ResQu: A Framework for Automatic Evaluation of Knowledge-Driven Automatic Summarization

A thesis submitted in partial fulfillment of the requirements for the degree of Master of Science

by

NISHITA JAYKUMAR
B.E., Visverwaraya Technological University, 2012

2016
Wright State University

Amit P. Sheth, Ph.D.
Thesis Director

Mateen Rizki, Ph.D.
Chair, Department of Computer Science and Engineering

Robert E. W. Fyffe, Ph.D.
Vice President for Research and Dean of the Graduate School

Committee on Final Examination

Amit P. Sheth, Ph.D.

Thomas C. Rindflesch, Ph.D.

Delroy Cameron, Ph.D.
ABSTRACT


Automatic generation of summaries that capture the salient aspects of a search resultset (i.e., automatic summarization) has become an important task in biomedical research. Automatic summarization offers an avenue for overcoming the information overload problem prevalent in large online digital libraries. However, across many of the knowledge-driven approaches for automatic summarization it is not always clear which features highly impact or influence the quality of a summary. Instead, there has been considerable focus on utilizing schema knowledge to facilitate browsing and exploration of generated summaries \textit{a posteriori}. Such informative features should not be ignored, since they could be utilized to help optimize the models that generate these \textit{semantic summaries} in the first place.

In this research, we adopt a leave-one-out approach to assess the impact of various features on the quality of automatically generated summaries that contain structured background knowledge. We first create the gold standard summaries, using information-theoretic methods, by extraction and validation, then the semantic summaries are transformed into an equivalent textual format. Finally, various similarity metrics, such as cosine similarity, euclidean distance, and Jensen-Shannon divergence are computed under different feature combinations, to assess summary quality against the textual gold standard. We report on the relative importance of the various features used to automatically generate the semantic summaries in a biomedical application. Our evaluation suggests that the proposed approach is an effective automatic evaluation method for assessing feature importance in automatically generated semantic summaries.
Contents

1 Introduction  
1.1 Motivation .......................................................... 3  
1.2 Thesis Organization ............................................. 5  

2 Characteristics of Semantic MEDLINE  
2.1 PubMed searching of MEDLINE ................................. 7  
2.2 Natural language processing ..................................... 8  
2.3 Automatic summarization ....................................... 8  
2.4 Graph output with links to input documents ................. 8  
2.5 Semantic MEDLINE and its usage .............................. 9  
2.6 Characteristic of summaries in Semantic MEDLINE ........ 9  
2.7 Multi-document summarization ................................. 10  
2.7.1 Documents are MEDLINE citations (titles and abstracts) 10  
2.8 Summarization method is semantic abstraction ............ 10  

3 Related Work  
3.1 Extractive Summarization ....................................... 13  
3.2 Abstractive Summarization ..................................... 14  
3.2.1 Opiniosis ....................................................... 15  
3.2.2 Flipboard ...................................................... 15  
3.3 Traditional (e.g. DUC) method ................................. 15  
3.3.1 High cost of creating a gold standard .................... 16  
3.4 Automatic method ................................................ 16  
3.4.1 Louis and Nenkova (2009) The Pyramid Approach ........ 16  
3.4.2 Automatic Summarization without Human Models ........ 17  
3.4.2.1 Based on word distributions ............................ 18  

4 Materials and Methods  
4.1 Materials ............................................................. 19  
4.1.1 Unified Medical Language System (UMLS) ................ 19  
4.1.2 MEDLINE ...................................................... 21  
4.1.3 The Medical Subject Headings (MeSH) .................... 21
4.1.4 MetaMap ....................................................... 22
4.1.5 Semantic MEDLINE ........................................... 22
4.1.6 SemRep .......................................................... 23
4.1.7 Automatic Summarizer .......................................... 23
4.1.8 Gold Standard resource ........................................ 25
4.2 Methods .................................................................. 26
4.2.1 Gold Standard Dataset Representation ..................... 27
4.2.2 Semantic Summary Generation and Representation ........ 30
4.2.3 Comparing semantic summaries to the gold standard .... 31
  4.2.3.1 Problem Formulation ......................................... 31
  4.2.3.2 Query Specification ........................................... 32
  4.2.3.3 Automatic Summarization .................................... 33
  4.2.3.4 Summary Generation .......................................... 35
  4.2.3.5 Summary Transformation ..................................... 35
  4.2.3.6 Dictionary Creation ............................................ 37
  4.2.3.7 Gold standard transformation .............................. 40
  4.2.3.8 Feature Matrix Creation ...................................... 40

5 Evaluation and Metrics ............................................... 43
5.1 Experimental Setup ................................................ 43
5.2 Metrics ................................................................... 45
  5.2.1 Cosine Similarity ................................................ 46
  5.2.2 Euclidean Distance .............................................. 46
  5.2.3 Jensen–Shannon Divergence ................................... 46

6 Conclusion and Future Work ......................................... 51

Bibliography .............................................................. 52
List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.1</td>
<td>A screenshot from the ACP smart Medicine page for the disease dementia</td>
<td>27</td>
</tr>
<tr>
<td>4.2</td>
<td>Screenshot from the UpToDate website for the disease insomnia</td>
<td>28</td>
</tr>
<tr>
<td>4.3</td>
<td>Screenshot from UpToDate showing the drug treatments for the disease migraine disorders</td>
<td>30</td>
</tr>
<tr>
<td>4.5</td>
<td>System architecture for automatic summary generation</td>
<td>33</td>
</tr>
<tr>
<td>4.6</td>
<td>Graphical summary depicting the treatment predication for the concept migraine disorders</td>
<td>34</td>
</tr>
<tr>
<td>5.1</td>
<td>Comparison of Similarity Metrics using Leave-One-Out.</td>
<td>49</td>
</tr>
<tr>
<td>5.2</td>
<td>Comparison of Similarity Metrics using Leave-One-Out.</td>
<td>50</td>
</tr>
<tr>
<td>5.3</td>
<td>Comparison of Similarity Metrics using Leave-One-Out.</td>
<td>50</td>
</tr>
</tbody>
</table>
List of Tables

5.1 Disease Treatments Statistics .............................................. 48
5.2 Root Mean-Squared Error between the baseline and Leave-one-out Methods 49
Acknowledgment
Dedicated to

my mother Kalpana, my father Jaykumar and my husband Akshay
Introduction

The amount of biomedical literature in digital format has grown considerably during the last decade. The US National Library of Medicine (NLM) reports that the MEDLINE repository – considered the largest collection of biomedical literature – now contains over 25 million references to journal articles accessible through the PubMed search interface. This number is only expected to increase in the foreseeable future, with an estimated one million articles anticipated annually based on the current trajectory. However, while the growth in biomedical publications is an encouraging trend, this rapidly increasing volume in scientific literature makes it difficult for researchers to find relevant information for their information needs. A considerable degree of searching and sifting through PubMed search results is still required. Users must necessarily pore over large volumes of text to gain meaningful insights.

One approach that has emerged to alleviate this information overload problem has been automatic summarization. Automatic text summarization is a process of taking a list of search results and distilling it into the most important content, and then presenting the search results in a condensed form to the user [25]. A typical automatic summarization system takes as input one or more documents, and produces a concise summary. Two types of summarization methodologies are are common in the literature: 1) extractive summarization and 2) abstractive summarization. Extractive summarization produces summaries that contain a subset of sentences from the original search results verbatim. This approach can

be likened to a condensation of the source documents according to a *wysiwyg* paradigm – *‘what you see is what you get’* [4]. On the contrary, abstractive summarization is dynamic and produces summaries in which the original content may have been truncated, transformed, or rephrased into an abstract representation of the source documents. Such systems commonly provide topical or thematic summaries that serve as broad labels for the content. This approach can be likened to a *‘what you know is what you get’* paradigm.

Regardless of the specific automatic summarization paradigm, evaluating the quality of the generated summaries is not necessarily a straightforward task. An effective summary (automatically generated or otherwise) should convey the most important information from source documents, in a concise format that is consistent with an information need. To determine whether these criteria are met, an automatic summarization evaluation approach may involve both a qualitative and quantitative evaluation of the summarization system. In the literature, two methods have often been adopted for automatic summarization evaluation: 1) *intrinsic evaluation* and 2) *extrinsic evaluation*. For intrinsic evaluation, highly skilled experts are employed to read documents in a corpus and create gold standard summaries. Automatically generated summaries can then be compared to this human-curated gold standard, using measures of document similarity. In extrinsic evaluation, human judges directly evaluate the quality of the automatically generated summaries (or their impact on a secondary task, such as search relevance or reading comprehension) through a discrete scoring system [16]. Extrinsic evaluations therefore assess the relative impact of summarization systems, while intrinsic evaluations mainly assess the coherence and informativeness of the summaries [26].

Given an automatic summarization method (extractive or abstractive) and an automatic summarization evaluation methodology (intrinsic or extrinsic), determining the quality and features that are important to automatically generated summaries is still not trivial. Effectively applying an intrinsic evaluation to a semantic summary, in particular, is considerably more challenging than evaluating a textual summary, comprised strictly of keywords.
The obvious issue is the misalignment between the structured representation of the semantic summary and the textual representation of the gold standard. We address this problem by first transforming the semantic summary into an equivalent textual representation that could be compared with the textual gold standard. Then, given a transformed summary, we apply a leave-one-out strategy to evaluate the impact of a subset of features on summary quality. To estimate the quality of the semantic summary, various similarity metrics are computed between the transformed summary and the textual gold standard dataset. The features that most significantly influence similarity are identified as important features in determining summary quality.

This pipeline is implemented in our system, called ResQu, to evaluate the quality of 20 automatically generated semantic summaries for a biomedical application. Our evaluation suggests that the leave-one-out approach on transformed semantic summaries is effective for automatically isolating the features that impact the quality of a semantic summary. The specific contributions of this research are as follows:

1. We transform semantic summaries into an equivalent textual representation that is an effective level of abstraction of their information.

2. We adopt a leave-one-out strategy to determine the features that influence the quality of automatically generated semantic summaries.

3. We evaluate the quality of the transformed semantic summaries relative to various gold standard datasets using numerous similarity metrics.

1.1 Motivation

PubMed is a search engine developed at the National Library of Medicine (NLM) that provides search results across more than 25 million biomedical citations. While PubMed searches can be customized to retrieve relevant documents based on its query language
constraints, PubMed search results are still presented as a list. Users must necessarily pore over large volumes of text in the list to find the information that satisfy their information needs. The system is a classical information retrieval-based approach to satisfying a user’s information need.

On the contrary, Semantic MEDLINE is a summarization tool that extracts and visualizes the most salient facts from MEDLINE search results, linking them to several structured resources to connect users more directly with the information being sought. However, while significant strides have been made at achieving informativeness in Semantic MEDLINE, a fundamental challenge is evaluating the overall effectiveness of the system, and then developing insights that can help improve the quality of the generated summaries. The goal of ResQu is therefore to develop an automatic method for evaluating and identifying the features that impact the quality of the automatically generated semantic summaries in Semantic MEDLINE. More specifically, **ResQu has been designed to: 1) automatically evaluate the semantic summaries produced by Semantic MEDLINE, and 2) to gain insights that will help improve the system.**

To achieve this in ResQu, we rely on the idea that facts in a corpus can be effectively represented using the distribution of words that co-occur with such facts – i.e., distributional semantics. Specifically, we posit that the distribution of words that co-occur with the facts captured in a semantic summary is an effective abstraction of the meaning of the semantic summary. We therefore leverage a textual representation of a semantic summary to derive insights into which features are most informative in generating the summaries. The idea is that by generating several summaries with specific features held-out, and computing their similarity to a textual gold standard, an automatic method for evaluating the quality and features that affect semantic summaries can be achieved.
The central thesis of this research is therefore that a semantic summary in biomedicine can be understood and potentially improved by leveraging distributional statistics between the structured knowledge that comprise the semantic summary and the words with which these structured constructs co-occur across a corpus.

1.2 Thesis Organization

The rest of this thesis is organized as follows: Chapter 2 presents a more detailed discussion on Semantic MEDLINE, as a tool for managing PubMed search results aimed at providing users more direct access to the information that may satisfy their information needs. Chapter 3 discusses related work in the field of automatic summarization and automatic summarization evaluation, with a focus on drawbacks in existing work and how such limitations are circumvented in ResQu. Chapter 4 describes materials and resources that help identify and evaluate the features that impact the quality of a semantic summary. This chapter also introduces the core approach in ResQu, including discussions on the data misalignment problem, feature definitions, semantic relatedness, and similarity scoring techniques. Chapter 5 focusses on various aspects of the evaluation, including a use case to which ResQu was applied, along with supporting empirical results. Finally, the conclusion, limitations and a discussion on future directions are covered in Chapter 6.
Characteristics of Semantic MEDLINE

The main motivation for this research is to develop a system that can lead to improvements in the quality of semantic summaries generated by Semantic MEDLINE – an automatic summarization system that manages PubMed search results. Semantic MEDLINE is an application developed at the National Library of Medicine, to manage the results of PubMed searches by summarizing and visualizing facts (also called semantic predications) extracted from MEDLINE citations and linking them to several structured resources in biomedicine to provide an integrated environment [12]. The system is intended to help health care professionals and consumers keep abreast of current research as well as assist researchers in mining the literature to generate hypotheses. At the core of Semantic MEDLINE are two existing tools: 1) SemRep [22], which extracts semantic predications from text in the form of an n-tuple consisting of a subject, predicate, and object, and 2) an automatic summarizer, which distills a collection of facts in a search resultset into a coherent summary in graphical format.

SemRep is a pattern-based relation extraction system designed specifically for biomedical literature, and uses domain knowledge from the Unified Medical Language System (UMLS)\(^1\) for distant supervision [18]. Given a sentence, containing a pair of mentions, SemRep grounds each mention in structured background knowledge, using UMLS Metathesaurus concepts as arguments and UMLS Semantic Network relations as predicates. Processing relies on an underspecified syntactic analysis based on the SPECIALIST Lexicon,

\(^1\text{UMLS – https://www.nlm.nih.gov/research/umls/}\)
and MedPost part-of-speech tagger. MetaMap maps simple noun phrases to Metathesaurus concepts, and “indicator rules” map syntactic elements to Semantic Network predicates. In the semantic abstraction paradigm of automatic summarization [9] semantic predications serve as an abstraction of the meaning of the information presented in the source text, intended as a salient overview of the content.

The automatic summarizer reduces the set of all semantic predications in a search resultset into the most important information in a condensed and coherent format. It inherent provides support for the practice of evidence-based medicine by allowing, for example, users to compare and contrast several treatments for a particular disease. This automatic summarization component in the semantic abstraction paradigm is actively used to help clinicians find the most salient information relevant to a given disease. Further details about the automatic summarizer are discussed in Chapter 4, Section 4.1.7.

Aside from fact extraction (through SemRep) and summarization (using the automatic summarizer), Semantic MEDLINE also provides a simple easy-to-use interface that allows users to pose both simple and complex queries. Semantic MEDLINE then retrieves results from PubMed, which can be visualized for better understanding of the literature and to generate hypotheses. We discuss details of the current search characteristics of PubMed search using MEDLINE in the next Section.

2.1 PubMed searching of MEDLINE

MEDLINE is an online repository that comprises over 25 million citations for biomedical literature from journals, and online books. MEDLINE citations and abstracts include the fields of biomedicine and health, covering portions of the life sciences, behavioral sciences, chemical sciences, and bioengineering. PubMed is a free search service developed and maintained by the National Center for Biotechnology Information (NCBI), at the U.S. National Library of Medicine (NLM), located at the National Institutes of Health (NIH).
It provides a simple search user-interface, for querying the articles stored in MEDLINE. Although PubMed search results come from MEDLINE, the search portal also provides access to additional sites and, which may provide full-text access to retrieved articles, while also providing links to the other NCBI molecular biology resources. Searching on PubMed is straight-forward and one can easily build complex queries to find answers to specific biomedical information needs. For example, to search for a disease concept such as Malaria, we can just type the term Malaria into the search box. The details of constructing queries for PubMed search is discussed later on.

2.2 Natural language processing

Traditional information retrieval techniques typically return excessive output when directed at large bibliographic databases. Natural Language Processing applications strive to extract salient content from the excessive data. A general architecture in an NLP system consists of two main components: background knowledge that includes biomedical knowledge resources and a framework that integrates NLP tools to process text. Systems differ in both components, which we review briefly. Additionally, the challenge facing current research efforts in biomedical NLP includes the paucity of large, publicly available annotated corpora, although initiatives that facilitate data sharing, system evaluation, and collaborative work between researchers in clinical NLP are starting to emerge.

2.3 Automatic summarization

2.4 Graph output with links to input documents

The Semantic MEDLINE that is used to provide semantic summaries to user queries, essentials extracts predications from the MEDLINE documents. Semantic predications are
nothing but relationships between biomedical concepts, represented as a triple in subject-predicate-object format. This semantic summary once generated can be visualized using the tool, where the concepts are the subject and the object represented as nodes in the graph and the relationships or the predicates are the edges connecting the nodes, as seen here.

2.5 Semantic MEDLINE and its usage

Semantic MEDLINE, a National Library of Medicine (NLM) natural language processing application, highlights relevant information in PubMed data. However, Semantic MEDLINE implements manually coded schemas, accommodating few information needs. Currently, there are only five such schemas, while many more would be needed to realistically accommodate all potential users. The aim of this project was to develop and evaluate a statistical algorithm that automatically identifies relevant bibliographic data; the new algorithm could be incorporated into a dynamic schema to accommodate various information needs in Semantic MEDLINE, and eliminate the need for multiple schemas.

2.6 Characteristic of summaries in Semantic MEDLINE

As mentioned earlier, Semantic MEDLINE is a summarization and visualization tool developed at the National Library of Medicine. It is an interactive too. The summaries generated in this tool using a structured schema-based summarizer. There are two types of summarization techniques. Abstractive and Extractive. We will discuss the details of each of the techniques in the later section.

The Semantic MEDLINE tool performs abstractive summarization, where the summary is a condensed abstract representation of the source documents. The schema defines the set of rules for selecting the set of relevant predications. The schema is represented as a set of predications in which the predicate is drawn from a relation in the UMLS Semantic Net-
work. For example, TREATS, ISA, OCCURS, CAUSES, etc. Each domain for the schema is defined in terms of semantic categorization in the Semantic Network. The SemRep is responsible for generating the predications from the input documents and these predications must conform to this schema in order to be included in the conceptual condensate; such predications are called “core predications.”

2.7 Multi-document summarization

Automatic summarization system are built for producing single-document or multi-document summarization. As the name suggests they are produced from single or multiple documents respectively. There exist many systems that produce single or multi-document summaries, studies report that multi-document summaries are more informative compared to single-document summaries because multiple aspects of a single topic is taken into consideration before providing a summary. Semantic MEDLINE is a multi-document summarizer, which takes all the document returned for a search query from PubMed and returns the summarized results.

2.7.1 Documents are MEDLINE citations (titles and abstracts)

For this work of multi-document summarization, the input or the source documents are nothing but the MEDLINE citations returned for a search query. Each citation consists of an unique identifier called as PMID, a title and an abstract of the research article.

2.8 Summarization method is semantic abstraction

As mentioned earlier, there are two techniques for automatic summarization, extractive and abstractive. Semantic MEDLINE uses abstractive summarization where SemRep
the semantic representation tool plays a key role in generating summaries. Predications are summarized using schemas.
Related Work

Automatic summarization has been around for decades, especially in the medical and biomedical domain. It is essential to these domain, as they provide quick answers to complex information needs. In the biomedical domain with the encouragement to publish finding and research conducted has resulted in huge amount of information on the web. For example, in PubMed alone which consists of the largest database of biomedical literature, there exists more than 25M research publications and finding. It seems only apt to have precise and well-developed automatic summarization systems, in order to assist and address physicians complex information needs. Such systems help physicians at point-of-care. The work by [1, 11], is a survey about various state-of-the-art summarization techniques. A considerable degree of searching and sifting through search results is still required. Users must necessarily pore over large volumes of text to gain meaningful insights. One approach that has emerged to alleviate this information overload problem has been automatic summarization.

Automatic text summarization is a process of taking a list of search results and distilling it into the most important content, and then presenting the search results in a condensed form to the user. A typical automatic summarization system takes as input one or more documents, and produces a concise summary. Two types of summarization methodologies are common in the literature: 1) extractive summarization and 2) abstractive summarization. For example in the the work by [10] Lot of work has been done using each of the methodology and also there has been a lot of discussion in the community regarding
which approach is the best for summarization purposes. On one hand extractive summarization is the process of dealing with the source content in a straightforward way, where the summaries usually always contain verbatim from the source documents which are being summarized, i.e., the condensed representation of the source or input document preserves the text from the original text by only selecting top scoring sentences to end up in the resultant summaries.

On the other hand abstractive summarization is basically a condensed abstract representation of the source. Like abstracts that exist in any research publication, describing or summarizing the entire work in a condensed format, which gives a preview of the essential points of the entire publication, abstractive summaries serve the same purpose in summarization. The textual condensate contains essential information from the source documents which are paraphrased or rephrased to express the inherently present information. Single-document summarization and multi-document summarization are other aspects which are important for configuring the summarization systems.

3.1 Extractive Summarization

As mentioned this type of summarization process, concentrates on creating extracts, where the extracts consists of most important sentences from the input in an unaltered format. Some of the work in extractive summarization including the pyramid approach by Nenkova et al, are discussed here. Extracts or Abstracts? When asked to write a summary of a text, people do not normally produce an extract of sentences from the original. Rather, they use their own wording and synthesis of the important information. Thus, exact match of system sentences with human model sentences, as required for recall and precision metrics, is not at all possible. As the field turns to the development of more advanced non-extractive summarizers, we will clearly need to move to a more sophisticated evaluation method which can handle semantic equivalence at varying levels of granularity. The
work by Elhadad et al., is yet another work which discusses about creating patient specific summaries of medical articles using extraction[17].

Automatic summarization has been around for many years and as mentioned earlier there has been constant effort for implementing automatic methods for evaluating these summaries and their systems. For example, in the work by Radev et al. a large scale fully automatic evaluation of 18,000 documents was performed without human effort, using an information retrieval scenario[15]. Like most systems, their system is also motivated by the assumption that the distribution of terms in a good summary is similar to the distribution of terms in the original document. In the work by Donaway et al. they conclude that automatic methods for comparing summary and input should be seriously considered as a alternative method of summary evaluation. The most wide scale study conducted using extrinsic methods was done within the framework of the TIPSTER and SUMMAC project, where systems were tested for efficiency and for finding text relevant to the search query. Intrinsic evaluations can be divided into two following groups: evaluations based on comparisons with the model summary and evaluations based on comparisons with source document. Nenkova et al., have implemented the pyramid approach for automatic summary evaluation. Here, we present ResQu, which is a system for automatic evaluation of summaries. We evaluate summary quality by directly estimating its closeness to the input. We adopt information-theoretic approaches for characterizing the similarity and differences between input and summary content.

3.2 Abstractive Summarization

There exists a lot of work in the abstractive summarization space. Below we discuss about some of the prominent examples of this kind of work.
3.2.1 Opinosis

The work by Ganesan et al. describes the system Opinosis[8] which is a graph-based summarization system. Opinosis generates concise abstractive summaries of highly redundant opinions. They report about the evaluation strategy that have better agreement with human summaries compared to baseline summaries. Opinosis uses a form of shallow abstraction and assumes no domain knowledge and uses shallow NLP, leveraging mostly the word order in the existing text and its inherent redundancies to generate informative abstractive summaries.

3.2.2 Flipboard

Flipboard is a social-network aggregation, magazine-format mobile app localized in more than 20 languages. They adopt the LexRank approach to automatic summarization[5]. Flipboard is yet another example of how abstractive summarization is more useful and informative. Although they perform single-document summarization. With limited space, delivering important information in a concise form is crucial. In the following section we discuss about the existing work in automatic summary evaluation, the limitations and also about how our work is different.

3.3 Traditional (e.g. DUC) method

Automatic summarization and its evaluation has been an ongoing effort of many large scale conferences such as the Document Understanding Conference(DUC) and Text Retrieval Conference(TREC). The refinement and usability analysis of these evaluation techniques have been the focus of large scale evaluation efforts such as the Document Understanding Conferences (DUC) (Baldwin et al., 2000; Harman and Over, 2004; Over et al., 2007) and TIPSTER SUMMAC reports (Mani et al., 2002)
3.3.1 High cost of creating a gold standard

They also report that the construction and creation of a gold standard is not trivial and involves a lot of manual effort for which is not time-consuming and tedious. Human annotators are not consistent or reliable. Quote text from nenkova’s paper

3.4 Automatic method

There has been other efforts for automatically evaluating the summaries.

3.4.1 Louis and Nenkova (2009) The Pyramid Approach

In the work by Nenkova et. al. they describe the importance of content selection in summarization. They also mention about the difficulty in evaluating content select. In the domain of automatic summarization, there exists no single best model summary. The DUC evaluation has a number of limitations, such as, the use of a single model summary. All research in summary evaluation has reported that no single summary exists. DUC cannot be used as a good metric for evaluation[19]. In the Pyramid approach, Nenkova et al., have worked on created the Summary Content units, which are basically created from annotation of a corpus of summaries and are not bigger than a clause.

For example: A1 In 1998 two Libyans indicted in 1991 for the Lockerbie bombing were still in Libya. B1 Two Libyans were indicted in 1991 for blowing up a Pan Am jumbo jet over Lockerbie, Scotland in 1988. C1 Two Libyans, accused by the United States and Britain of bombing. D2 Two Libyan suspects were indicted in 1991 Each SCU has a weight corresponding to the number of summaries it appears in; SCU1 has weight=4 and SCU2 has weight=33 :

SCU1 (w=4): two Libyans were officially accused of the Lockerbie bombing
A1 [two Libyans]1 [indicted]1
B1 [Two Libyans were indicted]1
C1 [Two Libyans,]1 [accused]1
D2 [Two Libyan suspects were indicted]1

SCU2 (w=3): the indictment of the two Lockerbie suspects was in 1991
B1 [in 1991]2
D2 [in 1991.]2

The remaining parts of the four sentences above end up as contributors to nine different SCUs of different weight and granularity. Instead of developing a method to elicit reliable judgements from humans, they choose to calibrate a method to characterize Human Summarization behaviour. This method not only scores a summary but also allows investigator to find what imp info is missing, thus helps target direct improvements in the summarizer.

The other work by Halteren and Teufel, 2003 also describes a similar approach for creating factoids. Nenkova also reports that there are many open questions about how to parameterize a summary for specific goals, making evaluation in itself a significant research question. Instead of attempting to develop a method to elicit reliable judgements, they chose to calibrate their method to human summarizer behavior.

**Limitation:** Creating an initial pyramid is laborious so large-scale application of the method would require an automated or semi-automated approach.

### 3.4.2 Automatic Summarization without Human Models

Another work by Louis et al., talks about automatic summary evaluation along the same lines. Their work mainly focuses on implementing an automatic approach for summary evaluation that does not require the creation of human models. The intuition of their work is that a summary contains the most representative information of the input. Hence one can expect the distribution of terms is the input and a good summary are similar. To
compare the distribution of terms in the summary to the input, the techniques used were, Kullback-Leibler divergence, Jensen- Shannon divergence. They report that their best feature Jensen-Shannon divergence, leads to a correlation as high as 0.9 with manual evaluations. They were motivated by the fact that Automatic text summarizers are expected to produce a condensed form of an input, retaining the most important content and presenting the information in a coherent fashion. They also report that the construction and creation of a gold standard is not trivial and involves a lot of manual effort for which is not time-consuming and tedious[15].

3.4.2.1 Based on word distributions

Many systems now typically use the distribution of words as a measure of distance for computing similarities between input and summaries. Method for comparing frequency distributions Cosine similarity, Euclidean distance, Jensen-shannon
Materials and Methods

This section discusses the core approach and implementation developed to identify and evaluate the features that impact the quality of a semantic summary. We first discuss the different datasets and resources utilized in this research, and later we discuss the specifics of the implementation to create and evaluate the features and semantic summaries.

4.1 Materials

Several datasets are used for our task of identifying and evaluating the features that impact the quality of a semantic summary. Some of these datasets include: 1) the Unified Medical Language System (UMLS), 2) MEDLINE, 3) Semantic MEDLINE, 4) the Medical Subject Headings (MeSH), and 5) a gold standard dataset.

4.1.1 Unified Medical Language System (UMLS)

The UMLS is a compendium of more than 130 biomedical vocabularies. It consists of three main components: 1) the Metathesaurus, 2) Semantic Network, and 3) SPECIALIST Lexicon. The Metathesaurus is the biggest component of the UMLS. It is a large biomedical thesaurus that is organized by concept, or meaning, and it unifies similar names for the same concept to a unique identifier from nearly 200 different vocabularies\(^1\). The Metathesaurus:

The Metathesaurus also expresses useful relationships between concepts and preserves the meanings, concept names, and relationships from each vocabulary item. Additionally, the Metathesaurus provides a way of mapping raw text queries to standard UMLS concepts. Mapping natural language text to UMLS concepts is an important task in our application since the first step in generating semantic summaries is formulating the search query that will be used to retrieve relevant documents from the corpus from which the summaries will be constructed. ResQu is inherently a semantic search system that translates natural language queries into structured knowledge, and then performs document search. Structured queries as well as structured knowledge extracted from documents in the corpus are both used in the search process. The Metathesaurus is the knowledge source that grounds concepts in the query and document to a common vocabulary.

The Semantic Network consists of 134 high-level categories for classifying the biomedical concepts that belong to the UMLS according to broad semantic types\(^2\). This network also contains 15 higher-level semantic groups, which are organized in a hierarchy. The Semantic Network also expresses explicit relationships between the semantic types and groups, through 54 unique predicates. In terms of organization, the semantic network is comparable to an ontology schema, while the Metathesaurus is comparable to the instances in the ontology. In ResQu, the Semantic Networks enables filtering information that constitute the semantic summaries.

The SPECIALIST Lexicon is a large syntactic lexicon of biomedical and general English terms, designed to provide the information needed for information extraction by various tools and natural language processing system. Although useful, we do not directly use it in this research, but use tools such as MetaMap and SemRep, that rely on it instead.

4.1.2 MEDLINE

The second dataset used in ResQu is collectively the MEDLINE database and the PubMed search service. MEDLINE is a repository of more than 25 million bibliographic citations maintained by NLM, which can be used to provide relevant documents for a search query (whether textual or semantic). It is an authoritative bibliographic database containing citations and abstracts for biomedical and health journals used by healthcare professionals, nurses, clinicians and researchers engaged in clinical care, public health and health policy development. PubMed provides an easy-to-use interface and powerful search functionality including basic and advanced search options to access MEDLINE articles. Users can effectively search for and retrieve information from PubMed using Medical Subject Headings or MeSH terms (discussed in Section 4.1.3). Document retrieval relies on semantic matching of MeSH terms in the query and documents in the MEDLINE corpus.

4.1.3 The Medical Subject Headings (MeSH)

The Medical Subject Headings (MeSH) is a controlled vocabulary (and thesaurus) of biomedical terms, organized in a hierarchical structure. Subject headings in MeSH are often used as search terms in PubMed to retrieve relevant documents. MeSH is organized into a category hierarchy of 16 trees (as of the time of this writing), with a maximum depth of 15 for more than 27,000 distinct descriptors. The focus of MeSH is on categorization and organization of biomedical knowledge. As such, MeSH does not contain any explicit predicates among descriptors. In practice, it is used for indexing, cataloging, and searching MEDLINE. In recent research [4], MeSH has been used separately to provide both implicit and explicit context for semantic predications to facilitate literature-based discovery.

To facilitate MEDLINE search, MeSH descriptors are manually assigned to scientific articles by individuals called MeSH Indexers. The Medical Term Indexer (MTI)\(^3\) is used

\(^3\)MTI Indexer: [https://ii.nlm.nih.gov/MTI/](https://ii.nlm.nih.gov/MTI/)
to first generate a list of candidate descriptors which are subsequently finalized by the indexers. The quality of these assignments is considered high and the assignments are relatively good indicators of the semantics of the content of the article to which they are assigned. The MeSH descriptors therefore serve a dual role: 1) as definitional knowledge in biomedicine and 2) as a layer of abstraction for the semantics of the content of biomedical articles.

MeSH is used in ResQu to provide descriptors for query formulation at the query specification step of the summarization task. However, note that since MeSH descriptors are also used by the underlying MEDLINE search system, MeSH is useful in ResQu both for structured query formulation and document search.

### 4.1.4 MetaMap

The next resource used in this research is MetaMap (specifically, MetaMap API developed at NLM). MetaMap is used to map biomedical text to the UMLS Metathesaurus concepts (not MeSH descriptors) or equivalently, to discover Metathesaurus concepts referred to in text. MetaMap uses a knowledge-intensive approach based on symbolic, natural-language processing (NLP) and computational-linguistic techniques. The system was developed by Aronson [2] and is widely used in biomedical text applications. We use MetaMap in ResQu to extract concepts from the gold standard dataset, and subsequently to create a semantic representation of its content.

### 4.1.5 Semantic MEDLINE

The next resource used in ResQu is Semantic MEDLINE (SemMedDB) [14][24]. Semantic MEDLINE is a Web application that manages the results of PubMed searches by summarizing and visualizing semantic predications extracted from MEDLINE citations and linking them to several structured resources to provide an integrated browsing and ex-
ploration environment. It consists of more than 65 million facts extracted from MEDLINE. Semantic MEDLINE leverages two datasets to manage MEDLINE search results. First, it leverages SemRep [23] to extract semantic predications (subject-predicate-object triples) from MEDLINE and then it uses an Automatic Summarizer [6] (discussed further in 4.1.7) to restrict the set of predications to only those that convey salient information from the search resultset. Semantic MEDLINE provides a simple user interface, in which a user can input a query and specify various search parameters such as the number of citations, date, etc. The search is performed on the entire MEDLINE collection to first retrieve relevant results. Semantic predications extracted from these documents in the search results using SemRep, are then visualized to provide users direct access to the information expressed within the documents, instead of the documents themselves.

4.1.6 SemRep

SemRep was developed by Rindflesch et al. [23] at NLM, and is trained specifically to extract semantic predications from biomedical free text in the form subject-predicate-object. For example, from the sentence: “We used hemofiltration to treat a patient with digoxin overdose that was complicated by refractory hyperkalemia,” SemRep extracts the following semantic predications: [Hemofiltration-TREATS-Patients], [Digoxin overdose-PROCESS_OF-Patients], and [hyperkalemia-COMPLICATES-Digoxin overdose]. SemRep predications extracted from MEDLINE are stored in SemMedDB [] and made publicly available for research.

4.1.7 Automatic Summarizer

The next resource used in ResQu is the Automatic Summarizer. After a PubMed query is issued to Semantic MEDLINE, SemRep extracts semantic predications from the MEDLINE citations (titles and abstracts). These are then put through a process of automatic abstraction summarization (discusses in Chapter 1) before the final output is produced,
which consists of semantic predications. The initial graph of predications is called the *predications graph*, denoted $G$, while the final graphical summary is called the *semantic summary*, denoted $S$. The automatic summarizer as described by Fizman et al. has four key steps[?]. The *Summarizer* uses the following features to distill the predications graph $G$ into a semantic summary $S$: 1) relevance, 2) connectivity, 3) novelty, and 4) saliency.

*Relevance* – is a knowledge-based feature derived by selecting semantic predications that address the user-selected seed topic for the summary. For example, if the user chose the UMLS topic *Coronary Arteriosclerosis*, the summary should include predications on this topic, such as *[Aspirin TREAT Coronary Arteriosclerosis]*.

*Connectivity* – is a feature that ensures the summary will also include “useful” additional predications, such as *Coronary Arteriosclerosis COEXISTS_WITH Inflammation* based on the connectedness or relevant concepts. Connectivity filtering would identify other predications which also include such arguments and add them to the Relevance group.

*Novelty* – is a knowledge-based feature that uses the hierarchical structure of the Metathesaurus to eliminate predications with generic (and hence uninformative) arguments. For example, *Coronary Arteriosclerosis ISA Vascular Disease(s)* is considered non-informative and therefore should be removed from the summary.

*Saliency* – is a feature that assigns bias to semantic predications that occur frequently. For example, the predication *tomography DIAGNOSIS Coronary Arteriosclerosis* would be included in the final output if it occurred a sufficient number of times in the search results.

For completeness, consider the the graphical summary produced by semantic MEDLINE for the query Migraine Disorders[MH] AND Humans[mh] AND Clinical Trial [PTYP]
AND 1860/01:2014/08[DCOM] shown in Fig:4.4a, and the associated semantic types in Fig:4.4b. As shown in the figure, the graph contains different kinds of semantic types including, TREATS, ISA, PREVENTS, LOCATION_OF, and PROCESS_OF these are all relationships listed in the Semantic Network.

### 4.1.8 Gold Standard resource

The final dataset used in ResQu is the gold standard dataset. To evaluate the impact of various features on the quality of a semantic summary, the summary is evaluated against a textual gold standard, using a leave-one-out strategy. We focus on the four key features that are used for summarization and perform the leave-one-out analysis to observe which features contribute to the better quality of the summaries. The gold standard dataset for our evaluation task was semi-automatically created using human judges on data generated from three websites. These specific websites are: 1) UpToDate 2) Micromedex, and 3) American College of Physicians - ACP Smart Medicine, which were chosen because they contain evidence-based, decision-support information designed for point-of-care use by internists and other physicians. A total of 20 search queries related to disease treatments were used to generate a subset of documents presented to domain experts for relevance and coherence adjudication (see Section:4.2.3.7 for further discussion). Consequently, the approach adopted in ResQu uses abstractive summarization with an intrinsic evaluation in this research.

Each of the resources were carefully chosen as authoritative sources of information from the physicians point-of-care. UpToDate, is the premier evidence-based clinical decision support resource, trusted worldwide by healthcare practitioners to help them make the right decisions at the point of care. It is proven to change the way clinicians practice medicine, and is the only resource of its kind associated with improved outcomes. A section of the webpage from UpToDate describing the drug treatments for the disease Insomnia can be seen here in the Fig:4.2 and in Fig:4.6, the drug treatments for the disease
migraine disorders is shown.

The next resources is Micromedex, which is also an evidence-based clinical resource. Micromedex is an unbiased, referenced Clinical Decision Support (CDS) for medication, toxicology, disease, acute care and alternative medicine. Easy-to-use online tool to effectively manage and update a hospital’s formulary and communicate the most current formulary information facility-wide. A tailored Micromedex interface providing trusted evidence and global guidance with a Pharmaceutical perspective.

Finally, The American College of Physicians (ACP) is a national organization of internists - physician specialists who apply scientific knowledge and clinical expertise to the diagnosis, treatment, and compassionate care of adults across the spectrum from health to complex illness. ACP is the largest medical-specialty organization and second-largest physician group in the United States. Its membership of 143,000 includes internists, internal medicine subspecialists, and medical students, residents, and fellows. For these reasons, we choose the three resources for creating the gold standard set.

Other candidate gold standard websites such as PatientsLikeMe were considered but not included because they are very subjective. For instance, in PatientsLikeMe, people suffering from a specific disease discuss their symptoms and what drug treatments worked for them. These are not necessarily certified by physicians. We were interested in creating a dataset which describes disease treatments to assist physicians and is proven research.

4.2 Methods

In order to identify and evaluate which features impact the quality of a semantic summary we must unify the gold standard dataset and the semantic summaries using the same representation so that the summary can be effectively evaluated against the gold standard, with some features excluded. As described in the work by Plaza et al. they report the impact of various knowledge sources in automatic summarization[20]. For the gold stan-
4.2 Gold Standard Dataset Representation

We create the gold standard dataset by identifying the resources and extracting the necessary data then we transform the dataset by using the bag-of-words model by vectorizing the gold standard. The format of the data on the three websites where different. The
Figure 4.2: Screenshot from the UpToDate website for the disease insomnia

drug treatments for a disease are displayed in a more structured format as tables in the ACP website compared to UpToDate. The data on Micromedex was semi-structured as well.

A table on the webpage from ACP Smart Medicine describing the drug treatments for the disease Insomnia can be seen here in the Fig:4.2. We use the Jericho HTML Parser\(^4\), which is a Java library to parse HTML pages in the gold standard. It provides high-level HTML pages manipulation functions. We first start by using this parser; we use this simple line of code to provide the start page for the parser to extract all text from the HTML markup.

Source source = new Source(new URL(startPage));

List<Element> bold = source.getAllElements(HTMLElementName.B);

for(Element ele: bold){
    String bold_ele = ele.getContent().toString();

---

\(^4\)Jericho HTML Parser: [http://jericho.htmlparser.net/docs/index.html](http://jericho.htmlparser.net/docs/index.html)
In the above line, if the HTML source page for the disease malaria is provided, the parser looks for custom, specified tags, such as drug treatments, etc., from the markup to extract all drug treatments from the supplied pages, which are bold-faced. This makes it easy as we can now look for all occurrences of the “bold” tag in specific sections of these webpages. Some website have more structured data. The ACP Smart Medicine website lists the set of drug treatments for a specific disease in a tabular format and we parse all the text between the `<td></td>` tags.

For each of the diseases, we extract the text between the HTML tags and serialize the textual data to files on disk. Next, we use the MetaMap API\(^5\) to map mentions of UMLS concepts in the gold standard text to UMLS concepts. This task of parsing the text files with the MetaMap Parser, enables creation of a curated list of treatments in the text, correlated with treatments present in the Metathesaurus. Finally, this restricted list of concepts pertaining to treatments was provided to the domain experts for review and for identifying the final list of concepts representative of drug treatments for various diseases, as taken from the gold standard. These concepts are then vectorized using the same technique used for the semantic summaries, which is covered in Section 4.2.3.7.

\(^5\)MetaMap: https://metamap.nlm.nih.gov/
4.2.2 Semantic Summary Generation and Representation

The semantic summaries initially consist of semantic predications, which are assertions extracted from biomedical literature consisting of a subject, predicate, and an object. The subject and the object are Metathesaurus concepts and the predicate represents the relationship generated between them, which are also present in the Semantic Network. For illustration, the summary for the disease migraine disorders is shown in Fig:4.4a, while the different semantic types are in Fig:4.4b.

Given a search query, the Automatic Summarizer generates a graphical summary of the initial search results from MEDLINE. This graphical summary is a collection of semantic predications extracted from the MEDLINE documents in the search resultset. The graphical summary visualizes the search results as a graph of interconnected facts. This semantic summary is then subsequently transformed into a textual format, effectively changing its
representation in order to overcome the misalignment between the semantic summary and the gold standard summary. In the next section we discuss details of the approach for transforming a semantic summary into an textual representation to enable evaluation against the gold standard summary.

4.2.3 Comparing semantic summaries to the gold standard

In this section, we discuss: 1) how the semantic summaries are transformed into an equivalent textual representation, 2) how the gold standard is represented, and 3) how the leave-one-out approach is applied to evaluate the semantic summary against the gold standard. We begin with the problem formulation.

4.2.3.1 Problem Formulation

Given a query $q$, a set of documents $D = \{d_1, d_2, \ldots, d_z\}$, a set of textual gold standard summaries for $q$, denoted $T = \{t_1, t_2, \ldots, t_n\}$, along with a feature set $F = \{f_1, f_2, \ldots, f_m\}$ used to generate a semantic summary $S$, then determine the subset of features $F'$ that gen-
erate a semantic summary $S_{\text{min}}$, which has the minimum similarity to the gold standard $T$, such that $|F'| = |F| - k$, for $k \geq 1$. The feature(s) $I = F - F'$ are said to be the most informative features.

To determine the feature set $I$ for a query $q$, we apply the following 4 steps: 1) query specification, 2) automatic summarization, 3) summary transformation, and 4) automatic summarization evaluation. We begin with query specification in the next Section.

4.2.3.2 Query Specification

The input to ResQu is a sequence of keywords, formatted for a MEDLINE search, in the form $q = (l, c_1, c_2, dt, ub)$, where $l$ is the label of an entity (or concept) in the UMLS, $c_1$ and $c_2$ are MeSH filters, $dt$ is the date range of documents to be included in the automatic summary generation task, and $ub$ is the upper bound on the number of search results to include for summarization (default = 5000). Since natural language query understanding is not the focus of the research, we manually map the label $l$ to a concept unique identifier (or CUI) in the UMLS using the Semantic Navigator. For example, the label Migraine is mapped to the UMLS concept C0149931, whose label is Migraine Disorders. We then further constrain the label by suggesting its use as a MeSH term in the query $q$. In this way, the label becomes Migraine Disorders[MH]. The filters $c_1$ are used to restrict search results to those for clinical trials conducted on humans. For example, $c_1 = Humans[MH]$ and $c_2 = Clinical Trial [PTYP]$. The final query may therefore be specified as a concatenation of clauses with the boolean $\text{AND}$ operator as $q = (\text{Migraine Disorders}[MH] \text{ AND Humans}[MH] \text{ AND Clinical Trial [PTYP]} \text{ AND } 1860/01:2014/08[DCOM])$, where $dt = 1860/01:2014/08[DCOM]$ and $ub$ is omitted for simplicity. If no date range is given ResQu uses the entire MEDLINE database. This query is useful for the initial document selection that drives the automatic summarization task, which is discussed in the next section.

---

### Automatic Summarization

Given a query formatted for a MEDLINE search, the next task is retrieval of an initial set of documents that will be condensed into a summary – that is, the actual automatic generation of the semantic summary. To automatically generate semantic summaries, we reuse the approach developed by Fiszman et al. [7], which creates a semantic summary as a graph of facts extracted from MEDLINE articles. Assertions extracted from MEDLINE are called *semantic predications*, which are also made available through Semantic MEDLINE. Fiszman takes the following steps, which we also execute in ResQu: 1) document selection, 2) semantic predication extraction, and 3) summary generation. These steps are captured in Figure 4.5.

#### Document Selection.

The goal of document selection is to use the search query to retrieve a list of search results from which the summary can be generated. Given a query $q$, specified and formatted accordingly, using the *User Query Processor*, the *Document Selector* (Figure 4.5, top...
then retrieves the set of MEDLINE documents $D = \{d_1, d_2, \ldots, d_n\}$ by utilizing the PubMed Search API. From the search results, the PubMed identifier (or PMID) of each article in $D$ is then passed to the Semantic Predication Extractor, whose task is to extract semantic predications from each document and create an initial labeled directed graph, called a *predications graph* $G = (V, E)$, which is discussed in the next section.

**Semantic Predication Extraction.**

The predications graph $G$, derived from the set of semantic predications extracted from the documents in $D$ is a labeled relational graph, in which $V$ is the set of unique concepts extracted from $D$ and $E$ is the set of unique edges that connect them. Concretely, a node in the graph represents a UMLS concept, such as *Migraine* and an edge represents a UMLS predicate. The set of documents $D$ is therefore input to the Semantic Predication Extractor (Figure 4.5, top center), which uses the SemRep Service to extract facts from each document. The predications graph is then delivered as input to the *Summarizer*, which applies various features to filter our non-informative semantic predications and create a more concise semantic summary reflective of the salient aspects of the search resultset. The specific features and approach used for this task are discussed in the next section.
4.2.3.4 Summary Generation.

The predications graph is then delivered as input to the Summarizer, which applies various features to filter our non-informative semantic predications and create a more concise semantic summary reflective of the salient aspects of the search result set.

4.2.3.5 Summary Transformation

As noted in the Section: introduction, the fundamental challenge in semantic summary evaluation is the misalignment between the representation of the semantic summary and the textual gold standard. In our specific case, the semantic summary is a labeled relational graph $S = (V', E')$, where $S \in G$, $V' \in V$ and $E' \in E$. We transform this relational summary into a textual summary by gleaning textual context for each node in the graph from MEDLINE.

Specifically, given a vertex $v$ in $V'$, we obtain the subset of documents $D(v)$ from $D$ in which the concept $v$ occurred, and use context clues around $v$ to create a sparse bag-of-words (bows) representation of $v$ from $D(v)$. Let this bow of context clues for $v$ be denoted $D(v, \ast)$, such that $D(v, \ast) = [(w_1, tf_1), (w_2, tf_2), \ldots, (w_j, tf_j)]$, where $w$ represents any word that co-occurs with $v$ in $D(v)$, and $tf_j$ represents the global term frequency of the word $w$ in $D(v)$. We can then combine the bows of each concept in $V'$ into a unified bag-of-words model of the summary $S'$, denoted $U'$, captured by Equation 4.1.

$$U' = \bigcup_{v \in V'} D(v, \ast). \quad (4.1)$$

The unified bag-of-words $U'$ can then be expressed as a weighted $n$-dimensional term-frequency inverse-document frequency (tf-idf) vector using the document frequencies $df$ of each word $w$ across the entire document set $D$ to normalize the term frequencies – as is standard practice in information retrieval. Recall that tf-idf is a weighting scheme that
estimates the importance of words in a corpus. For example, given a term \( t \), document \( d \) and a corpus of documents \( D \), the importance of the word \( t \) in document \( d \) is computed in Equation 4.2 as

\[
\text{tfidf}(t, d, D) = tf(t, d) \times \log \frac{N}{n_t},
\]

(4.2)

where \( N \) is the total number of document in the vocabulary, \( n_t \) is the number of documents in the corpus that contain the term \( t \) (i.e., document frequency) and \( tf(t, d) \) is the term frequency of the term \( t \) in document \( d \). In this way, we obtain

\[
\vec{S}' = [\hat{w}_1, \hat{w}_2, \ldots, \hat{w}_n],
\]

(4.3)

as the semantic summary vector for \( S' \), where \( \hat{w}_i \) is the tfidf weight of the \( i^{th} \) co-occurring term with \( v \) in the feature space, and \( n \) is the number of dimensions or features, typically the size of the vocabulary or less, depending on top-tailing. In a similar process, a tfidf vector for the gold standard dataset, denoted \( \vec{T} = \{\hat{t}_1, \hat{t}_2, \ldots, \hat{t}_n\} \), is created, where \( \hat{t}_i \) is the tfidf weight of the \( i^{th} \) term in this n-dimensional feature vector.

The transformation of the semantic summary into its tfidf vector \( S' \rightarrow \vec{S}' \) is handled by the Concept Mapper (Figure 4.5, top right). Note that alternatively transformed summaries could be achieved by leveraging context clues from documents in \( D \) that contain the semantic predications in the summary \( S' \) instead of only the concepts in \( V' \), independent of the predicates \( E' \). This is handled using the Predication Mapper (Figure 4.5, top right), which has been developed to perform this task. Evaluation of transformed summaries using context clues by semantic predication in MEDLINE remains as future work. In the next section we discuss the specific similarity metrics used to measure the quality of a summary under this transformation.
4.2.3.6 Dictionary Creation

A key challenge that abstraction summary evaluation faces is in the data representation. As mentioned earlier, there is a misalignment in the representation of the gold standard dataset and the semantic summary. To align the gold standard dataset and the semantic summary we represent each as a distribution of words. To achieve this, we first create a vocabulary of all words in the space and use it to help express the gold standard as a vector and the semantic summary as a vector, in which weights are assigned to each of the words that occur in the respective datasets. Words are weighted using the common term-frequency inverse-document frequency (TF-IDF) metric.

To create the vocabulary (or dictionary), we use the gensim library\(^7\) in python due to its ease of use. Gensim is a Python library for topic modeling, document indexing and similarity retrieval with large corpora. This library is used mostly for performing tasks relating natural language processing (NLP) and information retrieval (IR). The dictionary is a data structure that encapsulates the mapping between the normalized words and their index position in the features space using a simple integer id.

To create the dictionary, we iterate over each document in the corpus, tokenize the text using a whitespace, and add each token to the dictionary. The resulting feature space consists of 22803 terms from a list of 5000 documents for each of the 20 diseases that are encompassed in this research.

This small snippet of code describes some of the steps taken for dictionary creation and summary transformation.

```python
import numpy as np
from numpy import loadtxt

class TfidfLoader(object):
```

\(^7\)Gensim Python Library – [https://radimrehurek.com/gensim/](https://radimrehurek.com/gensim/)
def __init__(self, filepath=None, dictionary=None, 
    tfidfmodel=None):
    
    if not os.path.exists(filepath):
        print("File path does not exist \%s", filepath)
    else:
        if not dictionary:
            dictionary = self.load_dictionary(filepath)
        if not tfidfmodel:
            tfidfmodel = TfidfModel(dictionary=
                dictionary)

    self._dictionary = dictionary
    self._tfidfmodel = tfidfmodel

def load_dictionary(self, filepath):
    
    dictionary = Dictionary()
    with open(filepath, "rb") as f:
        for line in f.readlines():
            dictionary.add_documents([[word.lower() for 
                word in line.split()]])

    # print dictionary

    return dictionary
def load_data(input_file, delim="",):
    data = np.loadtxt(input_file, delimiter=delim)
    print(data.shape)
    print(data.shape[0])
    n_features = data.shape[1]-1
    if n_features == 1:
        X = data[:,0]
    else:
        X = data[:,np.arange(n_features)]
    y = data[:,n_features]
    assert len(X) == len(y)
    print("Number of training examples: %i" % X.shape[0])
    print("Number of features: %s" % X.shape[1])
    return X, y

# X,y = load_data("test1.txt");
# print(X.shape)

X,y = load_data("output-z1.csv");
print(X.shape)
4.2.3.7 Gold standard transformation

The Gold standard dataset was created using steps very similar to the semantic summary creation. As discussed in Section 4.1.8, after we have the list of concepts derived using a series of steps for extraction and finally compiling the list with the domain expert approval, we next get the list of PMIDs for each of the concepts in the list. We need to keep in mind that at this point each of the gold standard summaries is a list of Metathesaurus concepts. Once we have the list of PMIDs that each of the concepts occurs in, we fetch the PubMed articles for each of them. Then we concatenate the text from each of the articles, to create a single textual summary for a concept.

4.2.3.8 Feature Matrix Creation

In order to overcome the misalignment problem with the gold-standard and the topic summaries, we performed the following steps. We addressed the issue from the summary level, we had a total of 34 topics (20 disease topics, 14 drug topics) and a total of 48 datasets. This list consists of 20 disease treatments, 14 drug interactions and 14 drug adverse effects. Once this dataset is pre-processed, we turn to the gold-standard dataset which is created using the data on websites such as UptoDate, Micromedex, American College of Physicians and Mediscape. Next comes the transformation phase, where we have a gold standard set of documents, if for a given query Q1, we have a single gold standard (GS) summary, then we need to represent the GS as a feature vector and the input semantic summary as a feature vector. For example in disease topics, the semantic summary is nothing but a list of concepts discussing the drug treatments of the specific disease, then we can create one feature for the semantic summary and one for the GS. We are essentially trying to get the feature vector for the semantic summary. So first:

Once the dictionary is created using the dataset. We load the serialized dictionary, using a command such as:
dictionary = corpora.Dictionary.load('/dict-name.dict')

then, for each GS document text.split. then GS dictionary.add_documents(list of lists) to the dictionary This will create one dict with the all tokens (corpus + GS docs) in it Then one without the GS docs (Skip for now) Then what we need to do it essentially: in the summary we have a concept and for each concept get the list of PMIDs. We need to create for each concept a bag-of-words representation. To do that we will need to know, given a PMID what are the set of concepts that occur in it. Create a python dictionary and for each concept we need a tfidf vector:

For this first have a bag-of-words vector for the concept get each of the abstract it occurs in, now we have a text blob for each concept. For each concept in the summary, get each of the pmid it occurs and get the abstracts and pass this huge blob of text to the tfidf model. One dictionary for the corpus and using this create the feature matrix, with the dimensions as specified in the algorithm.

We create a feature matrix using the following algorithm:
Algorithm 1 Feature matrix creation

1: procedure PT–APPROACH
2: get the list of summaries for the 20 disease topics
3: get the list of summaries for the 14 disease topics
4: for each concept in the list of 34 topics do
5: get the list of PMIDS
6: for each PMID in the PMID list do
7: get a multi-document textual summary, by concatenating the text from the title and abstract of each of the Pubmed article.
8: end for
9: end for
10: Convert the list of concept token to bag of words using the dictionary
11: represent this dictionary as an n-dimensional numpy-array
12: for each textual summary of a topic do
13: create a sparse 1 dimensional vector representation of the data, having the same length of the dictionary (this partially addresses the misalignment problem.)
14: end for
15: next create a feature matrix
16: for each topic summary do
17: create one positive sample
18: create one negative sample using the NOT query.
19: end for
20: next to prevent class imbalance create a feature vector/matrix where a total size of the dataset should be (48 x 2) * 22803
21: now we have a feature matrix X (dimensions 96 x 22803)
22: create a labels vector of the dimensions 96 x 1
23: finally perform machine learning experiments, to automatically classify informative versus non-informative summaries.
24: end procedure
Evaluation and Metrics

We first describe the steps for experimental setup. The various transformational steps and then the evaluation

5.1 Experimental Setup

The title and abstracts from each of carefully chosen 34 concepts form the input dataset. Each pubmed article contains many fields and some of them include PMID, DATE, etc. The format of the pubmed article is as below:

**PMID** - 23415202

**TI** - [Sequential antibiotic therapy in complicated acute appendicitis].

**AB** - Initial results show a shorter hospital stay and lower costs for the group of children with sequential postoperative antimicrobial therapy.

In order to overcome the misalignment in the data of the input and the gold standard, with the help of the gensim library we created the bag-of-words model we were able to overcome this issue. The bag-of-words model essentially transforms the traditional textual document into a mapping of terms and its frequencies. With this model we use the straightforward tf-idf scoring. Here, we used tf-idf, a simple transformation which takes documents represented as bag-of-words counts and applies a weighting which discounts
common terms (or, equivalently, promotes rare terms). It also scales the resulting vector to unit length (in the Euclidean norm).

Our input data set consists of a list of 5000 documents for each of the 20 diseases and 14 drugs that are present in our seed dataset. For this we first have to create a "Corpus" from the input documents. Corpus is simply an object which, when iterated over, returns its documents represented as sparse vectors.

Next the TFIDF model is used for converting the documents from one vector representation into another. For the TF-IDF model to process the documents, the documents and its data should be in the bag-of-words representation. The intuition behind the transformation of the documents is as follows:

- To bring out hidden structure in the corpus, discover relationships between words and use them to describe the documents in a new and (hopefully) more semantic way.
- To make the document representation more compact. This both improves efficiency (new representation consumes less resources) and efficacy (marginal data trends are ignored, noise-reduction).

Next I implement the Vector space model for similarity computation using the python libraries. For this computation we need to implement the summary transformation module. The fundamental challenge in semantic summary evaluation is the misalignment between the representation of the semantic summary and the textual gold standard.

We transform this relational summary into a textual summary by gleaning textual context for each node in the graph from MEDLINE. We created a dictionary using all the textual summaries of all diseases as input. The dictionary encapsulates the mapping between normalized words and their integer ids. This converts a collection of words to its bag-of-words representation. We create a sparse vector representation of the data using the bag-of-words model. The tfidf score of a term ‘x’, it represents the total global term frequency of the word ‘x’. We can then combine the bows of each concept in V into a unified
bag-of-words model of the summary $S$, denoted $U$. Once we have transformed the semantic summary into a sparse vector representation, we transform the gold standard representation to the similar dimension.

This transformation gets rid of the misalignment issue and this facilitates the similarity scoring between the summaries. I compute the similarity using the traditional cosine similarity, euclidean distance and jensen shannon metrics. Finally given the values, we calculate the root-mean-squared-error from the baseline for each of these measures. For cosine-similarity: we expect that the feature that has the least mean-squared error from the baseline is the most informative feature.

On the contrary, for the the euclidean distance we expect the feature that has the largest euclidean distance from the baseline to be the most informative. In this research, we developed a method to automatically evaluate the quality of a semantic summary, which contains structured background knowledge. We use the leave-one-out strategy to determine which features impact the quality of a summary based on the similarity of the summary to a gold standard.

### 5.2 Metrics

We first discuss the different similarity metrics that we consider for our evaluation. We used several similarity metrics to compute the similarity between the n-dimensional gold standard summary vector $\vec{T} = \{\hat{t}_1, \hat{t}_2, \ldots, \hat{t}_n\}$ and the semantic summary vector $\vec{S} = [\hat{w}_1, \hat{w}_2, \ldots, \hat{w}_n]$. Each of these is discussed briefly in this section.
5.2.1 Cosine Similarity.

We compute the cosine similarity between a semantic summary vector $\mathbf{S}'$ and the gold standard summary vector $\mathbf{T}$ as

$$\text{cosine}(\mathbf{S}', \mathbf{T}) = \frac{\mathbf{S}' \cdot \mathbf{T}}{||\mathbf{S}'|| ||\mathbf{T}||}. \tag{5.1}$$

5.2.2 Euclidean Distance.

We compute the euclidean distance between a semantic summary vector $\mathbf{S}'$ and the gold standard summary vector $\mathbf{T}$ as

$$\epsilon(\mathbf{S}', \mathbf{T}) = \sqrt{\sum_{i=1}^{n} (\hat{w}_i - \hat{t}_i)^2}. \tag{5.2}$$

5.2.3 Jensen–Shannon Divergence.

Jensen-Shannon measure as described by Briet et al[3] is a symmetrized and smoothed version of the all important divergence measure of information theory, the Kullback-Leibler divergence. We also compute the JS-divergence between a semantic summary vector $\mathbf{S}'$ and the gold standard summary vector $\mathbf{T}$. Jensen-Shannon relies on Kullback-Liebler divergence, which we compute as

$$KL(\mathbf{S}' || \mathbf{T}) = \sum_{i=1}^{n} P(w_i) \log \frac{P(w_i)}{P(t_i)}, \tag{5.3}$$

where $P(w_i)$ is the probability of the $i^{th}$ term in the semantic summary vector of $S'$ and $P(t_i)$ is the probability of the $i^{th}$ term in the gold standard. The JS-Divergence is then
computed as
\[
JSD(S'||T') = \frac{1}{2}[KL(S'||M) + KL(T'||M)],
\] (5.4)

where \( M = \frac{1}{2}(S' + T') \). Given these metrics, our leave-one-out approach for assessing features in automatic summarization is achieved in 6 steps.

**Step 1:** Select 20 disease labels as topics for an information need.

**Step 2:** Use each query to generate a semantic summary.

**Step 3:** Transform each semantic summary into a semantic summary vector.

**Step 4:** Transform each gold standard into gold standard tfidf vectors.

**Step 5:** Compute the similarity between a semantic summary vector and its associated gold standard vector under different features.

**Step 6:** Determine the features that generate the most informative summary in each scenario.

In the next section we discuss our approach for evaluating the impact of various features on the transformed semantic summary using different similarity metrics.

To evaluate our system we investigate summary quality for an information need regarding “how drug interventions impact diseases through treatment.” The experiment consists of the following 20 topics on disease treatments: Acute sinusitis (3615), Migraine Disorders, Diabetes Mellitus (Non-Insulin-Dependent), Creutzfeldt-Jakob Syndrome (41), Tuberculosis, Sickle Cell Anemia, Glaucoma (Open-Angle), Sleeplessness, Torsades de Pointes (2863), Malaria, CA pneumonia, Bell Palsy (1071), Dementia, Pheochromocytoma (1889), Appendicitis, Acute myocardial infarction, Rabies, Pneumothorax, Obesity, and hypercholesterolemia. The five topics with numbers in parenthesis contained less than 5000 documents after searching MEDLINE, while all other topics were restricted to the top 5000 documents retrieved. These diseases cover frequently occurring curable diseases and some cover rarely occurring incurable diseases. We also perform some filtering to re-
Table 5.1: Disease Treatments Statistics

<table>
<thead>
<tr>
<th>Disease Topics</th>
<th>3615</th>
<th>Migraine Disorders</th>
<th>-</th>
</tr>
</thead>
<tbody>
<tr>
<td>Acute sinusitis</td>
<td>-</td>
<td>Creutzfeldt-Jakob Syndrome</td>
<td>41</td>
</tr>
<tr>
<td>Diabetes Mellitus, Non-Insulin-Dependent</td>
<td>-</td>
<td>Sickle Cell Anemia</td>
<td>-</td>
</tr>
<tr>
<td>Tuberculosis</td>
<td>-</td>
<td>Sleeplessness</td>
<td>-</td>
</tr>
<tr>
<td>Glaucoma, Open-Angle</td>
<td>-</td>
<td>Torsades de Pointes</td>
<td>2863</td>
</tr>
<tr>
<td>CA pneumonia</td>
<td>-</td>
<td>Malaria</td>
<td>1071</td>
</tr>
<tr>
<td>Dementia</td>
<td>-</td>
<td>Pneumothorax</td>
<td>-</td>
</tr>
<tr>
<td>Appendicitis</td>
<td>-</td>
<td>Pheochromocytoma</td>
<td>1889</td>
</tr>
<tr>
<td>Rabies</td>
<td>-</td>
<td>Acute myocardial infarction</td>
<td>-</td>
</tr>
<tr>
<td>Obesity</td>
<td>-</td>
<td>hypercholesterolemia</td>
<td>-</td>
</tr>
</tbody>
</table>

strict the context semantic summary according to our information need. The availability of structured background knowledge for constraining a semantic summary is one of the main advantages for generating semantic summaries, and is showcased in the Semantic MEDLINE search application [13], [24].

Specifically, for each of the 20 disease treatments topics the generated semantic summary was restricted to only contain semantic predications that have the predicate TREATS, since the information need is regarding disease treatments. Additionally, since the diseases are UMLS concepts themselves we filter out semantic predications that do not mention the disease as the object of the predication. For example, the semantic summary for the disease Dementia should contain the predications [donepezil TREATS Dementia] and [Risperidone TREATS Dementia] extracted from the search results for this topic.

Figure 5.3 shows the distribution of cosine similarity, euclidean distance, and jensen-shannon divergence scores for our leave-one-out approach applied to the four features (and the baseline, all features) used to generate the summaries by the Summarizer. Table 5.2 shows the root mean-squared error between the baseline and the leave-one-out methods. Assuming that the baseline is the upper bound in performance, for the cosine similarity scores, we expect that the feature that has the least mean-squared error from the baseline is the most informative feature, since it’s removal produces the least overall similarity to
Figure 5.1: Comparison of Similarity Metrics using Leave-One-Out.

Table 5.2: Root Mean-Squared Error between the baseline and Leave-one-out Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Cosine-RMSE</th>
<th>Euclidean-RMSE</th>
<th>JS-RMSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Leave-out-relevancy</td>
<td>0.263</td>
<td>0.315</td>
<td>0.187</td>
</tr>
<tr>
<td>Leave-out-connectivity</td>
<td>0.263</td>
<td>0.335</td>
<td>0.143</td>
</tr>
<tr>
<td>Leave-out-novelty</td>
<td>0.254</td>
<td>0.329</td>
<td>0.252</td>
</tr>
<tr>
<td>Leave-out-saliency</td>
<td><strong>0.237</strong></td>
<td>0.333</td>
<td><strong>0.281</strong></td>
</tr>
</tbody>
</table>

the gold standard – that is $S_{min}$. This feature is saliency. On the contrary, for the the euclidean distance we expect the feature that has the largest euclidean distance from the baseline to be the most informative, since it’s exclusion results in the highest error from the baseline. While this feature is connectivity, it is followed closely by saliency. Finally, for the jensen-shannon divergence we expect that feature that creates the largest divergence from the baseline to be the most informative feature. Again this feature is saliency.
Figure 5.2: Comparison of Similarity Metrics using Leave-One-Out.

Figure 5.3: Comparison of Similarity Metrics using Leave-One-Out.
Conclusion and Future Work

In this research, we developed a method to automatically evaluate the quality of a semantic summary, which contains structured background knowledge. We use the leave-one-out strategy to determine which features impact the quality of a summary based on the similarity of the summary to a gold standard. To compute similarity, we first transformed the graphical semantic summary into a textual summary, bootstrapping entities with context clues derived from MEDLINE. Our experiments enable us to identify which features are important in automatic summarization for the Semantic MEDLINE application, thereby ResQu to articulate a way forward for optimizing the quality of semantic summaries generated by Semantic MEDLINE.
Bibliography


[22] Thomas C Rindflesch and Marcelo Fiszman. The interaction of domain knowledge and linguistic structure in natural language processing: interpreting hypernymic


