UNIVERSITÉ DE NEUCHÂTEL
FACULTÉ DES SCIENCES

Institut d'informatique

T H È S E

Pour l'obtention du grade de docteur ès sciences par

Olena KUMMER

Feature Weighting Approaches in Sentiment Analysis of Short Text

Jury de Thèse:

Prof. **Jacques Savoy**, *directeur de thèse*
Université de Neuchâtel, Suisse

Prof. **Patrice Bellot**, *rapporteur*
Université Aix-Marseille, France

Prof. **Pascal Felber**, *rapporteur*
Université de Neuchâtel, Suisse

Prof. **Cyril Labbé**, *rapporteur*
Université Joseph Fourier, France

Soutenu le 3 septembre 2012

UNIVERSITÉ DE
NEUCHÂTEL

FACULTE DES SCIENCES
Secrétariat-Décanat de la faculté
Rue Emile-Argand 11
CH-2000 Neuchâtel

IMPRIMATUR POUR LA THESE

Feature weighting approaches in sentiment analysis of short text

Olena KUMMER

UNIVERSITE DE NEUCHATEL

FACULTE DES SCIENCES

La Faculté des sciences de l'Université de Neuchâtel autorise l'impression de la présente thèse

sur le rapport des membres du jury :

Prof. Jacques Savoy, Université de Neuchâtel, directeur de thèse
Prof. Pascal Felber, Université de Neuchâtel
Dr Cyril Labbé, Université Joseph Fournier, Geneoble, France
Prof. Patrice Bellot, LSIS, Université Aix-Marseille, France

Le doyen

Prof. Peter Kropf

Neuchâtel, le 3 décembre 2012

Téléphone : +41 32 718 21 00 E-mail : secretariat.sciences@unine.ch www.unine.ch/sciences

Abstract:

In this thesis, we propose a supervised classification scheme based on computation of the statistical scores for the textual features. More specifically, we consider binary classification (opinionated or factual, positive or negative) of the short text in the domains of movie reviews and newspaper articles. We analyze the performance of the proposed models on the corpora with the unequal sizes of the training categories.

Based on our participation in different evaluation campaigns, we analyze advantages and disadvantages of the classification schemes that use Z scores for the purpose of classifying a sentence into more than two categories, e.g. positive, negative, neutral and factual. As a new feature weighting measure, we give an adaptation of the calculation of the Kullback-Leibler divergence score, called KL score. Considering the performance of different weighting measures on training corpora with unequal sizes, we chose two best performing scores, Z score and KL score. Thus, we propose a new classification model based on the calculation of normalized Z score and KL score for the features per each classification category. One of the advantages of this model is its flexibility to incorporate external scores, for example, from sentiment dictionaries.

The experiments on datasets in Chinese and Japanese show a comparable level of performance of the proposed scheme with the results obtained on the English datasets without any use of natural language specific techniques. The advantage of the approaches analyzed in this thesis is that they can work as quick and easily interpretable baselines for short text classification.

Keywords: Sentiment Analysis, Opinion Detection, Natural Language Processing, Machine Learning, Data Mining, Feature Selection, Text Classification.

Acknowledgements

During my time at the University of Neuchâtel I met a lot of interesting people that in some way or another helped or encouraged me. First of all, I would like to thank Professor Jacques Savoy who gave me a life-changing opportunity to do my PhD under his supervision. I appreciate his help, patience and kindness over the past years.

I would also like to express my thanks to Professors Pascal Felber, Patrice Bellot and Cyril Labbé for being part of the jury.

I will guard many fond memories about the times spent in discussions and outings with the colleagues from the Institute of Informatics.

I want to say special thanks to Katya Wahler and Elvire Beziz, that are there for me no matter what. True friendship is rare and priceless.

The two people, my unconditional supporters, and for whom this work is very important, are my parents, Olga and Vladimir Zybarevy. I would not be able to be where I am without them. This work is dedicated to them.

At last but not least, I want to express my gratitude to my husband, Raphaël Kummer. Without his love and support I would not be able to finish this work. He helped me to go forward and not to lose track of the important things. I cannot find words to describe in how many ways I feel blessed to have him by my side.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The automatic detection of opinions and their polarity is the main subject of this PhD thesis. The main concern of this research domain is the detection of the opinionated or subjective content within a document. This may include many diverse subtasks, such as opinion detection, identification of the polarity of the sentiment expressed (positive, negative, or neutral), opinion target and holder identification, evaluation of the strength of the opinion. Another example could be detection of the opinion about one specific feature of a given product or service (e.g., shutter speed of a photo camera). All of these subtasks relate to the task of *opinion detection*, other most common terms include *opinion mining*, *sentiment analysis*. In this thesis, the term *sentiment analysis* (SA) is used interchangeably with the term *opinion mining*. This decision was influenced by the use of these definitions in the book by Pang *et al.* [Pang 2008] where they are used in the meaning of "the computational treatment of opinion, sentiment, and subjectivity in text".

In terms of the text size, one can differentiate the task of opinion detection on different levels of text granularity, such as: document, passage, sentence, or clause. We can consider short posts or sentences (e.g., tweets, Facebook), or paragraphs and even documents (reviews, blog posts, news articles). Additional challenge to opinion and sentiment classification present the differences in the ways the opinions are expressed depending on the topic and genre of the text. For example, movie reviews tend to have colloquial style and contain more spelling errors and neologisms than newspaper articles. At the same time, it is important to consider the robustness in performance of opinion mining approaches to the specificities of different written natural languages. Thus,

character-based languages as Chinese and Japanese, would require different decisions on the stage of initial text pre-processing and word tokenization than English, or other European languages.

The approaches to solve the problems discussed above encompass a variety of research areas, from Natural Language Processing (NLP), Machine Learning (ML), Information Retrieval (IR) to Computational Linguistics (CL). Taking into account the amount of subtasks in the domain, we limit our investigation to the binary task of opinion and sentiment detection (opinion or no opinion, positive or negative sentiment). The main focus of this thesis is to examine the effectiveness of the proposed classification approaches based on the calculation of the modified statistical scores for the task of opinion and sentiment classification of sentences. This is done on the example of the datasets pertaining to two domains (movie reviews and newspaper articles) and in three different natural languages (English, Chinese, and Japanese).

This chapter is constructed as follows. First, the context of the present research is presented in Section 1.1. Followed by the discussion of the motivations for this work. Section 1.3 describes the challenges and main approaches to the task of sentiment analysis. Next, research objectives are presented in Section 1.4, followed with the overview of the thesis in Section 1.5.

1.1 Context

During the last two decades the role of the average Internet user gradually changed from a mere observer, to an active participant in content generation on the Web. This phenomenon, frequently called Web 2.0, has resulted in a possibility to add and exchange information for millions of users by the means of social platforms, networks, blogs, tweets, reviews etc. Such wealth of information calls for specific tools and techniques in order to analyze and process this data. The result of such analysis could be potentially important and profitable in identify-

ing future trends for consumer research, market analysis, and other organizations.

One of the salient features of the Internet use nowadays is the ability and easiness with which the user can add comments, create short texts, tweets, or long posts, blogs, share them with their friends or contacts, engage in discussions, share information, give opinion on about all subjects related to human activity. All of this requires several clicks and no specific knowledge from the user side. The most popular sites include social networks, such as Facebook and MySpace, Twitter, web newspapers and magazines. More and more existing web sites and applications integrate a possibility for user interaction and opinion exchange in some form. The shared information includes the latest trends and fashions, likes and dislikes, and generally people's thoughts from specific subjects to everyday topics.

As a consequence of so much data produced and growing constantly, the importance and the value of the information that it could potentially provide becomes obvious. Therefore, the task of analysis and classification of user-generated content has aroused an extensive interest in the research community [Pang 2008, Savoy 2011]. As a result, there are several datasets that were created in the domain of movie reviews and newspaper articles that have been annotated as to opinion and/or sentiment contained in text. In this thesis, we limit our experiments to these two domains, as there exist annotated benchmark corpora facilitating the comparison to other studies in the field. The movie review datasets are generally characterized by the presence of slang, writing errors, abbreviations, jargon, all characteristics of the freely generated text. Thus, representing a challenging task for opinion classification. As opposed to the user-generated content, newspapers provide a more structured and less emotional text in terms of style, yet more subtle in opinion expression, thus rendering it also difficult for analysis.

1.2 Motivation

Sentiment analysis of user reviews, blog posts, and short texts could be of interest for many practical reasons. It represents a rich resource for marketing research, social analysts, and all interested in following opinions of the mass. For example, a manufacturer could be interested in reaction to the new product release to form a customers view of their product or to detect new consumer wishes, or a government agency interested in carrying out a pre-poll analysis of people's positive or negative attitudes towards some issues.

It has become a frequent practice for users to search how others rate a certain product or service before buying it. Thus, with the growing number of the reviews it is harder to evaluate the results using the common fact-oriented search engines. An ordinary search engine, such as *Google* or *Yahoo!* ranks websites according to their relevance to the query, while opinion search would include an additional task of selecting web pages containing opinionated text relevant to the topic. When given a query, one receives a single relevance list, limited by the topic relevance. This list is unable to objectively represent positive and negative reviews equally, or facilitate user comprehension of the overall sentiment of each item in the retrieved set.

Opinion mining can also be useful in a variety of other applications and platforms, such as recommendation systems [Terveen 1997], product ad placement strategies [Jin 2007], question answering [Somasundaran 2007, Stoyanov 2005] and information summarization [Seki 2004]. Another major area of application of opinion mining is research on political moods and thoughts of the voters [Yu 2008]. Sentiment analysis could be a key to summarizing and analyzing public opinion on different issues such as a new proposed regulation (e.g., nuclear plant ban), passing laws etc.

Besides marketing research and political analysis, opinion mining can be used as a tool to gather and analyze opinions, interactions in different so-

cial science applications. Blogs have become a medium by means of which thousands of users share their experiences. Thus, forum and blog search could be more sentiment oriented to facilitate the follow-up of bloggers moods and opinions [Savoy 2010]. Other opinion mining applications could include email prioritization [Durbin 2003], analysis of newspaper articles and in general sentiments expressed not only by individual users, but also in the press [Seki 2007, Seki 2008, Seki 2010].

Faced with all the variety of opinion mining applications, it is important to be able to distinguish opinionated sentences from factual. This process could be relatively simple in some cases, when a sentence contains direct speech, exclamations, or any *emotional description*, but usually it is a challenging task due to the nature and variability of sentiment expression available in the natural language. Consider these two examples:

1. Many citizens were also anxious that the tragedy could affect an already sagging economy.
2. It will accelerate the process of mutual economic interdependence and promote trade and finance between the countries of Europe and foster alliances and mergers between European corporations. If Japan spends money, the economy will pick up.

While the first example (1) is labeled as opinionated by all three human judges in NTCIR-8 evaluation campaign, the second sentence (2) was considered opinionated only by one of the judges. From an overview paper of the NTCIR-8 campaign the average inter-annotator agreement is around 73%, underlining the subjective nature of how written language is perceived, specifically by different human judges or by some people at different time periods [Seki 2010].

What is an opinion? Though we can find a dictionary definition, the detection of an opinion remains subjective. According to Collins dictionary, opinion is a judgment or belief not founded on certainty or proof. [Col 2003]. It is a personal view, attitude, or appraisal that falls short of absolute certainty.

Wiebe *et al.* [Wiebe 1994] introduced the notion of *subjectivity* that includes opinions, emotions, speculations on the information not open to objective verification. The question of what can be considered an opinion, however, is still something that is defined implicitly by human assessment. One of the challenges of SA that distinguishes it from the topic-oriented information retrieval, is that sentiment in text can be expressed in a subtle manner, or using neutral words or sarcasm, for example:

> 3. In terms of accounting and distribution strategy, it's simpler to work with than if each country had retained an individual currency.
> 4. And only these policies, coupled with diplomacy, can bring about a denuclearization and arms reduction on the Korean Peninsula.

Sentiment analysis represents a computational study of opinions, sentiments and emotions expressed in text. In order to avoid any misunderstandings and discrepancies with the reviewed literature, we would like to clarify the terminology used in this thesis. Since the problem of opinion analysis of large amount of text is quite recent, a lot of research was carried out in parallel, hence, a vast amount of different existing terminology. This leads to a situation where there are many terms adopted in the literature signifying sometimes one, sometimes different aspects of opinion mining.

The term *opinion mining* first appeared in the paper by Dave *et al.* [Dave 2003] that deals with problems focused on extracting and evaluating judgments on specific topics. The term *sentiment analysis* appeared in the work of Das *et al.* [Das 2001] where it is described as a study that focuses on the analysis of text with a sentiment prediction as a result. Though,it is possible to argue that each of these terms can be used in a broader or more specific sense [Pang 2008]. We chose two terms as descriptions of opinion mining task that is addressed in this thesis. First, the term *polarity detection* is used in the sense of classifying text according to its polarity, positive or negative. *Opinion detection* refers to the task of classifying the input text in opinionated and not opinionated (factual) categories. To denote the whole field of study as in [Pang 2008] and as has been

mentioned before, we use the terms of *opinion mining* and *sentiment analysis* interchangeably throughout the thesis.

1.3 Challenges in Sentiment Analysis

We consider a sentiment analysis task as a binary text classification task of a textual unit into two categories: opinionated and factual. The same can be done for sentiment polarity classification (positive and negative). If this general task is put this way, it ideally fits into the text categorization model. But does it really? While text categorization problems classify documents by topic, or topics (e.g., sport, politics, entertainment), SA has basically two or, maximum four (taking into account positive, negative, neutral, and factual) classification categories. Moreover, in text classification, the categories are usually very well separated in the sense of relation to different concepts by the nature of topic-based factual classification. In SA, it can be quite hard to differentiate neutral opinion from negative, or even no opinion at all. Consider the following examples:

> - While Kim's policy of engagement toward North Korea has come under fire at home, Obuchi expressed his support for his host.
> - And questions should not be pre-screened. U.S. policy toward North Korea could even emerge as a campaign issue in next year's U.S. presidential elections.
> - At the same time, Japan, the United States and South Korea should cooperate more closely on the missile issue.

The three sentences represent positive, neutral, and negative opinions respectively, as annotated by assessors in NTCIR-7 campaign [Seki 2008]. While the positive polarity is quite obvious, one could argue about the classification category for the other two sentences.

One of the important characteristics of the opinionated sentence is a de-

gree of strength of the sentiment expressed. Consider the following two sentences:

> - Malone does have a gift for generating nightmarish images that will be hard to burn out of your brain.
> - It won't hold up over the long haul, but in the moment, Finch's tale provides the forgettable pleasures of a Saturday matinee.

It is easier to estimate the strength of the opinion when dealing with movie reviews. Very frequently, the task of classifying movie reviews is carried out on datasets that besides the review itself have a rating (usually from one to five stars) given by the user himself/herself and serving as a class-label at the same time.

Another challenging factor in opinion mining, that makes it stand out from the problems of text classification, is the subtlety of opinion expression in text. True, if you consider the following sentence, you could notice that without the use of any words with negative connotation the sentence still expresses a negative sentiment:

> - The script, the gags, the characters are all direct-to-video stuff, and that's where this film should have remained.

In general, it has to be remarked that sentiment-bearing, subjective terms are very context specific. Although, there are words and phrases that retain their positive or negative meanings across different domains, some expressions can change their meaning from one domain to another. Thus, a phrase *It is quiet cheap.* could be considered positive in the domain of product reviews and negative in the domain of movie reviews.

Overall, we can divide the work in the area of SA based on the particular unit of text that is being analyzed. There are directions focused on word, clause/phrase, document sentiment analysis. As early as in 1997, Hatzivassiloglou *et al.* [Hatzivassiloglou 1997] focused on semantic orientation of words.

More recently, in 2002, Turney *et al.* [Turney 2002] used the same idea but calculated the scores of words using Internet hit counts. In the present thesis we focus on the task of sentence classification. We give an overview of the methods for document classification since they use sentence-level analysis. Thus, they take into account that the document consists of paragraphs and sentences that can have no opinion, or two clauses with completely opposite sentiment polarities. For example, Pang *et al.* [Pang 2004] first detect the objective parts of the document, label sentences as objective and subjective and reformulate the task as to find a minimum s-t cut [Kleinberg 2006] in the graph constructed from labeled sentences.

As we mentioned before, SA is a research area that lies in the crossroads of several major computer science fields, such as natural language processing, information retrieval, machine learning. Although more detailed literature overview will be presented in Chapter 2, here are some of the examples of how these research fields contribute to the domain of opinion mining. Thus, it has been remarked that a lot of opinionated phrases follow the pattern of "adjective+noun". In this case NLP techniques such as part-of-speech (POS) tagging, automatic marking of words with their corresponding parts of speech, may prove to be useful. Information retrieval concerns with the task of finding and ranking relevant documents to the query. All current well-known search engines perform this task relatively well for the factual search. Meanwhile, as it has been mentioned earlier, sentiment search remains quite difficult. One of the reasons for this could be that most algorithms are highly dependent on the presence of searched words and not on the overall context of the text relevant to the query. Nevertheless, some of the research has been focused on adaptation of IR metrics and algorithms to SA task. Machine learning approaches mostly predominate the specter of works in opinion mining. With the adaptation of supervised and semi-supervised approaches to sentiment analysis, the use of such techniques as SVM and naïve Bayes are known to provide one of the best performances.

Another important research direction is the cross-domain and cross-language sentiment analysis. As there is a lack of labeled data, especially in natural languages other than English, it might be interesting to transport a learned model from one language to another, or for that matter from one classification domain to another. Overall, the studies show that opinion mining is quite domain-specific. Another interesting question, is to experiment how the proposed approaches adapt to the data in other languages. Since there is an availability of some of the comparable data in Chinese and Japanese by the means of NTCIR campaigns [Seki 2007, Seki 2008, Seki 2010], experiments on these datasets with the proposed approaches were carried out.

In this thesis, we experiment with the classification approaches that are trained on the annotated data. In order to verify their effectiveness we carry out experiments on the annotated corpora. A lot of times their use limited by the size of the training data available. While some of the datasets, constructed specifically for experimental purposes, contain equal number of sentences in both classification categories, other datasets may not necessarily have the same virtue. Thus, the corpora used in NTCIR campaigns, contains sentences of the news articles, where, much less than a half of the sentences are judged as opinionated. Therefore, one has to take into account the difference in the size of the training sets per category and possible bias of the classification model. Since we experiment with both types of datasets, we use the terms *balanced* and *unbalanced* for datasets with equal and unequal training set sizes respectively.

1.4 Research Objectives

Faced with a task of opinion detection in text, we need to take a number of decisions concerning text pre-processing and generation of input for the chosen classification approach. In this section, after presenting an opinion mining system overview, we give the objectives of our research.

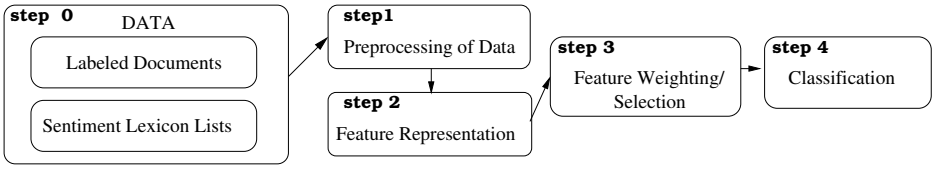

Figure 1.1: Overview of the Sentiment Analysis model.

The main steps taken when performing opinion and sentiment detection are presented in Figure 1.1. Let's assume that we have a dataset to categorize in two classes, e.g., opinionated and factual (step 0). First, some initial pre-processing of text is done (step 1). This usually involves parsing of text, tokenization, decapitalization of words, sometimes stripping the punctuation. On the step 2, one decides how to represent documents in the system, what to consider as a *feature*, an item that represents a part of text (part of the word or a short sequence of words).

It is important to decide whether to use directly words from text or apply a stemmer or more sophisticated analysis, such as lemmatizer. Next, the features are weighted in terms of their polarity and importance (step 3). And at last, a new document is classified based on the given model (step 4). It is fair to note, that the steps 1 to 3 do not necessarily need to depend on the given data, but can be extended with the use of other external resources, such as sentiment dictionary lists. With this framework in mind, here are the tasks addressed in this work:

Evaluation of textual pre-processing techniques: We investigate pre-processing approaches and combinations of thereof in order to determine the effective feature representation using different indexing schemes, stop lists, and stemming. The aim here is to determine whether a certain combination of pre-processing techniques gives better results on the tested corpora.

Evaluation of information measures for feature weighting: We carry out a comparative study of several information measures used for sentiment classification on a sentence level. We analyze the performance of the measures on balanced and unbalanced datasets. The aim is to identify the best performing measures given different training setups for the SA task on a sentence level.

New approach based on feature weighting for SA: We propose the modification of the computed scores in order to take into account such factors as the size of the dataset, and the dependency between terms in the sentence. We carry out experiments reweighting the computed scores using external evidence: word list, sentiment dictionaries.

Evaluation of the proposed model on Chinese and Japanese corpora: After participating in two NTCIR campaigns, we carry out experiments to verify how the proposed classification model performs on corpora in Chinese and Japanese. The aim is to investigate whether it is possible without any language-specific heuristics to obtain a similar level of performance of the proposed scheme on the corpora in a completely different natural language.

Another objective of this thesis is the development of a classification model that can be tuned for a specific natural language with the help of external resources. This kind of system could be optimized for participation in various IR campaigns, like TREC and NTCIR, as well as customized for other applications.

1.5 Thesis Organization

The rest of this thesis consists of six further chapters. Chapter 2 gives an overview of the related work. In Chapter 3 we describe the experimental setup, evaluation measures, and datasets used. In this work, we propose a classification scheme based on feature score calculation according to an information measure. Based on the previous work on the use of the Z score

for text classification, we present the Z score based classification algorithm for sentiment polarity detection, as well as the evaluation of other statistical measures in Chapter 4. Additionally, we propose an adaptation of the Kullback-Leibler divergence score (KL score) for the sentiment classification task.

Chapter 5 describes the modifications applied to the calculation of the two statistical scores, Z score and KL score, taking into account limitations of their application for the opinion classification task. We propose a classification model that is based on the combination of the two modified scores. We also give an evaluation of different statistics computed for the proposed model and their individual performance as class predictors for sentences.

In Chapter 6 we present further experiments with the proposed model and the sentiment lexicons. We also evaluate the use of the SVM with the scores introduced in the previous chapter. Furthermore, to test the adaptability of the proposed approach to other natural languages, we present our experiments with Chinese and Japanese corpora. Finally, Chapter 7 concludes the thesis presenting the main contributions and possible future developments.

Chapter 2

State of the Art

2.1 Introduction

This chapter is dedicated to the literature overview of the sentiment analysis domain. We give an overview of the two principally different approaches to opinion classification and discuss the related studies.

One of the first ideas that comes to mind when analyzing text is to count words with positive and negative polarity in the text and make a decision based on which count is bigger. For example, a set of positive words (a seed set) could include *beautiful*, *praise*, negative *hate*, *despise*, *blame*. Although, grammar has to be taken into account, for cases like *not so bad*, *incredibly awful*, *rarely useful* etc. Not mentioning, as it has been noted in Chapter 1, the formulation of opinion in neutral words, such as *stood up and left*, *without words to describe*.

The technique described constitutes the main idea of one of the two approaches to sentiment analysis. The words are assigned sentiment scores that show the strength of the specific sentiment expressed, then sentence subjectivity is derived from these sentiment scores. At last, a decision on the document subjectivity is made. This constitutes a lexicon-based technique that goes bottom-up, from words to sentences, and to document level.

Another major approach is top-down: first we get a set of labeled documents by their sentiment. Then, data mining or text categorization methods are used to learn the text model. The sentiment attached to words is derived from

the model. These two approaches are also called unsupervised and supervised, where the main distinction is that supervised approaches are based on a set of labeled data to learn a model.

Since these two techniques are the most used in sentiment analysis, we made a decision to cover the literature survey from these two perspectives. This chapter is organized as follows. Section 2.2 exposes history and background overview, while Section 2.3 presents unsupervised methods. Section 2.4 describes the most important supervised approaches. Section 2.5 concludes this chapter with a summary and discussion.

2.2 History and Background Overview

The first works in SA started to appear relatively recently, in the 90s [Kessler 1997, Hatzivassiloglou 1997]. The precursors of work in SA deal with detection of types of attitudes, private states, classification of narrative, metaphor, statistical grammar generation and analysis [Quirk 1985, Sharman 1990, Hearst 1992, Sack 1994, Wiebe 1990, Wiebe 1995]. The early works in the study of affect were highly influenced by related works in psychology and linguistics. Thus, some works include categorization of text into a set of finite categories describing emotions [Ortony 1990]. This approach is quite hard to formalize, since it involves the understanding of the psychological context, difficult to induce from written text. Other approaches include the study of affect in multi-dimensional semantic space [Osgood 1971, Watson 1985], where the emotions are classified along the two axis: good vs. bad and strong vs. weak.

In the last decade the interest to the topic of opinion mining grew substantially, so we can say that it constitutes one of the major research directions that is studied in natural language processing, machine learning, information

retrieval and text classification. This can be justified by several factors: availability of labeled data from Internet that can be used to learn models, and a high demand on analysis of the user-generated content. There are a number of conferences that have taken an interest in the subject, such as ACL, WWW, EMNLP, as well as workshops and campaigns - TREC [Macdonald 2007], NTCIR [Seki 2007, Seki 2008, Seki 2010]. Nevertheless, only recently some textbooks appeared on the topic [Pang 2008].

Before proceeding to the overview of the state of the art, it is necessary to note several challenges in analyzing the results in the domain. The same classification algorithms used by several researchers may show different performance, since classification would involve a number of decisions on how to pre-process the text data, e.g., use of stemming, stop word removal, etc. There is a choice of the text representation in the model (unigram, bigram), decisions on how to account for positional information in text, POS tags, and punctuation. Small differences in these decisions may lead to different performance of the same methods.

2.3 Unsupervised Approaches

It is possible to subdivide unsupervised or lexicon based approaches in two groups. First, we can find approaches that use a large corpora to find co-occurrences of a small seed list of words in order to find other words with sentiment connotations [Hatzivassiloglou 1997, Turney 2002]. Second, methods that use sentiment dictionaries and lists, such as WordNet [Miller 1995] or General Inquirer [Stone 1966] to infer the word sentiment polarity. The first strategy will be exposed in Section 2.3.1. The second strategy will be presented in Section 2.3.2.

2.3.1 Corpus-Based Methods

A lot of research in SA has been focused on extracting specific words, or given POS. Some of the POS (like adjectives) or sequences of POS (adjective-noun) have shown to be more effective in opinion detection than others. The work of Justeson *et al.* [Justeson 1995] adopts an NLP approach based on part-of-speech (POS [Toutanova 2000]) filtering. The words in the text are automatically processed and marked with the appropriate POS tags. Afterwards specific POS or given phrase patterns are filtered from text, for example two adjectives in a row, or adjective, conjunction, adjective.

Hatzivassiloglou *et al.* [Hatzivassiloglou 1997] describe an approach based on the idea that the conjoined adjectives have the same orientation, apart from *but* which is used with opposite orientation. They construct two clusters of adjectives using the conjunction counts based on the Wall Street Journal articles. Although they achieve quite high accuracy, it is important to note that they manually eliminated neutral adjectives on the first step. Other studies focused on analyzing single words and POS to automatically deduce the polarity of the word from the data presented [Hatzivassiloglou 2000].

One of the popular approaches to construct a sentiment lexicon is based on point-wise mutual information (PMI) between words. Thus, the idea is that pairs of words that occur often together would have a similar orientation. Turney [Turney 2002] proposes a technique that calculates a PMI between adverbs and adjectives, denoted as w and filtered out from an evaluated text, with the words *excellent* (positive) and *poor* (negative) using WWW corpus. He uses the AltaVista search engine and the following computational formula on the example of *excellent*:

$$PMI(w, excellent) = log_2(P(w\&excellent)/P(w)P(excellent)) \tag{2.1}$$

The underlying rationale of this method is that word pairs or phrases with positive semantic orientation would appear more often around a word *excellent*

than *poor*. The main drawback of the PMI method, calculated with the help of search engine hits, is its unreliable nature of constantly changing and tuned algorithms of the search engines that may provide biased and in some cases uncertain information.

Later, Turney *et al.* [Turney 2003] carried out experiments with Latent Semantic Indexing (LSI) and expansion of the set of seed words with obvious positive or negative polarity. LSI requires a construction of matrix that encodes, whether the two words or phrases appear in the same context, i.e. around the same words. The drawback of this techniques is the computational complexity, and hence, real difficulty of scaling the problem. Another conclusion from this work is that the results improve if the set of seed words includes words with strong semantic polarity. The performance dropped when using words randomly chosen from positive and negative semantically oriented lexicons.

Wiebe [Wiebe 2000] uses bootstrapping technique on a manually chosen seed list to find words with strong subjectivity. In order to achieve that, the words are clustered according to their distributions in the corpus. Later, Wiebe *et al.* [Wiebe 2005] extend this technique to learn subjective patterns of expressions. They use a seed word list to extract a subset of the training corpora for further analysis by the probabilistic classifier. Baroni *et al.* [Baroni 2004] also use a seed word list and calculate mutual information between words in this list and adjectives from text using the co-occurrence counts from the AltaVista search engine.

Although POS tagging is not shown to give substantial improvement in sentiment analysis [Pang 2008], Bekkerman *et al.* [Bekkerman 2006] show that using POS tagging with features is better than features by themselves for the unsupervised non-topical classification of documents by genre and style. They use the bag-of-words text representation with the morphological information (POS tags) and simultaneously cluster documents, words in documents and POS

n-grams of words by maximizing pair-wise mutual information. It is intuitive to understand the role of POS tags for style and genre classification. There, it is possible to measure statistically the use of certain POS, while in sentiment analysis adjectives, adverbs have been shown to carry more sentiment weight than other POS, they remain additional and secondary classification features.

Other approaches include the use of defined verb categories (*characterize, declare, conjecture, admire, judge, assess, say, complain, advise*) and their features (a verb corresponding to a given category occurring in the analyzed text) that may be pertinent as a classification feature [Bloom 2007]. However, words such as these cannot always work correctly as clues. For example, let's consider a word *said* in the two sentences below:

- The iPhone price is expensive, said Ann.
- The iPhone price is 600 dollars, said Ann.

Both sentences contain the clue word *said* but only the first one contains an opinion on the target product.

In the study carried out by Su *et al.* [Su 2008] on MPQA (Multi-Perspective Question Answering) and movie reviews corpora it is shown that publicly available sentiment lexicons can achieve the performance on par with the supervised techniques. They discuss opinion and subjectivity definitions across different lexicons and claim that it is possible to avoid any annotation and training corpora for sentiment classification. Overall, it has to be noted that opinion words identified with the use of the corpus-based approaches may not necessarily carry the opinion itself in all situations. For example, *He is looking for a good camera on the market.* Here, the word *good* does not indicate that the sentence is opinionated or expresses a positive sentiment.

2.3.2 Dictionary-Based Methods

One of the popular research trends include the use of the lexical database WordNet [Miller 1995]. It provides the grouping of words into synonym sets (called synsets) and semantic relationship between them, such as antonyms, hyponyms, etc. Kamps *et al.* [Kamps 2004] use WordNet to measure semantic orientation of adjectives by counting the number of synonym links from the analyzed adjective to the seed words, such as *good*, *bad*, etc. One of the successful uses of the WordNet in order to construct a semantic lexicon belongs to Esuli *et al.* [Esuli 2006b, Esuli 2006a]. Based on the small set of manually constructed words, they expand it using the WordNet synonym and antonym relationships of adjectives. This work led to the construction of the SentiWordNet that provides positive, negative and objective scores to each *gloss*, brief definition of the synset, in WordNet [Esuli 2006b]. One of the drawbacks of this lexicon is the variety of senses for some words that could take different scores. Therefore, a thorough POS analysis or word sense disambiguation are needed to accurately use this lexicon.

In their study, Andreevskaia *et al.* [Andreevskaia 2008] propose to use glosses and lexical relations from WordNet. They start with a small seed word list and extended it by the means of lexical relations in WordNet (synonymy, antonymy and hyponymy). Later, they extract words carrying sentiment from glosses, and assign a polarity to the extracted terms. This is done by computing the word's degree of membership in a specific category based on how many times this word has been assigned a specific category.

Dang *et al.* [Dang 2010] evaluate a lexicon enhanced method for sentiment classification of online product reviews. They group features in three groups: syntactic or content-free (function words, such as determiners, pronouns, conjunctions, prepositions; punctuation, structural features), content-specific (unigrams and bigrams occurring more than five times) and sentiment features (with POS tags extracted and weights derived from sentiment lexicons). They

propose to use a SentiWordNet scores only for adverbs and verbs retrieved with POS tagging. The experiments are carried out with the use of different groups of features. As a result, they show that the more feature groups are involved, the better the classification accuracy.

Zagibalov *et al.* [Zagibalov 2008b] use a seed word list and a notion of probability for the feature learning and weighting scheme, obtaining good results on the product review corpora. They argue that opinion classification lies in a continuum, rather than in discrete class classification. In their work, Rahayu *et al.* [Rahayu 2010] use opinion word recognition component in the system that extracts rationale from online reviews and ratings. They identify all the related adjectives to the specific product feature and use a training procedure with a seed list of opinion words to infer the polarities of the found adjectives.

Kennedy *et al.* [Kennedy 2006] use General Inquirer [Stone 1966] to classify reviews based on the number of positive and negative terms that they contain. General Inquirer assigns a label to each sense of the word out of the following set: *positive*, *negative*, *overstatement*, *understatement*, or *negation*. Negations reverse the term polarity while overstatement and understatements intensify or diminish the strength of the semantic orientation.

One of the major shortcomings in the use of the sentiment lexicon is the difficulty to identify opinion words and expressions specific to the domain or context where they are used. Another important point is that a word with a high opinion score in the lexicon might not always convey an opinion in the actual phrase. It is also true, that sometimes words with neutral or no sentiment orientation might bear an opinion. Thus, it presents an interest to overview the existing machine learning and information retrieval approaches to opinion mining that tackle these problems.

2.4 Supervised Approaches

In this section we present an overview of the state of the art of supervised approaches to sentiment classification. As it has been mentioned before, the field of sentiment analysis lies on the crossroads of several research fields. Natural language processing techniques are used mostly in feature pre-processing, lemmatization, stemming, POS tagging. Computational linguistic tools and techniques are concerned with language grammar generation and modeling. They provide an interesting formulation and approach to the task, that, nevertheless, cannot be directly applied to computer science methods and requires adaptation. Therefore, we decided to concentrate this overview on the most prominent and used techniques from machine learning (Section 2.4.1) and information retrieval (Section 2.4.2) domains.

2.4.1 Machine Learning Methods

The field of machine learning has provided many models that are used to solve various text classification problems. Among them are naïve Bayes, SVM, Decision Trees, Maximum Entropy, Hidden Markov Models. The detailed overview of these and other algorithms can be found in [Witten 2005]. So far, the most popular machine learning approaches used as baselines are Support Vector Machine (SVM) and naïve Bayes (NB).

Pang *et al.* [Pang 2002] analyzed several supervised machine learning algorithms on a movie reviews dataset, among them SVM, naïve Bayes and maximum entropy. They also tested different feature selection techniques. Features are usually considered to be words, or bigrams of words, that could have been somehow pre-processed, for example, stemmed or lemmatized. They report the best performance using SVM method with unigram text representation. It has to be noted that the authors took into account just the presence of a feature, and not its frequency. Naïve Bayes method gave slightly lower accuracy. Taking into ac-

count for POS tagging information may improve the effectiveness of naïve Bayes and maximum entropy methods, but tends to decrease the performance for SVM.

In the later study, Pang *et al.* [Pang 2004] propose to first separate subjective sentence from the rest of the text. They assume that two consecutive sentences would have similar subjectivity label, as the author is inclined not to change sentence subjectivity too often. Thus, labeling all sentences as objective and subjective they reformulate the task of finding the minimum s-t cut in a graph [Kleinberg 2006]. They carried out experiments on the movie reviews and movie plot summaries mined from the Internet Movie DataBase (IMDB), achieving an accuracy of around 85%.

Matsumoto *et al.* [Matsumoto 2005] use word sub-sequences (*n-grams*) and dependency trees of sentences to calculate the frequent patterns in the word usage across different sentences. They use an SVM model on the selected subtrees and achieve an accuracy of 88.1% with the language-independent features on a more recent version of the data set used by *Pang et al.* [Pang 2004].

Nigram *et al.* [Hurst 2004] use Winnow algorithm, an online learning algorithm that learns from the bag-of-words representation of documents a linear classifier. They employ an iterative training procedure, exploring a high order of *n-grams* of features (with $n > 3$). The χ^2 score is used for feature selection. They compare the use of SVM, Winnow and language models. The reported results show that the best performance was achieved by SVM with the score smoothing techniques for the selected feature set.

In their study of opinion detection in financial news, Brew *et al.* [Brew 2010] evaluate SVM and naïve Bayes, finding that SVM gave better performance for sentiment polarity classification task. On the other hand, naïve Bayes outperformed SVM in classification by relevance. Another variation of the SVM method was adopted by Mullen *et al.* [Mullen 2004] who use WordNet syntactic

relations together with topic relevance to calculate the subjectivity scores for words in text. They report an accuracy of 86% on the Pang *et al.* [Pang 2002] movie review dataset.

There has been work in incorporating positional information of the words in text. Thus, Raychev *et al.* [Raychev 2009] use multinomial naïve Bayes, together with the position information in the feature set, with the idea that the use of a particular word can have different subjective power depending on where in the document it occurs. They conducted their experiments on the movie dataset and achieve 89% of accuracy using unigrams and bigrams, which is a slight amelioration on the performance reported by Pang *et al.* [Pang 2004].

Qiu *et al.* [Qiu 2009] incorporate unsupervised and supervised approaches. First, they determine opinionated sentences using sentiment dictionary (a limited number of sentiment words with high scores), expanding this dictionary by the means of analyzing new reviews, iterating until the word set does not change anymore. In a second step, they train an SVM classifier on the documents that were classified with high confidence in the first step, and incorporate both results, achieving good performance on the dataset of product reviews in Chinese.

Melville *et al.* [Melville 2009] perform sentiment analysis of blogs by combining lexical knowledge with text classification, using multinomial naïve Bayes classifier that incorporates background knowledge and training examples. They build on work of Liu *et al.* [Liu 2004] that use labeled features as background knowledge. First, they create a representative "document" of each class and then compute the cosine similarity between each document and word, thus estimating the conditional probability of each word belonging to a class. Second, a generative background knowledge model estimates the class priors on the training data given a lexicon with positive or negative labeled words. Afterwards, they combine the two probability distributions and attain good results using linear pooling [Melville 2009].

We might also mention OpinionFinder, a more complex system that performs subjectivity analyses to identify opinions as well as sentiments and other private states (speculations, dreams, etc.) [Wilson 2005a]. This system is based on various classical computational linguistics components (tokenization, POS tagging as well as classification tools. For example, a naïve Bayes classifier is used to distinguish between subjective and objective sentences. A rule-based system identifies both speech events (*said*, *according to*) and direct subjective expressions (*is happy*, *fears*) within a given sentence. Of course such learning system requires both a training set and a deeper knowledge of a given natural language (morphological components, syntactic analyses, semantic thesauri).

One of the inter-language studies in sentiment analysis using machine learning techniques have been conducted by Boiy *et al.* [Boiy 2008]. They use a tree structure to model the data, with feature selection, unigrams, stems, negation and discourse features relevant to the specific language. Several metrics are used to compute the path from feature to the opinion target, about which a sentiment is expressed. They discuss language peculiarities among English, Dutch and French. Multinomial naïve Bayes, SVM and maximum entropy are compared, along with the discussion of the cascade and active learning architecture with the small training set. The main findings report that the more features used, the higher the accuracy of the obtained results. The best performance is achieved with a combination of unigram features and a number of language-specific features. The fact that the addition of different features improves the result could also be due to the overall overfitting to the particular dataset.

Some studies have been concerned with the inter domain adaptability and feature analysis and selection for different specific corpora. Thus, Tan *et al.* [Tan 2007] proposed to use a classifier trained in one domain to label the top n documents in the new domain. Then, a new classifier is learned on these labeled documents. Whitehead *et al.* [Whitehead 2009] build their own sets

of corpora that is mined from the web (Amazon). To reduce the size of the feature set they eliminate stop words (that appear in more than half of the reviews) and words with unique occurrences. They use the odds ratio selection method to eliminate terms that are not useful in discriminating between the two categories. They found that keeping 15-25% of the original lexicon is optimal. They show that a general model trained on all domains has an average of 80% accuracy performance on specific domains which is close to the average of 83% of a model trained on a single domain (sport, restaurant, etc.).

2.4.2 Information Retrieval Methods

Representation of the documents in most supervised approaches is based on the Vector Space model [Salton 1975]. Every document is represented by a multi-dimensional vector, where each dimension corresponds to some feature (term) in a document. Thus, a collection of documents can be represented, for example, as a term-document matrix, where an element (x, y) represents the number of times feature x was encountered in document y. The idea is that it is possible to separate two classes of documents represented as vectors in the feature space. The supervised model is said to be trained when a classifier, trained on the set of labeled documents, constructs a multi-plane that separates the two classes of documents with reasonable degree of error.

The use of the vector space model for document sentiment classification was explored in the work by Sarvabhotla *et al.* [Sarvabhotla 2011]. They compose two vectors for representing each document, the first is based on calculation of the average document frequency, while the second is built using the average subjective measure. They retain terms with higher than average document frequency and subjective measure. Basically, they statistically analyze frequencies in different collections (opinionated vs. objective) with the use of the Jaccard similarity measure for the final scoring. For the feature selection they apply Mutual Information and Fisher Discrimination Ratio and then train the

SVM model. Experiments were carried out on different portions of the movie reviews and show amelioration in performance in comparison to other feature weighting techniques using SVM classifier.

Paltoglou *et al.* [Paltoglou 2010] explore IR weighting measures on publicly available movie review datasets. They have good performance with BM25 and smoothing, showing that it is important to use term weighting functions that scale sublineary in relation to a number of times a term occurs in the document. They underline that the document frequency smoothing is a significant factor.

An improvement of one of the IR metrics is proposed in [Martineau 2009]. The so-called "Delta TFIDF" metric is used as a weighting scheme for features. This metric takes into account how the words are distributed in the positive vs. negative training corpora. As a classifier, they use SVM on the movie review corpus achieving an accuracy of 88%.

Church *et al.* [Church 1991] explore mutual information gain, t-test, scale statistics (mean and variance), POS tagging and phrase parsing to give an analysis of word collocations in specific grammar structures (subordinate conjunctures, examples of specific word associations). All of this relies heavily on language-specific preprocessing tools and heuristics. They investigate the notion of polysemy, when one word can have several meanings. Overall they try to give an overview of an opinion detection system which would choose a statistic, pre-process the corpus and select appropriate text unit for SA. They argue that mutual information gain is better for finding similarities between word pairs, whereas t-test is a good measure for detecting differences between synonyms. One of the main arguments is that the use of one single approach is not enough in order to achieve good effectiveness. They claim that such methods as SVM, naïve Bayes or HMM should be considered not alone but coupled with other heuristics.

There has been a trend in applying language models for opinion detection task [Hu 2007, Lavrenko 2001]. Usually, language models are trained on the labeled data and as an output they give probabilities of classified terms belonging to the class. Eguchi *et al.* [Eguchi 2006] use language models for both sentiment and topic classification. Hu *et al.* [Hu 2007] use language models with different smoothing techniques, but do not achieve good performance. Possible reasons could include the overfitting to the training set or overestimation of the importance of frequencies of terms in text for opinion detection.

2.5 Summary and Discussion

This chapter reviewed different sentiment analysis approaches from literature. The main distinction was drawn between the approaches that use labeled data (supervised) and lexicon-based methods (unsupervised).

Overall, the lexicon-based approaches can be subdivided into two groups. The first is based on semantic co-occurrence of words and their patterns from large corpora (or Internet), the second is based on the use of sentiment lexicons, such as WordNet or General Inquirer, to deduce semantic scores. The main drawback of the first group (dictionary-based approaches) is the difficulty to find domain-specific sentiment words and the cost of developing such external resources. On the other hand, the second group (corpus-based methods) solve this problem, but suffer from necessity of a large corpus, preferably in the same domain, time period and coming from the same region, in order to infer polarity of as many words as possible.

In general, it has to be noted that the performance of the text categorization methods and machine learning approaches is usually better than of the lexicon-based methods. This could be explained by the fact that the first are trained on the corpora having close relationship with the evaluation set. Therefore, models learned in one domain, e.g., movie reviews, do not usually

work well when applied to another domain, for example, product reviews. As a result, to obtain high accuracy using machine learning techniques it is necessary to train on the relevant data set. As a drawback of the supervised techniques one can also mention the lack of linguistic explanations on the weighting and importance of words in the model.

In their turn, sentiment lexicons and dictionaries constructed on general vocabulary (like WordNet) are domain independent and do not require prior training. Albeit, they suffer from poorer performance in comparison to machine-learning approaches. One of the reasons, as has been stated by Turney, "the whole is not necessarily the sum of the parts", meaning that knowing the semantic orientation of each word is not enough to infer the general sentiment of the phrase [Turney 2002]. Moreover, the way the lexicon scores are incorporated in the classification model can influence the obtained results. Most of the time, the score ranges, granularity and sometimes even the definition of subjectivity and opinion, differ across manually constructed lexicons.

Chapter 3

Experimental Setup and Corpora

3.1 Introduction

As follows from the discussions in previous chapters, there is a variety of different approaches to opinion and sentiment analysis. In this chapter, we give a general overview of the experimental setup, the main steps of the proposed approach, and the description of the testing corpora. We give the mathematical formulation to the problem of opinion and sentiment detection and discuss the choice of methods investigated in this thesis.

This chapter is organized as follows. Section 3.2 gives formal problem definition and describes the methodology used. In Section 3.3 we give an overview of the baselines in the SA domain. We explain the evaluation metrics in Section 3.4. The justification of the choice of benchmark datasets and the decisions made to facilitate the comparison of algorithms performance over different corpora is given in Section 3.5. Finally, Section 3.6 concludes the chapter.

3.2 Problem Formulation and Approach Description

In this section, we first present the justifications of the decisions taken when formulating problems considered in this thesis. Then, we provide the formal definition of the tasks under examination, followed by the model description and

illustration of the components of the proposed framework. We give an overview of the experiment setup, step by step, and the mathematical formulae to support the material presented later in this chapter.

3.2.1 Limitations of the Study

In this thesis, we evaluate and propose modifications to information measures used for the task of binary classification on a sentence level. In our opinion, the short number of features, that a sentence contains, represents an advantage for methods based on computation of some statistical scores. The longer the text, the more different features it contains, which is advantageous for supervised learning algorithms, like SVM, but presents more noise for the classification schemes based on statistical computation. Thus, we argue that the use of relatively simple classification schemes based on computation of information scores can provide comparable performance to the state-of-the-art methods on the sentence level.

We decided to consider the tasks of sentiment and opinion detection on the level of sentence for several reasons. First, most document-level SA methods partition the text into factual and opinionated sentences as the first step. On the other hand, opinion detection on a smaller scale (e.g., word) has been extensively researched and sentiment lexicons have been generated [Esuli 2006b, Wilson 2005a]. We chose the binary formulation of the classification task due to ambiguity in definition and judgment of "neutral" class sentences by the human experts themselves. Thus, if the annotators most often disagree on the labels "negative" and "neutral", it is unclear if the mistake of the system of assigning "neutral" class to a negative sentence is the same as assigning "negative" class to a positive sentence.

The choice of the natural languages of the tested corpora (English, Chinese, and Japanese) can be justified in the following way. There are many

resources and training corpora in English with well-studied results. In order to compare the performance of the proposed techniques in other languages we chose Chinese and Japanese, also due to the fact of the available corpora by the means of the NTCIR campaigns [Seki 2007, Seki 2008, Seki 2010].

3.2.2 Problem Formulation

Let $S = s_1, s_2, .., s_n$ be a collection of n sentences. Each of these sentences will be represented by a subset of all possible features $F = f_1, f_2, .., f_m$ that can appear in s_i. The example of features could be single words, like "papers" or word bigrams "if you", stemmed words, or sequences of *n-grams* of characters. For example, the word "although" would produce the following character *n-grams* (features): "alth", "ltho", "thou", "houg", "ough". Let n_k^s be the number of occurrences of the feature f_k for $k \in 1..m$ and a given sentence s. Thus, every sentence s can be represented as a vector in the feature space in the following way $\vec{s} = (n_1^s, n_2^s, .., n_m^s)$. Let us have a set of classes $C = \{c_1, c_2\}$, where c_1 is positive or opinionated and c_2 is negative or factual class, depending on the task. Our goal is to assign every sentence $s \in S$ to a specific class from C. The following tasks need to be solved:

Opinion classification: Determine if the sentence is subjective, i.e. contains an opinion.

Sentiment classification: Determine the polarity of the subjective sentence, positive or negative.

In our framework we assume that the sentence expresses a single opinion, and an opinion polarity if it is opinionated. However, on practice it is not usually so. For example, compare the two sentences:

> a. Just the labor involved in creating the layered richness of the imagery in this chiaroscuro of madness and light is astonishing.
> b. While the performances are often engaging, this loose collection of largely improvised numbers would probably have worked better as a one-hour tv documentary.

While the first sentence contains a single opinion, the second sentence contains two opinions of opposite polarity. Wilson *et al.* [Wilson 2005a] also pointed out that a single sentence may contain not only different opinions, but also factual clauses. The annotation on the clause level has been performed for the NTCIR collection. However, the evaluation campaigns use three sentiment polarity categories. In Section 3.5, we justify our decisions on how to consider these three categories used in the NTCIR corpus for the binary sentiment classification task.

3.2.3 Overview of the Proposed Approach

Detection of opinionated sentences or their sentiment polarity can benefit not only from the use of the supervised methods, but also from external sentiment lexicons in the analyzed natural language. We propose a model that includes several stages from representing the document collection in the model feature space to the use of simple to more sophisticated techniques to classify sentences. Our aim was to investigate whether it is possible to use light feature weighting schemes together with available sentiment lexicons and achieve performance comparable to supervised methods, like SVM and naïve Bayes. The proposed approach computes subjectivity or polarity orientation in the following stages:

1. First, we parse the sentence to construct a set of features that would represent this sentence in our model. Here, different stemming and parsing techniques are experimented with to determine the best features. We also evaluate the use of stop word lists.

2. In the second stage, we assign weights to features according to their importance in the specific class. We use different statistical weighting schemes, such as mutual information gain, Z score, Kullback-Leibler divergence.

3. In the third stage, we experiment with score modification and combination with the scores of sentiment lexicons to boost the weights of the features obtained in the previous stage.

4. In the fourth stage, we propose the use of the classification models based on the computed feature scores and statistics, taking into account dependency between the features occurring in the sentence.

5. At last, the system outputs its decision on the sentence subjectivity or sentiment polarity.

In the Figure 3.1 we illustrate the components of the proposed framework for sentiment and opinion classification.

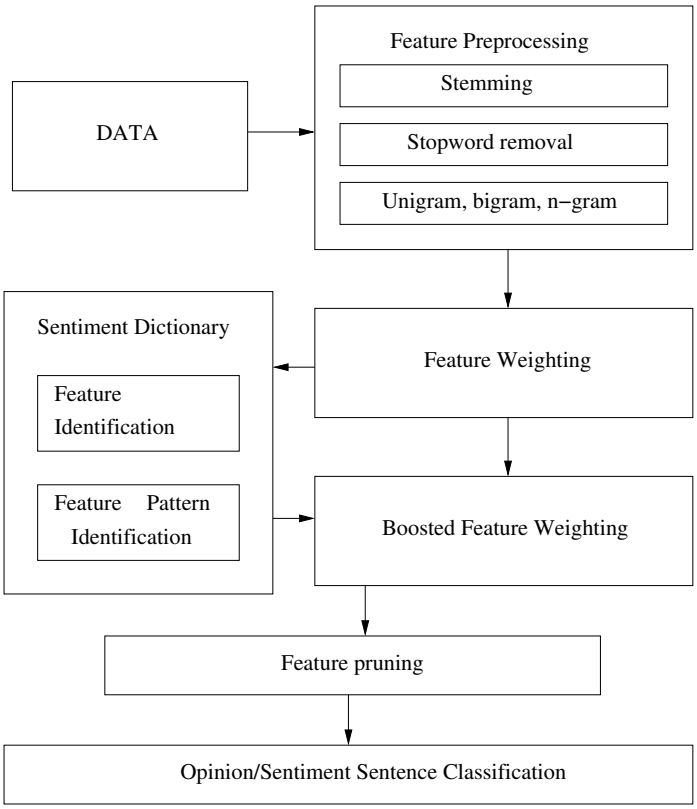

Figure 3.1: Overview of the experimental framework.

In the course of the work on this thesis we carried out a number of experiments where we compared the performance of the proposed classification models with other baselines. Most common baselines, commonly used in the field of SA, are naïve Bayes and SVM. We give their overview in the next section.

3.3 Baselines

In [Zubaryeva 2010a], we conducted several experiments on evaluation of different classification models for opinion detection. We compared our method principally to the two baselines that have shown to be effective for text categorization tasks: naïve Bayes and SVM.

3.3.1 Naïve Bayes

Naïve Bayes is a supervised learning method that possesses several important properties. First, it is stable in terms of the size of the training set, which is advantageous when a small training corpora is available. Also, it is easily extensible with the addition or removal of training examples [Wu 2009]. Using the notation introduced in Section 3.2 we assign a class C to the sentence s:

$$P(C|s) = \frac{P(C)P(s|C)}{P(s)} \tag{3.1}$$

To estimate $P(s|C)$ according to the assumption in naïve Bayes method, we assume that the features f_j are independent. The class of the unseen test sentence is predicted in the following way:

$$P(C|s) = \arg\max_C \frac{P(C)\prod_{j=1}^{m} P(f_j|C)^{n_j^s}}{P(s)} \tag{3.2}$$

In order to solve this problem we do not need to compute $P(s)$. In the training phase we need to obtain an estimate of $P(C)$ and $P(f_j|C)$. In order to account for the sparseness of the training data set a smoothing procedure is used. Since the estimate of frequency of the rare events is difficult to obtain from the limited training data, we use add-one or Laplace smoothing, that simply adds one to each count of feature f_j in calculating the prior probability for the occurrence of f_j. One clear drawback of the naïve Bayes algorithm is the independence assumption of features in text. This obviously does not hold in the real-world life. Nevertheless, naïve Bayes tends to perform well in the text classification problems.

3.3.2 SVM

A more sophisticated algorithm is Support Vector Machines (SVM) that usually tends to outperform naïve Bayes in text classification problems [Joachims 1998]. The idea of this method is to find a hyperplane $\vec{h}$ with the largest separation margin, so that the documents belonging to one class are separated from the documents belonging to another (two class problem). In other words:

$$\vec{h} = \sum_{i=1}^{n} \varphi_i C_i \vec{s_i}, \tag{3.3}$$

where $\varphi_i \geq 0$ and are found when solving the dual optimization problem. On the testing phase we just need to verify to which side of the hyperplane $\vec{h}$ the document belongs. In our experiments we use the SVM$^{\text{light}}$ package[1] with the default parameters. As one of the drawbacks of the SVM method, one can mention the limitation to only two class strategies and no possibility for the clear understanding of the system decision.

3.4 Evaluation in SA

The evaluation of SA systems is usually done experimentally. In order to be able to do analytical evaluation, one would need the formal specification of the notion of opinion and sentiment in text and the definition of the correctness and completeness of the problem that is solved. Basically, experimental evaluation gives an estimate of the classifier effectiveness and performance on the testing data set. It provides a possibility to statistically compare performance of different classifiers.

The supervised approaches that are discussed in this thesis use annotated data sets. In order to achieve the best approximation of the algorithm performance on the real data, 10-fold cross-validation is used. In this setup, the annotated data set is divided in ten equal parts. After this, the classifier is

[1] http://svmlight.joachims.org

trained ten times, each time on nine different parts, being tested on the left out part on every run. The performance is averaged over the ten runs.

3.4.1 Precision, Recall, F-measure and Accuracy

Evaluation metrics commonly used in SA have been adopted from the IR domain and describe the effectiveness of the system performance in the following way. *Precision* is the ratio of correctly classified documents out of the all documents that have been classified by the system. *Recall* is the ratio of the documents that were correctly identified with the documents manually annotated by a human expert. The formulas are the following:

$$Precision = \frac{TP_i}{TP_i + FP_i} \quad Recall = \frac{TP_i}{TP_i + FN_i} \tag{3.4}$$

where TP_i is the number of documents correctly classified to class i by the system, FP_i is the number of documents that were falsely classified by the system as belonging to class i, and FN_i is the number of documents that belong to class i, but were not correctly classified. Since we are dealing with the binary classification task, we compute the true positive (TP), false positive (FP), and false negative (FN) values over the two classes c_1 and c_2. In this context, the traditional IR precision gives an estimate of how well the system correctly classifies documents within both classes. Using the formulas (3.4) we can derive *F-measure*, a weighted mean of the precision and recall:

$$F_\beta = \frac{(1 + \beta^2) \cdot Precision \cdot Recall}{\beta^2 \cdot Precision + Recall} \tag{3.5}$$

The values of the *F-measure* lie in the interval $[0, 1]$. The higher the *F-measure*, the better the classification accuracy. In the equation (3.5), if $\beta = 1$ precision and recall are weighted evenly to account equally for both evaluation metrics. Then, it is called *F1-measure*:

$$F_1 = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall} \tag{3.6}$$

It is possible to give more weight to one of the metrics, for example precision. This is a usual practice when dealing with web application retrieval and

classification. Users do not go through all classified or retrieved documents on the Web, since there could be thousands of relevant items. In this context it is far more important for a system to output the most relevant and correct items on the top of the list.

Another important measure, widely used in SA, is *accuracy*. It takes into account the effectiveness of the classification in both classes (positive and negative, or opinionated and factual):

$$A = \frac{TP + TN}{N} \times 100, \tag{3.7}$$

where N is the number of all documents. In our experiments, we use *accuracy* for balanced and macro-averaged *F1-measure* for unbalanced datasets.

3.5 Test Environment and Datasets

One of the challenges of the traditional text categorization methods is the difficulty in finding annotated or labeled documents to train on. Usually, the sites with reviews, such as movie reviews, product reviews, provide a possibility for users to rate a movie on a scale from one to five stars. Then, the star rating is used as the grade of positivity/negativity of the review. The drawback of such setup is that the scale system does not provide a clear difference between 4 and 5, or 3 and 4 stars. Moreover, studies suggest that some users grade severely, while others tend to give grades mostly on the higher scale [Pang 2008]. Opinion reviews that have a star (scale) rating can be found usually in the domains of movie, book and product reviews.

Since movie reviews constitute a specific domain with particular vocabulary and word use as has been shown in [Pang 2005, Pang 2004], we also use the annotated news corpora from the NTCIR campaigns [Seki 2007, Seki 2008, Seki 2010]. In addition to this corpora, an MPQA (Multi-Perspective Question Answering) news article corpus is also used in the experiments [Wilson 2005b]. To solve the

tasks introduced in Section 3.2 we use four datasets (two generated from the NTCIR corpora, Subjectivity and MPQA datasets) for opinion detection task, and two datasets (Movie Review and NTCIR) for sentiment detection.

Therefore, in our experiments we use six data sets, three out of which were manually constructed from the NTCIR MOAT evaluation campaigns [Seki 2007, Seki 2008, Seki 2010]. We discuss these datasets and the decisions made about their creation in Section 3.5.1. The other three datasets are MPQA, Movie Review and Subjectivity datasets. They are discussed in Sections 3.5.2- 3.5.4.

3.5.1 NTCIR

The National Institute of Informatics (NII) every 18 months organizes the NTCIR campaigns (NII Test Collection for Information Retrieval Systems). NTCIR-6 workshop introduced the multilingual opinion analysis task in English, Chinese (traditional and simplified) and Japanese. The choice of these three languages can facilitate the comparison of the approaches used by participating groups. Though Seki *et al.* [Seki 2007] note that the natural language differences make it difficult to infer direct comparison between the results of the evaluation campaign.

Since we have three collections pertaining to the NTCIR 6, 7 and 8 [Seki 2007, Seki 2008, Seki 2010] campaigns carried out over a span of four years, we decided to organize and formalize our experiments using all of them together. Taking into account the way each of the three corpora was annotated and judged we give an explanation on how we used and evaluated our performance on these collections. First of all, one should note that through the three campaigns there exists a number of differences and discrepancies in corpora annotation and opinion evaluation. Whereas in NTCIR-6 and 7 there were three annotators and two judgment metrics (lenient and strict), NTCIR-8 MOAT subtask has only two annotators. With the three annotators judging during the first

two campaigns, lenient and strict metrics were computed based on the number of annotators agreeing on the polarity of the sentence. By lenient standard at least two annotators should agree on the given judgment of a sentence, by the strict standard (all three) in order for it to be counted as an opinionated sentence.

In order to keep consistency in terms of the annotations and the evaluation standards, we conduct our experiments in the following way. For opinion detection task we use all three corpora with lenient and strict standards. As opposed to the previous two campaigns, these metrics are defined for the NTCIR-8 test collection in the following way: strict when both annotators agree, lenient otherwise. Therefore, if an opinion unit is judged as opinionated by one annotator and not opinionated by another, it would be not opinionated in the strict standard and opinionated in the lenient. If two annotators do not agree about the sentiment we assign, the sentiment label is taken from the first annotator in both metrics.

One can argue that this way we label sentences as opinionated with a weaker confidence (only one annotator agreed). The main reason for this decision here is the amount of training data. From the corpus statistics the number of sentences judged differently by both annotators (as opinionated and factual) is only 544, and those that were judged opinionated but assigned a different polarity (e.g., negative and neutral) - 358.

For the NTCIR 6 and 7 campaigns there is a number of opinion sentences that were also judged ambiguously: 765 sentences with three distinct judgments out of 4 categories (not opinionated, positive, negative, neutral). They were classified as not opinionated. In this situation, we could also consider attributing a sentence at least to one of the polarity classes. It must be noted, that the most common combination of judgments in this case is "no opinion","negative opinion", "neutral opinion" representing the three classification categories that are hard to distinguish one from another with possible nuances in their polarity

evaluation (394 sentences with such judgments).

Thus, for the task where we care about opinionatedness but not the sentiment expressed, we can expect less inter-annotator disagreement for the NTCIR-8, and we can use both lenient and strict standards. In our experiments where we classify the sentiment (positive vs. negative) we use the strict standard. Thus, both annotators should agree on the polarity of the sentiment. We reserve to strict and lenient standards in both of the experimental setups that we investigate. For the opinion detection task we use the NTCIR 6, 7 and 8 corpora where we consider a sentence to be opinionated in a strict standard if all three judges agree that there is an opinion, without polarity distinction, for the lenient standard two of the judges must agree. The strict and lenient sets of results are the same for the NTCIR-8 corpora since it was annotated by only two annotators.

For the sentiment analysis setup we consider only positive and negative sentiment polarities. We exclude the neutral polarity as a class since it is ambiguous to interpret and classify, even by human judges, not only by the system [Seki 2007]. Nevertheless, if we encountered a neutral (mixed) judgment with the positive or negative we classified it as positive or negative respectively in the lenient standard. In the strict standard for the sentiment analysis setup all judges have to agree on a sentiment.

Through all three campaigns the text unit changed from evaluation on the sentence to evaluation on the sentence clause, with both judged on their opinion and relevance to the topic. Thus, we decided to reserve to a sentence as an opinionated unit, since it is less ambiguous for testing and evaluation taking into account that there were 73 clauses in all three collections in the whole. The statistics on the three collections are given in the Table 3.1[2].

[2]The collections do not include the Yomiuri articles used in original corpora of the NTCIR campaigns due to user agreement.

Statistics	Setup	Opinion Detection	Polarity Detection
Opinionated/Positive sent.	strict	598	74
Factual/Negative sent.	strict	9548	270
Opinionated/Positive sent.	lenient	2466	679
Factual/Negative sent.	lenient	7680	1332
Total	strict	10146	344
Total	lenient	10146	2011

Table 3.1: Sentence statistics for NTCIR English corpora (Sans Yomiuri).

There is one document EN-9803202A14DM319 containing 20 sentences that was retrieved for two topics. We decided to exclude this document from one of the topics in the evaluation, since otherwise it would represent noise during the training and difficulties with evaluation with different judgments for two different topics. Furthermore, both topics had one of the biggest number of documents retrieved per topic, 20 each, while some other topics have relatively small number of documents retrieved.

Since we have quite a limited number of sentences in the strict standard for the sentiment detection task, we decided to conduct experiments only on the lenient standard. In the lenient standard for the opinion detection task, there are 51 sentences that contain only punctuation marks, marking the end of the article but being annotated nevertheless in the collection and in the evaluation files. Here are some examples of sentences from the NTCIR corpora:

⟨DOC⟩⟨DOCID⟩KT2001_01987.0008.E⟨/DOCID⟩
⟨OP⟩n⟨/OP⟩
Hanvit Bank said that transactions at five branches in New York were halted right after the terrorist attack, but the resumption of business will be possible in a couple of days.
⟨/DOC⟩ ⟨DOC⟩⟨DOCID⟩st2001_048063.0034.E⟨/DOCID⟩
⟨OP⟩n⟨/OP⟩

> Its human resource manager, Mr Joseph Chia, said: "We spoke about it before, but never really thought about it seriously until Sept 11."
> ⟨/DOC⟩

The last sentence is not opinionated, but we clearly see the quotation marks, together with the word *said* that indicates direct speech and possible opinionated sentence. It is important to note that the opinionatedness of the sentence in the NTCIR corpora is defined only on the sentences relevant to the topics. Hence, NTCIR collection represents another level of difficulty of matching the relevance of the sentence to the topic.

For NTCIR dataset we parsed the topics from the three years of NTCIR campaigns, eliminating stop words from the keyword and description fields of the topics. Since sentence subjectivity is dependent whether or not the sentence is relevant to the topic in the NTCIR campaign, it is much less straightforward for experiments and evaluation to determine subjective sentences. For example, there are sentences that include quotations or expressions of opinion, that were not classified as subjective, since they are not relevant to the topic.

Therefore, we conducted experiments where a sentence was considered subjective if its relevant to the topic, i.e. relevant to the queries consisting of pre-processed description and key fields of the topic, and is considered subjective by the classifier. It is important to pay attention to the size of the training corpora. Since we have much less opinionated sentences, it is much easier to make a classification error when relevance of the sentence is referred from the context and has no or very little query terms.

To conclude, we have created three datasets from the annotated corpora used in the NTCIR campaigns. First, a dataset with positive and negative sentences, denoted as NTCIR SA. Second, a dataset with opinionated and factual sentences according to lenient standard, denoted as NTCIR OP len. Last, similar to previous dataset but divided into two classes according to strict

Example of topics
⟨TOPIC⟩ ⟨NUM⟩12⟨/NUM⟩ ⟨SLANG⟩KR⟨/SLANG⟩ ⟨TLANG⟩EN⟨/TLANG⟩ ⟨TITLE⟩Tiger Woods, sports star⟨/TITLE⟩ ⟨DESC⟩Find documents about sports media or related enterprises recognizing Tiger Woods as a sports star.⟨/DESC⟩ ⟨NARR⟩During his four full years on the PGA Tour, Tiger Woods (25) was voted athlete of the year for the 3rd time. Sportsmen's changing views of golf was the reason he won their votes. Documents about sports magazines or enterprises recognizing Tiger Woods as a sports star based on his record, skills or contribution to marketing are relevant. Documents about Tiger Woods' daily life or celebrity news outside of golf are irrelevant.⟨/NARR⟩ ⟨CONC⟩Tiger Woods, golf, golf genius, PGA⟨/CONC⟩ ⟨/TOPIC⟩
⟨TOPICDESCRIPTION⟩ ⟨TOPIC⟩N11⟨/TOPIC⟩ ⟨TITLE⟩What is the relationship between AOL and Netscape?⟨/TITLE⟩ ⟨NARRATIVE⟩I would like to know about the relationship between AOL and Netscape, background information, and their influence on each other after their merger.⟨/NARRATIVE⟩ ⟨/TOPICDESCRIPTION⟩
⟨TOPIC⟩ ⟨NUM⟩N01⟨/NUM⟩ ⟨TITLE⟩Euro⟨/TITLE⟩ ⟨QUESTION⟩What negative prospects were discussed about the Euro when it was introduced in January of 2002?⟨/QUESTION⟩ ⟨POLARITY⟩Negative⟨/POLARITY⟩ ⟨OPTYPE⟩perspective controversy⟨/OPTYPE⟩ ⟨CONC⟩Euro introduction⟨/CONC⟩ ⟨PERIOD⟩2002-01⟨/PERIOD⟩ ⟨/TOPIC⟩

Table 3.2: Example of topics from NTCIR campaigns.

standard, and referred to as NTCIR OP str.

3.5.2 MPQA

As a second dataset that contains newspaper articles we use the MPQA dataset[3] [Wilson 2005b]. The documents are manually annotated for opinions and other private states, such as beliefs, emotions, sentiments, speculations, etc. The problem with the MPQA dataset is that the annotation unit is at the phrase level, which could be a word, part of a sentence, a clause, a sentence itself, or a long phrase.

In order to be able to compare the performance of different methods on a similar domain as the NTCIR dataset and using a similar text granularity, we parsed the MPQA sentences into two classes: opinionated and factual. To infer the opinionatedness of a sentence unit we used the approach proposed by Wilson *et al.* [Wilson 2005b]. They define subjective expression as any word or phrase conveying an opinion, emotion, speculation, etc. They define the sentence-level opinion classification in terms of the phrase-level annotations. A sentence is considered opinionated if:

1. It contains a "GATE_direct-subjective" annotation with the attribute intensity not in ['low', 'neutral'] and not with the attribute 'insubstantial';

2. The sentence contains a "GATE_expressive-subjectivity" annotation with attribute intensity not in ['low'].

Here is the information on corpus statistics as reported in [Wilson 2005b]: there are 15,991 subjective expressions from 425 documents, containing 8,984 sentences. Most of the documents cover ten different topics. Additionally, a number of articles were randomly selected from a larger corpus of 270,000 documents. Five trained annotators performed the annotation of the original subset over a period

[3] http://www.cs.pitt.edu/mpqa/

of 15 months [Wilson 2005b]. Here are the examples of opinionated and factual sentences from the parsed MPQA corpus:

Opinionated

- This can hardly be described as ambitious as a 15 percent increase in energy efficiency was in any event expected given present productivity levels.
- The warnings by climate experts went unheeded in the US climate protection program.

Factual

- The bulk of the weapons have arrived from the DRC in that country's military cargo plane.
- There is also a gate between Cuba proper and the base that is sometimes opened for meetings between military commanders, or to repatriate Cubans who have been taken to the base.

In order to unify the experimental setup and facilitate closer comparison of algorithms performance over the datasets we consider a sentence as a unit of evaluation. Thus, we could have a more direct comparison with the NTCIR corpus. Only 25% of sentences contain only one "subjective expression", 47% two or more, and the rest has none. We propose the following scheme in our evaluation setup: consider sentences with two or more subjectivity expressions, and sentences containing one subjectivity expression longer than a word as opinionated sentences. Thus, we would exclude those sentences that could contain one subjective word, since it is more arguable if the whole sentence could be considered subjective.

3.5.3 Movie Review Dataset

In our study we use Sentence Polarity dataset v1.0[4] [Pang 2005], here referred to as Movie Review dataset. It contains 5331 positive and 5331 negative snippets of movie reviews. The dataset was gathered from RottenTomatoes[5] and the authors

[4] http://www.cs.cornell.edu/people/pabo/movie-review-data

[5] http://www.rottentomatoes.com/

assumed that snippets for reviews marked with "fresh" are positive, and those for reviews marked with "rotten" are negative. Some examples of positive and negative reviews:

Positive

- The film makes a strong case for the importance of the musicians in creating the motown sound.
- With one exception, every blighter in this particular south london housing project digs into dysfunction like it's a big, comforting jar of marmite, to be slathered on crackers and served as a feast of bleakness.
- You walk out of the good girl with mixed emotions - disapproval of Justine combined with a tinge of understanding for her actions.
- A somewhat crudely constructed but gripping, questing look at a person so racked with self-loathing, he becomes an enemy to his own race.

Negative

- What you expect is just what you get... assuming the bar of expectations hasn't been raised above sixth-grade height.
- What you end up getting is the vertical limit of surfing movies - memorable stunts with lots of downtime in between.
- Every potential twist is telegraphed well in advance, every performance respectably muted; the movie itself seems to have been made under the influence of rohypnol.
- If you go into the theater expecting a scary, action-packed chiller, you might soon be looking for a sign, an exit sign, that is.

As you can see, the movie reviews use a lot of metaphor, comparison, sometimes unclear language or reference to other movies or situations. Sometimes, a review has no overtly negative terms, which is hard to classify only using sentiment lexicons.

3.5.4 Subjectivity Dataset

The Subjectivity dataset contains 5000 subjective and 5000 objective sentences. The data was gathered from the Rotten Tomatoes website and plot summaries for movies from the Internet Movie Database[6]. They assume that all snippets retrieved from the Rotten Tomatoes are subjective and all plot summaries are objective, although some plot summaries can contain subjective expressions [Pang 2004]. Examples of objective and subjective sentences:

Plot

- Emerging from the human psyche and showing characteristics of abstract expressionism, minimalism and Russian constructivism, graffiti removal has secured its place in the history of modern art while being created by artists who are unconscious of their artistic achievements.
- As Jake and Sam Bicker throughout their trek across majestic India, they find Kabir, a museum curator who possesses an artifact which he claims can produce the musical tone required to open the door to the temple.

Review

- From the opening scenes, it's clear that "All about the Benjamins" is a totally formulaic movie.
- A coming-of-age tale from New Zealand whose boozy, languid air is balanced by a rich visual clarity and deeply felt performances across the board.

For this dataset only sentences and snippets more than ten words long were extracted [Pang 2004]. As well as with Movie Review and MPQA datasets described in the previous sections, we chose this dataset because of its popularity as a benchmark in SA research. Another reason for choosing the movie review domain is its difficulty in opinion detection and classification task.

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

3.6 Summary and Discussion

In this chapter, we first give our justification of the choices made when formulating the problem of the study. We restrict the scope of our study to the specific case of binary opinion and sentiment classification on a sentence level. This way, we eliminate the ambiguity of interpretation of the "neutral" sentiment polarity and the degree of its strength. At the same time, these constraints help obtain more reliable evaluation of the classification models.

The proposed framework is designed to solve two problems - opinion and sentiment polarity classification. In order to do so, we adopt an approach consisting of a series of steps. They include document pre-processing, feature representation and weighting, as well as the possibility to extend the model with the use of external sentiment dictionaries and lists. We discuss them more thoroughly in the chapters to follow.

After presenting the main stages of the classification approach adopted in this work, we gave an overview of the two methods, naïve Bayes and SVM. They are commonly adopted as baselines not only in opinion mining, but also in text classification domains. In the next chapter, we discuss our implementation and comparative analysis of these baselines and the proposed method.

In order to compare research methods it is important to apply standard evaluation metrics and have a set of evaluation corpora available. Therefore, we introduced the evaluation metrics, accuracy, precision, recall and F-measure, commonly accepted in the opinion mining domain. We also gave an overview of the benchmark datasets used in the evaluation. Three of these datasets were constructed from the last three NTCIR campaigns. Thus, we described the decisions that were taken when combining the three releases, as well as the changes in annotations and peculiarities of the data. NTCIR corpora, together with the MPQA dataset, consists of the newspaper articles pertaining to different topics. Movie Review and Subjectivity datasets contain movie reviews and plot

summaries (Subjectivity dataset). The choice of these benchmarks is due to the possibility to compare performance of the proposed approaches on different text collections within one domain, as well as within different domains.

Chapter 4

Feature Weighting and Z Score Classification Scheme

4.1 Introduction

The previous chapter introduced the general approach, experimental setup, and the corpora used to solve the task of sentiment and opinion detection on a sentence level. As discussed earlier, there is a multitude of decisions on text pre-processing and its representation in the model that a researcher has to make. The choices made on this stage can influence the results of the classification algorithms used.

In this chapter, we describe different variants of the text pre-processing setups that we consider in our study. The pre-processing procedures are applied to represent words as features in the model. Thus, they include unifying plural forms to singular, using case normalization, stemming and other techniques. Besides, the traditional consideration of unigram or bigram feature representation, stop word removal, and stemming, we present a new scheme for feature tokenization that we called *Wise Tokenizer* scheme.

After presenting our approach to feature representation, we describe the experiments on opinion and sentiment classification with the use of the Z score information measure and logistic regression. These experiments follow our consecutive participation in the NTCIR-7 and NTCIR-8 campaigns [Zubaryeva 2008, Zubaryeva 2010b]. In the NTCIR-8 evaluation

campaign we presented a modification of the initial approach achieving the best performance for the English language among the participating teams [Zubaryeva 2010b].

Next, we present the comparative evaluation of the baselines, naïve Bayes and SVM, with the classification model based on the Z score computation [Zubaryeva 2010a]. In this part of the study, we experiment on the joint NTCIR 6 and 7 newspaper corpora. We give the experiment results using different types of pre-processing setups, as well as variations on the smoothing parameters for naïve Bayes and the cost parameter for the SVM model. We also provide misclassification analysis on the example sentences from the dataset.

Finally, we present our adaptation of the Kullback-Leibler (KL) divergence score for calculating the "distance" of a feature to a category. In order to evaluate the performance of the KL score and other information measures, we describe the results of a classification on the six test corpora presented in Chapter 3. We analyze the results, based on their sensitivity to the size of the training set.

Thus, the remainder of this chapter is organized as follows. First we present the pre-processing setups for the analyzed datasets in Section 4.2. Section 4.3 describes the feature weighting framework. Sections 4.4 and 4.5 introduce the proposed model based on Z score and the improvements that we made for the NTCIR-8 campaign. We propose the comparison with the baseline models in Section 4.6. In Section 4.7, we present an adaptation of the Kullback-Leibler divergence metric to the sentiment and opinion classification task and compare the performance of several information association measures on the tested corpora. Finally, Section 4.8 concludes this chapter with summary and discussion.

4.2 Text Preprocessing Setups

In this section, we first introduce a model of feature categorization based on the feature relevance to the domain of the text. Afterwards, we analyze the details of the pre-processing setups and feature generation techniques.

Text features are usually classified into two categories depending on their specificity to the text. Thus, content-specific features are words or phrases with the particular meaning or pertinence to the domain. This could be terminology or jargon, specific to the topic or domain of the document. In the corpora from the movie review domain, content-specific features could be *movie*, *script*, *actor*, *oscar*, in product reviews - *sale*, *fast*, *light*, *functionality*. These type of features are easily defined within the documents of the same genre or on the same topic. The overview of the considered features is given in Figure 4.1.

Content-free features, as opposed to content-specific, can be found in any texts. They usually are comprised of lexical, syntactic and structural [Argamon 2010] terms. Since we are dealing with sentences, we do not consider the structural features, like paragraph or sentence length, text formatting, document structure. Lexical features represent diverse vocabulary from text. In the simplest form, these words comprise the semantic information in text. Features that do not carry any semantic information are called *functional features*. These could be prepositions, conjunctions, articles, identifiers, determiners and pronouns. These parts of speech are present in every text and do not carry useful meaning by themselves. This classification of features will be of use to us when we will analyze the performance of different classification approaches later.

Generally, text pre-processing is comprised of several steps that we can observe in Figure 4.2. It is not necessary to do all the steps, some subset more adapted for the task at hand can be chosen. First, the textual tokenization is performed: tokens or words are extracted from the sentence. As a next step POS tagging is usually done. In our research we tried to carry out experiments omitting this step

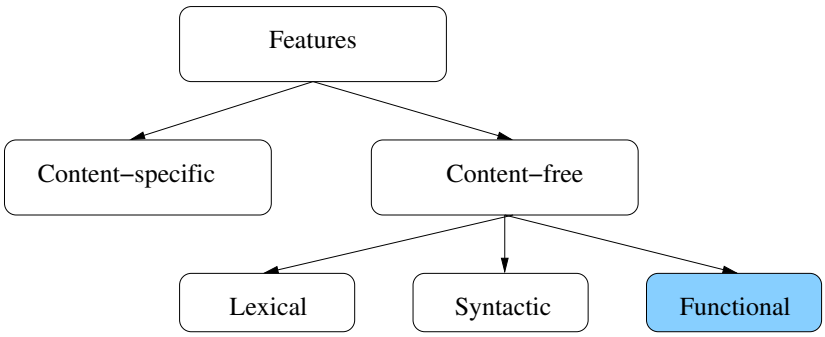

Figure 4.1: Feature classification used for sentence-level sentiment analysis.

in attempt to make the approach adaptable to texts in different natural languages.

As a next step, all the words are transformed to a lower case, excluding abbreviations. Of course, we cannot fully rely that these procedures will not make any error. Thus, the system would need a more sophisticated analysis, possibly with POS tagger, in order to take a correct decision, for example, on whether *IT* is an abbreviation in the sentence or not. Then we perform the elimination of the stop words. In this work, we define a small set of stop words that occurs in almost every sentence and text and does not carry any semantic value. These stop words are *a, an, and, at, by, for, in, of, on, that, the, to, too*. From the feature classification categories discussed above, these are functional features.

Next, we carry out a stemming procedure. In our experiments we use Porter stemmer [Porter 1997]. As one of the possible pre-processing setups we also consider the character *n-gram* tokenization of sentences. Thus, we use $n = 4$ as it has been shown to capture the essential information [Pang 2008].

As another tokenization scheme, we experiment with the bigrams, where we store features consisting of two tokens that follow each other. If this scheme is combined with stop word removal, the extraction of bigrams was done after eliminating words from the stop list. Intuitively, it helped capture phrases

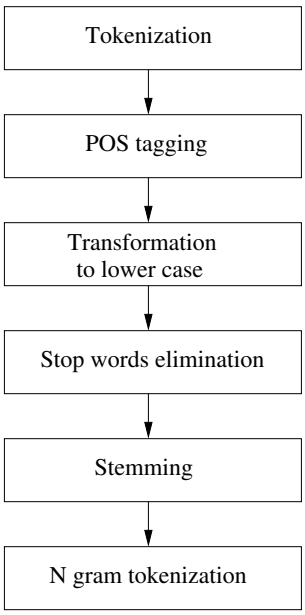

Figure 4.2: Text pre-processing steps.

consisting of two words with a preposition or an article in between that otherwise would not be stored as a feature, for example *now and then*, *put an end*, *offer an advice*, *attempt to do*, etc.

The experiments carried out evaluated different combination of the pre-processing steps: stemming, stop word removal, *n-gram* and bigram feature extraction. We always perform the tokenization and conversion to lower case. The aim is to find the best setup for capturing content-specific and lexical features with the proposed classification scheme, at the same time not using any language-dependent tools or techniques.

Feature selection in other natural languages than English is not a straightforward task. If we consider German the words can be compounds of several words carrying a semantic meaning that would be translated as an expression or a phrase in the English language. Russian has a complex case system for a nouns,

verbs, adjectives, as well as a system of suffixes for the nouns that can change the tone of the message expressed. Chinese and Japanese need text segmentation in order to capture the meaning of the tokens, since many times the ideogram can be a word by itself or part of other words. It is obvious that usually European languages are closer to English in terms of pre-processing setups and techniques used. Nevertheless, the particularities such as stop words and stemming need to be carefully considered for each of the natural languages in order to achieve higher effectiveness of the classification result.

4.2.1 Wise Tokenizer

Previously, we covered the common techniques for text-preprocessing. After analyzing the common errors for the specific text classification schemes, we remarked that it is possible to create a hybrid scheme between unigram and bigram tokenization.

We noticed that some of the prepositions combined with the previous term in the sentence can change its meaning and sometimes even its polarity. Consider the following examples: *take* and *take off*, *put* and *put up*. Therefore, together with the experiments with unigram and bigram representation we implemented a new indexing scheme called *Wise Tokenizer*.

This new text representation is obtained using the following procedure. All terms in the sentence are indexed separately, except terms that precede the prepositions that could change the meaning of the verb. These are the prepositions: *about, across, after, along, around, as, aside, at, away, back, behind, by, down, for, forth, front, in, into, it, near, of, off, on, out, outside, over, through, to, together, under, up, upon, with.*

Consider the following examples: *kick* and *kick out*, *put* and *put off*, *step* and *step up*. The preposition changes the meaning and the sentiment information that a particular expression carries. At the same time the use of the bigram

scheme for all words in the sentence may turn out to be obsolete in lots of cases, when a targeted extraction of words and prepositions may have more sense in terms of the wholeness of the meaning conveyed.

4.3 Feature Selection Framework

Feature selection is a procedure that helps reduce the dimensionality of the feature space. First, this lowers the computational costs when solving the problem. Second, it produces usually a better classifier that can have a better generalization strategy. When solving text categorization problem it is important to find a good trade off between the richness of features representing a high dimensional space and computational constraints involved when solving the categorization task. This procedure allows to capture salient features of the training data, thus giving an intuitive understanding of the most important terms in the analyzed categories. Another important advantage of feature selection is its ability to reduce the over-fitting to the training dataset. Over-fitting occurs when the model trained performs very well on the training dataset, but poorly on the unseen new instances.

There have been several studies that evaluated the feature selection approaches. Forman [Forman 2003] has given an extensive evaluation of various schemes in topical text classification task. *Zheng et al.* [Zheng 2004] experiment with feature selection schemes on the imbalanced data, when one class possesses more documents than other.

The idea of selecting salient features for each category can affect the classifier performance in the subsequent step. Since we want to determine if a sentence belongs to opinionated or factual category (same for positive/negative), our aim is to choose terms or features that are unique or most representative in that category. Ideally, we would like to have a set of not overlapping features that represent documents in each category. Realistically, it is not possible. Thus,

it is important to find a way to represent the importance of each feature in both categories (or in all categories when faced with more than two classes).

In order to determine the features that can help distinguish between factual and opinionated sentences on one hand, and between their polarities (positive and negative) on the other, we have selected different tokens as described in the previous section. The goal is therefore to design a method capable of selecting terms that clearly belong to one type of polarity compared to other possibilities. The approaches that use words and their frequencies or distributions are usually based on a contingency table (as shown in Table 4.1 using the four information elements).

	S	$C-$	$C = S \cup C-$
w	a	b	$a+b$
$\overline{w}$	c	d	$c+d$
	$a+c$	$b+d$	$n = a+b+c+d$

Table 4.1: Contingency table.

In this table, the letter a represents the number of occurrences (tokens) of the word w in the document set S (corresponding to a subset of the larger corpus C). The letter b denotes the number of tokens of the same word w in the rest of the corpus (denoted $C-$) while $a+b$ is the total number of occurrences in the entire corpus (denoted C). Similarly, $a+c$ indicates the total number of tokens in S. The entire corpus C corresponds to the union of the subset S and $C-$, and therefore is $(C = S \cup C-)$ containing n tokens $(n = a+b+c+d)$.

Based on the MLE (Maximum Likelihood Estimation) principle the values shown in a contingency table can be used to estimate various probabilities. For example, we can calculate the probability of the occurrence of the word w in the entire corpus C as $Pr(w) = (a+b)/n$ or the probability of finding in C a word belonging to the set S as $Pr(S) = (a+c)/n$.

In the next section we present the classification method based on the Z score statistics and logistic regression.

4.4 Z Score and Logistic Regression

Now to define the discrimination power a term t_i, we suggest defining a weight attached to it according to Muller's method [Muller 1992]. We assume that the distribution of the number of tokens of the word t_i follows a Binomial distribution with the parameters $Pr(t_i)$ and n'. The parameter $Pr(t_i)$ represented the probability of occurrence of the word t_i in the corpus C. This probability could be estimated as $(a+b)/n$. If we repeat this drawing $n' = a+c$ times, we will have an estimate of the number of occurrences of the word included in the subset S by $Pr(t_i) \cdot n'$. On the other hand, Table4.1 gives also the number of observations of the word t_i in S, and this value is denoted by a. A large difference between a and the product $Pr(t_i) \cdot n'$ is clearly an indication that the presence of a occurrences of the term t_i corresponds to an intrinsic characteristic of the subset S compared to the subset $C-$. In order to obtain a clear rule, we suggest computing the Z score attached to each feature t_i. If the mean of a Binomial distribution is $Pr(t_i) \cdot n'$, its variance is $n' \cdot Pr(t_i) \cdot (1 - Pr(t_i))$. These two elements are needed to compute the standard score as described in Equation 4.1.

$$Z\ Score(t_i) = \frac{a - (n' \cdot Pr(t_i))}{\sqrt{n' \cdot Pr(t_i) \cdot (1 - Pr(t_i))}} \tag{4.1}$$

If we decide to rewrite this formula in terms of variables used in Table 4.1, we would get the following:

$$Z\ Score(t_i) = \frac{a - ((a+c) \cdot ((a+b)/n))}{\sqrt{(a+c) \cdot ((a+b)/n) \cdot (1 - ((a+b)/n))}} \tag{4.2}$$

Taking into account that $n = a + b + c + d$ and rewriting the formula 4.2, we would get

$$\begin{aligned} Z\ Score(t_i) &= \frac{(a+b+c+d)\cdot a - (a+c)\cdot(a+b)}{(a+b+c+d)\cdot\sqrt{\frac{(a+c)\cdot(a+b)\cdot(c+d)}{(a+b+c+d)^2}}} \\ &= \frac{a^2 + a\cdot b + a\cdot c + a\cdot d - (a^2 + a\cdot b + a\cdot c + c\cdot b)}{\sqrt{(a+c)\cdot(a+b)\cdot(c+d)}} \quad (4.3) \\ &= \frac{a\cdot d - c\cdot b}{\sqrt{(a+c)\cdot(a+b)\cdot(c+d)}} \quad (4.4) \end{aligned}$$

In the Table 4.1 we can observe the 20 highest Z scores for negative and positive categories for the Movie Review dataset after stemming, and unigram indexing scheme were performed. One can see that a lot of this features relate to the category where they predominate.

#	Op Feature	Z Score	Noop Feature	Z Score
1	express	2.66	weather	2.58
2	credible	2.54	10	2.47
3	sadden	2.54	el	2.29
4	believe	2.44	nino	2.14
5	humiliate	2.35	1998	2.13
6	argue	2.29	attend	2.08
7	greater	2.11	shimbun	2.07
8	notion	2.01	mainichi	2.05
9	excuse	2.01	1995	2.01
10	sadden	1.92	met	1.96

Table 4.2: Distribution of the 10 highest Z scores across opinionated and not opinionated categories in NTCIR OP lenient corpus.

As a decision rule we consider the words having a Z score between -2 and 2 as terms belonging to a common vocabulary, as compared to the reference corpus (as for example *will*, *with*, *many*, *friend*, or *forced* in our example). This threshold was chosen arbitrary. A word having a Z score > 2 would be considered

as overused (e.g., *that*, *should*, *must*, *not*, or *government* in MOAT NTCIR-6 English corpus), while a Z score $<$ -2 would be interpreted as an underused term (e.g., *police*, *cell*, *year*, *died*, or *according*). The arbitrary threshold limit of 2 corresponds to the limit of the standard normal distribution, allowing us to find around 5% of the observations (around 2.5% less than -2 and 2.5% greater than 2). As shown in Figure 4.3, the difference between our arbitrary limit of 2 (drawn in solid line) and the limits delimiting the 2.5% of the observations (dotted line) are rather close.

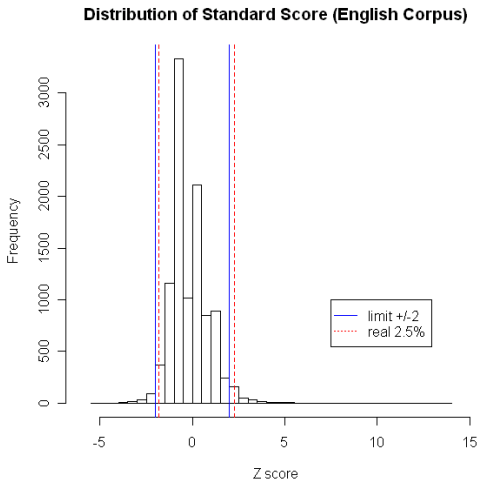

Figure 4.3: Distribution of the Z score (MOAT NTCIR-6 English corpus, opinionated).

Based on a training sample, we were able to compute the Z score for different words and retain only those having a large or small Z score value. Such a procedure is repeated for all classification categories (opinionated and factual). It is worth mentioning that such a general scheme may work with isolated words (as applied here) or *n-gram* (that could be a sequence of either characters or words), as well as with punctuations or other symbols (numbers, dollar sign), syntactic patterns (e.g., verb-adjective in comparative or superlative forms) or

other features (presence of proper names, hyper links, etc.)

When our system needs to determine the opinionatedness of a sentence, we first represent this sentence as a set of words. For each word, we can then retrieve the Z scores for each category. If all Z scores for all words are judged as belonging to the general vocabulary, our classification procedure selects the default category. If not, we may increase the weight associated with the corresponding category (e.g., for the opinionated class if the underlying term(s) is (are) overused in this category).

Such a simple additive process could be viewed as a first classification scheme, selecting the class having the highest score after enumerating all words occurring in a sentence. This approach assumes that the word order does not have any impact. We also assume that each sentence has a similar length.

For this model, we can define two variables, namely #SumOP indicating the sum of the Z score of terms overused in opinionated class (i.e. Z score $>$ 2) and appearing in the input sentence. Similarly, we can define SumNOOP for the other class. However, a large SumOP value can be obtained by a single word or by a set of two (or more) words. Thus, it could be useful to consider also the number of words (features) that are overused (or underused) in a sentence. Therefore, we can define #OpOver indicated the number of terms in the evaluated sentence that tends to be overused in opinionated documents (i.e. Z score $>$ 2) while #OpUnder indicated the number of terms that tends to be underused in the class of opinionated documents (i.e. Z score $<$ -2). Similarly, we can define the variables #NoopOver, #NoopUnder, but for the not opinionated category. With these additional explanatory variables, we can compute the corresponding subjectivity score Op_score for each sentence s_i as follows:

$$Op\_score(s_i) = \frac{\#OpOver(s_i)}{\#OpOver(s_i) + \#OpUnder(s_i)} \tag{4.5}$$

As a better way to combine different judgments we suggest following [Calvé 2000] and normalize the scores using the logistic regression. The logistic transformation $\pi(\mathbf{x})$ given by each logistic regression model is defined as:

$$\pi(\mathbf{x}) = \frac{e^{\beta_0+\sum_{i=1}^{k}\beta_i x_i}}{1+e^{\beta_0+\sum_{i=1}^{k}\beta_i x_i}} \tag{4.6}$$

where β_i are the coefficients obtained from the fitting, x_i are the variables, k is the number of explanatory variables, and $\mathbf{x}$ is the vector of all x_i, $i = 1..k$ variables. Thus, $\mathbf{x}$ could contain as elements Op_score, $SumOP$, $SumNOOP$, $OpOver$, $OpUnder$, their logarithms and logarithmic expressions normalized by the number of features with these scores. These coefficients reflect the relative importance of each variable in the final score.

For each sentence, we can compute the $\pi(\mathbf{x})$ corresponding to the two possible categories and the final decision is simply to classify the sentence according to the $max\,\pi(\mathbf{x})$ value. This approach takes account of the fact that some explanatory variables may have more importance than other in assigning the correct category.

4.5 Improvements of the Z Score Classification Model

On the described classification model in the Section 4.4 we made several improvements that are described in [Zubaryeva 2010b]. The experiments in the NTCIR-8 were also performed on the newspaper corpora. The implemented modifications to the original approach allowed us to achieve the best result in English MOAT subtask [Seki 2010, Zubaryeva 2010b].

The initial model based on the logistic regression is decomposed in two steps. This is so called cascade model that performs the classification in the following way. First, the system classifies the sentences in two categories:

opinionated and factual. In order to realize this, we first train our model on factual and opinionated part of training corpora (positive, negative and neutral sentences). This way we avoid partitioning the training data into four classes as in our previous work [Zubaryeva 2008]. Then, within the sentences classified as opinionated, the polarity detection was performed in a second stage. This approach allows to somewhat compensate for the small size of the positive, negative and neutral training sentences available from previous NTCIR campaigns.

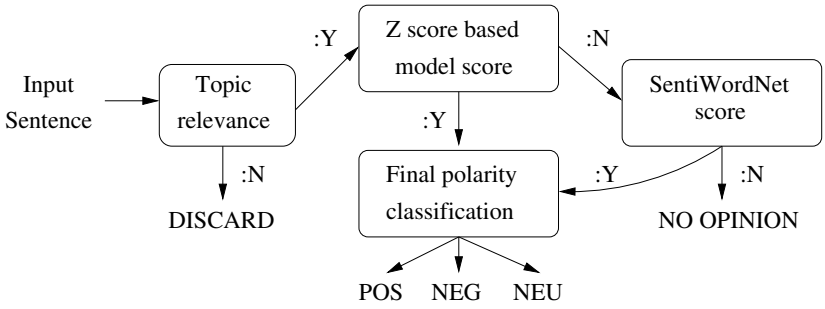

Figure 4.4: Two-step classification model based on Z score with the use of SentiWordNet.

Additionally, we incorporate the use of the linguistic component of the SentiWordNet sentiment polarity dictionary [Esuli 2006b]. Since newspaper articles do not tend to use casual language or too much ambiguity or allusion, we looked for lists describing the general semantic intensity and orientation of the words. For this purpose, we get a polarity score from the SentiWordNet for each word. SentiWordNet uses the synsets from the WordNet assigning to each of them a triple of scores: for positivity, negativity and objectivity of a synset $(swn^{pos}, swn^{neg}, swn^{obj})$. The objectivity score is defined as $swn^{obj} = 1 - (swn^{pos} + swn^{neg})$, where $swn^{pos} + swn^{neg} \leq 1$. Each word is also annotated with its POS, thus the score triples may vary depending on the POS of a word (synset). Since we do not use any additional syntactic information in our preprocessing experiments, we chose a synset with the highest sum of $swn^{pos} + swn^{neg}$.

Our idea is to combine the SentiWordNet scores with the proposed model's scores for each sentence. First we need to obtain the overall score based on SentiWordNet scores for the whole sentence for each category. As an example, let's take an opinionated sentence with negative polarity from the NTCIR-7 campaign: *With Tokyo's economy declining about 3 percent this year, this seems unlikely.* In Table 4.3 you can observe the values in the second and third columns that are the positivity and negativity scores assigned by the SentiWordNet [Esuli 2006b]. The objectivity score obtained is given in the last column [Esuli 2006b].

#	**Word/Synset**	swn^{pos}	swn^{neg}	swn^{obj}
1	Tokyo	0.0	0.0	1
2	economy	0.125	0.25	0.625
3	declining	0.0	0.0	1
4	about	0.375	0.0	0.625
5	percent	0.125	0.0	0.875
6	this	0.0	0.0	1
7	year	0.0	0.0	1
8	seem	0.0	0.0	1
9	unlikely	0.0	0.625	0.375

Table 4.3: SentiWordNet positive, negative and objectivity scores for each word in the example sentence.

Looking at the scores for the individual tokens in the example sentence, the SentiWordNet opinionated score will be the sum of swn^{pos} and swn^{neg} for each token (Equation 4.7). The not opinionated score will be a sum of objectivity score for each token, divided by the number of tokens or words in the sentence. Thus, we have an opinionated score and not opinionated scores defined as follows:

$$Op\_score^{SWN}(s_i) = \sum_{t \in s} (swn_{t_i}^{pos} + swn_{t_i}^{neg}) \tag{4.7}$$

$$Noop\_score^{SWN}(s_i) = \frac{\sum_{t \in s} swn_{t_i}^{obj}}{n} \tag{4.8}$$

where n is the length of the sentence. Therefore we would get the following values for our example sentence:

$$Op\_score^{SWN} = 0.125 + 0.25 + 0.375 + 0.125 + 0.625 = 1.5$$

$$Noop\_score^{SWN} = (1 + 0.625 + 1 + 0.625 + 0.875 + 1 + 1 + 1 + 0.375)/9 = 0.833$$

Thus, if opinionated score is more than not opinionated one, there is an opinion, otherwise not. This technique is favoring the opinionated score and is a heuristic approach that intuitively takes account of the rationalization that there are more not opinionated words than opinionated ones in the sentence. The presence of opinionated word weights more than the presence of the not opinionated one. This approach seems to give good results in practice for the NTCIR corpora. This could be due to the fact that newspaper articles tend to have few highly subjective terms, at the same time containing a lot of factual terminology.

Finally, the proposed classification model normalizes and sums of the opinionated and not opinionated scores obtained from the SentiWordNet [Esuli 2006b] with our Z score based opinionated and not opinionated scores for the sentence. For the final calculation of the polarity score, if our system classified the sentence as not opinionated but $Op\_score^{SWN} > Noop\_score^{SWN}$, the sentence is classified as opinionated. For polarity classification (positive, negative or neutral) in this case, we take the category with the highest sum of the Z scores for the sentence.

In [Zubaryeva 2010b] we submitted two runs. The first run was composed of the judgments of the initial classification model based on the logistic regression. The second run represented the judgment results of the improved cascade model, incorporating two stages of the classification, as well as the use of the SentiWordNet [Esuli 2006b]. The evaluation of our performance is given in Table 4.4.

Subtask	Runs	Precision	Recall	F1-measure
Opinion	*Run 1*	29.44%	62.84%	40.1%
Opinion	*Run 2*	19.32%	81.79%	31.26%
Polarity	*Run 1*	50.29%	29.58%	37.25%
Polarity	*Run 2*	48.35%	37.8%	42.43%

Table 4.4: NTCIR-8 MOAT evaluation of the two submitted runs.

The results obtained in the MOAT task [Zubaryeva 2010b] show precision, recall and F1-measure for opinion and polarity subtasks. The opinion subtask showed how well we were able to identify sentence with the opinion. The polarity subtasks showed performance in sentence classification according to positive, negative or neutral polarity. Run 2 gives low precision for the opinion subtask with however a high recall value. Thus, in comparison to Run 1, we can see that the use of the SentiWordNet [Esuli 2006b] improved precision but lowered recall, nevertheless, allowing us to achieve a quite high F1-measure in comparison to other teams. Overall, we noticed a general improvement in our results in relation to the initial model proposed in [Zubaryeva 2008]. It seems that with the growth of training data (NTCIR-6 and NTCIR-7 corpora) as well as the use of two-step classification with the bigram of words improves the system's performance. In order to evaluate some of the reasons of failure when doing opinion classification, we looked closely and analyzed the system's decision on a sample example from the NTCIR collection in the next subsection.

4.5.1 Error and Misclassification Analysis

Several experiments were conducted on the NTCIR-6 and NTCIR-7 MOAT corpora that help clarify some reasons why our method fails to make correct classification. As one of the corpora peculiarities pertinent to our classification system's performance, we determined that a great number of words especially in opinionated category, occur one to four times in the collection. This corresponds to the long tail of the Zipf's law. With such low frequencies of occurrence, they do not

carry reliable information to help the classification procedure. As an example let's take the following neutral in polarity sentence: *Half of the job is psychiatry.* If we eliminate the stop words, we end up with three words: *half*, *job* and *psychiatry*. The term *psychiatry* is a *hapax* term, meaning that it occurs only once in the collection, therefore we have no Z score for it. For the other two terms we have the following scores: *half* with -2.83 and *job* with 2.16. The Z score for the term *half* shows us that this term is overused in the not opinionated part of the corpora ($|-2.83| > 2$, where 2 is the threshold used to select the overused and underused features in the category). It's absolute value being bigger than the Z score of the term *job*, the system will classify the sentence in not opinionated category. As you can see, due to low frequencies of occurrences of many terms in the collection, when calculating the Z score for the sentence, we can end up in a situation where we have scores only for several terms, even in long sentences.

4.6 Comparison with Baselines

In this section we give an overview and analysis of the comparison study of the state-of-the-art approaches for opinion classification (SVM and naïve Bayes), as well as the Z score model for opinion classification on the English part of the NTCIR newspaper articles corpora.

In [Zubaryeva 2010a], we merged both of the corpora of NTCIR-6 and NTCIR-7, obtaining 10,145 sentences, 2,495 (or 24.6%) of which contain an opinion and 7,650 (or 75.4%) of factual sentences. It is important to note that we performed the classification on all three sentiment categories, positive, negative and mixed. In this study we adopted a specific terminology for features representing the text. Thus, since the use of the term "word" becomes ambiguous, we used "lemma" for the dictionary entry of the word, "term" for the form appearing in the text, and "word type" for every distinct term. Thus, in the sentence *the dogs saw the brown dog*, we would count six tokens, but only four lemmas (*the, dog, see, brown*) and five word types (*the, dogs, saw,*

brown, dog). Therefore, as features in the model we consider terms and lemmas, unstemmed and stemmed forms of words appearing in text.

We made some more precisions. In our system, all capitalized words are replaced with corresponding non capitalized word (e.g., *Jobs* and *jobs*). This modification is not always advantageous (e.g., the name *Steve Jobs*). However, if the word is written in all capital letters, it stays the same (e.g., *US*, *DOJ* or *AIDS*). As other natural languages, English has two possible orthographic variants for some words, e.g., *center, centre* or *defense, defence*, that our system does not group in the same index entry. Finally, we have not done a deep preprocessing as, for example, the elimination of the derivative suffices (e.g., *China* and *Chinese*). Similarly, we have not taken the account of the synonyms in order to group them under one entry (via, for example, WordNet thesaurus [Fellbaum 1998]).

Out of 10,145 sentences in our corpus, there are 219,038 words for 14,025 different terms (or 15,259 before the light stemming procedure). We have also ignored 41 very frequent terms that carry no information (e.g., *the, is, of, and, which*). Moreover, eliminating the terms that appeared three times or less, the feature space reduces from 14,025 to 5,021 terms, in other words gives a reduction of 64.2% (or from 13,160 to 4,652 lemmas (-65.8%)). Elimination of the less frequent terms facilitates the problem of the size of the feature space for our classifiers.

In order to give an idea of the most frequent terms, Table 4.5 indicates the fifteen most frequent terms in opinionated and factual parts of the corpus. In this table, we have indicated the term frequency (column tf) as well as the number of sentences that have at least one occurrence of the term (column df).

In order to determine if a sentence contains an opinion, we calculate the Z score for every term or lemma of the sentence. As a rule, we calculate the sum

of Z scores higher than 1 (denoted $sumPos$) and the sum of the Z scores lower than -1 (denoted $sumNeg$). If $sumPos > |sumNeg|$, a sentence is tagged as opinionated, otherwise it is not opinionated. From the conducted experiments we found that the simplified classification based on the computation of the Z score sums for each category gives similar or higher level of the performance than the use of the logistic regression with the Z score computation for the sentence-level opinion classification.

#	**Opinionated**			**Factual**		
	tf	**Term**	**df**	**tf**	**Term**	**df**
1	536	said	529	772	said	754
2	422	not	398	646	not	609
3	290	he	254	552	he	487
4	201	we	169	423	japan	386
5	175	I	152	394	two	383
6	166	US	143	386	US	359
7	166	government	161	371	government	353
8	158	should	151	368	korean	314
9	153	more	139	354	korea	318
10	141	japan	126	329	other	315
11	139	world	132	329	after	323
12	138	chinese	119	325	more	311
13	133	korea	120	315	south	297
14	127	economic	123	311	economic	292
15	116	other	111	306	country	292

Table 4.5: The most frequent terms in opinionated (2,495 sentences) and factual (7,650 sentences) categories.

4.6.1 Naïve Bayes and SVM Parameters

In order to evaluate different categorization models, we have adopted as a baseline the naïve Bayes approach [Mitchell 1997]. In this case, the categorization system has to choose between two possible hypotheses: h_0=*no opinion* and h_1=*opinion*. The selected category would be the one that has the maximum value in the Equation 4.9. There, we have t indicating the number of terms included in the current sentence and t_j - number of terms appearing in the sentence.

$$\arg\max_{h_i} Pr(h_i) \cdot \prod_{j=1}^{t} Pr(t_j|h_i), \ where \ \ Pr(t_j|h_i) = \frac{tf_{jh_i}}{n_{h_i}} \tag{4.9}$$

The conditional probabilities have to be estimated. For the *a priori* probabilities $Pr(h_i)$, the estimation is based on the relation between the number of not opinionated sentences (7,650), opinionated (2,495), and the total number of sentences in the corpus (10,145). For calculating the probabilities of different terms, we regroup all the sentences belonging to one category (together denoted as T_{h_i}). Based on the whole size of n_{h_i}, we estimate the probabilities with the equation 4.9. It gives us the relation between the lexical frequency of term t_j in T_{h_i} (denoted tf_{jh_i}) and the size of the corresponding corpus.

This estimation (maximum likelihood) has a tendency to overestimate the probabilities of terms present in the training corpus over the absent ones. In this case, the value of tf_{jh_i} being equal to 0, gives a zero probability of occurrence. Moreover, it is known that the word distribution follows the distribution of LNRE (*Large Number of Rare Events* [Baayen 2001]). As a correction, the smoothing procedure consists of adding a value to the numerator of our estimation formula and adding the size of the dictionary to the denominator [Manning 2002]. This formula is equivalent (according to the law of Lidstone) to smoothing all probabilities by the formula $p_j = (tf_{jh_i} + \lambda)/(n_{h_i} + \lambda \cdot |V|)$, where λ is a smoothing parameter (fixed at 0.3) and $|V|$ the size of the dictionary (e.g., 4,135 terms if h_0 and 2,095 if h_1).

For the SVM model we adopted a vector space model with the classic $tf \cdot idf$ weighting scheme [Boughanem 2008]. The tf indicates the frequency of the term occurrence in the sentence. The value $idf = \log df/n$ essentially corresponds to the logarithm of the inverse document frequency (denoted df). The latter indicates the number of sentences where this term appears, with n designating the number of sentences in the corpus.

As an alternative, we can normalize the two frequencies so that we can obtain a value in the range from 0 to 1. For the tf value we implemented the following weighting: $atf = 0.5 + 0.5 \cdot (tf/\max tf)$. The maximum of tf represents the maximum occurrence frequency in the considered sentence. For the idf, we simply divide idf by $\log n$, normalization denoted by $nidf$.

Having the vector form representation, we used the SVM^{light} (Support Vector Machines) system[1] that provides libraries to use the support vector machines learning model [Joachims 2001]. This model with the use of the polynomial or linear kernel functions (transforming the term representation space (weighting $tf \cdot idf$)) sometimes allows to reach a good level of performance at the cost of fast growth of the processing time during the learning stage.

4.6.2 Experimental Results: SVM, Naïve Bayes, Z Score Model

On the basis of the NTCIR corpus, we have evaluated different categorization strategies using 10-fold cross-validation. The average precision, recall and F1 values are indicated in the Table 4.6. There, we represent different models with terms and lemmas as features; with the use of smoothing (symbol λ) or not, or eliminating all the terms with the frequency lower than 4 (indicated as *min:4*). Comparing the best results obtained by the three classifiers, our model, based on the Z score, proposes the best strategy, with the use of the three different

[1] http://svmlight.joachims.org

computational parameters. Taking into the account only the F1-measure, we can see a clear difference in performance between the Z score (#16) and the SVM approach (#10) (57.34% vs. 45.33%, or a relative difference of 21%). With the naïve Bayes model, the best F1 has a value of 30.49% (#6), a relative difference of 46.83% with the Z score (#16).

To confirm this conclusion, we can apply a paired t-test (bilateral, significance level $\alpha = 5\%$). On the basis of term representation (with the smoothing λ et *min:4*), performance based on the F1-measure of the Z score method (#13) is significantly different to the one of the SVM model (#9) or naïve Bayes (#1, difference denoted by the symbol † in Table 4.6). Based on recall, we obtain the same confirmation, whereas for the precision the difference in performance between the Z score and SVM is not significant. If we take lemmas as features in the models (with the smoothing λ, *min:4* or #17 with #11 or #5), the difference in performance between a model based on the Z score and the other two is statistically significant, no matter if we are using precision, recall or the F1-measure.

In order to determine the feature representation most advantageous for the performance, let's look at the values in the table 4.6. They clearly indicate that the use of terms is better for the classifier based on the Z score. However, for the naïve Bayes model, this distinction does not bring any significant variation in the performance. The same situation can be observed with the SVM model. Feature representation with terms improves the precision, but somewhat lowers the recall. The difference in performance between these representations stays weak.

To reduce the feature representation space (up to 60%), we proposed to ignore the terms with the frequency less than four. Moreover, we can smooth the conditional probabilities using the approach proposed by Lidstone (with $\lambda = 0.3$). These techniques do not have a systematic effect on the performance

#	**Model, parameters**	**Prec.**	**Recall**	**F1**
1	Naïve Bayes, term, λ, min:4	18.89%†	67.45%†	29.52%†
2	Naïve Bayes, term, λ, min:0	19.50%	61.48%⋆	29.60%
3	Naïve Bayes, term, min:4	19.05%⋆	68.09%⋆	29.76%⋆
4	Naïve Bayes, term, min:0	13.20%⋆	37.08%⋆	19.46%⋆
5	Naïve Bayes, lemma, λ, min:4	18.21%†	63.03%†	28.26%†
6	Naïve Bayes, lemma, λ, min:0	20.15%	63.03%⋆	*30.49%*
7	Naïve Bayes, lemma, min:4	18.67%	64.80%⋆	28.99%
8	Naïve Bayes, lemma, min:0	14.64%⋆	42.01%⋆	21.69%⋆
9	SVM, term, $tf \cdot idf$	33.63%	67.76%†	44.37%†
10	SVM, term, $atf \cdot nidf$	34.99%⋆	64.97%	*45.33%*
11	SVM, lemma, $tf \cdot idf$	32.42%†	66.80%†	43.33%†
12	SVM, lemma, $atf \cdot nidf$	33.06%	67.19%	43.95%
13	Z score, term, λ, min:4	44.23%	82.72%	56.30%
14	Z score, term, λ, min:0	43.93%	**84.49%**⋆	56.50%
15	Z score, term, min:4	**45.54%**⋆	81.00%⋆	57.01%⋆
16	Z score, term, min:0	45.40%⋆	82.97%	**57.34%**⋆
17	Z score, lemma, λ, min:4	39.29%	81.71%	52.87%
18	Z score, lemma, λ, min:0	39.19%	83.80%	53.23%
19	Z score, lemma, min:4	41.70%	77.38%⋆	53.92%
20	Z score, lemma, min:0	41.46%	79.91%⋆	54.36%

Table 4.6: Evaluation of different classification strategies (cross-validation, 10 folds; 2,495 opinionated, 7,650 not opinionated).

among the different classifiers. However, we can deduce several tendencies. The performance variations stay weak in the runs with all terms or only those with the frequency higher than three. Reduction of the representation space tends to improve precision for the Z score model (e.g., #13 vs. #14) or improve the recall of the naïve Bayes runs (e.g., #1 vs. #2). In the latter case, it is necessary to note that the use of all terms and no smoothing techniques is detrimental to the

performance results (e.g., #4 or #8).

$\lambda = 0.001$	$\lambda = 0.1$	$\lambda = 0.3$	$\lambda = 0.5$	$\lambda = 0.75$	$\lambda = 1.0$
56.95%	56.70%	56.30%	56.05%	55.77%	55.39%

Table 4.7: F1 evaluation (10-fold cross-validation) with different values for the λ parameter (*Z Score model, term, min:4*).

In order to identify the important differences between the runs, we note that each run in the Table 4.6 starts with the name of the model (e.g., #1 (Naïve Bayes), #9 (SVM) or #13 (Z score)). The statistically significant differences with this model are indicated with the star $\star$. Finally, we were convinced that the value of the parameter λ in the probability estimation did not have a significant impact on the performance of the Z score model. The values of the F1-measure indicated in the table 4.7 confirm this hypothesis. Their calculation was based on the run #13 (see table 4.6) together with the performance values obtained in the run #15 (F1: 57.01%), where $\lambda = 0$. Substantial improvement in the performance cannot be found in tuning of this parameter.

4.6.3 Variations of the SVM Model

For the SVM model we can say that the weighting variations $tf \cdot idf$ or $atf \cdot nidf$ do not really modify the results. Moreover, the use of the sigmoid kernel functions (weighting $atf \cdot nidf$) improves somewhat the performance (F1: 46.26% vs. 45.33%, relative difference of 2.05%). This growth is due to the precision improvement (40.93% vs. 34.99%), accompanied with the decrease in recall (54.53% vs. 64.97%). On the contrary, the time of processing grows in the order of 2,000% limiting the interest for possible more complex representations. As an alternative, we can vary the parameter C (cost) indicating the error tolerance rate during learning. Thus, a low C value indicates a bigger error tolerance during the whole learning stage (soft margin), whereas a bigger value will penalize such errors.

As possible variations with our base SVM model (#9, Table 4.6, F1: 44.37%), we considered the values of C proposed by [Joachims 2002]. The latter and the corresponding F1 values are indicated in Table 4.8. With the best choice ($C = 0.05$), the best result is 50.19% (or relative increase in 13.1%). Contrary to the Z score (Table 4.7), the choice of value for the C parameter has an important impact, without the possibility, *a priori*, to know the optimal value (e.g., [Joachims 2002] puts $C = 5$ in the *Reuters* or *Ohsumed* [2] collection classification system).

Default	$C = 0.001$	$C = 0.05$	$C = 0.1$	$C = 1$	$C = 10$	$C = 1000$
44.37%	49.58%	50.19%	49.34%	48.06%	47.22%	44.18%

Table 4.8: F1 evaluation (cross-validation) with different values for the C parameter (SVM model, term, $tf \cdot idf$).

Finally, the difference between a retrospective evaluation (the same corpus for training and testing) and the cross-validation arises. Thus, with the $tf \cdot idf$ weighting scheme and the feature representation in terms, the F1-measure grows from 44.37% to 71.77%, relative growth of 61.7%. We can explain such difference by the fact that the classification decision in this approach is based on the subset of phrases. When the same phrase belongs to the training and testing set the decision is quite facilitated and the result is highly positively biased.

4.6.4 Analysis of Some Sentences

In order to explain the difference in decision-making between the classifiers, we have analyzed several runs and errors in the classification. From our point of view, the naïve Bayes approach does not discriminate the words based on their presence in all opinionated or not opinionated sentences. Really, the estimation

[2] http://www.davidlewis.com/resources/testcollections/reuters21578/ for Reuters, ftp://medir.oshsu.edu/pub/ohsumed for Oshumed

in the Equation 4.9 takes into account only the frequency of occurrence in the considered category. Thus, the three words having the biggest probabilities are indicated in both categories (see Table 4.5). They are *said* (with opinionated: $16.83 \cdot 10^{-3}$, without opinion: $8.11 \cdot 10^{-3}$), *not* (opinionated: $13.25 \cdot 10^{-3}$, without opinion: $6.79 \cdot 10^{-3}$) or *he* (opinionated: $9.11 \cdot 10^{-3}$, without opinion: $5,80 \cdot 10^{-3}$).

Negative opinion	*The hawks say the sub's mission was provocative, as it clearly infringed on South Korean waters.*
<NB>(0.365/0.635)	hawk(-), say(10.76/6.96), sub(1.35/0.77) mission(2.92/2.03), provocative(-), clearly(4.49/1.19) infringed(-), south(23.32/33.11), korean(32.42/38.68) water(2.29/4.76)
<SVM>	hawk(7.43), say(4.63), sub(6.83), mission(5.89) clearly(6.0), south(3.31), korean(3.22), water(5.33)
<Z score>(5.18/-6.87)	hawk(1.01), say(1.39), sub(0.56), mission(0.52) provocative(-), clearly(2.78), infringed(-), south(-2.95) korean(-2.05), water(-1.87)
Negative opinion	*You were often abused and humiliated.*
<NB>(0.397/0.603)	you(12.65/7.7), often(4.17/3.39), abused(-) humiliated(-)
<SVM>	you(4.64), often(5.42), abused(7.15)
<Z score>(1.76/0)	you(1.76), often(0.26), abused(-0.15), humiliated(-)

Table 4.9: Examples of sentence representation in different classification models.

In Table 4.9 we have presented several sentences where at least one model gives an error judgment. In every case we have given the true category, as well as the sentence itself. After the tag <NB> we have grouped the representation based on the naïve Bayes model. In this case, we have indicated in the parentheses the probabilities of the phrase belonging to the opinionated and not opinionated

categories, then, for each word, the same two probabilities (multiplied by 10^5). For the SVM model, we consider the $tf \cdot idf$ weighting scheme calculated for each word. Finally, for the Z score, we indicated each word followed by its Z score. At the beginning between the parentheses we have given the values of *sumPos* and *sumNeg* that are used to decide on the opinionated or not opinionated category.

In the first sentence the naïve Bayes and the Z score models take a wrong decision. In this case, the two words *south* and *korean* tilt the balance towards the *not opinionated* category without letting other terms to overrule this effect (e.g., words *clearly* or *say*). Again, potentially important words have low lexical frequency to influence the decision (e.g., *provocative* with the lexical frequency of three).

The second sentence is correctly classified by the Z score but not by the naïve Bayes model. The word *humiliated* will be ignored because its frequency is one (*hapax*). The word *abused* occurs only twice in the opinionated documents and will be ignored in the representation of the naïve Bayes approach. In the latter case, the decision has to be taken only on the basis of these two words (*you*, *often*) favoring the presence of opinion (2/1). However, the value of the probabilities *a priori* (1/4) is in favor of the not opinionated sentences which, in this case tilts the scales toward the decision of the *not opinionated* category.

4.7 Other Metrics for Feature Weighting

The Z score based approach presented in [Zubaryeva 2010b] was able to outperform other systems participating in NTCIR-8 MOAT campaign. These systems were using SVM and other ML approaches. We decided to perform an evaluation of similar statistical measures, such as information gain, χ^2, log likelihood, and odds ratio commonly used in feature selection procedures for text classification [Forman 2003, Mladenic 1999, Yang 1997]. We also introduce our take on the Kullback-Leibler (KL) divergence measure (KL score) presented

in [Schneider 2004] that we adapt for the sentiment and opinion classification tasks. In this section, we first present the KL-divergence based measure (KL score), then we give some experimental results and analysis of different information metrics, including Z score and KL score.

4.7.1 KL Score

In our experiments we adopted a feature selection measure described in [Schneider 2004] that is based on the Kullback-Leibler divergence (KL-divergence) measure. In this paper, the author seeks to find a measure that would lower the score of the features that have different distribution in the individual training documents of a given class from the distribution in the whole corpus. Thus, the scoring function would allow to select features that are representative of all documents in the class leading to more homogeneous classes. The scoring measure based on KL-divergence introduced in [Schneider 2004] yields an improvement over MI with naïve Bayes on Reuters dataset, frequently used as a text classification benchmark.

Schneider [Schneider 2004] shows how we can use the KL-divergence of a feature f_t over a set of training documents $S = d_1, ..., d_{|S|}$ and classes c_j, $j = 1, ..., |C|$ is given in the following way:

$$KL_t(f) = \tilde{K}_t(S) - \tilde{KL}_t(S) \tag{4.10}$$

where $\tilde{K}_t(S)$ is the average divergence of the distribution of f_t in the individual training documents from all training documents. The difference $KL_t(f)$ in the Equation 4.10 is bigger if the distribution of a feature f_t is similar in the documents of the same class and dissimilar in documents of different classes. $\tilde{K}_t(S)$ is defined in the following way:

$$\tilde{K}_t(S) = -p(f_t) \log q(f_t) \tag{4.11}$$

where $p(f_t)$ is the probability of occurrence of feature f_t (in the training set). This probability could be estimated as the number of occurrences of f_t in all training

documents, divided by the total number of features. Let N_{jt} be the number of documents in c_j that contain f_t, and $N_t = \sum_{j=1}^{|C|} N_{jt}/|S|$. Then $q(f_t|c_j) = \sum_{j=1}^{|C|} N_{jt}/|c_j|$ and $q(f_t) = N_t/|S|$. The second term from 4.10 is defined as follows:

$$\tilde{KL}_t(S) = -\sum_{j=1}^{|C|} p(c_j)p(f_t|c_j)\log q(f_t|c_j) \tag{4.12}$$

where $p(c_j)$ is the prior probability of category c_j, and $p(f_t|c_j)$ is the probability that the feature f_t appears in a document belonging to the category c_j. d_i. Using the maximum likelihood estimation with a Laplacean prior, Schneider [Schneider 2004] obtains:

$$p(f_t|c_j) = \frac{1 + \sum_{d_i \in c_j} n(f_t, d_i)}{|V| + \sum_{t=1}^{|V|} \sum_{d_i \in c_j} n(f_t, d_i)} \tag{4.13}$$

where $|V|$ is the training vocabulary size or the number of features indexed, $n(f_t, d_i)$ is the number of occurrences of f_t in d_i. It is important to note that the afore mentioned average diversion calculations are really approximations based on two assumptions: the number of occurrences of f_t is the same in all documents containing f_t, and all documents in the class c_j have the same length. These two assumptions may turn detrimental for long extract text classification as noted by the author himself [Schneider 2004], but turn out quite effective for a sentence classification setup where a phrase mostly consists of features that occur once, with usually low variations in sentence length. It is important to note that the computation of $p(f_t|c_j)$ should be done on a feature set with removed outliers, since they occur in all or almost all sentences in the corpora.

In sentence-based classification the pruning of the feature set can turn out quite detrimental to the classification accuracy. This is true if the size of the training set is not big enough in order to be sure that some important for classification features are not discarded. Thus, we propose to modify the KL-divergence measure for sentiment and opinion classification. In [Schneider 2004] it calculates the difference between the average divergence of the distribution of f_t in individual training documents from the global distribution, all this

averaged over all training documents in all classes. For the sentiment/opinion classification task it is interesting to calculate the difference between the average divergence in one class from the distribution over all classes. Therefore, we can obtain the average divergence of the distribution of f_t for each of the classification categories ($j \in POS, NEG$):

$$\tilde{KL_t}^{j}(S) = N_{jt} \cdot \tilde{p}_d(f_t|c_j) log \frac{\tilde{p_d}(f_t|c_j)}{p(f_t|c_j)} \tag{4.14}$$

Substituting $\tilde{KL_t}^{POS}(S)$ and $\tilde{KL_t}^{NEG}(S)$ in Equation 4.10 for each category we obtain measures that evaluate how different is the distribution of feature f_t in one category from the whole training set.

$$KL_t^{POS}(f) = \tilde{K_t}^{POS}(S) - \tilde{KL_t}^{POS}(S) \tag{4.15}$$

$$KL_t^{NEG}(f) = \tilde{K_t}^{NEG}(S) - \tilde{KL_t}^{NEG}(S) \tag{4.16}$$

This way, we obtain two sums $\sum KL_t^{POS}(f)$ and $\sum KL_t^{NEG}(f)$ over the features present in the sentence. The final difference of the two sums can serve as a prediction score of to which category the sentence is most similar.

4.7.2 Experiments with Different Information Measures

In text classification, after calculating the scores between every feature and every category the next steps are to sort the features by score, choose the best k features and use them later to train the classifier. For the task of sentiment classification on a sentence-based level the pruning of the feature set may lead to the elimination of infrequent features (several occurrences) and may cause the loss of important information needed for classification of the new instances. Here are some differences in the aspects of use of the feature selection measures in text classification and opinion/sentiment analysis contexts. First, the aim in topic text classification is to look for the set of topic-specific features that describe the classification category. In sentiment classification, though, the markers of the opinion could be carried by both topic-specific and context words that may also have small differences in distributions across

categories due to the short text length. If we look at the opinion review domain, the topic-specific features would be *movie*, *film*, *flick* and context words would be (*long*, *short*, *horror*, *satisfy*, *give up*). Thus, we are not interested to select just the top features describing the topic domain, but to take into account the variety of context-specific words that may carry the opinion.

Second, the usual text classification methods are designed for documents consisting of at least several hundreds of words, assuming that the features that could aid in classification repeat across the text several times. The format of a sentence does not let us make the same assumption. The opinion or sentiment polarity can be expressed with the help of one word/feature. There is substantial evidence from several studies that the presence/absence of a feature is a better indicator than the tf scores [Pang 2008]. Thus, for effective classification, the model should identify features that are strong indicators of opinion/sentiment, take into account the relations between the features in each category, and be able to adjust scores of the features that were not frequent enough in order to expand the set of features that are strong indicators of the sentiment.

The studies on feature selection metrics for text classification recommend several information metrics that are expected to give an amelioration in classification accuracy coupled with some ML classifier. The studies report odds ratio (OR) and IG to work well to identify the discriminating power of terms for further classification with naïve Bayes [Mladenic 1999, Yang 1997]. From the analysis performed in [Forman 2003], we expect a high performance in selecting discriminative features from the IG and Z score metrics. The authors showed that the χ^2 metric tends to perform similar to IG. At the same time an important finding that we are interested to confirm claimed in [Forman 2003] the complementary performance of the IG and Z score together. IG scores performs better than other metrics on the balanced datasets in terms of the available training documents, whereas Z score performs better on unbalanced datasets for the text classification task. Table 4.10 presents the overview of the metrics used

#	Measure	Formula
1	IG	$\sum_{c\in\{c_i,\overline{c_i}\}}\sum_{t\in\{f_k,\overline{f_k}\}} P(f,c)\cdot\log\frac{P(f,c)}{P(f)P(c)}$
2	χ^2	$\frac{N\cdot(P(f_k,c_i)\cdot P(\overline{f_k},\overline{c_i})-P(f_k,\overline{c_i})\cdot P(\overline{f_k},c_i))^2}{P(f_k)\cdot P(\overline{f_k})\cdot P(c_i)\cdot P(\overline{c_i})}$
3	Odds Ratio	$\log\left(\frac{P(f_k,c_i)\cdot(1-P(f_k,\overline{c_i}))}{(1-P(f_k,c_i))\cdot P(f_k,\overline{c_i})}\right)$
4	$Log-Likelihood*$	$2(\log L(P_1,K_1) + \log L(P_2,K_2) - \log L(Q,K_1) - \log L(Q,K_2))$
5	Z score	$\frac{a-(a+c)\cdot P(f)}{\sqrt{(a+c)\cdot P(f)\cdot(1-P(f))}}$
6	$KL\ score\star$	$-p(f_t)\log q(f_t) - \sum_{j=1}^{\lvert C\rvert} p(c_j)p(f_t\lvert c_j)\log q(f_t\lvert c_j)$ $-\sum\frac{a+b+1}{N^T+a+b}\log\frac{a}{N_d^{POS/NEG}} + \frac{1}{2}\frac{a}{a+b}\log\left(\frac{a}{2}\right)$

Table 4.10: Feature weighting measures. N is the number of distinct terms, a is the number of occurrences of f in the subcorpus, $a+c$ is the number of all terms in the subcorpus. $Log-Likelihood*$ - formula and notation as introduced in [Dunning 1993]. $KL\ score\star$ - as introduced in Section 4.7.1.

in our experiments. Using the notation from the Table 4.1, we can rewrite the formulae as in Table 4.11.

This time we expanded our experiment setup with publicly available corpora from the newspaper review (MPQA) and movie review (Movie Review and Subjectivity datasets) domains. One of the important characteristics of the latter datasets is that they are balanced in terms of the size of the training sets for both categories.

As a classification model we use a simple additive score of the features in the sentence computed for each category. Our aim is to determine the behavior of each of the metrics for the task of sentence sentiment and opinion classification in terms of their goodness and priority in feature weighting based on feature distribution across classification categories.

The results of the experiments are shown in Tables 4.12 and 4.13. For

#	Measure	Formula
1	IG	$\frac{a}{N}\log\frac{aN}{(a+b)(a+c)}+\frac{b}{N}\log\frac{bN}{(a+b)(b+d)}+\frac{c}{N}\log\frac{cN}{(a+c)(c+d)}$ $+\frac{d}{N}\log\frac{dN}{(b+d)(c+d)}$
2	χ^2	$\frac{N\cdot(ad-cb)^2}{(a+c)\cdot(b+d)\cdot(a+b)\cdot(c+d)}$
3	Odds Ratio	$\log\left(\frac{a\cdot d}{b\cdot c}\right)$
4	Log-Likelihood	$2\cdot(a\log a+b\log b+c\log c+d\log d-(a+b)\log(a+b)$ $-(a+c)\log(a+c)-(b+d)\log(b+d)-(c+d)$ $\log(c+d)+(a+b+c+d)\log(a+b+c+d))$
5	Z score	$\frac{ad-cb}{\sqrt{(a+c)\cdot(a+b)\cdot(c+d)}}$
6	KL score	$-\sum\frac{a+b+1}{N^T+a+b}\log\frac{a}{N_d^{POS/NEG}}+\frac{1}{2}\frac{a}{a+b}\log\left(\frac{a}{2}\right)$

Table 4.11: Feature weighting measures using the information elements presented in Table 4.1. $N=a+b+c+d$, N^T is the number of distinct terms, $N_d^{POS/NEG}$ is the number of documents in the specific category.

the unbalanced datasets we do not indicate the accuracy values, since the skew in the number of training documents between opinionated and factual sentences is to big (e.g., 598 opinionated and 9,548 factual sentences for the NTCIR OP strict dataset). A classifier that predicts all the time the sentence as not opinionated would achieve an accuracy of 94% on NTCIR OP strict dataset. Therefore, we rely more on the macro-averaged precision and recall calculated over the two categories in evaluation of the classifier performance. This way we give equal priority to both categories. It is also possible to use the micro-averaged precision and recall, which attributes the same importance to each document. Since our model always takes a classification decision and there are no sentences left unclassified in both categories, the micro-averaged precision and recall would be equivalent to the accuracy.

We use the setup with unigram indexing, short stop word elimination (several prepositions and verb forms: *a, the, it, is, of*) and the use of the Porter stemmer [Porter 1997]. The experiments on other character *n-gram*, bigram and

Eval	IG	CHI	OR	LL	Z sc	KL sc
Movie Review dataset						
Precision	57.72%	66.13%	65.76%	66.11%	65.43%	**67.26%**
Recall	57.36%	66.09%	65.73%	66%	65.29%	**72.01%**
F1	57.54%	66.11%	65.75%	66.05%	65.36%	**69.55%**
Accuracy	57.37%	66.1%	65.73%	66%	65.29%	**68.48%**
Subjectivity dataset						
Precision	74.19%	82.01%	76.27%	71.04%	81.89%	**90.94%**
Recall	74.17%	81.71%	76.01%	58.68%	81.77%	**90.93%**
F1	74.18%	81.86%	76.14%	64.26%	81.83%	**90.93%**
Accuracy	74.2%	81.7%	76.01%	58.68%	81.79%	**90.93%**
MPQA dataset						
Precision	61.28%	74.69%	67.38%	74.77%	74.55%	**75.53%**
Recall	53.92%	58.62%	57.83%	57.8%	57.28%	**61.39%**
F1	56.93%	65.68%	62.24%	65.19%	64.78%	**67.69%**
Accuracy	58.01%	62.66%	61.55%	61.95%	61.5%	**65.07%**

Table 4.12: Precision, recall, F1-measure, and accuracy of all metrics over the balanced corpora: Movie Review, Subjectivity and MPQA datasets.

WiseTokenizer setups consistently show lower performance than the first scheme. Without hurting the analysis, we give the evaluation results for the single setup.

It is possible to see that with a simple additive classification scheme we can observe some interesting differences in evaluation results over balanced (Movie Review, MPQA, Subjectivity dataset) and unbalanced datasets (NTCIR OP lenient, NTCIR OP strict, NTCIR SA). The MPQA dataset is not strictly balanced as it contains about a thousand opinionated sentences more than not opinionated. As you can see from Table 4.12 the introduced KL-based measure outperforms all other measures, achieving with a simple classification scheme quite decent performance on all three benchmark datasets. These results lead us

to the conclusion that the new KL-based measure adapted for sentiment/opinion classification can serve as a good baseline approach. It is interesting to note that the difference between the KL score and other measures is lowest for the MPQA dataset, where there are more opinionated documents. From this observation and further analysis of feature distribution in balanced and unbalanced datasets we can deduce that the weighting metrics and, especially KL score, are susceptible to the available size of the training corpora.

Overall, the performance of all measures reflects the classification difficulty of each of the datasets. Thus, we can observe that the Subjectivity dataset is easier for classification for all of the measures. The performance of the Z score is more or less on the same level with other feature weighting measures for the balanced datasets. The IG performed worse than we expected from the analysis

Eval	**IG**	**CHI**	**OR**	**LL**	**Z sc**	**KL sc**
NTCIR OP len. dataset						
Precision	71.70%	67.56%	68.46%	68.43%	70.79%	**71.73%**
Recall	50.64%	55.46%	55.54%	52.28%	**63.8%**	51.06%
F1	59.36%	60.91%	61.31%	59.25%	**67.11%**	59.66%
NTCIR OP str. dataset						
Precision	56.55%	**56.74%**	56.44%	52.97%	55.96%	52.96%
Recall	50.88%	59.61%	**68.62%**	50.09%	53.77%	50.09%
F1	53.56%	58.13%	**61.93%**	51.57%	54.84%	51.57%
NTCIR SA dataset						
Precision	61.89%	73.89%	**76.49%**	66.87%	70.01%	59.56%
Recall	55.56%	59.33%	62.14%	50.38%	**67.39%**	50.72%
F1	58.53%	65.81%	68.53%	67.51%	**68.66%**	52.82%

Table 4.13: Precision, recall, and macro-averaged F1-measure of all metrics over the unbalanced corpora: NTCIR OP lenient, NTCIR OP strict and NTCIR SA datasets.

in [Forman 2003], where it outperformed the χ^2 statistics which is not the case in our experiments.

Let's analyze the performance of the metrics on the unbalanced datasets (see Table 4.13). Looking at the results on the unbalanced datasets, we can see that the KL-based measure performs poorly compared to other metrics. The Z score achieves high recall on NTCIR OP lenient and NTCIR SA where the number of opinionated/positive sentences is not so drastically different between the two categories as in NTCIR OP strict. With the lowest number of the training sentences in an opinionated category for NTCIR OP strict dataset the odds ratio outperforms the Z score. However, it achieves the highest recall on the NTCIR SA dataset which has higher percentage of positive sentences to the whole number of sentences in the training set than NTCIR OP strict. Thus, we can conclude that the size of the training set per category should be a factor to take into consideration when choosing a feature weighting method.

In order to show how features are distributed across categories we give an example on the two datasets, one balanced, in terms of the size of the training set for both categories (Subjectivity dataset) and one unbalanced (NTCIR OP lenient). In Figures 4.5 and 4.6 you can see feature distribution across the two categories. The distribution of features for the balanced dataset is homogeneous across both axes. For the unbalanced dataset the features scatter more closely to the x-axis of the category with more training documents. We show both graphs for the features distributed over a maximum of 2,500 sentences per positive category, so as to have a comparable sized graphs for both datasets. For the NTCIR OP lenient there are only 2,466 opinionated sentences, Subjectivity dataset contains 5,000 sentences in each category. The most frequent features towards the top right corner of the graphs are the most common words: *the*, *a*, *that*, *of*, *it*, *is*, that occur in almost all sentences and are included in our short stop word list.

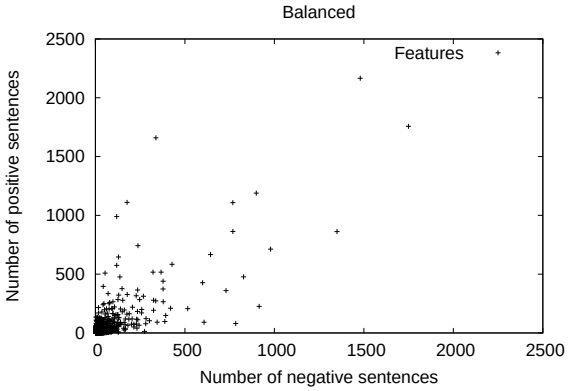

Figure 4.5: Feature distribution over positive and negative classes of sentences for Subjectivity dataset.

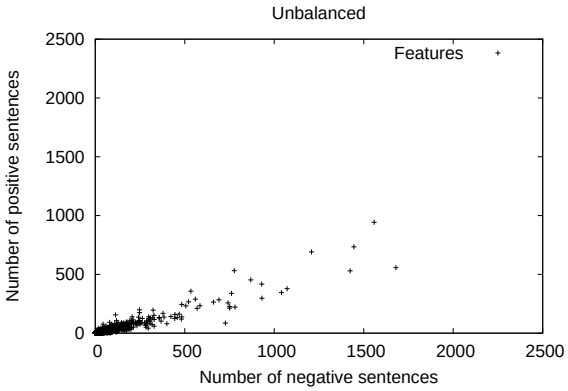

Figure 4.6: Feature distribution over opinionated and factual classes of sentences for NTCIR OP lenient dataset.

We can notice that similar to analysis in [Forman 2003] the top left and bottom right corners do not have any highly predictive features for the specific category. As you can see there are more correlated words across X-axis than Y-axis for the NTCIR OP lenient dataset since it is unbalanced. There are few words with high frequency in both categories and they are mostly unpredictive terms that are usually included in stop word list. When we are dealing with sentence classification the use of an extensive stop word list (more than several words) may turn out to be detrimental to the classification accuracy. For example, the feature *but* has an overall frequency of 1,162 in both categories with only 417 occurrences in a positive category. Thus, this feature has a high Z score for the negative category and is an important predictor for the classification model. Subject to the size of the classification unit with the lack of many predictors, a model needs to somehow incorporate the information that high-frequency features may carry.

Most of the discriminative features appear closer to the origin of the graphs since word distribution follows the Zipf's law. There are many features with the

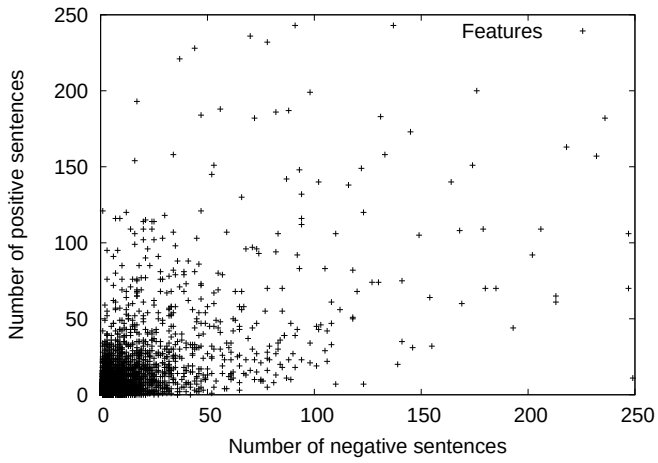

Figure 4.7: Feature distribution over positive and negative classes of sentences for the Movie Review dataset.

same a and b counts. Thus, the decision boundary of a feature weighting metric is important in this region of the graph. That is why similar to [Forman 2003], we perform an analysis where we represent the distribution of words over the two categories and the decision boundary of each of the metrics for selecting the 'best' 100 features. This selection is based on how each of the metrics favor the difference in feature distribution over the categories. For each of the feature weighting metrics presented in Table 4.11, we obtain a score for each of the features f_t in the dataset D and determine a threshold that selects exactly 100 features replicating similar evaluation performed in [Forman 2003]. In order to observe the behavior of the threshold curves for each of the measures closer, we present the zoom on the region close to the origin with the higher density of features (see Figure 4.7) on the example of the Movie Review feature distribution.

We can confirm the analysis conducted in [Forman 2003]: the Z score gives high scores to features unevenly distributed over the two categories even at lower counts of the positive and negative documents. Forman [Forman 2003] identifies Z score as the only metric that performed better for the problems with the high-skew (unbalanced datasets), whereas the IG performed best for

low-skew classes in text classification (balanced datasets). We can observe that not only Z score but also KL score are susceptible to the size of the category training set. KL score outperformed all other metrics on the balanced datasets, whereas it failed to do so on the unbalanced data. From the residual win analysis, on the classification instances where IG fails in [Forman 2003], the Z score metric performs the best. Thus, the author proposes to select a pair of Z score and IG for obtaining the best F1-measure. In our setup, we do not want to reduce the feature size but to obtain such feature scores during training that would allow us to improve the classification accuracy. In the next chapter we propose a procedure based on the KL score and Z score that takes into account the size of the training categories and the dependencies between the co-occurring features in the sentence.

The reason for choosing KL score is its outstanding performance on the benchmark datasets that outperformed IG, odds ratio and other metrics. The Z score we choose for two reasons. First, the previous studies by [Forman 2003, Zubaryeva 2010b] showed its potential for classification of the unbalanced data. Second, it allows us to choose a subset of high frequency features, highly correlated with one of the categories, excluding several stop word terms, that may be beneficial in order to find their frequent neighbors (co-occurrences within the sentence) to improve classification.

4.8 Summary and Discussion

In this chapter, we first present different setups for the text pre-processing. This includes taking decisions about word stemming, lemmatization, stop word removal, and tokenization. After giving an overview of the traditional tokenization schemes, we presented our text tokenization approach for English. We basically use the unigram scheme, with the exception of a small set of words, mostly composed of prepositions. If we encounter these words, we index them together with a preceding term. The aim of this tokenization scheme is to

capture expressions used mostly in colloquial speech.

After presenting the text pre-processing setups, we gave an overview of the feature selection framework and a Z score classification model. First, we account on the experiments carried out for the NTCIR-7 campaign [Seki 2008] using our Z Score method with the logistic regression for text classification in four different categories (positive, negative, neutral and not opinionated). For NTCIR-8, we used the two-step classification approach, since the size of the training set for each of the opinionated categories was inferior to the size of the not opinionated training set [Zubaryeva 2010b]. In order to improve the obtained classification results of the proposed scheme we used language-dependent tools, specifically, SentiWordNet. Using this vocabulary list with the scores of positivity, negativity and objectivity for each term we were able to obtain the best classification results for the English language in the NTCIR-8 campaign among other participating teams [Zubaryeva 2010b].

In order to compare the performance of the proposed Z score model, we carry out experiments with naïve Bayes and SVM on the NTCIR-6 and NTCIR-7 English corpora. Based on the experiment results, the Z score model outperforms the baseline performance on this dataset. The term feature outperforms the dictionary word representation for the SVM and Z score models. The reduction of the size of the feature space improves the precision of the Z score model.

There could be several reasons for the higher performance of the Z score model. Traditionally, in text classification, SVM and naïve Bayes are used on a selected set of features chosen by one or another combination of association measures. Faced with a shorter text length, in average 22-23 words, and therefore with lower values for frequencies on the training data, SVM and naïve Bayes usually give an average performance of an accuracy of $75 - 85\%$ on a dataset containing movie reviews without the use of natural language specific

methods [Pang 2008]. Due to the small amount of noise given the limited term variety and repetition on a sentence level, the classification based on the computation of the association measures may give similar or better performance than the state-of-the-art approaches.

We also evaluated several information measures, widely used for feature selection in topical text classification, such as odds ratio, information gain, log likelihood and χ^2 statistics. Additionally, we propose a new approach for calculation of a score based on Kullback-Leibler divergence for the opinion and sentiment classification tasks. To the best of our knowledge, this is a new adaptation of the KL divergence based metric for the task of sentiment and opinion classification.

When carrying out experiments based on the computation of information measures for the generated features from text, we noticed different performance of measures depending on the size of the training sets for each category. The degree of the NTCIR corpora represent unbalanced datasets with fewer opinionated and positive documents. The benchmark datasets, such as Movie Review and Subjectivity datasets, on the contrary are balanced, and were constructed in a way so as to have the same size of the training set for each of the classes.

From the classification performance of different weighting schemes based on our experiments and other studies, we choose the Z score that gives one of the best performance on unbalanced datasets. The proposed KL score, on the other hand, outperforms other scores for the balanced datasets.

It is important to note that the feature weighting metrics discussed in this chapter are used differently in sentiment/opinion classification domain than in text classification task. These scores cannot be used to prune the feature set, since the classification item is short and we may eliminate an infrequent, but important feature. The goal of the feature weighting scheme is to identify

topic-specific words in the context of the topical text categorization. While these words are also important in sentiment/opinion classification, they are not the only markers of the opinion. The ideal weighting scheme for the sentiment/opinion classification should not be too strict and weigh highly not only topic-specific words, but also select medium-frequency context words that a lot of times carry the sentiment.

Thus, depending on the size of the classes in the training dataset the opinion/sentiment classification model has to apply the best weighting scheme in order to learn the features or a group of features that convey the sentiment. Solving this problem on a sentence level presents an additional challenge, since even one word can change the polarity of a sentence. In order to create an adaptable model for different types of datasets (balanced and unbalanced) in the presence of limited number of features, we select two scores (KL and Z score) for further analysis and experiments in the next chapters.

Chapter 5

Z Score and KL Score Classification Scheme

5.1 Introduction

The measures discussed in the previous chapter evaluate the features individually, and do not take into account possible dependency based on feature co-occurrences in the sentence. However, their advantage is the possibility to select a set of features that can be indicators that the sentence most probably belongs to a specific category. After analyzing the behavior of the Z score, first we propose to use a normalizing factor (Φ). Second, we give our justification for selecting a set of features with high Z scores, that could indicate by their presence the polarity or opinionatedness of a sentence. Finally, we propose three variations of a classification model and individual analysis of the scores computed for the model. We argue that it is possible to apply the feature weighting measures successfully on the sentence-level opinion classification.

This chapter is organized as follows. Section 5.3 presents the score modification procedures. Then, we describe two methods for combining Z and KL scores based on the error produced on the training set (linear and logarithmic pools). We also present the procedure for choosing the so-called *confident features*, features that could serve as indicators of a specific sentiment expressed. Next, we propose three variations of the classification model in Section 5.3. We analyze the results of the experiments with the classification models and experiments with different negation strategies in Section 5.4 and Section 5.5

respectively. Finally, the summary and discussion for this chapter are given in Section 5.6.

5.2 Score Modification

Analyzing the results of the experiments discussed in the previous chapter, we decided to look closer at the reasons for higher performance of the Z score on unbalanced data sets. In this section, we first present the limitations of the computed Z score statistic for words in text. Given this, we propose a normalization procedure. It takes into account only the frequencies of features in the two categories and allows to reweigh the Z and KL scores. Next, we describe how we combine both scores together using the linear and logarithmic pools. At last, we propose to select so-called *confident features* that receive a high Z score in the specific category.

5.2.1 Limitations of the Z Score

Lafon [Lafon 1980, Lafon 1984] proposed an application of the hypergeometric law for the distribution of terms in the corpus. Labbé *et al.* [Labbé 1994] performed an analysis on the limitations of the hypergeometric scheme, strongly related to the Z score scheme. Here, we relate the conclusions of their analysis. They consider the calculation of the term score for features depending on their frequency and size of the corpus for the classification categories. These conclusions coincide with the observations that we made during the calculations of the Z score on the tested corpora.

In their experiments Labbé *et al.* [Labbé 1994] select features with a score higher than a certain threshold (defined as to represent 5% or 1% of all features). A high score value indicates that the feature is overused in the considered category. Due to the distribution of terms according to the Zipf's law in the corpus, the selection of features in this way is biased towards the features with

high frequency in the corpus. This explains the presence of function words (e.g., preposition, articles) that may receive high score values. Nevertheless, next to the function words, we can find other category-specific terms that have an elevated frequency and could serve as markers of the considered category. Also, the high score permits to select features with relatively small frequencies, but mostly present in one of the categories. Another angle of their analysis concerns the size of the each of the subcorpus according to the classification category. Thus, features receive higher scores for the category with the bigger size [Labbé 1994].

In our experiments we analyzed the features ranked by the Z score coming to the same conclusions for the tested datasets. Thus, the features with the highest Z score are usually function words that have the highest frequency in the corpus, as for example, *it, is, the, a, with*. Nevertheless, in the top and the middle of the list we encountered a lot of domain-specific and opinion-related words that could serve as appropriate descriptors of the category considered. Among these are *sympathetic, controversial, celebrity, heartwarm*, etc.

In this section we propose a classification model where we try to take into account the discussed limitations of the Z score. First, we propose to normalize the value of the Z score in order to give more weight to features that are differently distributed across the two categories. This normalization procedure is discussed in the following section. Second, in our classification model we propose to choose different thresholds for selecting features based not only on the Z score value, but also on their frequency in the corpus. These adaptations are discussed in Section 5.2.4.

5.2.2 Normalizing Z Score and KL Score

Given that we have low frequency counts for the majority of the features in the training set, we can reweigh the scores depending only on the feature frequencies

in both categories. The metrics as KL and Z scores assign higher value to the terms that are more frequent, while underestimating the relative difference in distribution of the feature across the two categories with the growth of frequency counts. In order to give more importance to the difference in how many times a feature is seen in both categories, we introduce a normalization measure Φ.

Given a set of features F, we introduce the following procedure to normalize the Z and KL scores. With the growth of the KL score, $KL^{\Phi}(f_i|c_j)$, a higher discriminative power is attributed to the feature f_i in the class c_j, where $f_i \in F$ and $c_j \in \{pos, neg\}$ (or $\{op, noop\}$). It is obvious that if the feature has equal or close to equal frequencies in both categories, the Φ will be close to 0, otherwise to 1.

$$\Phi = \frac{a-b}{a+b}, \tag{5.1}$$

where a and b are frequencies in the two categories. Taking this into account, we incorporate the Φ measure in the following way:

$$KL^{\Phi}(f_i|c_j) = KL(f_i|c_j) \cdot (1 + |\Phi(f_i|c_j)|) \tag{5.2}$$

Here if a feature is evenly distributed across both categories, a small value of Φ will not produce a big difference in the final $KL^{\Phi}(f_i|c_j)$. The higher the Φ value, the bigger the impact on a final KL score. Thus, we do not boost the features that already have high scores due to their high frequency in the corpus. We augment the scores only of those features that have a high skew in their frequency distribution across the categories.

The range of the Z scores can take negative and positive values. The negative value of the Z score means that the feature is underused in this category. At the same time the positive value means that the feature is overused in the specific category. In order to take this into account we calculate the

normalized Z score in this way:

$$Z^{\Phi}(f_i|c_j) = \begin{cases} Z(f_i|c_j) \cdot (1 + |\Phi(f_i|c_j)|) & \text{if } Z > 0 \text{ and } \Phi > 0, \\ & \text{or if } Z \leq 0 \text{ and } \Phi \leq 0 \\ Z(f_i|c_j) \cdot (1 - |\Phi(f_i|c_j)|) & \text{if } Z > 0 \text{ and } \Phi \leq 0 \\ & \text{or if } Z \leq 0 \text{ and } \Phi > 0 \end{cases}$$

For example, in the Table 5.1 you can observe the features from the Subjectivity dataset with their Z score values and the frequencies in the categories. The final column represents the value of the normalized $Z^{\Phi}(f_i|c_j)$ score.

Feature f_i	$Zscore$	a	b	$Z^{\Phi}(f_i)$
kind	3.57	73	15	5.95
shake	1.95	7	0	3.91
research	−3.32	4	8	−4.69
overact	−0.43	1	1	−0.58

Table 5.1: Normalization of the Z score.

Therefore, if Z score and Φ value are both positive or both negative for a feature f_i, we augment the Z score for the category c_j. If they have different signs, we lower the Z score with the growth of Φ. After obtaining the final KL and Z scores we combine them as described in the next subsection.

5.2.3 Combining Normalized Z Score and KL Score

One of the goals of the proposed model is to provide a framework that would be adaptable to different kinds of training data, ideally in different domains and natural languages. In Chapter 4 we saw that some of the association measures perform better when the training data is balanced (KL score), whereas others show better performance in the imbalanced setup (Z score). Without *a priori* knowledge of the dataset it could be difficult to choose the adequate metric for feature weighting. If we are combining evidence from the two sources it

is important to take into account how each of the experts classify the data. Experts that are similar in how they assess the information provide redundant information [Clemen 1999]. Therefore, the aggregation of these kind of experts gives little gain in the classification performance. From this point of view, a good combination of experts could be achieved if they make mistakes on different testing instances. Experimenting with the Z score and KL score combinations, we hope to achieve an improvement in the classification accuracy.

One possible approach in aggregating scores could be attributing coefficients based on how unbalanced the dataset is or the error of the measure on the training set. Another possibility is to attribute coefficients depending of the error the expert made on the training subset of the corpus. Since we can notice the correlation in the performance of the Z and KL scores depending on how balanced the dataset is, we decided to use the latter approach. Let us have K experts, each expert k assigning a probability of feature f_i given class c_j: $Pr_k(f_i|c_j)$. One of the possible ways to combine the K experts is to use the linear opinion pool [Clemen 1999]. The aggregation of the judgments is done in the following way:

$$P(f_i|c_j) = \sum_{k=1}^{K} \alpha_k Pr_k(f_i|c_j) \tag{5.3}$$

$P(f_i|c_j)$ represents the combined probability distribution, and $Pr_k(f_i|c_j)$ is the probability assigned by kth expert that the feature f_i occurs in the sentence of the class c_j. The weights α_k are non-negative and sum to one. Linear pool is a weighted linear combination of the experts' probabilities. Another possible combination is the logarithm pool:

$$P(f_i|c_j) = C \prod_{k=1}^{K} Pr_k(f_i|c_j)^{\alpha_k} \tag{5.4}$$

where C is a normalizing constant (equal to 1 in our experiments). In our setup where the weight of the decision of each expert (KL and Z scores) is dependent on the data provided, we chose the following way to define the α_k, $k = 1, .., K$. Based on the error that each expert makes during the classification on the training data,

we derive the following weights using the sigmoid scheme as in [Melville 2009]. Therefore,

$$\alpha_k = \log \frac{1 - err_k}{err_k} \tag{5.5}$$

where err_k is the error of the kth expert on the training set. The α_k's are normalized to sum to one. Running a simple classification scheme beforehand can be used to derive the weights. For example, in our experiments we use the $err_{KL} = 0.1$ and $err_{Zsc} = 0.4$ for the Subjectivity dataset (based on the error rates reported in Chapter 4). It is also possible for a data analyst to tune the α_k in the model during the training phase.

Before combining the Z and KL scores using the linear and log opinion pools we normalize the scores in the range of $[0, .., 1]$. Later, we denote both combinations of KL and Z scores as $C \cdot KL$ score. In order to obtain better performance we suggest to remove the outliers from the ranked list of Z and KL scores. Outliers are features that have a very high value of the Z score and KL score since they occur in almost all documents. From our experiments, we had to remove only several (1-3) features that were outliers before normalizing the scores.

5.3 Classification Model Based on Z Score and KL Score

In this section, we present our algorithm and details of the classification model using the statistics calculated on the previous steps.

5.3.1 Selecting Confident Features

In this section we describe how we select features characteristic to the specific classification category. For this purpose we use the Z score computed for each of the categories. This computation permits to identify the features that are "overused" in the considered part of the corpus. The features that we select

with the procedure described in this section, are called *confident features*. They are use to aid the classification and identify other features that may frequently co-occur in their neighborhood.

The higher the frequency of the feature in the training corpus, the more information we have about its distribution across the classification categories. As discussed in Section 5.2.1, the Z score values are generally higher for the frequent features, even if a feature is equally or almost equally distributed across the categories. At the same time, we obtain a high amount of features with low frequency, hence low Z scores. Some of these features can be helpful in classification, if they occur mostly in one of the categories. Let's analyze the behavior of the Z score function for features with low frequencies.

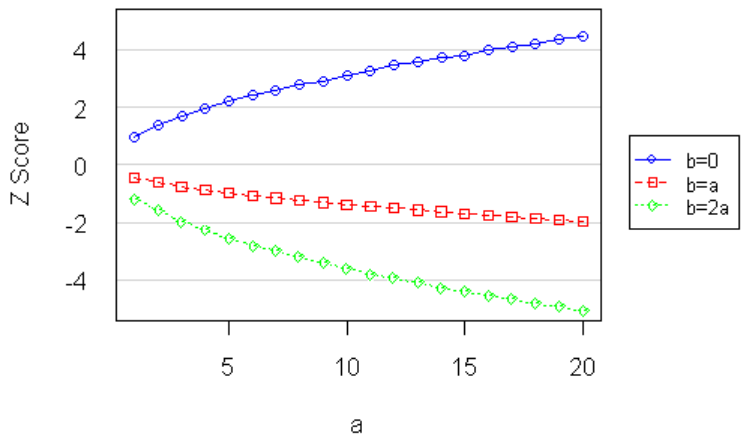

Figure 5.1: Z scores for different a, frequency in the sub corpus, when $a = a + b$, $a = (a + b)/2$, $a = (a + b)/3$.

In Figure 5.1 you can see the behavior of the Z score for different values of the term frequency in the category S, given that all other values are fixed. From the contingency Table 4.1, we vary the values of a and b, frequencies in the two categories, while c and d remain fixed. We remind that $a + c$, and $b + d$ constitute the frequencies of all features in the respective categories. In case of

$b = 0$, the feature is seen only in one category. The corresponding Z score grows as expected with the higher number of feature frequency, indicated by the value a. When a feature is seen equal amount of times in both categories ($a = b$) the Z score actually starts dropping with higher frequency. When a feature is observed only one third of the times in one category ($b = 2a$), the drop of the Z score is faster. Thus, the features with low frequency occurring predominately in one category can receive higher Z scores.

In order to take into account the limitation of the Z score, sensitive to feature frequency, we propose to divide the features based on their frequency values. In the table below we present the statistics on the number of features based on their frequency range for each corpus.

Frequency/ Dataset	**1**	**2**	**3**	**4-9**	**10-100**	**101-1000**	**>1000**
Movie Review	3,357	1,465	952	2,135	1,900	216	8
Subjectivity	3,599	1,367	895	2,122	1,986	217	9
MPQA	2,644	1,192	625	1,636	2,072	317	17
NTCIR OP str.	4,440	1,681	896	1,745	2,131	263	6
NTCIR OP len.	3,492	1,542	836	1,736	2,029	267	7
NTCIR SA	1,803	668	373	886	696	31	0

Table 5.2: Number of features in different frequency classes. Unigram scheme.

In order to facilitate the analysis of the features by their frequency, we present separately the number of features with frequencies equal to 1, 2, and 3. Confirming the Zipf's law, the highest number of features constitute features with very low frequency. In the last four columns, we give the number of features that have a frequency in a certain range. The features with the frequency in the range of $4 - 9$ and $10 - 100$ may be helpful in classification but would receive lower Z score values than features with frequency in the range of $100 - 1000$. The features in the last column repeat across different corpora and are included

in the stop word list, among them are, for example, *it, is, as, with, this, but, be.*

In order to select representative features for each of the categories based on the Z score value, we need to take into account the frequencies of the features in the specific dataset. For example, NTCIR SA dataset due to its smaller size has a different distribution of features across the frequency classes. In our classification model, we propose to divide the features into three classes based on their frequency, infrequent $(2-9)$, frequent $(10-100)$ and very frequent $(101-1000)$. Thus, it would be possible to choose a different threshold for each class in order to reduce the number of features that we select just due to their high frequency in the whole corpus, and not the localization in a specific category.

From the experiments in Chapter 4 we observed that the classification scheme based on the computation of Z score sums is usually biased towards one of the categories (e.g., number of false positives is higher than number of false negatives). Thus, it is possible to modify the threshold value not only for the features in a specific frequency range, but also depending on the category that we consider, e.g., positive or negative.

Depending on the size of the corpus, its contents, topic and style, features can be distributed differently according to their frequencies. Therefore, we experimentally tried different thresholds for each of the testing corpus on the three frequency ranges discussed above. We also took into account the category bias of the classification scheme based on the sum of the Z scores, thus, augmenting the threshold value for the category where we make the highest number of errors.

Based on the experiments, for the unbalanced datasets we chose a threshold of 0 for the classification category with the smallest size (positive and opinionated categories). In order not to choose out of all possible thresholds for each of the datasets, we decided to base the decision on the frequency

difference of the feature in the whole corpus and in the specific category. Therefore, we can select the features that are more frequent in the specific category than a certain threshold, without regarding the value of the Z score itself.

As a threshold we put a condition that a feature is seen more than certain percentage of times in a specific category from its total frequency in the corpus. Thus, in the Table 5.3 we present the condition for a feature to be chosen as a confident feature for a specific category if its frequency in the category is higher than a certain percent from its total frequency in the corpus. These thresholds were empirically chosen for the each of the frequency ranges. The set of features selected with the chosen thresholds is called a set of *confident features*.

Dataset/Frequency	$2-9$	$10-100$	$101-1000$
Balanced Op/Pos	$0.85*f^T$	$0.65*f^T$	$0.85*f^T$
Balanced Noop/Neg	$0.85*f^T$	$0.6*f^T$	$0.75*f^T$
Unbalanced Op/Pos	0	0	0
Unbalanced Noop/Neg	$0.85*f^T$	$0.65*f^T$	$0.85*f^T$

Table 5.3: Experimentally selected thresholds for selecting *confident features* for each classification category. f^T - total feature frequency in the corpus.

If we rank the list of Z scores of features for one category, we can observe that the features, occurring more times in the opposite category, receive a higher negative Z score, while the ones that are overused in the same category do not get equally high scores. For example, a feature *episode* in the Movie Review dataset has a frequency of 28 in the negative sub corpus (negative training sentences) and 6 in positive. The Z score of this feature in the negative category is 2.33, while in the positive -10.04.

Thus, specifying the threshold based on the feature frequency limits the proposed selection procedure by the characteristics of a particular dataset in consideration. The range of feature frequencies should be specified based on

the size of the training data, since this value can change with the growth of the training set.

Movie Review Dataset	
Positive Confident Features	**Negative Confident Features**
engross, portrait, refreshingly, examine, rich, wonderful, mesmerize, heartwarm, gem, cinema, enjoy, spare, vividly, solid, beauty, terrific, spielberg, iranian, refreshing	bore, drag, pointless, clumsy, jackass, horrible, bogus, mute, rumor, ramble, vague, repetitive, suspect, embarrass, devoid, unfortunately, pseudo, incomprehensible

Table 5.4: Examples of *confident features* for Movie Review dataset.

As you can see from the Table 5.4, the features selected for the specific category as confident features are representative of the polarity of the sentiment. For example, a movie review containing words as *bore* or *embarrass* would most probably belong to the negative category. Our motivation is to select such opinion/sentiment markers from text, so that we can assume with a higher degree of confidence (since they have bigger scores) that they pertain to the specific category. The idea is to identify such features and their frequent neighbors to enrich the information given to the model. We expect that the frequent neighbors of the *confident features* would be those features that were just not frequent enough to be selected in the *confident set*.

The obtained set of confident features denoted F_{conf} we use in two different ways: first, as the sentiment markers of the category, and second, to calculate other scores based on feature co-occurrences with features from F_{conf}. The details of the classification procedure using selected confident features and the normalized KL and Z scores are given in the next section.

5.3.2 Classification Approaches

In this section we present the statistical scores as well as three classification procedures using these scores. First, we try to take into account the dependency between the features. Based only on confident features, we have a simple text representation that ignores the position of each term in the sentence as well as its neighbors. For example, the fact that *almost* is followed by *make* or *give* by *up* is ignored, but one of the features in the pair may change the underlying meaning. To take account of this local proximity we extract the neighbors of each confident feature. More precisely, we consider three terms before and three after confident features. The motivation for this selection lies in the observation that many sentences contain phraseological expressions and features that influence the sentiment polarity of the whole sentence. The bigram indexing scheme does not always capture these expressions. Moreover the addition of a large set of features with low frequency (in case of bigrams) lowers the model's classification accuracy. Let's consider the following example:

> "It acknowledges and celebrates their cheesiness as the reason why people get a kick out of watching them today."

Here *kick out* is an idiomatic expression that has a different meaning and sentiment polarity than the word *kick*. In order to capture possible expressions and significant co-occurrences of other features with the confident feature, we use the Information Gain (IG) ratio (also called *expected mutual information*). Similar to Table 4.1, we can introduce a contingency table for two features f_{conf}, f_n as shown in Table 5.5.

	f_n	$\overline{f_n}$	
f_{conf}	a	b	$a+b$
$\overline{f_{conf}}$	c	d	$c+d$
	$a+c$	$b+d$	$n=a+b+c+d$

Table 5.5: Example of a contingency table for two features.

To compute the IG ratio between two features, we use the following formula.

$$\begin{aligned} IG(f_{conf}|f_n) &= -\sum_{f_k\in\{f_n,\overline{f_n}\}} P(f_k)\log P(f_k) + \\ &\quad +P(f_{conf})\cdot\sum_{f_k\in\{f_n,\overline{f_n}\}} P(f_k|f_{conf})\log P(f_k|f_{conf}) + \\ &\quad +P(\overline{f_{conf}})\cdot\sum_{f_k\in\{f_n,\overline{f_n}\}} P(f_k|\overline{f_{conf}})\log P(f_k|\overline{f_{conf}}) \end{aligned} \tag{5.6}$$

Using the notation in Table 5.5 we can estimate $P(f_n) = (a+c)/n$, $P(\overline{f_n}) = (b+d)/n$, $P(f_{conf}) = (a+b)/n$, $P(f_n|f_{conf}) = a/(a+b)$, $P(\overline{f_n}|f_{conf}) = b/(a+b)$, $P(\overline{f_{conf}}) = (c+d)/n$, $P(f_n|\overline{f_{conf}}) = c/(c+d)$, and $P(\overline{f_n}|\overline{f_{conf}}) = d/(c+d)$. Thus, in terms of the contingency table we get:

$$\begin{aligned} IG(f_{conf}|f_n) &= -\frac{a+c}{n}\log\frac{a+c}{n} - \frac{b+d}{n}\log\frac{b+d}{n} \\ &\quad +\frac{a+b}{n}\left(\frac{a}{a+b}\log\frac{a}{a+b} + \frac{b}{a+b}\log\frac{b}{a+b}\right) \\ &\quad +\frac{c+d}{n}\left(\frac{c}{c+d}\log\frac{c}{c+d} + \frac{d}{c+d}\log\frac{d}{c+d}\right) \\ &= -\frac{a+c}{n}\log\frac{a+c}{n} - \frac{b+d}{n}\log\frac{b+d}{n} + \frac{a}{n}\log\frac{a}{a+b} \\ &\quad +\frac{b}{n}\log\frac{b}{a+b} + \frac{c}{n}\log\frac{c}{c+d} + \frac{d}{n}\log\frac{d}{c+d} \\ &= \frac{a}{n}\log\frac{a\cdot n}{(a+b)(a+c)} + \frac{b}{n}\log\frac{b\cdot n}{(a+b)(b+d)} \\ &\quad +\frac{c}{n}\log\frac{c\cdot n}{(a+c)(c+d)} + \frac{d}{n}\log\frac{d\cdot n}{(b+d)(c+d)} \end{aligned} \tag{5.7}$$

When the IG ratio value is close to zero, we cannot detect a significant association between the two features. A positive value tends to indicate an association between the two terms.

Let's recapitulate the information that we calculate for the classification model. For each sentence, we compute the six statistics (1-6) presented above in the two possible categories (opinionated vs. factual or positive vs. negative). The

first three sums are directly used in the two classification approaches presented later in this section. The last three sums we use in the experiments to evaluate their individual performance and impact with the simple sum classification.

1. $\sum_{f_i \in Conf} Z^{\Phi}(f_i|c_j)$ - the sum of Z scores of all confident features
2. $\sum_{f_j \in Neigh(f_i)} C \cdot KL^{\Phi}(f_i|c_j) \cdot (1 + IG(f_i, f_j))$ - the sum of $C \cdot KL$ scores boosted by the IG score for neighbors of confident features
3. $\sum_{f_i \in \overline{Conf}} C \cdot KL^{\Phi}(f_i|c_j)$ - the sum of combined Z and KL scores of all features not in the confident set (linear or log pool combination)
4. $SumIG^{Conf} = \sum_{f_i \in Conf, f_j \in Neigh \cup s} IG(f_i|f_j)$ - the sum of IG scores of any confident feature with any of its neighbors that are present in the sentence s
5. $Support^{Conf}$ - the number of confident features that are neighbors of any feature present in the sentence
6. $SumZ \cdot IG = \sum_{f_i \in Conf} Z^{\Phi}(f_i|c_j) + SumIG^{Conf} \cdot Support^{Conf}$ - the sum of the Z scores of the confident features present and the product of $SumIG^{Conf}$ and $Support^{Conf}$

Table 5.6: Scores computed for the sentence s, $c_j \in \{pos, neg\}$ (or $\{op, noop\}$).

$SumIG^{Conf}$ represents the sum of the IG scores of neighbor features, present in the sentence, and all of their confident features, while $Support^{Conf}$ is the number of confident features for these neighbors. Therefore, we take into account how the features, present in the sentence, are influenced by the confident features in the neighborhood of which they occur in the training corpus.

The computation complexity of the steps above is $O(|F|^2)$, where $|F|$ is the number of features f_i in the category. The quadratic complexity is due to the calculation of the sum of the IG score between confident features and their neighbors. Since we are generalizing the number of confident features and neighbors the actual computation time is lower. Note that the calculation of the KL and the Z scores is done in the first pass over the feature set. We can store the neighbors of each feature in advance and calculate the $(1 + IG(f_i, f_j))$ in the

second pass after determining the threshold for the confident features. At the same pass we calculate the linear or log pool combination of the Z score and KL score, denoted by $C \cdot KL$, using their error rates on the specific corpora. This way, at the training stage we can do the calculation steps described above and store the scores for each feature in a file.

Algorithm 1 Classification Procedure 1 (CP 1)

Require: set of sentences S: s_i, $i \in 1, .., |S|$

Ensure: class c_j, $j \in \{pos, neg\}$ or $\{op, noop\}$

for s_i in $|S|$: **do**
  if $N^{Conf} \geq 2$ in s_i **then**
    return $\arg\max_{c_j} \sum_{f_i \in Conf} Z^{\Phi}(f_i|c_j)$
  end if
  if s_i contains neighbors of confident features **then**
    return $\arg\max_{c_j} \sum_{f_i \in Conf} \sum_{f_j \in Neigh(f_i)} C \cdot KL^{\Phi}(f_i|c_j) \cdot (1 + IG(f_i, f_j))$
    $+ \sum_{f_i \in \overline{Conf}} C \cdot KL^{\Phi}(f_i|c_j)$
  else
    return $\arg\max_{c_j} \sum_{f_i \in \overline{Conf}} C \cdot KL^{\Phi}(f_i|c_j)$
  end if
end for

Using the first three statistics in Table 5.6, and taking into account that N^{Conf} is the number of confident features in the sentence, we present the first classification procedure (CP 1). The proposed classification model is based on simple computations of sums and makes the following assumptions. We assume that confident features are good indicators of the category by themselves. If the number of confident features N^{Conf} is less than two, we use the KL score combination, boosted for the frequent neighbors of confident features if they are present in the sentence. This boost accounts for inter-feature dependency when we take into consideration the frequency of co-occurrence of the neighbor with a confident feature or features. We consider only the case when there are two and more confident features are present in the sentence. If we consider the

occurrence of just one confident feature, the results degrade. We cannot base our decision solely on one feature and need to take into account the surrounding text.

Algorithm 2 Simplified Classification Procedure 1 (SCP 1)

Require: set of sentences S: s_i, $i \in 1, .., |S|$

Ensure: class c_j, $j \in \{pos, neg\}$ or $\{op, noop\}$

1: **for** s_i in $|S|$: **do**
2: **if** $N^{Conf} \geq 2$ in s_i **then**
3: **return** $\arg\max_{c_j} \sum_{f_i \in Conf} Z^{\Phi}(f_i|c_j)$
4: **else**
5: **return** $\arg\max_{c_j} \sum_{f_i \in |F|} KL^{\Phi}(f_i|c_j)$
6: **end if**
7: **end for**

Based on CP 1, we propose a simplified classification procedure (SCP 1) that as input takes in $\sum_{f_i \in Conf} Z^{\Phi}(f_i|c_j)$ and $\sum_{f_i \in \overline{Conf}} KL^{\Phi}(f_i|c_j)$ and does not use any information about the neighbor features. Therefore, the SCP 1 model takes into account the presence of confident features, otherwise uses the KL^{Φ} score to perform classification.

To illustrate the simplified procedure, SCP 1, we present a positively opinionated sentence: *Magnificent drama well worth tracking down* from the Movie Review dataset that we analyze in the Table 5.7. As we can see from the table we have two scores for the two confident features found in different categories: *worth* and *track*. If we check the sum of the confident scores the classification of the sentence would be correct. Since there are two confident features present, the model compares the sums of the Z^{Φ} scores. We can also see that the model correctly classifies the sentence by comparing the sum of the KL^{Φ} scores for both categories (14.44 for positive and 12.92 for negative category).

	$Z^{\Phi}(f_{Conf}\|c)$		$Z^{\Phi}(f_i\|c)$		$KL^{\Phi}(f_i\|c)$	
f_i/c_j	c_{pos}	c_{neg}	c_{pos}	c_{neg}	c_{pos}	c_{neg}
magnificent	-	-	0.38	-2.19	0.66	0.32
drama	-	-	-1.43	-7.7	3.67	3.22
well	-	-	-3.39	-8.17	4.62	3.36
worth	1.17	-	1.17	-7.68	2.73	1.91
track	-	1.03	-2.77	1.03	0.24	0.79
down	-	-	-7.72	-0.01	2.52	3.32
Sum	1.17	1.03	-13.76	-24.72	14.44	12.92

Table 5.7: Scores computed with SCP 1 for the example sentence, $c_j \in \{pos, neg\}$.

After carrying out the experiments using the CP 1 presented above on the corpora from the movie review domain, we remarked that depending the sentences could be roughly divided into two groups. To the first group we attribute the sentences that present the opinion or sentiment in a straightforward manner. By this we mean that there are no comparisons, contrasting statements or expressions that may change the opinion polarity. The sentences that possess the latter characteristics we attribute to the second group. We present several examples of the sentences in question in Table 5.3.2.

One of the main characteristics of the sentences in the second group is often the presence of both sentiments by the means of comparison or contrasting statements. In order to identify the features that characterize these sentences we used the SentiWordNet lexicon [Esuli 2006b]. We extracted the features in the sentence that have positive and negative polarity scores equal to 0 and occur between the features with opposite polarity scores according to the SentiWordNet. After extracting these features we ranked them according to their frequency. Out of these features we manually chose a set of frequent terms, that we use to identify sentences belonging to the second group. These terms are: *but, if, though, although, while, despite, so, just, still, yet, even, only, than,*

ultimately, ever, what, who, when, either or, since, than, must. Let us call these terms *valent terms* and denote as T^{val}.

Group 1
1. A static and sugary little half-hour, after-school special about interfaith understanding, stretched out to 90 minutes. 2. The tenderness of the piece is still intact. 3. A feel-good picture in the best sense of the term.
Group 2
4. **While** it would be easy to give Crush the new title of two weddings and a funeral, it's a far more thoughtful film than any slice of Hugh Grant whimsy. 5. **Though** everything might be literate and smart, it never took off and always seemed static. 6. This may not have the dramatic gut-wrenching impact of other holocaust films, **but** it's a compelling story.

After dividing the testing sentences into two groups, we perform the CP 1 as described previously for the sentences in the first group. For the sentences in the second group we apply the following scheme. For each sentence s_i in the *Group 2* we generate a set of sentences S'. This set is obtained by excluding the valent term, its neighbors from the right and from the left, adding a new sentence s'_k to S' after each exclusion.

After obtaining the set S', we perform a Classification Procedure 1 on each sentence. As the final classification category we take the category with the majority vote obtained on the generated sentences S'. Here is an overview of the Classification Procedure 2:

Algorithm 3 Classification Procedure 2 (CP 2)

Require: set of sentences s_i, $i \in 1, .., |S|$, valent terms t_v, $v \in 1, .., |T^{val}|$

Ensure: class c_j, $j \in \{pos, neg\}$ or $\{op, noop\}$

Ensure: generated sentences s'_k, $k \in 1, .., |S'|$ from S

1: **for** s_i in S, where s_i contains t_v, $t_v \in T^{val}$: **do**
2: **for** t_v in T^{val}: **do**
3: $s'_k = s_i \backslash t_v$, $S' = S' \cup s'_k$
4: determine max 2 features before and after t_v: $NeighsLeft(t_v)$, $NeighsRight(t_v)$
5: $s'_k = s_i \backslash NeighsLeft(t_v)$, $S' = S' \cup s'_k$
6: $s'_k = s_i \backslash NeighsRight(t_v)$, $S' = S' \cup s'_k$
7: **end for**
8: **for** s'_k in S': **do**
9: apply Classification Procedure 1 on s'_k, store the c_j
10: increase the class counter $Cnt(c_j)$
11: **end for**
12: **return** $\arg\max_{c_j} Cnt(c_j)$
13: **end for**

This procedure generates a set of sentences, each of which contains less features than the original sentence. Thus, we try to exclude those features close to the valent terms that could be of different polarity than the sentence itself and hinder the accurate classification decision. We present the results and analysis of the two classification procedures and the classification approaches based on statistics in Table 5.6 in the next section.

5.3.3 Error and Misclassification Analysis

In order to have a better understanding of our underlying classification scheme, we have conducted a failure analysis of several examples from the Movie Review dataset. Of course, the most interesting cases are the sentences that were mis-

classified by our model. When inspecting them, we can see that most of the classification errors are related to the underlying ambiguity of the natural language. In the following examples, we have first presented sentences that were incorrectly labeled as negative (1-3). In the second part (sentences 4-7), we can find sentences incorrectly classified as positive by our model.

Positive sentences classified as negative.

1. Longley has constructed a remarkably coherent, horrifically vivid snapshot of those turbulent days.
2. Romanek keeps the film constantly taut... Reflecting the character's instability with a metaphorical visual style and an unnerving, heartbeat-like score.
3. Compelling revenge thriller, though somewhat weakened by a miscast leading lady.

Negative sentences classified as positive.

4. In the book-on-tape market, the film of "the kid stays in the picture" would be an abridged edition.
5. A mechanical action-comedy whose seeming purpose is to market the charismatic Jackie Chan to even younger audiences.
6. It's not so much a movie as a joint promotion for the national basketball association and teenaged rap and adolescent poster-boy lil' bow wow.

As you can see, these reviews would be difficult for an automatic classification model in several ways. First, they are characterized by the use of highly positive or negative words to express or intensify a completely opposite polarity, as we can see in the first three examples.

Another concern is when the sentence (e.g., Sentence #4) does not contain any overtly negative features, but nevertheless expresses a negative opinion by the means of the verb *abridged*. The fifth sentence gives a weak clue of negativity with the use of terms *mechanical* and *seeming*, while containing a highly positive *charismatic* feature. The use of slang expressions, such as

bow − wow, and a negative connotation of the term *promotion*, are also difficult to detect correctly. In this case, the first expression is quite infrequent in the corpus and the latter is mostly neutral by itself in its polarity (Sentence #6).

The last sentence represents a subset of misclassified examples from our observation where one part of the sentence displays an abundance of positive terms, while the other uses only several negative terms. In the current case, the not so common phraseological expression *shooting blanks* is the main reason for the misclassification of this review. All these examples demonstrate the complexity of the natural language and the need for developing language specific heuristics to better capture phraseological expressions, contrasting statements, sarcasm, and allusions made by the writer.

5.4 Experimental Results

We carried out experiments on all six datasets described in Chapter 3. First, we present the statistics computed for each category per each dataset. Next, we present the results and analysis of the carried out experiments.

5.4.1 Corpus Statistics

In order to take into account peculiarities of each dataset, we performed the analysis as to the number of features, distinct features and selected confident features for each corpus. Thus, Table 5.8 shows different statistics on the six corpora tested. As you can see the values for the dataset statistics vary from corpora to corpora. We can see that the bigger the number of distinct features in a category, the bigger is the number of selected confident features. This makes sense, as first of all, the features that occur only or are prevalent in one category would get selected as *confident features*.

Overall, we can notice that the length of the sentences tends to be bigger for the newspaper corpora than for the movie reviews. Another important issue

Corpus	N_D	N_F	N_F^{Dist}	$\overline{N_F}$	$\overline{N_F^{Dist}}$	N_F^{Conf}
Movie Review Pos	5,331	116,080	20,370	21.77	20.23	2,825
Movie Review Neg	5,331	116,176	21,052	21.79	20.3	3,573
Subjectivity Pos	5,000	129,316	22,790	25.86	23.58	3,024
Subjectivity Neg	5,000	119,069	21,651	23.81	22.08	3,075
MPQA Op	6,123	180,947	161,211	29.55	26.33	1,996
MPQA Noop	4,989	125,304	113,155	25.11	22.68	2,304
NTCIR OP str. Op	598	17,162	15,501	28.69	25.92	876
NTCIR OP str. Noop	9,548	237,728	215,581	24.89	22.58	7,439
NTCIR OP len. Op	2,466	68,847	62,190	27.91	25.21	1,442
NTCIR OP len. Noop	7,680	186,043	168,892	24.22	21.99	4,245
NTCIR SA Pos	679	19,384	17,379	28.54	25.59	836
NTCIR SA Neg	1,332	36,957	33,548	27.74	25.18	1,420

Table 5.8: Corpus statistics. N_D - number of documents, N_F - number of features, N_F^{Dist} - number of distinct features, $\overline{N_F}$ - mean number of features per sentence, $\overline{N_F^{Dist}}$ - mean number of distinct features per sentence, N_F^{Conf} - number of confident features.

is the inequality in the training sets for different NTCIR datasets. The first five datasets have about $10,000$ sentences each. NTCIR SA dataset has only around two thousand sentences. The most unbalanced dataset in terms of the number of the training sentences per category is the NTCIR OP strict dataset, that has only 598 sentences for the positive class and $9,548$ sentences for the not opinionated category. The NTCIR OP lenient and NTCIR SA datasets are much less unbalanced. The positive category of the NTCIR SA has the size that is around 51% from the size of the negative category and for the NTCIR OP lenient dataset the opinionated category is 32% the size of the factual category.

The inequality in the number of training sentences can lead to problems

when we over fit the model for only one category. As the proposed model is based on the computation of the Z and KL scores, the weights of the features are influenced by their frequency in the corpus and in the specific category. Therefore, for the unbalanced datasets, the number of selected confident features as their weights would favor the bigger category. For example, with the use of the 10 fold cross-validation in the NTCIR OP strict dataset we would have only 539 opinionated sentences to train on, compared to 8,594 factual training sentences.

The Movie Review, Subjectivity and MPQA datasets were constructed as benchmark datasets for sentiment and opinion classification tasks, while the NTCIR datasets are unbalanced, but more representative of the distribution of the opinionated sentences in the newspaper domain. This division of the training data presents more realistically the number of opinionated sentences in newspaper texts.

5.4.2 Experiments and Analysis

We carried out experiments with the proposed classification procedures, as described in Section 5.3.2. For the classification procedure CP 1 we took into account the log and linear pool combination of the normalized Z and KL scores. For the second classification procedure we used just the linear pool for the combination of normalized Z and KL scores. We carried out extensive experiments with different combinations of sums presented in Table 5.6. These combinations give different performance dependent on the dataset considered. We were not successful at creating the generalized classification model that would generally achieve higher performance compared with the simple classification schemes that are based on the computation of the sum of scores for the sentence. Thus, along with the classification procedures proposed in the previous section, we present the classification results of the statistics computed on the training corpus.

These statistics include the sum of IG scores of any feature present in the sentence with any confident feature ($SumIG^{Conf}$), the number of confident features that are neighbors of the features present in the sentence ($Support^{Conf}$), the sum of the normalized Z scores of the confident features present and the product of the previous two sums ($SumZ \cdot IG$). Finally, there is the sum of the Z^{Φ} scores of all confident features ($SumZ^{\Phi}$).

Approach	**Movie Review†**	**Subject.†**	**MPQA†**	**NTCIR SA⋆**	**NTCIR OP len⋆**	**NTCIR OP str⋆**
SCP 1	77.68%	**91.55%**	73.28%	55.61%	56.78%	51.66%
CP 1, lin. pool	76.44%	90.61%	69.01%	69.74%	65.11%	**57.73%**
CP 1, log pool	76.37%	90.45%	68.29%	69.25%	64.88%	54.07%
CP 2, lin. pool	**79.82%**	89.52%	71.62%	63.49%	60.81%	53.92%
$SumIG^{Conf}$	66.72%	83.64%	69.07%	67.12%	64.07%	55.69%
$Support^{Conf}$	62.64%	80.14%	55.79%	59.36%	55.28%	50.64%
$SumZ \cdot IG$	72.49%	90.17%	**73.87%**	**70.32%**	62.77%	54.08%
$SumZ^{\Phi}$	70.76%	84.64%	69.22%	70.24%	**68.34%**	53.72%

Table 5.9: Accuracy† and F1-measure⋆ of the proposed classification models using *Wise Tokenizer* scheme with 10-fold cross-validation over the six corpora.

In Table 5.9 you can see the results for the *Wise Tokenizer* setup (with stemming and stop words removal). The results on the unigram, bigram and n-gram indexing setups are given in the Appendix A. Note, that for the unbalanced datasets, we use the macro-averaged F1-measure instead of accuracy, as for the first three datasets. The results show that the use of the *Wise Tokenizer* pre-processing approach permits to obtain better results together with the investigated classification approaches. The unigram pre-processing gives close results to the *Wise Tokenizer* scheme. It is interesting to note that the *n-gram* scheme usually outperforms the bigram pre-processing, nevertheless both schemes give lower results than the unigram approach. Overall, schemes as unigram and

Wise Tokenizer do not generate as many features as the bigram and *n-gram* schemes, thus, performing better with the simple classification approaches based on feature weights. In our opinion, generation of more features than there are tokens in the sentence may produce noise for the model, where one word can contribute several times to the final sentiment score.

The combination of Z and KL scores using the log and linear pools generally does not give the amelioration in the performance for the balanced datasets. Nevertheless, we can see an amelioration in the performance for the CP 1 and CP 2 on the unbalanced datasets (NTCIR SA, OP lenient and strict). This may be due to the fact that the normalized scores favor the features which occur mostly in the category with the smaller size in terms of the number of the training sentences.

Since the error of the Z score is lower than the KL score on the unbalanced datasets, the final normalized $C \cdot KL$ score (used in CP 1 and CP 2) is more influenced by the Z score value. Hence, the terms overused in the opinionated (positive) category will receive higher Z score value, and therefore higher final $C \cdot KL$ score. The classification procedure based on using valent terms (CP 2) gave an amelioration only on the Movie Review dataset. This result can be due to the nature of the collection. From our experiments around half of the sentences in the Movie Review dataset contain one or more valent terms. Thus, the heuristic on generating a set of sentences without the valent term and its neighbors eliminates the parts of a sentence that may contain terms with opposite sentiment. On the other datasets this strategy did not bring any amelioration.

It is interesting to analyze the behavior of each of the computed statistics by itself without any classification procedures. This way, we can notice that on the unbalanced datasets they achieve a higher F1-measure than the proposed classification procedures. Taking into account the IG scores between the sentence

features and their neighbors that are confident features ($SumIG^{Conf}$) gives better results that taking into account just the number of these neighbors ($Support^{Conf}$). The normalized Z score, $SumZ^{\Phi}$, gives better results than the classification based on the simple Z score (see Chapter 4, Section 7).

The normalization with Φ (CP 1 and CP 2) has clearly much more effect on the result for the balanced datasets, where the increase in macro-averaged F1-measure is more than 5%. Finally, $SumZ \cdot IG$, which is the combination of the sum of the Z^{Φ} scores for the confident features with the product of $SumIG^{Conf}$ and $Support^{Conf}$, gave better results than the schemes by themselves.

Generally, from the experiments that were carried out, it turned out quite difficult to construct a classification scheme with the combination of several statistics that would give a high performance for all considered corpora and classification setups. From our point of view, any adaptation of the metric for feature selection or classification should take into account the size of the training data set per category, the elimination of very frequent features that could disturb the classification accuracy, the inter-dependency between the features in the specific category. Each of these factors can influence the choice of the statistics used in the classification scheme.

For some datasets, such as Subjectivity dataset, the KL score gives a high accuracy results by itself, other statistics do not bring any significant amelioration. At the same time, other datasets, especially if they unbalanced, profit from the combination of scores that take into account the influence of the size of the training corpora on the distribution of the statistical scores. With the linear or logarithmic pool combination of the Z and KL scores it is possible to normalize and reweigh the feature scores according to the training dataset at hand. In order to reduce the influence of the high frequency on the score values, we adopt a normalization procedure using Φ measure. These adaptations to the calculation of the feature scores can already improve the results of the

comparatively simple classification schemes.

We can see that indeed, as observed in Chapter 4, the scores complement each other and the proposed combination gives an increase in accuracy compared to the simple sum classification for the balanced datasets. Perhaps, a more thorough investigation in the choice of weights is needed in order to achieve better score combination on the unbalanced datasets.

In our opinion, it is important to take into account how the scores are normalized and combined depending on their frequency in the whole dataset and in the specific category. The main conclusion of this part of our experiment is that using standard stemmers and indexing techniques for English, with no additional sentiment lexicons, we were able to obtain comparable results to the approaches using SVM, naïve Bayes or human-annotated lexicons on the benchmark corpora.

We argue that the appropriate choice and combination of measures, given the classification task on the sentence level, can be used to obtain a set of scores for the model's features that could be used to derive adaptable classification schemes or for further feature selection. The main advantages of the proposed computational approach are the state of the art performance on the benchmark corpora, easiness of result analysis, and the extensibility. The latter includes the possibility to to take into account other scores and statistics, as well as word scores given in sentiment lexicons.

It is also possible to user the obtained feature scores as input parameters for other models. The baseline approaches, like SVM and naïve Bayes, usually require more data to build an effective classifier, unless they are using language-specific tools or sentiment lexicons. The main reasons why SVM quickly reaches a ceiling in performance on the tested data are presented in Chapter 6.

5.5 Experiments with Negation

Negation could be a very important component to take into account when performing sentiment classification. One can differentiate negation by its influence on the sentence or its components. Thus, a simple particle *not* can refer to the whole idea expressed in the sentence or a particular clause or phrase. Another type of negation is introduced by conjunctions at the beginning or the end of the clause. For example, let's take a sentence *Although bright, well-acted and thought-provoking, Tuck Everlasting suffers from a laconic pace and a lack of traditional action.* This is a sentence from the Movie Review dataset that introduces the positive evaluation with *although*, but expresses a negative opinion overall. It is also possible to observe the negation of single words, for example, with the prefix *un-* as in *unpleasant*. The expression of negation in the sentence can be realized in a lot of ways which are difficult to easily pinpoint without any prior language-specific techniques.

Several studies investigate taking into account the negation in the sentence in order to improve the classification accuracy. Thus, Wilson *et al.* [Wilson 2005b] report the variety of different negation expressions. A lot of studies use POS tagging to determine negation patterns. Na *et al.* [Na 2001] report only around 2% amelioration for the text classification task with the additional use of the negation check.

In order to try to improve our experimental results on the proposed model, we investigated the detection of the negation in the text. We suggest to analyze two forms of the negation, lenient and strict negation checks. First, if the negation word was detected the polarity orientation of the words that follow it was changed. As negation words for the strict negation check we took the particle *not*, and negative forms of verbs *don't, doesn't, can't, wouldn't*, as well as *never* and *noone*. For the lenient check we added the conjunctions *although, though, while, but, however, nevertheless, even though, unless, even if, despite*. If a negation word was found in the sentence the scores for all features occurring

after the word till the next punctuation mark were attributed to the opposite category. This way, we tried to take into account sentences that use complex structures with the negation clauses and words.

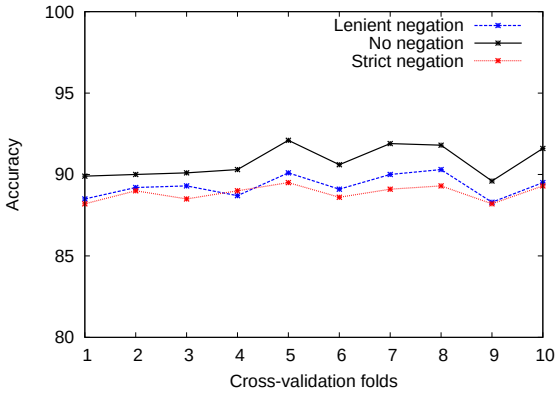

Figure 5.2: Negation strategies for the Subjectivity dataset.

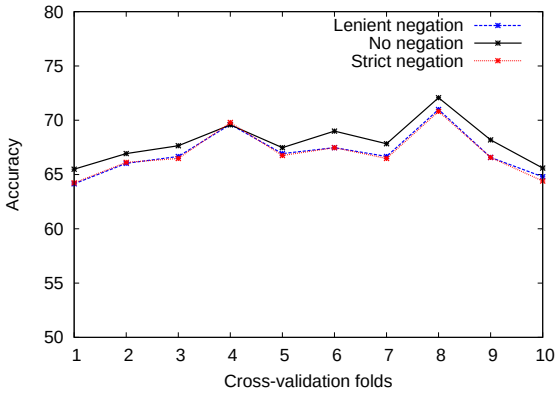

Figure 5.3: Negation strategies for the MPQA dataset.

In Figures 5.3 and 5.2 we show the accuracy over the ten folds for the proposed simplified classification procedure (SCP) with no negation check, lenient and strict negation checks. It is possible to see that the negation strategies adopted for the model lead to the deterioration in accuracy. For the MPQA dataset the lenient negation that also includes conjunctions further deteriorates the result. This could be explained by the fact that the newspaper corpora does not contain sentences similar to the movie review domain with different comparisons and clauses.

Overall, the results also deteriorate with the use of the negation checks for the movie review domain. We can see that the strict and lenient negation performs on the same level for the Subjectivity dataset. This is an indication that we should ameliorate the methods used, possibly with the use of POS tagging. Performing the misclassification analysis on the sentences misclassified with and

without negation check, we noticed the following reasons for committing errors.

First, it has to be noted that the presence of the negation in the sentence does not always indicate that the specific negation is used to reverse the sentiment polarity expressed. Most of the time, especially in short or one-sentence reviews, the negation constructions can be used to underline the sentiment or to contrast the final opinion with the reader's expectation. Second, the negation can refer to some aspect of the movie, for example, "The acting is fine but the script is not interesting". Consider the following sentences misclassified due to the use of the negation checks:

Positive classified as negative.

1. It's not so much enjoyable to watch as it is enlightening to listen to new sides of a previous reality, and to visit with some of the people who were able to make an impact in the theater world.
2. It's almost impossible not to be moved by the movie's depiction of sacrifice and its stirring epilogue in post-soviet Russia.
3. Although it bangs a very cliched drum at times, this crowd-pleaser's fresh dialogue, energetic music, and good-natured spunk are often infectious.

Negative classified as positive.

4. It's not nearly as fresh or enjoyable as its predecessor, but there are enough high points to keep this from being a complete waste of time.
5. So genial is the conceit, this is one of those rare pictures that you root for throughout, dearly hoping that the rich promise of the script will be realized on the screen. it never is, not fully.

Finally, our recommendation is to perform POS tagging for further semantic analysis of the sentence in order to use the negation information in the classification model. It is clear from the experiments carried out that a more sophisticated analysis as to the semantic structure of a text is needed to perform a successful negation check in the sentence. It is important to determine if the negation refers

to the sentiment expressed, and then how exactly it affects the sentiment: lowers its intensity or changes it.

5.6 Summary and Discussion

The Z score and the proposed adaptation of the KL-divergence score to sentiment and opinion classification tasks present two measures that weigh differently the features based on their categorical distributions. The experimental results confirm our intuition, based on the analysis performed in Chapter 4, that the two metrics can be used complementary to achieve higher accuracy for the sentence-level classification task. Since, the two metrics are asymmetrical, the model attributes different weights depending on the classification category. This permits a construction of a simple heuristic classification schemes, as proposed in Section 5.2.

In this chapter, we proposed a classification method that: a) combines the feature selection metrics evaluated previously, KL and Z scores normalized by Φ, b) uses interdependency information between the set of selected features computed with the use of IG, and c) identifies *valent terms* and reweighs the sentence category scores. We use SentiWordNet to identify these terms taking into account their frequent occurrence between the terms with the opposite sentiment polarity. The main motivation for this choice is to take into account the review sentences containing clauses with opposite sentiment polarities.

The experiments with the detection of negativity in sentences and changing the sentiment polarity orientation of the words did not give any improvement on the evaluated datasets. One of the main reasons for this is the negation of some other idea or proposition expressed in text, and not the sentiment itself. Another reason is the inability to take into account grammatical constructions and semantic analysis of the sentence in the proposed approaches.

After performing the failure analysis, we find that a lot of times the difficulty of the dataset influences the performance of the proposed classification scheme. Thus, allusions, metaphors, contrasting statements in a sentence render the task of sentiment and opinion detection difficult for the simple classification schemes proposed. At the same time, simple statistics, as taking into account the IG scores between the selected confident features and features present in the sentence, outperform other schemes on the unbalanced corpora.

Based on a relatively simple statistical approach, the proposed classification models were applied in two different contexts (opinion classification and sentiment classification). Our aim was to propose an approach adaptable for various domains and natural languages in the context of sentiment/opinion classification on the sentence-level. We exploited the difference in selection strategies of the Z and KL score based on the feature frequency distributions in the subsets of the corpus. These measures have been used as feature selection methods in text classification before. Nevertheless, we have not encountered the similar adaptation of the KL score to the opinion classification task in related literature. To the best of our knowledge, the proposed combination and use of Z and KL scores present a new model for the tasks of sentiment and opinion classification on a sentence level. Our approach gives similar performance on the balanced datasets to the state-of-the art methods that do not use any manual annotation. Based on polarity scores for terms, the decision taken by our model can be explained more easily than using the SVM model (based on a distance measure computed on a set of selected examples).

As a possible next step for amelioration of the model's performance we can look into the use of additional language-dependent tools. We continue experiments on the further use of the SentiWordNet with the proposed model. Another possible application of this feature selection method could be the use of SVM with the combined Z and KL scores as feature weights. These attempts are investigated in the next chapter.

Chapter 6

Further Analysis and Experiments

6.1 Introduction

In the previous chapter we presented and discussed several classification procedures based on calculation of the modified Z and KL scores. Given many factors, characterizing the evaluated datasets, we found it difficult to propose only one computed scheme that would perform equally well on all setups. Thus, we were interested to investigate different possible extensions and uses of the calculated scores and their modifications.

In this chapter we present further analysis and experiments of the proposed classification model. First, we analyze the reweighting strategy for the features that occur closer to the end of the sentence in Section 6.2. This strategy was evaluated in previous studies on the movie reviews consisting of several sentences [Raychev 2009]. Our aim was to verify if this would be true for the one-sentence movie reviews.

In Section 6.3 we perform a series of experiments with the KL score, Z score and their linear combination as feature selection strategies for the SVM classifier. Based on previous works in the field [Gabrilovich 2004, Joachims 2001], we analyze the distribution of feature scores in the dataset and the expected improvement for the SVM model from the feature selection techniques.

In Section 6.4 we analyze the performance of the classification model that uses sentiment lexicons (SentiWordNet and OpinionFinder) by themselves and a

combination of the sentiment lexicon score and our model's score. Next, in order to verify the performance of the proposed classification scheme for other natural languages, we present the experiments on Japanese and traditional Chinese newspaper corpora from the three NTCIR campaigns in Section 6.5.

6.2 Experiments with Positional Information

Several studies suggest that the position of a word, or consequently a feature, in movie reviews plays a role in the sentiment classification task. It has been noted that the opinion judgment is expressed mostly towards the end of the review at least in the English language [Raychev 2009, Pang 2008]. For example, let's take several reviews from the Subjectivity dataset:

> - There are moments in this account of the life of artist Frida Kahlo that are among cinema's finest this year. Unfortunately, they're sandwiched in between the most impossibly dry account of Kahlo's life imaginable.
> - If you're the kind of parent who enjoys intentionally introducing your kids to films which will cause loads of irreparable damage that years and years of costly analysis could never fix, I have just one word for you.

Pang *et al.* [Pang 2002] propose to weigh words according to the part of the sentence where they occur. They attribute different weights to the words occurring in first, second and third, the last part of the review, incorporating them with the unigram classifier. They did not obtain a great increase in accuracy than using unigrams. Beineke *et al.* [Beineke 2004] analyze movie reviews and conclude that the first and the last sentences usually are the most important for sentiment classification. Other studies perform experiments on documents and blogs, where the detection of the opinionated sentences for document opinion and sentiment classification can benefit from positional information [Pang 2004, Chenlo 2011, Heerschop 2011]. They confirm that the

overall polarity of posts depends on a few sentences taken from the beginning and the end of a review, and on the high-polarity sentences related to the query.

In another study, Raychev *et al.* [Raychev 2009] use the positional information with the naïve Bayes classifier. They incorporate the linear interpolation to the frequency counts of words to obtain the position-dependent fractional counts. It is important to note that the observation of the relevance of positional information tends to be useful for the specific domain, such as movie reviews. At the same time, if we consider reviews of the size of the sentence the assumption for boosting the score of the features towards the end may not hold. Consider the following examples, still from the Subjectivity dataset:

- A photographic marvel of sorts, and it's certainly an invaluable record of that special fishy community.
- Little is done to support the premise other than fling gags at it to see which ones stick.

In order to verify this, we decided to carry out several experiments with the proposed model CP 1 with linear score combination taking into account positional information. The proposed model turns out to be most advantageous for re-weighting the features based on their positional information, since we calculate statistical scores for all computed features in the dataset. These experiments were conducted on the Movie Review and Subjectivity datasets. For the newspaper corpora we obtained a degradation of the performance levels. Since we use a sentence as a classification item, it seems to be more appropriate to use a model that reweighs the scores of the sentence features according to their position, similar to the model described in [Raychev 2009].

We adopted the following procedure. Since we calculate the feature weights adopting bag-of-words model, it is possible to readjust those weights depending on the feature position in the sentence. Adopting similar procedure from [Raychev 2009], let's assume that we have the following features in the sentence $f_1, f_2, .., f_n$, where n is the size of the feature set of the sentence. Thus,

with our model we obtain the following weights for the sentence $w_1, w_2, .., w_n$. The reweighting formula would change each weight in the following manner: $w'_k = a + q \cdot \frac{p}{n-1}$, where $k = 1..n$ and p is the position of occurrence of the feature w_k, $p = 1..n$. Here, we defined the interval $[a; a+q]$ for the weight in $[0, 1]$, where $a = 0$ and $q = 1$ respectively. From our proposed model in Chapter 4, we can see that the multiplication of the scores by a will not change the resulting decision. Therefore, it is sufficient to consider the reweighting algorithm of $0 + q$.

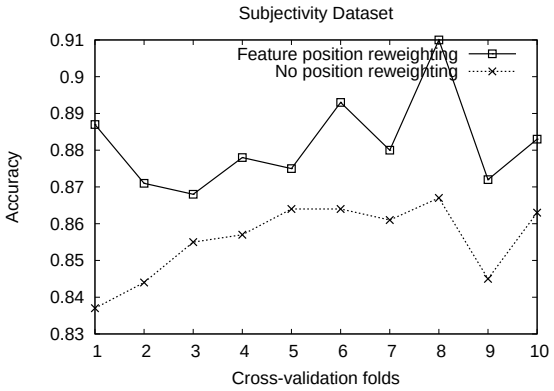

Figure 6.1: Accuracy over 10 folds for Subjectivity dataset.

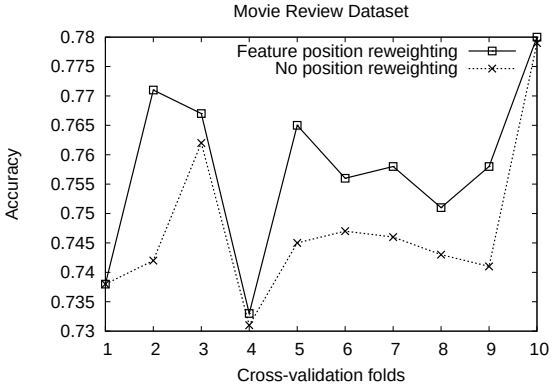

Figure 6.2: Accuracy over 10 folds for Movie Review dataset.

As expected from the study by [Raychev 2009], we obtained the amelioration in accuracy for the movie review datasets. The best amelioration was obtained with the unigram indexing setup. The accuracies over the 10 validation folds are presented in Figure 6.1 and Figure 6.2. There was almost no amelioration with other indexing approaches. In our opinion, the application of heuristics, as taking into account position of features in a sentence, should be taken in consideration with the pre-processing techniques applied. Thus, the schemes that produce more features than the number of tokens in a sentence may place too much weight on the features towards the end of a sentence.

6.3 Experiments with SVM

In our work we do not directly explore the class of SVM methods that is quite often used for sentiment classification tasks [Pang 2008, Whitelaw 2005, Zaidan 2007, Mullen 2004]. Several studies on SVM methods report that they do not perform well faced with the small frequencies of discriminative features used in the model [Joachims 2001]. Recently, the SVM methods have reached their plateau in the state-of-the-art performance and do not show any breakthrough on text classification tasks. The off-the-shelf solutions openly available online without further fine-tuning give average results on the benchmark datasets [Pang 2002, Zubaryeva 2010a]. Taking into account positional information and the use of IR metrics for feature selection, as well as POS tags tends to ameliorate the results [Paltoglou 2010].

Since our classification model is heavily based on feature selection methods, we were interested in evaluating the features selected by our method with the use of the SVM classifier. As pointed out in [Joachims 1998], SVM is able to learn a model independent of the dimension of the space with few irrelevant features present. The experiments on text categorization task show that even the features, that are ranked low according to their IG, are still relevant and contain the information needed for successful classification. Another particularity of the text classification tasks in the context of the SVM method is the sparsity of the input vector, especially when the input instance is a sentence, and not a document.

Gabrilovich *et al.* [Gabrilovich 2004] show that the feature selection procedure could sometimes be beneficial for the SVM classifier, contrary to the argument towards the use of all features needed to achieve the best result in text classification. Thus, they point out that the datasets that have several features with high IG scores, while the rest of the features have low IG scores, benefit from the feature selection procedures when classified with the SVM model. Gabrilovich *et al.* [Gabrilovich 2004] show an improvement in SVM accuracy

from 0.60 to 0.93 with aggressive feature selection on one of the datasets acquired from the Open Directory Project and characterized by several features with a high IG, while others have "markedly lower IG scores".

As pointed out by Joachims [Joachims 2001], the bound on the error rate of the SVM depends on the occurrence frequency of discriminative features, the difference in the vocabulary used in different categories and the redundancy of the training set. In other words, if the two classes use the same vocabulary and come from the same domain, e.g., monthly bestseller reviews, and could be differentiated only by a smaller subset of the words used, we can expect lower levels of the SVM performance. Therefore, the SVM model can benefit from the feature selection procedure when the two classes can be differentiated by a relatively small subset of words [Gabrilovich 2004].

In topical text classification a small set of topic-specific terms may aid the SVM performance. This is hardly the case of opinion and sentiment classification tasks, as the polarity can be expressed by different, not related to any topic, terms and expressions. Moreover, the classification on a sentence-level may be more difficult for the SVM classifier, as smaller number of clues are present in the sentence.

Following the analysis performed by Gabrilovich *et al.* [Gabrilovich 2004], we ranked the features according to their IG score (see Figure 6.3) to analyze the speed of decline of the IG values across the features. With the growth of the IG score, grows the discriminative power of the feature. They show that the Outlier Count (OC) reliably predicts the utility of feature selection for various datasets.

$$OutlierCount(D, F) = |f \in F : IG(f) > \mu_{IG} + 3 \cdot \sigma_{IG}|, \tag{6.1}$$

where μ_{IG} and σ_{IG} are average and standard deviation of information gain of the features f in F, where F is the set of all features in the dataset D. Equation 6.1 returns the number of outliers, OC. Low OC indicates that the dataset can profit from feature selection methods, whereas larger levels show that feature selection

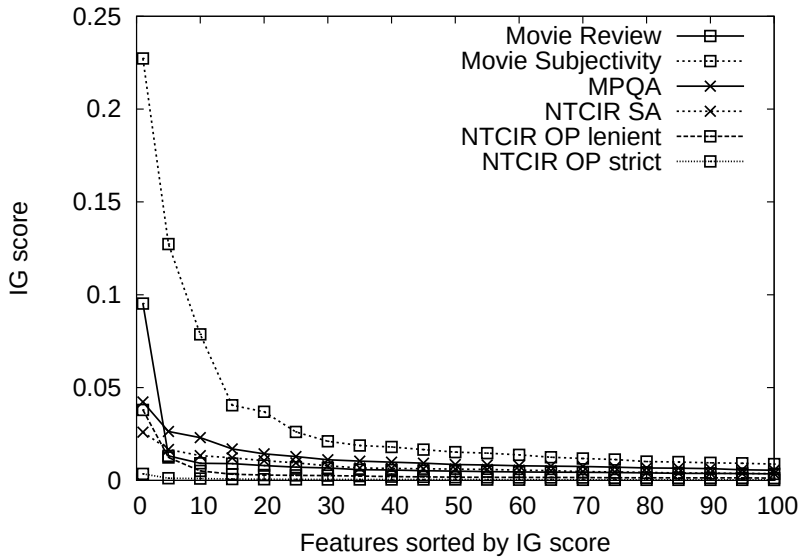

Figure 6.3: Distribution of features by IG score in several datasets.

for SVM shows degradation in accuracy. From the Figure 6.3 we can observe how the values of the feature IG scores decline with the increase of the feature rank for most of the datasets. Movie Review and Subjectivity tadatasets are the only two that contain a small number of top-ranked features with a much higher IG score. According to Gabrilovich *et al.* [Gabrilovich 2004] the feature selection procedure should give higher increase in accuracy for the SVM model on the Movie Review and Subjectivity datasets than on other corpora.

In order to verify this, we first give a description of the experiment setup and then the discussion of the results obtained. Joachims [Joachims 1998] observed that the text classification problems are linearly separable. Thus, a lot of the research dealing with text classification uses linear kernels [Forman 2003]. In our experiments we used SVM^{light} implementation with the linear kernel with the soft-margin constant $cost = 2.0$ [Joachims 1999]. We chose *cost* value based on the experimental results. Generally, the low *cost* value (by default 0.01) indicates a bigger error tolerance during training. With the growth of the

cost value the SVM model assigns larger penalty for margin errors.

As feature scores for the SVM model, we use Z and KL scores, as well as their normalized combination $C \cdot KL$ score, presented in the previous chapter. We also experimented with other types of kernels, namely with the radial basis function kernel. From our experiments, learning of the SVM model with this kernel takes substantially longer time and gives approximately the same level of the performance as the linear kernel.

Feature Selection									
Dataset	$C \cdot KL$ **score**			KL **score**			Z **score**		
% of features	*60%*	*80%*	*100%*	*60%*	*80%*	*100%*	*60%*	*80%*	*100%*
Movie Review†	71.56	73.43	**74.19**	65.93	**65.93**	65.88	**73.11**	71.32	64.24
Subjectivity†	87.8	88.89	**89.68**	84.72	**84.72**	84.69	84.09	**84.25**	82.25
MPQA†	73.41	74.52	**74.84**	68.42	68.42	**68.46**	73.82	**74.71**	67.39
NTCIR SA⋆	67.21	68.38	**68.48**	63.12	**63.12**	62.84	65.37	**65.37**	60.95
NTCIR OP len.⋆	60.18	**60.56**	59.41	50.0	50.0	50.0	**61.39**	57.35	58.09
NTCIR OP str.⋆	50.0	50.0	50.0	50.0	50.0	50.0	**51.95**	50.0	50.0

Table 6.1: Accuracy† and F1-measure⋆ of SVM^{light} with the linear kernel ($\gamma = 2.0$) and different percentage of features.

In Table 6.1, we give the accuracies for the balanced datasets (Movie Review, Subjectivity, and MPQA), whereas we report the macro-averaged F1-measure for the unbalanced datasets. We prune the ranked features by the score, accounting for at least 60% of the feature set. This is due to the fact that further pruning of features leads to drastic degradation in accuracy. Further elimination of features from the training model leads to the situation when some testing sentences are represented with one or two features only.

We can notice the same tendencies in performance of the SVM classifier

and our model in terms of the accuracies achieved on the balanced datasets. Thus, the Subjectivity dataset turns out to be the easiest for classification out of the three. The best accuracies are achieved with the $C \cdot KL$ measure, which does not benefit from the pruning of the feature set. It is interesting that pruning the feature set up to 60% and 80% of top ranked features ameliorated the accuracy of the Z score and the SVM model. This tendency remains the same for the unbalanced datasets. It means that the features with the high Z scores remain good sentiment markers. However, it is detrimental to include all of the features. The possible reason may be that a relatively high Z score may be attributed to the features with small frequencies that occur mostly in one category. The pruning of the feature set ranked by the KL score does not give much amelioration in accuracy.

The performance of the SVM model was quite low for the unbalanced datasets. Almost in all cases the model classified an instance to the category with the biggest size of the training set (factual, or negative). For the dataset with the least number of opinionated sentences compared to factual, NTCIR OP strict, the SVM failed to attribute any instance no matter the percentage of features chosen or metric. The only exception was achieved when pruning up to 60% of top features ranked by the Z score. In this case some of the opinionated instances were identified. As the size of the smallest training category increases, so does the performance of the SVM.

The SVM with the KL score failed to identify opinionated sentences for the NTCIR OP lenient dataset, whereas we achieve higher accuracy with pruned feature set and the Z score. The pruning of features for the $C \cdot KL$ measure, as before, does not prove to be efficient. On the NTCIR SA the $C \cdot KL$ score gives the highest performance of the three metrics. In our opinion, the Z score benefits unbalanced datasets more, whereas with the increase of the size of the smallest training category $C \cdot KL$ and KL scores are able to select relevant features for classification.

The experiments with the use of the SVM with the $C \cdot KL$ score, which is a linear combination of Z and KL scores, give the best performance out of the three metrics used with the SVM model. In our opinion, this is due to the fact that the important sentiment markers that are more descriptive of one category, get to be chosen in the combined metric which gives it an edge when used with the SVM model. The amelioration of the SVM model results are possible with the use of POS tagging or additional language-specific information. It is also important to note that the majority of studies that evaluate the SVM approach on the publicly available datasets, like Movie Review, use the earlier version with the text granularity level of a paragraph. Thus, it makes it difficult to compare the SVM model performance since it is highly influenced by the text size of the training instances [Joachims 2001].

6.4 Use of Sentiment Lexicons

A lot of studies have used annotated word lexicons for sentiment polarity and subjectivity classification [Kim 2009, Fahrni 2008, Devitt 2007, Verma 2008]. In this section we present our experiments with the use of the two popular lexicons: OpinionFinder and SentiWordNet. Other lexicons commonly used are General Inquirer [Stone 1966] and WordAffect [Valitutti 2004]. Due to the smaller set of annotated features in the latter lexicons and a big overlap in the number of entries with the first two, we did not include them in the experiment. The current version of the SentiWordNet includes the sentiment scores for all of the synsets of WordNet, which makes it more than 100,000 terms.

OpinionFinder is a name of a system that performs the sentiment analysis on document level, selecting subjective sentences [Wiebe 2005]. It detects the subjectivity on a sentence level, including agents who are sources of opinion,

direct subjective expressions and speech events, as well as sentiment expressions. With the OpinionFinder distribution comes a sentiment lexicon with the same name. Wiebe *et al.* [Wiebe 2005] created the OpinionFinder lexicon with the use of human annotation and machine learning techniques. The lexicon contains 6,856 entries. Each entry is labeled with a sentiment polarity label: positive, negative, or neutral. For instance, here is an entry from the OpinionFinder lexicon:

```
type=weaksubj len=1 word1=concern pos1=verb stemmed1=y
   priorpolarity=negative
type=strongsubj len=1 word1=stubborn pos1=adj stemmed1=n
   priorpolarity=negative
```

Here, we can see that the word *concern* in the entry is a "stemmed" verb with a weak subjectivity clue and a negative polarity. We are not sure that in all cases the "stem" given by OF is equal to the corresponding lemma.

Another lexicon that is frequently used in sentiment analysis is SentiWordNet [Esuli 2006b]. This is a sentiment lexicon that was built on top of WordNet. WordNet is a thesaurus that contains terms, called *glosses*, with a textual description and relationships between them [Miller 1995]. Some of the documented relationships between glosses include synonymy, hyponymy, entailment and others. A *synset* in SentiWordNet denotes a sense or a specific context which is described by a term or a set of terms. For example, *sorry, sad, pitiful, lamentable* and *distressing* belong to the same synset and are given the same scores.

SentiWordNet assigns three scores for each *synset*, and thus for every term in a synset. The scores attribute positive, negative and objective degrees of a sentiment to a synset. The values of scores range from 0 to 1. Each triple of scores sums to 1. Let's consider one of the entries:

```
a 00005473 0.75 0 direct#10 lacking compromising or mitigating
   elements; exact; "the direct opposite"
```

In the example above we can see that the first letter *a* encodes the POS, in this case adjective. Next, there is a unique ID of the synset. The next two values are positive and negative scores of the term or terms belonging to the synset. The latter are the degree of positivity and negativity of the term. Next, the synset is distinguished by an identifier #. Then, the sense of the term is explained. In this example, we have two scores 0.75 and 0 for the adjective *direct*. The objective score is derived by $1-(pos+neg)$, where *pos* and *neg* are positive and negative scores assigned by SentiWordNet.

The SentiWordNet scores are based on the decisions taken by a pool of classifiers [Esuli 2006b]. For the given example, six out of eight classifiers judged *direct* as positive, none as negative and two as objective, producing the above scores.

6.4.1 Incorporating Lexicon Scores in the Model

In this subsection, we describe how we derive scores for each feature from the SentiWordNet and OpinionFinder lexicons. As we do not use any additional information, such as POS tagging, it is difficult to identify the correct POS and sense of the feature. For instance, the term *necessary* has the following different entries in the SentiWordNet:

```
POS ID pos neg word sense
n 09367203 0 0.125 requisite#1 requirement#2 necessity#2
necessary#1 essential#1 anything indispensable
a 01580050 0.625 0 necessary#1 absolutely essential
a 00343552 0.125 0 necessary#2 unavoidably determined by prior
circumstances
```

From the related literature, one can distinguish two approaches usually adopted to calculate the sentiment scores from the SentiWordNet [Fahrni 2008, Devitt 2007, Verma 2008, Kim 2009]. The first approach is to choose the maximum out of all

of the sentiment scores of each of the synsets to which a term belongs. Thus, the final score $Score(f, c_{pos})$ for a feature f on the example of the positive class would be calculated in the following way:

$$Score\_SWN^{max}(f, c_{pos}) = \underset{f_k \in SWN(f)}{\arg\max} \{SWNscore(f_k, pos)\}, \tag{6.2}$$

where $SWN(f)$ denotes all synsets in SentiWordNet that contain feature f (from 1 to K), and $SWNscore(f_k, pos)$ denotes the positive score for the kth synset of a feature f. Using the above example, we obtain a score of 0.625 as a positive sentiment score.

Another strategy includes calculation of the average of the sentiment scores for a feature as used in [Fahrni 2008, Devitt 2007, Verma 2008]. As before, here is the final score computed with an averaging strategy for the feature f and positive class c_{pos}:

$$Score\_SWN^{ave}(f, c_{pos}) = \frac{\sum_{k=1}^{K} SWNscore(f_k, pos)}{|K|}, \tag{6.3}$$

From the previous example, we get $(0 + 0.625 + 0.125)/3 = 0.25$. Thus, we calculate two versions of the SWN scores that we use in our experiment setup. The opinionated score is computed as $Score\_SWN(f, c_{op}) = 1 - \arg\max\{Score\_SWN(f, c_{pos}), Score\_SWN(f, c_{neg})\}$ for both strategies of selecting the maximum score or averaging the scores over all of the synsets.

For the OpinionFinder score we compute the feature scores, similar to [Kim 2009], in the following way:

$$Score\_OF((f, c_{pos}) = \begin{cases} 1.0, & \text{if } f \text{ is } positive \text{ and } strong \\ 0.5, & \text{if } f \text{ is } positive \text{ and } weak \\ 0.0, & \text{otherwise} \end{cases} \tag{6.4}$$

$$Score\_OF(f, c_{op}) = \max\{Score(f, c_{pos}), Score(f, c_{neg})\}, \tag{6.5}$$

Score	Top Positive Features, Movie Review Dataset
$C \cdot KL$ score	film, as, but, you, like, more, all, make, have, story, character, can, most, good, comedy, perform, too, work, well, love, director, funny, feel, their, look, he, little, life, best, your, year
Z^{Φ} score	engross, portrait, refreshingly, examine, rich, wonderful, mesmerize, heartwarm, gem, cinema, enjoy, spare, vividly, solid, beauty, terrific, spielberg, iranian, refreshing, capture
SWN^{max}	soft, top, good, wonder, splendid, answer, golden, superb, sweet, prefer, fine, solid, attract, nice, urban, pretty, enough, profit, taste, manner, account, reserve
SWN^{ave}	superb, outstanding, good-natured, pleaser, good-time, wondrous, splendid, gorgeous, taste, likable, apt, patience, worthy, eager, thoroughly, esteem, sly, lucky, vibrant, quintessential
$OpFinder$	brave, hilarious, awesome, joke, upbeat, poetic, clearer, splendid, stir, meaning, strikingly, enjoy, humankind, comfort, gratify, enchant, devote, knowledge, sensible

Table 6.2: Top positive features as selected by different scores for Movie Review dataset.

The OpinionFinder does not provide the objective scores for the terms. It is possible to derive scores only for the subjective terms. Thus, in our experiments for each term that is not in the OpinionFinder lexicon, we assigned an objective score of 0.5.

We incorporate the scores for the two classification categories using the linear and logarithmic expert pools described in Chapter 5, Section 2.2. We derive the prior weights based on the performance of the lexicon scores $Score\_SWN^{max}(f,c)$, $Score\_SWN^{ave}(f,c)$ and $Score\_OF(f,c)$ on the tested corpora.

In order to facilitate further analysis and performance of our model combined with the scores derived from the sentiment lexicons, we give an example of the top features for the positive category selected by each of the lexicons, $C \cdot KL$ score and the normalized Z^{Φ} score for the Movie Review dataset. It is clear from the Table 6.2 that the terms chosen by the Z score and the $C \cdot KL$ score are mostly domain-specific. They include nouns and pronouns rather than mostly adjectives as features chosen by lexicons. Here we can also notice the difference in selecting the top features between the Z and $C \cdot KL$ scores. There is no overlap between the two sets of selected features, since the $C \cdot KL$ score attributes higher score value to more frequent features that are not domain-specific (e.g., *as, but, you, like, have*), while the Z^{Φ} score is high for the features with a high frequency in the category, rather than in the whole training corpora. The top features selected based on the lexicon scores for the Movie Review dataset contain mostly adjectives and nouns. They do not take into account the distribution of the functional and topic-specific terms in the domain.

6.4.2 Experimental Results

We conducted the experiments using different scores derived from the sentiment lexicons. As the SentiWordNet lexicon is quite long, for each of the tested datasets we reconstructed the SentiWordNet lexicon containing scores only for the features occurring in the specific dataset. The same process was repeated for the OpinionFinder (OF) lexicon. The results of the experiments for the unigram setup using only the lexicon scores for classification are presented in the Table 6.3.

Although OpinionFinder lexicon is smaller than SentiWordNet, the way we calculate the sentiment scores gives better results than the classification based on the SentiWordNet scores. Overall, the averaging and maximum strategies perform more or less on the same level. It is interesting to note that for the datasets containing movie reviews, the SWN^{ave} gives better results than SWN^{max}. Although, for the newspaper domain the latter strategy of choosing

	SWN^{max}	SWN^{ave}	OF
Movie Review†	55.87%	57.58%	61.78%
Subj. dataset†	50.63%	54.81%	59.76%
MPQA OP†	58.43%	55.16%	61.89%
NTCIR SA⋆	55.94%	54.27%	66.01%
NTCIR OP len⋆	54.12%	52.13%	57.88%
NTCIR OP str⋆	53.34%	51.62%	56.63%

Table 6.3: Accuracy† and F1-measure⋆ of the classification model based on sentiment lexicon scores.

the maximum score proved to be more effective. This may be due to the stylistic characteristics of the texts in different domains.

In order to incorporate the lexicon scores with the model scores we use the linear pool, as described in Chapter 5, Section 2.2. The weights for the lexicon scores and $C \cdot KL$ score we derive based on the error rate of each of the lexicon scores in Table 6.1 and the $C \cdot KL$ in Table 5.9 (CP 1, lin. pool). The results of the proposed classification scheme with $C \cdot KL$ score combined with sentiment lexicon scores are presented in Table 6.4.

Our model is based on the computation of statistical scores for all features that occur in the corpus. Thus, the weights of functional features and content-specific features receive a score for the polarity and opinionatedness. Intuitively, this way we assign a score to all terms without exception, capturing the tendency in the choice of words depending on the sentence category. Therefore, for the specific training corpus we derive weights for terms that could be considered not opinionated (according to SentiWordNet) or are not included in the lexicon at all (OpinionFinder). At the same time the derived weights for the terms that appear in the sentiment lexicons are based on their distribution in each category and could receive different scores than those found in lexicons.

Combination of $C \cdot KL$ score &	SWN^{max}	SWN^{ave}	OF
Movie Review†	72.3%	**72.61%**	72.56%
Subj. dataset†	90.65%	90.39%	**90.88%**
MPQA OP†	73.42%	**73.61%**	72.78%
NTCIR SA⋆	70.35%	69.82%	**70.46%**
NTCIR OP len⋆	62.68%	62.46%	**63.18%**
NTCIR OP str⋆	54.95%	55.01%	**56.35%**

Table 6.4: Accuracy† and F1-measure⋆ of the classification model based on the linear pool combination of the $C \cdot KL$ score and sentiment lexicon scores.

It is possible that the terms that are usually attributed one sentiment polarity may receive a high rank in the opposite category. For example, the words *tv*, *pinocchio*, *seagal* are in the top thirty features with the highest Z^{Φ} score for Movie Review dataset in the negative class, while *spielberg*, *cinema* and *capture* pertain to the top features in the positive class. Some of these occurrences are due to the usual language employed to describe the movie domain reviews. For instance, the word *tv* usually occurs in comparison structures, such as *...would probably have worked better as a one-hour tv documentary*, *typical anime, with cheap animation (like Saturday morning tv in the '60s)*, where the resemblance to anything *tv*-like is negative. At the same time, the word *cinema* occurs mostly in contexts *quality cinema*, *unusual, food-for-thought cinema*, *anyone with a passion for cinema, emerging in world cinema.*

Thus, we can see that normalizing only the scores of features with the sentiment lexicons will not always give amelioration. This is due to the fact that we lower the final score of the domain or topic-specific features and augment the scores of features with high lexicon scores. By changing the feature weights in this way we lower the performance of the model. In our opinion, further research is needed as to the development of an alternative method for incorporating the scores into the model. For example, normaliz-

ing the scores of features that occur closer to the end of the ranked list of scores. This way, we will not change the weights of the top features, that include a lot of topic-specific terms, that are usually scored low in the sentiment lexicons.

6.5 Experiments with Chinese and Japanese Corpora

As the Web contains a growing part of textual information in other languages than English, the comparison of the model's performance on corpora in other natural languages is of high importance. One of the goals of this work is to propose a robust classification model, easily adaptable across different natural languages and domains. The evaluation results of the initially proposed model based on the Z score and logistic regression (see Chapter 4) are given in [Zubaryeva 2008, Zubaryeva 2010b]. Without the use of additional language-specific resources we were not able to achieve the same level of performance as the one obtained for the English language task of the NTCIR 6 and 7 evaluation campaigns. In this section we present the experiment setup, results and analysis of the proposed classification model based on the normalized Z and KL scores on the newspaper corpora from the three NTCIR campaigns in Chinese and Japanese.

6.5.1 Sentiment Analysis in Chinese and Japanese

A lot of previous work in sentiment analysis and opinion mining used sentiment lexicons and other resources for classification in other languages [Banea 2008, Prettenhofer 2010, Kim 2006]. Since the annotated corpora is not easily available, a lot of researches mine the Internet forums or sites in other languages or employ translation services available online to create the corpora [Banea 2010, Kim 2006]. In [Banea 2010] the choice of other

languages to test the classification model is based on the availability of clear word delimitations.

Other approaches regarded the adaptation of English-specific lexicons and corpora to Asian languages. Although it is not a straightforward task, there has been a rise in exploration of different classification models for Chinese and Japanese [Hu 2004, Kobayashi 2004, Takamura 2006, Zagibalov 2008a]. Wan *et al.* [Wan 2008] automatically translate Chinese reviews into English, annotating the English reviews with the use of the sentiment lexicons using a rule-based system. Zagibalov *et al.* [Zagibalov 2008a] use a bootstrapping technique with a small number of selected seed words. They identify parts of text between punctuation marks and attribute a polarity score based on scores of terms occurring in the particular part of the text. The scores are derived by iteratively expanding the set of the seed words.

A lot of studies include experiments on corpora consisting of the Web forum listings reviews that were acquired by the researches themselves, which makes it harder to compare the approaches and their performance. In order to provide a benchmark for comparison, in recent years the NTCIR campaigns have produced an annotated corpora of the newspaper articles in Japanese and Chinese (simplified and traditional) [Seki 2007, Seki 2008, Seki 2010]. In [Zubaryeva 2009] we compared the Z score and logistic regression scheme with the baselines of the naïve Bayes and SVM using the NTCIR 6 and 7 Chinese corpora. We found that the proposed logistic model achieves better F1-measure scores than the baselines.

In order to verify the performance of the proposed classification scheme based on computation of the normalized Z and KL scores for other natural languages, we carried out experiments using the corpora provided by the NTCIR campaigns. We used no language-specific lexicons or heuristics. The adaptation of sentiment lexicon scores for Chinese and Japanese, as well as possible detection of the negation, may improve the classification accuracy of the proposed method.

6.5.2 Experimental Results

In our experiments we used the Chinese and Japanese corpora from the three NTCIR campaigns. The Japanese data consists of the Mainichi newspaper articles from the years 1998 to 2001. The Traditional Chinese data contains articles from 1998 to 2001 from the China Times, Commercial Times, China Times Express, Central Daily News, China Daily News, United Daily News, Economic Daily News, Min Sheng Daily, United Evening News, and Star News [Seki 2007, Seki 2008, Seki 2010]. In order to pre-process the collection and divide the annotated sentences into two categories, we performed the same rules described in Chapter 3, Section 5.1 for opinion and sentiment classification tasks. Thus, in Table 6.5 we obtained the following statistics on the number of sentences per category and evaluation metric.

Standard	**OP**	**NOOP**	**POS**	**NEG**
Japanese				
Strict	5,025	17,095	446	1,119
Lenient	7,094	15,026	773	1,652
Traditional Chinese				
Strict	5,697	17,521	1,270	1,858
Lenient	13,522	9,696	3,504	4,822

Table 6.5: Sentence statistics by category for the Japanese and Chinese NTCIR 6,7,8 corpora.

Similarly to English, we also use strict and lenient evaluation metrics (or standards). The strict version contains sentences that were attributed the same classification category by all of the judges. According to lenient standard, two of the three judges, for the NTCIR-6 and NTCIR-7 evaluation campaigns, must agree. In the Table 6.5 we give the number of sentences per each category

according to both standards after combining all three collections from the NTCIR 6, 7 and 8 campaigns.

We have assumed until now that words can be extracted from a sentence in order to define the needed features used to determine if the underlying information item conveys an opinion or not. Working with Chinese language this assumption no longer holds, and we need to determine indexing units by either applying an automating segmentation approach (based either on a morphological (e.g., CSeg&Tag) or a statistical method [Murata 2003], or considering *n-gram* indexing approach (unigram or bigram, for example). The unigram in this case is one Chinese character. Finally we may also consider a combination of both n-gram and word-based indexing strategies.

For the carried out experiments, we tried bigram and trigram indexing strategies. The experimental results show that bigram indexing outperforms trigram [Zubaryeva 2009]. For the experiments with the Z score and KL score model we adopted the bigram indexing scheme both for the Chinese and Japanese corpora as described in [Zubaryeva 2009]. The results of the performance are shown in Table 6.6.

For the Japanese language, the lenient standard results for the opinion detection setup are higher than those achieved in the strict setup. This can give an indication that our method is able to identify sentences that are considered opinionated by some judges, thus improving the performance of the model in the setup with the lenient standard. The results for the sentiment detection in Japanese do not differ much for both standards. This could be due to a rather small difference in the number of training sentences in each category in both setups.

The classification performance for the Chinese language gave higher results than the experiments in Japanese. One possible explanation could be

Corpora	Precision	Recall	F1
Japanese			
Strict OP	64.82%	66.15%	65.48%
Lenient OP	67.97%	68.62%	68.29%
Strict SA	68.91%	65.46%	67.12%
Lenient SA	67.56%	64.59%	66.04%
Traditional Chinese			
Strict OP	69.79%	72.21%	70.98%
Lenient OP	76.52%	74.63%	75.56%
Strict SA	78.60%	78.29%	78.44%
Lenient SA	71.34%	71.34%	71.34%

Table 6.6: Macro-averaged precision, recall and F1-measure of the $C \cdot KL$-score classification model performance.

the use of the unigram and bigram indexing scheme for Chinese as proposed in [Zubaryeva 2009]. The strict standard sentiment analysis setup gave better performance than the lenient standard. In our opinion, this is due to a smaller set of sentences, that were identified to have the same polarity by all of the judges. For the opinion detection task we perform better in the lenient setup. As with Japanese, the model was able to identify correctly sentences that were considered opinionated or factual only by some judges.

In our experiments we do not use any natural language specific tools or lexicons. The results show that the proposed scheme is able to achieve a similar level of performance in semantically different natural languages. This fact could serve as an argument in favor of using the proposed classification scheme based on computation of modified statistical scores as a baseline for opinion/sentiment classification across different natural languages. In the future it is interesting to investigate the performance of the model with the incorporation of sentiment lexicons in Chinese and Japanese. According to some studies [Zagibalov 2008a],

due to the morphology of the Chinese language, several heuristics (as detection of the special negation character in Chinese) may give further amelioration.

6.6 Summary and Discussion

In this chapter we presented further experiments and analysis of the combined scores and the proposed classification scheme in Chapter 5. First of all, we experimented with feature reweighting based on the position in a sentence. This heuristic coupled with unigram indexing scheme gives more amelioration compared to other indexing schemes. In our opinion, more investigation is needed in order to adapt the reweighting of features based on their position when a feature under consideration could be a part of a word or two words. We showed that the use of these scores as a feature selection technique coupled with SVM does not show improvement over the proposed classification schemes in the previous chapter. The only time the SVM model with $C \cdot KL$ score achieves a slightly higher accuracy is on the MPQA dataset. The low SVM performance with feature selection techniques can have two possible reasons. First, the lack of sufficient data, small number of features in a sentence, and second, the classification task, where content-specific and functional features may also play an important role in classification decision. Nevertheless, we were able to establish baseline results that could be achieved when using Z score and KL score as feature selection metrics with SVM. The previous conclusion about the higher relevancy to the category of the features in the top of the ranked Z score list, is confirmed when we obtain an amelioration in accuracy when pruning the feature set with the Z score. The reduction of the feature set for sentence classification leads to a drastic reduction of the number of features, thus degrading the performance.

As a possible extension of the proposed classification scheme we investigated the use of the sentiment lexicons for the English language. We used SentiWordNet [Esuli 2006b] and OpinionFinder [Wiebe 2005] lexicons due to

the large number of annotated features available compared to other resources. First, we conducted experiments using the lexicon scores alone. The results showed that the calculation of scores using the OpinionFinder lexicon overall outperformed the two models based on the combination of the SentiWordNet synset scores. The classification results based only on the scores derived from the sentiment lexicons are quite low, in some cases somewhat better than random classification.

This is due to a number of factors. Usually, sentiment lexicons attribute high scores to terms commonly associated with some sentiment polarity. On a sentence level, we may face a situation where there are only several, or sometimes no terms that bear strong sentiment according to the lexicon. The lexicon scores may also give a misleading information to the model, especially if irony, humor, comparison or allusion are used. In this case other content-specific and functional terms, present in the sentence, need to be evaluated.

In our opinion, more sophisticated methods for combination of expert scores for opinion and sentiment classification in text need to be developed. One of the possible approaches could be reweighting of the sentiment lexicon scores based on the domain where the training sentences come from. Thus, terms as *capture, director, movie, opening* would receive higher sentiment scores if the domain is movie reviews.

In order to verify the adaptability of the proposed statistical model to other natural languages, we performed the experiments on the newspaper corpora in traditional Chinese and Japanese from the three NTCIR campaigns [Seki 2007, Seki 2008, Seki 2010]. We parsed the sentences into two categories for each of the tasks (sentiment and opinion detection) according to strict and lenient standards. These standards are based on the number of judges that agree that a particular sentence in the collection belongs to a specific classification category. We used the bigram indexing setup for pre-processing the

sentences in Chinese and Japanese. The results achieved on the NTCIR datasets show similar level of performance with the NTCIR SA English corpus, that has approximately the same distribution of sentences in both categories. This performance is achieved without use of any language-specific tools, lexicons, or strategies for detecting negation or polarity change in the sentence. Therefore, we estimate it to be a good baseline for automated statistical opinion and sentiment classification for Chinese and Japanese datasets. The experiments on other languages with different text representation and grammar are needed in order to verify if the shown performance remains the same.

Possible future investigation, in our opinion, may include the classification of longer passages of text. It is important to analyze how the calculated statistical scores for the increased feature set, both in size and frequency, would influence the classification. It is possible that on the longer texts the SVM with the Z score and $C \cdot KL$ score feature selection methods may improve the performance. Another path, worth investigating, may be the experiments on a bigger number of natural languages and classification domains. The use of the sentiment lexicons may prove to be an advantage, though further analysis on combination of scores given by the model and sentiment lexicon are needed.

Chapter 7

Conclusion

In this thesis we consider the task of opinion and sentiment classification on the sentence level. We propose a supervised classification scheme based on computation of the statistical scores for the two classification categories (positive/negative, opinionated/factual). Given the defined constraints of this binary classification on the sentence level, we evaluate the proposed approaches from three different perspectives.

First, as a test corpora we use datasets pertaining to two different topical domains: movie reviews and news articles. Besides the obvious relevance of opinion and sentiment analysis in the latter domains, our choice was also influenced by the number and availability of the annotated corpora. We used six datasets in English, two out of which are related to the movie review domain.

Second, we take into account the characteristics of the dataset that we train on. From the NTCIR evaluation campaigns we obtain training datasets with unequal number of sentences in both categories. In contrast to artificially constructed datasets, they represent more real-to-life distribution of opinionated sentences in the news article domain. Using NTCIR datasets we can explore the behavior of the proposed models on the unbalanced corpora.

Last, but not least, we compare the performance of our model across several languages. We performed experiments on corpora in English, Chinese, and Japanese. Thus, we compare the performance of our approach and its adaptability to other natural languages that require different from English feature representation techniques, and have completely different syntax and

morphology.

In Section 7.1 we propose the analysis and discussion of the work, carried out in the course of this study, and how it meets the objectives presented in Chapter 1. In Section 7.2 we give our take on future directions and paths for prospective amelioration of methods and techniques for opinion and sentiment classification.

7.1 Contributions and Discussion

This thesis investigated the four objectives that we listed in the introduction. In this section, we compare the objectives and the work that was done in the course of the research. We discuss the obtained results and contributions of the presented approaches.

Evaluation of textual pre-processing techniques: In our experiments we investigate unigram, bigram and character *n-gram* indexing techniques. We vary the use of stemming and stop word removal. The results show that the proposed approaches benefit from light stemming and removal of a small list of very frequent stop words. Out of the traditional indexing schemes, we found that unigram gives the best performance.

Experiments, presented in Chapter 3, show that the Z score classification model benefits from no stemming and shows an improved precision on the NTCIR English news articles corpus. On the contrary, the experiments with various information measures show ameliorated performance with stemming and removal of a small number of stop words. It has to be noted that the difference in performance of different pre-processing setups is small.

We propose a new indexation scheme, *Wise Tokenizer*, that takes into account the use of prepositions and words composing frequent expressions. We found that this scheme outperforms the traditional ones on Movie

Review dataset. This is due to the use of colloquial expressions and phrases.

In our opinion, it is necessary to conduct initial experiments on feature representation in order to determine the setup parameters that allow achieving the highest accuracy with the chosen approach.

Evaluation of information measures for feature weighting: After initial promising results based on the Z score classification model, presented in Chapter 4, we investigated the use of several information measures for sentence level classification. Due to their widespread use in text classification domain, we evaluated Odds Ratio, IG, Log Likelihood, χ^2 statistics as well as the Z score. Additionally, we proposed a new adaptation of the Kullback-Leibler divergence in order to calculate the KL score. The comparative analysis on the tested dataset showed that the KL score outperforms other information measures on the balanced datasets, while Z score and Odds Ratio are able to identify relevant features in the unbalanced datasets.

Given the differences in the training size per category, we chose the best performing measures on the two types of corpora. Thus, the KL score is able to identify relevant features when we have equal amount of evidence of feature distribution in each category. The Z score, on the other hand, is able to assign higher scores to features more frequent in one specific category. The drawback is its sensibility to feature frequencies.

New approach based on feature weighting for SA: We propose a new method that includes the modification of Z and KL scores computation, as well as several classification procedures that take into account the drawbacks of the chosen information measures. These drawbacks include the assumption of feature independence, sensitivity to feature frequencies and available training size per category (balanced, unbalanced datasets). Thus, we use the normalization of Z and KL scores that takes into account the relative distribution of features per category. As a score combination

procedure we experiment with the log and linear pools. Albeit their low results compared to other scores computed for the model, we obtained the best performance using linear pool combination of Z and KL scores as feature weights for the SVM model.

Since both measures evaluate features independently, we miss out on the important information of frequent co-occurrences of features in a sentence. In order to measure the degree of "dependence" between the two features, we use the IG score. Obviously, this computation turns out to be costly if we take into account all features. To avoid this, we identify a smaller set of *confident features* that obtain high normalized Z scores. The more unequally a feature is distributed across the two categories, the higher normalized Z score it obtains, no matter its frequency. Thus, we penalize frequent terms, mostly function words, that otherwise were in the top of the ranked list by the Z score.

The presented approach, essentially based on feature score computation, can be extended with the use of other evidence, such as scores derived from the sentiment lexicons. The experiment results with the two lexicons, SentiWordNet and OpinionFinder, show that the proposed use of the scores with the classification model usually gives lower performance. Thus, we identified a need of more sophisticated techniques of feature score combination and lexicon score computation.

Generally, the proposed classification procedures achieve an average performance on the corpora examined. This has two implications. First, it shows the limit obtained on the performance of models, based on simple computation techniques, such as the use of the Z and KL scores. Second, it proves that such techniques can work as simple and easily interpreted baselines for short text classification tasks.

Evaluation of the proposed model on Chinese and Japanese corpora:
Using the annotated NTCIR corpora of news articles in traditional Chinese and Japanese, we verify the adaptability of the proposed classification approach. On the pre-processing stage we use a bigram indexing for both languages. Otherwise, no natural language-specific information or heuristics are used. The experiment results show similar level of performance to the one obtained on the NTCIR English datasets. Thus, the proposed approach can be applied to other natural languages. Since we do not use any stop word lists or negation checks, there is a high chance that the obtained results can be further improved.

7.2 Future work

The analysis and methods presented in this thesis give further understanding of the use of the information measures, namely Z and KL scores, for opinion and sentiment classification. The understanding of behavior of the proposed approaches may assist researchers in making decisions about feature weighting and classifier construction. The proposed methods provide an adaptable and extensible framework for sentiment and opinion classification in case of unbalanced datasets in various domains and natural languages.

At the same time, we can identify a number of issues that need further investigation in order to advance the understanding of how to build an effective classifier for a particular domain and task at hand. First of all, we can identify a need for different feature representation techniques that can capture more domain-specific information from text. One of the solutions can be a creation of hybrid tokenization schemes, that, besides including standard tokenization approaches, such as unigrams, can mine phraseological patterns or frequent subset of features from sentences. This may include POS patterns, or word patterns occurring in the sentence themselves. Such features may turn out to be quite domain-dependent. In our opinion, taking into account the sentiment

carried by domain-specific features is a necessary step for optimization of the algorithm's performance. Moreover, the mined feature patterns can better capture dependency information between the features at the step of text pre-processing.

Another research direction is the investigation of classification problems that involve more than two classification categories. For example, we may consider the task of sentiment polarity classification with a neutral class. In this case, we suspect that the model performance will drop due to the ambiguity of the "neutral" class definition. Although, we need to carry out further experiments, we expect that the model performance will be highly dependent on the nature of the classification task (topical, opinion, spam classification) and the size of the classification categories.

We can also consider opinion and sentiment detection on longer passages of text, e.g., paragraphs, documents. In our opinion, in the presence of a higher number of features the proposed models would encounter much more noise which can hinder its performance. In this case we would need a procedure that splits long text into sentences, or other short units, that could be classified separately by our method. Next, we would need another procedure that assigns the final classification category to the document.

In our model, we use the combination of the derived scores with the scores assigned by sentiment lexicons. We observed that while sentiment scores learned from the model tend to be high for domain-specific terms, lexicons assign higher scores to terms that have a specific sentiment connotation across different domains. When we combine the scores learned by the model with the lexicon scores, the terms scored high by the model obtain lower final scores due to the fact that they do not convey any sentiment according to the lexicons. At the same time, terms scored high by the lexicon may have lower scores in the model. In order to avoid the situation where we change the sentiment scores to our disadvantage, we can identify a set of features, for example according to

their POS tags or frequency in the corpus, whose scores we combine with the sentiment lexicon scores. It may turn out that the classification model benefits from the procedure, where the scores of different types of features are combined with sentiment lexicon scores using different weights. This would require further thorough investigation.

Appendix A

Experiments with Different Pre-Processing Setups

Approach	**Movie Review†**	**Subject.†**	**MPQA†**	**NTCIR SA⋆**	**NTCIR OP len⋆**	**NTCIR OP str⋆**
SCP 1	**74.78%**	**85.56%**	62.61%	61.7%	56.14%	51.39%
$SumIG^{Conf}$	66.35%	83.07%	**69.52%**	67.09%	61.76%	**54.26%**
$Support^{Conf}$	58.73%	79.32%	66.94%	61.71%	56.14%	51.17%
$SumZ \cdot IG$	72.61%	85.35%	68.71%	**68.21%**	62.88%	52.16%
$SumZ^{\Phi}$	71.39%	81.62%	66.63%	66.84%	**62.98%**	52.25%

Table A.1: Accuracy† and F1-measure⋆ of the proposed classification models using unigram scheme with 10-fold cross-validation over the six corpora.

Approach	**Movie Review†**	**Subject.†**	**MPQA†**	**NTCIR SA⋆**	**NTCIR OP len⋆**	**NTCIR OP str⋆**
SCP 1	**66.12%**	**87.96%**	**72.16%**	**67.29%**	65.84%	55.62%
$SumIG^{Conf}$	51.84%	79.65%	64.71%	62.23%	62.48%	**55.89%**
$Support^{Conf}$	51.13%	62.78%	61.72%	58.49%	60.53%	51.45%
$SumZ \cdot IG$	58.76%	87.23%	72.79%	69.81%	**66.21%**	55.04%
$SumZ^{\Phi}$	52.96%	77.31%	67.48%	62.07%	60.45%	52.75%

Table A.2: Accuracy† and F1-measure⋆ of the proposed classification models using bigram scheme with 10-fold cross-validation over the six corpora.

Approach	**Movie Review†**	**Subject.†**	**MPQA†**	**NTCIR SA⋆**	**NTCIR OP len⋆**	**NTCIR OP str⋆**
SCP 1	**72.81%**	87.19%	60.93%	64.42%	56.57%	51.21%
$SumIG^{Conf}$	66.92%	83.65%	65.04%	62.18%	**58.34%**	**54.23%**
$Support^{Conf}$	62.87%	82.08%	55.37%	63.14%	51.02%	50.47%
$SumZ \cdot IG$	71.66%	87.24%	**68.51%**	**67.02%**	57.63%	51.12%
$SumZ^{\Phi}$	68.39%	**88.26%**	65.12%	56.13%	55.54%	50.88%

Table A.3: Accuracy† and F1-measure⋆ of the proposed classification models using character *n-gram* ($n = 4$) scheme with 10-fold cross-validation over the six corpora.

Bibliography

[Andreevskaia 2008] A. Andreevskaia and S. Bergler. When specialists and generalists work together: overcoming domain dependence in sentiment tagging. In Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics, pages 290–298, 2008. 21

[Argamon 2010] S. Argamon, S. Dubnov and K. Burns. The structure of style algorithmic approaches to understanding manner and meaning. Springer-Verlag, Berlin, Heidelberg, 2010. 55

[Baayen 2001] H. Baayen. Word frequency distributions. Kluwer, Dordrecht, 2001. 73

[Banea 2008] C. Banea, R. Mihalcea, J. Wiebe and S. Hassan. Multilingual subjectivity analysis using machine translation. In Proceedings of the Conference on Empirical Methods in Natural Language Processing, pages 127–135. ACL, 2008. 148

[Banea 2010] C. Banea, R. Mihalcea and J. Wiebe. Multilingual subjectivity: are more languages better? In Proceedings of the 23rd International Conference on Computational Linguistics, pages 28–36. ACL, 2010. 148

[Baroni 2004] M. Baroni and S. Vegnaduzzo. Identifying subjective adjectives through web-based mutual information. In Proceedings of Konferenz zur Verarbeitung natürlicher Sprache (KONVENS), pages 17–24, 2004. 19

[Beineke 2004] P. Beineke, T. Hastie, C. Manning and S. Vaithyanathan. Exploring sentiment summarization. In Proceedings of the AAAI Spring Symposium on Exploring Attitude and Affect in Text: Theories and Applications, 2004. 132

[Bekkerman 2006] R. Bekkerman, K. Eguchi and J. Allan. Unsupervised non-topical classification of documents. Rapport technique IR-472, Center of Intelligent Information Retrieval, UMass Amherst, 2006. 19

[Bloom 2007] K. Bloom, S. Stein and S. Argamon. Appraisal extraction for news opinion analysis. In Proceedings of the 6th NTCIR Workshop Meeting on Evaluation of Information Access Technologies: Information Retrieval, Question Answering and Cross-Lingual Information Access, 2007. 20

[Boiy 2008] E. Boiy and M. Moens. A machine learning approach to sentiment analysis in multilingual Web texts. Information Retrieval, vol. 12, pages 526–558, 2008. 26

[Boughanem 2008] M. Boughanem and J. Savoy. Recherche d'information etats des lieux et perspectives. Hermes Science Publications, 2008. 74

[Brew 2010] A. Brew, D. Greene and P. Cunningham. Is it over yet? Learning to recognize good news in financial media. Rapport technique, UCD School of Computer Science and Informatics, 2010. 24

[Calvé 2000] A. Le Calvé and J. Savoy. Database merging strategy based on logistic regression. In Proceedings of Information Processing and Management, pages 341–359, 2000. 65

[Chenlo 2011] J. Chenlo and D. Losada. 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, 2011. 132

[Church 1991] K. Church, W. Gale, P. Hanks and D. Hindle. Lexical acquisition: exploiting online resources to build a lexicon, chapitre Using statistics in lexical analysis. Lawrence Erlbaum, 1991. 28

[Clemen 1999] R. Clemen and R. Winkler. Combining probability distributions from experts in risk analysis. Risk Analysis, vol. 19, no. 2, pages 187–203, 1999. 102

[Col 2003] Collins English dictionary and thesaurus. Harper Collins Publishers, 2003. 5

[Dang 2010] Y. Dang, Y. Zhang and H. Chen. A lexicon-enhanced method for sentiment classification: an experiment on online product reviews. IEEE Intelligent Systems, vol. 25, pages 46–53, July 2010. 21

[Das 2001] S. Das and M. Chen. Yahoo! for Amazon: opinion extraction from small talk on the web. In Proceedings of the 8th Asia Pacific Finance Association Annual Conference, 2001. 6

[Dave 2003] K. Dave, S. Lawrence and D. M. Pennock. Mining the peanut gallery: opinion extraction and semantic classification of product reviews. In Proceedings of the 12th international conference on World Wide Web, pages 519–528, 2003. 6

[Devitt 2007] A. Devitt and K. Ahmad. Sentiment polarity identification in financial news: a cohesion-based approach. In Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics, 2007. 140, 142, 143

[Dunning 1993] T. Dunning. Accurate methods for the statistics of surprise and coincidence. Computational Linguistics, vol. 19, no. 1, pages 61–74, 1993. 9, 85

[Durbin 2003] S. D. Durbin, J. N. Richter and D. Warner. A system for affective rating of texts. In 3rd Workshop on Operational Text Classifcation at the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2003. 5

[Eguchi 2006] K. Eguchi and V. Lavrenko. Sentiment retrieval using generative models. In Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing, pages 345–354. ACL, 2006. 29

[Esuli 2006a] A. Esuli and F. Sebastiani. Determining term subjectivity and term orientation for opinion mining. In Proceedings of the 11th Meeting of the European Chapter of the Association for Computational Linguistics (EACL), 2006. 21

[Esuli 2006b] A. Esuli and F. Sebastiani. SentiWordNet: a publicly available lexical resource for opinion mining. In Proceedings of the 5th Conference on Language Resources and Evaluation, 2006. 21, 32, 66, 67, 68, 69, 114, 141, 142, 153

[Fahrni 2008] A. Fahrni and M. Klenner. Old wine or warm beer: target-specific sentiment analysis of adjectives. In Proceedings of Symposium on Affective Language in Human and Machine, AISB Convention, pages 60–63, 2008. 140, 142, 143

[Fellbaum 1998] C. Fellbaum. Wordnet: an electronic lexical database. MIT Press, Cambridge, MA, 1998. 71

[Forman 2003] G. Forman. An extensive empirical study of feature selection metrics for text classification. The Journal of Machine Learning Research, Special Issue on Variable and Feature Selection., vol. 3, pages 1289–1305, 2003. 59, 80, 84, 89, 90, 91, 92, 137

[Gabrilovich 2004] E. Gabrilovich and S. Markovitch. Text categorization with many redundant features: using aggressive feature selection to make SVMs competitive with C4.5. In Proceedings of the 21st International Conference on Machine Learning, pages 321–328, 2004. 131, 135, 136, 137

[Hatzivassiloglou 1997] V. Hatzivassiloglou and K. McKeown. Predicting the semantic orientation of adjectives. In Proceedings of the Joint ACL/EACL Conference, pages 174–181, 1997. 8, 16, 17, 18

[Hatzivassiloglou 2000] V. Hatzivassiloglou and J. Wiebe. Effects of adjective orientation and gradability on sentence subjectivity. In Proceedings of the 18th International Conference on Computational Linguistics, pages 299–305, 2000. 18

[Hearst 1992] M. Hearst. Direction-based text interpretation as an information access refinement. In Text-based intelligent systems. L. Erlbaum Associates Inc., 1992. 16

[Heerschop 2011] B. Heerschop, F. Goossen, A. Hogenboom, F. Frasincar, U. Kaymak and F. de Jong. Polarity analysis of texts using discourse structure. In Proceedings of the 20th ACM International Conference on Information and Knowledge Management, pages 1061–1070, 2011. 132

[Hu 2004] M. Hu and B. Liu. Mining and summarizing customer reviews. In Proceedings of the 10th ACM International Conference on Knowledge Discovery and Data Mining, pages 168–177, 2004. 149

[Hu 2007] Y. Hu, R. Lu, X. Li, Y. Chen and J. Duan. A language modeling approach to sentiment analysis. In Proceedings of the International Conference on Computational Science, pages 1186–1193, 2007. 29

[Hurst 2004] M. Hurst and K. Nigam. Retrieving topical sentiments from online document collections. In Document Recognition and Retrieval XI, pages 27–34, 2004. 24

[Jin 2007] X. Jin, Y. Li, T. Mah and J. Tong. Sensitive webpage classification for content advertising. In Proceedings of the 1st International Workshop on Data Mining and Audience Intelligence for Advertising, 2007. 4

[Joachims 1998] T. Joachims. Text categorization with Support Vector Machines: learning with many relevant features. In Proceedings of the European Conference on Machine Learning, pages 137–142. Springer, 1998. 37, 135, 137

[Joachims 1999] T. Joachims. Making large-scale (SVM) learning practical. In Advances in Kernel Methods - Support Vector Learning, chapitre 11, pages 169–184. MIT Press, Cambridge, MA, 1999. 137

[Joachims 2001] T. Joachims. A statistical learning model of text classification with Support Vector Machines. In Proceedings of the ACM SIGIR Conference on Research and Development in Information Retrieval, pages 128–136, 2001. 74, 131, 135, 136, 140

[Joachims 2002] T. Joachims. Learning to classify text using support vector machines – methods, theory, and algorithms. Kluwer/Springer, 2002. 78

[Justeson 1995] J. Justeson and S. Katz. Principled disambiguation: discriminating adjective senses with modified nouns. Computational Linguistics, vol. 21, no. 1, pages 1–27, 1995. 18

[Kamps 2004] J. Kamps, R. J. Mokken, M. Marx and M. de Rijke. Using WordNet to measure semantic orientation of adjectives. In Proceedings of the 4th International Conference on Language Resources and Evaluation, volume IV, pages 1115–1118. European Language Resources Association, 2004. 21

[Kennedy 2006] A. Kennedy and D. Inkpen. Sentiment classification of movie reviews using contextual valence shifters. Computational Intelligence, vol. 22, no. 2, pages 110–125, 2006. 22

[Kessler 1997] B. Kessler, G. Nunberg and H. Schütze. Automatic detection of text genre. In Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics, pages 32–38, 1997. 16

[Kim 2006] S. M. Kim and E. H. Hovy. Identifying and analyzing judgment opinions. In Proceedings of the Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics, 2006. 148

[Kim 2009] J. Kim, J. J. Li and J. H. Lee. Discovering the discriminative views: measuring term weights for sentiment analysis. In Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP, pages 253–261, 2009. 140, 142, 143

[Kleinberg 2006] J. Kleinberg and E. Tardos. Algorithm design. Addison Wesley, 2006. 9, 24

[Kobayashi 2004] N. Kobayashi, K. Inui, Y. Matsumoto, K. Tateishi and T. Fukushima. Collecting evaluative expressions for opinion extraction. In Proceedings of the 1st International Joint Conference on Natural Language Processing, pages 596–605, 2004. 149

[Labbé 1994] C. Labbé and D. Labbé. Que mesure la spécificité du vocabulaire? Repris dans: Lexicometrica, 3, 2001., 1994. Grenoble, CERAT. 98, 99

[Lafon 1980] P. Lafon. Sur la variabilité de la fréquence des formes dans un corpus. Mots, vol. 1, pages 127–165, 1980. 98

[Lafon 1984] P. Lafon. Dépouillements et statistiques en lexicométrie. Slatkine-Champion, pages 97–110, 1984. 98

[Lavrenko 2001] V. Lavrenko and W. B. Croft. Relevance-based language models. In Proceedings of the ACM SIGIR Conference on Research and Development in Information Retrieval, pages 120–127, 2001. 29

[Liu 2004] B. Liu, X. Li, W. S. Lee and P. S. Yu. Text classification by labeling words. In Proceedings of the 19th National Conference on Artificial Intelligence, pages 425–430, 2004. 25

[Macdonald 2007] C. Macdonald, I. Ounis and I. Soboroff. Overview of the TREC 2007 Blog Track. In Proceedings of the Text REtrieval Conference, 2007. 17

[Manning 2002] C. D. Manning and H. Schütze. Foundations of statistical natural language processing. MIT Press, 2002. 73

[Martineau 2009] J. Martineau and T. Finin. Delta TFIDF: an improved feature space for sentiment analysis. In Proceedings of the International AAAI Conference on Weblogs and Social Media, 2009. 28

[Matsumoto 2005] S. Matsumoto, H. Takamura and M. Okumura. Sentiment classification using word sub-sequences and dependency sub-trees. In Pro-

ceedings of the 9th Pacific-Asia Conference on Knowledge Discovery and Data Mining, pages 301–311, 2005. 24

[Melville 2009] P. Melville, W. Gryc and R. D. Lawrence. Sentiment analysis of blogs by combining lexical knowledge with text classification. In Proceedings of the 15th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, pages 1275–1284, 2009. 25, 103

[Miller 1995] G. A. Miller. WordNet: a lexical database for English. Communications of the Association for Computing Machinery, vol. 38, pages 39–41, November 1995. 17, 21, 141

[Mitchell 1997] T. M. Mitchell. Machine learning. McGraw Hill, New York, 1997. 73

[Mladenic 1999] D. Mladenic and M. Grobelnik. Feature selection for unbalanced class distribution and Naive Bayes. In Proceedings of the 16th International Conference on Machine Learning, pages 258–267, 1999. 80, 84

[Mullen 2004] T. Mullen and N. Collier. Sentiment analysis using Support Vector Machines with diverse information sources. In Proceedings of the Conference on Empirical Methods in Natural Language Processing, pages 412–418, 2004. 24, 135

[Muller 1992] C. Muller. Principes et methodes de statistique lexicale. Champion, Paris, 1992. 61

[Murata 2003] M. Murata, Q. Ma and H. Isahara. Applying multiple characteristics and techniques to obtain high levels of performance in information retrieval. In Proceedings of the 3rd NTCIR Workshop Meeting, 2003. 151

[Na 2001] J. C. Na, H. Sui, C. Khoo, S. Chan and Y. Zhou. Effectiveness of simple linguistic processing in automatic sentiment classification of product reviews. In Proceedings of the Conference of the International Society for Knowledge Organization, pages 49–54, 2001. 125

[Ortony 1990] A. Ortony, G. L. Clore and A. Collins. The cognitive structure of emotions. Cambridge University Press, 1990. 16

[Osgood 1971] C. Osgood. Where do sentences come from? In Semantics: an interdisciplinary reader in philosophy, linguistics and psychology, pages 497–529. Cambridge University Press, 1971. 16

[Paltoglou 2010] G. Paltoglou and M. Thelwall. A study of information retrieval weighting schemes for sentiment analysis. In Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, pages 1386–1395. ACL, 2010. 28, 135

[Pang 2002] B. Pang, L. Lee and S. Vaithyanathan. Thumbs up?: sentiment classifcation using machine learning techniques. In Proceedings of the Conference on Empirical Methods in Natural Language Processing, pages 79–86, 2002. 23, 25, 132, 135

[Pang 2004] B. Pang and L. Lee. A sentimental education: sentiment analysis using subjectivity summarization based on Minimum Cuts. In Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics, 2004. 9, 24, 25, 39, 49, 132

[Pang 2005] B. Pang and L. Lee. 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, pages 115–124, 2005. 39, 47

[Pang 2008] B. Pang and L. Lee. Opinion mining and sentiment analysis. Foundations and Trends in Information Retrieval, vol. 2(1-2), pages 1–135, 2008. 1, 3, 6, 17, 19, 39, 56, 84, 94, 132, 135

[Porter 1997] M. F. Porter. Readings in information retrieval. Morgan Kaufmann Publishers Inc., 1997. 56, 86

[Prettenhofer 2010] P. Prettenhofer and B. Stein. Cross-language text classification using structural correspondence learning. In Proceedings of the 48th

Annual Meeting of the Association for Computational Linguistics, pages 1118–1127. ACL, 2010. 148

[Qiu 2009] L. Qiu, W. Zhang, C. Hu and K. Zhao. SELC: a self-supervised model for sentiment classification. In Proceedings of the 18th ACM International Conference on Information and Knowledge Management, pages 929–936, 2009. 25

[Quirk 1985] R. Quirk, S. Greenbaum, G. Leech and J. Svartvik. A comprehensive grammar of the english language. Longman, 1985. 16

[Rahayu 2010] D. Rahayu, S. Krishnaswamy, O. Alahakoon and C. Labbé. RnR: extracting rationale from online reviews and ratings. In Workshop Proceedings of the International Conference on Data Mining, pages 358–368, 2010. 22

[Raychev 2009] V. Raychev and P. Nakov. Language-independent sentiment analysis using subjectivity and positional information. In Proceedings of the International Conference in Recent Advances in Natural Language Processing, pages 360–364. ACL, 2009. 25, 131, 132, 133, 134

[Sack 1994] W. Sack. On the computation of point of view. In Proceedings of the 12th National Conference on Artificial Intelligence, 1994. 16

[Salton 1975] G. Salton, A. Wong and C. S. Yang. A vector space model for automatic indexing. Communications of the Association for Computing Machinery, vol. 18, pages 613–620, November 1975. 27

[Sarvabhotla 2011] K. Sarvabhotla, P. Pingali and V. Varma. Sentiment classification: a lexical similarity based approach for extracting subjectivity in documents. Information Retrieval, vol. 14, no. 3, pages 337–353, 2011. 27

[Savoy 2010] J. Savoy and O. Zubaryeva. Classification automatique d'opinions dans la blogosphere. In Statistical Analysis of Textual Data (JADT), pages 653–664, 2010. 5

[Savoy 2011] J. Savoy, L. Dolamic and O. Zubaryeva. Searching, translating and classifying information in cyberspace. In Proceedings of the 5th International MCETECH Conference on eTechnologies, 2011. 3

[Schneider 2004] K. M. Schneider. A new feature selection score for multinomial naive Bayes text classification based on KL-divergence. In Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics. ACL, 2004. 81, 82

[Seki 2004] Y. Seki, K. Eguchi and N. Kando. Analysis of multi-document viewpoint summarization using multi-dimensional genres. In Proceedings of the AAAI Spring Symposium on Exploring Attitude and Affect in Text: Theories and Applications, 2004. 4

[Seki 2007] Y. Seki, D. K. Evans, L. W. Ku, H. H. Chen, N. Kando and C. Y. Lin. Overview of opinion analysis pilot task at NTCIR-6. In Proceedings of the 6th NTCIR MOAT Workshop Meeting, pages 185–203, Tokyo, Japan, 2007. 5, 10, 17, 33, 39, 40, 42, 149, 150, 154

[Seki 2008] Y. Seki, D. K. Evans, L. W. Ku, L. Sun, H. H. Chen and N. Kando. Overview of multilingual opinion analysis task at NTCIR-7. In Proceedings of the 7th NTCIR MOAT Workshop Meeting, pages 185–203, Tokyo, Japan, 2008. 5, 7, 10, 17, 33, 39, 40, 93, 149, 150, 154

[Seki 2010] Y. Seki, L. W. Ku, L. Sun, H. H. Chen and N. Kando. Overview of multilingual opinion analysis task at NTCIR-8: a step toward cross lingual opinion analysis. In Proceedings of the 8th NTCIR MOAT Workshop Meeting, pages 209–220, 2010. 5, 10, 17, 33, 39, 40, 65, 149, 150, 154

[Sharman 1990] R. Sharman, F. Jelinek and R. Mercer. Generating a grammar for statistical training. In Proceedings of the 3rd DARPA Speech and Natural Language Workshop, pages 267–274, 1990. 16

[Somasundaran 2007] S. Somasundaran, T. Wilson, J. Wiebe and V. Stoyanov. QA with attitude: exploiting opinion type analysis for improving question

answering in on-line discussions and the news. In Proceedings of the International Conference on Weblogs and Social Media, 2007. 4

[Stone 1966] P. J. Stone. The general inquirer: a computer approach to content analysis. The MIT Press, 1966. 17, 22, 140

[Stoyanov 2005] V. Stoyanov, C. Cardie and J. Wiebe. Multi-perspective question answering using the OpQA corpus. In Proceedings of the Joint Human Language Technology Conference and the Conference on Empirical Methods in Natural Language Processing (HLT/EMNLP), pages 923–930, 2005. 4

[Su 2008] F. Su and K. Markert. From words to senses: a case study of subjectivity recognition. In Proceedings of the 22nd International Conference on Computational Linguistics, pages 825–832, 2008. 20

[Takamura 2006] H. Takamura, T. Inui and M. Okumura. Latent variable models for semantic orientations of phrases. In Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics, 2006. 149

[Tan 2007] S. Tan, G. Wu, H. Tang and X. Cheng. A novel scheme for domain-transfer problem in the context of sentiment analysis. In Proceedings of the 16th ACM Conference on Information and Knowledge Management, pages 979–982, 2007. 26

[Terveen 1997] L. Terveen, W. Hill, B. Amento, D. McDonald and J. Creter. PHOAKS: A system for sharing recommendations. Communications of the Association for Computing Machinery, vol. 40(3), pages 59–62, 1997. 4

[Toutanova 2000] K. Toutanova and C. D. Manning. Enriching the knowledge sources used in a Maximum Entropy part-of-speech tagger. In Proceedings of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora, pages 63–70, 2000. 18

[Turney 2002] P. D. Turney. Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews. In Proceedings of the 40th Annual Meeting on Association for Computational Linguistics, pages 417–424. ACL, 2002. 9, 17, 18, 30

[Turney 2003] P. D. Turney and M. L. Littman. Measuring praise and criticism: inference of semantic orientation from association. ACM Transactions on Information Systems, vol. 21, no. 4, pages 315–346, 2003. 19

[Valitutti 2004] R. Valitutti. WordNet-Affect: an affective extension of WordNet. In Proceedings of the 4th International Conference on Language Resources and Evaluation, pages 1083–1086, 2004. 140

[Verma 2008] S. Verma and P. Bhattacharyya. Incorporating semantic knowledge for sentiment analysis. In Proceedings of the 6th International Conference on Natural Language Processing. Macmillan Publishers, India, 2008. 140, 142, 143

[Wan 2008] X. Wan. Using bilingual knowledge and ensemble techniques for unsupervised Chinese sentiment analysis. In Proceedings of the Conference on Empirical Methods on Natural Language Processing, pages 553–561, 2008. 149

[Watson 1985] D. Watson and A. Tellegen. Toward a consensual structure of mood. Psychological Bulletin, vol. 98, pages 219–235, 1985. 16

[Whitehead 2009] M. Whitehead and L. Yaeger. Building a general purpose cross-domain sentiment mining model. In Proceedings of the 2009 WRI World Congress on Computer Science and Information Engineering, volume 4, pages 472–476. IEEE Computer Society, 2009. 26

[Whitelaw 2005] C. Whitelaw, N. Garg and S. Argamon. Using appraisal groups for sentiment analysis. In Proceedings of the 14th ACM International Conference on Information and Knowledge Management, pages 625–631. ACM, 2005. 135

[Wiebe 1990] J. Wiebe. Identifying subjective characters in narrative. In Proceedings of the International Conference on Computational Linguistics, 1990. 16

[Wiebe 1994] J. Wiebe. Tracking point of view in narrative. Computational Linguistics, vol. 20(2), pages 233–287, 1994. 6

[Wiebe 1995] J. Wiebe and R. Bruce. Probabilistic classifiers for tracking point of view. In Proceedings of the AAAI Spring Symposium on Empirical Methods in Discourse Interpretation and Generation, 1995. 16

[Wiebe 2000] J. Wiebe. Learning subjective adjectives from corpora. In Proceedings of the 17th National Conference on Artificial Intelligence, pages 735–740, 2000. 19

[Wiebe 2005] J. Wiebe and E. Riloff. Creating subjective and objective sentence classifiers from unannotated texts. In Proceedings of the 6th International Conference on Intelligent Text Processing and Computational Linguistics, pages 486–497, 2005. 19, 140, 141, 153

[Wilson 2005a] T. Wilson, P. Hoffmann, S. Somasundaran, J. Kessler, J. Wiebe, Y. Choi, C. Cardie, E. Riloff and S. Patwardhan. OpinionFinder: a system for subjectivity analysis. In Proceedings of the HLT/EMNLP Interactive Demonstrations, pages 34–35, 2005. 26, 32, 34

[Wilson 2005b] T. Wilson, J. Wiebe and P. Hoffmann. 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 354–362, 2005. 39, 46, 47, 125

[Witten 2005] I. H. Witten and E. Frank. Data mining: practical machine learning tools and techniques. Morgan Kaufmann, San Francisco, 2 édition, 2005. 23

[Wu 2009] X. Wu and V. Kumar. The top ten algorithms in data mining. Chapman & Hall/CRC, 1st édition, 2009. 36

[Yang 1997] Y. Yang and J. O. Pedersen. A comparative study on feature selection in text categorization. In Proceedings of the 14th International Conference on Machine Learning, pages 412–420. Morgan Kaufmann Publishers Inc., 1997. 80, 84

[Yu 2008] B. Yu, S. Kaufmann and D. Diermeier. Exploring the characteristics of opinion expressions for political opinion classification. In Proceedings of the 2008 International Conference on Digital Government Research, pages 82–91, 2008. 4

[Zagibalov 2008a] T. Zagibalov and J. Carroll. Automatic seed-word selection for unsupervised sentiment classification of chinese text. In Proceedings of the 22nd International Conference on Computational Linguistics, pages 1073–1080, 2008. 149, 152

[Zagibalov 2008b] T. Zagibalov and J. Carroll. Unsupervised classification of sentiment and objectivity in Chinese text. In Proceedings of the 3rd International Joint Conference on Natural Language Processing, pages 304–311, 2008. 22

[Zaidan 2007] O. F. Zaidan, J. Eisner and C. D. Piatko. Using annotator rationales to improve machine learning for text categorization. In Proceedings of the Annual Conference of the North American Chapter of the Association for Computational Linguistics, pages 260–267, 2007. 135

[Zheng 2004] Z. Zheng, X. Wu and R. Srihari. Feature selection for text categorization on imbalanced data. ACM SIGKDD Explorations Newsletter: Special issue on learning from imbalanced datasets, vol. 6, pages 80–89, 2004. 59

[Zubaryeva 2008] O. Zubaryeva and J. Savoy. Opinion and polarity detection within far-east languages in NTCIR-7. In Proceedings of the 7th NTCIR Workshop Meeting on Evaluation of Information Access Technologies: Information Retrieval, Question Answering and Cross-Lingual Information Access, pages 318–323, 2008. 53, 66, 69, 148

[Zubaryeva 2009] O. Zubaryeva and J. Savoy. Investigation in statistical language-independent approaches for opinion detection in English, Chinese and Japanese. In Proceedings of the 3rd International Workshop on Cross Lingual Information Access: Addressing the Information Need of Multilingual Societies, pages 38–45. ACL, 2009. 149, 151, 152

[Zubaryeva 2010a] O. Zubaryeva and J. Savoy. Evaluation de modeles de classification appliquees a la detection d'opinion. In Actes de 7ieme Conference en Recherche d'Information et Applications, pages 271–286, 2010. 36, 54, 70, 135

[Zubaryeva 2010b] O. Zubaryeva and J. Savoy. Opinion detection by combining machine learning & linguistic tools. In Proceedings of the 8th NTCIR Workshop Meeting on Evaluation of Information Access Technologies: Information Retrieval, Question Answering and Cross-Lingual Information Access, pages 221–227, 2010. 53, 54, 65, 68, 69, 80, 92, 93, 148