Opinion-aware Information Management: Statistical Summarisation and Knowledge Representation of Opinions

Marco Bonzanini

University of London

Thesis submitted for the degree of Doctor of Philosophy

at Queen Mary University of London

March 2015

Declaration of originality

I, Marco Bonzanini, confirm that the research included within this thesis is my own work or that where it has been carried out in collaboration with, or supported by others, that this is duly acknowledged below and my contribution indicated. Previously published material is also acknowledged below.

I attest that I have exercised reasonable care to ensure that the work is original, and does not to the best of my knowledge break any UK law, infringe any third party's copyright or other Intellectual Property Right, or contain any confidential material.

I accept that the College has the right to use plagiarism detection software to check the electronic version of the thesis.

I confirm that this thesis has not been previously submitted for the award of a degree by this or any other university.

The copyright of this thesis rests with the author and no quotation from it or information derived from it may be published without the prior written consent of the author.

Some parts of this work have been previously published as:

- S. Yahyaei, M. Bonzanini, and T. Roelleke. Cross-lingual text fragment alignment using divergence from randomness. In Proceedings of the 18th International Symposium in String Processing and Information Retrieval, SPIRE '11, pages 14–25, 2011.
- M. Bonzanini, M. Martinez-Alvarez, and T. Roelleke. Investigating the use of extractive summarisation in sentiment classification. In Proceedings of the 3rd Italian Information Retrieval (IIR) Workshop, 2012.
- H. Azzam, S. Yahyaei, M. Bonzanini, and T. Roelleke. A schema-driven approach for knowledge-oriented retrieval and query formulation. In Proceedings of the Third International Workshop on Keyword Search on Structured Data, KEYS '12, pages 39–46, 2012.
- M. Bonzanini. A knowledge-based approach for summarising opinions. In Proceedings of

the 35th international ACM SIGIR conference on Research and development in information retrieval, SIGIR '12, pages 991–991, 2012.

- M. Bonzanini, M. Martinez-Alvarez, and T. Roelleke. Opinion summarisation through sentence extraction: an investigation with movie reviews. In Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval, SIGIR '12, pages 1121–1122, 2012.

- M. Bonzanini, M. Martinez-Alvarez, and T. Roelleke. Extractive Summarisation via Sentence Removal: Condensing Relevant Sentences into a Short Summary. In Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval, SIGIR '13, pages 893–896, 2013.

Other publications and collaborations:

- T. Roelleke, M. Bonzanini, and M. Martinez-Alvarez. On the Modelling of Ranking Algorithms in Probabilistic Datalog. Invited Paper. In 7th International Workshop on Ranking in Databases, DBRank, 2013.

- M. Martinez-Alvarez, M. Bonzanini, and T. Roelleke. Mathematical Specification and Logic Modelling in the context of IR. In Proceedings of the 4th International Conference on the Theory of Information Retrieval, ICTIR '13, page 30, 2013.

- T. Roelleke, H. Azzam, M. Bonzanini, M. Martinez-Alvarez, and M. Lalmas. The D2Q2 Framework: On the Relationship and Combination of Language Modelling and TF-IDF. In Information Retrieval Workshop at LWA, 2013.

Abstract

Nowadays, an increasing amount of media platforms provide the users with opportunities for sharing their opinions about products, companies or people. In order to support users accessing opinion-based information, and to support engineers building systems that require opinion-aware reasoning, intelligent opinion-aware tools and techniques are needed. This thesis contributes methods and technology for opinion-aware information management from two different perspectives, namely document summarisation and knowledge representation.

Document summarisation has been widely investigated as a mean to reduce information overload. This thesis focuses on statistical models for summarisation, with a particular attention to divergence-based models, within the context of opinions. Firstly, topic-based document summarisation is addressed, contributing a study on divergence-based document to summary similarity and the definition of a novel algorithm for summarisation based on sentence removal. Secondly, summarisation models are tailored to opinion-oriented content and shown to be useful also when exploited for different tasks such as sentiment classification. Thirdly, summarisation models are applied to knowledge-oriented data, in order to tackle tasks such as entity summarisation. The comprehensive task addressed is the knowledge-based opinion-aware summarisation of content (free text, facts).

This thesis also contributes a broad discussion on knowledge representation of opinions. A thorough study on how to model opinions using traditional techniques, such as Entity-Relationship (ER) modelling, underlines that a high-level, opinion-aware layer of conceptual modelling is useful since it hides away implementation details. A conceptual and logical knowledge representation methodology for modelling opinions is hence proposed, with the purpose of guiding engineers towards the use of best practices during the development of sentiment analysis applications. Specifically, an extension of the traditional ER modelling and the definition of an automatic mapping procedure, to translate opinion-aware components of the conceptual model into a relational model, help achieving a clear separation between conceptual and logical modelling. The mapping procedure yields an automatic and replicable methodology to design applications which require opinion-aware reasoning.

Contents

List of Figures

Acknowledgements

This thesis is the outcome of a long journey, which I could not have completed without the help and support of many many people.

My first acknowledgement goes to my supervisor, Thomas Roelleke, who has guided me through many aspects of academic life. I deeply value his expertise, his technical suggestions and the casual extracurricular activities, like playing golf and tasting wine, which we enjoyed together.

My examiners, Dr. Emine Yilmaz and Prof. Stefan Rueger, have deeply improved the quality of my work, thanks to their comments and insights.

I am really grateful to have met some really talented people at Queen Mary, great colleagues whom I can call friends. Among the many who have shared a piece of my journey with me, I would like to mention some of the former and current colleagues, in and outside the RIM Group: Sirvan Yahyaei, Miguel Martinez-Alvarez, Hany Azzam, Nuzhah Gooda Sahib, Tassos Tombros, Tony Stockman, Fabrizio Smeraldi, Yihan Tao and Kayras Bhesania. Some of them helped me shaping some of my ideas, or simply spent some time with me discussing about work, life and whatnot. There have been beers, muffins, barbecues, gym sessions, pool matches, football matches, fancy lunches, not-so-fancy lunches, and much more, yet we managed to get quite some work done. I am truly thankful for all the moments we have shared.

I left these last words to mention the people who are closer to me. I am really blessed for the unconditional support from my family: my parents Giovanni and Graziana, and my sister Anna. The last, but not least, person whom I want to thank is my girlfriend Daniela. She is the one who has encouraged me since the very beginning of this journey, and the one who has supported me the most, always trying to lift my spirit during bright and dark days, and providing a pleasant environment to go back to after work. She has also helped me improving some of my papers and some parts of this thesis by asking some smart questions which always started with *I am not an expert, but...*

Chapter 1

Introduction

1.1 Motivation

The expansion of the Web is providing an increasing number of social media such as blogs, discussion forums and other services, where users can express their opinions about products, companies or people. Finding out what other people think has always been an important part of our decision-making process [Pang and Lee, 2008]. Customers can exploit this opinion-oriented information before buying a product, watching a movie, or hiring a professional. Companies can acquire information about the opinion of their customers towards their products, also in comparison to the competitors' ones. Finding some opinion-oriented content about a particular product is nowadays not a difficult task for common users. On the other side, processing the amount of available information can be very challenging for an individual user. In order for this information to be effective and not overwhelming, intelligent sentiment-aware tools and techniques are needed.

One of the important tasks for smart information management is text summarisation, i.e. the process of identifying the key information from a document and presenting it in a shortened version. A well-crafted summary is beneficial for a user who can quickly grasp the main points of a document. The user can then decide whether reading the whole text is worthwhile or not, saving precious time while discarding unimportant information. The cost of performing the summarisation task by means of a professional (human) abstractor has triggered substantial work

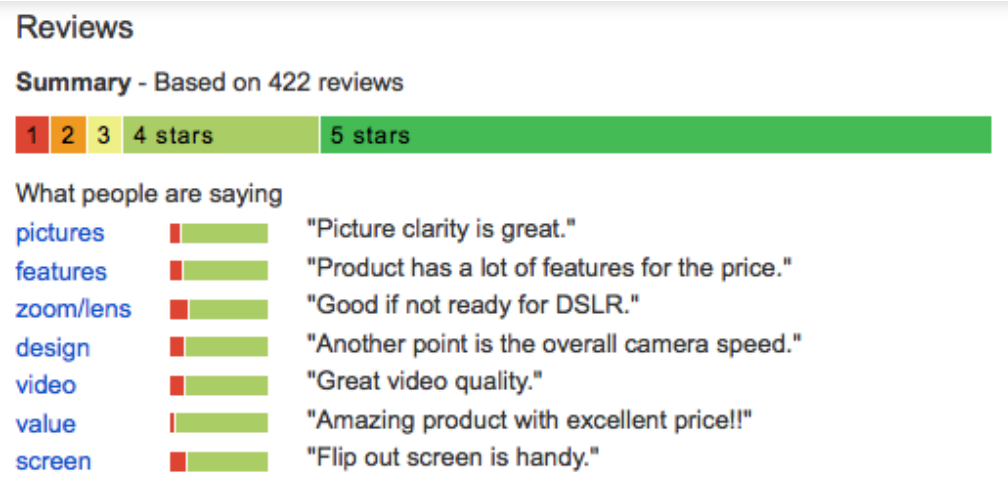
to automate the process since the 1950s [Luhn, 1958], and the interest in this area is presently very active [Nenkova and McKeown, 2011].

User-generated and opinion-oriented content brings new challenges to summarisation. Firstly, the main focus is on the opinion expressed in the document rather than its topic. An opinion-oriented summariser must consider the sentiment as core information to be captured. A second aspect is inconsistency. Dealing with inconsistent sources can be typical in some application domains, such as real-time news reports, where inaccuracies can be amended over time. The key difference with sentiment analysis relies on the fact that inconsistencies do not necessarily depend on inaccuracies, but simply, different reviewers can have different opinions. A good summary should balance positive and negative aspects from different reviews. A third challenge is related to the nature of web comments, which are commonly very short and informal. Summarising web comments can be seen as a sort of multi-document summarisation task, where each document is a single comment, and it can be one-sentence long. Extracting the key aspects from a very short document can be particularly challenging. Users often ask technical questions, which are on topic but do not express opinions. Sentiments are also expressed in a way which is unclear when the context is not explicit, due to the use of sarcasm or implicit notions like direct quotations from a movie. In the next chapter, Figure 2.8 will report some examples of web comments, where the use of sarcasm and direct quotations are difficult to understand out of context.

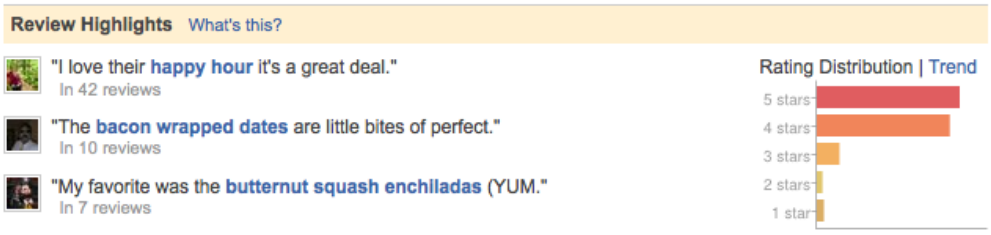
Previous investigations on sentiment analysis have described how this task is particularly domain-specific [Pang et al., 2002]. Reviews on bank services are heavily different from movie reviews, in terms of jargon and formalism. Moreover, opinions can be expressed on individual components of the object under analysis, and sometimes "the whole is not necessarily the sum of the parts" [Turney, 2002]. As an example, the two domains of banks and movies can be compared. Once positive reviews on bank services are put together, the result is a positive opinion about the bank. On the other side, charming actors and a fascinating soundtrack do not necessarily lead to an enjoyable movie (e.g. the plot could be terribly boring). The use of domain-specific knowledge can be crucial in the process of extracting opinions from such reviews. Figure 1.1 shows examples of review summaries from Google Shopping[1] and Yelp[2].

These examples share some common aspects. Firstly, the summaries are produced analysing and

[1] http://www.google.com/shopping (Accessed December 2014)
[2] http://www.yelp.com (Accessed December 2014)



(a) Example of camera reviews summary from Google Shopping.



(b) Example of restaurant reviews summary from Yelp.

Figure 1.1: Examples of review summaries.

aggregating multiple sources, in these cases multiple textual reviews as well as numerical rating. Secondly, the selection of the textual information to show is based on statistics, i.e. "what people are saying" about the products is to be intended as "what most of the people are saying". On the other side, one main difference to consider is the explicit identification of features/aspects to discuss in the first summary, while such features/aspects simply emerge from the text in the second summary.

The amount of available data is not the only aspect which is nowadays increasing, as another important factor to consider is the variety and heterogeneity of data, with e.g. linked data and structured data being associated to unstructured data (like the textual and rating information in the examples). While richer data also means more opportunities to exploit such rich information, the common document representation for information retrieval applications is still based on bag-of-words, i.e. each document is represented by the set of its terms. Opinions add an extra dimension to the space of mere term statistics. For example, within the context of sentiment analysis, observing the term *good* in a movie review may provide more information than observing the

movie title in the same document. Similarly, knowing that a user's favourite director was the one working on the making of a movie may influence the user's opinion about the movie itself.

This thesis explores the area of opinion-aware information management, drawing the attention towards two main aspects: how to summarise content which carries opinions, and how to represent opinions and knowledge related to opinions.

In terms of document summarisation, the thesis focuses on statistical extractive techniques. Term statistics are used to score sentences which will be extracted verbatim to form the summary. Such extractive methods based on statistics provide an advantage over more sophisticated natural language processing methodologies in terms of computational complexity, so they are easier to apply in a large scale context. The downsides include the readability of the output if e.g. the chosen sentences together do not provide a coherent and consistent summary. Statistical extractive methods are good candidates for applications like highlight generation (as the examples in Figure 1.1), because the required output is not a fluent and coherent summary, but rather a one-sentence on-focus snippet. The thesis approaches the problem of extractive summarisation firstly for the general case, i.e. topic-based summarisation. An introduction on similarity and divergence-based sentence scoring approaches leads to the definition of a novel summarisation technique based on sentence removal. Secondly, opinion-bearing content is considered. In particular, it is important to identify words or sentences which carry opinions, in order to treat them separately. Several approaches for pre-processing of opinion-bearing terms based on dictionaries are discussed. The main advantage of using dictionaries for sentiment analysis applications is that they are relatively cheap to obtain or generate. The main limitation is the lack of context, as some terms can carry very different sentiments in different conditions. The different pre-processing approaches are evaluated in the context of sentiment summarisation using the aforementioned sentence removal algorithm. This thesis also contributes a study on subjectivity detection at the sentence level. Using labelled data, supervised learning approaches are used to distinguish subjective sentences (which carry opinions) from objective sentences (which do not carry opinions). Subjectivity detection is used as means to sentiment summarisation, enabling the user to understand the overall sentiment of a document without the need to read the whole text.

The aspects regarding knowledge representation have also been studied firstly from a general, topic-based point of view, and secondly with respect to opinions. The connection between the work on statistical summarisation and knowledge representation is the definition of knowledge-

based summarisation, which is the process of building a summary from a knowledge base. This thesis builds upon the case of knowledge-oriented retrieval to lay the ground for the knowledge representation applied to knowledge-based summarisation. A major application of this technique is entity summarisation, which aims at summarising an entity (e.g. an actor or a product) by ranking the relationships it is involved in. In other words, the knowledge related to an entity is used to create an entity profile. As a concrete example, actors are summarised by listing the movies they are famouse for. One interesting question is how to represent knowledge related to opinions, or how to integrate opinions in a knowledge base, in order to represent concepts such as *good actor* or *terrible food*. In order to tackle this question, the thesis reviews traditional knowledge modelling techniques, such as Entity-Relationship Modelling, and illustrates the shortcomings of these techniques. The thesis then contributes a methodology to extend such techniques to represent opinions by keeping a clear separation between conceptual and logical layer. In this way, the definition of *what* to represent and *how* to represent it are kept decoupled, encouraging the use of best practices for data design. The next step discussed in this thesis is the implementation of an opinion-aware conceptual model into a logical (relational) data model, by an automatic translation of the model. In this way, the proposed opinion-aware knowledge representation supports a guided data design process, forming the foundation for verticals which have to support opinion-oriented requirements.

The reminder of this chapter is organised as follows: Section 1.2 enumerates the main research questions addressed by this thesis. Section 1.3 lists the main contributions of this thesis, chapter by chapter. Section 1.4 provides the overall outline of the thesis.

1.2 Research Questions

This thesis addresses a number of research questions as listed below. The two main areas of study are statistical models for extractive summarisation and knowledge representation, in the context of opinion-bearing information.

Within the work related to statistical models for extractive summarisation, the following research questions are investigated:

1. How do geometric and information-theoretic methods (e.g. cosine and divergence) compare in the context of sentence selection, in particular w.r.t. summarisation quality?

2. Given an iterative approach to produce summaries by removing the less important sentences, how does it perform in terms of summarisation quality, compared to approaches which select the most important sentences?

3. Given an approach to recognise opinion-bearing terms, how does term boosting affect the quality of sentence selection for sentiment summarisation, and how can terms be treated in order to improve quality on sentiment summarisation?

4. How is it possible to summarise a document while preserving its overall polarity information?

The second pillar of this thesis is knowledge representation. In particular, the following research questions are investigated within this context:

5. What kind of technologies and methodologies are needed in order to enable knowledge-based summarisation?

6. What kind of expressivity and flexibility do traditional conceptual modelling techniques provide in terms of modelling opinions? In other words, are traditional conceptual modelling concepts enough to model opinions?

7. How is it possible to provide additional semantic concepts in order to model opinions at the conceptual level, and how to de-couple such modelling from the logical layer?

8. What else is needed in order to support sentiment analysis applications which model opinions at the conceptual layer? In other words, once a conceptual modelling of opinions is available, how to map it into the logical layer?

1.3 Summary of Contributions

The contributions of this thesis can be grouped into two main areas: statistical models for extractive summarisation and knowledge representation. The work has been carried out with a focus on user-generated and opinion-bearing content. The structure of the thesis reflects this perspective. The contributions, listed by chapter, are hence summarised as follows:

- **Chapter 3:** Extractive Summarisation based on Statistical Models

Investigation on Divergence-based Methods. The first part of the chapter lays the groundwork about similarity measures based on divergence, which support the following discussion on specific summarisation-oriented tasks.

Summarisation via Sentence Removal. An algorithm for extractive summarisation based on sentence removal is proposed. The approach iteratively removes unimportant sentences until the desired output size is reached.

- **Chapter 4:** Opinion-based Extractive Summarisation based on Statistical Models

  Treatment of Opinion-bearing Terms. An investigation on how to treat opinion-bearing terms w.r.t. sentiment summarisation is carried out. Different techniques include stop-word removal, term frequency boosting, bi-grams and negation detection based on dictionaries.

  Summarisation via Subjectivity Detection. Subjectivity detection at a sentence level is used as a mean for sentiment summarisation. An investigation on whether sentiment summarisation helps sentiment classification is proposed. In particular, summarisation via subjectivity detection preserves the polarity of a document while shortening its text. This is beneficial for a user who does not need to read the full text in order to understand the polarity of a review.

- **Chapter 5:** Knowledge-based Summarisation

  Definition of Knowledge-based Summarisation. The overall process of knowledge-based summarisation is discussed, as a technique to build summaries exploiting a knowledge base, and a parallel with knowledge-based (semantic) retrieval is drawn.

  Knowledge Representation for Summarisation. Concepts from knowledge-based retrieval are brought into the context of summarisation. The knowledge representation supports the modelling of objects (not just documents) such as persons or movies and it allows to apply summarisation techniques in order to extract the most important facts about specific objects.

  Entity Summarisation. A particular application of knowledge-based summarisation is entity summarisation, where the content to summarise is not a document, but a given object/entity, like a movie or an actor. Techniques for entity summarisation based

on the proposed knowledge representation are formalised, and a scenario based on actors and movies is discussed.

- **Chapter 6:** Knowledge Representation of Opinions

  Representation of Opinions at the Conceptual Layer. A study on how to represent opinions using traditional Entity-Relationship (ER) and Enhanced ER modelling is carried out. The study shows how traditional methodologies fall short in representing the semantics of opinions and motivates the discussion for an opinion-aware enhancement of traditional ER Modelling (ERM).

  Opinion-aware Enhancement of ERM. New concepts for modelling opinions are added to traditional ER, providing a high-level conceptual specification to represent opinions and semantics about opinions. This methodology supports a clear separation of the representation of opinions between conceptual layer and logical layer.

  Mapping to the Logical Layer. Once opinion-aware conceptual modelling tools are available, the question is how to support sentiment analysis application which take advantage of them. A procedure to automatically map opinion-aware conceptual schemata into logical (relational) schemata is proposed. The mapping process generates relations for contextual and global opinions. Overall, the chapter provides groundwork for enhanced ER modelling capable to capture opinion-oriented requirements. The proposed knowledge representation supports a guided data and software design process.

1.4 Thesis Outline

The remaining chapters are outlined as follows:

Chapter 2 presents the background and reviews related work. The main sections discuss concepts of information retrieval, automatic document summarisation, sentiment analysis, summarisation of opinion-oriented content and knowledge representation.

Chapter 3 discusses extractive summarisation techniques based on statistical methods, with a focus on similarity and divergence. A novel technique for summarisation, based on sentence removal, is proposed.

Chapter 4 studies the treatment of opinion-bearing terms in the context of extractive summarisation. Summarisation via subjectivity detection at the sentence level is investigated.

Chapter 5 examines knowledge-based summarisation. Methodologies for knowledge-based summarisation are discussed, and the task of entity summarisation is introduced.

Chapter 6 discusses the knowledge representation of opinions.

Chapter 7 concludes the thesis and summarises the findings.

Chapter 2

Background and Literature Review

2.1 Concepts of Information Retrieval

Information Retrieval (IR) is the discipline concerned with the representation, storage, organisation of, and access to information items. The purpose is to provide the users with easy access to the information they are interested in [Baeza-Yates and Ribeiro-Neto, 1999].

The goal of an IR system is to supply relevant documents to the user, in response to his information need. An IR process starts with the user representing his information need in the form of a query, i.e. a set of keywords. The IR system supports the user by finding documents to satisfy the information need, usually ranking the documents according to their relevance to the query. Figure 2.1 shows the traditional conceptual model of information retrieval [Croft, 1993]. The collection of documents is processed to produce an internal document representation, called index. Similarly, the (keyword-based) query is an internal representation for the information need. Such representations are matched by the system using a retrieval function. The result is a ranked list of document, with the goal of ranking relevant documents above the non-relevant ones. Once the user is presented with the ranked list of documents, a relevance feedback step can be included in the process, e.g. to reformulate the query.

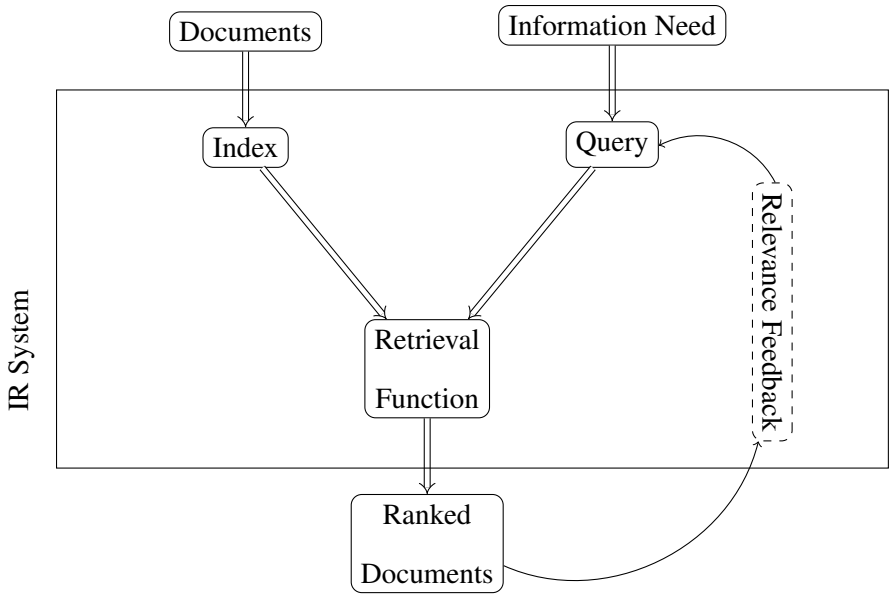


Figure 2.1: Conceptual model of Information Retrieval.

2.1.1 Document Representation

IR systems typically transform the full text version of documents into an internal representation which reduces their complexity and make them easier to manage. The internal representation is called index, and its purpose is to maintain an accurate description of the documents, while discarding details which are not important for the scope of the IR system. The indexing process assigns a set of features to a document identifier. A common representation consists of a *bag-of-words*, a simplified version of the document where grammar and word order are ignored, while word frequencies and word positions (for phrase match) are considered.

Figure 2.2 shows an example of possible representations for two documents, d_1 and d_2. The features considered in this examples are word frequencies, i.e. the number of times each term appears in a document. In a key-value representation, each feature is represented by the term itself as key and its frequency as value. In a vector-based representation, each document is represented as a vector of frequencies. The zeros in the vectors correspond to terms which are present in the collection but not in the document.

Before the documents are indexed, they are usually the subject of some pre-processing steps. For example, the documents in Figure 2.2 have been tokenised into individual terms, all the tokens have been lowercased and the punctuation has been removed. Other common pre-processing steps worth mentioning are stop-word removal and stemming.

Full text documents	
d_1	Peter is a sailor.
d_2	Peter likes boats. He also likes Mary.
Key-value index	
d_1	peter: 1; is: 1; a: 1; sailor: 1
d_2	peter: 1; likes: 2; boats: 1; he: 1; also: 1; mary: 1
Vector-based index	
d_1	[1, 1, 1, 1, 0, 0, 0, 0, 0]
d_2	[1, 0, 0, 0, 2, 1, 1, 1, 1]

Figure 2.2: Example of document representations.

Stop-word removal is the process of removing words which are not content-bearing, such as articles, propositions, etc. Luhn observed that, when ordering the terms in a collection by their frequency, the most significant words were not the most frequent nor the most rare [Luhn, 1958]. Later work by van Rijsbergen has shown that it is possible to remove words which do not bear particulat meaning per se, like *the* or *and*, without losing significant content [van Rijsbergen, 1979]. The process of stop-word removal has the advantage of reducing the size of the index (up to 50% according to van Rijsbergen), while removing non-content-bearing material from the documents. The disadvantages include the possibility of removing terms which are important for specific queries (e.g. the famous passage by Shakespeare "to be or not to be" is entirely composed by stop-words). Additionally, some more advanced representation beyond mere terms (e.g. phrases, sentence dependencies, etc.) could not be possible if stop-words were removed.

Stemming is the process of reducing a term to its *stem* (i.e. base or root form), with the purpose of lowering the possibility of mismatch between two terms with a slightly different spelling but bearing the same meaning. For example, the terms *fishing*, *fisher* and *fishes* could all be stemmed into *fish*, so querying for the term *fishing* would also return documents about fishes and fishers (all belonging to the same conceptual "class"). A popular stemming approach is called *suffix stripping*, adopted by a number of stemmers including the widely-used Porter Stemmer [Porter, 1980].

Both the document and the query have to undergo the same pre-processing steps, otherwise a retrieval function could not match the query terms with the indexed terms.

As previously observed, weights are assigned to terms (i.e. features) during the indexing process, in order to accurately represent the documents. From the point of view of an IR system, it is

important that such weights help to discriminate between documents as much as possible, so the correct (relevant) documents are retrieved for the user.

The document representation example in Figure 2.2 is showcased simply using the number of occurrences of the terms within a document as term weights. This feature is referred to as *term frequency* (TF) and it is one of the most commonly used in IR, as it became popular through the SMART system by Salton [1971]. The TF weights assign more importance to terms which are frequent within a given document. Spärck-Jones proposed to consider also information about the discriminative power of a term across the collection, i.e. how well a term describes a document across the collection [Spärck Jones, 1972]. In other words, a term which only occurs in a few documents is highly discriminative. This characteristic of a term is reflected by its *inverse document frequency* (IDF). The IDF weights assign more importance to terms which are rare across the collection.

Different IR researchers have provided different motivations for revised/normalised versions of these traditional term weighting schemes, e.g. [Church and Gale, 1995, Roelleke, 2003, Aizawa, 2003, Robertson, 2004], so nowadays TF and IDF can be seen as families of weighting functions. Equation 2.1 and Equation 2.2 outline the definition of some of these interpretations. Figure 2.3 introduces the notation.

$$\mathrm{TF}(t,d) := \begin{cases} n_L(t,d) & \text{total term frequency} \\ \frac{n_L(t,d)}{N_L(d)} & \text{maximum-likelihood estimate} \\ \frac{n_L(t,d)}{n_L(t,d)+K_d} & \text{BM25-motivated normalisation} \end{cases} \tag{2.1}$$

$$\mathrm{IDF}(t,c) := \begin{cases} -\log_2 \frac{n_D(t,c)}{N_D(c)} & \text{traditional} \\ -\log_{N_D(c)} \frac{n_D(t,c)}{N_D(c)} & \text{normalised IDF} \\ -\log_2 \frac{n_D(t,c)}{N_D(c)-n_D(t,c)} & \text{BIR-motivated IDF} \end{cases} \tag{2.2}$$

2.1.2 Evaluation of IR Systems

As previously mentioned, the purpose of an IR system is to help users in finding documents which are relevant to their information need. The concept of *relevance* is one of the most important in

Symbol	Meaning
t, d, q, c, r	term t, document d, query q, collection c, relevant r
D_c	set of all documents $\{d_1, ...\}$ in a collection c
D_r	set of relevant documents
T_c	set of all terms $\{t_1, ...\}$ in a collection c
T_r	set of terms which occur in relevant documents
L_c	set of all locations in a collection c
L_r	set of locations in relevant documents
$n_L(t,d)$	number of locations of the term t in document d
$N_L(d)$	total number of locations in document d (document length)
$n_L(t,q)$	number of locations of the term t in query q
$N_L(q)$	total number of locations in query q (query length)
$n_L(t,c)$	number of locations of the term t in collection c
$N_L(c)$	total number of locations in collection c ($= \|L_c\|$)
$n_L(t,r)$	number of locations of the term t in the set L_r
$N_L(r)$	total number of locations in the set L_r ($= \|L_r\|$)
$n_D(t,c)$	number of documents in which the term t occurs in collection c
$N_D(c)$	total number of documents in collection c ($= \|D_c\|$)
$n_D(t,r)$	number of relevant documents in which the term t occurs
$N_D(r)$	total number of relevant documents
avgdl(c)	average document length of documents in collection c ($= N_L(c)/N_D(c)$)
pivdl(d,c)	pivoted document length; pivdl$(d,c) = N_L(d)/avgdl(c)$
K_d	normalisation factor for BM25
	also written as $K_d(c)$ if the collection c is made explicit

Figure 2.3: Notation for symbols used in IR models.

IR, intuitively well-known but often not completely understood [Mizzaro, 1997]. In this section, we utilise the broadly-accepted idea that relevance describes the usefulness of a document in fulfilling an information need. In this intuitive and informal definition, several aspects such as task or context, as well as timeliness or novelty, are not explicitly considered. A deeper discussion about relevance goes beyond the scope of this section (see e.g. [Mizzaro, 1997]).

In order to evaluate the usefulness of an IR system, different aspects can be considered. For example, a timely response and ease of use are features to examine. This section, though, is mainly concerned with measuring the *effectiveness* of a system, which is related with the aforementioned

concept of relevance.

Before defining evaluation metrics, we observe that documents can be categorised according to their state. When an IR system provides documents to the user in response to a query, such documents are classified as *retrieved*, while the remaining documents are classified as *non-retrieved*. The relevance state (i.e. being relevant or non-relevant for a query) is orthogonal to the retrieval state, so the documents can be categorised in one of the four following classes: relevant retrieved, relevant non-retrieved, non-relevant retrieved, non-relevant non-retrieved.

Effectiveness measures can be defined in terms of these states. Two most commonly adopted measures are *precision* and *recall*. Precision is defined as the portion of retrieved documents which are relevant, while recall is defined as the portion of relevant documents which are retrieved, formally:

$$Precision = \frac{|Relevant \cap Retrieved|}{|Retrieved|} \tag{2.3}$$

$$Recall = \frac{|Relevant \cap Retrieved|}{|Relevant|} \tag{2.4}$$

Precision and recall can be combined into their weighted harmonic mean, called *F-measure* and generally defined as [van Rijsbergen, 1979]:

$$F_\beta = \frac{(1+\beta^2) \cdot Precision \cdot Recall}{\beta^2 \cdot Precision + Recall} \tag{2.5}$$

The parameter β is used to adjust the relative weight between precision and recall, and can be interpreted as "recall is β times more important than precision". When $\beta = 1$, precision and recall are assigned the same important, and the F-measure is usually defined as F_1. Other options are F_2 (recall is twice more important than precision) or $F_{0.5}$ (precision is twice more important than recall).

2.2 Automatic Document Summarisation

Document Summarisation is the task of presenting a shortened version of a document, or a set of documents, containing the most important information expressed in the source. A well-crafted summary provides benefits to the users who can quickly digest information without spending a huge amount of time to read the whole source.

Professional human abstractors can produce high-quality summaries, but they often require domain-specific knowledge, and the time and cost to employ professionals could be unaffordable. Intelligent tools for summarisation are crucial in the process of information reduction.

The automatic production of high-quality summaries is not a straightforward task. Jing pointed out how the non-trivial operations performed by a human abstractor are particularly difficult to be captured by a machine [Jing, 2002]. Some of these operations include:

- *Sentence reduction*: non-essential words or phrases are removed from the sentence, without breaking the meaning of the sentence.
- *Sentence combination*: a few sentences are combined into one; this operation is often used in combination with sentence reduction.
- *Syntactic transformation*: the syntactic structure of a sentence is changed; this approach can be used in both sentence reduction and sentence combination; for example, a transitive verb might be transformed into a passive construction, or *vice-versa*.
  Example: The ball has been hit by the striker $\rightarrow$ The striker hits the ball.
- *Paraphrasing*: words or phrases are substituted by their paraphrases (e.g. synonyms).
- *Generalisation (specialisation)*: phrases are transformed into a more general (specific) description.
  Example (generalisation): He works on user-based summaries, and the evaluation of summarisation systems $\rightarrow$ He works on summarisation.
  Example (specialisation): The seminar is given by the lecturer of DCS129 Probability and Matrices $\rightarrow$ The seminar is given by Dr. Tombros.
- *Reordering*: the order of the sentence in the summary might change with respect to the original document, for example the concluding sentence of a document might be place at the beginning of the abstract.

The output of a summariser can appear verbatim in the original source (extract) or can be partially modified from the original source (abstract). The main focus of this thesis is on the extraction problem. Daumé III and Marcu [2002] suggested that document summarisation systems producing *extracts* can be categorised under one of the following three classes:

- Extractive summarisers, which perform the summarisation by extracting the most important sentences in the text
- Headline generators, which produce a short list of words, that are representative of the content of the text given as input
- Sentence simplification systems, which delete unimportant words and phrases, hence compressing long sentences

The name of the first class can be addressed as rather generic (i.e. also the other classes are *extractive* summarisers), but it is the common naming used to indicate sentence extraction tools. Using extractive summarisers, sentences are ranked according to a combination of some specific features. For example, statistical features like IDF could be combined to location-based features (e.g. the first or the last paragraph are often the most "important"). Section 2.2.3 discusses more details about sentence extraction.

Headline generators, also known as highlights generators, are a class of extractive summarisation systems facing the problem of key facts extraction. Their output is not necessarily a coherent and grammatical summary, but headlines can be as useful and informative as fluent summary for the end users. An example of key fact extraction system for news articles is proposed by Kastner and Monz [2009]. The features of the text, considered to be a good indicator of the importance of a phrase or a word, are identified through a manual investigation of a training corpus:

- Position of the sentence in the document, assuming that some facts are placed at the beginning of the document to emphasise them
- Number or dates, which might be central in news reports
- Source attribution, such as "according to a source", or "officials say"
- Negations, which are often used with the purpose of presenting contradictory information
- Causal or Temporal adverbs, a manually selection of phrases (e.g. "in order to", "for two weeks")
- News agency name, as the journalistic context of their data suggests the importance of this detail

- Bonus words, a manual selection of words boosting the importance of a sentence (e.g. “sensational”, or “historic”)
- Verb classes, two manual selections of verbs, defined as talkVerbs (e.g. “report”, or “mention”) and actionVerbs (e.g. “provoke”, or “use”) and their WordNet synonyms
- Proper nouns, which are commonly considered a good indicator of relevance

Finally, sentence simplification systems aims to shorten long sentences by deleting unimportant words. Removing stop-words is the first trivial approach, but it clearly leads to sentences that are not grammatical. A less aggressive approach consists in removing some specific class of words, like adverbs or adjectives. As an example, the sentence “Peter is a clumsy sailor” could be shorten into “Peter is a sailor”. For the specific case of sentiment analysis, removing adverbs or attributes would not yield positive results, as these classes of words can also convey opinions.

2.2.1 A Generic Model for Text Summarisation

This section outlines a generic model for text summarisation. Different summarisation systems can implement different approaches, but all the operations they perform can be grouped into three generic phases. Figure 2.4 shows the pipeline to produce a summary from a text source. The main steps to perform summarisation are described as follows:

- **Text Analysis**: the input text is broken into segments of a desired granularity, typically sentences. Desired features are extracted from the segments (e.g. word frequencies, position, cue words, etc.). The result is an internal representation of the source text.
- **Selection**: fragments are scored according to the desired features. A ranking can also be influenced by factors like novelty (or diversity) in order to prevent redundancy. The result is an internal representation of the summary.
- **Synthesis**: segments are extracted to produce the summaries. The number of segments depends on the desired output size (absolute or relative to the input size). The segments will appear verbatim, usually in the same order as they appear in the source. Revision strategies can be applied at this point to improve coherence and readability. The result is the final summary.

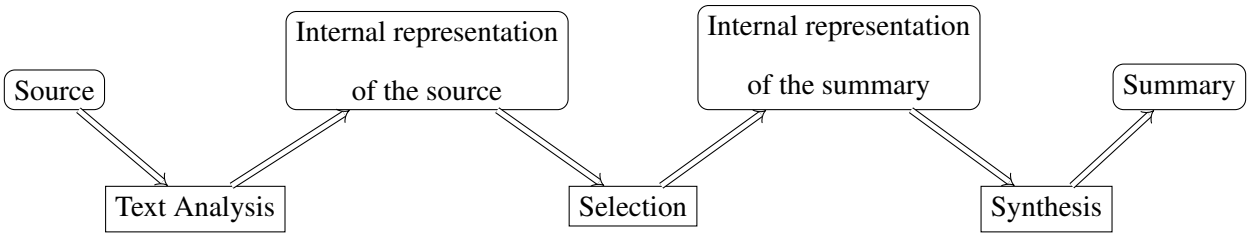


Figure 2.4: Generic architecture of a summarisation system

2.2.2 Summary Properties

There are several points of view to consider when producing a summary. Different aspects to consider for the development of a summarisation system are described hereafter.

Abstract vs. Extract

If a summary only contains verbatim material from the original text is commonly referred to as an extract. On the other side, the term abstract is used to identify a summary in which at least some material is not present verbatim in the original text.

Single vs. Multiple Sources

The source could be a single document or a set of document. Multiple sources can confirm or contradict some information. The application domain determines whether this is a problem or not. For example, news releases which contradict each other can be problematic, while user reviews which express different opinions on the same item can be completely legitimate.

Generic vs. User-Oriented

A generic summary is static, i.e. it does not change for different users. User-oriented summaries are dynamic, tailored for a specific user. A special case of user-oriented summaries are query-biased summaries, e.g. snippets in search engine result pages [Tombros and Sanderson, 1998].

Summary Function

Depending on the function, summaries can be classified into three non-exclusive categories [Maybury and Mani, 2001]:

- *Indicative*: if the summary acts as a preview indicating whether reading the whole docu-

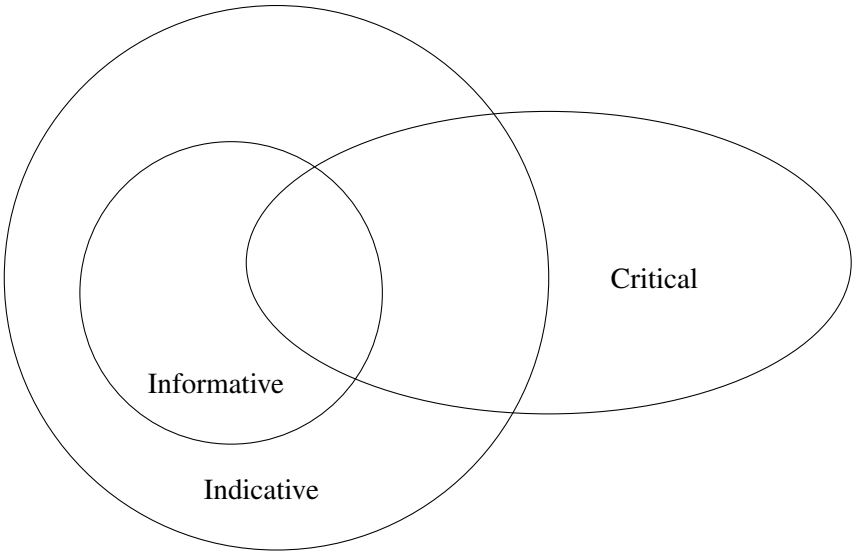


Figure 2.5: Different summary functions as suggested in [Maybury and Mani, 2001]

ment is worthwhile.

- *Informative*: if the summary covers all the salient points of the source document.
- *Critical*: if the summary adds a critical view and expresses opinions on the source document. Also known as *evaluative summary*.

The relationship between the three categories are shown in Figure 2.5.

Reduction of the Information Source

As a summary is supposed to be shorter than the original document, we can consider either a specific *target length* (e.g. a 250 words abstract), or a *compression rate*, also known as *condensation rate* or *reduction rate* (e.g. $\sim$15% of the original document).

Other Properties

For a summary to be useful, we need to consider the level of *informativeness*. This depends on two aspects: a) the fidelity to the original source (i.e. the summary should not provide information which contradicts the source); and b) the relevance for the user's interests. Another desired property for the summary is to be *coherent*. Sentences in a coherent summary are syntactically correct, and they show a global cohesion. From this point of view, the main problem for abstracts is the need to produce a grammatical summary; when considering extracts, one of the issues is the presence of *dangling anaphora*. This problem raises when a selected sentence contains a

reference to a previous one, which is not selected. A basic example is given in Figure 2.6.

Source	1. Peter is a sailor.
	2. He is a friend of Mary.
	3. They both live in London.
Summary	2. He is a friend of Mary.

Figure 2.6: Example of dangling anaphora: who is "He" in the summary?

Dangling anaphora lead to cohesion problems, i.e. the sentences are locally grammatical, but from a global point of view they sound inconsistent. Dangling anaphora resolution is the task of fixing the dangling anaphora problem. The first step consists in identifying anaphoric references. This can be done through the use of syntactical rules, e.g. looking for "he", "which", etc. In the following step we can either decide to remove the whole sentence containing the anaphor, or to include the previous sentence and checking whether it resolves the problem. The second approach can introduce further issues regarding the length of the summary, as well as new anaphoric references to resolve. More sophisticated approaches need a deeper understanding of the text, but there is not effective solution for fixing global semantic discontinuities.

2.2.3 Sentence Extraction

Traditional sentence extraction techniques apply different methods for determining the importance of sentences [Nenkova and McKeown, 2011]. Experiments in sentence extraction have been reported since 1958 by Luhn [Luhn, 1958]. In his seminal work, sentences are extracted according to their significance score. For each sentence, the significance score is calculated using statistical information from word frequency and word distribution. The rationale beyond Luhn's approach was driven by the simple idea that some words in a document are descriptive of its content. By using frequencies to identify descriptive words, Luhn observed that extremely common words do not describe the content of a document. This observation led to the concept of stop-word removal as described in Section 2.1. It is worth noting that common stop-words are not the only non-descriptive words: other frequent words in specific domains (e.g. the word *sport* in a collection of documents about sport) as well as low-frequency words are also non-descriptive. Luhn applied arbitrary thresholds to remove extremely frequent and extremely rare words. Luhn's observations are reflected in many of nowadays summarisation systems, which apply the same idea

about identifying descriptive words, by using term weights such as TF-IDF, rather than raw frequencies, in order to overcome the need for arbitrary thresholds [Nenkova and McKeown, 2011].

Another statistical tool based on word frequencies is the application of the log-likelihood ratio (LLR) test [Dunning, 1993] for the identification of highly descriptive words. Such words are commonly referred to as topic signatures [Lin and Hovy, 2000]. The main difference between the use of TF-IDF weights and the LLR test is the fact that the LLR test automatically provides a cut-off threshold to determine whether a word is descriptive or not. Lin and Hovy introduced the use of the LLR test to identify topic signatures within the context of single-document summarisation [Lin and Hovy, 2000] but the method has been applied successfully also in the context of multi-document summarisation, in particular in the news domain [Conroy et al., 2004, Finley et al., 2004, Conroy et al., 2006].

Edmundson investigated the use of different features, combined with word frequencies, for identifying significant sentences [Edmundson, 1969]. In particular, the features he included were cue word presence and structural information such as title word presence as well as sentence location. Cue words are terms which do not explicitly describe the content of a sentence by themselves, but are good indicators of its importance. Title words are terms in the document which also appear in the main title or in one of the paragraph headers. The assumption is that the author of a document would prefer to include in the title words which are representative of the content, so their presence in a sentence is a good evidence of its importance. Finally, the location of a sentence is also an indication of its importance. This can be due to cultural peculiarities, like e.g. the traditional British editorial style, which suggests to concentrate the important aspects of an article at the beginning of the document. In this way, parts of the article can be cut from the middle or at the end during later revisions, in order to fit the article in the desired template for print, without compromising the message. The scores for the individual features are computed for all the sentences, and then combined using a weighted linear combination to select the sentences to extract.

Instead of using subjective weights to combine features, a different approach proposed by Kupiec et al. [1995] consists in translating the summarisation task into a sentence classification task. Using a corpus composed by documents and summaries, Kupiec et al. trained a Naive Bayes classifier in order to determine whether a sentence has to be selected to generate the summary. The features they observed were sentence length (i.e. short sentences are unlikely to be

important), fixed-phrase presence (similar to the cue words used by Edmundson), paragraph location, thematic word presence (based on word frequencies) and uppercase word presence (e.g. proper names or acronyms).

2.2.4 Application Scenarios

The task of automatic document summarisation can be applied in several contexts. We suggest a non-exhaustive list of traditional application scenarios, mainly inspired by [Maybury and Mani, 2001]:

- Multimedia news summaries (e.g. headlines generation).
- Physicians' aids: summarise and compare the recommended treatments for a patient.
- Meeting aid: e.g. what happened at the last week's meeting.
- Search engine result pages: e.g. snippet generation.
- Intelligence gathering: create a 500-word biography of a police suspect
- Hand-held devices: create a screen-sized summary of news or e-mails.

More recent trends have seen the development of interest towards other application scenarios. In particular:

- Opinion mining: e.g. what the users say about a new product.
- Microblog summarisation: e.g. summarisation of short web comments on a given topic.

The expansion of Web 2.0 services have seen the increase of user-generated content. Popular web sites like Twitter[1] and Facebook [2] are driven by user-generated content. Users can discuss topics and express opinions, usually in the form of short comments. Many other web sites offer the possibility of reviewing products[3], movies[4] or hotels[5]. Commonly, web users write short comments on a specific topic, rather than long and detailed reviews. The language used is typically

[1] http://www.twitter.com (Accessed December 2014)
[2] http://www.facebook.com (Accessed December 2014)
[3] http://www.amazon.com (Accessed December 2014)
[4] http://www.rottentomatoes.com (Accessed December 2014)
[5] http://www.booking.com (Accessed December 2014)

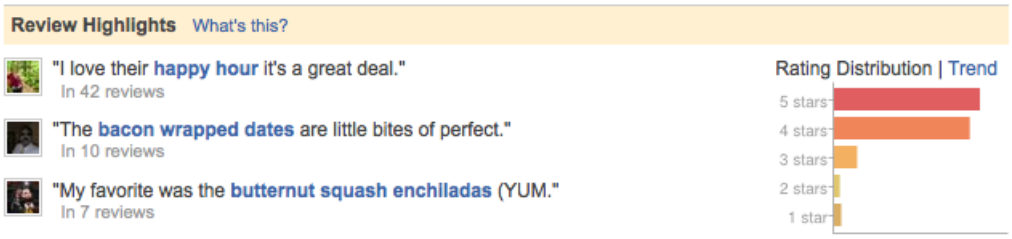


Figure 2.7: Example of restaurant review from Yelp (Accessed December 2014). The phrase "happy hour" is the most discussed among different reviews, and hence selected first to form the review highlights.

informal, and it often contains typos, acronyms, emoticons or some particular jargon. Previous investigations in microblog summarisation [Sharifi et al., 2010] have also shown that different users tend to employ similar words when describing a particular topic. From this point of view, frequent phrases can be used to extract individual sentences which are good representatives of the topic itself. As an example, Figure 2.7 shows an extractive summary of restaurant reviews on Yelp. The choice of the sentences to include in the summary is based on phrase frequencies. In particular, the summary shows how the highlighted phrases are frequently discussed among different reviewers, and hence are good candidate to form the review highlights.

2.2.5 Personalised Summarisation

With the idea of maximising the density of relevant sentences, summaries can be built in a dynamic way, tailoring the summary to the user's interests. A user-model can be used to generate personalised summaries [Diaz and Gervas, 2007]. Query-biased summaries are a particular type of user-oriented dynamic summaries [Tombros and Sanderson, 1998]. In search engine result pages, each retrieval result is paired with a query-biased summary. Such a snippet gives the user an indication of why a particular result is relevant for the submitted query. When asking users to identify relevant documents, Tombros and Sanderson have shown how the query-biased summaries significantly improve both accuracy and speed of user relevance judgement compared to static predefined summaries.

2.2.6 Evaluation of Summarisation Systems

The evaluation of summarisation systems can be performed through different approaches:

- Intrinsic system-based evaluation
- Intrinsic subjective evaluation
- Task-based evaluation

All the approaches can have disadvantages. Intrinsic approaches evaluate the summary per se, i.e. aiming at answering the question "*how good is this summary?*". On the other side, task-based evaluation considers the summary in the context of supporting the user to perform a given task, i.e. it tries to answer the question "*how useful is this summary to perform this task?*". Moreover, intrinsic evaluation is partitioned into system-based and subjective evaluation. Intrinsic system-based evaluation requires gold standard summaries in order to compare the output of a system with such gold standards. Obtaining gold standard summaries can be an expensive procedure. Additionally, low agreement between different judges is sometimes observed: humans do not agree on what should be in a summary, so multiple gold standards for the same summary are required to avoid bias towards a specific judge. In case of dynamic summaries, as in query-biased summarisation or personalised summarisation, an intrinsic system-based evaluation is not viable, because a gold standard (e.g. a notion of "best" summary) cannot be easily defined outside the scope of the user preferences or task. Subjective evaluation requires human judgements about the quality of the summaries. This is often achieved with the use of questionnaires which ask users about different aspects of the summaries, such as readability, usefulness, clarity, cohesion, etc. Appropriate questionnaires and tasks can be hard to design and the process can be particularly time-consuming. For these reasons, they are usually not suitable for comparing systems during the development process. Careful experimental design is needed to establish the validity of the tests.

Intrinsic System-based Evaluation

The intrinsic approach is a system-based approach, focused on the comparison of the machine-produced summary to a gold standard, which is produced by a human summariser (e.g. a professional abstractor). A first issue concerns the availability of such gold standards, as the effort needed by human professionals to craft them can be very expensive.

The most commonly used evaluation tool for summarisation is ROUGE [Lin, 2004b]. ROUGE stands for Recall-Oriented Understudy for Gisting Evaluation. It includes five different measures

for automatically evaluating the quality of a summary compared to an ideal human-produced summary (or to a set of gold standards). The five evaluation measures, between a candidate summary X and a gold standard Y, are defined as follow:

- ROUGE-N, a recall-oriented measure, based on n-grams co-occurrence statistics. This measure gives the ratio between the number of matching n-grams and the total number of n-grams in the gold summary. X and Y are defined as a sequence of n-grams.

$$\text{ROUGE-N}(X,Y) = \frac{|X \cap Y|}{|Y|} \tag{2.6}$$

  where N is the size of the n-gram g. Typically, ROUGE-1 and ROUGE-2 are reported. These measures are based on unigrams and bigrams, respectively. Lin has shown that in particular ROUGE-1 correlates well with human judgement [Lin, 2004a]. While the original definition as in [Lin, 2004b] describes ROUGE-N as recall-oriented, the software package itself provides also a precision-oriented measure (where the denominator is $|X|$) and the respective F-score.

- ROUGE-L, based on LCS (Longest Common Subsequence) [Cormen et al., 2001]. The candidate summary and the gold summary are represented as sequences of terms (i.e. unigrams), and their LCS is the longest subsequence common to both summaries. The LCS function used in the following equations returns the length of the LCS between the candidate summary and the gold summary. Such length is normalised over the two summary lengths, to obtain precision-and-recall-like measures, and hence the F-measure

$$P_{LCS} = \frac{LCS(X,Y)}{|X|} \tag{2.7}$$

$$R_{LCS} = \frac{LCS(X,Y)}{|Y|} \tag{2.8}$$

$$\text{ROUGE-L}(X,Y) = \frac{(1+\beta^2)R_{LCS}P_{LCS}}{R_{LCS}+\beta^2 P_{LCS}} \tag{2.9}$$

- ROUGE-W, a different version of ROUGE-L, based on a Weighted LCS. The weights are calculated to reflect the number of consecutive matches in the LCS.

- ROUGE-S, based on skip-bigram co-occurrence statistics. A Skip-bigram is a bi-gram which allows for an arbitrary number of other tokens to be in between its two terms. The formulas to calculate precision and recall for ROUGE-S are the same as ROUGE-N, with the only difference in the tokens being counted.

- ROUGE-SU, an extension of ROUGE-S, which combines the count of skip-bigrams and with the count of unigrams in order to reduce the sensitivity to word order. Shared evaluation tracks such as DUC[6] and TAC[7] often report ROUGE-SU4.

Other automatic approaches for summary evaluation include the use of readability metrics. Such metrics are a measure of linguistic quality as they describe a piece of text in terms of how easy it is to read. Some of the options to calculate readability include Flesch Reading Ease (FRE) index, Coleman-Liau grade level (CLGL), and SMOG index[8]. Although these metrics are not tailored for summarisation, and they do not describe whether the summary covers the topic of the original source adequately, they have been used in the context of summarisation and document simplification, e.g. by Leveling and Jones [2012].

Intrinsic Subjective Evaluation

Human judges can be involved in the evaluation process for subjective evaluation. Different types of user-oriented studies can be set up. The users/judges can be given a questionnaire where they provide a subjective assessment on the summary, e.g. to describe how coherent, fluent, or well-written the user finds the summary. Other options include the possibility of asking the user to re-create the original source from the summary, to show how well the summary covers the topic.

A more structured approach to human-based assessment is Pyramid [Nenkova et al., 2007, Nenkova and Passonneau, 2004]. The Pyramid framework uses multiple model summaries which are manually analysed by human judges. Similar sentences from different model summaries are grouped into a Summary Content Unit (SCU). Each SCU is assigned to a specific level of the pyramid. Each level is assigned a label from 1 (lowest level) to n (highest level). The highest level correspond to the total number of model summaries available. The level of a specific SCU depends on the number of model summaries it can be found in. For example, if a SCU is found in 3 out 4 model summaries, it will be placed on the third level of the pyramid. The system-produced summaries are then manually assessed against the SCUs. Good summaries contain a large number of higher-level SCUs, while summaries with many lower-level SCUs are considered to be less informative. Pyramid overcomes some of the limitations of automatic assessment, in particular it can capture rephrasing and paraphrasing, because two synonyms are

[6]Document Understanding Conference - http://duc.nist.gov/ (Accessed December 2014)
[7]Text Analysis Conference - http://www.nist.gov/tac (Accessed December 2014)
[8]http://www.readabilityformulas.com (Accessed December 2014)

likely to be considered with the same SCU (while ROUGE would not count them as overlapping terms). On the other side, it still requires gold standard summaries and it also require more human labour for the assessment. Louis and Nenkova reported high correlation between Pyramid and ROUGE scores [Louis and Nenkova, 2008], suggesting ROUGE as a lower-cost option to obtain results similar to subjective evaluation.

Task-based Evaluation

Task-based evaluation involves a specific task which has to be completed by the user (or by a system). In task-based evaluation, a particular dimension, or more than one, is chosen to observe the usefulness of a summary, e.g. the time needed for completing the task, or the accuracy in reaching the target. While subjective evaluation aims at measuring the satisfaction of users towards a particular system, task-based evaluation aims at observing efficiency and/or effectiveness of such system.

2.3 Sentiment Analysis and Opinion Mining

Sentiment Analysis (also known as Opinion Mining, or Opinion Extraction) is the study of sentiments, opinions and emotions expressed in text, by means of a computer software [Liu, 2010]. Products, companies, people or anything else are all topics we can express opinions about. Opinions can be expressed by professional reviewers, for example on a specialised blog, or by generic users, for example as a short comment on a discussion forum or through a social network. Opinion-oriented text inherits the intrinsic ambiguity of natural language. A single review might express an overall positive opinion on a product, even though several negative aspects could be reported. This ambiguity is made more challenging by the informal setting of on-line social media and by some specific domain like movie reviews [Zhuang et al., 2006].

Figure 2.8 shows some examples of web comments taken from Rotten Tomatoes. These short comments are pretty clear for a human reader who has some knowledge of the domain (movies and actors), but a machine could fail in understanding the irony or sarcasm beyond some of them. For example, in line 2 a direct citation from the movie is reported, indicating that the reviewer has particularly appreciated a specific scene. In line 3, we can observe a similar situation with one of the soundtrack songs. In line 4, one of the actor is heavily criticised without an explicit review on his performance. Lines 5 and 6 use again irony referring to the catchy aspects of the

#	Movie	Comment
1	*The Godfather*	The best there is, the best there was, and the best there ever will be
2	*The Goodfellas*	Everyone takes a beating sometime
3	*Fight Club*	I'm still singing "where is my mind" in the shower :)
4	*The Hangover*	Zach Galifianakis... The only thing more painful then watching a movie you're in is trying to spell your name
5	*The King's Speech*	The Kings sp..spe..spee...speech is a g..g..g..go..good movie
6	*Thor*	I didn't realize how much you could do with a hammer

Figure 2.8: Examples of user-generated reviews from Rotten Tomatoes.

main characters in the respective movies. The only comment with an explicit opinion is the first one, although the same comment has no meaning if taken outside of its context (i.e. "the best" is not associated with the word "movie").

In the context of sentiment analysis, several terms, such as opinion, sentiment, emotion, attitude or feeling, are often used with similar meanings. The following sections aim at clarifying the terminology.

2.3.1 Sentiment, Opinion and Polarity

In general, an **opinion** involves personal judgement or appraisal, or simply a view on a specific topic or object. An opinion can be seen as an expression of recommendation (e.g. to buy a product) or support (e.g. to a governmental decision). In these terms, an opinion can be positive or negative, weak or strong, and can also be neutral if no recommendation is expressed. The feeling expressed by an opinion is called **sentiment**.

For example, the sentences "I like this restaurant" and "I love this restaurant", from two hypothetical restaurant reviews, are both giving positive sentiments, but the second one is stronger. Moreover, often review web sites allow users to summarise their point of view into a rating scale, for example a "stars system", such that a "5 stars" vote and a "4 stars" one are both expressing positive sentiments, but with the first one being more emphasised. Different sentiment scales can be normalised, without lack of generality, into the [-1; +1] range of real numbers. We can consider the sentiment $s \in [-1;+1], s \in \mathbb{R}$ on the Sentiment Scale as in Figure 2.9, with the following

meaning:

- $s = 0$ represents a neutral sentiment
- $s \in (0; +1]$ represents a positive recommendation (i.e. like)
- $s \in [-1; 0)$ represents a negative recommendation (i.e. dislike)
- if s is undefined/unknown, no recommendation has been expressed[9]

The two extremes -1 and +1 represent the strongest negative and positive sentiments, respectively, and can be associated to the two cognitive states *hate* and *love*. The sign of the sentiment s is also called **polarity**. In other words, the polarity can be positive or negative, but it does not provide any information about the strength of the sentiment.

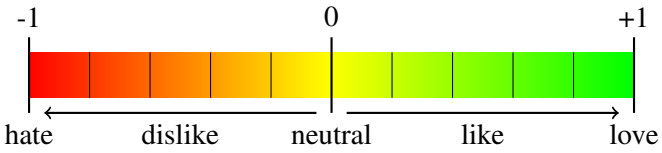


Figure 2.9: Sentiment Scale, employing a commonly used colour scheme to indicate polarity: green for positive and red for negative.

2.3.2 Subjectivity and Objectivity

An opinion can be backed up by personal experience or factual knowledge. Sentences expressing factual information are referred to as **objective**. On the other side, sentences expressing a personal view or belief are called **subjective**. Previous research has focused mainly on subjectivity detection at a sentence level [Pang and Lee, 2004, Bonzanini et al., 2012], because subjective sentences and opinionated sentences (i.e. sentences suggesting a positive or negative sentiment) are two strongly related, but still different notions. Other work has also proposed ways to exploit objective sentences, looking for desirable and undesirable facts [Zhang and Liu, 2011]. The following examples can help to distinguish the two types of sentences:

- "I think it is 6 o'clock" is a subjective sentence which shows a personal belief, but no sentiment is expressed.

[9] We explicitly distinguish between neutral sentiment and no expression of sentiment

- "I am in love with my new car" is a subjective sentence expressing a positive sentiment.
- "Our new pasta machine broke down during the first use" is an objective sentence, as it describes a fact, but it is also carrying a negative sentiment towards the product, because the fact itself was not desired.
- "I bought this GPS navigator for my father" is an objective sentence, which does not express any sentiment.

2.3.3 Explicit vs. Implicit

We have already provided examples of opinionated text whose orientation is easy to understand for a human reader, but hard to classify for a machine. Informal language, irony, or metaphors are all contributing to the ambiguity of the language. In a subjective sentence, the polarity of the opinion might be explicitly clear, or it might be implicitly given by the context. For example, the sentence "This is a great movie" does not leave any doubt about the object under analysis and the positive opinion expressed about it, thanks to the use of the term "great". On the other hand, if we say that "Everybody should watch this movie", we are not explicitly giving a positive opinion on the movie, although we are implicitly recommending it.

The explicit-vs-implicit dualism can also be referred to specific features. In some cases, the object feature under analysis is explicitly mentioned, as in the sentence "Brando's performance is outstanding". In other cases, the feature is given by the context or by other indicators, as in the sentence "This phone is too large". In the last example, the feature under analysis is the size of the phone, which is not mentioned anywhere. The term "large" acts as a feature indicator. Other adjectives might be general (e.g. "good" or "bad") and hence not mappable to a specific feature.

2.3.4 Emotions

Another notion, strongly connected with sentiment, is the concept of **emotion**. Emotions are subjective and internal feelings, such as happiness and sadness. Extensive research on this topic has been carried out in several fields, including for instance sociology and psychology, and many frameworks for emotion categorisation have been proposed, but there is no general agreement on a set of basic emotions. It is worth noting that emotions can lead to the generation of opinions, and a distinction between rational evaluation (e.g. a positive sentiment given by a quality/price

balance) and emotional evaluation (e.g. a positive sentiment given by an emotional connection with the brand) has been proposed [Liu, 2012].

2.3.5 Sentiment Analysis Tasks

The most explored task in Sentiment Analysis is sentiment classification. Given a piece of text (a document, a sentence, a tweet, etc.), the purpose of sentiment classification is to classify such a text according to its polarity. Several other tasks need anyway to be tackled by Sentiment Analysis applications. For example, entity extraction is required when a document may contain references to multiple entities, and aspect extraction is also required in order to capture different opinions on different aspects (i.e. attributes) of an entity. In most applications, multiple opinions from different people are analysed. Opinion summarisation can be applied to provide a concise view over a set of opinions.

The following paragraphs provide an overview on the main sentiment analysis tasks.

2.3.6 Sentiment Classification

One of the main tasks in Sentiment Analysis is sentiment classification (also called polarity classification), i.e. the classification of a document according to its overall sentiment. Early work in this area includes the use of traditional machine learning techniques, as shown by Pang and Lee [2002]. In particular, Pang and Lee employed traditional machine learning techniques, such as Naive Bayes, Maximum Entropy and Support Vector Machine, to classify movie reviews according to their overall polarity. Their work has shown that simple features like unigram presence can be very effective for this task. Other approaches, based on unsupervised techniques, have also been applied to this task by Turney [2002] . Specifically, Pointwise Mutual Information (PMI), defined in Equation 2.10, has been used to calculate the semantic orientation of a phrase, measuring the similarity of the phrase itself and the terms “excellent” and “poor”, as shown in Equation 2.11.

$$\mathrm{PMI}(t_1, t_2) := \log\left(\frac{P(t_1 \cap t_2)}{P(t_1)P(t_2)}\right) \tag{2.10}$$

$$\mathrm{SO}(t) := \mathrm{PMI}(t, \text{“excellent”}) - \mathrm{PMI}(t, \text{“poor”}) \tag{2.11}$$

Other work has tackled the polarity classification problem as a regression task. In other words, the focus is not only on binary polarity, but also in identifying different degrees of polarity. This is the case of reviews which adopt a “star-rating” system, i.e. a review can be classified, for example, using a 1-5 star system, where 1 star indicates a strongly negative opinion, and 5 stars indicate a strongly positive opinion [Pang and Lee, 2005].

2.3.7 Subjectivity Detection

An important intuition behind sentiment classification is that not all the sentences in a review are expressing opinions [Liu, 2010]. For example, reviewers could include a brief overview of the plot when they are commenting on a movie, or a short excerpt on their experience when they are commenting on a digital product. Identifying the subjective sentences of a review, and filtering out the objective ones, has been shown to be an effective approach for sentiment classification [Pang and Lee, 2004].

OpinionFinder is a popular system performing subjectivity analysis [Wilson et al., 2005a]. It has been developed to support other applications providing information about subjectivity at a sentence level. It can be successfully integrated into other systems to perform different tasks, for example opinion retrieval. The use of sentence-level evidence has been shown to improve the performance of an opinion retrieval system, which combines relevance and polarity to retrieve blog posts [Chenlo and Losada, 2011].

2.3.8 Feature-based Analysis

Reviews often contain a mixture of different opinions about the same object. This is explained by the fact that a reviewer might want to comment on different aspects of the same object, even expressing contrasting opinions. For example, an actor could play a brilliant performance in a movie, but the overall recommendation is negative because of a boring plot. Classifying documents according to the overall polarity does not provide this information [Liu, 2010].

Previous work on digital product reviews involved the use of natural language processing tech-

niques such as part-of-speech tagging [Hu and Liu, 2004b, Hu and Liu, 2004a]. Nouns and noun phrases are identified as product features, and frequent features can be associated to opinion words. As a result, sentences can be labelled according to the feature they describe as well as the polarity. Sentences from multiple documents about the same feature can be aggregated. Ranking the features according to their frequency allows to generate multi-document feature-based summaries, simply selecting sentences about the most frequent features.

Similar work in the movie domain has suggested the application of a multi-knowledge based approach which integrates WordNet, statistical analysis and movie knowledge [Zhuang et al., 2006]. Identifying feature-opinion pairs, sentences can again be organised to form feature-based summaries.

A sentiment summariser for local services has been proposed in [Blair-Goldensohn et al., 2008]. The approach is related to the previous ones, but it combines both dynamic aspects, where no previous knowledge is assumed, and static aspects, where domain-specific knowledge can be exploited. Similarly, the aggregation of per-aspect sentiments can lead to the generation of aspect-based summaries.

A different line of work involves the use of models to jointly predict the latent topical facets of documents and their respective sentiments. Different approaches have studied extensions of a Hidden Markov Model (HMM) structure [Mei et al., 2007] and extensions of Latent Dirichlet Allocation (LDA) [Titov and McDonald, 2008, Lin et al., 2012]

2.3.9 Evaluation of Sentiment Analysis Classifiers

Viewing sentiment analysis as a classification problem, evaluation measures like precision, recall and F-measure (described in Section 2.1.2) are typically used in this context. Different experiments in sentiment analysis have proposed different classification methods to reflect the users' behaviour. In particular, the main difference consists in using either a thumbs-up/thumbs-down system (i.e. two classes, one positive and one negative) or a "star-system" (i.e. N classes, to differentiate between several levels of positive/negative feelings). Another difference consists in deciding whether to classify the document as an individual unit, or to mine specific feature/opinion pairs like proposed by Zhuang et al. [2006]. In the second case, precision and recall can be defined in terms of correctly mined feature/opinion pairs. Specifically, precision is the proportion of mined pairs which are correct, while recall is the proportion of correct pairs which are mined.

Formally, the equations from Section 2.1.2 can be rewritten as:

$$Precision = \frac{|Correct| \cap |Mined|}{|Mined|} \tag{2.12}$$

$$Recall = \frac{|Correct| \cap |Mined|}{|Correct|} \tag{2.13}$$

Experiments in feature/opinion pairs mining, like the one proposed by Zhuang et al., have employed different human annotators to assign classes to feature words and opinion words. When a feature/opinion pair is given the same class by three out of four people, it is saved as the ground-truth result, as statistical investigations have shown that the consistency achieved by three people is more than 80%. This threashold is used to avoid the cost of employing additional human annotators, whose contribution would not add more value to the ground-truth [Zhuang et al., 2006].

2.4 Summarisation of Opinion-oriented Content

Sentiment Summarisation, or Opinion Summarisation, is a summarisation task tailored to opinion-oriented content. Previous work in summarisation of opinion-oriented documents has been focusing on different variations of this task. Broadly speaking, the goal of a sentiment summariser is to produce a summary which bears the same opinion(s) of its source. The source is typically a document, or a set of documents, discussing a specific target (e.g. a review on a movie). An intrinsic sentiment-based summary should hence be a short paragraph, a single sentence or a set of keywords, providing the overall sentiment expressed in the source. Other variations on this task include aspect-based summarisation (e.g. creating a sentiment-oriented summary on the main aspects of the target, like soundtrack and special effects of a movie) and contrastive summarisation (e.g. providing a summary of "pros and cons" of a product).

The idea of a single sentence extraction, to determine the polarity of the whole document, has been suggested by Beineke et al. [2004], although results on the polarity classification task have not been reported in their study. The exploratory data analysis has shown that a number of features can be indicative of whether a particular sentence could be picked as a good summary candidate. In particular, the location of the sentence and the word choice are important features to consider. Specifically, a candidate summary sentence is more likely placed either at the beginning or at the end of a review rather than in the middle. Moreover, terms like *movie*, *film* or the title of

the movie itself are also strong indicators of a candidate summary sentence. The word choice is strongly influenced by the data domain.

Dealing with short web comments, an approach for extracting the top sentiment keywords and for showing them in a tag cloud, has been proposed by Potthast and Becker [2010]. Their technique is based on the use two dictionaries of terms V^+ and V^-, commonly used to express positive and negative opinions, which are extended with words sharing the same semantic orientation. The semantic orientation of an unknown word w is measured by the degree of association with the words in the dictionaries:

$$\mathrm{SO}(w) = \sum_{w^+ \in V^+} \mathrm{assoc}(w, w^+) - \sum_{w^- \in V^-} \mathrm{assoc}(w, w^-) \tag{2.14}$$

If the value of the semantic orientation exceeds a given threshold ε ($-\varepsilon$), the word will be added to the dictionary of positive (negative) terms. The association function used in their experiments is Pointwise Mutual Information as described by Turney and Littman [2003]. Potthast and Becker showcased their approach implementing *OpinionCloud*, a browser add-on which identifies positive and negative terms from web comments related to a YouTube[10] video or a Flickr[11] image, and shows them in a tag-cloud. However, their study did not report quantitative results on polarity classification or sentiment summarisation.

2.4.1 Intrinsic Sentiment Summarisation

Intrinsic Sentiment Summarisation produces summaries which describe the main sentiment discussed in the source. For example, if a set of reviews about a particular camera mainly discuss how bad its battery life is, an intrinsic summary could be given by the single sentence: *The battery life is too short*.

This task is assessed with an intrinsic evaluation, i.e. the system-generated summaries are compared against gold standard (human-generated) summaries. While the idea of describing the main sentiment sounds intuitively clear, there are details that can influence the generation of gold standard summaries, and hence the evaluation. For example, one distinction can be made on whether the main opinion is the one expressed on the main target as a whole (e.g. a camera), or the one expressed on the most discussed aspect of the target (e.g. the battery life of a camera). A second

[10] http://www.youtube.com (Accessed December 2014)
[11] http://www.flickr.com (Accessed December 2014)

distinction can regard the presence of multiple aspects discussed in the summary: does the summary contain a single opinion on a single aspect/target, or does the summary discuss multiple aspects? Moreover, if multiple aspects are discussed, does the summary contrast positive and negative aspects in a balanced way? Different studies have approached the generation of gold standards in distinct ways. At present, a single, well-established, benchmark is not commonly adopted and the problem of intrinsic sentiment summarisation has not been widely studied.

Ganesan et al. [2010] proposed Opinosis, an approach for abstractive summarisation applied to opinion-oriented data. Their technique employs a graph data structure to represent natural language, where each node is a word unit. In this way, the abstractive summarisation problem is translated into a graph path finding problem. Opinosis has been shown to be effective on highly redundant opinions, i.e. it has been used to summarise sets of sentences about a very specific topic, for example the battery life of an iPod, or the voice quality of a Garmin GPS system. This approach can however be regarded as domain independent, as it is designed to capture redundancy, but it does not show properties or behaviours specific to opinion-oriented data.

Later work by Ganesan et al. [2012] on intrinsic sentiment summarisation has been focusing on "micropinion generation". The term *micropinion* [sic] is used to indicate an extremely concise summary of opinions, e.g. composed by up to 10 words. The generated summaries are not only concise, but also representative and readable. Representativeness is achieved using a modified function for mutual information, while readability is achieved exploiting an n-gram language model. Their approach has been shown to be effective when compared to a number of state-of-the-art baselines, including the aforementioned Opinosis and different keyword extraction systems.

Di Fabbizio et al. [2011] introduced Starlet, an approach for multi-document summarisation of evaluative text which introduces rating distribution as a summarisation feature. Other features include n-grams and part-of-speech tags. Starlet was tested in the restaurant review domain, manually including specific aspects (e.g. food or atmosphere) to be mentioned in the summaries. The KL-divergence between the target aspect rating distribution and the predicted rating distribution was used to optimise the feature selection with respect to ROUGE scores. The assumption is that content which better mimics the rating distribution also represents the sentiments expressed in the summary. Their experiments showed the effectiveness of Starlet in the restaurant domain, both quantitatively (i.e. ROUGE scores) and qualitatively (e.g. readability, coherence, etc.).

2.4.2 Aspect-based Summarisation

Feature-based Sentiment Analysis, as described in Section 2.3.8, can easily lead to the generation feature-based summaries. Once the key features, or aspects, are identified for a particular class of objects, one sentence per feature can be selected to form the summary. The sentence selection can be based on phrase frequency.

Summaries created in this way are usually displayed in a table-like fashion. An example of a product-related summary is shown in Figure 2.10. This is different from intrinsic summarisation because the generated summaries often are not coherent, although they can be useful for the final user because they provide a bird's eye view on a specific target. Most of the work in this area has focused on aspect identification and polarity classification rather than summarisation per se, for example [Hu and Liu, 2004b, Hu and Liu, 2004a, Zhuang et al., 2006, Blair-Goldensohn et al., 2008].

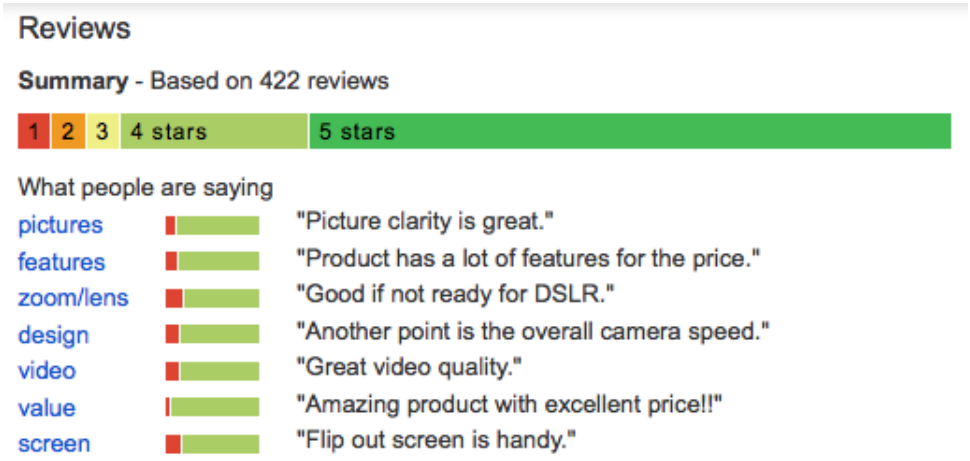


Figure 2.10: Example of product review summary (from Google Product).

2.4.3 Contrastive Summarisation

Most sentiment analysis applications deal with contrastive data, when they consider opinions coming from different opinion holders. For example, two different users can have opposite feelings about the same movie, and hence they can write reviews which convey contrastive (opposite) polarities.

Most of the work in sentiment summarisation has been focusing on summarising the main opinion about a target, where "main" can be used as a synonym for "most supported". Contrastive

summarisation aims instead at providing equal representation for different (contrastive) viewpoints, in a fashion which resembles the *par condicio*[12] concept to guarantee an equal amount of media exposure to different political parties. For example, a contrastive summariser for products could provide three positive and three negative aspects about a given mobile phone.

Research on contrastive summarisation has been anticipated and inspired by previous work on *contradiction detection* [Harabagiu et al., 2006, De Marneffe et al., 2008]. Generally speaking, two sentences (or two facts) are contradicting if they are unlikely to be both true at the same time. For particular multi-document summarisation applications, like real-time news summarisation, contradictions can be an issue because they can lead to inconsistencies in the summary. On the other side, in sentiment summarisation, contradictions naturally happen because of the subjective nature of the content to summarise.

Kim and Zhai [2009]firstly introduced the problem of Contrastive Opinion Summarization. Their work frames the problem as an optimisation problem, and their solution is based on the use of two similarity measures between pairs of sentences. The first measures content similarity, within the same group of opinions (either both positive or both negative). In this way, good representative sentences are identified for each group. The second measures cross-group contrastive similarity, i.e. it measures the similarity of two sentences with opposite polarity (excluding their difference in sentiment). The generated summaries are informative, and they can help users to digest contradictory opinions effectively.

The same task has been tackled by Paul et al. [2010], who proposed a two-stage approach for this problem. In the first stage, they use an extension of LDA to jointly model and extract topics (i.e. aspects) and viewpoints (i.e. polarities). This is motivated by the observation that some words would provide a specific polarity only in a specific context. For example, the word *unpredictable* would be considered positive if associated to a movie plot, and negative if associated to a steering wheel. In order to capture viewpoints, they used bag-of-words and dependency relations as features. In the second stage, they proposed an extension of the LexRank algorithm [Erkan and Radev, 2004], coined *Comparative LexRank*.

A different view on contrastive summarisation has been proposed by Lerman and McDonald [2009], who framed the task as a comparison of two different entities. In particular, they compared the usefulness of contrastive summaries generated individually or jointly. Individual

[12] In Latin: literally “same condition”.

summaries are generated by minimising the divergence between the full text and the candidate summary, i.e. by maximising:

$$\text{score}(S) = -\,\text{divergence}_{KL}(P_T, P_S), \tag{2.15}$$

where T represents the full text (concatenation of reviews for a single product), $S \in T$ represents a candidate summary, and P_T and P_S are they respective probability distributions. When a user is comparing two products (i.e. two summaries), if the two summaries are individually generated as above, they might contain information about completely different aspects, and hence they might not be very useful. Lerman and McDonald have shown that summary pairs generated with a joint model will contain more common aspects for the users to compare, and hence will be more useful. For two products x and y, the jointly-generated summary pair is the one which maximises:

$$\begin{aligned} \text{score}(S_x, S_y) = &-\,\text{divergence}_{KL}(T_x, S_x) \\ &-\,\text{divergence}_{KL}(T_y, S_y) \\ &+\,\text{divergence}_{KL}(T_x, S_y) \\ &+\,\text{divergence}_{KL}(T_y, S_x). \end{aligned} \tag{2.16}$$

In this way, summaries with a low divergence with their respective product and high divergence with the other product are rewarded. Aspects which are highly frequent for one product but not for the other are discarded in favour of common aspects, thus the product comparison is easier for the final user.

2.5 Knowledge Representation

Knowledge Representation is the area of Artificial Intelligence (AI) concerned with how to symbolically represent knowledge and how to automatically manipulate it by means of computer programs [Brachman and Levesque, 2004]. One of the most widely used tool in knowledge representation is the Entity-Relationship Model (ERM) and its object-oriented representation, the Enhanced ERM (E-ERM).

2.5.1 Enhanced Entity-Relationship Modelling and Fuzzy Databases

Traditional ER models include only basic concepts such as entities, relationships and attributes. Additional semantic modelling concepts have been added to the traditional models, pushed by the need for advanced applications, which the basic concepts are insufficiently able to represent.

One of the main advances in ER modelling is the introduction of inheritance, also called *is-a* relationship [Connolly and Begg, 2005]. In a hierarchy-based representation, some entity types (superclasses) include distinct groupings of its occurrences (subclasses), which all need to be represented in the data model. For example, in a hypothetical airline scenario, `Pilot`, `Cabin Crew` and `Clerk` are all subclasses of the entity `Staff Member`. All staff members can share some attributes (e.g. basic salary and hiring date), but the individual subclasses can be characterised by additional attributes (e.g. for Pilots, date of fit-for-flying test) and can be involved in different relationships (e.g. only Clerks deal with bookings). The process of identifying superclasses and subclasses can be approached in different directions. Specialisation, a top-down approach, consists in identifying peculiarities of the entity occurrences of the superclass to derive the subclasses. On the other side, generalisation, a bottom-up approach, aims at identifying similarities between different subclasses in order to define their superclass.

In a specialisation/generalisation hierarchy, two types of constraints can be applied: participation and disjointness. The participation constraint determines whether every occurrence of the superclass must also participate in a subclass. The disjoint constraint indicates whether a member of the superclass can be member of only one, or more than one, subclass.

Traditional and enhanced ER models can be used to represent opinions. Chapter 6 shows that these approaches would include implementation details at the conceptual layer, and suggests how to extend traditional ER models in order to separate such implementation details from the conceptual design, i.e. focusing on *what* to model rather than *how* to model it.

Figure 2.11 introduces the main symbols used in this thesis for traditional E-ER diagrams.

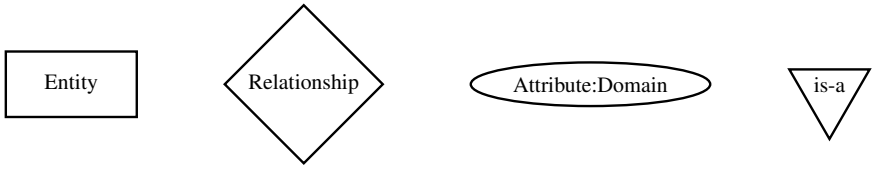


Figure 2.11: Symbols used in Enhanced Entity-Relationship diagrams.

An additional field of database research which can be relevant to the modelling of opinions is the area of fuzzy databases. In fuzzy set theory, the membership of an element x to a set F is expressed by a membership function, denoted as $\mu_F(x)$ whose value is a real number in the interval $[0,1]$. In other words, the element x may have a finite degree of being a member of F, rather than either being or not being a member of the set F. A fuzzy set F is hence described as follows:

$$F = \{\mu_F(x_1)/x_1, \mu_F(x_2)/x_2, ..., \mu_F(x_n)/x_n\}$$

Imprecise and uncertain information occur in many real world applications. Classical data models may show some limitations in representing and manipulating such information. Fuzzy set theory and fuzzy logic have been introduced in the classical data models to deal with imprecision and uncertainty, and extensions of ER/E-ER concepts have been proposed to incorporate fuzzy data in conceptual data modelling [Zvieli and Chen, 1986, Chen and Kerre, 1998]. In particular, Zvieli and Chen pointed out that fuzziness may occur at three different levels. At the first level, the conceptual model is fuzzy: entity sets, relationship sets and attribute sets have a degree of fuzziness. In the second level, a specific occurrence of an entity or a relationship may be fuzzy. At the third level, attributes of a specific entity or relationship may be fuzzy.

Dealing with uncertainty is also pivotal in sentiment analysis applications, and previous research has shown the association between sentiment analysis and fuzzy set theory. However, most of the work in this context has focused on the lexical level, e.g. [Andreevskaia and Bergler, 2006], rather than the data modelling.

2.5.2 Knowledge-oriented IR

IR traditionally deals with the retrieval of documents. As discussed in Section 2.1, some of the main concepts to capture are "relevance" and "content" of documents. One could view "content" as an attribute of entity document, and "relevance" could be seen as a ternary relationship between document, query and user.

Already for the simple case where content is a *bag-of-words* and ranking algorithms rely on the availability of various statistics, the ER model is not really a conceptual model that helps solving IR tasks. The insufficient expressiveness becomes even more evident when facing *knowledge-oriented* IR tasks as they increasingly occur in many applications. Knowledge bases can be automatically generated from high-quality knowledge sources like Wikipedia and other

semantically explicit data repositories, such as ontologies and taxonomies, which explain entities (e.g. persons, movies, locations, etc.) and relationships between entities (e.g. bornIn, actedIn, etc.). Such knowledge bases can be integrated into content-oriented retrieval systems in order to produce a more semantic-aware search experience [Van Zwol and Van Loosbroek, 2007]. Semantic annotations can help traditional content-oriented retrieval system to directly retrieve objects instead of just documents and document elements [Bilotti et al., 2007]. In order to integrate different types of knowledge and effectively enable semantic search on top of the consolidated data, a general purpose data model to represent facts and content knowledge is needed [Azzam and Roelleke, 2011, Azzam et al., 2012].

Without expanding on the details, the case of IR, and in particular, the case of *semantic IR* underline that an extended ER model that supports notions such as *content* and *relevance* could be useful. The opinion-aware ER model proposed in Chapter 6 is related in several ways to the requirements for IR. For IR tasks, the notion of *relevant document* can be modelled with facilities similar to what is later described for *good camera*. Moreover, the subjectivity of opinions mean that a context needs to be modelled, and this is a requirement similar to the IR case where relevance and users need to be considered. The ultimate case is opinion-oriented IR where the queries ask for *good reviews of popular books about database technology*, where such a query combines opinion-oriented criteria with traditional content-oriented criteria.

In summary, the current ER model (basic ER plus object-oriented methods) is a widely used methodology, but it lacks high-level concepts to model requirements as they occur for tasks such as sentiment analysis and information retrieval.

Chapter 5, which focuses on knowledge-oriented summarisation, and Chapter 6, which focuses on the conceptual modelling of opinions, take advantage of a generic data model, based on the relational implementation of the Probabilistic Object-Relational Content Model [Azzam and Roelleke, 2011]. The generic model is shown on the right-hand side of Figure 2.12. In particular, the figure highlights the transition from standard Object-Relational Model (ORM), where only facts (entities, relationships and attributes) are represented, to Object-Relational Content Model (ORCM) where content knowledge is also included in the representation (the extensions are *emphasised*).

classification(ClassName, Object)
relship(RelshipName, Subject, Object)
attribute(AttrName, Object, Value)
part_of(SubObject, SuperObject)
is_a(SubClass, SuperClass)

(a) ORM: Object-Relational Model

classification(ClassName, Object, *Context*)
relship(RelshipName, Subject, Object, *Context*)
attribute(AttrName, Object, Value, *Context*)
part_of(SubObject, SuperObject)
is_a(SubClass, SuperClass, *Context*)
term(Term, Context)

(b) ORCM: Object-Relational *Content* Model

Figure 2.12: From Object-Relational Modelling to Object-Relational Content Modelling.

Chapter 3

Extractive Summarisation based on Statistical Models

3.1 Introduction

This chapter focuses on extractive summarisation tasks using statistical methods. Section 2.2.6 has previously discussed evaluation techniques for summarisation. In particular, in the context of system-based evaluation, the main assumption behind evaluation frameworks like ROUGE is that the content of a good candidate summary has to overlap with the content of a gold standard summary. Such assumption is brought into the context of selecting the verbatim material from the source to include in the summary. Specifically, the aim is to quantify how similar, or how different, a summary is from its source. For this reason, this chapter discusses the use of similarity and divergence-based methods to select or discard candidate sentences for a summary. Such methodology has the advantadge of being less computationally expensive than other approaches which perform abstraction (e.g. graph-based approaches like [Ganesan et al., 2010])

The chapter is structured as follows. Section 3.2 firstly introduces the basic concepts of similarity and divergence, and their use in Information Retrieval. Section 3.3 proposes the use of similarity and divergence for sentence selection for summarisation, introducing a novel algorithm based on sentence removal. Section 3.4 shows the experimental study to evaluate the performance of the sentence removal algorithm on intrinsic sentiment summarisation. The main contribution of this chapter is the aforementioned sentence removal algorithm, and its evaluation in an intrinsic summarisation task.

3.2 Similarity and Divergence in Information Retrieval

3.2.1 Measuring Similarity

A similarity measure, or similarity function, is a real-valued function which describes how similar two objects are. In the context of information retrieval and text summarisation, such objects are typically documents and queries, as well as sentences and summaries. A commonly used similarity function is cosine similarity, which specifies the degree of similarity between two vectors as the cosine of the angle between the two vectors. Cosine similarity can be used to match documents and queries: in this model, a document and a query are represented as vectors in n-dimensional space, with each of the n dimension representing a term. The presence of terms in a document can be represented in different ways. The first options consists in using binary weights, 0 or 1, to describe absence or presence, respectively. In order to portray the importance of a term rather than its mere presence, another option consists in using term weights such as TF or TF-IDF, as described in Section 2.1.1.

Given a document d and a query q represented as vectors of n term weights, $\vec{d} =< d_1, ..., d_n >$ and $\vec{q} =< q_1, ..., q_n >$, their cosine similarity is defined as follows:

$$\text{sim}(d,q) = \cos(\angle(\vec{d},\vec{q})) = \frac{\vec{d}\cdot\vec{q}}{||\vec{d}||\cdot||\vec{q}||} = \frac{\sum_i^n d_i q_i}{\sqrt{\sum_i^n (d_i)^2}\cdot\sqrt{\sum_i^n (q_i)^2}} \tag{3.1}$$

The geometric interpretation of cosine similarity is visualised in Figure 3.1, where a document and a query are represented in a 2-dimensional space (i.e. two terms) for simplicity.

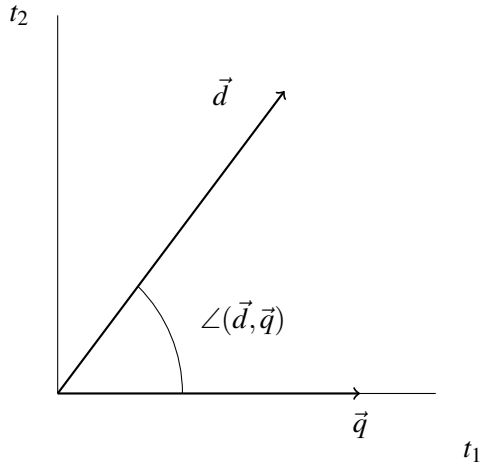


Figure 3.1: Graphical representation of document and query vectors in 2-dimensional space.

Cosine similarity is in general a real number in the range [-1, 1], with 1 indicating vectors with identical orientation and -1 indicating opposed vectors. In Information Retrieval applications, since term weights in vectors are non-negative, the similarity function is used in the positive space, hence the outcome is bounded to [0,1], with higher values indicating higher similarity between the two vectors.

3.2.2 Measuring Divergence

In statistics, divergence is a function which describes "how distant" a probability distribution is from another probability distribution. Intuitively this notion is opposite to similarity, i.e. akin distributions will have low divergence and high similarity, although the two concepts do not complement each other. Divergence is also close to the concept of mathematical distance, but somehow weaker as divergence is not necessarily symmetric, nor it has to satisfy the triangle inequality.

Formally, given a space S and two probability distributions P and Q, a divergence is a function $D(\cdot||\cdot) : S \times S \rightarrow \mathbb{R}$, such that:

- $D(P||Q) \geq 0, \forall P, Q \in S$
- $D(P||Q) = 0$ iif $P = Q$

In Information Retrieval, a commonly used divergence function is the Kullback-Leibler divergence, or KL-divergence in short. KL-divergence is interpreted as the measure of information lost when the distribution Q is used to approximate the distribution P. For the probability distributions P and Q, KL-divergence is defined as follow:

$$D_{KL}(P||Q) := \sum_i P_i \log \frac{P_i}{Q_i} \tag{3.2}$$

KL-divergence is tied to the concepts of entropy (H) and cross-entropy (H_{cross}). Specifically, its information theoretic interpretation is defined as follows:

$$\begin{aligned} D_{KL}(P||Q) = &\quad H_{cross}(P,Q) - H(P) \\ &= -\sum_i P_i \log(Q_i) + \sum_i P_i \log(P_i) \\ &= \quad \sum_i P_i \log \frac{P_i}{Q_i} \end{aligned} \tag{3.3}$$

This connection with information theory motivates the use of KL-divergence as a retrieval model [Zhai, 2008], as the opposite of divergence can be seen as a similarity function:

$$\text{RSV}_{KL}(d,q) := -D_{KL}(P_q||P_d) \tag{3.4}$$

Moreover, in the context of information retrieval, KL-divergence has also been used by Cronen-Townsend et al. to define the concept of query clarity [Cronen-Townsend et al., 2002]. A query is clear if the term probability of a query is different from the term probability of the collection, i.e. clarity is defined as the divergence between the within-query term probability distribution, $P_q(t) = P(t|q)$, and the collection-wide term probability distribution, $P_c(t) = P(t|c)$:

$$D_{KL}(P_q||P_c) = \sum_t P_q(t) \log \frac{P_q(t)}{P_c(t)} \tag{3.5}$$

As pointed out by Roelleke [2013], KL-divergence can also be related to the RSV functions of LM and TF-IDF.

The next section discusses an application of similarity and divergence in the context of text summarisation.

3.3 Extractive Summarisation based on Sentence Removal

This section discusses an application of multi-document summarisation. Automatic document summarisation is an important task which provides an efficient access to information. It has been extensively explored as means to reduce the information overload [Nenkova and McKeown, 2011]. As previously described in Section 2.2, the purpose of a summariser is to provide the user with the most important information from the original source, in a short form.

The expansion of the Web, in particular the increasing number of social media sources such as blogs, discussion forums and other review-related services, provides an application case. Specifically, in the context of opinions, document summarisation can help to find out a concise way to express what the different users think about products and services. Given a set of reviews, a retrieval system could respond to information needs such as *find opinions about the sound quality of the new iPod*, but the users would still be required to read a number of sentences in order to understand what the central opinion is.

In order to provide a snippet representing such pivotal opinion, a summariser can be joined to the aforementioned retrieval system, as a second-stage component. Figure 3.2 provides an overview of such a two-stage system, showing the pipeline which leads from the information need (e.g. tell me the major opinion about a topic) to the generation of a short answer to the query.

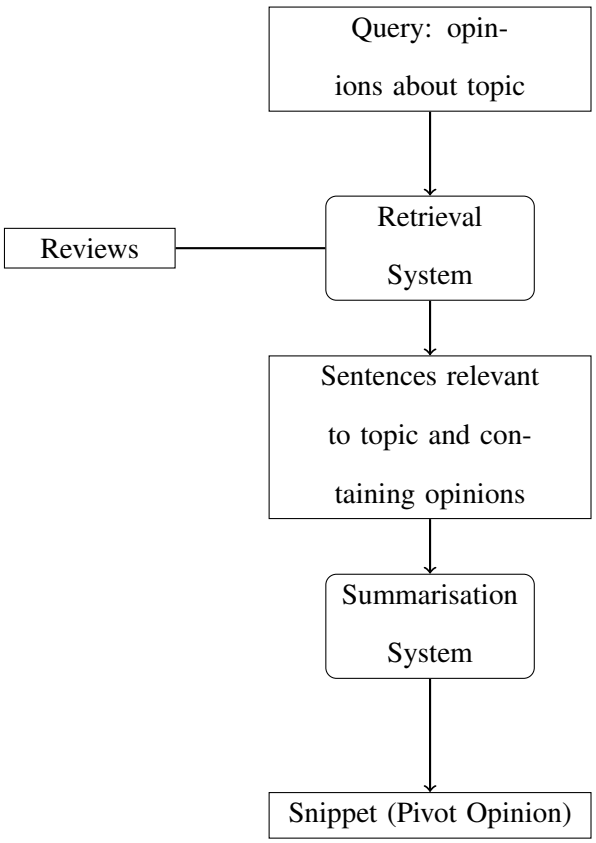


Figure 3.2: Two-stage system for summarising opinions.

This section focuses on how an extractive summariser can provide such a short snippet from a set of redundant sentences, similarly to the scenario described above.

The main contribution of this work consists in the definition of a novel approach to summari-

sation, based on Sentence Removal (SR). This technique removes the less important sentences until the desired summary length is reached. With this approach, the summarisation procedure considers the importance of a candidate summary as a whole, rather than focusing on the importance of a single sentence, and tries to maximise the coverage of relevant information. Different scoring techniques can define the importance of sentences. In particular, cosine similarity and divergence are investigated.

In the following sections, the task of extractive summarisation is first formalised; secondly, some sentence selection strategies are discussed; finally, a summarisation approach based on sentence removal is proposed and examined.

3.3.1 Modelling Extractive Summarisation

In this section the task of extractive summarisation is formally defined.

A collection Q has a number of topics $< q_1, ..., q_m >$. A topic q_j is composed by a number of documents $< d_1, ..., d_k >$, each of which is composed by a number of sentences. A topic is hence composed by all the sentences $< s_1, ..., s_n >$ belonging to the k documents. The case $k = 1$ is called single-document summarisation, while $k > 1$ represents multi-document summarisation. The task of extractive summarisation is to select the subset of sentences and to combine them into a summary which better represents the topic. In order to form the summary, a length limit has to be considered, based on the number of sentences or the number of words.

For each sentence s belonging to a topic q, its probability distribution over terms is given by:

$$P(t|s,Q) = \sum_{d \in q} P(t|d,Q) \cdot P(d|s,Q) \tag{3.6}$$

$P(d|s)$ can be obtain via the Bayesian rule:

$$P(d|s,Q) = \frac{P(s|d,Q) \cdot P(d)}{P(s)} \tag{3.7}$$

then calculating $P(s|d,Q)$ as follows:

$$P(s|d,Q) = \prod_{t \in s} P(t|d,Q) \tag{3.8}$$

and considering a uniform distribution for $P(d)$ and $P(s)$.

For the estimation of $P(t|d,Q)$, linear smoothing is adopted:

$$P(t|d,Q) = \lambda \cdot P(t|d) + (1-\lambda) \cdot P(t|Q) \tag{3.9}$$

where $P(t|d)$ is the relative frequency of a term t in the document d, and $P(t|Q)$ is the relative frequency of a term t in the whole collection of topics Q, i.e. the background model. The parameter λ, defined as real number $0 \leq \lambda \leq 1$, is interpreted as a Dirichlet mixture, i.e. $\frac{|d|}{|d|+\mu}$, where $|d|$ is the document length, and μ is defined as the average document length.

3.3.2 Sentence Selection Strategies

Traditionally, summarisation approaches build the summary by selecting the most significant sentences from the original source. In order to measure the significance of a sentence, one can employ different ranking techniques. Once the sentences are ranked, the top l sentences will form the summary.

The ranking techniques analysed in this work are based on similarity and divergence. Sentences can be ranked with the purpose of maximising their similarity with the given topic, i.e.:

$$\text{score}_{\texttt{SIM}}(s,t) := \text{sim}(P_s, P_q) \tag{3.10}$$

where *sim* can be any similarity metric, P_s is the term probability distribution for the sentence s, as defined in Equation 3.6, and P_q is the relative frequency distribution over the topic q. In this work, cosine similarity is employed as similarity metric. A baseline which forms the candidate summary picking the top l sentences according to Equation 3.10 is referred to as $\text{Greedy}_{\texttt{SIM}}$ in the experiments.

A different approach to score sentences consists in minimising a measure of dissimilarity between a sentence and the given topic. In particular, KL-divergence, previously defined in Equation 3.2, is used in this work. KL-divergence quantifies the proximity of two probability distributions. Specifically, it measures the information lost when approximating a probability distribution P with a candidate distribution Q.

Given a topic q to summarise, for each sentence s belonging to the topic, one can calculate the following score:

$$\text{score}_{\texttt{DIV}}(s,q) := -D_{\texttt{KL}}(P_q||P_s) \tag{3.11}$$

where the negative sign indicates that the lowest divergence gives the highest score. A baseline which forms the candidate summary picking the top l sentences according to Equation 3.11 is referred to as $\text{Greedy}_{\texttt{DIV}}$ in the experiments.

Rather than selecting sentences individually, a different possibility is to select directly the subset of sentences which maximises the chosen score. The brute-force approach consists in enumerating all the combinations of l sentences (desired output length) out of the n forming the given topic. The concatenation of the l sentences with the highest score will form the summary. This is different from selecting one sentence at a time, as the language model for the concatenation will be different from the language model for the individual sentences. The two baselines implementing the brute-force approach adopting the scores as in Equations 3.10 and 3.11 are referred to as, respectively, $\text{BF}_{\texttt{SIM}}$ and $\text{BF}_{\texttt{DIV}}$ in the experiments.

3.3.3 Sentence Removal Algorithm

This section describes the proposed approach to extractive summarisation via a Sentence Removal (SR) algorithm. Instead of selecting important sentence, the idea behind this technique is based on removing iteratively the less important ones, until the desired output size is reached. With this method, the algorithm tries to maximise the importance of the candidate summary as a whole, and does not only focus on the importance of a single sentence. In other words, the purpose is to condense the information in the original source while trying to ensuring its coverage in the summary at the same time. Algorithm 1 shows the procedure to obtain the candidate summary. The procedure starts with the candidate summary q' containing all the original set of sentences, and then iterates until the summary reaches the desired length l. During each iteration, the procedure removes one sentence such that the score between the candidate summary and the original set of sentences is maximised. The score in line 6 can be computed using again Equations 3.10 or 3.11 In the evaluation section, the systems implementing the SR algorithm as in Algorithm 1 are denoted with $\text{SR}_{\texttt{SIM}}$ and $\text{SR}_{\texttt{DIV}}$ depending on the scoring function.

A different version of the sentence removal algorithm can be obtained with a variation in the way the candidate summary, at each iteration step, is selected. Rather than computing the score between the candidate summary and the original set of sentences, one can compute the score between the candidate summary q'_i, and the candidate summary at the previous iteration q'. In this case, line 6 of the procedure has to be replaced with:

$$q' \leftarrow \arg\max_{q'_i}(\text{score}(q', q'_i)) \tag{3.12}$$

The systems implementing this variation of the algorithm are labelled as $\text{SR}'_{\texttt{SIM}}$ and $\text{SR}'_{\texttt{DIV}}$.

Algorithm 1 Sentence Removal algorithm.

Input: q {topic to summarise}

Input: l {output size in n. of sentences}

1: $q' \leftarrow t$

2: **while** $|q'| > l$ **do**

3: **for all** s_i in q' **do**

4: $q'_i \leftarrow q' \setminus s_i$

5: **end for**

6: $q' \leftarrow \arg\max_{q'_i}(\text{score}(q, q'_i))$

7: **end while**

8: **return** q'

3.4 Evaluation

Sentiment Summarisation is the task of summarising the sentiment expressed in a document, or in a set of documents. The task can be defined in different ways as discussed in Section 2.4. This section evaluates the methodology based on sentence removal, discussed in the previous section, to the case of intrinsic sentiment summarisation, even though the methodology is not specifically tailored for opinion-oriented content.

3.4.1 Opinosis Dataset

The Opinosis dataset [Ganesan et al., 2010] is a collection of opinion-oriented data, mainly used for the evaluation of intrinsic sentiment summarisation systems. The data have been collected from popular review web-sites, namely TripAdvisor[1], Amazon and Edmunds[2]. The collection is divided into 51 topics, each topic represents an aspect of a product or service, for example *Battery Life of the Amazon Kindle*, or *Food quality of the Holiday Inn London*. The list of topics has been manually crafted by 2 humans who have been asked to construct opinion-seeking queries, consisting of an entity name (e.g. *Amazon Kindle*) and a topic of interest (e.g. *Battery Life*). Each topic includes a number of sentences (min. 50, max. 575, avg. 139), where the query terms for the related entity appear. For each topic, 4 or 5 gold standard (human-written) summaries are

[1] http://www.tripadvisor.com (Accessed December 2014)

[2] http://www.edmunds.com (Accessed December 2014)

provided. These gold standards have been obtained by leveraging Amazon's Online Workforce[3], asking different workers to summarise the topics in a concise way. The gold standards have been manually reviewed by Ganesan et al. in order to remove summaries with very little or no correlation with the majority. The data are not labelled according to their sentiment, and different opinions can be expressed. The gold standard summaries hence present the main opinion for each topic, in a concise way (approximatively 2 sentences each). Another characteristic of the data is to be highly redundant, e.g. the topic itself is often repeated in different sentences. Figure 3.3 reports a sample of opinion-oriented data from the dataset.

Sample reviews	The room was quiet apart from the hum of the minibar . The hotel was clean and the room was a decent size . *[more sentences]*
Sample summary	The rooms are small but adequate and clean. Service is good.

Figure 3.3: Sample of opinion-oriented data from Opinosis.

3.4.2 Set-up

The experiments are run over the Opinosis collection described in Section 3.4.1. Since the task is intrinsic summarisation and gold standard summaries are available, the ROUGE framework (see Section 2.2.6) is employed to quantitatively assess the agreement between system-produced summaries and human-composed summaries. Multiple human summaries are available for each topic, so the evaluation can achieve better correlation with human judgement as observed by Lin [Lin, 2004a]. In particular, results on ROUGE-1, ROUGE-2 and ROUGE-SU4 are reported. This is in line with common shared evaluation tasks [Ganesan et al., 2010]. ROUGE is based on the *n*-gram overlap between system and human summaries, so precision and recall values, as well as their harmonic mean (F_1-score) are reported. Given the brevity of the summaries, capturing all the relevant information is particularly challenging. Recall is hence particularly important, i.e. it is desirable to show *all* the relevant opinions in the summaries, yet maintaining their succinct nature. For this reason, the results for F_2-scores, which emphasise the importance of recall over precision, are also reported.

[3]http://www.mturk.com [Accessed December 2014]

Section 3.3 has discussed the two versions of the sentence removal algorithm, as well as the two baselines, namely a greedy approach and a brute force approach. Each of these four approaches can employ a different way to calculate sentence similarity. Specifically, cosine similarity and KL-divergence are used in this evaluation, providing a total of 8 different candidates for the experimental study. On top of these systems, this study also reports the results for MEAD [Radev et al., 2004], a state-of-the-art extractive summariser based on cluster centroids.

Figure 3.4 summarises all the candidates for the experiments on intrinsic summarisation.

Candidate	Description
MEAD	The MEAD system as introduced in [Radev et al., 2004]
$\text{Greedy}_{\texttt{SIM}}$	Greedy approach, sentence similarity based on cosine
$\text{BF}_{\texttt{SIM}}$	Brute force approach, sentence similarity based on cosine
$\text{SR}_{\texttt{SIM}}$	Sentence removal algorithm, sentence similarity based on cosine
$\text{SR}'_{\texttt{SIM}}$	Variation of sentence removal algorithm, sentence similarity based on cosine
$\text{Greedy}_{\texttt{DIV}}$	Greedy approach, sentence similarity based on divergence
$\text{BF}_{\texttt{DIV}}$	Brute force approach, sentence similarity based on divergence
$\text{SR}_{\texttt{DIV}}$	Sentence removal algorithm, sentence similarity based on divergence
$\text{SR}'_{\texttt{DIV}}$	Variation of sentence removal algorithm, sentence similarity based on divergence

Figure 3.4: List of candidates for the experiments on intrinsic summarisation.

3.4.3 Results

The numerical results are split over three figures in order to report on the three chosen metrics. Figure 3.5 shows ROUGE-1 scores, Figure 3.6 shows ROUGE-2 scores, and Figure 3.7 shows ROUGE-SU4 scores.

The MEAD baseline is extremely competitive with respect to recall, but shows a drop of performance on the precision side. Overall, the results show that there is not an individual system which clearly outperforms all the others in every metric. In general, the variation of the sentence removal algorithm, SR$'$, is outperformed by the original definition of SR, for both the cosine and the divergence-based settings. The SR algorithm consistently achieves the best recall results within the same scoring function groups when compared to greedy or brute-force approaches.

Observing the cosine-based systems, the greedy baseline shows the best F_1-scores for ROUGE-1, while the brute-force approach shows the best F_1-scores in ROUGE-2 and ROUGE-SU4, but in all cases the results are not significantly better than the SR algorithm. On the F_2-scores side, the SR algorithm with cosine similarity shows overall positive results, being substantially better than any other system, including the divergence-based ones and MEAD, with the second-best results being consistently outside a 95% confidence interval for all the ROUGE-2 and ROUGE-SU4, while showing second-best results in terms of ROUGE-1, where the top performance is provided by the greedy baseline with cosine similarity. Within the divergence-based side, a similar behaviour of brute-force and greedy baselines can be observed: the greedy approach shows the best F_1-score in ROUGE-1, while the brute-force approach achieves the best F_1-scores in ROUGE-2 and ROUGE-SU4, showing good performance in precision. The F_2-scores for brute-force are slightly better than the SR ones in ROUGE-2 and ROUGE-SU4, while SR achieves the best ROUGE-1 score for the divergence-based systems. In all cases, within the divergence-based systems, the top results are not significantly better than the second-best ones.

ROUGE-1				
	Recall	Precision	F_1-score	F_2-score
MEAD	**49.32** †	9.16	15.15	26.27
$\text{Greedy}_{\texttt{SIM}}$	32.98	29.40	**29.66**	**32.19**
$\text{BF}_{\texttt{SIM}}$	25.40	*31.70*	27.50	26.45
$\text{SR}_{\texttt{SIM}}$	*37.46*	19.41	24.63	31.58
$\text{SR}'_{\texttt{SIM}}$	17.17	26.74	20.46	18.49
$\text{Greedy}_{\texttt{DIV}}$	26.42	**32.58**	*28.43*	*27.46*
$\text{BF}_{\texttt{DIV}}$	21.43	31.04	24.79	22.84
$\text{SR}_{\texttt{DIV}}$	*47.05*	9.57	15.57	26.38
$\text{SR}'_{\texttt{DIV}}$	15.60	12.70	13.33	14.92

Figure 3.5: ROUGE-1 scores on the Opinosis dataset. The best overall results are shown in bold. The best results within the same scoring function are shown in *italic*. Best results labelled with a † show that the second-best results are outside their 95% confidence interval.

ROUGE-2				
	Recall	Precision	F_1-score	F_2-score
MEAD	**10.58**	2.78	3.08	5.43
$\text{Greedy}_{\texttt{SIM}}$	6.43	2.78	3.66	4.71
$\text{BF}_{\texttt{SIM}}$	5.75	*7.78*	*6.39*	5.95
$\text{SR}_{\texttt{SIM}}$	*9.29*	5.18	6.23	**7.54** †
$\text{SR}'_{\texttt{SIM}}$	2.64	4.68	3.28	2.86
$\text{Greedy}_{\texttt{DIV}}$	3.99	6.64	4.88	4.29
$\text{BF}_{\texttt{DIV}}$	5.54	**8.36**	**6.50**	*5.86*
$\text{SR}_{\texttt{DIV}}$	*8.67*	1.77	2.88	4.70
$\text{SR}'_{\texttt{DIV}}$	1.44	1.20	1.25	1.34

Figure 3.6: ROUGE-2 scores on the Opinosis dataset. The best overall results are shown in bold. The best results within the same scoring function are shown in *italic*. Best results labelled with a † show that the second-best results are outside their 95% confidence interval.

ROUGE-SU4				
	Recall	Precision	F_1-score	F_2-score
MEAD	**23.16** †	1.02	1.89	4.34
$\text{Greedy}_{\texttt{SIM}}$	12.12	3.00	4.19	5.94
$\text{BF}_{\texttt{SIM}}$	5.43	*10.27*	*6.42*	5.66
$\text{SR}_{\texttt{SIM}}$	*13.80*	5.44	6.31	**8.28** †
$\text{SR}'_{\texttt{SIM}}$	3.03	8.72	4.16	3.37
$\text{Greedy}_{\texttt{DIV}}$	3.91	11.22	5.47	4.39
$\text{BF}_{\texttt{DIV}}$	4.97	**12.10**	**6.59**	*5.48*
$\text{SR}_{\texttt{DIV}}$	*20.10*	1.10	2.03	4.16
$\text{SR}'_{\texttt{DIV}}$	2.96	2.23	2.11	2.38

Figure 3.7: ROUGE-SU4 scores on the Opinosis dataset. The best overall results are shown in bold. The best results within the same scoring function are shown in *italic*. Best results labelled with a † show that the second-best results are outside their 95% confidence interval.

3.4.4 Analysis

The experimental study does not identify one particular approach which is consistently performing better than the others.

The MEAD baseline shows the best recall across all the different metrics, and in two out of three cases its results are significantly better than the second-best system. At the same time, the performances of MEAD in terms of precision drop dramatically and this affects also the F_1 (harmonic mean between precision and recall) and F_2 (recall weight is twice as important as precision) scores. This behaviour can be linked to the fact that MEAD tends to select longer sentences. When longer on-topic sentences are selected, more terms can match the gold standard summaries, hence the higher recall, but at the same time such sentences can carry more information which is not relevant, hence the poor performances in precision.

A similar behaviour can be observed for $\text{SR}_{\texttt{DIV}}$: a high recall, though not as high as MEAD, associated with a very low precision. This finding is also associated with a tendency in selecting longer sentences. In general, MEAD and $\text{SR}_{\texttt{DIV}}$ produce the longest summaries, with an average of $\sim$80 terms. $\text{SR}_{\texttt{SIM}}$ follows with an average of $\sim$60 terms per summary, which explains the lower recall and higher precision than the other two systems. All the other six approaches (brute-force, greedy and the variation of SR, all combined with cosine or divergence) produce much shorter summaries, with an average of $\sim$16 terms between them.

Four of the six approaches which produce shorter summaries, namely $\text{BF}_{\texttt{SIM}}$, $\text{SR}'_{\texttt{SIM}}$, $\text{Greedy}_{\texttt{DIV}}$ and $\text{BF}_{\texttt{DIV}}$, show an opposite tendency in the distribution of precision and recall. For such approaches, precision is in fact consistently higher than recall for the three different ROUGE metrics. While the higher precision is intuitively associated to the shorter summaries, it is not clear why the same attitude is not observed also for $\text{Greedy}_{\texttt{SIM}}$ and $\text{SR}'_{\texttt{DIV}}$.

Overall, the behaviour of SR confirm its original purpose: shortening the summary iteratively while maximising the coverage of information from the source.

3.4.5 Discussion

This section has discussed the evaluation of the novel approach for document summarisation based on sentence removal, introduced in Section 3.3. While the approach is not explicitly tailored for sentiment-oriented data, the evaluation has been performed using a sentiment-oriented

data-set for the task of intrinsic summarisation. The availability of gold standard summaries allowed to utilise a framework for automatic evaluation such as ROUGE. The choice of different ROUGE metrics to report has been based on common practices applied in popular shared evaluation tasks. The design of the sentence removal algorithm, as well as the two baselines, supports the use of different sentence scoring functions. The two functions chosen for this evaluation are cosine similarity and KL-divergence, both being well-established in different Information Retrieval applications.

The experimental study does not identify one of the sentence scoring functions as being consistently outperforming the other. Similarly, none of the approaches regularly outperforms all the others across all the different metrics, although $\text{SR}_{\texttt{SIM}}$ is consistent in being the top-performer with respect to F_2 scores. Summary length is linked to performances, as in general longer summaries provide higher recall and lower precision. This is particularly highlighted for MEAD, whose F_1 and F_2 scores are deeply affected by the drop in precision. $\text{SR}_{\texttt{SIM}}$ is also affected by this tendency but in a milder way. In particular, the drop in precision does not prevent its performances in terms of F_2 scores to be consistently and significantly higher than all the other approaches. Overall, experimental results confirm the intuition behind the sentence removal algorithm, which tries to maximise the coverage of original information while iteratively shortening the summary.

3.5 Summary

This chapter has focused on similarity and divergence measures, applied to document summarisation tasks. Firstly, similarity and divergence have been discussed, in particular in the context of information retrieval, using cosine similarity and KL-divergence. This introduction has laid the ground for the main contribution of the chapter, namely the definition of a novel algorithm for document summarisation based on sentence removal.

The sentence removal technique is based on the idea of removing unimportant sentences from a document, until a desired document length is reached. This approach is also based on computing similarity scores, between the source document and the candidate summaries. The similarity scores can be computed using different techniques. In particular, cosine similarity and KL-divergence are discussed. The approach has been evaluated in the context of intrinsic sentiment

summarisation, even though the technique is not only tailored to opinion-oriented content.

The experimental study has confirmed the intuition behind the sentence removal algorithm, as it tries to maximise the coverage of the original information while shortening the summary. The results do not identify a specific similarity scoring technique as consistently outperforming the other. Overall, the sentence removal algorithm shows consistent top-performance in terms of F_2 score.

Chapter 4

Opinion-based Extractive Summarisation based on Statistical Models

4.1 Introduction

The previous chapter has opened the discussion on statistical methods for extractive summarisation tasks. The methodologies proposed so far are general purpose, without a specific focus on sentiment analysis applications. This chapter continues the discussion on extractive summarisation, taking opinions into explicit consideration.

The starting point of this chapter is a simple observation: in natural language, not all terms are created equal. Depending on the context, different groups of terms can be regarded as more important (or less important) than others, and for this reason they could be subject of a particular treatment. For example, in information retrieval, some terms which are known not to be content-bearing are directly discarded, i.e. stop-word removal is a common pre-processing step. A similar intuition also arises in sentiment analysis tasks, although with an opposite perspective: opinion-bearing terms are object of a particular interest. Opinion-bearing terms (opinion terms in short) are the key to identify whether a sentence is expressing a sentiment or not, and whether the polarity is positive or negative. Figure 4.1 reports some examples of sentences from the Opinosis dataset [Ganesan et al., 2010], highlighting the opinion terms and providing their respective polarity, as well as the overall polarity of the single sentences. The overall polarity

can be influenced by the presence of negation terms such as *not* or *without*. It is interesting to note that such terms are usually part of stop-word lists, and hence their removal would affect the possibility of correctly identifying the polarity.

Sentence	Term polarity	Sentence polarity
Staff are *friendly*	positive	positive
The beds were not very *comfortable*	positive	negative
But I really *liked* the hotel	positive	positive

Figure 4.1: Examples of sentences with opinion-bearing terms from the Opinosis dataset. Opinion terms are in *italic*. The sentence polarity can differ from the term polarity due to negations.

This chapter contributes to the treatment of opinions in the context of extractive summarisation, in two different ways.

Firstly, opinions are considered at the word level, by identifying opinion-bearing terms and pre-processing them before summarisation. The identification step leverages resources such as dictionaries, which are fairly cheap to obtain. Several options for pre-processing, commonly used in retrieval and classification tasks, are considered in this chapter, in order to investigate how such pre-processing step affects summarisation.

Secondly, opinions are considered at the sentence level, by leveragin labelled data in order to classify sentences as either subjective (they carry opinions) or objective (they do not carry opinions). Summaries built on top of a subjectivity detection component are evaluated against full-text documents in a polarity classification task. The purpose of such summarisation is to provide the user with a short document which preserves the original polarity.

The remainder of this chapter is organised as follows: Section 4.2 discussed the pre-processing of opinion terms based on dictionaries. Section 4.3 evaluates the different pre-processing approaches on the intrinsic sentiment summarisation task, using the sentence removal algorithm presented in the previous chapter. Section 4.4 introduces subjectivity detection and discusses its use to build summaries which preserve the original polarity. Section 4.5 evaluates the use of subjective summaries in the context of sentiment classification.

4.2 Preprocessing of Opinion Terms based on Dictionaries

Dictionaries are used in information retrieval for a number of different applications, e.g. spelling correction or phonetic correction. A dictionary, in its most generic form, is a list of terms which can be looked up by the information retrieval system for a given purpose. For example, a list of stop-words can be regarded as a dictionary. Dictionaries can assume also more complex forms, to provide additional semantic information other than a mere list of terms. This is the case, for example, of *thesauri*, which list terms in groups of synonyms and related concepts, or *gazetteers*, which provide an index of geographical names. Dictionary-based approaches are fairly popular also in sentiment analysis applications, because they are relatively cheap to obtain or generate. In this section, some potential uses of dictionaries of opinion-bearing terms are discussed.

4.2.1 Opinion Terms as Stop-words

Intuitively, opinion terms should be treated differently from stop-words. In fact, whereas stop-words are commonly seen as unimportant as they do not carry information, opinion-bearing terms are key indicators of polarity and hence they are usually crucial in a sentiment analysis application. On the other side, one aspect to consider when analysing opinion-oriented content such as user-generated reviews is that opinion-bearing terms are present in the vast majority of documents, so their informativeness (i.e. their IDF score) is usually low. This observation supports the idea of treating the opinion-bearing terms as stop-words. Figure 4.2 provides an example from a restaurant review, where stop-words and opinion-words are removed.

Full-text representation	The food was amazing and there was a great atmosphere
Stop-word removal	food amazing great atmosphere
Stop-word and opinion removal	food atmosphere

Figure 4.2: Example of stop-word and opinion-word removal.

In this example, the terms *amazing* and *great* are listed as opinion-bearing words in a dedicated dictionary. Such terms are preserved during a regular stop-word removal step. The last line of Figure 4.2 shows the effect of applying both stop-word and opinion-word removal: only the terms *food* and *atmosphere* remain in the representation. While the opinion information is lost during this removal process, the focus is moved towards content words which are representative of what the review is discussing, in this case specific aspects of the restaurant.

4.2.2 Boosting Frequencies

Boosting term frequencies can be used in order to assign more importance to opinion words. Specifically, if a term belongs to a dictionary of opinion-bearing words, its weight can be increased by repeating multiple times the term itself. Figure 4.3 shows an example with a restaurant review where the term *good* is identified as an opinion-bearing term and its frequency is boosted. The last column of the figure shows the effect of the boosting with respect to the maximum-likelihood estimation of TF.

	Document	$\frac{n_L(\text{“}good''\text{,}d)}{N_L(d)}$
Regular representation	Breakfast with a *good* selection of food	$1/7 = 0.14$
Boosting $\times 2$	Breakfast with a *good good* selection of food	$2/8 = 0.25$
Boosting $\times 3$	Breakfast with a *good good good* selection of food	$3/9 = 0.33$

Figure 4.3: Example of term frequency boosting applied on opinion words.

4.2.3 Phrases and N-grams

A phrase is a sequence of words, of even a single word, which forms a constituent, i.e. it functions as a single unit within the structure of a sentence [Manning and Schütze, 1999]. Examples of phrases include compounds (e.g. *disk drive*), phrasal verbs (e.g. *make up*) or phrasal nouns (e.g. *the big room*). In the context of sentiment analysis, an interesting application consists in identifying phrases such as *good food*, i.e. a phrase where an opinion-bearing term matches a topic term. Phrases can be arbitrarily long and estimating the probability of specific sequences can become difficult in some text collections. A common approach to estimate such probabilities consists in approximating the language model of a sequence using an *n*-gram model.

An *n*-gram is a sequence of *n* adjacent elements taken from a sequence of tokens (in the case of information retrieval and text analytics applications, tokens are usually terms). The most commonly used types of *n*-grams, besides unigrams (single terms), are bigrams and trigrams, respectively sequences of two and three adjacent terms. Figure 4.4 shows an example of how a sentence can be represented as a sequence of unigrams, bigrams, trigrams.

The following equation can be employed in order to calculate the probability of observing the

Sentence s	they serve great food
Unigrams	”they”, ”serve”, ”great”, ”food”
Bigrams	”they serve”, ”serve good”, ”good food”
Trigrams	”they serve good”, ”serve good food”

Figure 4.4: Example of sentence representation with unigrams, bigrams or trigrams.

sentence $s =< t_1, t_2, ..., t_n >$ as a sequence of terms:

$$P(s) = \prod_{i=0}^{n+1} P(t_i | t_0, ..., t_{i-1}) \tag{4.1}$$

In general, t_0 and $_{n+1}$ indicate the beginning and the end of the sentence, and are often substituted with surrogate terms (placeholders) such as “<s>” and “</s>”.

Using a unigram language model, the terms are independent, hence the individual term probabilities can be written as:

$$P(s) = \prod_{i=1}^{n} P(t_i) \tag{4.2}$$

The tokens <s> and </s> are not explicitly included in Equation 4.2 because their individual probabilities are both equal to 1, i.e. all sentences have a beginning and an end. As previously mentioned, what Equation 4.2 shows is that the order of the terms is not important, and that the context is ignored, for example:

$$P(\text{“good food”}) = P(\text{“food good”}) = P(\text{“good”}) \cdot P(\text{“food”}) \tag{4.3}$$

The first step for considering the context in which the terms appear, i.e. for considering some form of term dependencies, consists in adopting a bigram-based language model. With such model, the probability of observing each term will be conditioned by the probability of the previous term, i.e. the individual term probabilities from Equation 4.1 can be rewritten as:

$$P(t_i | t_0, ..., t_{i-1}) \approx P(t_i | t_{i-1}) \tag{4.4}$$

Hence, Equation 4.1 itself can be rewritten as:

$$P(s) = \prod_{i=1}^{n+1} P(t_i|t_{i-1}) \tag{4.5}$$

A different approach consists in combining the concepts of dictionary-based pre-processing of opinion terms with what discussed in this section about phrases and bigrams. Specifically, a dictionary of opinion-bearing terms can be used to identify such terms and to create special tokens by merging the opinion-terms to the following term. The purpose is to capture phrases like *good food* or *great atmosphere*. Figure 4.5 provides an example where the term *great* is identified as opinion-bearing term.

Sentence	they serve *great* food.
Regular unigrams	"they", "serve", "great", "food"
Opinion bigrams	"they", "serve", "great_food"

Figure 4.5: Example of hybrid approach where an opinion-bearing term is joined to the following term to create a new token.

In a unigram representation, the two terms *great* and *food*, would be shown as different tokens. In this example, they are instead merged into a single token which represents the topic (i.e. *food*) and the related opinion (i.e. *great*) as a whole.

In general, the use of bigrams and *n*-grams in sentiment analysis applications has been widely used, mainly for sentiment classification. Despite the intuition which motivates their use, i.e. the ability to capture phrases like *good food*, experimental results in previous work related to sentiment classification do not seem to be conclusive about the importance of using bigrams and *n*-grams [Pang et al., 2002, Pang and Lee, 2008].

4.2.4 Dealing with Negation

Negative qualifiers such as *not* or *no* are commonly used in several different contexts, including opinion-bearing content [Potts, 2011]. Typical examples in sentiment analysis include the use of phrases such as *not good* or *not bad*, where the mere observation of opinion terms in a document could lead to identifying the opposite polarity for the term itself. This use of negation is referred to as *local*, because the negative qualifier and the related opinion term are particularly close to each other. Further examples of the use of negation in opinion-oriented sentences include longer-distance dependencies such as the negation of the proposition of the negation of the

subject [Wilson et al., 2005b]. Figure 4.6 shows some examples of non-local uses of negation associated with opinion terms, highlighting how the role of the negative qualifier can change in different contexts.

Sentence	Term polarity	Sentence polarity
does *not* look *good*	Positive	Negative
no one thinks that it is *good*	Positive	Negative
not only *good* but *amazing*	Positive	(Strongly) Positive

Figure 4.6: Examples of non-local uses of negation in opinion-oriented sentences from [Wilson et al., 2005b]. Negative qualifiers and opinion terms are *emphasised*.

Rather than identifying the polarity of phrases which follow a negation qualifier, a different approach consists in removing such terms, i.e. treating such terms as stop-words. Once a negation qualifier is identified, a window of the following n terms is removed from the document. In the experimental study discussed in Section 4.3, such approach is applied to consider local negation, i.e. window size = 1, meaning that only the first term after a negation qualifier is removed.

4.2.5 Limitations of Dictionary-based Approaches

As previously mentioned, dictionaries of opinion-bearing terms are commonly used in sentiment analysis because they are relatively cheap to obtain. One of their main limitations is related to the importance of the context. Terms with a positive connotation in a particular context, might assume an opposite polarity when considered in a different context. For example, the term *unpredictable* assumes a positive connotation when used to describe a movie plot, because an unpredictable movie is interesting and not boring. On the other side, when the term is used to describe the steering system of a car, the polarity is certainly negative because one desirable feature of the steering system consists in being predictable.

4.3 Evaluation of Preprocessing of Opinion-bearing Terms for Summarisation

This section discusses different approaches for treating opinion-bearing terms within the task of Sentiment Summarisation. Section 3.4 has previously discussed an experimental study to assess the performance of the sentence removal algorithm applied to an intrinsic sentiment summarisation task. While the methodology discussed in this section is very similar to the one exam-

ined in Section 3.4, the focus is on the assessment of different options for the pre-processing of opinion-bearing terms. In other words, the target is not to identify the best performing approach for summarisation, but to highlight what kind of effects are produced by different ways to treat opinion-bearing terms to the different summarisers.

4.3.1 Set-up

The overall set-up for these experiments is essentially the same as the one described in Section 3.4. The Opinosis data-set is employed to perform an intrinsic sentiment summarisation evaluation based on ROUGE metrics. The reader is pointed to Section 3.4 for an overview of the summarisation approaches utilised in these experiments, which are summarised in Figure 3.4, with the exception of the MEAD baseline which is here not considered.

Several options for obtaining a dictionary of opinion-bearing terms are available. The one chosen for this evaluation is a merge of the two dictionaries (positive and negative terms) described in [Dadvar et al., 2011].

The list of negative qualifiers used for negation removal is composed by the following terms: *no*, *not*, *rather*, *hardly*, *without*. Such list has been composed by simplifying a dictionary of negative qualifiers also discussed in [Dadvar et al., 2011]. The window size for negation removal has been fixed to 1, i.e. only local negation is considered in these experiments.

Figure 4.7 summarises all the different treatments of opinion-bearing terms which are analysed in this evaluation.

Treatment	Description
Opinions as stop-words	Opinion terms are treated as stop-words, i.e. removed
Boosting frequencies	Opinion terms are repeated *in loco* a number of times ($2 \leq n \leq 10$)
Opinion-based bigrams	Unigram features are combined with opinion-based unigrams.
Negation-based removal	Terms which follow a negation are removed (window size=1)

Figure 4.7: List of treatments of opinion-bearing terms analysed in the experiments on intrinsic summarisation.

4.3.2 Results

This section discusses the key points of the evaluation. The complete view on all the experimental results on different treatments of opinion-bearing terms is reported in Appendix A.

Opinion Terms as Stop-words

The numerical results for the treatment of opinion terms as stop-words are split over three figures in order to report on the three chosen metrics. Figure 4.8 shows ROUGE-1 scores, Figure 4.9 shows ROUGE-2 scores, and Figure 4.10 shows ROUGE-SU4 scores. These figures also show whether removing the opinion terms improves the performances (results in bold indicate a higher performance than the equivalent run without pre-processing of opinion terms). Overall, removing the opinion terms improves the ROUGE-1 scores only in 7 cases out of 32 (∼21% of the cases), while in the remaining 25 cases the performance deteriorates. In terms of ROUGE-2 instead, removing the opinion terms improves the scores in 24 cases out of 32 (75% of the cases). Finally, in terms of ROUGE-SU4, opinion terms removal improves the performances in 27 out of 32 cases (∼84% of the cases).

ROUGE-1 - Opinions as stop-words				
	Recall	Precision	F_1-score	F_2-score
$\text{Greedy}_{\texttt{SIM}}$	30.92	28.65	27.99	30.44
$\text{BF}_{\texttt{SIM}}$	22.11	28.53	24.14	23.15
$\text{SR}_{\texttt{SIM}}$	**38.69**	18.40	23.75	31.70
$\text{SR}'_{\texttt{SIM}}$	16.35	25.90	19.59	17.65
$\text{Greedy}_{\texttt{DIV}}$	25.52	31.89	27.72	26.58
$\text{BF}_{\texttt{DIV}}$	20.91	30.80	24.38	22.34
$\text{SR}_{\texttt{DIV}}$	46.71	**10.22**	**16.43**	**27.25**
$\text{SR}'_{\texttt{DIV}}$	**16.61**	**12.90**	**13.91**	**15.71**

Figure 4.8: ROUGE-1 scores on the Opinosis dataset after opinion-terms removal. The results which outperform the equivalent run without pre-processing of opinion terms, reported in Figure 3.5, are shown in **bold**.

ROUGE-2 - Opinions as stop-words				
	Recall	Precision	F_1-score	F_2-score
$\text{Greedy}_{\texttt{SIM}}$	**8.95**	**8.49**	**8.20**	**8.85**
$\text{BF}_{\texttt{SIM}}$	5.39	7.04	5.88	5.65
$\text{SR}_{\texttt{SIM}}$	**10.16**	**4.96**	**6.33**	**8.40**
$\text{SR}'_{\texttt{SIM}}$	**2.93**	**5.05**	**3.60**	**3.20**
$\text{Greedy}_{\texttt{DIV}}$	**7.36**	**8.82**	**7.85**	**7.61**
$\text{BF}_{\texttt{DIV}}$	**5.57**	8.18	6.46	**5.95**
$\text{SR}_{\texttt{DIV}}$	8.54	**1.81**	2.92	**4.90**
$\text{SR}'_{\texttt{DIV}}$	**1.81**	**1.49**	**1.56**	**1.73**

Figure 4.9: ROUGE-2 scores on the Opinosis dataset after opinion-terms removal. The results which outperform the equivalent run without pre-processing of opinion terms, reported in Figure 3.6, are shown in **bold**.

ROUGE-SU4 - Opinions as stop-words				
	Recall	Precision	F_1-score	F_2-score
$\text{Greedy}_{\texttt{SIM}}$	10.70	**10.78**	**8.81**	**10.71**
$\text{BF}_{\texttt{SIM}}$	**5.52**	10.01	6.33	**6.04**
$\text{SR}_{\texttt{SIM}}$	**14.49**	**4.41**	**5.68**	**9.94**
$\text{SR}'_{\texttt{SIM}}$	**3.07**	8.58	**4.19**	**3.52**
$\text{Greedy}_{\texttt{DIV}}$	**7.16**	**12.05**	**8.31**	**7.79**
$\text{BF}_{\texttt{DIV}}$	**5.05**	11.85	**6.60**	**5.70**
$\text{SR}_{\texttt{DIV}}$	**20.65**	**1.26**	**2.29**	**5.06**
$\text{SR}'_{\texttt{DIV}}$	**3.45**	**2.32**	**2.37**	**3.14**

Figure 4.10: ROUGE-SU4 scores on the Opinosis dataset after opinion-terms removal. The results which outperform the equivalent run without pre-processing of opinion terms, reported in Figure 3.7, are shown in **bold**.

Boosting Frequencies

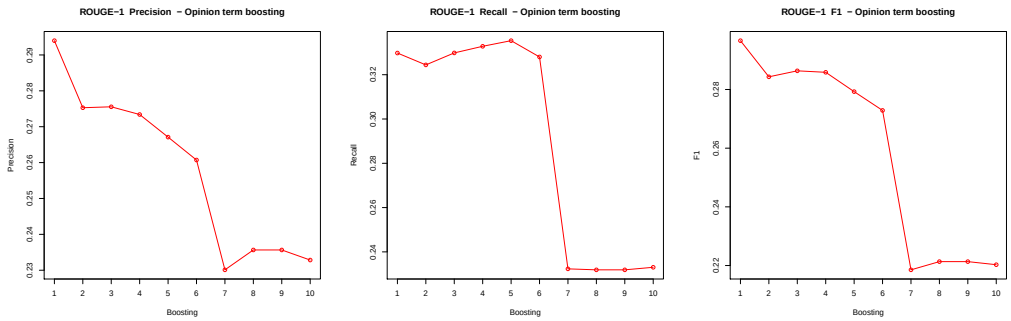
The complete numerical results for the frequency boosting approach are reported in Appendix A. In this section, the results on ROUGE-1 scores are plotted to observe how frequency boosting affects the different summarisation approaches. Results on ROUGE-2 and ROUGE-SU4 are

comparable in the sense that they show a similar tendency, hence the charts are here omitted.

The results are organised as follows:

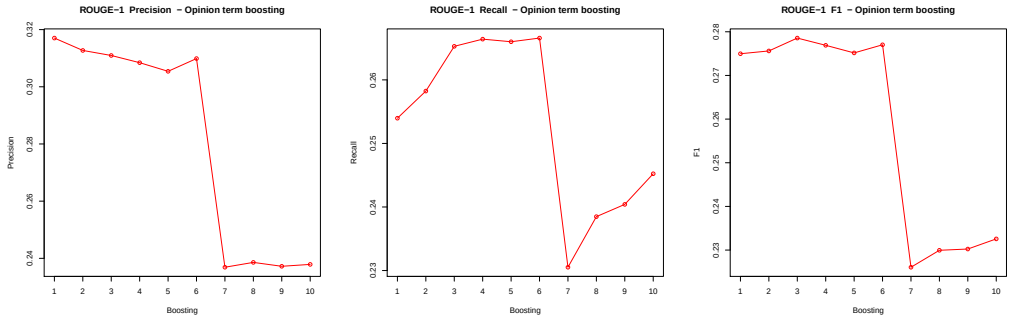
- $\text{Greedy}_{\texttt{SIM}}$ on Figure 4.11
- $\text{BF}_{\texttt{SIM}}$ on Figure 4.12
- $\text{SR}_{\texttt{SIM}}$ on Figure 4.13
- $\text{SR}'_{\texttt{SIM}}$ on Figure 4.14
- $\text{Greedy}_{\texttt{DIV}}$ on Figure 4.15
- $\text{BF}_{\texttt{DIV}}$ on Figure 4.16
- $\text{SR}_{\texttt{DIV}}$ on Figure 4.17
- $\text{SR}'_{\texttt{DIV}}$ on Figure 4.18

with each figure divided into three subfigure to represent precision, recall and F_1 scores.



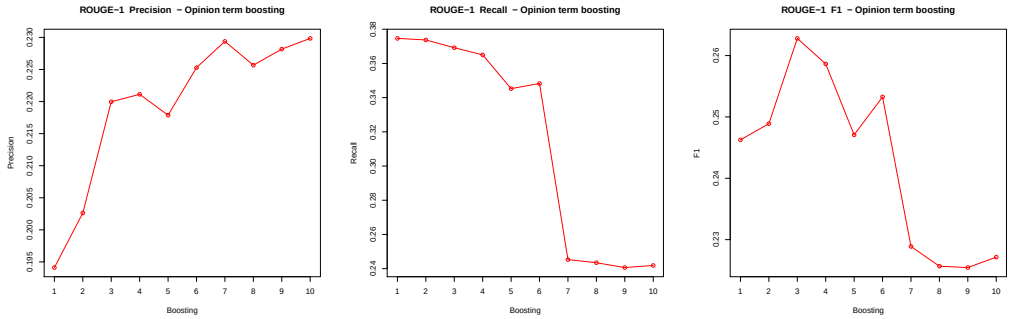
(a) Precision (b) Recall (c) F_1-score

Figure 4.11: Effect of term frequency boosting on ROUGE-1 scores for the $\text{Greedy}_{\texttt{SIM}}$ system.



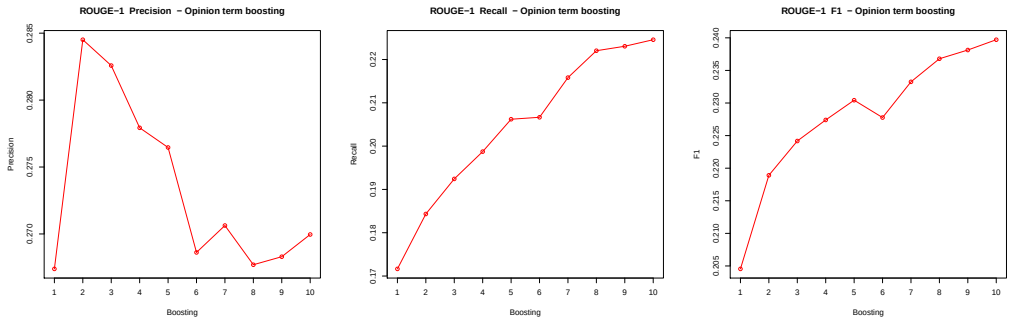
(a) Precision (b) Recall (c) F_1-score

Figure 4.12: Effect of term frequency boosting on ROUGE-1 scores for the $\text{BF}_{\texttt{SIM}}$ system.



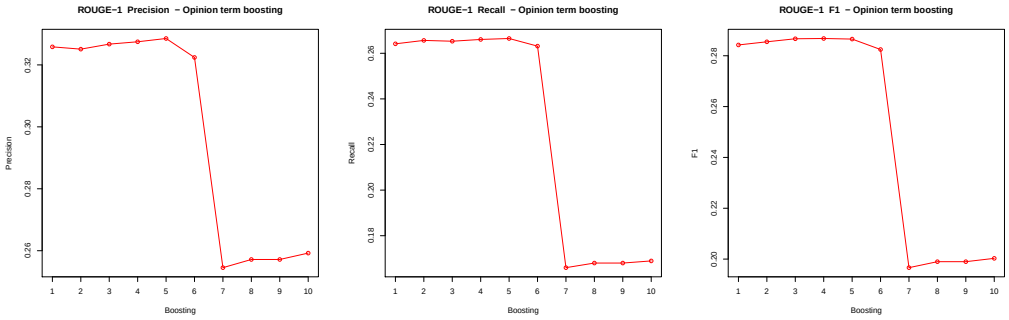
(a) Precision (b) Recall (c) F_1-score

Figure 4.13: Effect of term frequency boosting on ROUGE-1 scores for the $\text{SR}_{\texttt{SIM}}$ system.



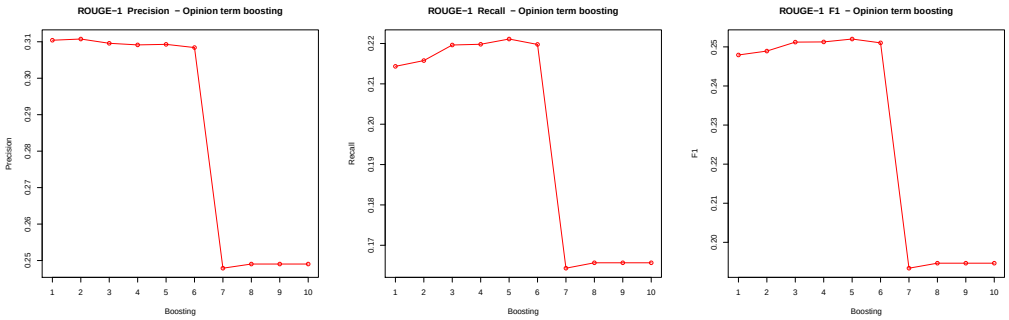
(a) Precision (b) Recall (c) F_1-score

Figure 4.14: Effect of term frequency boosting on ROUGE-1 scores for the $\text{SR}'_{\texttt{SIM}}$ system.



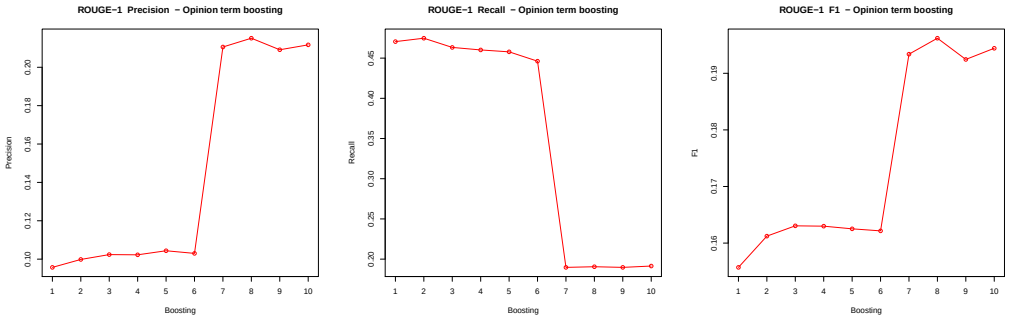
(a) Precision (b) Recall (c) F_1-score

Figure 4.15: Effect of term frequency boosting on ROUGE-1 scores for the Greedy$_{\texttt{DIV}}$ system.



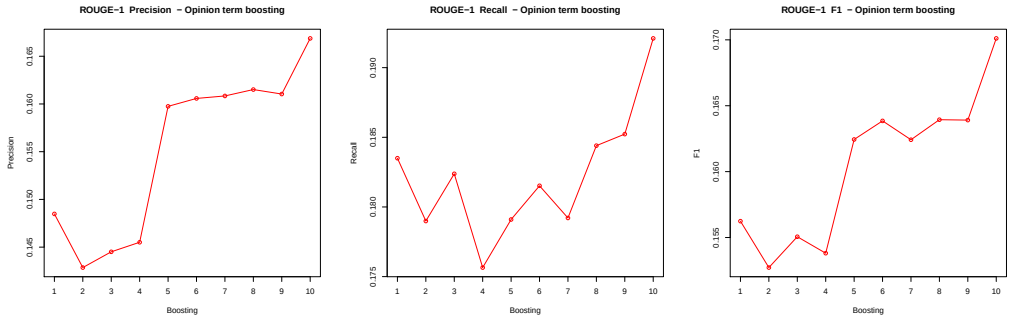
(a) Precision (b) Recall (c) F_1-score

Figure 4.16: Effect of term frequency boosting on ROUGE-1 scores for the BF$_{\texttt{DIV}}$ system.



(a) Precision (b) Recall (c) F_1-score

Figure 4.17: Effect of term frequency boosting on ROUGE-1 scores for the SR$_{\texttt{DIV}}$ system.



(a) Precision (b) Recall (c) F_1-score

Figure 4.18: Effect of term frequency boosting on ROUGE-1 scores for the $\mathrm{SR}'_{\mathtt{DIV}}$ system.

The main observable effect is the drop of performances for the majority of the summarisation approaches when frequencies are boosted to higher levels, i.e. when the opinion terms are repeated 7 times or more. The variation of the sentence removal algorithm, SR$'$, is the only system to show instead the opposite tendency: with higher frequency boosts, the quality of the system improves.

Opinion-based Bigrams

The numerical results for the sentiment summarisation task performed including opinion-based bigrams are split over three figures in order to report on the three chosen metrics. Figure 4.19 shows ROUGE-1 scores, Figure 4.20 shows ROUGE-2 scores, and Figure 4.21 shows ROUGE-SU4 scores. These figures also show whether including the opinion-based bigrams improves the performances (results in bold indicate a higher performance than the equivalent run without pre-processing of opinion terms).

Overall, including opinion-based bigrams improves the ROUGE-1 scores in 20 cases out of 32 ($\sim$62% of the cases), while in the remaining 12 cases the performance deteriorates. In terms of ROUGE-2 instead, opinion-based bigrams improve the scores in 29 cases out of 32 ($\sim$90% of the cases). Finally, in terms of ROUGE-SU4, opinion-based bigrams improve the performances in 25 out of 32 cases ($\sim$78% of the cases).

ROUGE-1 - Opinion-based Bigrams				
	Recall	Precision	F_1-score	F_2-score
$\text{Greedy}_{\texttt{SIM}}$	**33.29**	27.16	28.10	**31.85**
$\text{BF}_{\texttt{SIM}}$	**26.12**	29.02	25.89	**26.65**
$\text{SR}_{\texttt{SIM}}$	**38.28**	18.35	23.53	**31.45**
$\text{SR}'_{\texttt{SIM}}$	**19.59**	26.70	**21.22**	**20.69**
$\text{Greedy}_{\texttt{DIV}}$	**26.56**	30.08	26.76	**27.20**
$\text{BF}_{\texttt{DIV}}$	**22.31**	28.27	23.53	**23.29**
$\text{SR}_{\texttt{DIV}}$	45.88	**10.17**	**16.23**	**26.95**
$\text{SR}'_{\texttt{DIV}}$	**19.80**	**14.91**	**16.08**	**18.58**

Figure 4.19: ROUGE-1 scores on the Opinosis dataset after inclusion of opinion-based bigrams. The results which outperform the equivalent run without pre-processing of opinion terms, reported in Figure 3.5, are shown in **bold**.

ROUGE-2 - Opinion-based Bigrams				
	Recall	Precision	F_1-score	F_2-score
$\text{Greedy}_{\texttt{SIM}}$	**9.34**	**7.90**	**8.04**	**9.01**
$\text{BF}_{\texttt{SIM}}$	**7.31**	**8.66**	**7.51**	7.54
$\text{SR}_{\texttt{SIM}}$	**9.80**	4.80	6.07	**8.11**
$\text{SR}'_{\texttt{SIM}}$	**3.99**	**5.62**	**4.40**	**4.23**
$\text{Greedy}_{\texttt{DIV}}$	**7.18**	**8.51**	**7.46**	**7.41**
$\text{BF}_{\texttt{DIV}}$	**6.22**	8.07	**6.69**	**6.52**
$\text{SR}_{\texttt{DIV}}$	**8.88**	**1.95**	**3.10**	**5.19**
$\text{SR}'_{\texttt{DIV}}$	**2.45**	**1.80**	**1.97**	**2.28**

Figure 4.20: ROUGE-2 scores on the Opinosis dataset after inclusion of opinion-based bigrams. The results which outperform the equivalent run without pre-processing of opinion terms, reported in Figure 3.6, are shown in **bold**.

ROUGE-SU4 - Opinion-based Bigrams				
	Recall	Precision	F_1-score	F_2-score
$\text{Greedy}_{\texttt{SIM}}$	11.78	**9.50**	**8.59**	**11.24**
$\text{BF}_{\texttt{SIM}}$	**7.35**	**10.47**	**7.14**	**7.81**
$\text{SR}_{\texttt{SIM}}$	**14.74**	4.31	5.61	**9.93**
$\text{SR}'_{\texttt{SIM}}$	**4.50**	**9.11**	**4.89**	**5.01**
$\text{Greedy}_{\texttt{DIV}}$	**7.67**	11.08	**7.70**	**8.17**
$\text{BF}_{\texttt{DIV}}$	**5.89**	10.52	6.32	**6.46**
$\text{SR}_{\texttt{DIV}}$	19.84	**1.28**	**2.26**	**5.09**
$\text{SR}'_{\texttt{DIV}}$	**4.73**	**2.99**	**2.97**	**4.24**

Figure 4.21: ROUGE-SU4 scores on the Opinosis dataset after inclusion of opinion-based bigrams. The results which outperform the equivalent run without pre-processing of opinion terms, reported in Figure 3.7, are shown in **bold**.

Negation-based removal

The numerical results for the sentiment summarisation task performed removing terms which appear after a negative qualifier are split over three figures in order to report on the three chosen metrics. Figure 4.22 shows ROUGE-1 scores, Figure 4.23 shows ROUGE-2 scores, and Fig-

ure 4.24 shows ROUGE-SU4 scores. These figures also show whether removing terms after a negative qualifier improves the performances (results in bold indicate a higher performance than the equivalent run without pre-processing of opinion terms).

Overall, negation-based removal improves the ROUGE-1 scores in only in 6 cases out of 32 ($\sim$19% of the cases), while in the remaining 26 cases the performance deteriorates. In terms of ROUGE-2 instead, negation-based removal improves the scores in 10 cases out of 32 ($\sim$31% of the cases). Finally, in terms of ROUGE-SU4, negation-based removal improves the performances in 12 out of 32 cases ($\sim$37% of the cases).

ROUGE-1 - Negation-based Removal				
	Recall	Precision	F_1-score	F_2-score
$\text{Greedy}_{\texttt{SIM}}$	17.86	21.37	19.07	18.47
$\text{BF}_{\texttt{SIM}}$	16.20	20.57	17.67	16.92
$\text{SR}_{\texttt{SIM}}$	23.92	14.93	16.82	21.35
$\text{SR}'_{\texttt{SIM}}$	17.17	26.74	20.46	18.49
$\text{Greedy}_{\texttt{DIV}}$	16.52	25.41	19.58	17.76
$\text{BF}_{\texttt{DIV}}$	16.10	24.31	18.94	17.27
$\text{SR}_{\texttt{DIV}}$	18.03	**19.22**	**18.08**	18.26
$\text{SR}'_{\texttt{DIV}}$	**18.36**	**14.81**	**15.61**	**17.52**

Figure 4.22: ROUGE-1 scores on the Opinosis dataset after negation-based removal. The results which outperform the equivalent run without pre-processing of opinion terms, reported in Figure 3.5, are shown in **bold**.

ROUGE-2 - Negation-based Removal				
	Recall	Precision	F_1-score	F_2-score
$\text{Greedy}_{\texttt{SIM}}$	3.47	**4.03**	3.65	3.57
$\text{BF}_{\texttt{SIM}}$	3.16	4.18	3.52	3.32
$\text{SR}_{\texttt{SIM}}$	3.82	2.02	2.42	3.24
$\text{SR}'_{\texttt{SIM}}$	**3.24**	**5.25**	**3.91**	**3.51**
$\text{Greedy}_{\texttt{DIV}}$	3.05	4.91	3.67	3.30
$\text{BF}_{\texttt{DIV}}$	3.17	5.02	3.79	3.42
$\text{SR}_{\texttt{DIV}}$	2.53	**2.92**	2.63	2.60
$\text{SR}'_{\texttt{DIV}}$	**2.26**	**1.77**	**1.88**	**2.14**

Figure 4.23: ROUGE-2 scores on the Opinosis dataset after negation-based removal. The results which outperform the equivalent run without pre-processing of opinion terms, reported in Figure 3.6, are shown in **bold**.

ROUGE-SU4 - Negation-based Removal				
	Recall	Precision	F_1-score	F_2-score
$\text{Greedy}_{\texttt{SIM}}$	3.90	**5.92**	**4.37**	4.18
$\text{BF}_{\texttt{SIM}}$	3.26	5.77	3.81	3.57
$\text{SR}_{\texttt{SIM}}$	7.23	3.07	3.24	5.69
$\text{SR}'_{\texttt{SIM}}$	**3.44**	**9.24**	**4.67**	**3.93**
$\text{Greedy}_{\texttt{DIV}}$	3.33	8.72	4.48	3.80
$\text{BF}_{\texttt{DIV}}$	3.39	8.48	4.49	3.85
$\text{SR}_{\texttt{DIV}}$	3.90	**5.07**	**3.96**	4.09
$\text{SR}'_{\texttt{DIV}}$	**4.00**	**2.92**	**2.78**	**3.72**

Figure 4.24: ROUGE-SU4 scores on the Opinosis dataset after negation-based removal. The results which outperform the equivalent run without pre-processing of opinion terms, reported in Figure 3.7, are shown in **bold**.

4.3.3 Analysis

The experimental study considers different approaches, based on the use of a dictionary, to treat opinion-bearing terms. The first message provided by this study is that despite the simplicity of using a dictionary, such approaches affect the performances of summarisation systems in a noticeable way.

The first approach consists in considering the opinion-bearing terms as stop-words and hence removing them. Given a sentiment-oriented task, this approach might seem counter-intuitive as opinion-bearing terms can carry crucial information. The rationale behind this approach is that most of the sentences contain opinion-bearing terms, which hence become less informative. The results show contrastive effects. In fact, ROUGE-1 scores are hurt by opinion term removal, while ROUGE-2 and ROUGE-SU4 scores are improved. Overall, the $SR_{DIV'}$ approach is the only one whose scores for opinion-term removal are always higher than the corresponding regular run.

The second approach for opinion treatment can be seen as the opposite of the first one. Rather than removing the opinion-bearing terms, this approach consists in repeating the opinion terms in place, in order to boost their frequency scores. The rationale behind this approach is clearly the idea of boosting the importance of opinion terms by incrementing their presence in the text. Once again, the results are contrastive. Some of the summarisation approaches benefit from frequency boosts. In particular, the two variations of SR' show better results with higher frequencies, regardless of the metric analysed. Most of the other approaches show instead a degradation of their performances, in particular when the opinion terms are repeated 7 times or more. When reaching a frequency boost of 7, there is in fact a dramatic drop in performances for the variation of Greedy, BF and in part also for SR.

The third approach for opinion treatment is the use of opinion-based bigrams. Regular unigrams (i.e. single terms) are combined to opinion-based bigrams to capture phrases such as *good food*. This is the only approach which produces a substantial improvement in performances. For all the metrics, most of the results are overall better than the equivalent regular (e.g. unigram-only) runs. Previous research in sentiment classification, e.g. [Pang et al., 2002], has shown opposite results, i.e. the best performances were achieved by unigram-only runs. Even though the study proposed in this thesis has considered only opinion-based bigrams, while Pang et al. have used all the combinations of bigrams and trigrams, the contrast between the findings of the two studies

probably confirms that the benefits of exploiting *n*-grams in sentiment analysis tasks have not been definitively clarified.

The fourth and last approach for opinion treatment is negation-based removal. The idea is to consider the terms which appear after a negative qualifier as stop-words. While this approach is not explicitly opinion-oriented, most of the negations used in in sentiment-bearing text is connected to the use of opinion terms. Overall, this approach does not benefit the performances of most summarisation systems, with the only exception of $SR'_{\texttt{DIV}}$ which is the only approach consistently showing better results than the corresponding regular run for all the metrics.

4.3.4 Discussion

The aim of this section was to validate some of the intuitions regarding the treatment of opinion-bearing terms, proposing approaches based on the use of a dictionary of opinion-bearing terms. The benefits of using dictionaries consist in their relatively low cost, in terms of generation or acquisition. As previously discussed, the downside of dictionaries mainly consists in the lack of context, i.e. opinion terms assume different polarities in different contexts, hence it is difficult to obtain a general purpose list of opinion-bearing terms.

The results have partially confirmed the difficulty in terms of lack of generality for dictionary-based approaches. On the other side, this experimental study has shown that some of the results previously obtained without pre-processing of opinion-bearing terms (see Section 3.4) can be improved without changing the summarisation approaches.

Four approaches for the pre-processing of opinion-based terms have been applied: opinions terms as stop-words, boosting frequencies, opinion-based bigrams and negation-based removal. Among these four approaches, the combination of opinion-based bigrams is the one showing better overall performances and consistent improvements over the regular runs (i.e. unigram-only). The other three approaches have shown both improvements and degradation of performances in different metrics.

4.4 Sentiment Classification via Subjectivity Detection

One of the main tasks in the field of Sentiment Analysis is the classification of opinionated documents according to the overall sentiment, i.e. whether positive or negative. A common behaviour

among reviewers is to summarise the overall sentiment of the review in a single sentence, or in a short passage. On the other hand, the rest of the review can express a feeling which is different from the overall judgement. This can be explained by the presence of several aspects or features that the reviewers want to comment on. The review shown in Figure 4.25 can be considered as an example. The review is taken from RottenTomatoes, a popular web-site which aggregates professional and non-professional reviews and comments about movies. The words or phrases carrying opinions are marked in italic. Several sentences express disappointment about different aspects of the movie, and simply counting the negative sentences would lead to classify the review as negative. The overall recommendation, described in the last sentence, is instead positive. It is also worth noting that some expressions, like "too easily", do not carry a negative sentiment per se, but must be put into context to be understood. In a similar way, terms normally related to negative feelings, like "trauma", are not used to denote a negative opinion.

I was particularly *disappointed* that the film didn't deal more with the trauma of learning one's life is a tv show [...] I almost felt that he got over it *too easily* for the sake of the film's pacing [...] Perhaps it's not fair to criticize a movie for what it isn't, but it seems like there were some *missed opportunities* here. But on its own terms, the movie is *well made*.

Figure 4.25: Example of review from RottenTomatoes. Opinion-oriented phrases are *emphasised.*

Moreover, often a review contains sentences which do not provide any information about opinions, i.e. they are not subjective. This is the case of movie reviews, where a short picture of the plot can be given to open the review, without commenting on it. Previous work has shown how the capability of identifying subjective sentences can improve the sentiment classification [Pang and Lee, 2004].

The main question investigated in this section is whether summarisation techniques can be applied for the purpose of understanding the polarity of a document. More specifically, the aim is to capture the summary passage, i.e. the short passage, or even the single sentence, which gives the overall sentiment of the review. From the user's perspective, the advantage of having a summarised review consists in a reduced effort to understand the message of the document, given that the key information is preserved. Traditional sentence extraction techniques can be applied for this task, although a more opinion-oriented approach is needed, since the goal is not to better describe the topic of the review in a single sentence, but to capture its overall polarity.

In order to verify whether the summarisation task preserves the information about the sentiment of reviews, text classification has to be performed on the original documents and on the produced summaries.

Figure 4.26 describes the pipeline for the movie review classification. The reviews can be classified directly (full text) or can be summarised in different ways. Firstly, through the summarisation component, sentence extraction based on statistical or positional approaches can be performed. Secondly, through the subjectivity detection component, objective sentences are filtered out, keeping all and only the subjective ones to form the summary. Thirdly, through a pipeline of both components, subjective extracts can be further summarised.

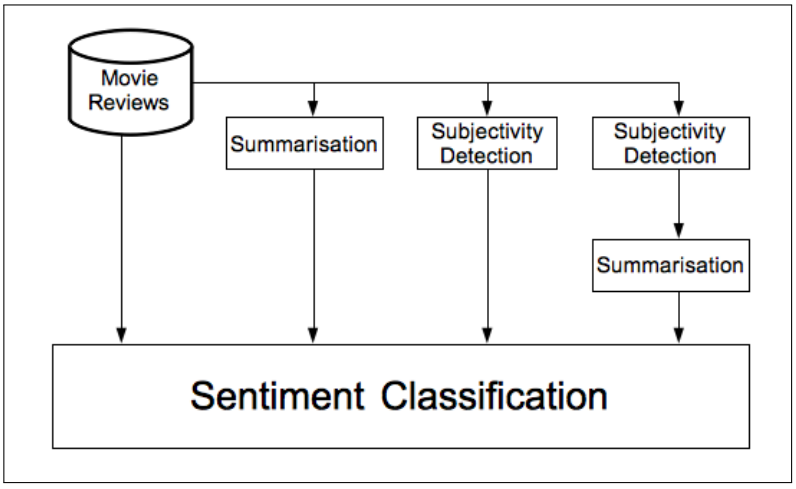


Figure 4.26: Pipeline of the review summarisation and classification

4.4.1 Sentiment Classification

Sentiment classification is a text classification task, where a label indicates the polarity of the document rather than its topic. The task can be approached from different points of view. For example, identifying the overall sentiment of a document is different from mining the polarity of individual aspects like soundtrack, plot, etc. In this work, only the polarity of the document as a whole is considered, i.e. whether the overall recommendation of a review is positive or negative. Section 2.3.6 has previously discussed different aspects of sentiment classification.

Traditional machine learning approaches, for example Support Vector Machine (SVM) or Naive Bayes (NB), can be applied for this classification task. Both the aforementioned approaches are families of machine learning models which belong to the class of supervised

learning, i.e. they require labelled training data to learn how to classify an unseen item. These classifiers have their own peculiarities and are well-known in text classification research [Caruana and Niculescu-Mizil, 2006]. It is important to note that the purpose of the work discussed in this section is not to achieve the best overall classification quality per se, but to validate whether subjectivity detection can be used as a means of sentiment summarisation, with the purpose of preserving the polarity information. From this perspective, off-the-shelf tools which implement traditional classifiers represent a valid baseline.

The next section discusses subjectivity detection as a means of sentiment summarisation.

4.4.2 Subjectivity Detection

Subjectivity detection is the task of identifying subjective sentences, i.e. sentences which carry opinions, as opposed to objective sentences, i.e. sentences which discuss facts. From this point of view, subjectivity detection can be regarded as a special case of text classification and hence it can be approached with traditional machine learning technique. Figure 4.27 shows examples of subjective and objective sentences.

Subjective	it's a very tasteful rock and roll movie .
	it is a film that will have people walking out halfway through
Objective	set on an island off the coast of florida
	then , in 1974 , something incredible happened

Figure 4.27: Example of subjective and objective sentences.

When applying summarisation to sentiment classification, one main issue is that topic-oriented summarisation approaches based on statistics do not take into account the subjective nature of the documents. In this sense, subjectivity detection can be used to identify subjective sentences, filtering out the objective ones. Looking at the subjectivity detection problem as a text classification task, the problem can be approached with traditional machine learning techniques like previously discussed in the context of sentiment summarisation. From this point of view, a dataset of labelled training data is needed when applying supervised learning methodologies such as SVM or NB. Section 4.5.2 will introduce the Subjectivity dataset, a collection of subjective and objective sentences which are used for the experimental study.

Given the availability of such dataset, the next step consists in applying it to train a classifier

which will be applied to the opinion-bearing content under analysis. Filtering out objective sentences from reviews and aggregating only the subjective ones can already be seen as a summarisation approach. Preliminary observations on a dataset of movie reviews (see Section 4.5.1 and Section 4.5.3) show that the subjective sentences account approximatively for the 50-60% of the full text reviews. In other words nearly half of the sentences are objective and can be seen as potential noise in the context of sentiment classification.

It is important to point out how this approach is particularly domain-specific. Similarly to the discussion about ambiguities in sentiment analysis (e.g. the polarity of term *unpredictable*), context is extremely important in subjectivity detection. The datasets mentioned in this section and used in the experimental study are both related to the movie domain, and this is what allows the classifier to be trained over one dataset and to be run over the other one. In other words, unexpected results can be observed when training the subjectivity detection classifier with data in the movie domain and then classifying unseen documents from, for example, a dataset of hotel reviews.

4.5 Evaluation of Sentiment Classification via Subjectivity Detection

Sentiment Classification is a particular case of Text Classification, where the classes do not represent the topic of a document, but the opinion expressed in it. Typically, documents are classified in either positive or negative, although different variations are possible (e.g. using also a "neutral" class, or using a rating system or "stars"). In this section, experiments in sentiment classification over a collection of movie reviews are reported. The results of the classification of full-text reviews are compared against the classification of summarised reviews. The purpose is not to achieve the best classification performance per se, but to verify whether the summarisation step preserves the polarity information (see Section 4.4).

4.5.1 Polarity Dataset

The Polarity Dataset [Pang et al., 2002, Pang and Lee, 2004] contains 2,000 documents manually chosen from IMDb[1] reviews. The main criterion to include a review in the dataset is whether a rating is expressed in terms of stars. Each document is a movie review written by an IMDb user

[1] http://www.imdb.com

on their website. Each document has been automatically labelled as either positive or negative according to its star rating, as described in [Pang et al., 2002], and the two classes are perfectly balanced. The original rating system used inconsistent formats (e.g. X out of 10, X out of 5) so the ratings have been normalised discarding the values in the middle (i.e. the "neutral" ones). The data are already lowercased and tokenised. The Polarity Dataset was firstly introduced in [Pang et al., 2002] and then extended in [Pang and Lee, 2004]. The latter version is the one used for these experiments. The main application of this dataset is sentiment classification. Figure 4.28 shows a sample of textual data, as well as the respective polarity annotation, from the polarity dataset.

Positive	the music is well-chosen and scored . and the pacing of the story was brisk ..
Negative	the whole film is really much ado about nothing . there is absolutely nothing scary about the story .

Figure 4.28: Sample of opinion-oriented data from the Polarity dataset.

4.5.2 Subjectivity Dataset

The Subjectivity Dataset [Pang and Lee, 2004] contains 10,000 sentences from the movie domain. The data have been automatically crawled from from IMDb plots and from RottenTomatoes user-generated comments. Two criteria have been used to include a sentence in the dataset. Firstly, sentences shorter than 10 words have been discarded. Secondly, only sentences published after the release of the Polarity Dataset, to avoid collisions, have been included. Each sentence is labelled in either objective (i.e. it does not carry any opinion) or subjective (i.e. it does carry opinions) and the two classes are balanced. The criteria used to build the collection consist in assuming that all the RottenTomatoes comments are subjective, while all the IMDb movie plots are objective. The data are already lowercased and tokenised. The subjectivity Dataset was firstly introduced in [Pang and Lee, 2004] and it is mainly used for subjectivity detection. Figure 4.29 reports a sample of textual data from the collection, as well as the subjectivity annotation.

Subjective	it's a very tasteful rock and roll movie .
	it is a film that will have people walking out halfway through
Objective	set on an island off the coast of florida
	then , in 1974 , something incredible happened

Figure 4.29: Sample of opinion-oriented data from the Subjectivity dataset.

4.5.3 Set-up

The sentiment classification experiments are run over the Polarity dataset described in Section 4.5.1. The data are organised in a way which allows for a 10-fold cross-validation as described in [Pang and Lee, 2004]. Therefore, the experiments are run 10 times using iteratively 90% of the collection for training and 10% of the collection for testing. The reported results are the average of the 10 runs.

The classification is performed using traditional machine learning techniques; specifically, Naive Bayes (NB) and Support Vector Machine (SVM) classifiers are considered. The classifiers use binary features, in particular term (unigram) presence, i.e. term frequencies and other statistics are not considered. The feature selection for NB is based on document frequency, being a commonly used selection strategy.

The main summarisation approach considered in these experiments is based on subjectivity detection. Other approaches are also considered, both independently (i.e. summarising the full text review) or combined to subjectivity detection (i.e. summarising only the subjective sentences). The considered techniques are the following:

- Luhn's traditional approach, as representative of statistical approaches;
- positional approaches, based on the intuition that the location of the sentence within the document reflects its significance;
- subjectivity detection, used to filter out sentences which do not express opinions;
- combinations of subjectivity detection with the other approaches.

Figure 4.30 shows the list of candidates for producing summaries.

Candidate	Description
Subjective	Only, and all, subjective sentences selected, via subjectivity detection
Luhn-N	*N* sentences selected via Luhn's approach [Luhn, 1958]
First-N	First *N* sentences selected
Last-N	Last *N* sentences selected
Subjective-Luhn-N	*N* sentences selected with Luhn's, after subjectivity detection
Subjective-First-N	First *N* sentences selected, after subjectivity detection
Subjective-Last-N	Last *N* sentences selected, after subjectivity detection

Figure 4.30: List of candidates for building summaries to use in the sentiment classification experiments.

Luhn's approach

As a representative of summarisation approaches based on statistics, The traditional Luhn's approach [Luhn, 1958] is used to score the sentences according to their significance. The top N sentences are selected to create the summary. The results for this approach are labelled as *Luhn-N*, where N is the number of sentence used to create the summary. The significance score of a sentence is based on clustering of sentence tokens using a distance threshold (5 is the one used in this study). For each cluster, the score is computed taking the ratio between the square of the number of significant words in the cluster, over the total number of words in the cluster. The significant words are chosen according to their frequency, i.e. the terms with higher TF, excluding stop words, are considered significant. The significance score for a sentence will be the maximum score for any of its clusters.

Position-based approaches

A different family of summarisers is built on top of an empirical observation: often reviewers tend to summarise their overall feeling in a sentence or in a short paragraph, placed either at the beginning or at the end of the review. In this case, a summary can be created simply selecting the N opening sentences, or the N closing sentences. Results for these approaches are labelled as *First-N* and *Last-N*, respectively.

Subjectivity detection

For the subjectivity detection, the Subjectivity dataset (see Section 4.5.2), consisting of subjective and objective sentences, is used to train the classifiers. This data-set contains 5,000 subjective sentences, taken from RottenTomatoes snippets, and 5,000 objective sentences, which are taken from IMDb plots. The main idea behind the creation of the subjectivity data-set consists in assuming that the review snippets from RottenTomatoes contain only opinionated sentences, while the movie plots taken from IMDb contain non-opinionated, and hence objective, sentences. Firstly, the classifiers are tested on the subjectivity dataset itself, using a five-folding cross-validation approach. The micro-averaged F_1 results are not substantially different between the two analysed classifiers (88.85 for NB vs. 88.68 for SVM). Given this preliminary observation, the classifiers can be considered reliable enough for the subjectivity detection task which leads to the generation of subjective extracts. The full Subjectivity dataset is then used as training data for the NB classifier, while the sentences from the documents to summarise are used as testing data. The sentences from the reviews labelled as subjective are aggregated to form the summary, while the objective sentences are simply discarded.

Combinations of approaches

As shown in Figure 4.30, all the previous approaches can be combined with the subjectivity detection. This is obtained by running the desired summariser over a subjective summary, i.e. over the set of sentences obtained after subjectivity detection.

4.5.4 Results

Figure 4.31 reports the results of the micro-averaged F_1 scores on the review data-set. This evaluation measure is chosen as it is one of the most commonly used in text classification [Sebastiani, 2002]. The macro-averaged results are not reported as they are very similar to the micro-averaged ones, given the dataset is well balanced, i.e. the two classes contain the same number of document.

The first observation is that statistics and positional summarisation approaches do not provide any improvement to the sentiment classification results for the full-text. On the contrary, the performances are substantially worse for both NB and SVM. Overall, picking the last sentences provide better results than picking the first ones or applying Luhn, both for subjective and non-

	NB	SVM		NB	SVM
Full Review	83.31	**87.10**	Subjective-Full	**84.61**	86.82
Luhn-1	70.12	70.28	Subjective-Luhn-1	71.02	70.50
Luhn-3	75.47	74.96	Subjective-Luhn-3	74.92	74.91
First-1	68.94	68.82	Subjective-First-1	69.33	68.90
Last-1	70.61	70.49	Subjective-Last-1	70.90	71.15
First-3	70.81	70.43	Subjective-First-3	71.12	71.07
Last-3	75.58	76.57	Subjective-Last-3	75.49	76.26

Figure 4.31: Micro-averaged F_1 scores for sentiment classification on the Polarity dataset.

subjective summaries.

The quality of sentiment classification for subjective extracts is instead in line with the full-text classification. More precisely, the classification of subjective extracts through NB achieves a 1.5% better result compared to the classification of full text. On the SVM side, the classification of subjective extracts is performed slightly worse than the classification of full text. In other words, the subjectivity detection step preserves the most important information about polarity, and this aspect is captured by both classifiers.

4.5.5 Analysis

There are two aspects to consider when discussing the quality of the classification results provided by summaries: firstly, subjectivity detection is clearly an opinion-oriented approach, while the other approaches are not explicitly tailored to opinions; secondly, the size of the summary matters. The full text reviews are in average 32 sentences long. Using subjectivity detection creates summaries which compress the reviews to an average of 50-60% the original size, i.e. approximatively 16-18 sentences. On the other side, the positional and statistical summarisation examined in these experiments propose summaries with only one sentence or three sentences. The intuition about the existence of a short passage, which describes the overall opinion of a review, possibly placed at the end of the review itself, can be empirically observed in some documents, but overall the experimental results do not support this idea.

In order to double check the importance of subjectivity, experiments on objective extracts clas-

sification have also been performed. The objective sentences have been aggregated, building the counterparts of the subjective extracts. The micro-averaged F_1 values for the objective extracts classification were below 75% for both classifiers, hence substantially worse than both the full review and subjective extract classification. When further summarisation is performed on the subjective extracts, the results drop again. On the two sides of Figure 4.31, a similar behaviour between summaries created from the full text and summaries created from the subjective extracts can be observed.

As further analysis, the classification of the individual summaries can be compared to the respective full-text documents. In other words, the question is whether the classifiers assign the same label to the full-text document and its respective summary, without considering the correctness of the label. In 91% of the cases, the subjective summaries are assigned to the same label of the correspondent full-text review. For all the other summarisation approaches, this value drops below 80%, and in some cases below 70%. This is further evidence of the connection between subjectivity and polarity.

4.5.6 Discussion

The aim of this section was to verify whether it is possible to summarise a review while preserving its overall polarity. Sentence extraction techniques purely based on statistical or positional approaches do not capture the subjectivity of a review, and hence are inadequate to summarise the overall sentiment expressed in the document. On the contrary, subjectivity detection produces results which are comparable to full-text classification. Further summarisation, based on statistical and positional features, applied on top of subjectivity detection, again fails to capture the polarity of a document. One of the intuitions explained in Section 4.4 regards the existence of a single sentence, or a short passage, which describes the overall opinion expressed in a review. Such intuition can be empirically observed in some of the analysed documents, especially towards the end of a review, but overall the experimental results do not support this idea.

The link between subjectivity and polarity is not only shown by the overall polarity classification results, but also by the comparison of the individual labels assigned by the classifiers to the full-text documents and the subjective summaries. In most of the cases, without considering the correctness of a specific label, the subjective summary is assigned to the same label of the respective full-text document. This is beneficial for a human reader because the polarity is preserved

despite the shorter length of the document, i.e. through the subjective extract, a user would need to read only ∼60% of the original review in order to understand its overall polarity.

4.6 Summary

This chapter has discussed problems and methodologies related to the treatment of opinions in the context of sentiment summarisation.

After the analysis on summarisation models proposed in the previous chapter, one question to answer is whether opinion-bearing terms require a special treatment in sentiment analysis application. A thorough discussion on the potential uses of dictionary-based pre-processing of opinion-bearing terms has been explained and applied to the task of sentiment summarisation. Dictionaries in sentiment analysis are fairly popular because they offer the benefit of being cheap to obtain or to generate. At the same time, dictionaries also present shortcomings. The main limitation consists in the importance of the domain or context: the same term can assume positive or negative connotations depending on the context.

The different treatments for opinion terms examined in this chapter include: considering opinion terms as stop-words, boosting frequencies of opinion terms, concatenating opinion terms with other local terms in order to obtain *n*-grams and dealing with negative qualifiers. The different pre-processing strategies have been evaluated on the intrinsic sentiment summarisation task, analysing if and how summarisation quality can be improved by treating opinion terms.

A different line of work has discussed subjectivity detection at the sentence level as a means to sentiment summarisation. As previously mentioned, summarising with respect to sentiment is different from summarising with respect to topic. Traditional machine learning approaches can be applied to subjectivity detection, i.e. the task of classifying sentences as subjective (opinion-bearing) or objective (not opinion-bearing). The outcome of such classification can be used to produce subjective extracts, which are summaries containing only those sentences labelled as subjective. The main question to answer is whether subjective extract can provide the same results in terms of polarity classification quality as the full-text documents. The benefit for users are immediate to understand: if the summaries convey the same overall polarity of the full-text, a user can recognise the polarity without the need for reading the whole text. The experimental study has confirmed this intuition.

Chapter 5

Knowledge-based Summarisation

5.1 Introduction

The previous chapters have focused on summarisation and opinion-oriented summarisation, motivated by the intention of leveraging the incredible amount of textual data available via web resources. The availability of high-quality sources of structured data and linked data, e.g. DBpedia[1], as well as the possibility of semantically annotating textual data, provide the ground for new applications in the context of summarisation.

This chapter focuses on the definition and modelling of knowledge-based summarisation. Different from sentence extraction summarisation, knowledge-based summarisation aims at constructing a summary by exploiting structured data. A knowledge base can be populated using the existing structure of a document (e.g. XML markup) or extracting facts from the free text of the document to summarise. External sources of knowledge can also be included to augment and improve the knowledge representation. The result of a knowledge-based summarisation process is the set of most relevant propositions about the topic being summarised. Examples of knowledge-based summaries include the set of most important facts described in a document, e.g. who-did-what, or the list of the top movies a given actor is known for, e.g. to answer questions such as "*who is Woody Allen?*". While the last example can be seen as related to Question Answering (QA), i.e. the task of answering questions posed in natural language, in the context of

[1] http://www.dbpedia.org

```
# Keyword-based query
?- M[woody allen
     actor director
     mia farrow];
```

```
# Knowledge-based query
?- M[actor(p1) &
     director(p1) &
     p1.name("Woody Allen") &
     actor(p2) &
     p2.name("Mia Farrow")];
```

Figure 5.1: Keyword-based query vs. knowledge-based query.

this thesis it is an example of Entity Summarisation, described in Section 5.4. Other applications where a knowledge-based approach to summarisation can be beneficial include aspect-based summarisation and contrastive summarisation.

This chapter contributes the definition of knowledge-based summarisation. Before discussing knowledge-based summarisation, Section 5.2 considers knowledge-oriented retrieval, drawing a parallel with traditional keyword-based retrieval and laying the ground for the knowledge representation used for knowledge-based summarisation. The process of knowledge-based summarisation is detailed in Section 5.3. Entity summarisation is discussed as an application scenario in Section 5.4 and evaluated in Section 5.5.

5.2 Knowledge Representation

Traditional IR models are based on *terms* or *keywords*. In keyword-based search, documents and queries are represented as bag-of-words. Such a flat representation does not expose the semantics beyond the text. In a knowledge-oriented approach, documents and queries are alternatively represented as *propositions*, i.e. as sets of classifications, relationships and attributes. With this richer representation, the concepts described in a document, and referred to in a query, can be made explicit. This leads to a deeper understanding of the meaning of the text. Figure 5.1 shows an example of how to formulate a keyword-based query compared to a knowledge-based one, using a logic-oriented syntax.

The keyword-based representation can be formulated in plain English as “retrieve the movies which contain the terms *woody*, *allen*, *actor*, ...”. There is no explicit semantics to explain what the terms in the query mean nor how the concepts are related. On the other side, in the knowledge-based representation the semantics is explicit. The query can be phrased as “retrieve the movies

directed by Woody Allen and in which both Woody Allen and Mia Farrow are actors". Using such a logic-based representation, the concepts are defined and related, therefore the knowledge-based representation is characterised by a deeper understanding of the text.

Chapter 2 introduced the Probabilistic Object Relational Content Model, shown in Figure 2.12. The benefits of such representation, discussed in Section 2.5.2 and in [Azzam et al., 2012], include the possibility of representing facts (classifications, relationships and attributes) and content knowledge (terms) in one coherent framework, as well as the possibility of defining knowledge-oriented retrieval models, i.e. models for IR which build evidence spaces not purely based on terms, but also on classifications, relationships and attributes. In this way, the consolidated data model can be effectively employed to integrate different types of knowledge from different data sources. At the same time, the data model can be exploited for retrieval or ranking purposes.

5.3 The Process of Knowledge-based Summarisation

This section discusses the overall process for building a knowledge-based summary. The process builds upon the generic model for text summarisation described in Section 2.2.1. In particular, during the initial text analysis step, knowledge extraction and knowledge augmentation can be performed, in order to produce a richer (semantic) internal representation of the source. The subsequent selection step will deal with propositions as well as terms.

5.3.1 Knowledge Extraction

Knowledge extraction tools can be used in order to generate classifications and relationships from natural language. For example, the sentence "Peter is a sailor" may be parsed by a knowledge extractor, and the output is "sailor(peter)", where the formal output shows that "sailor" is a class, and "peter" is an object, and a member of the class "sailor". In a similar way, relationships such as "peter.friendOf(mary)" are extracted, where a relationship can describe any semantic link between two objects (in this case, "peter" is the subject of the relationship, while "mary" is the object). Figure 5.2 shows an example of relationships extracted with ASSERT, an off-the-shelf shallow parser [Pradhan et al., 2004], where subjects and objects are in a relationship through a transitive predicate, which serves, in our example, also as relationship name. Some anaphoric

	Subject	Relationship	Object	Context
1	maximus	leads	roman army	/movieId/plot[1]
2	marcus	appoints	maximus	/movieId/plot[1]
3	commudus	kills	marcus	/movieId/plot[1]
4	commodus	claims	throne	/movieId/plot[1]
5	maximus	kills	commodus	/movieId/plot[1]

Figure 5.2: Example of semantic relationships extracted with ASSERT.

resolution, to disambiguate the meaning of "he", "she", "him", etc., is achievable by observing the subject and object of the previous relationships.

5.3.2 Knowledge Augmentation

One of the advantages of such a knowledge representation consists in the possibility of augmenting the knowledge through its integration with external knowledge bases. Resources like WordNet[2], DBpedia [Bizer et al., 2009] or other domain-specific ontologies can be exploited for this purpose. In feature-based sentiment analysis, a knowledge-based approach is particularly suitable, as the description of complex domains can benefit from the knowledge representation. Complex objects made of different components and attributes can be described in the knowledge base, and linked to opinion terms and phrases.

5.3.3 Summary Generation

Given a representation of a document consisting of the terms, classifications, and relationships extracted from the genuine document, the question is how to generate a summary from the knowledge-based representation.

Terms, classifications and relationships are seen as *propositions*. In traditional knowledge-based representations, only classifications and relationships are considered. A proposition-context-based representation that combines terms, classifications, and relationships is proposed in [Fuhr et al., 1998]. Classifications are arity-1 predicates, e.g. "sailor(peter)" is a classification, where "sailor" is the predicate name. Relationships are arity-2 predicates, e.g. "friendOf(peter,

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

mary)" is a relationship, where "friendOf" is the predicate name. The syntactic form "peter.friendOf(mary)" is just another form to express the relationship. Terms are arity-0 predicates; e.g. "sailing()" is a term, where "sailing" is the predicate, and the predicate has no argument. Classifications can also be expressed as relationships between an entity and a class. For example the sentence "Maximus is a Roman general" can be represented as "maximus.typeOf(general)", where "typeOf" is the relationship used to assign an entity (maximus) to its class (general).

Similarly to sentence extraction summarisation, statistical models can be used to rank propositions in order to obtain the most important ones to include in a summary. In this way, it is possible to define, for example, analogies of IDF for terms, classes, relationships and attributes, as shown in Figure 5.3 (examples in Probabilistic Datalog [Fuhr, 1995]). In this example, given the sample of knowledge in Figure 5.2, the proposition-based IDF can be exploited to retrieve the three most discriminative relationship involving "Maximus" as a subject of the relationships. Classifications and relationships can then be used to fill in templates, creating general-purpose summaries, or to show a personalised summary.

```
1  # Predicate−based IDF's:
2  pidf_term (Term)|MAX_IDF() :− term(Term, Context);
3  pidf_class (Name)|MAX_IDF() :− className(Name, Context);
4  pidf_relationship (Name)|MAX_IDF() :− relationshipName(Name, Context);


6  # Show the most discrimative relationships involving maximus
7  ?− relationship (Name, "maximus", Obj, Context) & pidf_relationship (Name);
```

Figure 5.3: PDatalog example of predicate-based IDF's.

Section 2.5.2 has introduced examples of knowledge-oriented retrieval models, which can be employed to rank propositions using the generic schema, which is application independent. On top of the generic schema which considers terms, classifications, relationships and attributes, new layers of relations with specific semantics can be built. This process is referred to as *semantic lifting* by Azzam and Roelleke [Azzam and Roelleke, 2011], as it "lifts" the basic classifications and relationships into more semantic propositions. Once the higher, more semantic, layers of the models are built, application-specific knowledge-oriented retrieval models can be derived and applied. Examples are discussed within the context of entity summarisation, introduced as a particular case of knowledge-based summarisation in the following section.

5.4 Knowledge-based Entity Summarisation

Entity summarisation is a major example of knowledge-based summarisation. The knowledge about a specific entity can be used to create entity profiles. For example, an athlete can be summarised by his/her top achievements in sport, a company can be represented by its top selling products, and an actor can be portrayed by the movies he is mostly known for.

This last movie-related scenario is employed as a case to discuss the semantic lifting introduced in the previous section, as well as to showcase the application of knowledge-oriented ranking models. The IMDb data-set, which will be discussed more in detail in Section 5.5, provides structural information which can be parsed into a relational representation as in Figure 5.4.

AttrName	Context	Content	MovieId
title	241272/title[1]	”Ocean’s Twelve”	241272
year	241272/year[1]	”2004”	241272
language	241272/language[1]	”English”	241272
genre	241272/genre[1]	”Comedy”	241272
actors	241272/actors[1]	”Clooney George”	241272
actors	241272/actors[2]	”Pitt Brad”	241272

Figure 5.4: Sample of knowledge representation of IMDb data.

For the purpose of this example, i.e. summarising the profile of movie-related people by means of the movies they are mostly famous for, most of the information from the IMDb data-set can be filtered out. Figure 5.5 shows an example of PD code which populates the `relationship` relation with information about actors, directors and other team members. This constitutes the first layer, or Layer-0, of the generic data model.

The process of semantic lifting then enriches the first, generic layer with more semantic relations, creating the Layer-1. Figure 5.6 shows two ways of creating semantically lifted relations. The first way generates the relations `actsIn`, `directorOf` and `isInTeamOf` by exploiting the value of `relationship(RelshipName)`, i.e. what was the value of the attribute `RelshipName` is now part of the schema as name of a new relation. The second way creates the new relation `activity` by filtering some values from `relationship`.

The newly created relations can be used to compute some basic statistics as well as tuple-based

```
1  # The relation imdb_structure_attribute contains all the structural information from IMDb.

3  # Actors:
4   relationship ('actsIn', Person, MovieId, 'imdb') :-
5            imdb_structure_attribute (actors, ElementId, Person, MovieId);

7  # Directors:
8   relationship ('directorOf', Person, MovieId, 'imdb') :-
9            imdb_structure_attribute (team1, ElementId, Person, MovieId);

11 # The rest of the team:
12  relationship ('isInTeamOf', Person, MovieId, 'imdb') :-
13           imdb_structure_attribute (team, ElementId, Person, MovieId);
```

Figure 5.5: IMDb example: basic relations (Layer-0).

```
1  # Semantic Lifting
2   actsIn (Person, Movie) :- relationship (actsIn, Person, Movie, DB);
3   directorOf (Person, Movie) :- relationship (directorOf, Person, Movie, DB);
4  isInTeamOf(Person, Movie) :- relationship (isInTeamOf, Person, Movie, DB);

6   activity_space {
7  (actsIn);
8  (directorOf);
9  (isInTeamOf);
10 };
11  activity (RelshipName, Person, Movie) :-
12           relationship (RelshipName, Person, Movie, IMDB) &
13           activity_space (RelshipName);
```

Figure 5.6: IMDb example: relations obtained through semantic lifting (Layer-1).

probabilities, obtained by aggregation over the relations at Layer-1, as shown in Figure 5.7.

As Figure 5.6 and Figure 5.7 illustrate, the data model is now more application-specific. Given the new, more semantic relations, the process can continue defining higher layers of semantic data on top of the lower ones. For example, for task of generating the profile of an actor, it is interesting to know in which *popular* movies he has acted in. Concepts such as *popular movie* or

```
# Basic stats
sum_activities_per_person DISTINCT(sum($0), Person) :-
        activity(Activity, Person, Movie);
avg_activities DISTINCT(avg(NumActivities)) :-
        sum_activities_per_person(NumActivities, Person);
# Activity-related probabilities
p_activity_given_person SUM(Activity, Person) :-
        activity(Activity, Person, Movie) | (Person);
p_activity_person SUM(Activity, Person) :-
        activity(Activity, Person, Movie) | ();
activity_person_freq SUM(Activity, Person) :-
        activity(Activity, Person, Movie);
p_activity SUM(Activity) :-
        activity(Activity, Person, Movie) | ();
p_activity_max_itf (Activity)|MAX_ITF() :-
        activity(Activity, Person, Movie);
# Person-related probabilities
p_person SUM(Person) :-
        activity(Activity, Person, Movie) | ();
p_person_given_movie SUM(Person) :-
        activity(Activity, Person, Movie) | (Movie);
p_person_max_itf (Person)|MAX_ITF() :-
        activity(Activity, Person, Movie);
# Movie-related probabilities
p_movie_given_person SUM(Movie) :-
        activity(Activity, Person, Movie) | (Person);
p_movie SUM(Movie) :-
        activity(Activity, Person, Movie) | ();
p_movie_max_itf (Movie)|MAX_ITF() :-
        activity(Activity, Person, Movie);
```

Figure 5.7: IMDb example: stats and probabilities from Layer-1.

popular actor can be defined one level above Layer-1. Figure 5.8 presents one option to represent such concepts. Other options to delineate the idea of popularity include the possibility of merging external data, e.g. box office revenue (a movie is popular if many people watch it) or ratings from reviews (a movie is popular if many people like it).

```
1  hasActivityIn_all (Person, Movie) :-
2          activity (A, Person, Movie);
3  hasActivityIn DISTINCT(Person, Movie) :-
4          activity (A, Person, Movie);

6  # A person is popular if he/she has activities in many movies.
7   p_Activity_person_is_popular SUM(Person) :-
8          hasActivityIn_all (Person, Movie) |() ;
9  # A person is popular if he/she has at least one activity in many movies.
10 p_Movie_person_is_popular SUM(Person) :-
11         hasActivityIn (Person, Movie) |() ;
12         #Note the difference between has_activity_all and has_activity .
13         #The first one considers several activities per movie ( activity freq ),
14         #the second one considers aggregates the activities (movie freq ).

16 # Select a default event space (here, Activity ):
17  p_person_is_popular (Person) :-
18         p_Activity_person_is_popular (Person);

20 # A movie is popular if many popular persons have activities in it .
21 p_movie_is_popular SUM(Movie) :-
22         hasActivityIn (Person, Movie)|(Movie) &
23         p_person_is_popular (Person);
```

Figure 5.8: IMDb example: more semantic relations on Layer-2.

The final aspect to consider in order to generate entity summaries is how to proceed with the fact ranking. While Figure 5.7 has already shown the formulation of some probabilities base on different evidence spaces, Figure 5.9 proposes some further formulations.

Once the definitions of the different probabilistic components are in place, the notion of "top facts to know" can be constructed. Figure 5.10 illustrates several candidates.

To provide an example from the IMDb data-set, Figure 5.11 shows the summary/profile for the entity *Leonardo DiCaprio*, obtained running a TF-total-like approach with the following query:

```
?- top_facts_total("Leonardo DiCaprio", Activity, MovieId) &
movie_title(MovieId, MovieTitle);
```

```
1  # BM25-alike quantification of frequencies

3  0.01 inv_avgdl();
4  0.01 inv_avg_activities_per_person();
5  p_activity_person_frac(Activity, Person) :- p_activity_person(Activity, Person) FRAC & inv_avgdl();
6  p_activity_given_person_frac(Activity, Person) :- p_activity_given_person(Activity, Person) FRAC &
       inv_avg_activities_per_person();

8  1.0 one();
9  p_activity_person_frac(Activity, Person) :- activity_person_freq(Activity, Person) FRAC & one();
10 activity_person_frac(Activity, Person) :- activity_person_freq(Activity, Person) FRAC & one();
```

Figure 5.9: IMDb example: BM25-like probability and frequency formulation.

5.5 Evaluation

Entity summarisation is a particular case of the task of knowledge-based summarisation introduced in this chapter. The purpose is to create a short profile of a particular entity from a knowledge base which contains several facts about several entities. The scenario chosen to evaluate different ways to pick the top-facts about an entity is given by the IMDb data-set, which contains information about movies, actors, directors and other people working in the movie industry.

5.5.1 IMDb Dataset

The IMDb dataset is a collection of movie-related data, offered by the popular Internet Movie Database website in textual form[3]. The collection is a subset of the actual data available on the website, and it was firstly introduced by Kim et al. within the context of known-item search [Kim et al., 2009]. Kim et al. provided a test-bed which includes 50 queries and relevance assessment, which are not used in the context of this research. The collection contains approximatively 437,000 documents formatted in XML, each of which represents a movie. For each movie, information such as title, year, genre, actors and team members are available. Figure 5.12 shows a sample of the raw data, while Figure 5.13 shows a sample of the knowledge-oriented representation.

[3] http://www.imdb.com/interfaces#plain

```
1  # "TF_total"
2   top_facts_total (PersonId, Activity, MovieId) :-
3              activity (Activity, PersonId, MovieId) &
4              p_activity_person (Activity, PersonId) &
5              p_activity_max_itf (Activity) &
6             p_movie(MovieId) &
7             p_movie_is_popular (MovieId);

9  # ranking-equivalent to "TF_total"
10  top_facts_total_freq (PersonId, Activity, MovieId) :-
11             activity (Activity, PersonId, MovieId) &
12             activity_person_freq (Activity, PersonId) &
13             p_activity_max_itf (Activity) &
14            p_movie(MovieId) &
15            p_movie_is_popular (MovieId);

17 # "TF_total" with P( activity |person)
18  top_facts_total_given_person (PersonId, Activity, MovieId) :-
19             activity (Activity, PersonId, MovieId) &
20             p_activity_given_person (Activity, PersonId) &
21             p_activity_max_itf (Activity) &
22            p_movie(MovieId) &
23            p_movie_is_popular (MovieId);

25 # "TF" frac
26  top_facts_frac (PersonId, Activity, MovieId) :-
27             activity (Activity, PersonId, MovieId) &
28              p_activity_given_person_frac (Activity, PersonId) &
29             p_activity_max_itf (Activity) &
30            p_movie_is_popular (MovieId);
```

Figure 5.10: IMDb example: definition of top-facts about movie-related people.

5.5.2 Set-up

The experiments are run over a portion of the IMDb collection, which has been introduced in Section 5.5.1. The collection has been parsed into a relational form exploiting its XML-based structure (sample previously presented in Figure 5.13). From this tabular representation, a sam-

#	Activity	Movie ID	Movie Title
1	isInTeamOf	24996	”Aviator The”
2	isInTeamOf	373869	”Wolf of Wall Street The”
3	isInTeamOf	283898	”Rise of Theodore Roosevelt The”
4	actsIn	41461	”Boffo! Tinseltown’s Bombs and Blockbusters”
5	isInTeamOf	950	”11th Hour The”

Figure 5.11: IMDb example: summary generated for the entity “Leonardo DiCaprio”.

```
<movie id="241272">
   <title>Ocean's Twelve</title>
   <year>2004</year>
   <language>English</language>
   <genre>Comedy</genre>
   <genre>Crime</genre>
   <genre>Thriller</genre>
   <actors>
      <actor>Clooney, George</actor>
      <actor>Pitt, Brad</actor>
   </actors>
   <team>
        <director>Soderbergh, Steven</director>
   </team>
</movie>
```

Figure 5.12: Sample of XML data from IMDb.

AttrName	Context	Content	MovieId
title	241272/title[1]	”Ocean’s Twelve”	241272
year	241272/year[1]	”2004”	241272
language	241272/language[1]	”English”	241272
genre	241272/genre[1]	”Comedy”	241272
actors	241272/actors[1]	”Clooney George”	241272
actors	241272/actors[2]	”Pitt Brad”	241272

Figure 5.13: Sample of knowledge representation of IMDb data.

ple of all the activities performed by 61 movie-related people has been extracted, creating a knowledge base of 4,113 activities in total. 10 actors have been arbitrarily selected as entities to summarise. The list of chosen entities is shown in Figure 5.14 for reference.

Entity	Description	Entity	Description
e_1	Woody Allen	e_6	Cameron Diaz
e_2	Pierce Brosnan	e_7	Leonardo Di Caprio
e_3	Russell Crowe	e_8	Mia Farrow
e_4	Matt Damon	e_9	Brad Pitt
e_5	Robert De Niro	e_{10}	Martin Scorsese

Figure 5.14: List of entities chosen for the entity summarisation task.

Since golden standard summaries for this particular task do not exist, two options have been considered to produce a set of reference summaries to compare with: firstly, the IMDb website has been crawled, storing the "known for" information from the actors' individual pages, including 4 movies per actor; secondly, a list of favourite movies has been compiled by a human judge, including between 2 and 5 movies per actor.

The summary size can be set to an arbitrary number of top facts to retrieve. Two different approaches are employed to choose the evaluation metrics. Firstly, the whole set of facts for each actor is retrieved. The quality of the ranking is evaluated using Mean Reciprocal Rank (MRR) of the first fact which appears in a golden standard summary (*à la* known-item search, with several known-items, but only the first is considered). MRR averages the Reciprocal Rank (RR) scores across all the queries (entities in this case), as described in Equation 5.1, where $E = \{e_1, e_2, ..., e_n\}$ represents the set of entities to summarise and $\texttt{rank}_i$ represents the ranking position of the first known fact for the entity e_i.

$$MRR := \frac{1}{|E|} \sum_{i=1}^{|E|} \frac{1}{\texttt{rank}_i} \tag{5.1}$$

Secondly, the summary size has been set to 10 facts per actor, and the set of retrieved facts is compared to the golden standard summaries using precision and recall (comparable to using Precision@10 and Recall@10).

Two different candidates for ranking the facts and creating the summaries have been analysed.

The candidates are different variations of the "top facts" ranking, previously described in Section 5.4. In particular, the ranking based on "$\text{TF}_{\texttt{total}}$" and "$\text{TF}_{\texttt{frac}}$" are considered. Figure 5.15 summarises the candidates.

Candidate	Description
Total	Ranking based on "$\text{TF}_{\texttt{total}}$" as in Figure 5.10
Frac	Ranking based on "$\text{TF}_{\texttt{frac}}$" as in Figure 5.10

Figure 5.15: List of candidates for the experiments on entity summarisation.

5.5.3 Results

Figure 5.16 shows the numerical results for the entity summarisation task run against the human-judged golden standard summaries and against the IMDB "known for" golden standard summaries.

Judge	Candidate	MRR	P@10	R@10
Human	$\text{TF}_{\texttt{total}}$	**38.66**	**12.00**	**40.66**
	$\text{TF}_{\texttt{frac}}$	23.16	09.00	35.66
IMDb	$\text{TF}_{\texttt{total}}$	28.20	09.00	20.00
	$\text{TF}_{\texttt{frac}}$	**40.31**	**11.00**	**27.50**

Figure 5.16: Results over the IMDb data-set for the entity summarisation task. The best results are highlighted in **bold**.

The two different sets of golden standard summaries lead to opposite results: for the human-judged summaries, $\text{TF}_{\texttt{total}}$ performs better than $\text{TF}_{\texttt{frac}}$ for all the different metrics; on the other side, for the IMDb-based references, $\text{TF}_{\texttt{frac}}$ performs better.

5.5.4 Analysis

In addition to the obvious observation that different judgements can lead to opposite results, the relatively small size of the data-set and of the result set allows for an inspection at a smaller granularity.

Given the pool of IMDb-based judgements, it is interesting to observe a particular situation where the $\text{TF}_{\texttt{total}}$ performed much worse than the $\text{TF}_{\texttt{frac}}$. This is the case, for example, of entity e_9

(Brad Pitt). The Reciprocal Rank (RR) produced for this query by $TF_{\texttt{frac}}$ is 100% (i.e. $\frac{1}{1}$), i.e. the fact ranked in first position is one of the facts identified in the golden standard summaries. On the other side, the $TF_{\texttt{total}}$ ranks the same fact in position 38, producing a RR of only 2.63% (equal to $\frac{1}{38}$). The position of this fact also impacts the other two metrics, Precision@10 and Recall@10, as the fact is ranked outside the top-10 so its contribution is zero.

5.5.5 Discussion

The aim of this section was to set-up an evaluation for the novel task of knowledge-based summarisation, analysing the particular case of entity summarisation.

In order to perform this experimental study, a subset of the IMDb test collection has been assembled with a focus on relationship between movie-related persons and movies. Two different options for golden standards are provided. This section has provided quality measures for the different summarisation/ranking candidates, although rather than verifying quality per se, it is probably more interesting to open a discussion about the ability of knowledge-based technology to tackle "complex" needs.

The generic data model discusses in Chapter 2 and applied in Section 5.4 for entity summarisation is application agnostic, in the sense that it provides the foundations on top of which one can model application-specific data and ranking functions. Through this data model, different and possibly inconsistent data sources can be integrated into one congruent framework. It provides a facility to rapidly develop mashups which can be tailored for application-specific needs at the higher, more semantic, levels.

Within the movie scenario, an example of complex need is how to represent popularity. The concept of popularity is subjective and difficult to measure. For example, one could use box office revenue as evidence of popularity, or ratings from a review web-site. The approach proposed in this section popularity by using the available data, i.e. a person is popular is he/she works in many movies, and a movie is popular is many popular persons work in it. The nature of the data model, as well as the descriptive approach shown in the examples with the use of a declarative language like Probabilistic Datalog, promote and simplify the process of quickly building semantic components on top of the basic relations. In this way, tasks such as representing popularity can be easily personalised and adjusted to the user's needs.

5.6 Summary

This chapter has discussed the definition and modelling of knowledge-based summarisation, which aims at building a summary by exploiting a knowledge base. The output of a knowledge-based summariser is the set of the most important facts about the topic to summarise.

The first component which enables knowledge-based summarisation is the knowledge representation. This chapter has discussed the use of a generic data model introduced in [Azzam and Roelleke, 2011] and shown to be successful for knowledge-oriented retrieval [Azzam et al., 2012]. The data model constitutes a coherent framework to merge knowledge from different sources. The basic relations from the generic data model can be involved in the generation of more semantic, application-specific details. Entity summarisation is a specific application of this technology, as the knowledge about a specific entity can be used to create entity profiles. For example, an athlete can be summarised by his/her top achievements in sport, a company can be represented by its top selling products and an actor can be summarised by mentioning the movie he is mostly known for. This chapter has showcased entity summarisation in the context of movie-related people. The task was to create a short profile of actors and directors by ranking the "top facts" about them, i.e. by ranking the movies they are famous for.

Sentiment adds an extra dimension to this representation. For example, in the sentence "Maximus is a *brave* general", the term "brave" is expressing a positive polarity on the class, while in the sentence "Peter is a *good* friend of Mary" the sentiment is related to the relationship. A more interesting and challenging scenario consists in having a brilliant performance by an actor who is playing an evil character. A review commenting on this aspect is clearly prone to ambiguities. In a knowledge-oriented approach the sentiment dimension would be inferred to the correct class, leading to the understanding of the sentiment carried by the review. The next chapter engages in details with the problem of knowledge representing of opinions.

Chapter 6

Knowledge Representation of Opinions

6.1 Introduction

This chapter discusses the conceptual modelling and knowledge representation of opinions. Conceptual design is an important step in the database design process, which produces a high-level data model, entirely independent from implementation details. Conceptual schemata provide benefits in terms of data independence, design aid and connections with the enterprise world [Zaniolo and Melkaoff, 1982]. Entity-Relationship (ER) modelling and its graphical facility [Chen, 1976] are widely used as a standard tool in the data/application design process.

The basic concepts in ER design (diagrams) are entities, relationships and attributes. Entities correspond to real-world objects (for example, *Employee* and *Department*). Relationships are concepts that connect entities (for example, *works_in*(Employee, Department)).

Entities and relationships usually have attributes (e.g. Employee.Name). Traditional ER models can be extended in order to capture additional concepts such as aggregation and generalisation [Smith and Smith, 1977]. Enhanced Entity-Relationship (E-ER) models [Connolly and Begg, 2005] allow to model superclasses/subclasses (generalisation and specialisation), aggregation (*is_part_of* relationships) and composition (an aggregation with a strong ownership between the whole and the part). A subclass is a specialisation of a superclass (and a superclass is a generalisation of a subclass, e.g. Manager *is_a* Employee). A subclass inherits the attributes of a superclass. Aggregation is a whole-part relationship between objects (e.g.

Lens *is_part_of* Camera). The E-ER model reduces the gap between object-oriented concepts and traditional data modelling.

As discussed throughout this thesis, many of today's applications require the representation of opinions. Similar to the issue regarding object-oriented concepts, there is a gap between opinion-aware requirements and conceptual data modelling. Therefore, this chapter investigates the conceptual representation of opinions to support the development of applications such as sentiment analysis.

The starting point for the development of opinion-aware applications is the correct representations of the opinions themselves. Traditional ER/E-ER models can be utilised for this purpose, but the semantics of opinions are not captured in their essence. In particular, opinions are typically represented as additional attributes or subclasses, implicitly suggesting how opinions will be represented on the logical layer.

At the conceptual layer, the design of opinions should discuss what the opinions are about rather than how to represent them. For this reason, in order to integrate opinions into traditional object-oriented and object-relational modelling, a methodology to extend ER/E-ER models is proposed. The aim is to enrich the data representation to capture the semantics of opinions and feelings. Scenarios such as *good student*, *good friend* or *good price* (student is an entity, friend is a relationship, price is an attribute) are modelled through the notion of polarity. Graphically, a specific notation is proposed in order to augment E-ER models and capture polarity.

The main contributions discussed in this chapter are:

1. a methodology to clearly separate the representation of opinions between conceptual (*what* to represent) and logical (*how* to represent it) layer, which enriches E-ER modelling with:
   (a) a very **high-level** conceptual specification about which entity, attributes and relationships require opinion-aware reasoning;
   (b) **additional semantic** information about the opinions;
2. a **mapping** from the opinion-aware conceptual layer to the logical (i.e. relational) layer.

The remainder of the chapter is organised as follows. Section 6.2 discusses the problem of conceptual modelling of opinion and proposes how to include opinions into E-ER modelling. Section 6.3 explains the mapping of a conceptual model to a logical layer. The end of this section

(Section 6.3.4) engages with technical details regarding the logical model (e.g. SQL concepts to support modelling of opinions), and some of these details go beyond what constitutes the main contribution, namely a *conceptual model for opinions* with a well-defined *mapping process*. The technical details of the effect on the logical layer, however, are presented in this chapter in order to be comprehensive and underline why the semantic modelling of opinions is not just desirable but even required. Section 6.4 showcases the development of best practice for opinion-oriented modelling in sentiment analysis applications. Section 6.5 deepens the general discussion about the need for semantic modelling of opinions. Section 6.6 concludes the chapter.

6.2 Conceptual Modelling of Opinions

This section discusses the conceptual modelling of opinions.

Section 6.2.1 shows the modelling of a system which stores documents expressing opinions, where there is a lack of semantics of the opinions and of the world of the objects which the opinions are about.

Section 6.2.2 proposes the modelling of opinions within the respective entities, using traditional E-ER components. In this way, the previous lack of semantics is partially overcome, but implementation details are introduced at the conceptual level.

Section 6.2.3 proposes an extension of ER/E-ER models in order to integrate the semantics of opinions at the conceptual layer, keeping it untied from implementation details.

The diagrams follow the notation introduced in Figure 2.11.

6.2.1 Modelling a Review System with Traditional E-ER Models

Sentiment Analysis applications deal with documents which carry opinions about a particular target. A document can be, for example, a review or a comment. A target is in general anything a person can review on, even another document. When commenting on a review, the comment might express agreement or disagreement with the review itself (i.e. the review would be the target of the comment).

Figure 6.1 shows a simplified example of an ER model of a system which stores documents expressing opinions about targets. For both documents and targets, multiple subclasses can be included (for simplicity, only two per side are shown). The participation in the hierarchy can

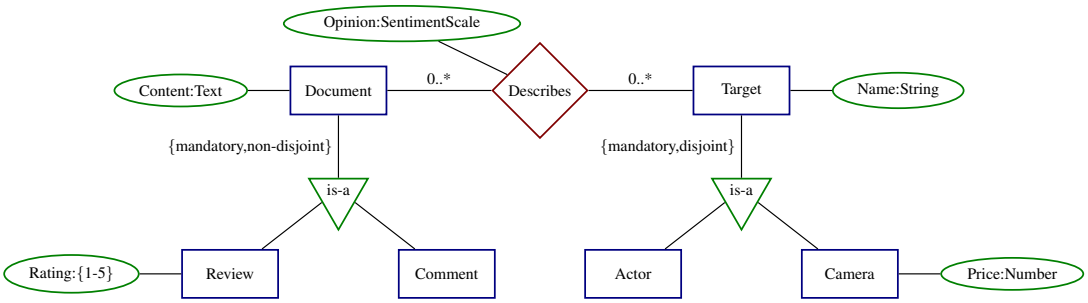


Figure 6.1: Traditional ER model of documents expressing opinions about targets. Different targets include cameras and actors, different documents include reviews and comments. Additional subclasses for targets and document types, as well as extra attributes can be included (here omitted for simplicity).

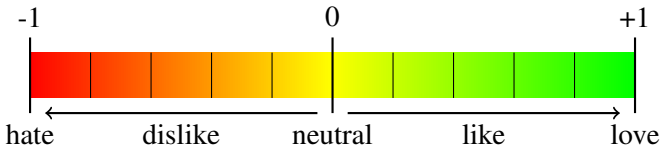


Figure 6.2: Sentiment Scale.

be assumed to be mandatory for both sides, non-disjoint (and) for documents and disjoint (or) for targets. Additional details are also not included for simplicity, for example extra attributes, relationships with other entities (e.g. users/authors of documents) or the *Document is_a Target* relationship (as discussed above, reviews can discuss other reviews). It is worth noting that, while a review is typically associated to a specific item (e.g. a movie), it can also contain comments about different targets (e.g. actors, soundtrack, other movies, other reviews, etc.), hence the use of a many-to-many relationship to express opinions. Such a representation at the conceptual layer can easily be translated into a logical representation such as:

```
1 review(DocID, AuthorID, ReleaseDate, Content, ...)
2 describes(DocID, TargetID, Opinion)
3 camera(TargetID, Name, Price, ...)
4 actor(TargetID, Name, DateOfBirth, ...)
```

If *Opinion* is defined on the sentiment scale of $[-1,+1]$, one can decide a threshold of "goodness" in order to retrieve the best products. At the implementation layer, a natural language query like "retrieve cameras with positive/good reviews" can be translated into queries such as:

```
1 -- Using numeric/arbitrary threshold
```

```
2  SELECT TargetID, Name
3  FROM camera JOIN describes ON (target.TargetID=describes.TargetID)
4  WHERE Opinion >= 0.7 -- arbitrary threshold
5  ORDER BY Opinion DESC


7  -- Using vague predicates
8  SELECT TargetID, Name
9  FROM camera JOIN describes ON (target.TargetID=describes.TargetID)
10 WHERE Opinion IS GOOD -- vague predicate
11 ORDER BY Opinion DESC
```

Although the implementation of such a system seems straightforward, a key aspect is missing in the conceptual layer: the *essence* of an opinion. Moreover, such an implementation does not provide any semantics about the world of the target, but rather describes a system which deals with documents about the target.

6.2.2 Modelling Opinions with Traditional E-ER Concepts

A different modelling approach consists in integrating opinions within the entities that are being represented, using traditional ER or E-ER concepts. For example, in a scenario related to electronic products, a camera would be one of the possible targets. In the modelling of opinions related to cameras, one can identify *good camera* and *bad camera* as subclasses of the entity camera. A different point of view could consist in interpreting the overall quality of the camera as an attribute of the entity itself. Figure 6.3(a) and Figure 6.3(b) show these two possible representations, providing illustrations for the camera scenario.

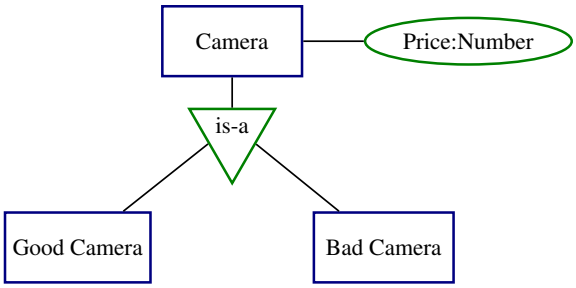
For the example in Figure 6.3(a), the mapping to the logical model can be derived as follows:

```
1  good_camera(CameraID, ModelName, ...)
2  bad_camera(CameraID, ModelName, ...)
```

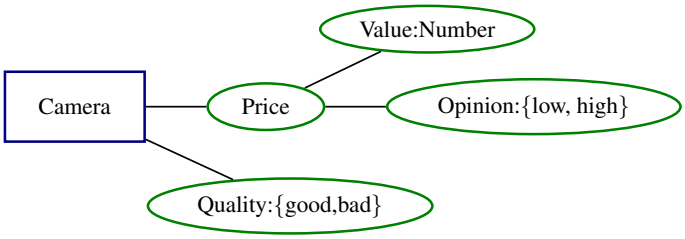
On the other side, representing the opinion as an attribute, like the model in Figure 6.3(b), would yield the logical model as follows:

```
1  camera(CameraID, ModelName, Price, ..., Quality)
```

The key difference between the two options consists in modelling the opinion as part of the schema, i.e. the opinion is embedded in the schema, or as an instance of the data. This difference



(a) Opinions modelled as subclasses.



(b) Opinions modelled as attributes.

Figure 6.3: Traditional ER model expressing opinions about entities and attributes. Artificial components break the "flow" of the conceptual (semantic) model.

is also transferred on the query side, e.g. in SQL:

```
1  -- Opinion embedded in the schema
2  SELECT *
3  FROM good_camera;
4  -- Opinion as instance of the data
5  SELECT *
6  FROM camera
7  WHERE quality='good';
```

The discussion so far shows that on one hand there is no explicit support for opinion-aware concepts on the conceptual layer, and on the other hand, with regards to traditional techniques, there are many ways to model opinions. In order to achieve a consolidated and guided modelling approach, new opinion-aware concepts are required.

6.2.3 Integrating New Opinion Concepts into E-ER Diagrams

More articulated conceptual models could be proposed, in order to capture more aspects of opinions. For example interesting details include who the opinion holder is (e.g. different people have different opinions on the same target), or when the opinion has been expressed (e.g. opinions can

change over time). Extending the movie scenario, one could also consider the fact that the same actor can perform brilliantly or poorly in different movies, defining overlapping specialisation in the subclass-based example, or allowing multi-valued attributes in the attribute-based example. Nevertheless, there is still a lack of meaning around the concept of opinions, i.e. all these models suggest *how* to represent opinions. In this research, it is argued that there is a need for a deeper representation of opinions and for a separation between *how* and *what*. In other words, it is suggested that at the conceptual layer, designers can specify which opinions will be considered (the "what"), while implementation details (the "how") will be encapsulated within the logical layer. In order to fulfil such a need, different levels of opinion-related information are identified and modelled by the proposed extension.

The first level, or **Level-0** (L0), is used to declare which entities, attributes or relationships can be target of opinions. The graphical extension to obtain a L0 Opinion-based Enhanced ER consists in marking the target object (entity, attribute or relationship) with a **polarity label**. In particular, a $\pm$ symbol is placed next to the object name. The polarity label represents the presence of a subjective view, on a positive-vs-negative scale, about the object on which it is placed. For the attributes, the subjective view is expressed on the value of the attributes. For example, the price of a camera is represented as a number, but a customer could consider it a good price or a bad price despite its absolute value. For entities and relationships, the subjective view is expressed on the object as a whole, either as an aggregate of the opinions on different attributes, or as an implicit generic representation of the main qualities which characterise the entity. For example, the notion of good actor or bad actor implicitly refers to the acting skill, or the talent, of the actor. Figure 6.4 shows the camera example represented with a L0 diagram (Figure 6.5 shows the legend, including L1 symbols). In this example, two opinions are expressed: an opinion on the entity Camera as a whole and an opinion on the attribute Price. The opinion on the entity as a whole can be seen as the equivalent translation of the previous examples where subclasses (Good Camera and Bad Camera) or artificial attributes (Quality) were employed.

Level-0 diagrams do not model any additional detail about the opinions, as they simply state that opinions are considered in the model. If extra information is available, a different level of opinion-based diagrams is used to model such information.

Level-1 (L1) diagrams attach the description of opinion meta-data to those objects that are marked with polarity labels. In particular, the designer can specify naming conventions for the

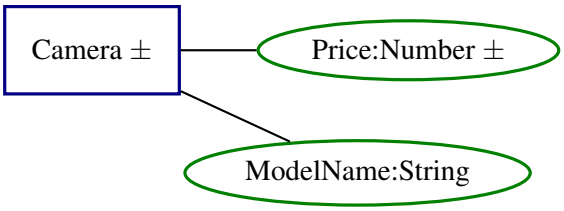
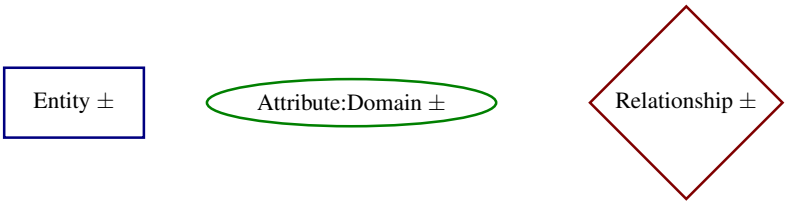
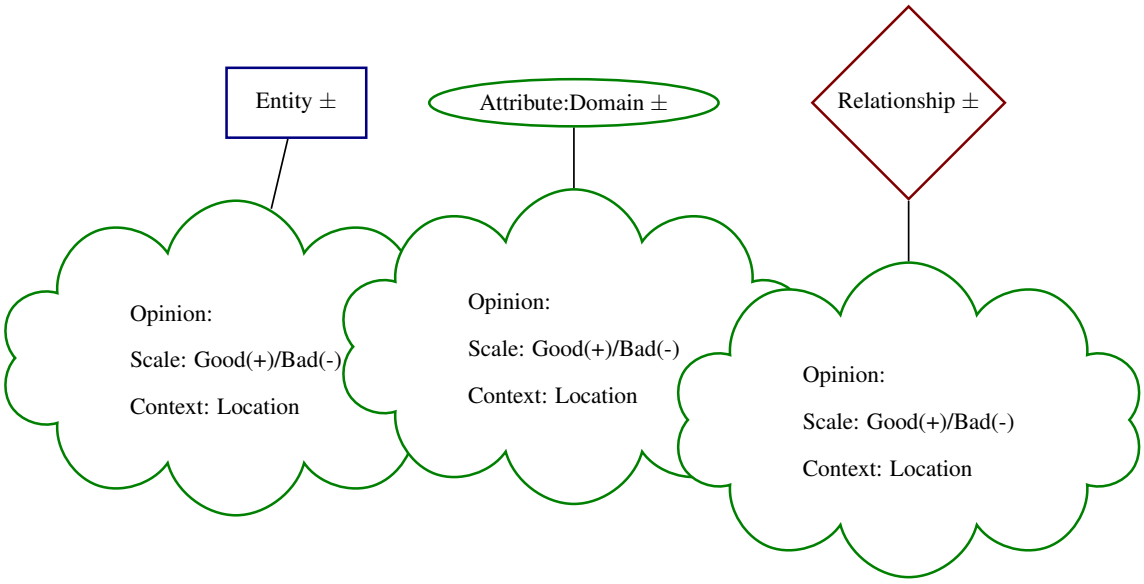


Figure 6.4: First example of an Opinion-extended ER Diagram (Level-0 model). The diagram shows how the requirements for opinion-aware reasoning are declared at the conceptual layer. Opinions are expressed on the entity `Camera` as a whole and on the attribute `Price`.



(a) Level-0: polarity labels only.



(b) Level-1: polarity labels and polarity tags.

Figure 6.5: **New Symbols** proposed for Opinion-extended ER Diagrams. Entity, attribute and relationship are expressed in a Level-0 fashion in Figure 6.5(a), where **polarity labels** are placed next to the component names. Figure 6.5(b) shows Level-1 ER components, with **polarity tags** attached to the respective components in the form of *thought-balloons*. The polarity tags for Level-1 components show the default values.

opinions, the domain of the opinion (i.e. the scale with the two extremes of the polarity), and information about the context where the opinions are expressed.

The default name for opinionated components is simply "Opinion", and the domain of opinions is by default considered to be on a good-vs-bad scale. Subjectivity can truly be expressed on different polarised scales, for example sweet-vs-bitter or easy-vs-hard. In some cases, there is no explicit concept or mention of desirability of one of the extreme against the other. For example, while it is reasonable to assume that a movie-goer would choose a good movie instead of a bad one, knowing whether a pub-goer is looking for a sweet cocktail or a bitter one is not so obvious.

The polarised components in ER diagrams are enriched using **polarity tags** to attach meta information, lifting the model to Level-1. Polarity tags are graphically represented as thought-balloon ellipses. Figure 6.6 shows how to attach the meta-information to components marked with polarity labels. For the entity *Camera*, the opinion is expressed on the overall quality, hence the term "Quality" used to name such opinion. The quality of a camera is described on a good-vs-bad scale, with good being the positive extreme and bad being the negative one. The context in which the opinion is given is a Review DB. For the attribute *Price*, the opinion is expressed on its value. The price is described as low (positive, good) or high (negative, bad). The context in this case is the same Review DB. Given the expressiveness of natural language, the polarity low-vs-high for the price could also be described in a number of different ways, for example cheap-vs-expensive, or simply good-vs-bad. All these terms are meaningful in the given scenario, i.e. the price is low (or good, or cheap) for the given camera, not in absolute terms.

Scenario I: Students and Exams

The first example scenario consists in students taking exams, and obtaining marks for the exams they take. Figure 6.7 shows how this scenario can be represented with a many-to-many relationship between the entities Student and Exam.

Now let us consider the following example from the above scenario: a *good* student takes a *difficult* exam.

While the notion of "difficult exam" can be highly subjective, the idea of a good student is usually associated with her good grades, e.g. how many A's she obtains. On the other hand, one can think about a "good student" as a student with a positive attitude towards his/her studies and the way he/she approaches lecturers and fellow students, e.g. being diligent and engaged.

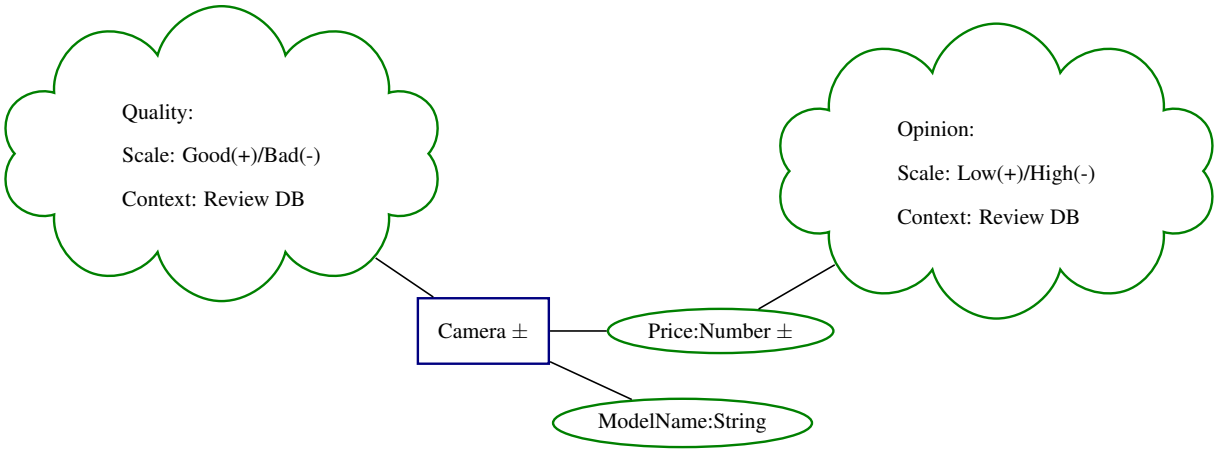


Figure 6.6: Second example of an Opinion-extended ER Diagram (Level-1 model). Meta-information related to the opinions are attached to the entity `Camera` and to the attribute `Price`.

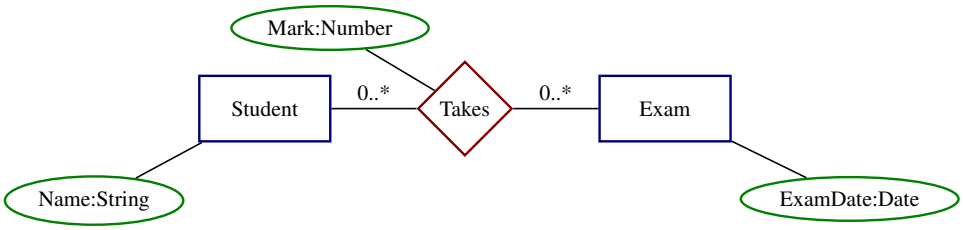


Figure 6.7: Traditional ER representation of students taking exams and getting marks for their exams.

Both the difficulty of an exam and the attitude of a student are subjective attributes, i.e. different persons can have different ideas about them. Moreover, both attributes can be represented on a scale, e.g. from poor to excellent attitude, and from easy to hard difficulty, and they can be translated into a sentiment scale as in Figure 6.2.

Figure 6.8 enhances the representation of students and exams shown in Figure 6.7, adding the polarity labels and the polarity tags.

Scenario II: Actors and Movies

The second example scenario consists in actors playing characters in movies. In particular, from the opinions point of view, it is interesting to represent the performance of the actor, his or her overall acting skills, and the attitude/morality of the character, as well as the overall opinion about the movie itself. Let us consider the following example: a *great* actor plays *excellently*

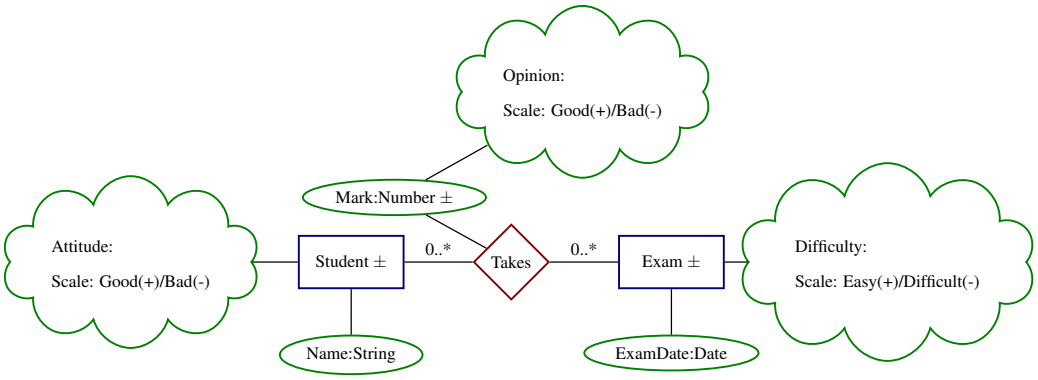


Figure 6.8: Opinion-extended representation of students taking exams. Opinions are expressed for `Student` (attitude), `Exam` (difficulty) and `Mark`.

the role of an *evil* character. Figure 6.9 shows the ER model for this scenario. The interesting part of this example is the co-occurrence of opposite feelings towards concepts which are part of the object a user may want to retrieve. In a retrieval system based on natural language, the query "good actor playing a bad guy" would probably be problematic when retrieving free-text documents (e.g. movie reviews) because of the co-occurrence of the terms good and bad (i.e. opposites on the polar scale) within the same query.

In this example, the ternary relationship is the glue which allows for representing scenarios like actors playing multiple characters in the same movie, or multiple actors playing the same character in the same movie, or the same character appearing in different movies, played by different actors. The three entities, as well as the ternary relationship, are all tagged to be considered with their respective overall opinions. In particular, having different opinions on entity `Actor` and on relationship `Plays` allows for representing good actors who perform badly in a particular movie. Moreover, a movie packed of good actors and performances, could still be perceived as a bad movie for other reasons.

The two example scenarios serve as showcases to introduce the importance of modelling concepts which are able to capture the semantics of opinions.

6.3 Mapping Opinion-enhanced Conceptual Models into Logical Models

This section describes the methodology for translating a conceptual model with opinion components into the logical model. On the conceptual layer, we considered:

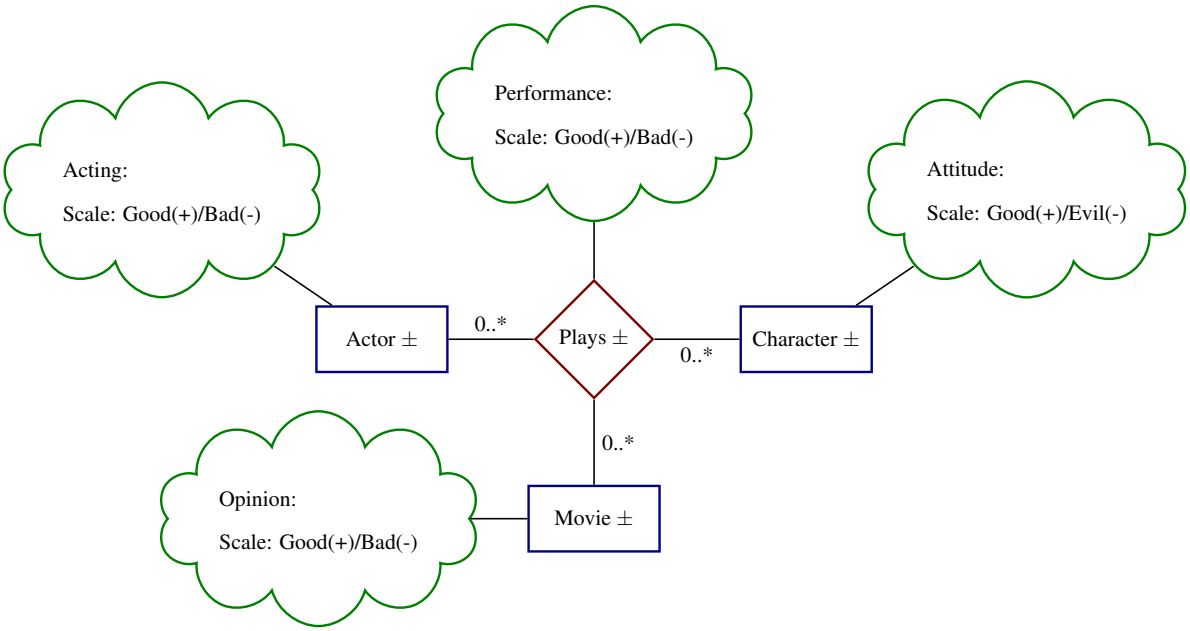


Figure 6.9: Opinion-extended representation of actors playing characters in movies. Opinions are expressed for `Movie`, `Actor` (acting skill), `Character` (attitude) and `Plays` (performance, i.e. how the actor performed in that particular play).

1. Opinionated entities (e.g. good student)
2. Opinionated relationship (e.g. peter is a good friend of mary)
3. Opinionated attributes (e.g. price of car is high, exam mark of student is good)

Whereas on the conceptual model, entities, attributes and relationships are labelled to be associated with opinions, on the logical layer, conventional attributes are employed to implement the conceptual model. This can be viewed as similar to what is known for multi-valued attributes. On the conceptual layer, there is a distinction between single-valued and multi-valued attributes, on the logical layer, all attributes are atomic, i.e. multi-valued attributes imply a decomposition to achieve the first normal form.

The mapping process is developed in four sections. Section 6.3.1 takes the view point of the *relational model* (schema-oriented). This is followed by Section 6.3.2 to cover aspects of *triplet-based and object-relational* modelling (schema-free). Then, Section 6.3.3 introduces a formalisation of the algorithm underlying the mapping process. Finally, Section 6.3.4 discusses a high-level SQL support for opinion-aware modelling, and this is considered in context with the

high-level SQL support regarding object-oriented modelling (available in most of today's DB systems).

6.3.1 Relational Model

Whereas on the conceptual layer we simply label attributes, entities and relationships to be associated with opinions, on the logical layer (relational model), there are several schema issues to face. It is important to note that on the conceptual layer, the designer does not specify an additional attribute for the entity for which opinion-based reasoning is required. The mapping to the relational model injects automatically an attribute for modelling the opinion. Thus, the mapping decides about the implementation of opinion-based reasoning, the designer can focus on the conceptual model. Opinion-aware modelling means to declare on the conceptual layer that opinions are to be considered. Opinions are often derived from other attributes. Example scenarios include:

A student is a good student if
he/she has good exam marks AND
he/she has good attitude

The exam mark of a student is good if
(the exam was difficult AND the mark $>$65) OR
(the exam was easy AND the mark $>$70)

The attitude of a student is good if
the attitude is hard working OR
the attitude is diligent

Opinionated components can be seen as a special type of multi-valued attributes. Multi-valued attributes are attributes which can contain more than one single value at the same time. Similarly, the polarity of an opinion can be at the same time positive or negative, depending on the different opinion holders.

During the translation process, normalisation maps multi-valued attributes using an additional relation. In a similar way, the proposed translation for opinions takes advantage of an additional

relation. For the student-exam example (Scenario I) in Figure 6.8, the translation to the logical schema is proposed as follows:

```
1  # Relations representing entities :
2  Student( StudentID, FirstName, LastName, ... )
3   Student_Attitude (StudentID, Context, Opinion)

5  Exam( ModuleID, Date, ... )
6   Exam_Difficulty (ModuleID, Date, Context, Opinion)

8  # Relations representing relationships between entities :
9  StudentTakesExam( StudentID, ModuleID, Date, Mark )
10 StudentTakesExam_Mark_Opinion( StudentID, ModuleID, Date, Context, Opinion);
```

The context attribute is what guarantees different people to have their own individual opinion on a single target. The context can say who the opinion holder is (e.g. if it is given as an object ID) or where the opinion was expressed (e.g. if it is given as a review URI). If the context is taken out of the picture, the opinions expressed have to be considered as overall/aggregated opinions. To illustrate, we show next a data instance of the student/exam scenario, with the context omitted for simplicity, i.e. the opinions are aggregated.

```
1  # Example instance ( context omitted for simplicity ):
2  Exam("CS101", "2013−06−30");
3   Exam_Difficulty ("CS101", "2013−06−30", "easy");
4  StudentTakesExam:
5          ( student1 , "CS101", "2013−06−30", 70)
6          ( student2 , "CS101", "2013−06−30", 60)
7   Student_Attitude :
8          ( student1 , "good")
9          ( student2 , "bad")

11 StudentTakesExam_Mark:
12         ( student1 , "CS101−2013", "2013−06−30", "good")
13         ( student2 , "CS101−2013", "2013−06−30", "bad")
```

In the next section we consider the mapping into a generic object-relational schema.

6.3.2 Triplet (Object-Relational) Model

A different approach for mapping the conceptual schema into a logical schema takes advantage of a triplet-based approach combined with an object-relational model (see Section 2.5.2). With this technique, a generic schema drives the representation of knowledge in which entity types become attribute values. For example, "class(student, student1)" is derived from "triplet(typeof, student1, student)", and "attribute(firstName, student1, "Peter")" is derived from "triplet(firstName, student1, "Peter")". The general schema is "triplet(Predicate, Subject, Object)". The object-relational triplet store is a mediator between a purely triplet-based representation, and the traditional representation where entity types (e.g. "student(Name,...)") are represented in the schema. To illustrate the opinion-enhanced generic object-relational schema, the same data instance of the student attitude, as pictured in the previous sections, is modelled as follows.

```
1  class ( student ,  student1 )
2   attribute  (name, student1 ,  "John  Jenkins")
3  class ( student ,  student2 )
4   attribute  (name, student2 ,  "Peter  Pan")

6   relationship  ( takes ,  student1 ,  exam1)
7   relationship  ( takes ,  student2 ,  exam1)

9    relationship_attribute   (mark, student1 ,  takes ,  exam1, 70)
10   relationship_attribute   (mark, student2 ,  takes ,  exam1, 69)
```

In accordance to Algorithm 2 and the triplet-based model, the generic schema for expressing opinions is then as follows:

```
1  # Opinion-oriented model for  entities :
2   class_Opinion ( excellent ,  student ,  student1 )
3   class_Opinion (good, student ,  student2 )

5  # The above  relation  can be  rewritten  according  to
6  # the  naming conventions  declared  by the  polarity  tag
7  # in  Figure  9 ( suffix  _Attitude  instead  of  _Opinion)
8   class_Attitude  ( excellent ,  student ,  student1 )
9   class_Attitude  (good, student ,  student2 )
```

```
11 # Opinion−oriented model for relationship attributes :
12   relationship_attribute_Opinion (good, mark, student1 , takes , exam1)
13   relationship_attribute_Opinion (good, mark, student2 , takes , exam1)
```

This example elicits that for opinionated entities and attributes, the mapping process generates relations that carry as the first attribute the opinion associated with the classification or attribute. The example also highlights for the polarity tag "attitude" of student how the default naming conventions can be overridden according to the information declared in the polarity tags. Overall, the logical model rests on the approach to add a special attribute to relations such that the first attribute models the opinion. Thus, conceptual layers can be automatically mapped into expressions that conform to the logical schema.

This extension is compatible and complementary to earlier work in [Fuhr and Roelleke, 1998, Lalmas and Roelleke, 2003] where for the modelling of incompleteness and inconsistencies the generic, object-relational schema was extended to model truth values (true, false, incomplete, inconsistent).

The two extremes of modelling have been discussed: the relational world in which semantic concepts are reflected in the schema, and the triplet world, in which generic relations are applied and the semantic concepts are ordinary attribute values. Next, the mapping process is formalised for the case of the relational model.

6.3.3 Mapping Algorithm

This section summarises the steps needed to derive relations from an opinion-aware conceptual schema. This procedure enhances the traditional methodology to derive a relational schema from traditional ER/E-ER models.

The formal procedure to derive the additional relations needed for representing opinions is shown in Algorithm 2.

Let E be the set of all entity types (e.g. Movie, Actor, etc.), and R be the set of all relationship types (e.g. plays, takes, etc.) in the conceptual schema:

$$E = \{E_1, ..., E_n\}$$

$$R = \{R_1, ..., R_m\}$$

Each entity type E_i is associated with the set of its attributes A_i. Similarly, each relationship types R_j is associated with the set of its attributes A_j. Let us define $E_O \subseteq E$ as the set of all opinionated entity types, i.e. the entity types for which we require opinion-based reasoning. Similarly, let us define $R_O \subseteq R$ as the set of all opinionated relationship types. The description of the steps necessary to create the relations, including examples, is detailed in the following subsections. In particular, the cardinality of the relationships will influence the choice of which step to consider. For each opinionated component, two relations have to be created: one for the individual opinions and one for the aggregate opinions. For L0 diagrams, the names of the relations will be in the format `ObjectName_Opinion` and `ObjectName_Opinion_aggregate`. The placeholder `ObjectName` will be replaced to reflect the specific object the opinion is about, e.g. `Actor_Opinion` for an opinion on the entity `Actor`, or `StudentTakesExam_Mark_Opinion` for an opinion about the mark (attribute over relationship). For L1 diagrams, if a specific name is given for the opinion, it will be injected to substitute the term "Opinion". For example, the opinion on the camera in Figure 6.6 will generate the relations `Camera_Quality` and `Camera_Quality_aggregate`.

Step 1 - Opinions about entities

For each opinion expressed directly on entities (i.e. $\pm$ symbol on entity), two relations have to be created. The relation for individual opinions contains a primary key and an extra attribute containing the opinion. The primary key is composed by the primary key from the relation representing the entity and the context. The relation for aggregated opinions is similar, but without the context attribute. For example, the opinion on the entity Student as in Figure 6.8 will generate:

1 Student_Attitude (StudentID, Context, Opinion)
2 Student_Attitude_aggregate (StudentID, Opinion)

Step 2 - Opinions about binary 1: and 1:1 relationships*

For each opinion expressed directly on binary relationships (i.e. $\pm$ symbol on relationship), two relations have to be created. For one-to-many and one-to-one relationships, during traditional mapping the two entities involved are identified as parent and child, with a copy of the parent's primary key posted into the child's relation, acting as foreign key. The relation for individual opinions contains the primary key, composed by the primary key from the child entity (e.g. the "many" side in 1:*) plus the context, the posted copy of primary key of the parent's entity, acting as foreign key, and an extra attribute for the opinion. The relation for aggregate opinions

is similar, but without the context attribute. For example, a *Staff manages Department* relationship, with a one-to-many cardinality and an opinion on the relationship itself, would yield the following relations:

```
1 Staff (StaffID, ...);
2 Department(DepartmentID, ..., managedBy)
3         managedBy references Staff ( StaffID ) ;
4 Department_managedBy_Opinion(DepartmentID, Context, Opinion)
5         DepartmentID references Department(DepartmentID);
6 Department_managedBy_Opinion_aggregate(DepartmentID, Opinion)
7         DepartmentID references Department(DepartmentID);
```

The naming rules applied here are the ones for modelling an opinion about the attribute of an entity (managedBy in this case), because of the foreign key migration due to the 1:* cardinality of the relationship.

*Step 3 - Opinions about binary *:* relationships*

For each opinion expressed directly on binary many-to-many relationships (i.e. $\pm$ symbol on relationship), two relations have to be created. During the traditional mapping, a bridge relation, holding the primary keys from both entities, is created. The opinion-aware relation for individual opinions contains a copy of the primary key attributes from the bridge relation, the context as part of the primary key, and an extra attribute for the opinion. The relation for aggregate opinions is similar, but without the context attribute. For instance, if we consider a *Staff serves Customer* many-to-many relationship, with an opinion on the relationship itself, we would obtain the following relations:

```
1 Staff (StaffID, ...);
2 Customer(CustomerID, ...);
3 StaffServesCustomer (StaffID, CustomerID)
4         StaffID references Staff ( StaffID )
5         CustomerID references Customer(CustomerID);
6 StaffServesCustomer_Opinion (StaffID, CustomerID, Context, Opinion)
7         ( StaffID , CustomerID) references StaffServesCustomer( StaffID , CustomerID);
8 StaffServesCustomer_Opinion_aggregate (StaffID, CustomerID, Opinion)
9         ( StaffID , CustomerID) references StaffServesCustomer( StaffID , CustomerID);
```

Step 4 - Opinions about complex relationships

For each opinion expressed on relationships with degree n ($n > 2$), two relations have to be created. The traditional mapping of such relationships would yield a bridge relation, holding the

primary key attributes of all the entities involved, individually acting as foreign key towards the respective entity. The relation about the individual opinions contains a copy of all these attributes, the context as part of the primary key, and an extra attribute for the opinion. The relation about the aggregate opinions is similar, but without the context attribute. For example, the `Plays` relationship in Figure 6.9 generates the following:

```
1  Actor(ActorID, ...);
2  Character(CharacterID, ...);
3  Movie(MovieID, ...);
4  Plays(AID, CID, MID)
5          AID references Actor(ActorID)
6          CID references Character(CharacterID)
7          MID references Movie(MovieID);
8  Plays_Performance(AID, CID, MID, Context, Opinion)
9          (AID, CID, MID) references Plays(AID, CID, MID);
10 Plays_Performance_aggregate(AID, CID, MID, Opinion)
11         (AID, CID, MID) references Plays(AID, CID, MID);
```

Step 5 - Opinions about attribute values

For each opinion expressed on a specific attribute, i.e. $\pm$ symbol on attribute, two relations have to be created. The relation about individual opinions contains the primary key of the relation which the attribute belongs to, the context as part of the primary key, and an extra attribute for the opinion. The relation about aggregate opinions is similar, but without the context attribute. For example, the `Mark` of an exam as represented in Figure 6.8 would yield the following:

```
1  StudentTakesExam(StudentID, Module, Date, Mark)
2          StudentID references Student(StudentID)
3          (Module, Date) references Exam(Module,Date);
4  StudentTakesExam_Mark_Opinion(StudentID, Module, Date, Context, Opinion)
5          (StudentID, Module, Date) references StudentExamMark(StudentID, Module, Date);
6  StudentTakesExam_Mark_Opinion_aggregate(StudentID, Module, Date, Opinion)
7          (StudentID, Module, Date) references StudentExamMark(StudentID, Module, Date);
```

The 5 steps presented provide a coherent way to produce the additional relations required for opinion-aware reasoning.

Algorithm 2 Mapping algorithm.

Input: E {Set of all entity types}

Input: R {Set of all relationship types}

Input: E_O {Set of all opinionated entity types}

Input: R_O {Set of all opinionated relationship types}

1: **for all** $E_i \in E$ **do**
2: **if** $E_i \in E_O$ **then**
3: create relations for global opinion and contextual opinion for entity set E_i {Step 1}
4: **end if**
5: **for all** $a_k \in A_i$ **do**
6: **if** a_k is opinionated **then**
7: create relations for global opinion and contextual opinion for attribute a_k {Step 5}
8: **end if**
9: **end for**
10: **end for**
11: **for all** $R_j \in R$ **do**
12: **if** $R_j \in R_O$ **then**
13: create relations for global opinion and contextual opinion for reship set R_j {Steps 2-4}
14: **end if**
15: **for all** $a_k \in A_j$ **do**
16: **if** a_k is opinionated **then**
17: create relations for global opinion and contextual opinion for attribute a_k {Step 5}
18: **end if**
19: **end for**
20: **end for**
21: **return** void

6.3.4 Opinion-enhanced SQL

Throughout this chapter, the separation of conceptual and logical layers has been emphasised. The examples have shown placeholders for the opinions, such as “good” and “bad”, but on the conceptual layer there is no explicit suggestion on how to implement opinions, i.e. what the value domain for attributes labelled as opinions is. Section 2.3 has introduced the sentiment scale, which can be used as a data type for representing opinions. The sentiment scale represents the polarity, which puts two opposite values in contrast. From this point of view, the same scale can be applied not only for good-vs-bad scenarios, but for any sort of value-based duality, like e.g. tall-vs-short, cheap-vs-expensive, etc. Considering also the fact that, in user-generated content, opinions are commonly expressed in natural language, attaching opinion-related keywords to the scale also provides a semantic definition for the two extremes, for example -1 indicates “cheap”, while +1 indicates “expensive” (0 would be “average”, or “fair”). Fuzzy attributes could be considered to implement sentiment scales, but this consideration is outside the scope of this thesis. Also, one could consider the use of controlled vocabularies or thesauri to augment this representation, clustering terms like “excellent”, “great”, or “fine” within the same extreme of the scale.

In order to illustrate the case for opinion-oriented concepts the case for object-oriented concepts, that led to the enhanced ER model, is briefly revisited.

A superclass/subclass relationship as in Figure 6.3(a) can be implemented with different relations and foreign keys, although an object-oriented extension of SQL would allow for a syntax as follows:

```
1  -- Superclass (base relation for Camera)
2  CREATE TABLE camera(
3          CameraID INT PRIMARY KEY,
4          Price NUMBER(6,2),
5          ModelName VARCHAR(200)
6  );

8  -- Disjoint subclasses
9  CREATE TABLE good_camera(
10         -- good_camera definition
11 ) INHERITS (camera);
```

```
13 CREATE TABLE bad_camera(
14         -- bad_camera definition
15 ) INHERITS (camera);
```

Alternatively, if the opinions are implemented using traditional attributes, as in Figure 6.3(b), the conceptual design would lead to an implementation such as:

```
1  -- OpinionScale data type as in Figure 1
2  CREATE DOMAIN OpinionScale AS REAL
3          CHECK (VALUE>=-1 AND VALUE <=+1);


5  -- Base relation for Camera
6  CREATE TABLE Camera(
7          CameraID INT PRIMARY KEY,
8          ModelName VARCHAR(200),
9          Price NUMBER(6,2),
10         CameraQuality OpinionScale,
11         PriceQuality OpinionScale
12 );
```

Following the principle of separating *what* from *how*, the data definition language can be extended in order to integrate opinion-based concepts in SQL. Specifically, a relation or an attribute can be declared `WITH OPINION`, meaning that opinion-based reasoning is required:

```
1  -- Base relation for Camera, with opinion enhancement
2  CREATE TABLE Camera WITH OPINION(good, bad) (
3          CameraID INT PRIMARY KEY,
4          ModelName VARCHAR(200),
5          Price NUMBER(6,2) WITH OPINION(low, high)
6  );
```

Through the mapping, the `WITH OPINION` declaration can automatically create the opinion-oriented relations needed for opinion-based reasoning:

```
1  -- Uniform Resource Identifier , used for Context
2  CREATE DOMAIN URI AS VARCHAR(500);


4  -- Opinion on Camera
5  CREATE TABLE Camera_Opinion(
6          CameraID INT,
```

```
7           Context URI,
8           Opinion OpinionScale,
9           PRIMARY KEY(CameraID, Context),
10          FOREIGN KEY CameraID REFERENCES Camera(CameraID)
11  );

13  -- Opinion on Camera(Price)
14  CREATE TABLE Camera_Price_Opinion(
15          CameraID INT,
16          Context URI,
17          Opinion OpinionScale,
18          PRIMARY KEY(CameraID, Context),
19          FOREIGN KEY CameraID REFERENCES Camera(CameraID)
20  );
```

Having considered an opinion-enhanced layer of SQL to underline the case of how opinion should be modelled, we are considering next a real-world scenario where the application of opinion-aware modelling is useful.

6.4 Example Application: Movie Database

This section showcases the use of opinion-enhanced modelling in a real-world application. In particular, the development of a movie database is discussed, with inspiration from e.g. The Internet Movie Database, a popular entertainment web site, which allows users to browse information about movies, TV series and video-games. Available data include details about actors, fictional characters and users' reviews.

The proposed representation is shown in Figure 6.10. A superclass MoviePerson represents all the people who work on a movie, including cast members, producers, editors, etc. In this example, for simplicity only actors, directors and producers are included. These three subclasses are overlapping, e.g. actors can also be directors, even in the same movie. A different entity is Character, i.e. the fictional character who is interpreted by an actor. The same character can appear in several movies and can be played by different actors, even within the same movie (e.g. young/old characters). An actor can play different characters, not only during his/her career, but also within the same movie (e.g. Eddie Murphy in The Nutty Professor). In order to represent

the connection between actor, character and movie, the ternary relationship `Plays` is used. For director and producer, there are individual binary relationships with the entity movie. Movie is clearly the central entity in this representation, and it is also the entity linked to users' reviews. Opinions can be expressed on movies, movie persons, characters' attitude (e.g. personality) and the relationship `Plays` (i.e. the performance).

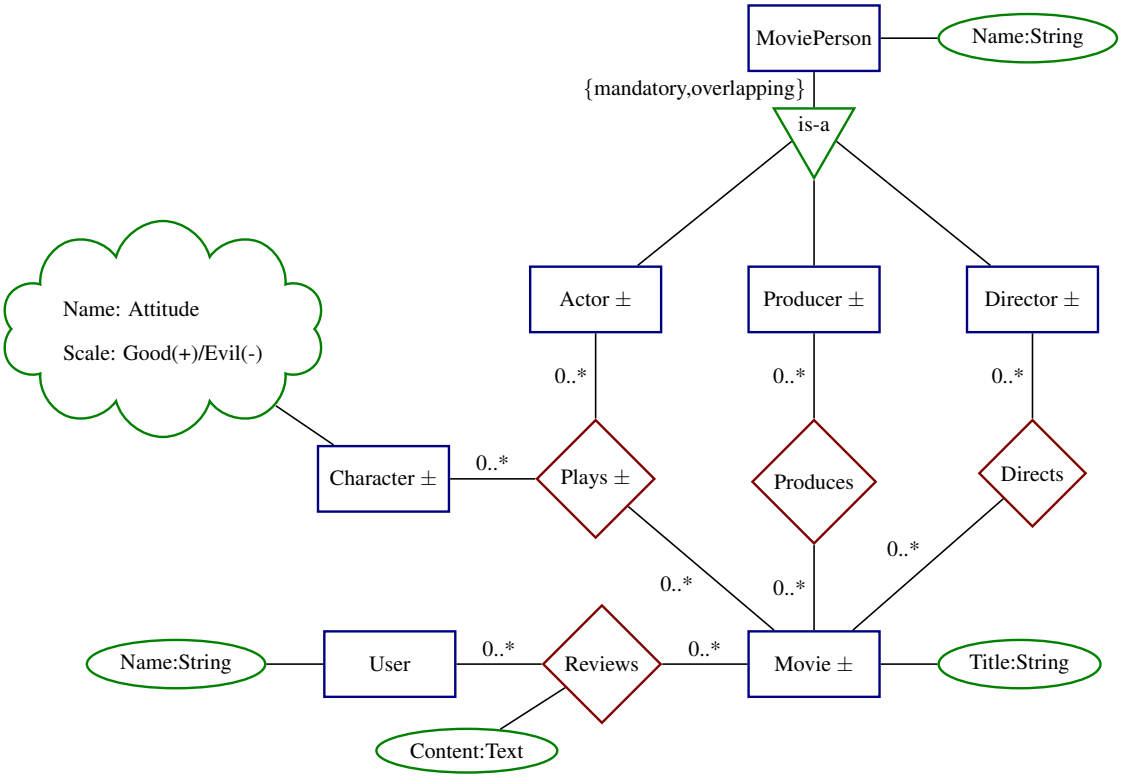


Figure 6.10: Opinion-aware data model of a movie database application (e.g. IMDb).

From the conceptual model in Figure 6.10, the following relations are derived, applying only the traditional modelling techniques first.

```
1  MoviePerson(PersonID, Name, ...);
2  Character (CharacterID, Name, ...);
3  Movie(MovieID, Title, ...);
4  User(UserID, Name, ...);
5  Reviews(UserID, MovieID, Content, ...)
6          UserID references User(UserID)
7          MovieID references Movie(MovieID);
8  Directs (DirectorID, MovieID)
9          DirectorID references MoviePerson(PersonID)
10         MovieID references Movie(MovieID);
```

```
Produces(ProducerID, MovieID)
        ProducerID references MoviePerson(PersonID)
        MovieID references Movie(MovieID);
Plays(MovieID, ActorID, CharacterID)
        MovieID references Movie(MovieID)
        ActorID references MoviePerson(PersonID)
        CharacterID references Character(CharacterID);
```

Once the basic relations are derived, following the procedure described in Algorithm 2 the additional opinion-oriented relations are generated as follows (relations for aggregate opinions are omitted for brevity):

```
# Opinions on entities
Actor_Opinion(ActorID, Context, Opinion)
        ActorID references MoviePerson(PersonID);
Director_Opinion(DirectorID, Context, Opinion)
        DirectorID references MoviePerson(PersonID);
Producer_Opinion(ProducerID, Context, Opinion)
        ProducerID references MoviePerson(PersonID);
Movie_Opinion(MovieID, Context, Opinion)
        MovieID references Movie(MovieID);
Character_Attitude(CharacterID, Context, Opinion)
        CharacterID references Character(CharacterID);
# Opinions on relationships
Plays_Opinion(MovieID, ActorID, CharacterID, Context, Opinion)
        (MovieID, ActorID, CharacterID) references Plays(MovieID, ActorID, CharacterID);
```

The next section elaborates a deeper discussion on the design process, on the semantic modelling of opinions and on its benefits for applications which require opinion-aware reasoning.

6.5 Discussion

6.5.1 Design is Subjective

Database design is, after all, a subjective process. During the design of opinions, the lack of tools specifically conceived for this kind of applications can lead developers to include implementation details at the conceptual level. Traditional modelling techniques, with a layered design process, aim at providing guidelines for designer and advocate the separation of conceptual, logical and

physical layers during the design process. The proposed extension to ER diagrams has to be considered with the same spirit. It is in fact a straightforward approach to support a better abstraction at the conceptual design stage and to delegate implementation details to the successive stages.

The representation of opinions is supported providing different levels of semantics. Firstly, the extension offers the graphical tool to state that opinions have to be considered on a particular entity, attribute or relationship (Level-0). Secondly, additional semantics about the opinion itself can be included in the model (Level-1). Since Level-1 information is not, strictly speaking, about the world being modelled (e.g. movies and actors), but rather about the world where the opinions are expressed (e.g. a review system), there is an important question regarding how much information has to be included, at this level, from the conceptual design point of view.

In particular, it is not immediately clear whether the domain of opinions should be included at this level or later. For example, a designer could use the sentiment scale to model polarised opinions, or could use a rating scale (e.g. 1-5 star system). This calls for a debate on the use of an opinion ontology, i.e. a more structured view of the opinions. When querying for a good product, a user is probably also interested to retrieve the product when the judgement is *very* good. On the other side, it is not directly obvious how to treat negation. From a binary logic point of view, if *good* and *bad* are two extremes, the expression *not bad* means *good*. On the opinion scale, if *bad* is the extreme -1, *not bad* means everything which is not -1 (e.g. average, good, very good). In natural language, the phrase *not bad* can be used to indicate both mild and strong satisfaction, so the aforementioned user could be less interested in retrieving products which are just passable.

Already for traditional attributes, the uncertainty and inconsistencies of attribute values are an important issue to be sorted in data integration tasks. In the opinion framework, this is particularly true. Opinions are often contrasting. Different persons can develop opposite feelings from the same facts or events. For example, while it is possible to observe that the totality of moviegoers like (or dislike) a particular movie, it is more common to have some discrepancy in the judgement (e.g. 95% of the audience liked the movie, while 5% was disappointed). It is also possible to witness the absence of sentiments towards a movie or a specific aspect of the movie (e.g. nobody has discussed about a supporting actor yet).

These examples, as well as the previous discussion on the overlap of positive and negative sentiments about the same entity, suggest that the dualism of positive-versus-negative opinions cannot be consistently treated from a logic point of view. This underlines that the opinion-aware mod-

elling is highly subjective and requires to provide the designers with concepts to express semantic details about opinions.

6.5.2 On the Benefit of Semantic Modelling of Opinions

Why would one want to enhance the enhanced ER model with just more concepts? It is interesting that such question will be asked, though from an abstraction point of view the high-level modelling of semantics is desirable. In order to provide a satisfying answer to this question, a discussion, which engages with the mid-conceptual and logical layers below the abstract opinion-aware entity-relationship layer, is needed. The case is detailed by drawing the analogy regarding inheritance (as has been developed on a more technical level in Section 6.3.4).

Although it seems obvious how to reflect inheritance in a relational model, it is considered useful to provide on the conceptual layer the expressiveness to specify class hierarchies. This is although on the logical layer, most designers may propose anyway a model (known as the multiple-relation model) where the subclass table has a foreign key pointing to the superclass table. For example, for persons, students and employees, the conceptual and logical models are:

```
1  # Conceptual approach to specify class hierarchy:
2  person(Id, Name, DateOfBirth, Age);
3  student(Id, Qualification) ISA person;
4  employee(Id, JobTitle, Salary) ISA person;
5  employedStudent() ISA student, employee;

7  # Logical approach (usage of foreign keys to model the class hierarchy):
8  person(Id, Name, DateOfBirth, Age);
9  student(Id, PersonId{FK}, Qualification);
10 employee(Id, PersonId{FK}, JobTitle, Salary);
11 employedStudent(StudentId{FK}, EmployeeId{FK});
```

Even though the model is obvious for a designer, there is no doubt that it makes sense to specify on the conceptual (semantic) layer the subclass-superclass relationship. The elegance of the conceptual model becomes even more evident when considering data manipulation and query statements.

```
1  # Object-oriented SQL layer:
2  INSERT INTO employedStudent (StudentId=123, EmployeeId=707, Name='Peter');
```

```
4  SELECT Name
5  FROM employedStudent
6  WHERE Age < 30
7  AND Qualification = 'MSc'
8  AND Salary > 20000;
```

Given the semantic information about *classes*, a mapping translates the conceptual SQL layer to the logical SQL layer.

```
1  # Relational SQL layer:
2  INSERT INTO person (Name='Peter'); # PersonId to be generated
3  INSERT INTO student (StudentId=123);
4  INSERT INTO employee (EmployeeId=707);
5  INSERT INTO employedStudent (StudentId=123, EmployeeId=707);

7  SELECT person.Name
8  FROM person, student, employee, employedStudent
9  WHERE person.Age < 30
10 AND student. Qualification = 'MSc'
11 AND employee.Salary > 20000
12 AND student.PersonId = person.Id
13 AND employee.PersonId = person.Id
14 AND employedStudent.StudentId = student.Id
15 AND employedStudent.EmployeeId = employee.Id;
```

The logical layer is much more complicated (loaded with technical details) whereas the conceptual SQL layer supports a concise, self-describing formulation with a focus on the semantic concepts. The case is similar but different for opinion-aware modelling.

It is similar since a conceptual layer is more elegant and concise, and the mapping guides the design on the logical layer. It is different, since for opinion-oriented modelling, there are many more design options regarding the logical schema modelling of opinions than for the object-oriented logical schema. This makes an opinion-oriented conceptual model even more desirable than an object-oriented model (since for the latter, the logical schema is somewhat straightforward anyway, but for opinion-oriented modelling, there are manifold design options).

In analogy to the story line of inheritance, the next examples underline the benefit of a conceptual opinion-aware layer.

```
# Conceptual approach to specify opinions:
person(Id, Name, DateOfBirth, Age(CurrentYear−YearOfBirth));
student[good,bad](Id, Qualification[good,bad]) ISA person;
employee(Id, JobTitle, Salary[high,low]) ISA person;
employedStudent() ISA student, employee;
examResult (..., Mark[good,poor])
exam[ difficult ,easy ](...)

# Logical approach (extra relations and attributes to specify opinions):
# ... same as above for the inheritance scenario
# followed by the opinion−specific part
# mapping option: no−context, i.e. opinions are global
student_Opinion(Id, Opinion);
 student_Qualification_Opinion (Id, Qualification , Opinion);
employee_Salary_Opinion(Id, Salary, Opinion);
 ...
# On this layer, opinion attributes can be stored or derived.
# Taking advantage from traditional concepts.
```

The following statements illustrate the effect of specifying the semantics.

```
# Opinion−oriented SQL layer:
INSERT INTO good:student (StudentId=123, Name='Peter');
INSERT INTO examResult (StudentId=123, ExamId=740, good:Mark=70);
INSERT INTO difficult:exam (ExamId=740, Title='Databases');

SELECT student.Name
FROM good:student, examResult, difficult :exam
WHERE opinion(examResult.Mark) = 'good'
AND examResult.ExamId = exam.Id;
```

Given the semantic information about *opinions*, a mapping translates the conceptual SQL to:

```
# Relational SQL layer:
INSERT INTO student (StudentId=123, Name='Peter');
INSERT INTO student_Opinion (StudentId=123, Opinion='good');
INSERT INTO examResult (StudentId=123, ExamId=740, Mark=70);
INSERT INTO examResult_Mark_Opinion (StudentId=123, ExamId=740, Opinion='good');
INSERT INTO exam (ExamId=740, Title='Databases');
INSERT INTO exam_Opinion (ExamId=740, Opinion='difficult');
```

```
9  SELECT person.Name
10 FROM person, student, student_Opinion ,
11         examResult, examResult_Mark_Opinion,
12         exam, exam_Opinion
13 WHERE student_Opinion.Opinion = 'good'
14 AND exam_Opinion.Opinion = ' difficult '
15 AND examResult_Mark_Opinion.Opinion = 'good'
16 AND examResult.ExamId = exam.Id
17 AND ... <various join conditions to relate tables >;
```

The example underlines that the the conceptual, opinion-aware SQL provides high-level concepts that lead to a more concise and self-describing formulation of SQL statements (similar to the case for object-oriented SQL).

6.5.3 Impact on Best Practices

The methodology proposed in this chapter encourages best practice for the development of applications where the representation of opinions is required. In general terms, best practice is viewed as practice that is successful, replicable, and measurable. The story of DB design and the layered design process involving a conceptual model, a logical model, and a physical model confirm best practice in data design.

When traditional database design methodologies are applied in the context of opinion-related requirements (sentiment analysis), one of the arising issues is that the "how-to" of the modelling of opinions is evident in the conceptual and the logical modelling steps. The methodology proposed in this chapter carries forward best practice for the modelling of opinions. It enforces the separation between conceptual and logical modelling, allowing the designer to focus at the conceptual stage on what-to-model rather than on how-to-model. Given an opinion-aware conceptual model with a well-defined mapping process, the practice is replicable and measurable. The same modelling concepts can be applied in different domains, and the conceptual model is not loaded with details about the how-to-implement opinion-based requirements. Also, the logical models of different applications show level of conformity that help to minimise the integration efforts in system integration tasks.

6.6 Summary

This chapter advocates the conceptual modelling of opinions. Rather than specifying in the traditional ER model details about *how* opinions are captured, this chapter shows a methodology to specify on the conceptual layer *that* opinions are to be considered. The proposed mapping process maps the conceptual model to the logical model, and this mapping process encapsulates the implementation details and leads to a consolidated (template-like) schema of the logical layer.

To make the case of an opinion-aware ER modelling approach, the facilities of enhanced ERM's have been re-considered. The additional concepts (special relationships and their mapping) regarding inheritance (class hierarchy) have been reminded. Firstly, these enhancements are reconsidered to investigate to what degree they can be utilised to model opinions. Secondly, the motivation is to align the opinion-oriented enhancement with existing enhancements. The additional concepts for modelling opinions are carefully laid out to co-exist with – and take advantage of – traditional concepts.

The main contributions of this chapter are (1) the conceptual modelling of opinions, and (2) the mapping of a conceptual, opinion-aware ER model to a logical (relational) model. Regarding the conceptual model, the enhancement is based on labelling the entities, attributes and relationship for which opinion-based reasoning is required. There are two levels regarding the semantic details provided in the conceptual model. On Level-0, the designer simply specifies "that" opinion-aware reasoning is required, using polarity labels. On Level-1, the designer can use so-called polarity tags (graphically, "thought-balloons"), where polarity tags are containers for opinion-oriented meta information used to enrich the model. Polarity tags can carry detailed information such as name and type (value domain) of opinions. The mapping process generates several database objects on the logical layer.

Given the opinion-aware ERM, the mapping process (see Algorithm 2) is aware of the entities, relationships and attributes that are labelled as "opinionated". For each opinion-labelled entity, relationship and attribute the mapping process generates relations. For example, for the entities student and camera, and opinionated attributes studentTakesExam.Mark and camera.Price, the relations student_Opinion and camera_Opinion (for the entities) and studentTakesExam_Mark_Opinion and camera_Price_Opinion are created. The main motivations and aspects of the mapping process are:

1. The mapping process creates various relations for opinionated entities, relationships and attributes. Defaults apply unless the designer specifies more semantic details (polarity tags) on the conceptual layer (Level-1).

2. The generated schema includes relations for contextual and global opinions. Opinions can be implemented as derived attributes, and this is in particular evident for global opinions derived from context-based opinions (e.g. opinions from several reviews aggregated into one global opinion).

3. The name of the opinion-oriented relation is by default constructed from the entity (relationship and attribute) names and the suffix "_Opinion". On specification of a polarity tag in the conceptual model, the name of the tag constitutes the suffix, and several other semantic details that affect the logical layer can be specified on the conceptual layer.

4. Given the consolidated logical model, opinion-aware layers of SQL can provide concise and elegant syntax components to express opinion-oriented queries.

The conceptual specification of requirements regarding opinion-based reasoning and the implementation (schema) is guided by a pre-defined and potentially generalisable design pattern.

This research impacts on the many development tasks that require to capture opinions. Opinion-related requirements are in particular evident in today's opinion-oriented (review-oriented) web services such as restaurant reviews, hotel reviews, business reviews, product reviews, and movie reviews. For providing an application-oriented case, the opinion-aware modelling of a movie database system has been considered.

Overall, this chapter provides the groundwork for enhanced ER modelling that is capable to capture opinion-related requirements. The aim was to achieve a conceptual model with a balanced enhancement that on one hand is "minimal" and on the other hand provides expressiveness to specify semantic details about opinions. Regarding the minimal (Level-0) model, the framework supports the specification of requirements such as "we want to model contextual (review-based) opinions about movies" and "we want to model global opinions about actors". Famous and good actors might perform poorly in a particular movie, and modelling such facets of opinion-based knowledge led to the framework presented in this chapter.

The proposed opinion-aware knowledge representation supports a guided data and software de-

sign process. It complements the other methodologies and standards applied for knowledge representation. Given the opinion-oriented concepts (polarity labels and polarity tags) and the mapping process (conceptual to logical model), the opinion-aware ER model forms a foundation for the many verticals and the many developers that face requirements as they occur in opinion-oriented applications.

Chapter 7

Conclusions

This thesis has explored several aspects of the comprehensive task of opinion-aware knowledge-oriented summarisation. The two main lines of work within this thesis have been the use of statistical models for extractive summarisation and knowledge representation, in the context of sentiment analysis applications. Chapter 3 has opened the discussion on statistical models, offering an investigation on divergence-based methods which have been applied to summarisation. Chapter 4 has then brought opinions into play, discussing opinion-oriented approaches to tackle summarisation-related tasks. Chapter 5 has functioned as the link between the two pillars, examining the use of knowledge-based technology applied to summarisation. Chapter 6 has finally focused on the knowledge representation of opinions.

7.1 Contributions

The contributions of this thesis have been summarised in Chapter 1, where the main research questions have also been proposed. This section aggregates the contributions of the individual chapters by answering those research questions.

How do geometric and information-theoretic methods (e.g. cosine and divergence) compare in the context of sentence selection, in particular w.r.t. summarisation quality?

Different tasks related to summarisation, sentence-to-document and summary-to-document similarity have been outlined and investigated.

KL-divergence and cosine similarity have been applied to calculate the similarity between a sentence and a document, or between a sentence and a candidate summary. Experimental results do not show a clear winner in the context of intrinsic sentiment summarisation, where the sentence removal algorithm has been compared to different baselines. All the approaches could utilise either divergence or cosine (or potentially other methods) to calculate sentence similarity. While the sentence removal algorithm has consistently performed better on some metrics, there was no particular indication that either divergence or cosine could consistently offer better performances.

Given an iterative approach to produce summaries by removing the less important sentences, how does it perform in terms of summarisation quality, compared to approaches which select the most important sentences?

The sentence removal algorithm has been compared to different baselines, namely MEAD, greedy selection and brute-force selection. Overall, the sentence removal algorithm tends to maintain the maximum coverage of the topic while iterating. As a result, it has shown the best overall results for F_2 different ROUGE metrics, being significantly superior than the second-best result.

The use of the sentence removal algorithm has been envisioned as a second component of a two-stage system, where the first component retrieves relevant, on-topic sentences and the summarisation component condenses such sentences into a short summary. The main reason for using the sentence removal algorithm as a second-stage component is that it executes in quadratic time (same as the brute-force approach) while the greedy approach is linear. In this way, the starting point is a relatively small set of sentences so the quadratic complexity does not critically hurt the running time.

Given an approach to recognise opinion-bearing terms, how does term boosting affect the quality of sentence selection for sentiment summarisation, and how can terms be treated in order to improve quality on sentiment summarisation?

There are different approaches to treat terms which are recognised as opinion-bearing. This thesis has analysed some possibilities for pre-processing of opinion-bearing terms, in particular treating opinion as stop-words, frequency boosting, including opinion-based bigrams and negation-based removal.

The only approach which has consistently shown performance improvements across different

metrics is the inclusion of opinion-based bigrams. This approach consist in merging two term into a single bigram, being the first term an opinion-bearing one. In this way, phrases like *good food* can be captured.

Boosting frequencies, i.e. repeating opinion-bearing terms multiple times in place, has shown discordant results for lower boosts, while in general performances drop dramatically for higher boosts (e.g. ≥ 7).

How is it possible to summarise a document while preserving its overall polarity information?

The idea of identifying a short passage, or even a single sentence, which captures the overall polarity of a given review has been backed by some empirical evidence: many reviewers express different opinions about different aspects of the topic they are discussing, and they often close the review with a short passage where the overall polarity is stated.

Statistical and positional summarisation methods are not able to summarise a review while presenting its overall polarity.

On the other side, subjectivity detection allows to filter out sentences which do not express opinions, hence to eliminate noise. Experimental results have shown that subjectivity detection is able to preserve the same polarity expressed in the full-text review, while shortening the review itself. This is beneficial for a user who does not need to read the full-text in order to understand the polarity expressed in the document.

What kind of technologies and methodologies are needed in order to enable knowledge-based summarisation?

This thesis have introduced knowledge-based summarisation as a different approach than sentence selection. The parallel is with knowledge-oriented (i.e. semantic) retrieval, where a knowledge base is exploited for the task. In order to enable knowledge-based summarisation, a knowledge representation suitable for the task is needed. This thesis has discussed and exploited a generic data model which is application agnostic and allows to build more semantic layers on top of itself. A movie scenario has been discussed as a use case, showing how the basic relations can be semantically "lifted" in order to produce a data model tailored for the specific application. The use of the generic data model as well as a declarative language, such as Probabilistic Datalog, allows to quickly and easily produce complex semantic components of the data model.

What kind of expressivity and flexibility do traditional conceptual modelling techniques provide in terms of modelling opinions? In other words, are traditional conceptual modelling concepts enough to model opinions?

A thorough discussion on how to represent opinions at the conceptual level have been undertaken. The main outcome is that traditional (ER or E-ER) modelling techniques introduce implementation details at the conceptual level. In other words, while it is possible to build systems which handle sentiment-oriented data, there is a lack of semantics in the data modelling, i.e. the essence of opinions is not well-represented. It is argued in this thesis that more semantic concepts are needed in order to represent opinions while supporting the expressivity and flexibility needed by sentiment analysis application. The key idea is that there is a need to separate the *what* (opinions need to be integrated) from the *how* (implementation details).

How is it possible to provide additional semantic concepts in order to model opinions at the conceptual level, and how to de-couple such modelling from the logical layer?

This thesis has tackled the problem of supporting opinion-aware knowledge representation by enhancing the traditional (ER and E-ER) conceptual modelling. The proposed opinion-aware conceptual modelling approach is based on two levels. At the first level, the graphical extension of ER models is given by *polarity labels*, which consists of a $\pm$ symbol placed next to entities, attributes or relationships for which the data designer require opinion-aware reasoning. At the second level, polarity tags in the form of though-balloons are attached to the components already marked with polarity labels. In this way, additional opinion-oriented information can be injected in the model without explicitly suggesting implementation details.

What else is needed in order to support sentiment analysis applications which model opinions at the conceptual layer? In other words, once a conceptual modelling of opinions is available, how to map it into the logical layer?

Traditionally, data designers have been provided with general guidelines to facilitate the translation of a conceptual model (e.g. ER diagram) into a logical model (e.g. relational schema). This thesis has proposed a procedure to automatically translate opinion-aware conceptual schemata into logical schemata. The procedure takes into account the opinion-oriented information included in the enriched ER diagram (i.e. polarity labels and polarity tags) to produce additional relations which support the modelling of individual and aggregated opinions. Individual opin-

ions are opinions expressed in a particular context (e.g. a given review), which can be linked to a specific opinion holder. Aggregated opinions are not connected to one context as they reflect the overall (aggregated) opinion. Both individual and aggregated opinions are useful in sentiment analysis applications. This thesis has also continued the discussion on the semantic modelling of opinions by suggesting potential uses of opinion-aware technology not only at the conceptual but also at the application level, e.g. opinion-enhanced SQL.

7.2 Limitations and Future Work

This section outlines potential extensions to the work discussed in this thesis. The main question is whether the outcome of this research can be brought into a production environment. In order to provide a comprehensive answer to this question, some aspects, which have not been discussed in this thesis, should be investigated in more details.

An important aspect to examine is the user's perspective. In terms of document summarisation, a key feature to consider is the purpose of the summary, more precisely how the end-user will utilise the summarisation system and benefit from it. The experimental studies carried out in Chapter 3 and Chapter 4 w.r.t. summarisation have employed an automatic evaluation methodology, i.e. the ROUGE framework, in order to provide numerical results for comparisons between systems. The benefit of using a framework like ROUGE, assuming gold standard summaries are available, is the ability to quickly compare a large number of summarisation systems in a relatively cheap way. Using ROUGE in the early stages of development translates into the ability of performing rapid prototyping which requires several iterations and refinements. Involving the final users too early in the process would be extremely expensive and would slow down the development. On the other side, ROUGE does not assess the final usefulness of a summary. Even without a formal definition of usefulness, it is important to note that this is strongly dependent on the context, i.e. the purpose of the summary mentioned earlier. For this reason, users should be involved at the end of the development process, either in a task-oriented evaluation or in the assessment of qualitative aspects of the summaries like readability or coherence.

The discussion on knowledge modelling, especially Chapter 6 on knowledge representation, can also benefit from taking the final user's perspective into account. In this case, users are data designers or engineers of opinion-oriented applications. The methodology advocated in Chap-

ter 6 enriches the data model with opinion-oriented concepts, following and encouraging the use of best practices, in particular the separation between conceptual, logical and physical layers during the design process. Data designers should anyway be involved in the evaluation of such methodology mainly for two reasons. Firstly, data designers can clarify the definition of requirements for the development of opinion-oriented applications. Secondly, data designers can assess whether the adoption of such methodology provides benefits such as speeding up the design process, improving the clarity of the data model, and overall supporting all the opinion-oriented requirements of the application.

Another aspect to consider is the scalability of the design approach proposed in Chapter 6. In particular, the methodology requires all the attributes to be known, in order to label them with polarity labels and polarity tags. In reality, this might not always be possible in the early design stages, because for some applications new attributes of some entities are brought up by new data. This is the case of on-line reviews where users discuss different, sometimes unseen by the system, aspects of products. More in general, the challenge is the combination of structured and unstructured data which is becoming more and more a necessity in sentiment analysis applications.

In terms of building a real product, real data also pose other challenges which have not been discussed in this thesis. For example, this work could be linked to a deeper discussion about how to handle authority, how to assess user's trustiness and how to handle spam. While spam detection is in general a well understood problem, the tasks of detecting fake reviews or identifying trustworthy users is still an open research area.

7.3 Research Outlook

This work has shown that integrating concepts of opinions in a coherent knowledge representation framework, in order to support a specific task such as summarisation, is challenging. There is a number of different aspects to consider when facing the comprehensive task of opinion-aware knowledge-based summarisation, such as knowledge representation, statistical modelling of summarisation and integration of opinions. This thesis has disassembled such aspects and tackled them individually. One main challenge in this work is the integration of all these aspects into one solid framework.

The contributions of this research enrich the field of knowledge representation with the purpose of supporting data designers or knowledge engineers in the development of applications which tackle complex information needs.

Bibliography

[Aizawa, 2003] Aizawa, A. (2003). An information-theoretic perspective of tf–idf measures. *Information Processing & Management*, 39(1):45–65.

[Andreevskaia and Bergler, 2006] Andreevskaia, A. and Bergler, S. (2006). Mining wordnet for a fuzzy sentiment: Sentiment tag extraction from wordnet glosses. In *EACL*, pages 209–216, Stroudsburg, PA, USA. The Association for Computational Linguistics.

[Azzam and Roelleke, 2011] Azzam, H. and Roelleke, T. (2011). A generic data model for schema-driven design in information retrieval applications. In *ICTIR, Bertinoro, Italy*, volume 6931, pages 164–175, Berlin, Germany. Springer.

[Azzam et al., 2012] Azzam, H., Yahyaei, S., Bonzanini, M., and Roelleke, T. (2012). A schema-driven approach for knowledge-oriented retrieval and query formulation. In *Proceedings of the Third International Workshop on Keyword Search on Structured Data*, KEYS '12, pages 39–46, New York, NY, USA. ACM.

[Baeza-Yates and Ribeiro-Neto, 1999] Baeza-Yates, R. A. and Ribeiro-Neto, B. (1999). *Modern Information Retrieval*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA.

[Beineke et al., 2004] Beineke, P., Hastie, T., Manning, C., and Vaithyanathan, S. (2004). Exploring sentiment summarization. In *AAAI Spring Symposium on Exploring Attitude and Affect in Text: Theories and Applications*.

[Bilotti et al., 2007] Bilotti, M. W., Ogilvie, P., Callan, J., and Nyberg, E. (2007). Structured retrieval for question answering. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 351–358, New York, NY, USA. ACM, ACM.

[Bizer et al., 2009] Bizer, C., Lehmann, J., Kobilarov, G., Auer, S., Becker, C., Cyganiak, R., and Hellmann, S. (2009). Dbpedia-a crystallization point for the web of data. *Web Semantics: Science, Services and Agents on the World Wide Web*, 7(3):154–165.

[Blair-Goldensohn et al., 2008] Blair-Goldensohn, S., Hannan, K., McDonald, R., Neylon, T., Reis, G., and Reynar, J. (2008). Building a sentiment summarizer for local service reviews. In *WWW Workshop on NLP in the Information Explosion Era*.

[Bonzanini et al., 2012] Bonzanini, M., Martinez-Alvarez, M., and Roelleke, T. (2012). Opinion summarisation through sentence extraction: an investigation with movie reviews. In *Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '12, pages 1121–1122, New York, NY, USA. ACM.

[Brachman and Levesque, 2004] Brachman, R. J. and Levesque, H. J. (2004). *Knowledge representation and reasoning*. Morgan Kaufmann.

[Caruana and Niculescu-Mizil, 2006] Caruana, R. and Niculescu-Mizil, A. (2006). An empirical comparison of supervised learning algorithms. In *Proceedings of the 23rd International Conference on Machine learning, ICML 2006*, pages 161–168. ACM.

[Chen and Kerre, 1998] Chen, G. and Kerre, E. (1998). Extending er/eer concepts towards fuzzy conceptual data modeling. In *Proceedings of the 1998 IEEE International Conference on Fuzzy Systems*, volume 2, pages 1320–1325, Piscataway, NJ, USA. IEEE, IEEE.

[Chen, 1976] Chen, P. P. (1976). The entity-relationship model - toward a unified view of data. *ACM Transactions on Database Systems (TODS)*, 1(1):9–36.

[Chenlo and Losada, 2011] Chenlo, J. and Losada, D. (2011). Effective and efficient polarity estimation in blogs based on sentence-level evidence. In *Proceedings of the 20th ACM international conference on information and knowledge management*, pages 365–374. ACM.

[Church and Gale, 1995] Church, K. and Gale, W. (1995). Inverse document frequency (idf): A measure of deviation from poisson. In *Proceedings of the Third Workshop on Very Large Corpora*, pages 121–130.

[Connolly and Begg, 2005] Connolly, T. M. and Begg, C. E. (2005). *Database systems: a practical approach to design, implementation, and management*. Addison-Wesley Longman, Boston, MA, USA.

[Conroy et al., 2006] Conroy, J., Schlesinger, J., and O'Leary, D. (2006). Topic-focused multi-document summarization using an approximate oracle score. In *Proceedings of the COLING/ACL*, pages 152–159. ACL.

[Conroy et al., 2004] Conroy, J. M., Schlesinger, J. D., Goldstein, J., and O'leary, D. P. (2004). Left-brain/right-brain multi-document summarization. In *Proceedings of the Document Understanding Conference (DUC 2004)*.

[Cormen et al., 2001] Cormen, T., Leiserson, C., Rivest, R. L., and Stein, C. (2001). *Introduction to Algorithms*, chapter 15.4, pages 350–355. MIT Press and McGraw-Hill, second edition.

[Croft, 1993] Croft, W. B. (1993). Knowledge-based and statistical approaches to text retrieval. *IEEE Expert*, 8(2):8–12.

[Cronen-Townsend et al., 2002] Cronen-Townsend, S., Zhou, Y., and Croft, W. (2002). Predicting query performance. In *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 299–306. ACM.

[Dadvar et al., 2011] Dadvar, M., Hauff, C., and de Jong, F. (2011). Scope of negation detection in sentiment analysis. In *Proceedings of the Dutch-Belgian Information Retrieval Workshop, (DIR 2011)*, pages 16–19. University of Amsterdam.

[Daumé III and Marcu, 2002] Daumé III, H. and Marcu, D. (2002). A noisy-channel model for document compression. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pages 449–456. Association for Computational Linguistics.

[De Marneffe et al., 2008] De Marneffe, M.-C., Rafferty, A. N., and Manning, C. D. (2008). Finding contradictions in text. In *ACL*, volume 8, pages 1039–1047.

[Di Fabbrizio et al., 2011] Di Fabbrizio, G., Aker, A., and Gaizauskas, R. (2011). Starlet: Multi-document summarization of service and product reviews with balanced rating distributions. In *IEEE International Conference on Data Mining (ICDM 2011) Workshop - Sentiment Elicitation from Natural Text for Information Retrieval and Extraction (SENTIRE 2011)*, Vancouver, Canada.

[Diaz and Gervas, 2007] Diaz, A. and Gervas, P. (2007). User-model based personalized summarization. *Information Processing & Management*, 43(6):1715–1734.

[Dunning, 1993] Dunning, T. (1993). Accurate methods for the statistics of surprise and coincidence. *Computational linguistics*, 19(1):61–74.

[Edmundson, 1969] Edmundson, H. (1969). New methods in automatic extracting. *Journal of the ACM (JACM)*, 16(2):264–285.

[Erkan and Radev, 2004] Erkan, G. and Radev, D. R. (2004). Lexrank: Graph-based lexical centrality as salience in text summarization. *J. Artif. Intell. Res.(JAIR)*, 22(1):457–479.

[Finley et al., 2004] Finley, L., Andrew, H., et al. (2004). Lite-gistexter at duc 2004. In *Proceedings of the 2004 Document Understanding Conference (DUC 2004), Boston, MA*.

[Fuhr, 1995] Fuhr, N. (1995). Probabilistic datalog - a logic for powerful retrieval methods. In *Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 282–290. ACM.

[Fuhr et al., 1998] Fuhr, N., Goevert, N., and Roelleke, T. (1998). Dolores: A system for logic-based retrieval of multimedia objects. In *SIGIR*, pages 257–265.

[Fuhr and Roelleke, 1998] Fuhr, N. and Roelleke, T. (1998). HySpirit — a probabilistic inference engine for hypermedia retrieval in large databases. In Schek, H.-J., Saltor, F., Ramos, I., and Alonso, G., editors, *Proceedings of the 6th International Conference on Extending Database Technology (EDBT)*, pages 24–38, New York, NY, USA. ACM.

[Ganesan et al., 2010] Ganesan, K., Zhai, C., and Han, J. (2010). Opinosis: a graph-based approach to abstractive summarization of highly redundant opinions. In *Proceedings of the 23rd International Conference on Computational Linguistics*, pages 340–348. Association for Computational Linguistics.

[Ganesan et al., 2012] Ganesan, K., Zhai, C., and Viegas, E. (2012). Micropinion generation: an unsupervised approach to generating ultra-concise summaries of opinions. In *Proceedings of the 21st international conference on World Wide Web (WWW 2012)*, pages 869–878. ACM.

[Harabagiu et al., 2006] Harabagiu, S., Hickl, A., and Lacatusu, F. (2006). Negation, contrast and contradiction in text processing. In *AAAI*, volume 6, pages 755–762.

[Hu and Liu, 2004a] Hu, M. and Liu, B. (2004a). Mining and summarizing customer reviews. In *Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 168–177. ACM.

[Hu and Liu, 2004b] Hu, M. and Liu, B. (2004b). Mining opinion features in customer reviews. In *Proceedings of the National Conference on Artificial Intelligence*, pages 755–760. AAAI.

[Jing, 2002] Jing, H. (2002). Using hidden markov modeling to decompose human-written summaries. *Computational Linguistics*, 28(4):527–543.

[Kastner and Monz, 2009] Kastner, I. and Monz, C. (2009). Automatic single-document key fact extraction from newswire articles. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*, pages 415–423. Association for Computational Linguistics.

[Kim and Zhai, 2009] Kim, H. D. and Zhai, C. (2009). Generating comparative summaries of contradictory opinions in text. In *Proceedings of the 18th ACM Conference on Information and Knowledge Management (CIKM 2009)*, pages 385–394. ACM.

[Kim et al., 2009] Kim, J., Xue, X., and Croft, W. B. (2009). A probabilistic retrieval model for semistructured data. In *Proceedings of the 31st European Conference on Information Retrieval, ECIR 2009*, pages 228–239. Springer.

[Kupiec et al., 1995] Kupiec, J., Pedersen, J., and Chen, F. (1995). A trainable document summarizer. In *Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 68–73. ACM.

[Lalmas and Roelleke, 2003] Lalmas, M. and Roelleke, T. (2003). Four-valued knowledge augmentation for structured document retrieval. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems (IJUFKS), Special issue on management of uncertainty and imprecision in multimedia information systems*, 11(1):67–86.

[Lerman and McDonald, 2009] Lerman, K. and McDonald, R. (2009). Contrastive summarization: an experiment with consumer reviews. In *Proceedings of human language technologies: The 2009 annual conference of the North American chapter of the association for computational linguistics, companion volume: Short papers*, pages 113–116. Association for Computational Linguistics.

[Leveling and Jones, 2012] Leveling, J. and Jones, G. J. (2012). Making results fit into 40 characters: a study in document rewriting. In *Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval*, pages 1107–1108. ACM.

[Lin, 2004a] Lin, C. (2004a). Looking for a few good metrics: Automatic summarization evaluation-how many samples are enough. In *Proceedings of the 4th NTCIR Workshop*, pages 1–10.

[Lin, 2004b] Lin, C. (2004b). Rouge: A package for automatic evaluation of summaries. In *Proceedings of the Workshop on Text Summarization Branches Out*, pages 74–81. Association for Computational Linguistics.

[Lin et al., 2012] Lin, C., He, Y., Everson, R., and Ruger, S. (2012). Weakly supervised joint sentiment-topic detection from text. *Knowledge and Data Engineering, IEEE Transactions on*, 24(6):1134–1145.

[Lin and Hovy, 2000] Lin, C.-Y. and Hovy, E. (2000). The automated acquisition of topic signatures for text summarization. In *Proceedings of the 18th conference on Computational linguistics-Volume 1*, pages 495–501. Association for Computational Linguistics.

[Liu, 2010] Liu, B. (2010). Sentiment analysis and subjectivity. *Handbook of Natural Language Processing*.

[Liu, 2012] Liu, B. (2012). Sentiment analysis and opinion mining. *Synthesis Lectures on Human Language Technologies*, 5(1):1–167.

[Louis and Nenkova, 2008] Louis, A. and Nenkova, A. (2008). Automatic summary evaluation without human models. In *Notebook Papers and Results, Text Analysis Conference (TAC-2008), Gaithersburg, Maryland (USA)*.

[Luhn, 1958] Luhn, H. (1958). The automatic creation of literature abstracts. *IBM Journal of research and development*, 2(2):159–165.

[Manning and Schütze, 1999] Manning, C. D. and Schütze, H. (1999). *Foundations of statistical natural language processing*. MIT press.

[Maybury and Mani, 2001] Maybury, M. and Mani, I. (2001). Tutorial notes on automatic summarization. Technical report, MITRE Corporation.

[Mei et al., 2007] Mei, Q., Ling, X., Wondra, M., Su, H., and Zhai, C. (2007). Topic sentiment mixture: modeling facets and opinions in weblogs. In *Proceedings of the 16th international conference on World Wide Web*, pages 171–180. ACM.

[Mizzaro, 1997] Mizzaro, S. (1997). Relevance: The whole history. *Journal of the American society for information science*, 48(9):810–832.

[Nenkova and McKeown, 2011] Nenkova, A. and McKeown, K. (2011). Automatic summarization. *Foundations and Trends in Information Retrieval*, 5(2-3):103–233.

[Nenkova and Passonneau, 2004] Nenkova, A. and Passonneau, R. (2004). Evaluating content selection in summarization: The pyramid method. In *HLT-NAACL*, pages 145–152.

[Nenkova et al., 2007] Nenkova, A., Passonneau, R., and McKeown, K. (2007). The pyramid method: Incorporating human content selection variation in summarization evaluation. *ACM Transactions on Speech and Language Processing (TSLP)*, 4(2):4.

[Pang and Lee, 2004] Pang, B. and Lee, L. (2004). A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, pages 271–278, Stroudsburg, PA, USA. The Association for Computational Linguistics, The Association for Computational Linguistics.

[Pang and Lee, 2005] Pang, B. and Lee, L. (2005). Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *Proceedings of the 43rd annual meeting of the Association for Computational Linguistics*, volume 43, pages 115–124.

[Pang and Lee, 2008] Pang, B. and Lee, L. (2008). Opinion mining and sentiment analysis. *Foundations and Trends in Information Retrieval*, 2(1-2):1–135.

[Pang et al., 2002] Pang, B., Lee, L., and Vaithyanathan, S. (2002). Thumbs up?: sentiment classification using machine learning techniques. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10*, pages 79–86. Association for Computational Linguistics.

[Paul et al., 2010] Paul, M. J., Zhai, C., and Girju, R. (2010). Summarizing contrastive viewpoints in opinionated text. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing (EMNLP 2010)*, pages 66–76. Association for Computational Linguistics.

[Porter, 1980] Porter, M. F. (1980). An algorithm for suffix stripping. *Program: electronic library and information systems*, 14(3):130–137.

[Potthast and Becker, 2010] Potthast, M. and Becker, S. (2010). Opinion Summarization of Web Comments. In *Proceedings of the 32nd European Conference on Information Retrieval, ECIR 2010*, pages 668–669.

[Potts, 2011] Potts, C. (2011). On the negativity of negation. In *Proceedings of SALT*, volume 20, pages 636–659.

[Pradhan et al., 2004] Pradhan, S., Ward, W., Hacioglu, K., Martin, J., and Jurafsky, D. (2004). Shallow semantic parsing using support vector machines. In *Proceedings of HLT/NAACL*, pages 233–240.

[Radev et al., 2004] Radev, D., Allison, T., Blair-Goldensohn, S., Blitzer, J., Çelebi, A., Dimitrov, S., Drabek, E., Hakim, A., Lam, W., Liu, D., Otterbacher, J., Qi, H., Saggion, H., Teufel, S., Topper, M., Winkel, A., and Zhang, Z. (2004). MEAD - a platform for multidocument multilingual text summarization. In *LREC 2004*, Lisbon, Portugal.

[Robertson, 2004] Robertson, S. (2004). Understanding inverse document frequency: on theoretical arguments for idf. *Journal of documentation*, 60(5):503–520.

[Roelleke, 2003] Roelleke, T. (2003). A frequency-based and a poisson-based definition of the probability of being informative. In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, pages 227–234. ACM.

[Roelleke, 2013] Roelleke, T. (2013). *Information Retrieval Models: Foundations and Relationships*, volume 5. Morgan & Claypool Publishers.

[Salton, 1971] Salton, G. (1971). *The SMART retrieval system-experiments in automatic document processing*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA.

[Sebastiani, 2002] Sebastiani, F. (2002). Machine learning in automated text categorization. *ACM computing surveys (CSUR)*, 34(1):1–47.

[Sharifi et al., 2010] Sharifi, B., Hutton, M., and Kalita, J. (2010). Experiments in microblog summarization. In *Social Computing (SocialCom), 2010 IEEE Second International Conference on*, pages 49–56. IEEE.

[Smith and Smith, 1977] Smith, J. and Smith, D. (1977). Database abstractions: aggregation and generalization. *ACM Transactions on Database Systems (TODS)*, 2(2):105–133.

[Spärck Jones, 1972] Spärck Jones, K. (1972). A statistical interpretation of term specificity and its application in retrieval. *J. Doc*, 28(1):11–20.

[Titov and McDonald, 2008] Titov, I. and McDonald, R. (2008). Modeling online reviews with multi-grain topic models. In *Proceedings of the 17th international conference on World Wide Web*, pages 111–120. ACM.

[Tombros and Sanderson, 1998] Tombros, A. and Sanderson, M. (1998). Advantages of query biased summaries in information retrieval. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 2–10. ACM.

[Turney, 2002] Turney, P. (2002). Thumbs up or thumbs down? semantic orientation applied to unsupervised classification of reviews. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL'02)*.

[Turney and Littman, 2003] Turney, P. D. and Littman, M. L. (2003). Measuring praise and criticism: Inference of semantic orientation from association. *ACM Transactions on Information Systems (TOIS)*, 21(4):315–346.

[van Rijsbergen, 1979] van Rijsbergen, C. J. (1979). *Information Retrieval*. Butterworths, London, United Kingdom.

[Van Zwol and Van Loosbroek, 2007] Van Zwol, R. and Van Loosbroek, T. (2007). Effective use of semantic structure in xml retrieval. In *Advances in Information Retrieval*, pages 621–628. Springer, Berlin, Germany.

[Wilson et al., 2005a] Wilson, T., Hoffmann, P., Somasundaran, S., Kessler, J., Wiebe, J., Choi, Y., Cardie, C., Riloff, E., and Patwardhan, S. (2005a). Opinionfinder: A system for subjectivity analysis. In *Proceedings of HLT/EMNLP on Interactive Demonstrations*, pages 34–35. Association for Computational Linguistics.

[Wilson et al., 2005b] Wilson, T., Wiebe, J., and Hoffmann, P. (2005b). Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of the conference on human language technology and empirical methods in natural language processing*, pages 347–354. Association for Computational Linguistics.

[Zaniolo and Melkaoff, 1982] Zaniolo, C. and Melkaoff, M. (1982). A formal approach to the definition and the design of conceptual schemata for databased systems. *ACM Transactions on Database Systems (TODS)*, 7(1):24–59.

[Zhai, 2008] Zhai, C. (2008). Statistical language models for information retrieval. *Synthesis Lectures on Human Language Technologies*, 1(1):1–141.

[Zhang and Liu, 2011] Zhang, L. and Liu, B. (2011). Identifying noun product features that imply opinions. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers*, volume 2, pages 575–580, Stroudsburg, PA, USA. The Association for Computational Linguistics.

[Zhuang et al., 2006] Zhuang, L., Jing, F., and Zhu, X. (2006). Movie review mining and summarization. In *Proceedings of the 15th ACM international conference on Information and knowledge management*, pages 43–50. ACM.

[Zvieli and Chen, 1986] Zvieli, A. and Chen, P. P. (1986). Entity-relationship modeling and fuzzy databases. In *Proceedings of the Second International Conference on Data Engineering*, pages 320–327, Piscataway, NJ, USA. IEEE Computer Society, IEEE.

Appendix A

Treatment of Opinion-bearing Terms: Complete Experimental Results

---

This appendix contains all the experimental results for the intrinsic sentiment summarisation task run over the Opinosis data-set as described in Section 4.3.

For each particular treatment of opinion-bearing terms, the eight candidates described in Section 3.4 (the MEAD baseline not included) are evaluated on the Opinosis data-set. The metrics reported are ROUGE-1, ROUGE-2 and ROUGE-SU4 and, for each metric, precision, recall and F_1-scores are measured.

All the different treatments of opinion-bearing terms have been discussed in Section 4.3. Figure A.1 shows a summary of such treatments which are detailed in the following sections.

Treatment	Description
Opinions as Stop-words	Opinion terms are treated as stop-words, i.e. removed
Boosting frequencies	Opinion terms are repeated *in loco* a number of times ($2 \leq n \leq 10$)
Opinion-based bigrams	Unigram features are combined with opinion-based unigrams.
Negation removed	Terms which follow a negation are removed (window size=1)

Figure A.1: List of treatments of opinion-bearing terms analysed in the experiments on intrinsic summarisation.

A.1 Opinions as Stop-words

ROUGE-1 - Opinions as stop-words				
Candidate	Metric	Measure	Score	Confidence interval
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-1	Recall	30.92	(95%-conf.int. 0.28281 - 0.33690)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-1	Precision	28.65	(95%-conf.int. 0.25791 - 0.31384)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-1	F_1-score	27.99	(95%-conf.int. 0.25844 - 0.30187)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-1	Recall	25.52	(95%-conf.int. 0.23041 - 0.27967)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-1	Precision	31.89	(95%-conf.int. 0.29826 - 0.34066)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-1	F_1-score	27.72	(95%-conf.int. 0.25571 - 0.29796)
$\text{BF}_{\texttt{SIM}}$	ROUGE-1	Recall	22.11	(95%-conf.int. 0.20174 - 0.23994)
$\text{BF}_{\texttt{SIM}}$	ROUGE-1	Precision	28.54	(95%-conf.int. 0.26066 - 0.31248)
$\text{BF}_{\texttt{SIM}}$	ROUGE-1	F_1-score	24.14	(95%-conf.int. 0.22466 - 0.25917)
$\text{BF}_{\texttt{DIV}}$	ROUGE-1	Recall	20.91	(95%-conf.int. 0.19057 - 0.22746)
$\text{BF}_{\texttt{DIV}}$	ROUGE-1	Precision	30.80	(95%-conf.int. 0.28756 - 0.32977)
$\text{BF}_{\texttt{DIV}}$	ROUGE-1	F_1-score	24.38	(95%-conf.int. 0.22618 - 0.26186)
$\text{SR}_{\texttt{SIM}}$	ROUGE-1	Recall	38.69	(95%-conf.int. 0.36878 - 0.40620)
$\text{SR}_{\texttt{SIM}}$	ROUGE-1	Precision	18.40	(95%-conf.int. 0.16286 - 0.20798)
$\text{SR}_{\texttt{SIM}}$	ROUGE-1	F_1-score	23.75	(95%-conf.int. 0.21907 - 0.25579)
$\text{SR}_{\texttt{DIV}}$	ROUGE-1	Recall	46.71	(95%-conf.int. 0.44832 - 0.48669)
$\text{SR}_{\texttt{DIV}}$	ROUGE-1	Precision	10.22	(95%-conf.int. 0.09285 - 0.11263)
$\text{SR}_{\texttt{DIV}}$	ROUGE-1	F_1-score	16.43	(95%-conf.int. 0.15207 - 0.17742)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-1	Recall	16.35	(95%-conf.int. 0.14796 - 0.17986)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-1	Precision	25.89	(95%-conf.int. 0.23349 - 0.28434)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-1	F_1-score	19.60	(95%-conf.int. 0.17932 - 0.21272)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-1	Recall	16.61	(95%-conf.int. 0.14743 - 0.18574)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-1	Precision	12.90	(95%-conf.int. 0.11627 - 0.14144)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-1	F_1-score	13.91	(95%-conf.int. 0.12591 - 0.15149)

Figure A.2: ROUGE-1 scores on the Opinosis data-set. Opinion-bearing terms treated as stop-words, i.e. removed.

ROUGE-2 - Opinions as stop-words				
Candidate	Metric	Measure	Score	Confidence interval
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-2	Recall	08.95	(95%-conf.int. 0.07498 - 0.10469)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-2	Precision	08.49	(95%-conf.int. 0.06905 - 0.10111)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-2	F_1-score	08.20	(95%-conf.int. 0.06749 - 0.09644)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-2	Recall	07.36	(95%-conf.int. 0.05742 - 0.08873)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-2	Precision	08.82	(95%-conf.int. 0.07160 - 0.10463)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-2	F_1-score	07.85	(95%-conf.int. 0.06264 - 0.09398)
$\text{BF}_{\texttt{SIM}}$	ROUGE-2	Recall	05.39	(95%-conf.int. 0.04287 - 0.06603)
$\text{BF}_{\texttt{SIM}}$	ROUGE-2	Precision	07.04	(95%-conf.int. 0.05550 - 0.08627)
$\text{BF}_{\texttt{SIM}}$	ROUGE-2	F_1-score	05.88	(95%-conf.int. 0.04652 - 0.07106)
$\text{BF}_{\texttt{DIV}}$	ROUGE-2	Recall	05.57	(95%-conf.int. 0.04415 - 0.06834)
$\text{BF}_{\texttt{DIV}}$	ROUGE-2	Precision	08.18	(95%-conf.int. 0.06548 - 0.09762)
$\text{BF}_{\texttt{DIV}}$	ROUGE-2	F_1-score	06.46	(95%-conf.int. 0.05165 - 0.07739)
$\text{SR}_{\texttt{SIM}}$	ROUGE-2	Recall	10.16	(95%-conf.int. 0.08728 - 0.11533)
$\text{SR}_{\texttt{SIM}}$	ROUGE-2	Precision	04.96	(95%-conf.int. 0.03994 - 0.06041)
$\text{SR}_{\texttt{SIM}}$	ROUGE-2	F_1-score	06.33	(95%-conf.int. 0.05296 - 0.07347)
$\text{SR}_{\texttt{DIV}}$	ROUGE-2	Recall	08.54	(95%-conf.int. 0.07259 - 0.09811)
$\text{SR}_{\texttt{DIV}}$	ROUGE-2	Precision	01.81	(95%-conf.int. 0.01499 - 0.02128)
$\text{SR}_{\texttt{DIV}}$	ROUGE-2	F_1-score	02.92	(95%-conf.int. 0.02451 - 0.03404)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-2	Recall	02.93	(95%-conf.int. 0.02176 - 0.03726)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-2	Precision	05.05	(95%-conf.int. 0.03663 - 0.06672)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-2	F_1-score	03.60	(95%-conf.int. 0.02714 - 0.04597)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-2	Recall	01.81	(95%-conf.int. 0.01198 - 0.02493)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-2	Precision	01.49	(95%-conf.int. 0.00948 - 0.02067)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-2	F_1-score	01.56	(95%-conf.int. 0.01018 - 0.02143)

Figure A.3: ROUGE-2 scores on the Opinosis data-set. Opinion-bearing terms treated as stop-words, i.e. removed.

ROUGE-SU4 - Opinions as stop-words				
Candidate	Metric	Measure	Score	Confidence interval
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-SU*	Recall	10.70	(95%-conf.int. 0.09150 - 0.12397)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-SU*	Precision	10.78	(95%-conf.int. 0.09000 - 0.12500)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-SU*	F_1-score	08.81	(95%-conf.int. 0.07551 - 0.10112)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-SU*	Recall	07.16	(95%-conf.int. 0.05929 - 0.08412)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-SU*	Precision	12.05	(95%-conf.int. 0.10624 - 0.13539)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-SU*	F_1-score	08.31	(95%-conf.int. 0.07138 - 0.09590)
$\text{BF}_{\texttt{SIM}}$	ROUGE-SU*	Recall	05.52	(95%-conf.int. 0.04544 - 0.06504)
$\text{BF}_{\texttt{SIM}}$	ROUGE-SU*	Precision	10.01	(95%-conf.int. 0.08537 - 0.11677)
$\text{BF}_{\texttt{SIM}}$	ROUGE-SU*	F_1-score	06.33	(95%-conf.int. 0.05527 - 0.07244)
$\text{BF}_{\texttt{DIV}}$	ROUGE-SU*	Recall	05.05	(95%-conf.int. 0.04216 - 0.05921)
$\text{BF}_{\texttt{DIV}}$	ROUGE-SU*	Precision	11.85	(95%-conf.int. 0.10435 - 0.13343)
$\text{BF}_{\texttt{DIV}}$	ROUGE-SU*	F_1-score	06.60	(95%-conf.int. 0.05727 - 0.07527)
$\text{SR}_{\texttt{SIM}}$	ROUGE-SU*	Recall	14.49	(95%-conf.int. 0.13140 - 0.15914)
$\text{SR}_{\texttt{SIM}}$	ROUGE-SU*	Precision	04.41	(95%-conf.int. 0.03416 - 0.05610)
$\text{SR}_{\texttt{SIM}}$	ROUGE-SU*	F_1-score	05.68	(95%-conf.int. 0.04866 - 0.06495)
$\text{SR}_{\texttt{DIV}}$	ROUGE-SU*	Recall	20.65	(95%-conf.int. 0.18873 - 0.22510)
$\text{SR}_{\texttt{DIV}}$	ROUGE-SU*	Precision	01.26	(95%-conf.int. 0.01032 - 0.01531)
$\text{SR}_{\texttt{DIV}}$	ROUGE-SU*	F_1-score	02.29	(95%-conf.int. 0.01918 - 0.02739)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-SU*	Recall	03.08	(95%-conf.int. 0.02535 - 0.03686)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-SU*	Precision	08.58	(95%-conf.int. 0.07029 - 0.10307)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-SU*	F_1-score	04.19	(95%-conf.int. 0.03567 - 0.04899)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-SU*	Recall	03.45	(95%-conf.int. 0.02749 - 0.04254)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-SU*	Precision	02.32	(95%-conf.int. 0.01944 - 0.02670)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-SU*	F_1-score	02.37	(95%-conf.int. 0.01995 - 0.02754)

Figure A.4: ROUGE-SU4 scores on the Opinosis data-set. Opinion-bearing terms treated as stop-words, i.e. removed.

A.2 Boosting Frequencies

ROUGE-1 - Frequency of opinion-bearing terms boosted $\times 2$				
Candidate	Metric	Measure	Score	Confidence interval
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-1	Recall	32.44	(95%-conf.int. 0.30064 - 0.34859)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-1	Precision	27.53	(95%-conf.int. 0.25208 - 0.29854)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-1	F_1-score	28.43	(95%-conf.int. 0.26639 - 0.30167)
$\text{BF}_{\texttt{SIM}}$	ROUGE-1	Recall	25.82	(95%-conf.int. 0.24326 - 0.27387)
$\text{BF}_{\texttt{SIM}}$	ROUGE-1	Precision	31.27	(95%-conf.int. 0.28945 - 0.33704)
$\text{BF}_{\texttt{SIM}}$	ROUGE-1	F_1-score	27.56	(95%-conf.int. 0.26070 - 0.29201)
$\text{SR}_{\texttt{SIM}}$	ROUGE-1	Recall	37.37	(95%-conf.int. 0.35090 - 0.39739)
$\text{SR}_{\texttt{SIM}}$	ROUGE-1	Precision	20.26	(95%-conf.int. 0.18052 - 0.22536)
$\text{SR}_{\texttt{SIM}}$	ROUGE-1	F_1-score	24.89	(95%-conf.int. 0.23191 - 0.26558)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-1	Recall	18.43	(95%-conf.int. 0.16895 - 0.20109)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-1	Precision	28.45	(95%-conf.int. 0.25840 - 0.31092)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-1	F_1-score	21.89	(95%-conf.int. 0.20202 - 0.23551)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-1	Recall	26.57	(95%-conf.int. 0.24338 - 0.29042)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-1	Precision	32.51	(95%-conf.int. 0.30511 - 0.34629)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-1	F_1-score	28.55	(95%-conf.int. 0.26673 - 0.30500)
$\text{BF}_{\texttt{DIV}}$	ROUGE-1	Recall	21.58	(95%-conf.int. 0.19190 - 0.23984)
$\text{BF}_{\texttt{DIV}}$	ROUGE-1	Precision	31.08	(95%-conf.int. 0.28820 - 0.33555)
$\text{BF}_{\texttt{DIV}}$	ROUGE-1	F_1-score	24.89	(95%-conf.int. 0.22747 - 0.27049)
$\text{SR}_{\texttt{DIV}}$	ROUGE-1	Recall	47.48	(95%-conf.int. 0.45257 - 0.49667)
$\text{SR}_{\texttt{DIV}}$	ROUGE-1	Precision	09.98	(95%-conf.int. 0.09166 - 0.10904)
$\text{SR}_{\texttt{DIV}}$	ROUGE-1	F_1-score	16.12	(95%-conf.int. 0.15070 - 0.17246)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-1	Recall	17.90	(95%-conf.int. 0.16064 - 0.19776)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-1	Precision	14.29	(95%-conf.int. 0.12918 - 0.15751)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-1	F_1-score	15.27	(95%-conf.int. 0.14052 - 0.16515)

Figure A.5: ROUGE-1 scores on the Opinosis data-set. Frequency of opinion-bearing terms boosted $\times 2$.

ROUGE-2 - Frequency of opinion-bearing terms boosted $\times 2$				
Candidate	Metric	Measure	Score	Confidence interval
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-2	Recall	09.35	(95%-conf.int. 0.07911 - 0.10807)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-2	Precision	08.01	(95%-conf.int. 0.06777 - 0.09375)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-2	F_1-score	08.19	(95%-conf.int. 0.07095 - 0.09385)
$\text{BF}_{\texttt{SIM}}$	ROUGE-2	Recall	07.57	(95%-conf.int. 0.06531 - 0.08692)
$\text{BF}_{\texttt{SIM}}$	ROUGE-2	Precision	09.64	(95%-conf.int. 0.08289 - 0.11101)
$\text{BF}_{\texttt{SIM}}$	ROUGE-2	F_1-score	08.27	(95%-conf.int. 0.07185 - 0.09412)
$\text{SR}_{\texttt{SIM}}$	ROUGE-2	Recall	09.07	(95%-conf.int. 0.07671 - 0.10452)
$\text{SR}_{\texttt{SIM}}$	ROUGE-2	Precision	04.89	(95%-conf.int. 0.03977 - 0.05861)
$\text{SR}_{\texttt{SIM}}$	ROUGE-2	F_1-score	06.02	(95%-conf.int. 0.05028 - 0.07041)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-2	Recall	04.15	(95%-conf.int. 0.03089 - 0.05309)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-2	Precision	06.50	(95%-conf.int. 0.05020 - 0.08030)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-2	F_1-score	04.94	(95%-conf.int. 0.03762 - 0.06175)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-2	Recall	07.46	(95%-conf.int. 0.06116 - 0.08926)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-2	Precision	09.15	(95%-conf.int. 0.07570 - 0.10875)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-2	F_1-score	08.03	(95%-conf.int. 0.06610 - 0.09512)
$\text{BF}_{\texttt{DIV}}$	ROUGE-2	Recall	06.08	(95%-conf.int. 0.04838 - 0.07473)
$\text{BF}_{\texttt{DIV}}$	ROUGE-2	Precision	08.67	(95%-conf.int. 0.07089 - 0.10246)
$\text{BF}_{\texttt{DIV}}$	ROUGE-2	F_1-score	06.97	(95%-conf.int. 0.05635 - 0.08358)
$\text{SR}_{\texttt{DIV}}$	ROUGE-2	Recall	09.93	(95%-conf.int. 0.08381 - 0.11506)
$\text{SR}_{\texttt{DIV}}$	ROUGE-2	Precision	01.99	(95%-conf.int. 0.01673 - 0.02309)
$\text{SR}_{\texttt{DIV}}$	ROUGE-2	F_1-score	03.24	(95%-conf.int. 0.02734 - 0.03740)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-2	Recall	02.29	(95%-conf.int. 0.01573 - 0.03021)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-2	Precision	01.73	(95%-conf.int. 0.01195 - 0.02381)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-2	F_1-score	01.89	(95%-conf.int. 0.01310 - 0.02530)

Figure A.6: ROUGE-2 scores on the Opinosis data-set. Frequency of opinion-bearing terms boosted $\times 2$.

ROUGE-SU4 - Frequency of opinion-bearing terms boosted $\times 2$				
Candidate	Metric	Measure	Score	Confidence interval
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-SU*	Recall	11.26	(95%-conf.int. 0.09776 - 0.12867)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-SU*	Precision	09.50	(95%-conf.int. 0.08053 - 0.11030)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-SU*	F_1-score	08.73	(95%-conf.int. 0.07711 - 0.09769)
$\text{BF}_{\texttt{SIM}}$	ROUGE-SU*	Recall	07.02	(95%-conf.int. 0.06227 - 0.07913)
$\text{BF}_{\texttt{SIM}}$	ROUGE-SU*	Precision	11.65	(95%-conf.int. 0.10131 - 0.13257)
$\text{BF}_{\texttt{SIM}}$	ROUGE-SU*	F_1-score	08.00	(95%-conf.int. 0.07190 - 0.08857)
$\text{SR}_{\texttt{SIM}}$	ROUGE-SU*	Recall	13.84	(95%-conf.int. 0.12166 - 0.15630)
$\text{SR}_{\texttt{SIM}}$	ROUGE-SU*	Precision	05.04	(95%-conf.int. 0.04025 - 0.06096)
$\text{SR}_{\texttt{SIM}}$	ROUGE-SU*	F_1-score	06.20	(95%-conf.int. 0.05350 - 0.07096)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-SU*	Recall	03.89	(95%-conf.int. 0.03265 - 0.04550)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-SU*	Precision	10.25	(95%-conf.int. 0.08721 - 0.11929)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-SU*	F_1-score	05.27	(95%-conf.int. 0.04531 - 0.06052)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-SU*	Recall	07.66	(95%-conf.int. 0.06478 - 0.08981)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-SU*	Precision	12.45	(95%-conf.int. 0.11042 - 0.13977)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-SU*	F_1-score	08.73	(95%-conf.int. 0.07619 - 0.09901)
$\text{BF}_{\texttt{DIV}}$	ROUGE-SU*	Recall	05.55	(95%-conf.int. 0.04460 - 0.06722)
$\text{BF}_{\texttt{DIV}}$	ROUGE-SU*	Precision	12.34	(95%-conf.int. 0.10832 - 0.14009)
$\text{BF}_{\texttt{DIV}}$	ROUGE-SU*	F_1-score	07.11	(95%-conf.int. 0.05996 - 0.08304)
$\text{SR}_{\texttt{DIV}}$	ROUGE-SU*	Recall	21.35	(95%-conf.int. 0.19240 - 0.23332)
$\text{SR}_{\texttt{DIV}}$	ROUGE-SU*	Precision	01.18	(95%-conf.int. 0.00990 - 0.01385)
$\text{SR}_{\texttt{DIV}}$	ROUGE-SU*	F_1-score	02.15	(95%-conf.int. 0.01844 - 0.02478)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-SU*	Recall	03.80	(95%-conf.int. 0.03070 - 0.04619)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-SU*	Precision	02.71	(95%-conf.int. 0.02267 - 0.03218)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-SU*	F_1-score	02.69	(95%-conf.int. 0.02342 - 0.03067)

Figure A.7: ROUGE-SU4 scores on the Opinosis data-set. Frequency of opinion-bearing terms boosted $\times 2$.

ROUGE-1 - Frequency of opinion-bearing terms boosted $\times 3$				
Candidate	Metric	Measure	Score	Confidence interval
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-1	Recall	32.98	(95%-conf.int. 0.30463 - 0.35561)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-1	Precision	27.56	(95%-conf.int. 0.25188 - 0.29836)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-1	F_1-score	28.64	(95%-conf.int. 0.26874 - 0.30350)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-1	Recall	26.53	(95%-conf.int. 0.24367 - 0.28708)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-1	Precision	32.67	(95%-conf.int. 0.30445 - 0.34866)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-1	F_1-score	28.67	(95%-conf.int. 0.26805 - 0.30612)
$\text{BF}_{\texttt{SIM}}$	ROUGE-1	Recall	26.53	(95%-conf.int. 0.24794 - 0.28287)
$\text{BF}_{\texttt{SIM}}$	ROUGE-1	Precision	31.09	(95%-conf.int. 0.28700 - 0.33597)
$\text{BF}_{\texttt{SIM}}$	ROUGE-1	F_1-score	27.86	(95%-conf.int. 0.26209 - 0.29499)
$\text{BF}_{\texttt{DIV}}$	ROUGE-1	Recall	21.96	(95%-conf.int. 0.19636 - 0.24242)
$\text{BF}_{\texttt{DIV}}$	ROUGE-1	Precision	30.96	(95%-conf.int. 0.28667 - 0.33358)
$\text{BF}_{\texttt{DIV}}$	ROUGE-1	F_1-score	25.12	(95%-conf.int. 0.22973 - 0.27177)
$\text{SR}_{\texttt{SIM}}$	ROUGE-1	Recall	36.92	(95%-conf.int. 0.34499 - 0.39350)
$\text{SR}_{\texttt{SIM}}$	ROUGE-1	Precision	22.00	(95%-conf.int. 0.19755 - 0.24081)
$\text{SR}_{\texttt{SIM}}$	ROUGE-1	F_1-score	26.28	(95%-conf.int. 0.24378 - 0.28074)
$\text{SR}_{\texttt{DIV}}$	ROUGE-1	Recall	46.33	(95%-conf.int. 0.44266 - 0.48429)
$\text{SR}_{\texttt{DIV}}$	ROUGE-1	Precision	10.24	(95%-conf.int. 0.09179 - 0.11429)
$\text{SR}_{\texttt{DIV}}$	ROUGE-1	F_1-score	16.30	(95%-conf.int. 0.15057 - 0.17648)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-1	Recall	19.24	(95%-conf.int. 0.17532 - 0.21049)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-1	Precision	28.26	(95%-conf.int. 0.25721 - 0.30960)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-1	F_1-score	22.42	(95%-conf.int. 0.20630 - 0.24219)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-1	Recall	18.24	(95%-conf.int. 0.16382 - 0.20246)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-1	Precision	14.45	(95%-conf.int. 0.13007 - 0.16007)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-1	F_1-score	15.51	(95%-conf.int. 0.14131 - 0.16841)

Figure A.8: ROUGE-1 scores on the Opinosis data-set. Frequency of opinion-bearing terms boosted $\times 3$.

ROUGE-2 - Frequency of opinion-bearing terms boosted $\times 3$				
Candidate	Metric	Measure	Score	Confidence interval
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-2	Recall	09.92	(95%-conf.int. 0.08204 - 0.11881)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-2	Precision	08.12	(95%-conf.int. 0.06808 - 0.09464)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-2	F_1-score	08.42	(95%-conf.int. 0.07152 - 0.09742)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-2	Recall	07.40	(95%-conf.int. 0.06103 - 0.08826)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-2	Precision	09.09	(95%-conf.int. 0.07682 - 0.10522)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-2	F_1-score	08.00	(95%-conf.int. 0.06710 - 0.09417)
$\text{BF}_{\texttt{SIM}}$	ROUGE-2	Recall	07.39	(95%-conf.int. 0.06281 - 0.08524)
$\text{BF}_{\texttt{SIM}}$	ROUGE-2	Precision	09.05	(95%-conf.int. 0.07699 - 0.10618)
$\text{BF}_{\texttt{SIM}}$	ROUGE-2	F_1-score	07.90	(95%-conf.int. 0.06791 - 0.09151)
$\text{BF}_{\texttt{DIV}}$	ROUGE-2	Recall	06.16	(95%-conf.int. 0.04881 - 0.07551)
$\text{BF}_{\texttt{DIV}}$	ROUGE-2	Precision	08.61	(95%-conf.int. 0.07024 - 0.10179)
$\text{BF}_{\texttt{DIV}}$	ROUGE-2	F_1-score	07.02	(95%-conf.int. 0.05652 - 0.08414)
$\text{SR}_{\texttt{SIM}}$	ROUGE-2	Recall	09.49	(95%-conf.int. 0.07828 - 0.11273)
$\text{SR}_{\texttt{SIM}}$	ROUGE-2	Precision	05.55	(95%-conf.int. 0.04436 - 0.06623)
$\text{SR}_{\texttt{SIM}}$	ROUGE-2	F_1-score	06.70	(95%-conf.int. 0.05470 - 0.07965)
$\text{SR}_{\texttt{DIV}}$	ROUGE-2	Recall	09.44	(95%-conf.int. 0.07883 - 0.10972)
$\text{SR}_{\texttt{DIV}}$	ROUGE-2	Precision	01.96	(95%-conf.int. 0.01621 - 0.02311)
$\text{SR}_{\texttt{DIV}}$	ROUGE-2	F_1-score	03.15	(95%-conf.int. 0.02649 - 0.03685)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-2	Recall	04.57	(95%-conf.int. 0.03434 - 0.05773)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-2	Precision	06.69	(95%-conf.int. 0.05216 - 0.08262)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-2	F_1-score	05.30	(95%-conf.int. 0.04073 - 0.06553)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-2	Recall	02.38	(95%-conf.int. 0.01632 - 0.03106)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-2	Precision	01.81	(95%-conf.int. 0.01244 - 0.02447)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-2	F_1-score	01.97	(95%-conf.int. 0.01351 - 0.02628)

Figure A.9: ROUGE-2 scores on the Opinosis data-set. Frequency of opinion-bearing terms boosted $\times 3$.

ROUGE-SU4 - Frequency of opinion-bearing terms boosted $\times 3$				
Candidate	Metric	Measure	Score	Confidence interval
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-SU*	Recall	11.54	(95%-conf.int. 0.09800 - 0.13598)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-SU*	Precision	09.39	(95%-conf.int. 0.07851 - 0.10915)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-SU*	F_1-score	08.74	(95%-conf.int. 0.07706 - 0.09880)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-SU*	Recall	07.57	(95%-conf.int. 0.06405 - 0.08784)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-SU*	Precision	12.47	(95%-conf.int. 0.11084 - 0.13919)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-SU*	F_1-score	08.75	(95%-conf.int. 0.07649 - 0.09871)
$\text{BF}_{\texttt{SIM}}$	ROUGE-SU*	Recall	07.29	(95%-conf.int. 0.06395 - 0.08321)
$\text{BF}_{\texttt{SIM}}$	ROUGE-SU*	Precision	11.37	(95%-conf.int. 0.09817 - 0.13011)
$\text{BF}_{\texttt{SIM}}$	ROUGE-SU*	F_1-score	08.07	(95%-conf.int. 0.07155 - 0.09072)
$\text{BF}_{\texttt{DIV}}$	ROUGE-SU*	Recall	05.70	(95%-conf.int. 0.04657 - 0.06790)
$\text{BF}_{\texttt{DIV}}$	ROUGE-SU*	Precision	12.18	(95%-conf.int. 0.10681 - 0.13836)
$\text{BF}_{\texttt{DIV}}$	ROUGE-SU*	F_1-score	07.22	(95%-conf.int. 0.06089 - 0.08396)
$\text{SR}_{\texttt{SIM}}$	ROUGE-SU*	Recall	13.59	(95%-conf.int. 0.11876 - 0.15496)
$\text{SR}_{\texttt{SIM}}$	ROUGE-SU*	Precision	06.01	(95%-conf.int. 0.04881 - 0.07147)
$\text{SR}_{\texttt{SIM}}$	ROUGE-SU*	F_1-score	07.18	(95%-conf.int. 0.06122 - 0.08211)
$\text{SR}_{\texttt{DIV}}$	ROUGE-SU*	Recall	20.25	(95%-conf.int. 0.18340 - 0.22270)
$\text{SR}_{\texttt{DIV}}$	ROUGE-SU*	Precision	01.27	(95%-conf.int. 0.01018 - 0.01572)
$\text{SR}_{\texttt{DIV}}$	ROUGE-SU*	F_1-score	02.27	(95%-conf.int. 0.01875 - 0.02723)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-SU*	Recall	04.25	(95%-conf.int. 0.03577 - 0.05006)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-SU*	Precision	09.97	(95%-conf.int. 0.08571 - 0.11506)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-SU*	F_1-score	05.57	(95%-conf.int. 0.04803 - 0.06394)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-SU*	Recall	03.88	(95%-conf.int. 0.03112 - 0.04760)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-SU*	Precision	02.78	(95%-conf.int. 0.02269 - 0.03344)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-SU*	F_1-score	02.78	(95%-conf.int. 0.02346 - 0.03251)

Figure A.10: ROUGE-SU4 scores on the Opinosis data-set. Frequency of opinion-bearing terms boosted $\times 3$.

ROUGE-1 - Frequency of opinion-bearing terms boosted $\times 4$				
Candidate	Metric	Measure	Score	Confidence interval
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-1	Recall	33.28	(95%-conf.int. 0.30682 - 0.35926)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-1	Precision	27.34	(95%-conf.int. 0.24949 - 0.29731)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-1	F_1-score	28.59	(95%-conf.int. 0.26783 - 0.30453)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-1	Recall	26.61	(95%-conf.int. 0.24038 - 0.29031)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-1	Precision	32.75	(95%-conf.int. 0.30529 - 0.35041)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-1	F_1-score	28.68	(95%-conf.int. 0.26472 - 0.30889)
$\text{BF}_{\texttt{SIM}}$	ROUGE-1	Recall	26.64	(95%-conf.int. 0.24805 - 0.28471)
$\text{BF}_{\texttt{SIM}}$	ROUGE-1	Precision	30.84	(95%-conf.int. 0.28469 - 0.33320)
$\text{BF}_{\texttt{SIM}}$	ROUGE-1	F_1-score	27.69	(95%-conf.int. 0.25978 - 0.29438)
$\text{BF}_{\texttt{DIV}}$	ROUGE-1	Recall	21.98	(95%-conf.int. 0.19677 - 0.24272)
$\text{BF}_{\texttt{DIV}}$	ROUGE-1	Precision	30.91	(95%-conf.int. 0.28658 - 0.33339)
$\text{BF}_{\texttt{DIV}}$	ROUGE-1	F_1-score	25.13	(95%-conf.int. 0.22977 - 0.27180)
$\text{SR}_{\texttt{SIM}}$	ROUGE-1	Recall	36.50	(95%-conf.int. 0.33846 - 0.38982)
$\text{SR}_{\texttt{SIM}}$	ROUGE-1	Precision	22.11	(95%-conf.int. 0.19746 - 0.24442)
$\text{SR}_{\texttt{SIM}}$	ROUGE-1	F_1-score	25.86	(95%-conf.int. 0.23858 - 0.27812)
$\text{SR}_{\texttt{DIV}}$	ROUGE-1	Recall	46.02	(95%-conf.int. 0.43806 - 0.48156)
$\text{SR}_{\texttt{DIV}}$	ROUGE-1	Precision	10.23	(95%-conf.int. 0.09246 - 0.11303)
$\text{SR}_{\texttt{DIV}}$	ROUGE-1	F_1-score	16.30	(95%-conf.int. 0.15046 - 0.17589)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-1	Recall	19.87	(95%-conf.int. 0.18002 - 0.21811)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-1	Precision	27.79	(95%-conf.int. 0.25591 - 0.30003)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-1	F_1-score	22.74	(95%-conf.int. 0.20931 - 0.24544)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-1	Recall	17.57	(95%-conf.int. 0.15896 - 0.19171)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-1	Precision	14.55	(95%-conf.int. 0.13099 - 0.16161)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-1	F_1-score	15.38	(95%-conf.int. 0.14107 - 0.16655)

Figure A.11: ROUGE-1 scores on the Opinosis data-set. Frequency of opinion-bearing terms boosted $\times 4$.

ROUGE-2 - Frequency of opinion-bearing terms boosted $\times 4$				
Candidate	Metric	Measure	Score	Confidence interval
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-2	Recall	09.92	(95%-conf.int. 0.08207 - 0.11969)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-2	Precision	07.96	(95%-conf.int. 0.06685 - 0.09340)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-2	F_1-score	08.32	(95%-conf.int. 0.07003 - 0.09666)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-2	Recall	07.29	(95%-conf.int. 0.05891 - 0.08811)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-2	Precision	08.88	(95%-conf.int. 0.07362 - 0.10374)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-2	F_1-score	07.84	(95%-conf.int. 0.06430 - 0.09311)
$\text{BF}_{\texttt{SIM}}$	ROUGE-2	Recall	07.86	(95%-conf.int. 0.06747 - 0.08948)
$\text{BF}_{\texttt{SIM}}$	ROUGE-2	Precision	09.40	(95%-conf.int. 0.08057 - 0.10910)
$\text{BF}_{\texttt{SIM}}$	ROUGE-2	F_1-score	08.28	(95%-conf.int. 0.07155 - 0.09483)
$\text{BF}_{\texttt{DIV}}$	ROUGE-2	Recall	06.18	(95%-conf.int. 0.04917 - 0.07559)
$\text{BF}_{\texttt{DIV}}$	ROUGE-2	Precision	08.63	(95%-conf.int. 0.07053 - 0.10202)
$\text{BF}_{\texttt{DIV}}$	ROUGE-2	F_1-score	07.04	(95%-conf.int. 0.05679 - 0.08414)
$\text{SR}_{\texttt{SIM}}$	ROUGE-2	Recall	09.36	(95%-conf.int. 0.07754 - 0.11097)
$\text{SR}_{\texttt{SIM}}$	ROUGE-2	Precision	05.67	(95%-conf.int. 0.04580 - 0.06748)
$\text{SR}_{\texttt{SIM}}$	ROUGE-2	F_1-score	06.63	(95%-conf.int. 0.05474 - 0.07811)
$\text{SR}_{\texttt{DIV}}$	ROUGE-2	Recall	08.79	(95%-conf.int. 0.07200 - 0.10417)
$\text{SR}_{\texttt{DIV}}$	ROUGE-2	Precision	01.79	(95%-conf.int. 0.01456 - 0.02141)
$\text{SR}_{\texttt{DIV}}$	ROUGE-2	F_1-score	02.90	(95%-conf.int. 0.02376 - 0.03443)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-2	Recall	04.65	(95%-conf.int. 0.03431 - 0.05836)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-2	Precision	06.44	(95%-conf.int. 0.04990 - 0.07920)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-2	F_1-score	05.28	(95%-conf.int. 0.03994 - 0.06530)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-2	Recall	02.32	(95%-conf.int. 0.01579 - 0.03058)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-2	Precision	01.81	(95%-conf.int. 0.01256 - 0.02413)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-2	F_1-score	01.95	(95%-conf.int. 0.01353 - 0.02558)

Figure A.12: ROUGE-2 scores on the Opinosis data-set. Frequency of opinion-bearing terms boosted $\times 4$.

ROUGE-SU4 - Frequency of opinion-bearing terms boosted $\times 4$				
Candidate	Metric	Measure	Score	Confidence interval
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-SU*	Recall	11.65	(95%-conf.int. 0.09915 - 0.13689)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-SU*	Precision	09.26	(95%-conf.int. 0.07698 - 0.10769)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-SU*	F_1-score	08.69	(95%-conf.int. 0.07612 - 0.09817)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-SU*	Recall	07.71	(95%-conf.int. 0.06331 - 0.09019)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-SU*	Precision	12.58	(95%-conf.int. 0.11156 - 0.14122)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-SU*	F_1-score	08.82	(95%-conf.int. 0.07565 - 0.10082)
$\text{BF}_{\texttt{SIM}}$	ROUGE-SU*	Recall	07.38	(95%-conf.int. 0.06420 - 0.08399)
$\text{BF}_{\texttt{SIM}}$	ROUGE-SU*	Precision	11.24	(95%-conf.int. 0.09652 - 0.12832)
$\text{BF}_{\texttt{SIM}}$	ROUGE-SU*	F_1-score	07.99	(95%-conf.int. 0.07030 - 0.08997)
$\text{BF}_{\texttt{DIV}}$	ROUGE-SU*	Recall	05.71	(95%-conf.int. 0.04667 - 0.06797)
$\text{BF}_{\texttt{DIV}}$	ROUGE-SU*	Precision	12.16	(95%-conf.int. 0.10661 - 0.13817)
$\text{BF}_{\texttt{DIV}}$	ROUGE-SU*	F_1-score	07.23	(95%-conf.int. 0.06106 - 0.08400)
$\text{SR}_{\texttt{SIM}}$	ROUGE-SU*	Recall	13.28	(95%-conf.int. 0.11570 - 0.14960)
$\text{SR}_{\texttt{SIM}}$	ROUGE-SU*	Precision	06.15	(95%-conf.int. 0.04961 - 0.07385)
$\text{SR}_{\texttt{SIM}}$	ROUGE-SU*	F_1-score	06.86	(95%-conf.int. 0.05822 - 0.07899)
$\text{SR}_{\texttt{DIV}}$	ROUGE-SU*	Recall	19.96	(95%-conf.int. 0.17974 - 0.21956)
$\text{SR}_{\texttt{DIV}}$	ROUGE-SU*	Precision	01.26	(95%-conf.int. 0.01022 - 0.01540)
$\text{SR}_{\texttt{DIV}}$	ROUGE-SU*	F_1-score	02.26	(95%-conf.int. 0.01870 - 0.02696)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-SU*	Recall	04.52	(95%-conf.int. 0.03772 - 0.05322)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-SU*	Precision	09.61	(95%-conf.int. 0.08323 - 0.10985)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-SU*	F_1-score	05.76	(95%-conf.int. 0.04969 - 0.06616)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-SU*	Recall	03.61	(95%-conf.int. 0.02972 - 0.04291)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-SU*	Precision	02.88	(95%-conf.int. 0.02331 - 0.03502)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-SU*	F_1-score	02.79	(95%-conf.int. 0.02382 - 0.03203)

Figure A.13: ROUGE-SU4 scores on the Opinosis data-set. Frequency of opinion-bearing terms boosted $\times 4$.

ROUGE-1 - Frequency of opinion-bearing terms boosted $\times 5$				
Candidate	Metric	Measure	Score	Confidence interval
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-1	Recall	33.54	(95%-conf.int. 0.30743 - 0.36247)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-1	Precision	26.71	(95%-conf.int. 0.23838 - 0.29409)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-1	F_1-score	27.93	(95%-conf.int. 0.25827 - 0.30096)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-1	Recall	26.65	(95%-conf.int. 0.24112 - 0.29005)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-1	Precision	32.85	(95%-conf.int. 0.30465 - 0.35344)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-1	F_1-score	28.66	(95%-conf.int. 0.26401 - 0.30779)
$\text{BF}_{\texttt{SIM}}$	ROUGE-1	Recall	26.60	(95%-conf.int. 0.24459 - 0.28562)
$\text{BF}_{\texttt{SIM}}$	ROUGE-1	Precision	30.54	(95%-conf.int. 0.28092 - 0.32960)
$\text{BF}_{\texttt{SIM}}$	ROUGE-1	F_1-score	27.52	(95%-conf.int. 0.25498 - 0.29355)
$\text{BF}_{\texttt{DIV}}$	ROUGE-1	Recall	22.11	(95%-conf.int. 0.19715 - 0.24412)
$\text{BF}_{\texttt{DIV}}$	ROUGE-1	Precision	30.93	(95%-conf.int. 0.28658 - 0.33358)
$\text{BF}_{\texttt{DIV}}$	ROUGE-1	F_1-score	25.20	(95%-conf.int. 0.23030 - 0.27314)
$\text{SR}_{\texttt{SIM}}$	ROUGE-1	Recall	34.53	(95%-conf.int. 0.31585 - 0.37425)
$\text{SR}_{\texttt{SIM}}$	ROUGE-1	Precision	21.79	(95%-conf.int. 0.19302 - 0.24334)
$\text{SR}_{\texttt{SIM}}$	ROUGE-1	F_1-score	24.71	(95%-conf.int. 0.22865 - 0.26628)
$\text{SR}_{\texttt{DIV}}$	ROUGE-1	Recall	45.78	(95%-conf.int. 0.43148 - 0.48328)
$\text{SR}_{\texttt{DIV}}$	ROUGE-1	Precision	10.44	(95%-conf.int. 0.09250 - 0.11836)
$\text{SR}_{\texttt{DIV}}$	ROUGE-1	F_1-score	16.25	(95%-conf.int. 0.14933 - 0.17660)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-1	Recall	20.62	(95%-conf.int. 0.18628 - 0.22732)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-1	Precision	27.65	(95%-conf.int. 0.25646 - 0.30059)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-1	F_1-score	23.04	(95%-conf.int. 0.21274 - 0.24933)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-1	Recall	17.91	(95%-conf.int. 0.16390 - 0.19423)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-1	Precision	15.97	(95%-conf.int. 0.14187 - 0.17926)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-1	F_1-score	16.24	(95%-conf.int. 0.14830 - 0.17634)

Figure A.14: ROUGE-1 scores on the Opinosis data-set. Frequency of opinion-bearing terms boosted $\times 5$.

ROUGE-2 - Frequency of opinion-bearing terms boosted $\times 5$				
Candidate	Metric	Measure	Score	Confidence interval
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-2	Recall	09.79	(95%-conf.int. 0.07995 - 0.11865)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-2	Precision	07.79	(95%-conf.int. 0.06351 - 0.09316)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-2	F_1-score	08.09	(95%-conf.int. 0.06690 - 0.09576)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-2	Recall	07.36	(95%-conf.int. 0.05978 - 0.08811)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-2	Precision	08.94	(95%-conf.int. 0.07535 - 0.10409)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-2	F_1-score	07.86	(95%-conf.int. 0.06540 - 0.09321)
$\text{BF}_{\texttt{SIM}}$	ROUGE-2	Recall	07.26	(95%-conf.int. 0.06215 - 0.08363)
$\text{BF}_{\texttt{SIM}}$	ROUGE-2	Precision	08.83	(95%-conf.int. 0.07524 - 0.10307)
$\text{BF}_{\texttt{SIM}}$	ROUGE-2	F_1-score	07.72	(95%-conf.int. 0.06588 - 0.08924)
$\text{BF}_{\texttt{DIV}}$	ROUGE-2	Recall	06.25	(95%-conf.int. 0.04949 - 0.07621)
$\text{BF}_{\texttt{DIV}}$	ROUGE-2	Precision	08.65	(95%-conf.int. 0.07065 - 0.10233)
$\text{BF}_{\texttt{DIV}}$	ROUGE-2	F_1-score	07.08	(95%-conf.int. 0.05723 - 0.08443)
$\text{SR}_{\texttt{SIM}}$	ROUGE-2	Recall	08.93	(95%-conf.int. 0.07429 - 0.10456)
$\text{SR}_{\texttt{SIM}}$	ROUGE-2	Precision	05.62	(95%-conf.int. 0.04567 - 0.06803)
$\text{SR}_{\texttt{SIM}}$	ROUGE-2	F_1-score	06.36	(95%-conf.int. 0.05298 - 0.07546)
$\text{SR}_{\texttt{DIV}}$	ROUGE-2	Recall	08.96	(95%-conf.int. 0.07307 - 0.10731)
$\text{SR}_{\texttt{DIV}}$	ROUGE-2	Precision	01.95	(95%-conf.int. 0.01504 - 0.02451)
$\text{SR}_{\texttt{DIV}}$	ROUGE-2	F_1-score	03.02	(95%-conf.int. 0.02437 - 0.03657)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-2	Recall	04.86	(95%-conf.int. 0.03641 - 0.06077)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-2	Precision	06.39	(95%-conf.int. 0.04946 - 0.07847)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-2	F_1-score	05.35	(95%-conf.int. 0.04101 - 0.06605)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-2	Recall	02.54	(95%-conf.int. 0.01761 - 0.03287)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-2	Precision	02.34	(95%-conf.int. 0.01571 - 0.03275)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-2	F_1-score	02.29	(95%-conf.int. 0.01602 - 0.03014)

Figure A.15: ROUGE-2 scores on the Opinosis data-set. Frequency of opinion-bearing terms boosted $\times 5$.

ROUGE-SU4 - Frequency of opinion-bearing terms boosted $\times 5$				
Candidate	Metric	Measure	Score	Confidence interval
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-SU*	Recall	11.90	(95%-conf.int. 0.10093 - 0.13921)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-SU*	Precision	09.15	(95%-conf.int. 0.07484 - 0.10808)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-SU*	F_1-score	08.44	(95%-conf.int. 0.07279 - 0.09742)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-SU*	Recall	07.69	(95%-conf.int. 0.06348 - 0.08986)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-SU*	Precision	12.61	(95%-conf.int. 0.11082 - 0.14157)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-SU*	F_1-score	08.71	(95%-conf.int. 0.07440 - 0.09966)
$\text{BF}_{\texttt{SIM}}$	ROUGE-SU*	Recall	07.35	(95%-conf.int. 0.06296 - 0.08326)
$\text{BF}_{\texttt{SIM}}$	ROUGE-SU*	Precision	11.09	(95%-conf.int. 0.09602 - 0.12674)
$\text{BF}_{\texttt{SIM}}$	ROUGE-SU*	F_1-score	07.95	(95%-conf.int. 0.06889 - 0.08975)
$\text{BF}_{\texttt{DIV}}$	ROUGE-SU*	Recall	05.80	(95%-conf.int. 0.04691 - 0.06934)
$\text{BF}_{\texttt{DIV}}$	ROUGE-SU*	Precision	12.17	(95%-conf.int. 0.10666 - 0.13817)
$\text{BF}_{\texttt{DIV}}$	ROUGE-SU*	F_1-score	07.29	(95%-conf.int. 0.06142 - 0.08480)
$\text{SR}_{\texttt{SIM}}$	ROUGE-SU*	Recall	12.42	(95%-conf.int. 0.10589 - 0.14354)
$\text{SR}_{\texttt{SIM}}$	ROUGE-SU*	Precision	06.17	(95%-conf.int. 0.04909 - 0.07577)
$\text{SR}_{\texttt{SIM}}$	ROUGE-SU*	F_1-score	06.37	(95%-conf.int. 0.05382 - 0.07358)
$\text{SR}_{\texttt{DIV}}$	ROUGE-SU*	Recall	20.34	(95%-conf.int. 0.18063 - 0.22651)
$\text{SR}_{\texttt{DIV}}$	ROUGE-SU*	Precision	01.42	(95%-conf.int. 0.01052 - 0.01960)
$\text{SR}_{\texttt{DIV}}$	ROUGE-SU*	F_1-score	02.31	(95%-conf.int. 0.01909 - 0.02748)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-SU*	Recall	05.13	(95%-conf.int. 0.04133 - 0.06378)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-SU*	Precision	09.59	(95%-conf.int. 0.08309 - 0.11014)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-SU*	F_1-score	06.06	(95%-conf.int. 0.05169 - 0.07075)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-SU*	Recall	03.68	(95%-conf.int. 0.03073 - 0.04310)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-SU*	Precision	03.56	(95%-conf.int. 0.02800 - 0.04469)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-SU*	F_1-score	03.08	(95%-conf.int. 0.02634 - 0.03528)

Figure A.16: ROUGE-SU4 scores on the Opinosis data-set. Frequency of opinion-bearing terms boosted $\times 5$.

ROUGE-1 - Frequency of opinion-bearing terms boosted $\times 6$				
Candidate	Metric	Measure	Score	Confidence interval
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-1	Recall	32.80	(95%-conf.int. 0.30300 - 0.35389)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-1	Precision	26.07	(95%-conf.int. 0.23364 - 0.28866)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-1	F_1-score	27.29	(95%-conf.int. 0.25160 - 0.29487)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-1	Recall	26.31	(95%-conf.int. 0.23659 - 0.28824)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-1	Precision	32.24	(95%-conf.int. 0.29749 - 0.34868)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-1	F_1-score	28.25	(95%-conf.int. 0.25863 - 0.30582)
$\text{BF}_{\texttt{SIM}}$	ROUGE-1	Recall	26.66	(95%-conf.int. 0.24558 - 0.28765)
$\text{BF}_{\texttt{SIM}}$	ROUGE-1	Precision	30.99	(95%-conf.int. 0.28519 - 0.33582)
$\text{BF}_{\texttt{SIM}}$	ROUGE-1	F_1-score	27.70	(95%-conf.int. 0.25734 - 0.29758)
$\text{BF}_{\texttt{DIV}}$	ROUGE-1	Recall	21.98	(95%-conf.int. 0.19659 - 0.24176)
$\text{BF}_{\texttt{DIV}}$	ROUGE-1	Precision	30.84	(95%-conf.int. 0.28606 - 0.33295)
$\text{BF}_{\texttt{DIV}}$	ROUGE-1	F_1-score	25.10	(95%-conf.int. 0.22967 - 0.27191)
$\text{SR}_{\texttt{SIM}}$	ROUGE-1	Recall	34.83	(95%-conf.int. 0.32235 - 0.37638)
$\text{SR}_{\texttt{SIM}}$	ROUGE-1	Precision	22.53	(95%-conf.int. 0.20222 - 0.24913)
$\text{SR}_{\texttt{SIM}}$	ROUGE-1	F_1-score	25.33	(95%-conf.int. 0.23789 - 0.27021)
$\text{SR}_{\texttt{DIV}}$	ROUGE-1	Recall	44.62	(95%-conf.int. 0.42113 - 0.47044)
$\text{SR}_{\texttt{DIV}}$	ROUGE-1	Precision	10.30	(95%-conf.int. 0.09279 - 0.11385)
$\text{SR}_{\texttt{DIV}}$	ROUGE-1	F_1-score	16.22	(95%-conf.int. 0.14990 - 0.17496)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-1	Recall	20.67	(95%-conf.int. 0.18530 - 0.22931)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-1	Precision	26.86	(95%-conf.int. 0.24873 - 0.29136)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-1	F_1-score	22.78	(95%-conf.int. 0.21001 - 0.24650)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-1	Recall	18.15	(95%-conf.int. 0.16526 - 0.19819)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-1	Precision	16.06	(95%-conf.int. 0.14254 - 0.17991)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-1	F_1-score	16.38	(95%-conf.int. 0.14915 - 0.17839)

Figure A.17: ROUGE-1 scores on the Opinosis data-set. Frequency of opinion-bearing terms boosted $\times 6$.

ROUGE-2 - Frequency of opinion-bearing terms boosted $\times 6$				
Candidate	Metric	Measure	Score	Confidence interval
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-2	Recall	09.00	(95%-conf.int. 0.07421 - 0.10889)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-2	Precision	07.18	(95%-conf.int. 0.05859 - 0.08582)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-2	F_1-score	07.53	(95%-conf.int. 0.06218 - 0.08991)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-2	Recall	07.09	(95%-conf.int. 0.05668 - 0.08628)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-2	Precision	08.58	(95%-conf.int. 0.07101 - 0.10207)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-2	F_1-score	07.57	(95%-conf.int. 0.06148 - 0.09093)
$\text{BF}_{\texttt{SIM}}$	ROUGE-2	Recall	07.61	(95%-conf.int. 0.06533 - 0.08740)
$\text{BF}_{\texttt{SIM}}$	ROUGE-2	Precision	09.29	(95%-conf.int. 0.07966 - 0.10730)
$\text{BF}_{\texttt{SIM}}$	ROUGE-2	F_1-score	08.10	(95%-conf.int. 0.06946 - 0.09328)
$\text{BF}_{\texttt{DIV}}$	ROUGE-2	Recall	06.23	(95%-conf.int. 0.04924 - 0.07591)
$\text{BF}_{\texttt{DIV}}$	ROUGE-2	Precision	08.61	(95%-conf.int. 0.07084 - 0.10128)
$\text{BF}_{\texttt{DIV}}$	ROUGE-2	F_1-score	07.06	(95%-conf.int. 0.05707 - 0.08416)
$\text{SR}_{\texttt{SIM}}$	ROUGE-2	Recall	08.73	(95%-conf.int. 0.07256 - 0.10260)
$\text{SR}_{\texttt{SIM}}$	ROUGE-2	Precision	05.51	(95%-conf.int. 0.04496 - 0.06641)
$\text{SR}_{\texttt{SIM}}$	ROUGE-2	F_1-score	06.25	(95%-conf.int. 0.05174 - 0.07350)
$\text{SR}_{\texttt{DIV}}$	ROUGE-2	Recall	08.21	(95%-conf.int. 0.06730 - 0.09747)
$\text{SR}_{\texttt{DIV}}$	ROUGE-2	Precision	01.79	(95%-conf.int. 0.01414 - 0.02169)
$\text{SR}_{\texttt{DIV}}$	ROUGE-2	F_1-score	02.84	(95%-conf.int. 0.02275 - 0.03411)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-2	Recall	04.89	(95%-conf.int. 0.03605 - 0.06072)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-2	Precision	06.13	(95%-conf.int. 0.04755 - 0.07503)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-2	F_1-score	05.27	(95%-conf.int. 0.04028 - 0.06508)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-2	Recall	02.54	(95%-conf.int. 0.01761 - 0.03287)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-2	Precision	02.34	(95%-conf.int. 0.01571 - 0.03275)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-2	F_1-score	02.29	(95%-conf.int. 0.01601 - 0.03013)

Figure A.18: ROUGE-2 scores on the Opinosis data-set. Frequency of opinion-bearing terms boosted $\times 6$.

ROUGE-SU4 - Frequency of opinion-bearing terms boosted $\times 6$				
Candidate	Metric	Measure	Score	Confidence interval
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-SU*	Recall	11.45	(95%-conf.int. 0.09797 - 0.13354)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-SU*	Precision	08.75	(95%-conf.int. 0.07236 - 0.10342)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-SU*	F_1-score	08.21	(95%-conf.int. 0.07015 - 0.09514)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-SU*	Recall	07.61	(95%-conf.int. 0.06252 - 0.08986)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-SU*	Precision	12.24	(95%-conf.int. 0.10533 - 0.13917)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-SU*	F_1-score	08.59	(95%-conf.int. 0.07237 - 0.09889)
$\text{BF}_{\texttt{SIM}}$	ROUGE-SU*	Recall	07.46	(95%-conf.int. 0.06384 - 0.08521)
$\text{BF}_{\texttt{SIM}}$	ROUGE-SU*	Precision	11.49	(95%-conf.int. 0.09877 - 0.13160)
$\text{BF}_{\texttt{SIM}}$	ROUGE-SU*	F_1-score	08.11	(95%-conf.int. 0.07015 - 0.09267)
$\text{BF}_{\texttt{DIV}}$	ROUGE-SU*	Recall	05.73	(95%-conf.int. 0.04667 - 0.06846)
$\text{BF}_{\texttt{DIV}}$	ROUGE-SU*	Precision	12.11	(95%-conf.int. 0.10632 - 0.13755)
$\text{BF}_{\texttt{DIV}}$	ROUGE-SU*	F_1-score	07.24	(95%-conf.int. 0.06119 - 0.08407)
$\text{SR}_{\texttt{SIM}}$	ROUGE-SU*	Recall	12.08	(95%-conf.int. 0.10445 - 0.13911)
$\text{SR}_{\texttt{SIM}}$	ROUGE-SU*	Precision	06.33	(95%-conf.int. 0.05125 - 0.07624)
$\text{SR}_{\texttt{SIM}}$	ROUGE-SU*	F_1-score	06.44	(95%-conf.int. 0.05659 - 0.07297)
$\text{SR}_{\texttt{DIV}}$	ROUGE-SU*	Recall	19.33	(95%-conf.int. 0.17189 - 0.21513)
$\text{SR}_{\texttt{DIV}}$	ROUGE-SU*	Precision	01.29	(95%-conf.int. 0.01058 - 0.01545)
$\text{SR}_{\texttt{DIV}}$	ROUGE-SU*	F_1-score	02.27	(95%-conf.int. 0.01917 - 0.02641)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-SU*	Recall	05.20	(95%-conf.int. 0.04164 - 0.06438)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-SU*	Precision	09.06	(95%-conf.int. 0.07883 - 0.10444)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-SU*	F_1-score	05.97	(95%-conf.int. 0.05038 - 0.07022)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-SU*	Recall	03.83	(95%-conf.int. 0.03139 - 0.04580)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-SU*	Precision	03.60	(95%-conf.int. 0.02845 - 0.04511)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-SU*	F_1-score	03.15	(95%-conf.int. 0.02692 - 0.03614)

Figure A.19: ROUGE-SU4 scores on the Opinosis data-set. Frequency of opinion-bearing terms boosted $\times 6$.

ROUGE-1 - Frequency of opinion-bearing terms boosted $\times 7$				
Candidate	Metric	Measure	Score	Confidence interval
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-1	Recall	23.23	(95%-conf.int. 0.20440 - 0.25982)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-1	Precision	23.01	(95%-conf.int. 0.20549 - 0.25661)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-1	F_1-score	21.85	(95%-conf.int. 0.19758 - 0.23973)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-1	Recall	16.60	(95%-conf.int. 0.14885 - 0.18476)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-1	Precision	25.45	(95%-conf.int. 0.22951 - 0.28085)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-1	F_1-score	19.66	(95%-conf.int. 0.17764 - 0.21572)
$\text{BF}_{\texttt{SIM}}$	ROUGE-1	Recall	23.05	(95%-conf.int. 0.20623 - 0.25687)
$\text{BF}_{\texttt{SIM}}$	ROUGE-1	Precision	23.69	(95%-conf.int. 0.21567 - 0.26021)
$\text{BF}_{\texttt{SIM}}$	ROUGE-1	F_1-score	22.61	(95%-conf.int. 0.20701 - 0.24690)
$\text{BF}_{\texttt{DIV}}$	ROUGE-1	Recall	16.43	(95%-conf.int. 0.14738 - 0.18135)
$\text{BF}_{\texttt{DIV}}$	ROUGE-1	Precision	24.79	(95%-conf.int. 0.22320 - 0.27345)
$\text{BF}_{\texttt{DIV}}$	ROUGE-1	F_1-score	19.34	(95%-conf.int. 0.17478 - 0.21164)
$\text{SR}_{\texttt{SIM}}$	ROUGE-1	Recall	24.53	(95%-conf.int. 0.22077 - 0.27168)
$\text{SR}_{\texttt{SIM}}$	ROUGE-1	Precision	22.94	(95%-conf.int. 0.20807 - 0.25406)
$\text{SR}_{\texttt{SIM}}$	ROUGE-1	F_1-score	22.89	(95%-conf.int. 0.20891 - 0.25038)
$\text{SR}_{\texttt{DIV}}$	ROUGE-1	Recall	18.97	(95%-conf.int. 0.16883 - 0.21168)
$\text{SR}_{\texttt{DIV}}$	ROUGE-1	Precision	21.06	(95%-conf.int. 0.19045 - 0.23108)
$\text{SR}_{\texttt{DIV}}$	ROUGE-1	F_1-score	19.34	(95%-conf.int. 0.17614 - 0.21196)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-1	Recall	21.58	(95%-conf.int. 0.19371 - 0.24016)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-1	Precision	27.06	(95%-conf.int. 0.25067 - 0.29241)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-1	F_1-score	23.33	(95%-conf.int. 0.21574 - 0.25269)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-1	Recall	17.92	(95%-conf.int. 0.16244 - 0.19561)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-1	Precision	16.08	(95%-conf.int. 0.14303 - 0.18000)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-1	F_1-score	16.24	(95%-conf.int. 0.14901 - 0.17572)

Figure A.20: ROUGE-1 scores on the Opinosis data-set. Frequency of opinion-bearing terms boosted $\times 7$.

ROUGE-2 - Frequency of opinion-bearing terms boosted $\times 7$				
Candidate	Metric	Measure	Score	Confidence interval
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-2	Recall	05.24	(95%-conf.int. 0.03908 - 0.06642)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-2	Precision	05.07	(95%-conf.int. 0.03675 - 0.06639)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-2	F_1-score	04.87	(95%-conf.int. 0.03579 - 0.06212)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-2	Recall	03.39	(95%-conf.int. 0.02509 - 0.04372)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-2	Precision	05.47	(95%-conf.int. 0.04166 - 0.06975)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-2	F_1-score	04.09	(95%-conf.int. 0.03037 - 0.05181)
$\text{BF}_{\texttt{SIM}}$	ROUGE-2	Recall	05.71	(95%-conf.int. 0.04429 - 0.07036)
$\text{BF}_{\texttt{SIM}}$	ROUGE-2	Precision	05.55	(95%-conf.int. 0.04284 - 0.06919)
$\text{BF}_{\texttt{SIM}}$	ROUGE-2	F_1-score	05.42	(95%-conf.int. 0.04215 - 0.06678)
$\text{BF}_{\texttt{DIV}}$	ROUGE-2	Recall	03.17	(95%-conf.int. 0.02356 - 0.04115)
$\text{BF}_{\texttt{DIV}}$	ROUGE-2	Precision	05.16	(95%-conf.int. 0.03865 - 0.06651)
$\text{BF}_{\texttt{DIV}}$	ROUGE-2	F_1-score	03.84	(95%-conf.int. 0.02859 - 0.04918)
$\text{SR}_{\texttt{SIM}}$	ROUGE-2	Recall	05.46	(95%-conf.int. 0.04232 - 0.06752)
$\text{SR}_{\texttt{SIM}}$	ROUGE-2	Precision	04.90	(95%-conf.int. 0.03749 - 0.06188)
$\text{SR}_{\texttt{SIM}}$	ROUGE-2	F_1-score	04.99	(95%-conf.int. 0.03863 - 0.06195)
$\text{SR}_{\texttt{DIV}}$	ROUGE-2	Recall	03.79	(95%-conf.int. 0.02642 - 0.05041)
$\text{SR}_{\texttt{DIV}}$	ROUGE-2	Precision	04.17	(95%-conf.int. 0.02890 - 0.05477)
$\text{SR}_{\texttt{DIV}}$	ROUGE-2	F_1-score	03.82	(95%-conf.int. 0.02658 - 0.05054)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-2	Recall	05.39	(95%-conf.int. 0.04071 - 0.06760)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-2	Precision	06.42	(95%-conf.int. 0.05045 - 0.07831)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-2	F_1-score	05.65	(95%-conf.int. 0.04371 - 0.06871)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-2	Recall	02.47	(95%-conf.int. 0.01777 - 0.03188)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-2	Precision	02.28	(95%-conf.int. 0.01588 - 0.03211)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-2	F_1-score	02.22	(95%-conf.int. 0.01619 - 0.02916)

Figure A.21: ROUGE-2 scores on the Opinosis data-set. Frequency of opinion-bearing terms boosted $\times 7$.

ROUGE-SU4 - Frequency of opinion-bearing terms boosted $\times 7$				
Candidate	Metric	Measure	Score	Confidence interval
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-SU*	Recall	06.54	(95%-conf.int. 0.05140 - 0.08038)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-SU*	Precision	07.08	(95%-conf.int. 0.05699 - 0.08652)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-SU*	F_1-score	05.59	(95%-conf.int. 0.04611 - 0.06735)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-SU*	Recall	03.40	(95%-conf.int. 0.02774 - 0.04121)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-SU*	Precision	08.85	(95%-conf.int. 0.07400 - 0.10519)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-SU*	F_1-score	04.58	(95%-conf.int. 0.03809 - 0.05426)
$\text{BF}_{\texttt{SIM}}$	ROUGE-SU*	Recall	06.33	(95%-conf.int. 0.05162 - 0.07695)
$\text{BF}_{\texttt{SIM}}$	ROUGE-SU*	Precision	07.37	(95%-conf.int. 0.06115 - 0.08713)
$\text{BF}_{\texttt{SIM}}$	ROUGE-SU*	F_1-score	06.04	(95%-conf.int. 0.05123 - 0.07158)
$\text{BF}_{\texttt{DIV}}$	ROUGE-SU*	Recall	03.40	(95%-conf.int. 0.02833 - 0.04023)
$\text{BF}_{\texttt{DIV}}$	ROUGE-SU*	Precision	08.69	(95%-conf.int. 0.07272 - 0.10148)
$\text{BF}_{\texttt{DIV}}$	ROUGE-SU*	F_1-score	04.57	(95%-conf.int. 0.03849 - 0.05341)
$\text{SR}_{\texttt{SIM}}$	ROUGE-SU*	Recall	06.89	(95%-conf.int. 0.05625 - 0.08296)
$\text{SR}_{\texttt{SIM}}$	ROUGE-SU*	Precision	06.86	(95%-conf.int. 0.05692 - 0.08222)
$\text{SR}_{\texttt{SIM}}$	ROUGE-SU*	F_1-score	06.07	(95%-conf.int. 0.05072 - 0.07151)
$\text{SR}_{\texttt{DIV}}$	ROUGE-SU*	Recall	04.58	(95%-conf.int. 0.03635 - 0.05669)
$\text{SR}_{\texttt{DIV}}$	ROUGE-SU*	Precision	06.15	(95%-conf.int. 0.05146 - 0.07262)
$\text{SR}_{\texttt{DIV}}$	ROUGE-SU*	F_1-score	04.65	(95%-conf.int. 0.03835 - 0.05496)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-SU*	Recall	05.67	(95%-conf.int. 0.04520 - 0.07053)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-SU*	Precision	09.17	(95%-conf.int. 0.07957 - 0.10555)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-SU*	F_1-score	06.24	(95%-conf.int. 0.05319 - 0.07264)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-SU*	Recall	03.78	(95%-conf.int. 0.03089 - 0.04519)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-SU*	Precision	03.66	(95%-conf.int. 0.02892 - 0.04549)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-SU*	F_1-score	03.12	(95%-conf.int. 0.02687 - 0.03563)

Figure A.22: ROUGE-SU4 scores on the Opinosis data-set. Frequency of opinion-bearing terms boosted $\times 7$.

ROUGE-1 - Frequency of opinion-bearing terms boosted $\times 8$				
Candidate	Metric	Measure	Score	Confidence interval
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-1	Recall	23.19	(95%-conf.int. 0.20467 - 0.25836)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-1	Precision	23.57	(95%-conf.int. 0.21174 - 0.26016)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-1	F_1-score	22.13	(95%-conf.int. 0.20104 - 0.24165)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-1	Recall	16.80	(95%-conf.int. 0.15057 - 0.18703)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-1	Precision	25.72	(95%-conf.int. 0.23118 - 0.28444)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-1	F_1-score	19.90	(95%-conf.int. 0.17986 - 0.21808)
$\text{BF}_{\texttt{SIM}}$	ROUGE-1	Recall	23.85	(95%-conf.int. 0.21419 - 0.26399)
$\text{BF}_{\texttt{SIM}}$	ROUGE-1	Precision	23.86	(95%-conf.int. 0.21806 - 0.26106)
$\text{BF}_{\texttt{SIM}}$	ROUGE-1	F_1-score	22.99	(95%-conf.int. 0.21206 - 0.24941)
$\text{BF}_{\texttt{DIV}}$	ROUGE-1	Recall	16.57	(95%-conf.int. 0.14813 - 0.18319)
$\text{BF}_{\texttt{DIV}}$	ROUGE-1	Precision	24.90	(95%-conf.int. 0.22403 - 0.27419)
$\text{BF}_{\texttt{DIV}}$	ROUGE-1	F_1-score	19.47	(95%-conf.int. 0.17589 - 0.21292)
$\text{SR}_{\texttt{SIM}}$	ROUGE-1	Recall	24.35	(95%-conf.int. 0.21973 - 0.27020)
$\text{SR}_{\texttt{SIM}}$	ROUGE-1	Precision	22.57	(95%-conf.int. 0.20663 - 0.24837)
$\text{SR}_{\texttt{SIM}}$	ROUGE-1	F_1-score	22.57	(95%-conf.int. 0.20793 - 0.24546)
$\text{SR}_{\texttt{DIV}}$	ROUGE-1	Recall	19.05	(95%-conf.int. 0.16897 - 0.21379)
$\text{SR}_{\texttt{DIV}}$	ROUGE-1	Precision	21.52	(95%-conf.int. 0.19406 - 0.23908)
$\text{SR}_{\texttt{DIV}}$	ROUGE-1	F_1-score	19.62	(95%-conf.int. 0.17683 - 0.21760)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-1	Recall	22.20	(95%-conf.int. 0.20042 - 0.24443)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-1	Precision	26.77	(95%-conf.int. 0.24743 - 0.28894)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-1	F_1-score	23.68	(95%-conf.int. 0.21894 - 0.25732)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-1	Recall	18.44	(95%-conf.int. 0.16852 - 0.20118)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-1	Precision	16.15	(95%-conf.int. 0.14424 - 0.17973)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-1	F_1-score	16.39	(95%-conf.int. 0.15187 - 0.17585)

Figure A.23: ROUGE-1 scores on the Opinosis data-set. Frequency of opinion-bearing terms boosted $\times 8$.

ROUGE-2 - Frequency of opinion-bearing terms boosted $\times 8$				
Candidate	Metric	Measure	Score	Confidence interval
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-2	Recall	05.35	(95%-conf.int. 0.04045 - 0.06701)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-2	Precision	05.35	(95%-conf.int. 0.03972 - 0.06932)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-2	F_1-score	05.06	(95%-conf.int. 0.03794 - 0.06438)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-2	Recall	03.41	(95%-conf.int. 0.02531 - 0.04372)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-2	Precision	05.50	(95%-conf.int. 0.04219 - 0.06978)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-2	F_1-score	04.11	(95%-conf.int. 0.03073 - 0.05200)
$\text{BF}_{\texttt{SIM}}$	ROUGE-2	Recall	05.85	(95%-conf.int. 0.04563 - 0.07192)
$\text{BF}_{\texttt{SIM}}$	ROUGE-2	Precision	05.56	(95%-conf.int. 0.04282 - 0.06909)
$\text{BF}_{\texttt{SIM}}$	ROUGE-2	F_1-score	05.47	(95%-conf.int. 0.04253 - 0.06708)
$\text{BF}_{\texttt{DIV}}$	ROUGE-2	Recall	03.17	(95%-conf.int. 0.02356 - 0.04115)
$\text{BF}_{\texttt{DIV}}$	ROUGE-2	Precision	05.16	(95%-conf.int. 0.03865 - 0.06651)
$\text{BF}_{\texttt{DIV}}$	ROUGE-2	F_1-score	03.84	(95%-conf.int. 0.02859 - 0.04918)
$\text{SR}_{\texttt{SIM}}$	ROUGE-2	Recall	05.37	(95%-conf.int. 0.04096 - 0.06739)
$\text{SR}_{\texttt{SIM}}$	ROUGE-2	Precision	04.75	(95%-conf.int. 0.03554 - 0.06025)
$\text{SR}_{\texttt{SIM}}$	ROUGE-2	F_1-score	04.85	(95%-conf.int. 0.03699 - 0.06031)
$\text{SR}_{\texttt{DIV}}$	ROUGE-2	Recall	03.84	(95%-conf.int. 0.02681 - 0.05108)
$\text{SR}_{\texttt{DIV}}$	ROUGE-2	Precision	04.33	(95%-conf.int. 0.02963 - 0.05788)
$\text{SR}_{\texttt{DIV}}$	ROUGE-2	F_1-score	03.94	(95%-conf.int. 0.02735 - 0.05236)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-2	Recall	05.71	(95%-conf.int. 0.04387 - 0.07037)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-2	Precision	06.63	(95%-conf.int. 0.05313 - 0.08035)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-2	F_1-score	05.94	(95%-conf.int. 0.04687 - 0.07196)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-2	Recall	02.64	(95%-conf.int. 0.01874 - 0.03444)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-2	Precision	02.30	(95%-conf.int. 0.01618 - 0.03233)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-2	F_1-score	02.27	(95%-conf.int. 0.01667 - 0.02975)

Figure A.24: ROUGE-2 scores on the Opinosis data-set. Frequency of opinion-bearing terms boosted $\times 8$.

ROUGE-SU4 - Frequency of opinion-bearing terms boosted $\times 8$				
Candidate	Metric	Measure	Score	Confidence interval
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-SU*	Recall	06.46	(95%-conf.int. 0.05126 - 0.07960)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-SU*	Precision	07.37	(95%-conf.int. 0.06026 - 0.08926)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-SU*	F_1-score	05.72	(95%-conf.int. 0.04710 - 0.06850)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-SU*	Recall	03.46	(95%-conf.int. 0.02844 - 0.04140)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-SU*	Precision	08.99	(95%-conf.int. 0.07524 - 0.10610)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-SU*	F_1-score	04.66	(95%-conf.int. 0.03865 - 0.05458)
$\text{BF}_{\texttt{SIM}}$	ROUGE-SU*	Recall	06.69	(95%-conf.int. 0.05492 - 0.08026)
$\text{BF}_{\texttt{SIM}}$	ROUGE-SU*	Precision	07.46	(95%-conf.int. 0.06249 - 0.08796)
$\text{BF}_{\texttt{SIM}}$	ROUGE-SU*	F_1-score	06.18	(95%-conf.int. 0.05263 - 0.07238)
$\text{BF}_{\texttt{DIV}}$	ROUGE-SU*	Recall	03.47	(95%-conf.int. 0.02873 - 0.04097)
$\text{BF}_{\texttt{DIV}}$	ROUGE-SU*	Precision	08.78	(95%-conf.int. 0.07403 - 0.10204)
$\text{BF}_{\texttt{DIV}}$	ROUGE-SU*	F_1-score	04.65	(95%-conf.int. 0.03901 - 0.05419)
$\text{SR}_{\texttt{SIM}}$	ROUGE-SU*	Recall	06.91	(95%-conf.int. 0.05543 - 0.08394)
$\text{SR}_{\texttt{SIM}}$	ROUGE-SU*	Precision	06.61	(95%-conf.int. 0.05569 - 0.07887)
$\text{SR}_{\texttt{SIM}}$	ROUGE-SU*	F_1-score	05.88	(95%-conf.int. 0.04921 - 0.06933)
$\text{SR}_{\texttt{DIV}}$	ROUGE-SU*	Recall	04.64	(95%-conf.int. 0.03652 - 0.05752)
$\text{SR}_{\texttt{DIV}}$	ROUGE-SU*	Precision	06.48	(95%-conf.int. 0.05367 - 0.07878)
$\text{SR}_{\texttt{DIV}}$	ROUGE-SU*	F_1-score	04.85	(95%-conf.int. 0.03955 - 0.05918)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-SU*	Recall	05.88	(95%-conf.int. 0.04717 - 0.07328)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-SU*	Precision	09.02	(95%-conf.int. 0.07716 - 0.10383)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-SU*	F_1-score	06.42	(95%-conf.int. 0.05466 - 0.07453)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-SU*	Recall	04.04	(95%-conf.int. 0.03308 - 0.04853)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-SU*	Precision	03.65	(95%-conf.int. 0.02900 - 0.04556)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-SU*	F_1-score	03.13	(95%-conf.int. 0.02709 - 0.03539)

Figure A.25: ROUGE-SU4 scores on the Opinosis data-set. Frequency of opinion-bearing terms boosted $\times 8$.

ROUGE-1 - Frequency of opinion-bearing terms boosted $\times 9$				
Candidate	Metric	Measure	Score	Confidence interval
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-1	Recall	23.19	(95%-conf.int. 0.20467 - 0.25836)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-1	Precision	23.57	(95%-conf.int. 0.21174 - 0.26016)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-1	F_1-score	22.13	(95%-conf.int. 0.20104 - 0.24165)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-1	Recall	16.80	(95%-conf.int. 0.15057 - 0.18703)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-1	Precision	25.72	(95%-conf.int. 0.23118 - 0.28444)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-1	F_1-score	19.90	(95%-conf.int. 0.17986 - 0.21808)
$\text{BF}_{\texttt{SIM}}$	ROUGE-1	Recall	24.04	(95%-conf.int. 0.21649 - 0.26507)
$\text{BF}_{\texttt{SIM}}$	ROUGE-1	Precision	23.72	(95%-conf.int. 0.21707 - 0.25999)
$\text{BF}_{\texttt{SIM}}$	ROUGE-1	F_1-score	23.02	(95%-conf.int. 0.21224 - 0.25012)
$\text{BF}_{\texttt{DIV}}$	ROUGE-1	Recall	16.57	(95%-conf.int. 0.14813 - 0.18319)
$\text{BF}_{\texttt{DIV}}$	ROUGE-1	Precision	24.90	(95%-conf.int. 0.22403 - 0.27419)
$\text{BF}_{\texttt{DIV}}$	ROUGE-1	F_1-score	19.47	(95%-conf.int. 0.17589 - 0.21292)
$\text{SR}_{\texttt{SIM}}$	ROUGE-1	Recall	24.06	(95%-conf.int. 0.21526 - 0.26720)
$\text{SR}_{\texttt{SIM}}$	ROUGE-1	Precision	22.82	(95%-conf.int. 0.20853 - 0.25037)
$\text{SR}_{\texttt{SIM}}$	ROUGE-1	F_1-score	22.55	(95%-conf.int. 0.20729 - 0.24685)
$\text{SR}_{\texttt{DIV}}$	ROUGE-1	Recall	18.97	(95%-conf.int. 0.16835 - 0.21246)
$\text{SR}_{\texttt{DIV}}$	ROUGE-1	Precision	20.91	(95%-conf.int. 0.18972 - 0.22932)
$\text{SR}_{\texttt{DIV}}$	ROUGE-1	F_1-score	19.25	(95%-conf.int. 0.17451 - 0.21126)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-1	Recall	22.31	(95%-conf.int. 0.20193 - 0.24558)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-1	Precision	26.83	(95%-conf.int. 0.24734 - 0.28970)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-1	F_1-score	23.81	(95%-conf.int. 0.22001 - 0.25879)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-1	Recall	18.52	(95%-conf.int. 0.16969 - 0.20362)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-1	Precision	16.10	(95%-conf.int. 0.14414 - 0.17895)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-1	F_1-score	16.39	(95%-conf.int. 0.15135 - 0.17619)

Figure A.26: ROUGE-1 scores on the Opinosis data-set. Frequency of opinion-bearing terms boosted $\times 9$.

ROUGE-2 - Frequency of opinion-bearing terms boosted $\times 9$				
Candidate	Metric	Measure	Score	Confidence interval
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-2	Recall	05.35	(95%-conf.int. 0.04045 - 0.06701)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-2	Precision	05.35	(95%-conf.int. 0.03972 - 0.06932)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-2	F_1-score	05.06	(95%-conf.int. 0.03794 - 0.06438)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-2	Recall	03.41	(95%-conf.int. 0.02531 - 0.04372)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-2	Precision	05.50	(95%-conf.int. 0.04219 - 0.06978)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-2	F_1-score	04.11	(95%-conf.int. 0.03073 - 0.05200)
$\text{BF}_{\texttt{SIM}}$	ROUGE-2	Recall	05.83	(95%-conf.int. 0.04543 - 0.07187)
$\text{BF}_{\texttt{SIM}}$	ROUGE-2	Precision	05.44	(95%-conf.int. 0.04184 - 0.06755)
$\text{BF}_{\texttt{SIM}}$	ROUGE-2	F_1-score	05.40	(95%-conf.int. 0.04179 - 0.06632)
$\text{BF}_{\texttt{DIV}}$	ROUGE-2	Recall	03.17	(95%-conf.int. 0.02356 - 0.04115)
$\text{BF}_{\texttt{DIV}}$	ROUGE-2	Precision	05.16	(95%-conf.int. 0.03865 - 0.06651)
$\text{BF}_{\texttt{DIV}}$	ROUGE-2	F_1-score	03.84	(95%-conf.int. 0.02859 - 0.04918)
$\text{SR}_{\texttt{SIM}}$	ROUGE-2	Recall	05.47	(95%-conf.int. 0.04259 - 0.06833)
$\text{SR}_{\texttt{SIM}}$	ROUGE-2	Precision	05.00	(95%-conf.int. 0.03832 - 0.06221)
$\text{SR}_{\texttt{SIM}}$	ROUGE-2	F_1-score	05.02	(95%-conf.int. 0.03889 - 0.06172)
$\text{SR}_{\texttt{DIV}}$	ROUGE-2	Recall	03.68	(95%-conf.int. 0.02549 - 0.04880)
$\text{SR}_{\texttt{DIV}}$	ROUGE-2	Precision	04.10	(95%-conf.int. 0.02840 - 0.05400)
$\text{SR}_{\texttt{DIV}}$	ROUGE-2	F_1-score	03.74	(95%-conf.int. 0.02593 - 0.04948)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-2	Recall	05.65	(95%-conf.int. 0.04346 - 0.06953)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-2	Precision	06.55	(95%-conf.int. 0.05171 - 0.08004)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-2	F_1-score	05.88	(95%-conf.int. 0.04628 - 0.07173)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-2	Recall	02.40	(95%-conf.int. 0.01636 - 0.03214)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-2	Precision	02.17	(95%-conf.int. 0.01470 - 0.03093)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-2	F_1-score	02.11	(95%-conf.int. 0.01503 - 0.02823)

Figure A.27: ROUGE-2 scores on the Opinosis data-set. Frequency of opinion-bearing terms boosted $\times 9$.

ROUGE-SU4 - Frequency of opinion-bearing terms boosted $\times 9$				
Candidate	Metric	Measure	Score	Confidence interval
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-SU*	Recall	06.46	(95%-conf.int. 0.05126 - 0.07960)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-SU*	Precision	07.37	(95%-conf.int. 0.06026 - 0.08926)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-SU*	F_1-score	05.72	(95%-conf.int. 0.04710 - 0.06850)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-SU*	Recall	03.46	(95%-conf.int. 0.02844 - 0.04140)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-SU*	Precision	08.99	(95%-conf.int. 0.07524 - 0.10610)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-SU*	F_1-score	04.66	(95%-conf.int. 0.03865 - 0.05458)
$\text{BF}_{\texttt{SIM}}$	ROUGE-SU*	Recall	06.71	(95%-conf.int. 0.05551 - 0.08047)
$\text{BF}_{\texttt{SIM}}$	ROUGE-SU*	Precision	07.26	(95%-conf.int. 0.06080 - 0.08537)
$\text{BF}_{\texttt{SIM}}$	ROUGE-SU*	F_1-score	06.11	(95%-conf.int. 0.05203 - 0.07120)
$\text{BF}_{\texttt{DIV}}$	ROUGE-SU*	Recall	03.47	(95%-conf.int. 0.02873 - 0.04097)
$\text{BF}_{\texttt{DIV}}$	ROUGE-SU*	Precision	08.78	(95%-conf.int. 0.07403 - 0.10204)
$\text{BF}_{\texttt{DIV}}$	ROUGE-SU*	F_1-score	04.65	(95%-conf.int. 0.03901 - 0.05419)
$\text{SR}_{\texttt{SIM}}$	ROUGE-SU*	Recall	06.74	(95%-conf.int. 0.05399 - 0.08182)
$\text{SR}_{\texttt{SIM}}$	ROUGE-SU*	Precision	06.75	(95%-conf.int. 0.05659 - 0.08016)
$\text{SR}_{\texttt{SIM}}$	ROUGE-SU*	F_1-score	05.85	(95%-conf.int. 0.04924 - 0.06880)
$\text{SR}_{\texttt{DIV}}$	ROUGE-SU*	Recall	04.58	(95%-conf.int. 0.03609 - 0.05700)
$\text{SR}_{\texttt{DIV}}$	ROUGE-SU*	Precision	06.04	(95%-conf.int. 0.05071 - 0.07122)
$\text{SR}_{\texttt{DIV}}$	ROUGE-SU*	F_1-score	04.61	(95%-conf.int. 0.03778 - 0.05494)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-SU*	Recall	05.85	(95%-conf.int. 0.04720 - 0.07273)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-SU*	Precision	09.03	(95%-conf.int. 0.07769 - 0.10408)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-SU*	F_1-score	06.45	(95%-conf.int. 0.05523 - 0.07491)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-SU*	Recall	04.03	(95%-conf.int. 0.03339 - 0.04854)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-SU*	Precision	03.63	(95%-conf.int. 0.02886 - 0.04499)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-SU*	F_1-score	03.13	(95%-conf.int. 0.02712 - 0.03543)

Figure A.28: ROUGE-SU4 scores on the Opinosis data-set. Frequency of opinion-bearing terms boosted $\times 9$.

ROUGE-1 - Frequency of opinion-bearing terms boosted $\times 10$				
Candidate	Metric	Measure	Score	Confidence interval
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-1	Recall	23.30	(95%-conf.int. 0.20485 - 0.25953)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-1	Precision	23.28	(95%-conf.int. 0.20877 - 0.25804)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-1	F_1-score	22.03	(95%-conf.int. 0.19950 - 0.24161)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-1	Recall	16.89	(95%-conf.int. 0.15170 - 0.18738)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-1	Precision	25.92	(95%-conf.int. 0.23298 - 0.28667)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-1	F_1-score	20.03	(95%-conf.int. 0.18086 - 0.22009)
$\text{BF}_{\texttt{SIM}}$	ROUGE-1	Recall	24.52	(95%-conf.int. 0.22098 - 0.27034)
$\text{BF}_{\texttt{SIM}}$	ROUGE-1	Precision	23.79	(95%-conf.int. 0.21632 - 0.26130)
$\text{BF}_{\texttt{SIM}}$	ROUGE-1	F_1-score	23.26	(95%-conf.int. 0.21290 - 0.25230)
$\text{BF}_{\texttt{DIV}}$	ROUGE-1	Recall	16.57	(95%-conf.int. 0.14813 - 0.18319)
$\text{BF}_{\texttt{DIV}}$	ROUGE-1	Precision	24.90	(95%-conf.int. 0.22403 - 0.27419)
$\text{BF}_{\texttt{DIV}}$	ROUGE-1	F_1-score	19.47	(95%-conf.int. 0.17589 - 0.21292)
$\text{SR}_{\texttt{SIM}}$	ROUGE-1	Recall	24.19	(95%-conf.int. 0.21722 - 0.26835)
$\text{SR}_{\texttt{SIM}}$	ROUGE-1	Precision	22.98	(95%-conf.int. 0.21009 - 0.25178)
$\text{SR}_{\texttt{SIM}}$	ROUGE-1	F_1-score	22.71	(95%-conf.int. 0.20909 - 0.24836)
$\text{SR}_{\texttt{DIV}}$	ROUGE-1	Recall	19.14	(95%-conf.int. 0.16994 - 0.21450)
$\text{SR}_{\texttt{DIV}}$	ROUGE-1	Precision	21.17	(95%-conf.int. 0.19137 - 0.23221)
$\text{SR}_{\texttt{DIV}}$	ROUGE-1	F_1-score	19.44	(95%-conf.int. 0.17629 - 0.21360)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-1	Recall	22.46	(95%-conf.int. 0.20297 - 0.24847)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-1	Precision	27.00	(95%-conf.int. 0.24829 - 0.29157)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-1	F_1-score	23.97	(95%-conf.int. 0.22086 - 0.26062)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-1	Recall	19.21	(95%-conf.int. 0.17478 - 0.21141)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-1	Precision	16.69	(95%-conf.int. 0.15094 - 0.18471)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-1	F_1-score	17.01	(95%-conf.int. 0.15767 - 0.18377)

Figure A.29: ROUGE-1 scores on the Opinosis data-set. Frequency of opinion-bearing terms boosted $\times 10$.

ROUGE-2 - Frequency of opinion-bearing terms boosted $\times 10$				
Candidate	Metric	Measure	Score	Confidence interval
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-2	Recall	05.37	(95%-conf.int. 0.04063 - 0.06715)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-2	Precision	05.33	(95%-conf.int. 0.03964 - 0.06909)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-2	F_1-score	05.06	(95%-conf.int. 0.03793 - 0.06437)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-2	Recall	03.41	(95%-conf.int. 0.02531 - 0.04372)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-2	Precision	05.51	(95%-conf.int. 0.04239 - 0.06985)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-2	F_1-score	04.11	(95%-conf.int. 0.03073 - 0.05200)
$\text{BF}_{\texttt{SIM}}$	ROUGE-2	Recall	06.06	(95%-conf.int. 0.04745 - 0.07386)
$\text{BF}_{\texttt{SIM}}$	ROUGE-2	Precision	05.57	(95%-conf.int. 0.04259 - 0.06963)
$\text{BF}_{\texttt{SIM}}$	ROUGE-2	F_1-score	05.58	(95%-conf.int. 0.04362 - 0.06828)
$\text{BF}_{\texttt{DIV}}$	ROUGE-2	Recall	03.27	(95%-conf.int. 0.02405 - 0.04265)
$\text{BF}_{\texttt{DIV}}$	ROUGE-2	Precision	05.29	(95%-conf.int. 0.03962 - 0.06806)
$\text{BF}_{\texttt{DIV}}$	ROUGE-2	F_1-score	03.95	(95%-conf.int. 0.02891 - 0.05098)
$\text{SR}_{\texttt{SIM}}$	ROUGE-2	Recall	05.60	(95%-conf.int. 0.04359 - 0.06967)
$\text{SR}_{\texttt{SIM}}$	ROUGE-2	Precision	05.00	(95%-conf.int. 0.03853 - 0.06181)
$\text{SR}_{\texttt{SIM}}$	ROUGE-2	F_1-score	05.08	(95%-conf.int. 0.03957 - 0.06239)
$\text{SR}_{\texttt{DIV}}$	ROUGE-2	Recall	03.68	(95%-conf.int. 0.02549 - 0.04880)
$\text{SR}_{\texttt{DIV}}$	ROUGE-2	Precision	04.13	(95%-conf.int. 0.02853 - 0.05449)
$\text{SR}_{\texttt{DIV}}$	ROUGE-2	F_1-score	03.74	(95%-conf.int. 0.02603 - 0.04969)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-2	Recall	05.74	(95%-conf.int. 0.04415 - 0.07119)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-2	Precision	06.65	(95%-conf.int. 0.05284 - 0.08130)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-2	F_1-score	05.98	(95%-conf.int. 0.04709 - 0.07337)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-2	Recall	02.52	(95%-conf.int. 0.01736 - 0.03429)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-2	Precision	02.29	(95%-conf.int. 0.01559 - 0.03202)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-2	F_1-score	02.23	(95%-conf.int. 0.01576 - 0.02937)

Figure A.30: ROUGE-2 scores on the Opinosis data-set. Frequency of opinion-bearing terms boosted $\times 10$.

ROUGE-SU4 - Frequency of opinion-bearing terms boosted $\times 10$				
Candidate	Metric	Measure	Score	Confidence interval
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-SU*	Recall	06.55	(95%-conf.int. 0.05144 - 0.08033)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-SU*	Precision	07.25	(95%-conf.int. 0.05856 - 0.08791)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-SU*	F_1-score	05.69	(95%-conf.int. 0.04697 - 0.06822)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-SU*	Recall	03.48	(95%-conf.int. 0.02858 - 0.04152)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-SU*	Precision	09.09	(95%-conf.int. 0.07649 - 0.10682)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-SU*	F_1-score	04.70	(95%-conf.int. 0.03904 - 0.05495)
$\text{BF}_{\texttt{SIM}}$	ROUGE-SU*	Recall	06.98	(95%-conf.int. 0.05774 - 0.08283)
$\text{BF}_{\texttt{SIM}}$	ROUGE-SU*	Precision	07.38	(95%-conf.int. 0.06129 - 0.08764)
$\text{BF}_{\texttt{SIM}}$	ROUGE-SU*	F_1-score	06.28	(95%-conf.int. 0.05349 - 0.07330)
$\text{BF}_{\texttt{DIV}}$	ROUGE-SU*	Recall	03.48	(95%-conf.int. 0.02880 - 0.04124)
$\text{BF}_{\texttt{DIV}}$	ROUGE-SU*	Precision	08.81	(95%-conf.int. 0.07432 - 0.10204)
$\text{BF}_{\texttt{DIV}}$	ROUGE-SU*	F_1-score	04.67	(95%-conf.int. 0.03924 - 0.05446)
$\text{SR}_{\texttt{SIM}}$	ROUGE-SU*	Recall	06.83	(95%-conf.int. 0.05477 - 0.08352)
$\text{SR}_{\texttt{SIM}}$	ROUGE-SU*	Precision	06.82	(95%-conf.int. 0.05751 - 0.08064)
$\text{SR}_{\texttt{SIM}}$	ROUGE-SU*	F_1-score	05.94	(95%-conf.int. 0.05015 - 0.06966)
$\text{SR}_{\texttt{DIV}}$	ROUGE-SU*	Recall	04.62	(95%-conf.int. 0.03656 - 0.05733)
$\text{SR}_{\texttt{DIV}}$	ROUGE-SU*	Precision	06.17	(95%-conf.int. 0.05132 - 0.07266)
$\text{SR}_{\texttt{DIV}}$	ROUGE-SU*	F_1-score	04.66	(95%-conf.int. 0.03841 - 0.05557)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-SU*	Recall	05.93	(95%-conf.int. 0.04781 - 0.07373)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-SU*	Precision	09.14	(95%-conf.int. 0.07829 - 0.10550)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-SU*	F_1-score	06.54	(95%-conf.int. 0.05596 - 0.07677)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-SU*	Recall	04.33	(95%-conf.int. 0.03566 - 0.05278)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-SU*	Precision	03.83	(95%-conf.int. 0.03108 - 0.04695)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-SU*	F_1-score	03.35	(95%-conf.int. 0.02922 - 0.03796)

Figure A.31: ROUGE-SU4 scores on the Opinosis data-set. Frequency of opinion-bearing terms boosted $\times 10$.

A.3 Opinion-based bigrams

ROUGE-1 - Unigrams and opinion-based bigrams				
Candidate	Metric	Measure	Score	Confidence interval
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-1	Recall	33.29	(95%-conf.int. 0.30916 - 0.35739)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-1	Precision	27.16	(95%-conf.int. 0.24165 - 0.30262)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-1	F_1-score	28.10	(95%-conf.int. 0.25893 - 0.30303)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-1	Recall	26.56	(95%-conf.int. 0.24428 - 0.28976)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-1	Precision	30.08	(95%-conf.int. 0.27387 - 0.32705)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-1	F_1-score	26.76	(95%-conf.int. 0.25064 - 0.28684)
$\text{BF}_{\texttt{SIM}}$	ROUGE-1	Recall	26.12	(95%-conf.int. 0.24130 - 0.28193)
$\text{BF}_{\texttt{SIM}}$	ROUGE-1	Precision	29.02	(95%-conf.int. 0.26160 - 0.31863)
$\text{BF}_{\texttt{SIM}}$	ROUGE-1	F_1-score	25.90	(95%-conf.int. 0.24195 - 0.27597)
$\text{BF}_{\texttt{DIV}}$	ROUGE-1	Recall	22.31	(95%-conf.int. 0.19887 - 0.24761)
$\text{BF}_{\texttt{DIV}}$	ROUGE-1	Precision	28.27	(95%-conf.int. 0.25801 - 0.30490)
$\text{BF}_{\texttt{DIV}}$	ROUGE-1	F_1-score	23.53	(95%-conf.int. 0.21775 - 0.25311)
$\text{SR}_{\texttt{SIM}}$	ROUGE-1	Recall	38.29	(95%-conf.int. 0.35954 - 0.40744)
$\text{SR}_{\texttt{SIM}}$	ROUGE-1	Precision	18.35	(95%-conf.int. 0.16398 - 0.20525)
$\text{SR}_{\texttt{SIM}}$	ROUGE-1	F_1-score	23.53	(95%-conf.int. 0.21910 - 0.25315)
$\text{SR}_{\texttt{DIV}}$	ROUGE-1	Recall	45.88	(95%-conf.int. 0.43902 - 0.47794)
$\text{SR}_{\texttt{DIV}}$	ROUGE-1	Precision	10.17	(95%-conf.int. 0.09254 - 0.11175)
$\text{SR}_{\texttt{DIV}}$	ROUGE-1	F_1-score	16.23	(95%-conf.int. 0.15140 - 0.17395)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-1	Recall	19.59	(95%-conf.int. 0.17282 - 0.22206)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-1	Precision	26.70	(95%-conf.int. 0.23882 - 0.29477)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-1	F_1-score	21.22	(95%-conf.int. 0.19318 - 0.23122)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-1	Recall	19.80	(95%-conf.int. 0.17444 - 0.22301)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-1	Precision	14.91	(95%-conf.int. 0.13478 - 0.16451)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-1	F_1-score	16.08	(95%-conf.int. 0.14674 - 0.17463)

Figure A.32: ROUGE-1 scores on the Opinosis data-set. Unigrams and opinion-based bigrams.

ROUGE-2 - Unigrams and opinion-based bigrams				
Candidate	Metric	Measure	Score	Confidence interval
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-2	Recall	09.35	(95%-conf.int. 0.07966 - 0.10806)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-2	Precision	07.90	(95%-conf.int. 0.06412 - 0.09620)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-2	F_1-score	08.04	(95%-conf.int. 0.06741 - 0.09412)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-2	Recall	07.18	(95%-conf.int. 0.05835 - 0.08708)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-2	Precision	08.52	(95%-conf.int. 0.06960 - 0.10305)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-2	F_1-score	07.46	(95%-conf.int. 0.06090 - 0.08936)
$\text{BF}_{\texttt{SIM}}$	ROUGE-2	Recall	07.31	(95%-conf.int. 0.06179 - 0.08466)
$\text{BF}_{\texttt{SIM}}$	ROUGE-2	Precision	08.66	(95%-conf.int. 0.07215 - 0.10289)
$\text{BF}_{\texttt{SIM}}$	ROUGE-2	F_1-score	07.51	(95%-conf.int. 0.06328 - 0.08720)
$\text{BF}_{\texttt{DIV}}$	ROUGE-2	Recall	06.22	(95%-conf.int. 0.05030 - 0.07479)
$\text{BF}_{\texttt{DIV}}$	ROUGE-2	Precision	08.07	(95%-conf.int. 0.06430 - 0.09633)
$\text{BF}_{\texttt{DIV}}$	ROUGE-2	F_1-score	06.69	(95%-conf.int. 0.05378 - 0.08009)
$\text{SR}_{\texttt{SIM}}$	ROUGE-2	Recall	09.80	(95%-conf.int. 0.08269 - 0.11302)
$\text{SR}_{\texttt{SIM}}$	ROUGE-2	Precision	04.80	(95%-conf.int. 0.03845 - 0.05869)
$\text{SR}_{\texttt{SIM}}$	ROUGE-2	F_1-score	06.07	(95%-conf.int. 0.05097 - 0.07208)
$\text{SR}_{\texttt{DIV}}$	ROUGE-2	Recall	08.88	(95%-conf.int. 0.07430 - 0.10267)
$\text{SR}_{\texttt{DIV}}$	ROUGE-2	Precision	01.95	(95%-conf.int. 0.01585 - 0.02348)
$\text{SR}_{\texttt{DIV}}$	ROUGE-2	F_1-score	03.10	(95%-conf.int. 0.02584 - 0.03629)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-2	Recall	04.00	(95%-conf.int. 0.03041 - 0.04968)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-2	Precision	05.62	(95%-conf.int. 0.04339 - 0.07125)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-2	F_1-score	04.40	(95%-conf.int. 0.03407 - 0.05434)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-2	Recall	02.45	(95%-conf.int. 0.01659 - 0.03326)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-2	Precision	01.80	(95%-conf.int. 0.01192 - 0.02474)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-2	F_1-score	01.97	(95%-conf.int. 0.01321 - 0.02629)

Figure A.33: ROUGE-2 scores on the Opinosis data-set. Unigrams and opinion-based bigrams.

ROUGE-SU4 - Unigrams and opinion-based bigrams				
Candidate	Metric	Measure	Score	Confidence interval
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-SU*	Recall	11.78	(95%-conf.int. 0.10304 - 0.13374)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-SU*	Precision	09.50	(95%-conf.int. 0.07691 - 0.11474)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-SU*	F_1-score	08.59	(95%-conf.int. 0.07335 - 0.09870)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-SU*	Recall	07.67	(95%-conf.int. 0.06494 - 0.09086)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-SU*	Precision	11.08	(95%-conf.int. 0.09399 - 0.12841)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-SU*	F_1-score	07.70	(95%-conf.int. 0.06737 - 0.08822)
$\text{BF}_{\texttt{SIM}}$	ROUGE-SU*	Recall	07.35	(95%-conf.int. 0.06290 - 0.08441)
$\text{BF}_{\texttt{SIM}}$	ROUGE-SU*	Precision	10.47	(95%-conf.int. 0.08707 - 0.12238)
$\text{BF}_{\texttt{SIM}}$	ROUGE-SU*	F_1-score	07.14	(95%-conf.int. 0.06325 - 0.08031)
$\text{BF}_{\texttt{DIV}}$	ROUGE-SU*	Recall	05.89	(95%-conf.int. 0.04749 - 0.07142)
$\text{BF}_{\texttt{DIV}}$	ROUGE-SU*	Precision	10.52	(95%-conf.int. 0.09003 - 0.12051)
$\text{BF}_{\texttt{DIV}}$	ROUGE-SU*	F_1-score	06.32	(95%-conf.int. 0.05443 - 0.07273)
$\text{SR}_{\texttt{SIM}}$	ROUGE-SU*	Recall	14.74	(95%-conf.int. 0.13129 - 0.16469)
$\text{SR}_{\texttt{SIM}}$	ROUGE-SU*	Precision	04.31	(95%-conf.int. 0.03422 - 0.05321)
$\text{SR}_{\texttt{SIM}}$	ROUGE-SU*	F_1-score	05.61	(95%-conf.int. 0.04794 - 0.06504)
$\text{SR}_{\texttt{DIV}}$	ROUGE-SU*	Recall	19.84	(95%-conf.int. 0.17979 - 0.21573)
$\text{SR}_{\texttt{DIV}}$	ROUGE-SU*	Precision	01.28	(95%-conf.int. 0.01023 - 0.01593)
$\text{SR}_{\texttt{DIV}}$	ROUGE-SU*	F_1-score	02.26	(95%-conf.int. 0.01885 - 0.02674)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-SU*	Recall	04.50	(95%-conf.int. 0.03506 - 0.05618)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-SU*	Precision	09.11	(95%-conf.int. 0.07559 - 0.10755)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-SU*	F_1-score	04.89	(95%-conf.int. 0.04164 - 0.05647)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-SU*	Recall	04.74	(95%-conf.int. 0.03733 - 0.05866)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-SU*	Precision	02.99	(95%-conf.int. 0.02492 - 0.03494)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-SU*	F_1-score	02.97	(95%-conf.int. 0.02577 - 0.03367)

Figure A.34: ROUGE-SU4 scores on the Opinosis data-set. Unigrams and opinion-based bigrams.

A.4 Negation removed

ROUGE-1 - Negation removal, window size=1				
Candidate	Metric	Measure	Score	Confidence interval
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-1	Recall	17.86	(95%-conf.int. 0.15921 - 0.19689)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-1	Precision	21.38	(95%-conf.int. 0.19684 - 0.23082)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-1	F_1-score	19.07	(95%-conf.int. 0.17350 - 0.20780)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-1	Recall	16.52	(95%-conf.int. 0.14797 - 0.18330)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-1	Precision	25.41	(95%-conf.int. 0.22838 - 0.27975)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-1	F_1-score	19.58	(95%-conf.int. 0.17740 - 0.21445)
$\text{BF}_{\texttt{SIM}}$	ROUGE-1	Recall	16.20	(95%-conf.int. 0.14571 - 0.17772)
$\text{BF}_{\texttt{SIM}}$	ROUGE-1	Precision	20.57	(95%-conf.int. 0.18603 - 0.22646)
$\text{BF}_{\texttt{SIM}}$	ROUGE-1	F_1-score	17.67	(95%-conf.int. 0.16109 - 0.19223)
$\text{BF}_{\texttt{DIV}}$	ROUGE-1	Recall	16.10	(95%-conf.int. 0.14351 - 0.17882)
$\text{BF}_{\texttt{DIV}}$	ROUGE-1	Precision	24.32	(95%-conf.int. 0.21918 - 0.26773)
$\text{BF}_{\texttt{DIV}}$	ROUGE-1	F_1-score	18.94	(95%-conf.int. 0.17084 - 0.20833)
$\text{SR}_{\texttt{SIM}}$	ROUGE-1	Recall	23.93	(95%-conf.int. 0.20827 - 0.27298)
$\text{SR}_{\texttt{SIM}}$	ROUGE-1	Precision	14.93	(95%-conf.int. 0.13257 - 0.16516)
$\text{SR}_{\texttt{SIM}}$	ROUGE-1	F_1-score	16.82	(95%-conf.int. 0.15440 - 0.18113)
$\text{SR}_{\texttt{DIV}}$	ROUGE-1	Recall	18.03	(95%-conf.int. 0.16219 - 0.19797)
$\text{SR}_{\texttt{DIV}}$	ROUGE-1	Precision	19.22	(95%-conf.int. 0.17190 - 0.21243)
$\text{SR}_{\texttt{DIV}}$	ROUGE-1	F_1-score	18.08	(95%-conf.int. 0.16353 - 0.19798)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-1	Recall	17.16	(95%-conf.int. 0.15419 - 0.18930)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-1	Precision	26.74	(95%-conf.int. 0.24081 - 0.29590)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-1	F_1-score	20.46	(95%-conf.int. 0.18689 - 0.22330)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-1	Recall	18.36	(95%-conf.int. 0.16564 - 0.20373)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-1	Precision	14.81	(95%-conf.int. 0.13481 - 0.16238)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-1	F_1-score	15.61	(95%-conf.int. 0.14482 - 0.16763)

Figure A.35: ROUGE-1 scores on the Opinosis data-set. Terms after a negation removed, window size=1.

ROUGE-2 - Negation removal, window size=1				
Candidate	Metric	Measure	Score	Confidence interval
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-2	Recall	03.47	(95%-conf.int. 0.02587 - 0.04513)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-2	Precision	04.04	(95%-conf.int. 0.03081 - 0.05136)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-2	F_1-score	03.65	(95%-conf.int. 0.02784 - 0.04678)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-2	Recall	03.06	(95%-conf.int. 0.02202 - 0.03953)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-2	Precision	04.91	(95%-conf.int. 0.03603 - 0.06189)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-2	F_1-score	03.67	(95%-conf.int. 0.02700 - 0.04664)
$\text{BF}_{\texttt{SIM}}$	ROUGE-2	Recall	03.16	(95%-conf.int. 0.02350 - 0.04081)
$\text{BF}_{\texttt{SIM}}$	ROUGE-2	Precision	04.18	(95%-conf.int. 0.03122 - 0.05349)
$\text{BF}_{\texttt{SIM}}$	ROUGE-2	F_1-score	03.52	(95%-conf.int. 0.02620 - 0.04485)
$\text{BF}_{\texttt{DIV}}$	ROUGE-2	Recall	03.17	(95%-conf.int. 0.02293 - 0.04175)
$\text{BF}_{\texttt{DIV}}$	ROUGE-2	Precision	05.02	(95%-conf.int. 0.03678 - 0.06479)
$\text{BF}_{\texttt{DIV}}$	ROUGE-2	F_1-score	03.79	(95%-conf.int. 0.02759 - 0.04928)
$\text{SR}_{\texttt{SIM}}$	ROUGE-2	Recall	03.82	(95%-conf.int. 0.02762 - 0.04997)
$\text{SR}_{\texttt{SIM}}$	ROUGE-2	Precision	02.02	(95%-conf.int. 0.01487 - 0.02631)
$\text{SR}_{\texttt{SIM}}$	ROUGE-2	F_1-score	02.42	(95%-conf.int. 0.01805 - 0.03065)
$\text{SR}_{\texttt{DIV}}$	ROUGE-2	Recall	02.53	(95%-conf.int. 0.01812 - 0.03341)
$\text{SR}_{\texttt{DIV}}$	ROUGE-2	Precision	02.92	(95%-conf.int. 0.02016 - 0.03929)
$\text{SR}_{\texttt{DIV}}$	ROUGE-2	F_1-score	02.63	(95%-conf.int. 0.01867 - 0.03518)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-2	Recall	03.24	(95%-conf.int. 0.02444 - 0.04187)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-2	Precision	05.25	(95%-conf.int. 0.03980 - 0.06681)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-2	F_1-score	03.91	(95%-conf.int. 0.02981 - 0.04953)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-2	Recall	02.26	(95%-conf.int. 0.01547 - 0.03067)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-2	Precision	01.77	(95%-conf.int. 0.01210 - 0.02409)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-2	F_1-score	01.88	(95%-conf.int. 0.01321 - 0.02546)

Figure A.36: ROUGE-2 scores on the Opinosis data-set. Terms after a negation removed, window size=1.

ROUGE-SU4 - Negation removal, window size=1				
Candidate	Metric	Measure	Score	Confidence interval
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-SU*	Recall	03.90	(95%-conf.int. 0.03138 - 0.04740)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-SU*	Precision	05.92	(95%-conf.int. 0.05142 - 0.06749)
$\text{Greedy}_{\texttt{SIM}}$	ROUGE-SU*	F_1-score	04.37	(95%-conf.int. 0.03674 - 0.05129)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-SU*	Recall	03.33	(95%-conf.int. 0.02706 - 0.04020)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-SU*	Precision	08.72	(95%-conf.int. 0.07222 - 0.10273)
$\text{Greedy}_{\texttt{DIV}}$	ROUGE-SU*	F_1-score	04.48	(95%-conf.int. 0.03734 - 0.05287)
$\text{BF}_{\texttt{SIM}}$	ROUGE-SU*	Recall	03.26	(95%-conf.int. 0.02681 - 0.03889)
$\text{BF}_{\texttt{SIM}}$	ROUGE-SU*	Precision	05.77	(95%-conf.int. 0.04824 - 0.06841)
$\text{BF}_{\texttt{SIM}}$	ROUGE-SU*	F_1-score	03.81	(95%-conf.int. 0.03247 - 0.04427)
$\text{BF}_{\texttt{DIV}}$	ROUGE-SU*	Recall	03.39	(95%-conf.int. 0.02758 - 0.04081)
$\text{BF}_{\texttt{DIV}}$	ROUGE-SU*	Precision	08.48	(95%-conf.int. 0.07103 - 0.09825)
$\text{BF}_{\texttt{DIV}}$	ROUGE-SU*	F_1-score	04.49	(95%-conf.int. 0.03731 - 0.05270)
$\text{SR}_{\texttt{SIM}}$	ROUGE-SU*	Recall	07.23	(95%-conf.int. 0.05594 - 0.09113)
$\text{SR}_{\texttt{SIM}}$	ROUGE-SU*	Precision	03.07	(95%-conf.int. 0.02499 - 0.03696)
$\text{SR}_{\texttt{SIM}}$	ROUGE-SU*	F_1-score	03.25	(95%-conf.int. 0.02803 - 0.03644)
$\text{SR}_{\texttt{DIV}}$	ROUGE-SU*	Recall	03.90	(95%-conf.int. 0.03272 - 0.04577)
$\text{SR}_{\texttt{DIV}}$	ROUGE-SU*	Precision	05.07	(95%-conf.int. 0.04164 - 0.06033)
$\text{SR}_{\texttt{DIV}}$	ROUGE-SU*	F_1-score	03.96	(95%-conf.int. 0.03331 - 0.04573)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-SU*	Recall	03.44	(95%-conf.int. 0.02829 - 0.04064)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-SU*	Precision	09.24	(95%-conf.int. 0.07698 - 0.10942)
$\text{SR}'_{\texttt{SIM}}$	ROUGE-SU*	F_1-score	04.67	(95%-conf.int. 0.03975 - 0.05453)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-SU*	Recall	04.00	(95%-conf.int. 0.03243 - 0.04920)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-SU*	Precision	02.92	(95%-conf.int. 0.02442 - 0.03432)
$\text{SR}'_{\texttt{DIV}}$	ROUGE-SU*	F_1-score	02.78	(95%-conf.int. 0.02449 - 0.03149)

Figure A.37: ROUGE-SU4 scores on the Opinosis data-set. Terms after a negation removed, window size=1.

Index