International Conference on World Wide Web (WWW 2015)*, pages 184–194. ACM, 2015.

Jianpeng Cheng and Mirella Lapata. Neural summarization by extracting sentences and words. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL 2016)*, volume 1, pages 484–494, 2016.

Jackie Chi Kit Cheung and Gerald Penn. Unsupervised sentence enhancement for automatic summarization. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP 2014)*, pages 775–786. ACL, 2014.

Sumit Chopra, Michael Auli, and Alexander M Rush. Abstractive sentence summarization with attentive recurrent neural networks. In *Proceedings of the 2016 Annual Conference of the North American Chapter of the Association for Computational Linguistics - Human Language Technologies (NAACL HLT 2016)*, pages 93–98. ACL, 2016.

Janara Christensen, Stephen Soderland, Gagan Bansal, et al. Hierarchical summarization: Scaling up multi-document summarization. In *Proceedings of the 52nd annual meeting of the association for computational linguistics (ACL 2014)*, volume 1, pages 902–912. ACL, 2014.

James Clarke and Mirella Lapata. Models for sentence compression: A comparison across domains, training requirements and evaluation measures. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th Annual Meeting of the Association for Computational Linguistics (COLING-ACL 2006)*, pages 377–384. ACL, 2006.

James Clarke and Mirella Lapata. Modelling compression with discourse constraints. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL 2007)*, pages 1–11. ACL, 2007.

Arman Cohan and Nazli Goharian. Scientific article summarization using citation-context and article's discourse structure. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP 2015)*, pages 390–400. ACL, 2015.

William W Cohen, Robert E Schapire, and Yoram Singer. Learning to order things. In *Proceedings of the 12th Annual Conference on Neural Information Processing Systems (NIPS 1998)*, pages 451–457, 1998.

John M Conroy and Dianne P O'leary. Text summarization via hidden markov models. In *Proceedings of the 24th Annual International ACM SIGIR Conference on Research & Development in Information Retrieval (SIGIR 20)*, pages 406–407. ACM, 2001.

John M Conroy, Judith D Schlesinger, and Dianne P O'Leary. Topic-focused multi-document summarization using an approximate oracle score. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th Annual Meeting of the Association for Computational Linguistics (COLING-ACL 2006)*, pages 152–159. ACL, 2006.

Herma CH Coumou and Frans J Meijman. How do primary care physicians seek answers to clinical questions? a literature review. *Journal of the Medical Library Association (JMLA)*, 94(1):55, 2006.

Hoa Trang Dang. Overview of duc 2005. In *Proceedings of the 2005 Document Understanding Conference (DUC 2005)*, pages 1–12, 2005.

Dipanjan Das and André FT Martins. A survey on automatic text summarization. *Literature Survey for the Language and Statistics II course at CMU*, 4:192–195, 2007.

Sashka T Davis, John M Conroy, and Judith D Schlesinger. Occams–an optimal combinatorial covering algorithm for multi-document summarization. In *Proceedings*

of the12th IEEE International Conference on Data Mining Workshops (ICDMW 2012), pages 454–463. IEEE, 2012.

Dina Demner-Fushman and Jimmy Lin. Answering clinical questions with knowledge-based and statistical techniques. *Computational Linguistics*, 33(1): 63–103, 2007.

Giuseppe Di Fabbrizio, Amanda Stent, and Robert Gaizauskas. A hybrid approach to multi-document summarization of opinions in reviews. In *Proceedings of the 8th International Natural Language Generation Conference (INLG 2014)*, pages 54–63, 2014.

Robert L Donaway, Kevin W Drummey, and Laura A Mather. A comparison of rankings produced by summarization evaluation measures. In *Proceedings of the 2000 North American Chapter of the Association for Computational Linguistics - Applied Neural Language Processing Conference: Workshop on Automatic Summarization (NAACL-ANLP 2000)*, pages 69–78. ACL, 2000.

Yajuan Duan, Zhumin Chen, Furu Wei, Ming Zhou, and Heung-Yeung Shum. Twitter topic summarization by ranking tweets using social influence and content quality. *Proceedings of the 24th International Conference on Computational Linguistics (COLING 2012)*, pages 763–780, 2012.

Greg Durrett, Taylor Berg-Kirkpatrick, and Dan Klein. Learning-based single-document summarization with compression and anaphoricity constraints. *arXiv preprint arXiv:1603.08887*, 2016.

Harold P Edmundson. New methods in automatic extracting. *Journal of the ACM (JACM)*, 16(2):264–285, 1969.

Samira Ellouze, Maher Jaoua, and Lamia Hadrich Belguith. An evaluation summary method based on a combination of content and linguistic metrics. In *Recent Advances in Natural Language Processing (RANLP)*, pages 245–251, 2013.

Micha Elsner and Deepak Santhanam. Learning to fuse disparate sentences. In *Proceedings of the Workshop on Monolingual Text-To-Text Generation*, pages 54–63. ACL, 2011.

Günes Erkan and Dragomir R Radev. Lexrank: Graph-based lexical centrality as salience in text summarization. *Journal of Artificial Intelligence Research*, pages 457–479, 2004.

Yimai Fang and Simone Teufel. A summariser based on human memory limitations and lexical competition. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2014)*, pages 732–741. ACL, 2014.

Yimai Fang and Simone Teufel. Improving argument overlap for proposition-based summarisation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL 2016)*, volume 2, pages 479–485, 2016.

Yimai Fang, Haoyue Zhu, Ewa Muszyńska, Alexander Kuhnle, and Simone Teufel. A proposition-based abstractive summariser. In *Proceedings of the 26th International Conference on Computational Linguistics (COLING 2016)*, pages 567–578. ACL, 2016.

Mohamed Abdel Fattah and Fuji Ren. Ga, mr, ffnn, pnn and gmm based models for automatic text summarization. *Computer Speech & Language*, 23(1):126–144, 2009.

Christiane Fellbaum. *WordNet*. Wiley Online Library, 1998.

Lijun Feng, Martin Jansche, Matt Huenerfauth, and Noémie Elhadad. A comparison of features for automatic readability assessment. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters (COLING 2010)*, pages 276–284. ACL, 2010.

Katja Filippova. *Dependency Graph Based Sentence Fusion and Compression.* PhD thesis, Technische Universität, 2010a.

Katja Filippova. Multi-sentence compression: finding shortest paths in word graphs. In *Proceedings of the 23rd International Conference on Computational Linguistics (COLING 2010)*, pages 322–330. ACL, 2010b.

Katja Filippova and Michael Strube. Sentence fusion via dependency graph compression. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing (EMNLP 2008)*, pages 177–185. ACL, 2008.

Marcelo Fiszman, Thomas C Rindflesch, and Halil Kilicoglu. Abstraction summarization for managing the biomedical research literature. In *Proceedings of the HLT-NAACL Workshop on Computational Lexical Semantics (HLT-NAACL 2004)*, pages 76–83. ACL, 2004.

Daniel Fried and Kevin Duh. Incorporating both distributional and relational semantics in word representations. *arXiv preprint arXiv:1412.4369*, 2014.

Maria Fuentes, Enrique Alfonseca, and Horacio Rodríguez. Support vector machines for query-focused summarization trained and evaluated on pyramid data. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics: Interactive Poster and Demonstration Sessions (ACL 2007)*, pages 57–60. ACL, 2007.

Dimitrios Galanis, Gerasimos Lampouras, and Ion Androutsopoulos. Extractive multi-document summarization with integer linear programming and support vector regression. *Proceedings of the 24th International Conference on Computational Linguistics (COLING 2012)*, pages 911–926, 2012.

Michel Galley and Kathleen McKeown. Lexicalized markov grammars for sentence compression. In *Proceedings of the 2007 Annual Conference of the North Ameri-*

can Chapter of the Association for Computational Linguistics - Human Language Technologies (NAACL HLT 2007), pages 180–187. ACL, 2007.

Mahak Gambhir and Vishal Gupta. Recent automatic text summarization techniques: a survey. *Artificial Intelligence Review*, 47(1):1–66, 2017.

Kavita Ganesan, ChengXiang Zhai, and Jiawei Han. Opinosis: a graph-based approach to abstractive summarization of highly redundant opinions. In *Proceedings of the 23rd International Conference on Computational Linguistics (COLING 2010)*, pages 340–348. ACL, 2010.

Juri Ganitkevitch, Benjamin Van Durme, and Chris Callison-Burch. Ppdb: The paraphrase database. In *Proceedings of the 2013 Annual Conference of the North American Chapter of the Association for Computational Linguistics - Human Language Technologies (NAACL HLT 2013)*, pages 758–764. ACL, 2013.

Dehong Gao, Wenjie Li, and Renxian Zhang. Sequential summarization: A new application for timely updated twitter trending topics. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (ACL 2013)*, volume 2, pages 567–571. ACL, 2013.

Tao Ge, Wenzhe Pei, Heng Ji, Sujian Li, Baobao Chang, and Zhifang Sui. Bring you to the past: Automatic generation of topically relevant event chronicles. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (ACL-IJCNLP 2015)*, volume 1, pages 575–585. ACL, 2015.

Tao Ge, Lei Cui, Baobao Chang, Sujian Li, Ming Zhou, and Zhifang Sui. News stream summarization using burst information networks. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP 2016)*, pages 784–794. ACL, 2016.

Pierre-Etienne Genest and Guy Lapalme. Fully abstractive approach to guided summarization. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (ACL 2012)*, pages 354–358. ACL, 2012.

Shima Gerani, Yashar Mehdad, Giuseppe Carenini, Raymond T Ng, and Bita Nejat. Abstractive summarization of product reviews using discourse structure. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP 2014)*, pages 1602–1613. ACL, 2014.

George Giannakopoulos, Vangelis Karkaletsis, George Vouros, and Panagiotis Stamatopoulos. Summarization system evaluation revisited: N-gram graphs. *ACM Transactions on Speech and Language Processing (TSLP)*, 5(3):1–5, 2008.

Daniel Gillick, Benoit Favre, and Dilek Hakkani-Tür. The icsi summarization system at tac 2008. In *Proceedings of the 1st Text Analysis Conference (TAC 2008)*, 2008.

Yihong Gong and Xin Liu. Generic text summarization using relevance measure and latent semantic analysis. In *Proceedings of the 24th Annual International ACM SIGIR Conference on Research & Development in Information Retrieval (SIGIR 2001)*, pages 19–25. ACM, 2001.

Joshua T Goodman. A bit of progress in language modeling. *Computer Speech & Language*, 15(4):403–434, 2001.

Philip John Gorinski and Mirella Lapata. Movie script summarization as graph-based scene extraction. In *Proceedings of the 2015 Annual Conference of the North American Chapter of the Association for Computational Linguistics - Human Language Technologies (NAACL HLT 2015)*, pages 1066–1076. ACL, 2015.

Yvette Graham et al. Re-evaluating automatic summarization with bleu and 192 shades of rouge. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP 2015)*, pages 128–137. ACL, 2015.

Tom Griffiths. Gibbs sampling in the generative model of latent dirichlet allocation. 2002.

Evidence-Based Medicine Working Group et al. Evidence-based medicine. a new approach to teaching the practice of medicine. *Journal of the American Medical Association (JAMA)*, 268(17):2420, 1992.

Jiatao Gu, Zhengdong Lu, Hang Li, and Victor OK Li. Incorporating copying mechanism in sequence-to-sequence learning. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL 2016)*, volume 1, pages 1631–1640. ACL, 2016.

Caglar Gulcehre, Sungjin Ahn, Ramesh Nallapati, Bowen Zhou, and Yoshua Bengio. Pointing the unknown words. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL 2016)*, volume 1, pages 140–149. ACL, 2016.

Aria Haghighi and Lucy Vanderwende. Exploring content models for multi-document summarization. In *Proceedings of the 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics - Human Language Technologies (NAACL HLT 2009)*, pages 362–370. ACL, 2009.

Allan Hanbury. Medical information retrieval: an instance of domain-specific search. In *Proceedings of the 35th Annual International ACM SIGIR Conference on Research & Development in Information Retrieval (SIGIR 2012)*, pages 1191–1192. ACM, 2012.

Taher H Haveliwala. Topic-sensitive pagerank. In *Proceedings of the 11th International Conference on World Wide Web (WWW 2002)*, pages 517–526. ACM, 2002.

Zhanying He, Chun Chen, Jiajun Bu, Can Wang, Lijun Zhang, Deng Cai, and Xiaofei He. Document summarization based on data reconstruction. In *Proceedings of the 26th Conference on Artificial Intelligence (AAAI 2012)*, 2012.

Peter A Heeman. Pos tagging versus classes in language modeling. In *Proceedings of the 6th Workshop on Very Large Corpora (WVLC 1998)*. ACL, 1998.

William R Hersh, M Katherine Crabtree, David H Hickam, Lynetta Sacherek, Charles P Friedman, Patricia Tidmarsh, Craig Mosbaek, and Dale Kraemer. Factors associated with success in searching medline and applying evidence to answer clinical questions. *Journal of the American Medical Informatics Association (JAMIA)*, 9(3):283–293, 2002.

Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997.

Deirdre Hogan. Empirical measurements of lexical similarity in noun phrase conjuncts. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics: Interactive Poster and Demonstration Sessions (ACL 2007)*, pages 149–152. ACL, 2007.

Kai Hong and Ani Nenkova. Improving the estimation of word importance for news multi-document summarization. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2014)*, pages 712–721. ACL, 2014.

Kai Hong, Mitchell Marcus, and Ani Nenkova. System combination for multi-document summarization. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP 2015)*, pages 107–117. ACL, 2015.

Eduard Hovy, Chin-Yew Lin, Liang Zhou, and Junichi Fukumoto. Automated summarization evaluation with basic elements. In *Proceedings of the 5th Conference on Language Resources and Evaluation (LREC 2006)*, pages 604–611, 2006.

Baotian Hu, Qingcai Chen, and Fangze Zhu. Lcsts: A large scale chinese short text summarization dataset. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP 2015)*, pages 1967–1972. ACL, 2015.

Po Hu, Donghong Ji, Chong Teng, and Yujing Guo. Context-enhanced personalized social summarization. *Proceedings of the 24th International Conference on Computational Linguistics (COLING 2012)*, pages 1223–1238, 2012.

Yue Hu and Xiaojun Wan. Ppsgen: Learning-based presentation slides generation for academic papers. *IEEE transactions on knowledge and data engineering*, 27 (4):1085–1097, 2015.

Kuo-Chuan Huang, James Geller, Michael Halper, Yehoshua Perl, and Junchuan Xu. Using wordnet synonym substitution to enhance umls source integration. *Artificial Intelligence in Medicine*, 46(2):97–109, 2009.

Lei Huang, Yanxiang He, Furu Wei, and Wenjie Li. Modeling document summarization as multi-objective optimization. In *Proceedings of the 3rd International Symposium on Intelligent Information Technology and Security Informatics (IITSI 2010)*, pages 382–386. IEEE, 2010.

Srinivasan Iyer, Ioannis Konstas, Alvin Cheung, and Luke Zettlemoyer. Summarizing source code using a neural attention model. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL 2016)*, volume 1, pages 2073–2083. ACL, 2016.

Ray Jackendoff. *The architecture of the language faculty.* Number 28. MIT Press, 1997.

Jayanth Jayanth, Jayaprakash Sundararaj, and Pushpak Bhattacharyya. Monotone submodularity in opinion summaries. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP 2015)*, pages 169–178. ACL, 2015.

Heng Ji, Benoit Favre, Wen-Pin Lin, Dan Gillick, Dilek Hakkani-Tur, and Ralph Grishman. Open-domain multi-document summarization via information extraction: Challenges and prospects. In *Multi-source, multilingual information extraction and summarization*, pages 177–201. Springer, 2013.

Yangfeng Ji, Gholamreza Haffari, and Jacob Eisenstein. A latent variable recurrent neural network for discourse relation language models. In *Proceedings of the 2016 Annual Conference of the North American Chapter of the Association for Computational Linguistics - Human Language Technologies (NAACL HLT 2016)*, pages 332–342. ACL, 2016.

Hongyan Jing. Sentence reduction for automatic text summarization. In *Proceedings of the 6th Conference on Applied Natural Language Processing (ANLP 2000)*, pages 310–315. ACL, 2000.

Karen Sparck Jones. Automatic summarizing: factors and directions. *Advances in Automatic Text Summarization*, pages 1–12, 1999.

Karen Sparck Jones. Automatic summarising: a review and discussion of the state of the art. Technical report, University of Cambridge, Computer Laboratory, 2007.

Karen Sparck Jones and Julia R Galliers. *Evaluating natural language processing systems: An analysis and review*, volume 1083. Springer Science & Business Media, 1995.

Karen Sparck Jones and Julia R Galliers. Evaluating natural language processing systems: An analysis and review. 1996.

Rafal Jozefowicz, Oriol Vinyals, Mike Schuster, Noam Shazeer, and Yonghui Wu. Exploring the limits of language modeling. *arXiv preprint arXiv:1602.02410*, 2016.

Joel Judd and Jugal Kalita. Better twitter summaries? In *Proceedings of the 2013 Annual Conference of the North American Chapter of the Association for Computational Linguistics - Human Language Technologies (NAACL HLT 2013)*, pages 445–449. ACL, 2013.

Mikael Kågebäck, Olof Mogren, Nina Tahmasebi, and Devdatt Dubhashi. Extractive summarization using continuous vector space models. In *Proceedings of the 2nd Workshop on Continuous Vector Space Models and their Compositionality (CVSC 2014)*, pages 31–39, 2014.

Chris Kedzie, Fernando Diaz, and Kathleen McKeown. Real-time web scale event summarization using sequential decision making. *arXiv preprint arXiv:1605.03664*, 2016.

Yuta Kikuchi, Tsutomu Hirao, Hiroya Takamura, Manabu Okumura, and Masaaki Nagata. Single document summarization based on nested tree structure. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (ACL 2014)*, volume 2, pages 315–320. ACL, 2014.

Yuta Kikuchi, Graham Neubig, Ryohei Sasano, Hiroya Takamura, and Manabu Okumura. Controlling output length in neural encoder-decoders. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP 2016)*, pages 1328–1338. ACL, 2016.

Walter Kintsch and Teun A Van Dijk. Toward a model of text comprehension and production. *Psychological Review*, 85(5):363, 1978.

Hayato Kobayashi, Masaki Noguchi, and Taichi Yatsuka. Summarization based on embedding distributions. In *Proceedings of the 2015 Conference on Empirical*

Methods in Natural Language Processing (EMNLP 2015), pages 1984–1989. ACL, 2015.

Philipp Koehn, Abhishek Arun, and Hieu Hoang. Towards better machine translation quality for the german–english language pairs. In *Proceedings of the 3rd Workshop on Statistical Machine Translation (WMT 2008)*, pages 139–142. ACL, 2008.

Matt Kusner, Yu Sun, Nicholas Kolkin, and Kilian Weinberger. From word embeddings to document distances. In *Proceedings of the 32nd International Conference on Machine Learning (ICML 2015)*, pages 957–966. ACM, 2015.

Quoc Le and Tomas Mikolov. Distributed representations of sentences and documents. In *Proceedings of the 31st International Conference on Machine Learning (ICML 2014)*, pages 1188–1196, 2014.

Tao Lei, Regina Barzilay, and Tommi Jaakkola. Rationalizing neural predictions. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP 2016)*, pages 107–117. ACL, 2016.

Chen Li, Fei Liu, Fuliang Weng, and Yang Liu. Document summarization via guided sentence compression. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing (EMNLP 2013)*, pages 490–500. ACL, 2013.

Chen Li, Yang Liu, Fei Liu, Lin Zhao, and Fuliang Weng. Improving multi-documents summarization by sentence compression based on expanded constituent parse trees. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP 2014)*, pages 691–701. ACL, 2014.

Jing Li, Wei Gao, Zhongyu Wei, Baolin Peng, and Kam-Fai Wong. Using content-level structures for summarizing microblog repost trees. In *Proceedings of the*

2015 Conference on Empirical Methods in Natural Language Processing (EMNLP 2015), pages 2168–2178. ACL, 2015a.

Jiwei Li and Claire Cardie. Timeline generation: Tracking individuals on twitter. In *Proceedings of the 23rd International Conference on World Wide Web (WWW 2014)*, pages 643–652. ACM, 2014.

Jiwei Li, Thang Luong, and Dan Jurafsky. A hierarchical neural autoencoder for paragraphs and documents. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (ACL-IJCNLP 2015)*, volume 1, pages 1106–1115. ACL, 2015b.

Liangda Li, Ke Zhou, Gui-Rong Xue, Hongyuan Zha, and Yong Yu. Enhancing diversity, coverage and balance for summarization through structure learning. In *Proceedings of the 18th International Conference on World Wide Web (WWW 2009)*, pages 71–80. ACM, 2009.

Piji Li, Lidong Bing, Wai Lam, Hang Li, and Yi Liao. Reader-aware multi-document summarization via sparse coding. In *Proceedings of the 28th International Joint Conference on Artifical Intelligence (IJCAI 2015)*, pages 1270–1276, 2015c.

Yanran Li and Sujian Li. Query-focused multi-document summarization: Combining a topic model with graph-based semi-supervised learning. In *Proceedings of the 25th International Conference on Computational Linguistics (COLING 2014)*, pages 1197–1207. ACL, 2014.

Chin-Yew Lin. Improving summarization performance by sentence compression: a pilot study. In *Proceedings of the 6th International Workshop on Information Retrieval with Asian Languages*, pages 1–8. ACL, 2003.

Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. volume 8. ACL, 2004.

Chin-Yew Lin and Eduard Hovy. The automated acquisition of topic signatures for text summarization. In *Proceedings of the 18th Conference on Computational Linguistics (COLING 2000)*, pages 495–501. ACL, 2000.

Chin-Yew Lin and FJ Och. Looking for a few good metrics: Rouge and its evaluation. In *Proceedings of the 4th NII Testbeds and Community for Information Access Research: Workshops (NTCIR 2004)*, 2004.

Hui Lin and Jeff Bilmes. A class of submodular functions for document summarization. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL-HLT 2011)*, pages 510–520. ACL, 2011.

Marina Litvak and Mark Last. Cross-lingual training of summarization systems using annotated corpora in a foreign language. *Information Retrieval*, 16(5):629–656, 2013.

Fei Liu, Jeffrey Flanigan, Sam Thomson, Norman Sadeh, and Noah A Smith. Toward abstractive summarization using semantic representations. In *Proceedings of the 2015 Annual Conference of the North American Chapter of the Association for Computational Linguistics - Human Language Technologies (NAACL HLT 2015)*, pages 1077–1086. ACL, 2015a.

He Liu, Hongliang Yu, and Zhi-Hong Deng. Multi-document summarization based on two-level sparse representation model. In *Proceedings of the 29th Conference on Artificial Intelligence (AAAI 2015)*, pages 196–202, 2015b.

Xiaohua Liu, Yitong Li, Furu Wei, and Ming Zhou. Graph-based multi-tweet summarization using social signals. *Proceedings of the 24th International Conference on Computational Linguistics (COLING 2012)*, pages 1699–1714, 2012a.

Yan Liu, Sheng-hua Zhong, and Wenjie Li. Query-oriented multi-document summarization via unsupervised deep learning. In *Proceedings of the 26th Conference on Artificial Intelligence (AAAI 2012)*, pages 1699–1705, 2012b.

Elena Lloret and Manuel Palomar. Towards automatic tweet generation: A comparative study from the text summarization perspective in the journalism genre. *Expert Systems with Applications*, 40(16):6624–6630, 2013.

Elena Lloret, Laura Plaza, and Ahmet Aker. The challenging task of summary evaluation: an overview. *Language Resources and Evaluation*, 52(1):101–148, 2018.

Annie Louis and Ani Nenkova. Automatically evaluating content selection in summarization without human models. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing (EMNLP 2009)*, pages 306–314. ACL, 2009a.

Annie Louis and Ani Nenkova. Predicting summary quality using limited human input. In *Proceedings of the 2nd Text Analysis Conference (TAC 2009)*, 2009b.

Annie Louis and Ani Nenkova. Automatically assessing machine summary content without a gold standard. *Computational Linguistics*, 39(2):267–300, 2013.

Vanessa Loza, Shibamouli Lahiri, Rada Mihalcea, and Po-Hsiang Lai. Building a dataset for summarization and keyword extraction from emails. In *Proceedings of the 9th Conference on Language Resources and Evaluation (LREC 2014)*, pages 2441–2446, 2014.

Beier Lu. *Health Query Expansion Using WordNet and UMLS*. PhD thesis, Simon Fraser University, 2015.

Hans Peter Luhn. The automatic creation of literature abstracts. *IBM Journal of Research and Development*, 2(2):159–165, 1958.

Wencan Luo and Diane Litman. Summarizing student responses to reflection prompts. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP 2015)*, pages 1955–1960. ACL, 2015.

Shulei Ma, Zhi-Hong Deng, and Yunlun Yang. An unsupervised multi-document summarization framework based on neural document model. In *Proceedings of the 26th International Conference on Computational Linguistics (COLING 2016)*, pages 1514–1523. ACL, 2016.

Debanjan Mahata, John Kuriakose, Rajiv Ratn Shah, and Roger Zimmermann. Key2vec: Automatic ranked keyphrase extraction from scientific articles using phrase embeddings. In *Proceedings of the 2018 Annual Conference of the North American Chapter of the Association for Computational Linguistics - Human Language Technologies (NAACL HLT 2018)*, volume 2, pages 634–639. ACL, 2018.

I Mani and MT Maybury. Advances in automatic text summarization, 1999.

Inderjeet Mani. *Automatic summarization*, volume 3. John Benjamins Publishing, 2001.

Iain J Marshall, Joël Kuiper, and Byron C Wallace. Automating risk of bias assessment for clinical trials. *IEEE Journal of Biomedical and Health Informatics*, 19 (4):1406–1412, 2015.

David Martinez, Arantxa Otegi, Aitor Soroa, and Eneko Agirre. Improving search over electronic health records using umls-based query expansion through random walks. *Journal of Biomedical Informatics*, 51:100–106, 2014.

Ryan McDonald. Discriminative sentence compression with soft syntactic evidence. In *Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2006)*. ACL, 2006.

Ryan McDonald. A study of global inference algorithms in multi-document summarization. In *Proceedings of the 29th European Conference on Information Retrieval (ECIR 2007)*, pages 557–564. Springer, 2007.

Donald Metzler and Tapas Kanungo. Machine learned sentence selection strategies for query-biased summarization. In *Proceedings of the 2008 SIGIR Learning to Rank Workshop (LR4IR 2008)*, pages 40–47. ACM, 2008.

Rada Mihalcea and Paul Tarau. Textrank: Bringing order into text. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing (EMNLP 2004)*. ACL, 2004.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Proceedings of the 27th Annual Conference on Neural Information Processing Systems (NIPS 2013)*, pages 3111–3119, 2013a.

Tomas Mikolov, Wen-tau Yih, and Geoffrey Zweig. Linguistic regularities in continuous space word representations. In *Proceedings of the 2013 Annual Conference of the North American Chapter of the Association for Computational Linguistics - Human Language Technologies (NAACL HLT 2013)*, volume 13, pages 746–751. ACL, 2013b.

George A Miller. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41, 1995.

Muhidin A Mohamed and Mourad Oussalah. Similarity-based query-focused multi-document summarization using crowdsourced and manually-built lexical-semantic resources. In *Proceedings of the 2015 IEEE Trustcom/BigDataSE/ISPA*, volume 2, pages 80–87. IEEE, 2015.

Diego Mollá, María Elena Santiago-Martínez, Abeed Sarker, and Cécile Paris. A

corpus for research in text processing for evidence based medicine. *Language Resources and Evaluation*, pages 1–23, 2015.

Christof Monz. Statistical machine translation with local language models. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing (EMNLP 2011)*, pages 869–879. ACL, 2011.

Hajime Morita, Ryohei Sasano, Hiroya Takamura, and Manabu Okumura. Subtree extractive summarization via submodular maximization. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (ACL 2013)*, volume 1, pages 1023–1032. ACL, 2013.

Fleur Mougin, Anita Burgun, and Olivier Bodenreider. Using wordnet to improve the mapping of data elements to umls for data sources integration. In *Proceedings of the American Medical Informatics Association Annual Symposium (AMIA 2006)*, volume 2006, page 574. American Medical Informatics Association, 2006.

Ramesh Nallapati, Bowen Zhou, Cicero dos Santos, Ça glar Gulçehre, and Bing Xiang. Abstractive text summarization using sequence-to-sequence rnns and beyond. *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning (CoNLL 2016)*, pages 280–290, 2016.

Roberto Navigli and Simone Paolo Ponzetto. Babelnet: Building a very large multilingual semantic network. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics (ACL 2010)*, pages 216–225. ACL, 2010.

Ani Nenkova and Kathleen McKeown. A survey of text summarization techniques. In *Mining Text Data*, pages 43–76. Springer, 2012.

Ani Nenkova, Kathleen McKeown, et al. Automatic summarization. *Foundations and Trends® in Information Retrieval*, 5(2–3):103–233, 2011.

Jun-Ping Ng and Viktoria Abrecht. Better summarization evaluation with word embeddings for rouge. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP 2015)*, pages 1925–1930. ACL, 2015.

Jun-Ping Ng, Yan Chen, Min-Yen Kan, and Zhoujun Li. Exploiting timelines to enhance multi-document summarization. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (ACL 2014)*, volume 1, pages 923–933. ACL, 2014.

Jeffrey Nichols, Jalal Mahmud, and Clemens Drews. Summarizing sporting events using twitter. In *Proceedings of the 2012 ACM International Conference on Intelligent User Interfaces (ACM IUI 2012)*, pages 189–198. ACM, 2012.

Masaaki Nishino, Norihito Yasuda, Tsutomu Hirao, Shin-ichi Minato, and Masaaki Nagata. A dynamic programming algorithm for tree trimming-based text summarization. In *Proceedings of the 2015 Annual Conference of the North American Chapter of the Association for Computational Linguistics - Human Language Technologies (NAACL HLT 2015)*, pages 462–471. ACL, 2015.

Andrei Olariu. Efficient online summarization of microblogging streams. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2014)*, pages 236–240. ACL, 2014.

Jahna Otterbacher, Güneş Erkan, and Dragomir R Radev. Using random walks for question-focused sentence retrieval. In *Proceedings of the 2005 Conference on Human Language Technology and Empirical Methods in Natural Language Processing (HLT-EMNLP 2005)*, pages 915–922. ACL, 2005.

You Ouyang, Wenjie Li, Sujian Li, and Qin Lu. Applying regression models to query-focused multi-document summarization. *Information Processing & Management*, 47(2):227–237, 2011.

Paul Over and J Yen. An introduction to duc 2004 intrinsic evaluation of generic new text summarization systems, 2004. *National Institute of Standards and Technology (NIST)*, 2004.

Karolina Owczarzak and Hoa Trang Dang. Overview of the tac 2010 summarization track. In *Proceedings of the 3rd Text Analysis Conference (TAC 2010)*, 2010.

Karolina Owczarzak and Hoa Trang Dang. Overview of the tac 2011 summarization track: Guided task and aesop task. In *Proceedings of the 4th Text Analysis Conference (TAC 2011)*, 2011.

Tatsuro Oya, Yashar Mehdad, Giuseppe Carenini, and Raymond Ng. A template-based abstractive meeting summarization: Leveraging summary and source text relationships. In *Proceedings of the 8th International Natural Language Generation Conference (INLG 2014)*, pages 45–53, 2014.

Bo Pang, Lillian Lee, et al. Opinion mining and sentiment analysis. *Foundations and Trends® in Information Retrieval*, 2(1–2):1–135, 2008.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics (ACL 2002)*, pages 311–318. ACL, 2002.

Daraksha Parveen and Michael Strube. Integrating importance, non-redundancy and coherence in graph-based extractive summarization. In *Proceedings of the 28th International Joint Conference on Artifical Intelligence (IJCAI 2015)*, pages 1298–1304, 2015.

Daraksha Parveen, Hans-Martin Ramsl, and Michael Strube. Topical coherence for graph-based extractive summarization. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP 2015)*, pages 1949–1954. ACL, 2015.

Rebecca J Passonneau, Ani Nenkova, Kathleen McKeown, and Sergey Sigelman. Applying the pyramid method in duc 2005. In *Proceedings of the 2005 Document Understanding Conference (DUC 2005)*, 2005.

Rebecca J Passonneau, Emily Chen, Weiwei Guo, and Dolores Perin. Automated pyramid scoring of summaries using distributional semantics. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics: Short Papers (ACL 2013)*. ACL, 2013.

Romain Paulus, Caiming Xiong, and Richard Socher. A deep reinforced model for abstractive summarization. *arXiv preprint arXiv:1705.04304*, 2017.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in python. *Journal of Machine Learning Research (JMLR)*, 12(Oct):2825–2830, 2011.

Yulong Pei, Wenpeng Yin, Qifeng Fan, et al. A supervised aggregation framework for multi-document summarization. *Proceedings of the 24th International Conference on Computational Linguistics (COLING 2012)*, pages 2225–2242, 2012.

Maxime Peyrard and Judith Eckle-Kohler. Optimizing an approximation of rouge-a problem-reduction approach to extractive multi-document summarization. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL 2016)*, volume 1, pages 1825–1836, 2016.

Daniele Pighin, Marco Cornolti, Enrique Alfonseca, and Katja Filippova. Modelling events through memory-based, open-ie patterns for abstractive summarization. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (ACL 2014)*, volume 1, pages 892–901. ACL, 2014.

Mohammad Taher Pilehvar and Roberto Navigli. From senses to texts: An all-in-one

graph-based approach for measuring semantic similarity. *Artificial Intelligence*, 228:95–128, 2015.

Mohammad Taher Pilehvar, David Jurgens, and Roberto Navigli. Align, disambiguate and walk: A unified approach for measuring semantic similarity. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (ACL 2013)*, pages 1341–1351. ACL, 2013.

Laura Plaza, Alberto Díaz, and Pablo Gervás. A semantic graph-based approach to biomedical summarisation. *Artificial Intelligence in Medicine*, 53(1):1–14, 2011.

Maja Popović. Morpheme-and pos-based ibm1 scores and language model scores for translation quality estimation. In *Proceedings of the 7th Workshop on Statistical Machine Translation (MT 2012)*, pages 133–137. ACL, 2012.

Xian Qian and Yang Liu. Fast joint compression and summarization via graph cuts. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing (EMNLP 2013)*, pages 1492–1502. ACL, 2013.

Dragomir R Radev, Eduard Hovy, and Kathleen McKeown. Introduction to the special issue on summarization. *Computational Linguistics*, 28(4):399–408, 2002.

Dragomir R Radev, Simone Teufel, Horacio Saggion, Wai Lam, John Blitzer, Hong Qi, Arda Celebi, Danyu Liu, and Elliott Drabek. Evaluation challenges in large-scale document summarization. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics (ACL 2003)*, pages 375–382. ACL, 2003.

Dragomir R Radev, Hongyan Jing, Małgorzata Styś, and Daniel Tam. Centroid-based summarization of multiple documents. *Information Processing & Management*, 40(6):919–938, 2004.

Peter Rankel, John M Conroy, Eric V Slud, and Dianne P O'Leary. Ranking human and machine summarization systems. In *Proceedings of the 2011 Conference on*

Empirical Methods in Natural Language Processing (EMNLP 2011), pages 467–473. ACL, 2011.

Marc'Aurelio Ranzato, Sumit Chopra, Michael Auli, and Wojciech Zaremba. Sequence level training with recurrent neural networks. *arXiv preprint arXiv:1511.06732*, 2015.

Lawrence H Reeve, Hyoil Han, and Ari D Brooks. The use of domain-specific concepts in biomedical text summarization. *Information Processing & Management*, 43(6):1765–1776, 2007.

Pengjie Ren, Furu Wei, CHEN Zhumin, MA Jun, and Ming Zhou. A redundancy-aware sentence regression framework for extractive summarization. In *Proceedings of the 26th International Conference on Computational Linguistics (COLING 2016)*, pages 33–43. ACL, 2016.

Korbinian Riedhammer, Benoit Favre, and Dilek Hakkani-Tür. Long story short–global unsupervised models for keyphrase based meeting summarization. *Speech Communication*, 52(10):801–815, 2010.

Cody Rioux, Sadid A Hasan, and Yllias Chali. Fear the reaper: A system for automatic multi-document summarization with reinforcement learning. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP 2014)*, pages 681–690. ACL, 2014.

Alex Rodriguez and Alessandro Laio. Clustering by fast search and find of density peaks. *Science*, 344(6191):1492–1496, 2014.

Gaetano Rossiello. Neural abstractive text summarization. In *Proceedings of the Doctoral Consortium of AI*IA 2016 co-located with the 15th International Conference of the Italian Association for Artificial Intelligence (AI*IA 2016)*, pages 70–75, 2016.

Sascha Rothe and Hinrich Schütze. Cosimrank: A flexible & efficient graph-theoretic similarity measure. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (ACL 2014)*, pages 1392–1402, 2014.

Sascha Rothe and Hinrich Schütze. Autoextend: Extending word embeddings to embeddings for synsets and lexemes. *arXiv preprint arXiv:1507.01127*, 2015.

Alexander M Rush, Sumit Chopra, and Jason Weston. A neural attention model for abstractive sentence summarization. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP 2015)*, pages 379–389. ACL, 2015.

David L Sackett, William MC Rosenberg, JA Muir Gray, R Brian Haynes, and W Scott Richardson. Evidence based medicine: what it is and what it isn't. *British Medical Journal (BMJ)*, 312(7023):71–72, 1996.

Ivan A Sag, Timothy Baldwin, Francis Bond, Ann Copestake, and Dan Flickinger. Multiword expressions: A pain in the neck for nlp. In *Proceedings of the 3rd International Conference on Computational Linguistics and Intelligent Text Processing (CICLing 2002)*, pages 1–15. Springer, 2002.

Horacio Saggion. Unsupervised learning summarization templates from concise summaries. In *Proceedings of the 2013 Annual Conference of the North American Chapter of the Association for Computational Linguistics - Human Language Technologies (NAACL HLT 2013)*, pages 270–279. ACL, 2013.

Horacio Saggion, Juan-Manuel Torres-Moreno, Iria da Cunha, and Eric SanJuan. Multilingual summarization evaluation without human models. In *Proceedings of the 23rd International Conference on Computational Linguistics (COLING 2010)*, pages 1059–1067. ACL, 2010.

Kamal Sarkar. Syntactic trimming of extracted sentences for improving extractive multi-document summarization. *Journal of Computing*, 2(7):177–184, 2010.

Abeed Sarker, Diego Mollá, and Cécile Paris. An approach for query-focused text summarisation for evidence based medicine. In *Proceedings of the 14th Conference on Artificial Intelligence in Medicine in Europe (AIME 2013)*, pages 295–304. Springer, 2013.

Abeed Sarker, Diego Mollá, and Cécile Paris. Automatic evidence quality prediction to support evidence-based decision making. *Artificial Intelligence in Medicine*, 64 (2):89–103, 2015.

Abeed Sarker, Diego Mollá, and Cecile Paris. Query-oriented evidence extraction to support evidence-based medicine practice. *Journal of Biomedical Informatics*, 59:169–184, 2016.

Frank Schilder and Ravikumar Kondadadi. Fastsum: fast and accurate query-based multi-document summarization. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics on Human Language Technologies: Short Papers (ACL:HLT 2008)*, pages 205–208. ACL, 2008.

Ariel S Schwartz and Marti A Hearst. A simple algorithm for identifying abbreviation definitions in biomedical text. In *Biocomputing*, pages 451–462. World Scientific, 2002.

Abigail See, Peter J Liu, and Christopher D Manning. Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL 2017)*, volume 1, pages 1073–1083. ACL, 2017.

Elaheh ShafieiBavani, Mohammad Ebrahimi, Raymond Wong, and Fang Chen. Appraising umls coverage for summarizing medical evidence. In *Proceedings of the 26th International Conference on Computational Linguistics (COLING 2016)*, pages 513–524. ACL, 2016a.

Elaheh ShafieiBavani, Mohammad Ebrahimi, Raymond Wong, and Fang Chen. An efficient approach for multi-sentence compression. In *Proceedings of Machine Learning Research (PMLR): Asian Conference on Machine Learning (ACML 2016)*, volume 63, pages 414–429, 2016b.

Elaheh ShafieiBavani, Mohammad Ebrahimi, Raymond Wong, and Fang Chen. On improving informativity and grammaticality for multi-sentence compression. *arXiv preprint arXiv:1605.02150*, 2016c.

Elaheh ShafieiBavani, Mohammad Ebrahimi, Raymond Wong, and Fang Chen. A query-based summarization service from multiple news sources. In *Proceedings of the 2016 IEEE International Conference on Services Computing (SCC 2016)*, pages 42–49. IEEE, 2016d.

Elaheh ShafieiBavani, Mohammad Ebrahimi, Raymond Wong, and Fang Chen. A semantically motivated approach to compute ROUGE scores. *arXiv preprint arXiv:1710.07441*, 2017.

Elaheh ShafieiBavani, Mohammad Ebrahimi, Raymond Wong, and Fang Chen. A graph-theoretic summary evaluation for rouge. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP 2018)*, pages 762–767. ACL, 2018a.

Elaheh ShafieiBavani, Mohammad Ebrahimi, Raymond Wong, and Fang Chen. Summarization evaluation in the absence of human model summaries using the compositionality of word embeddings. In *Proceedings of the 27th International Conference on Computational Linguistics (COLING 2018)*, pages 905–914. ACL, 2018b.

Beaux Sharifi, Mark-Anthony Hutton, and Jugal Kalita. Summarizing microblogs automatically. In *Proceedings of the 2010 Annual Conference of the North Ameri-*

can Chapter of the Association for Computational Linguistics - Human Language Technologies (NAACL HLT 2010), pages 685–688. ACL, 2010a.

Beaux Sharifi, Mark-Anthony Hutton, and Jugal K Kalita. Experiments in microblog summarization. In *Proceedings of the 2nd IEEE International Conference on Social Computing (SocialCom 2010)*, pages 49–56. IEEE, 2010b.

Chao Shen and Tao Li. Learning to rank for query-focused multi-document summarization. In *Proceedings of the 11th IEEE International Conference on Data Mining (ICDM 2011)*, pages 626–634. IEEE, 2011.

Dou Shen, Jian-Tao Sun, Hua Li, Qiang Yang, and Zheng Chen. Document summarization using conditional random fields. In *Proceedings of the 16th International Joint Conferences on Artificial Intelligence (IJCAI 2007)*, volume 7, pages 2862–2867, 2007.

Zhongmin Shi, Gabor Melli, Yang Wang, Yudong Liu, Baohua Gu, Mehdi M Kashani, Anoop Sarkar, and Fred Popowich. Question answering summarization of multiple biomedical documents. In *Advances in Artificial Intelligence*, pages 284–295. Springer, 2007.

Priya Sidhaye and Jackie Chi Kit Cheung. Indicative tweet generation: An extractive summarization problem? In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP 2015)*, pages 138–147. ACL, 2015.

Abhishek Singh and Wei Jin. Ranking summaries for informativeness and coherence without reference summaries. In *Proceedings of the 29th International Florida Artificial Intelligence Research Society Conference (FLAIRS 2016)*, pages 104–109, 2016.

Ko Skorokhod et al. Adaptive method of automatic abstracting and indexing. *Information Processing 71*, pages 1179–1182, 1972.

Daniel DK Sleator and Davy Temperley. Parsing english with a link grammar. 1995.

Wei Song, Lim Cheon Choi, Soon Cheol Park, and Xiao Feng Ding. Fuzzy evolutionary optimization modeling and its applications to unsupervised categorization and extractive summarization. *Expert Systems with Applications*, 38(8):9112–9121, 2011.

Josef Steinberger and Karel Ježek. Evaluation measures for text summarization. *Computing and Informatics*, 28(2):251–275, 2012.

Andreas Stolcke. Srilm-an extensible language modeling toolkit. In *Proceedings of the 7th International Conference on Spoken Language Processing (ICSLP 2002)*, 2002.

Sainbayar Sukhbaatar, Jason Weston, Rob Fergus, et al. End-to-end memory networks. In *Proceedings of the 29th Annual Conference on Neural Information Processing Systems (NIPS 2015)*, pages 2440–2448, 2015.

Rodney L Summerscales. *Automatic summarization of clinical abstracts for evidence-based medicine*. PhD thesis, Illinois Institute of Technology, 2013.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. In *Proceedings of the 28th Annual Conference on Neural Information Processing Systems (NIPS 2014)*, pages 3104–3112, 2014.

Hiroya Takamura, Hikaru Yokono, and Manabu Okumura. Summarizing a document stream. In *Proceedings of the 33rd European Conference on Information Retrieval (ECIR 2011)*, pages 177–188. Springer, 2011.

Yuhui Tao, Shuigeng Zhou, Wai Lam, and Jihong Guan. Towards more effective text summarization based on textual association networks. In *Proceedings of the 4th International Conference on Semantics, Knowledge and Grid (SKG 2008)*, pages 235–240. IEEE, 2008.

Kapil Thadani and Kathleen McKeown. Supervised sentence fusion with single-stage inference. In *Proceedings of the 6th International Joint Conference on Natural Language Processing (IJCNLP 2013)*, pages 1410–1418. ACL, 2013.

Kristina Toutanova, Dan Klein, Christopher D Manning, and Yoram Singer. Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of the 2003 Annual Conference of the North American Chapter of the Association for Computational Linguistics - Human Language Technologies (NAACL HLT 2003)*, pages 173–180. ACL, 2003.

Kristina Toutanova, Chris Brockett, Ke M Tran, and Saleema Amershi. A dataset and evaluation metrics for abstractive compression of sentences and short paragraphs. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP 2016)*, pages 340–350. ACL, 2016.

Giang Tran, Eelco Herder, and Katja Markert. Joint graphical models for date selection in timeline summarization. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (ACL-IJCNLP)*, volume 1, pages 1598–1607. ACL, 2015.

Stephen Tratz and Eduard Hovy. Bewte: basic elements with transformations for evaluation. In *Proceedings of the 1st Text Analysis Conference (TAC 2008)*, 2008.

Jérémy Trione, Benoit Favre, and Frédéric Béchet. Beyond utterance extraction: Summary recombination for speech summarization. In *Interspeech*, pages 680–684, 2016.

Joseph Turian, Lev Ratinov, and Yoshua Bengio. Word representations: a simple and general method for semi-supervised learning. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics (ACL 2010)*, pages 384–394. ACL, 2010.

Emmanouil Tzouridis, Jamal Abdul Nasir, LUMS Lahore, and Ulf Brefeld. Learning to summarise related sentences. In *Proceedings of the 25th International Conference on Computational Linguistics (COLING 2014)*. ACL, 2014.

Stijn Van Dongen. A cluster algorithm for graphs. *Information Systems*, (10):1–40, 2000.

Lucy Vanderwende, Hisami Suzuki, Chris Brockett, and Ani Nenkova. Beyond sumbasic: Task-focused summarization with sentence simplification and lexical expansion. *Information Processing & Management*, 43(6):1606–1618, 2007.

Vladimir Naumovich Vapnik. An overview of statistical learning theory. *IEEE Transactions on Neural Networks*, 10(5):988–999, 1999.

Patrick Verga and Andrew McCallum. Row-less universal schema. *arXiv preprint arXiv:1604.06361*, 2016.

Xiaojun Wan. Using only cross-document relationships for both generic and topic-focused multi-document summarizations. *Information Retrieval*, 11(1):25–49, 2008.

Xiaojun Wan and Jianmin Zhang. Ctsum: extracting more certain summaries for news articles. In *Proceedings of the 37th Annual International ACM SIGIR Conference on Research & Development in Information Retrieval (SIGIR 2014)*, pages 787–796. ACM, 2014.

Xiaojun Wan, Jianwu Yang, and Jianguo Xiao. Manifold-ranking based topic-focused multi-document summarization. In *Proceedings of the 20th International Joint Conference on Artifical Intelligence (IJCAI 2007)*, volume 7, pages 2903–2908, 2007.

Chan Wang, Lixia Long, and Lei Li. Hownet based evaluation for chinese text summarization. In *Proceedings of the 4th International Conference on Natural Lan-*

guage Processing and Knowledge Engineering (NLP-KE 2008), pages 1–6. IEEE, 2008a.

Dingding Wang and Tao Li. Weighted consensus multi-document summarization. *Information Processing & Management*, 48(3):513–523, 2012.

Dingding Wang, Tao Li, Shenghuo Zhu, and Chris Ding. Multi-document summarization via sentence-level semantic analysis and symmetric matrix factorization. In *Proceedings of the 31th Annual International ACM SIGIR Conference on Research & Development in Information Retrieval (SIGIR 2008)*, pages 307–314. ACM, 2008b.

Dingding Wang, Shenghuo Zhu, Tao Li, and Yihong Gong. Multi-document summarization using sentence-based topic models. In *Proceedings of the 47th Annual Meeting of the Association for Computational Linguistics and the 4th International Joint Conference on Natural Language Processing (ACL-IJCNLP 2009)*, pages 297–300. ACL, 2009.

Dingding Wang, Shenghuo Zhu, Tao Li, Yun Chi, and Yihong Gong. Integrating document clustering and multidocument summarization. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 5(3):14, 2011.

Lu Wang and Claire Cardie. Domain-independent abstract generation for focused meeting summarization. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (ACL 2013)*, volume 1, pages 1395–1405. ACL, 2013.

Lu Wang and Wang Ling. Neural network-based abstract generation for opinions and arguments. In *Proceedings of the 2016 Annual Conference of the North American Chapter of the Association for Computational Linguistics - Human Language Technologies (NAACL HLT 2016)*, pages 47–57. ACL, 2016.

Lu Wang, Hema Raghavan, Vittorio Castelli, Radu Florian, and Claire Cardie. A sentence compression based framework to query-focused multi-document summarization. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (ACL 2013*, pages 1384–1394. ACL, 2013.

Lu Wang, Hema Raghavan, Claire Cardie, and Vittorio Castelli. Query-focused opinion summarization for user-generated content. In *Proceedings of the 25th International Conference on Computational Linguistics (COLING 2014)*, pages 1660–1669. ACL, 2014.

Lu Wang, Hema Raghavan, Vittorio Castelli, Radu Florian, and Claire Cardie. A sentence compression based framework to query-focused multi-document summarization. *arXiv preprint arXiv:1606.07548*, 2016.

Kristian Woodsend and Mirella Lapata. Multiple aspect summarization using integer linear programming. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL 2012)*, pages 233–243. ACL, 2012.

Wenting Xiong and Diane Litman. Empirical analysis of exploiting review helpfulness for extractive summarization of online reviews. In *Proceedings of the 25th International Conference on Computational Linguistics (COLING 2014)*, pages 1985–1995. ACL, 2014.

Rui Yan, Han Jiang, Mirella Lapata, Shou-De Lin, Xueqiang Lv, and Xiaoming Li. i, poet: Automatic chinese poetry composition through a generative summarization framework under constrained optimization. In *Proceedings of the 26th International Joint Conference on Artifical Intelligence (IJCAI 2013)*, pages 2197–2203, 2013.

Su Yan and Xiaojun Wan. Srrank: leveraging semantic roles for extractive multi-

document summarization. *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, 22(12):2048–2058, 2014.

Zi Yang, Keke Cai, Jie Tang, Li Zhang, Zhong Su, and Juanzi Li. Social context summarization. In *Proceedings of the 34th Annual International ACM SIGIR Conference on Research & Development in Information Retrieval (SIGIR 2011)*, pages 255–264. ACM, 2011.

Jin-ge Yao, Xiaojun Wan, and Jianguo Xiao. Compressive document summarization via sparse optimization. In *Proceedings of the 28th International Joint Conference on Artifical Intelligence (IJCAI 2015)*, pages 1376–1382, 2015.

Jin-ge Yao, Xiaojun Wan, and Jianguo Xiao. Recent advances in document summarization. *Knowledge and Information Systems (KAIS)*, 53(2):297–336, 2017.

Wenpeng Yin and Yulong Pei. Optimizing sentence modeling and selection for document summarization. In *Proceedings of the 28th International Joint Conference on Artifical Intelligence (IJCAI 2008)*, pages 1383–1389, 2015.

Wenpeng Yin, Lifu Huang, Yulong Pei, and Lian'en Huang. Relationlistwise for query-focused multi-document summarization. In *Proceedings of the 24th International Conference on Computational Linguistics (COLING 2012)*, pages 2961–2976. ACL, 2012.

Naitong Yu, Minlie Huang, Yuanyuan Shi, et al. Product review summarization by exploiting phrase properties. In *Proceedings of the 26th International Conference on Computational Linguistics (COLING 2016)*, pages 1113–1124. ACL, 2016.

David M Zajic, Bonnie Dorr, Jimmy Lin, and Richard Schwartz. Sentence compression as a component of a multi-document summarization system. In *Proceedings of the 2006 Document Understanding Conference (DUC 2006)*, 2006.

David M Zajic, Bonnie J Dorr, and Jimmy Lin. Single-document and multi-document summarization techniques for email threads using sentence compression. *Information Processing & Management*, 44(4):1600–1610, 2008.

Benyu Zhang, Hua Li, Yi Liu, Lei Ji, Wensi Xi, Weiguo Fan, Zheng Chen, and Wei-Ying Ma. Improving web search results using affinity graph. In *Proceedings of the 28th Annual International ACM SIGIR Conference on Research & Development in Information Retrieval (SIGIR 2005)*, pages 504–511. ACM, 2005.

Jianmin Zhang, Jin-ge Yao, and Xiaojun Wan. Towards constructing sports news from live text commentary. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL 2016)*, volume 1, pages 1361–1371. ACL, 2016.

Xiang Zhang and Yann LeCun. Text understanding from scratch. *arXiv preprint arXiv:1502.01710*, 2015.

Yang Zhang, Yunqing Xia, Yi Liu, and Wenmin Wang. Clustering sentences with density peaks for multi-document summarization. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics - Human Language Technologies (NAACL HLT 2015)*, pages 1262–1267. ACL, 2015.

Lin Zhao, Lide Wu, and Xuanjing Huang. Using query expansion in graph-based approach for query-focused multi-document summarization. *Information Processing & Management*, 45(1):35–41, 2009.

Xin Wayne Zhao, Yanwei Guo, Rui Yan, Yulan He, and Xiaoming Li. Timeline generation with social attention. In *Proceedings of the 36th Annual International ACM SIGIR Conference on Research & Development in Information Retrieval (SIGIR 2013)*, pages 1061–1064. ACM, 2013.

Markus Zopf, Eneldo Loza Mencía, and Johannes Fürnkranz. Beyond centrality and structural features: Learning information importance for text summarization. In *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning (CoNLL 2016)*, pages 84–94, 2016a.

Markus Zopf, Eneldo Loza Mencía, and Johannes Fürnkranz. Sequential clustering and contextual importance measures for incremental update summarization. In *Proceedings of the 26th International Conference on Computational Linguistics (COLING 2016)*, pages 1071–1082. ACL, 2016b.