SENTIMENT ANALYSIS IN STREAMS OF MICROBLOGGING POSTS

Jasmina Smailović

Doctoral Dissertation
Jožef Stefan International Postgraduate School
Ljubljana, Slovenia

Supervisor: Asst. Prof. Dr. Martin Žnidaršič, Jožef Stefan Institute, Ljubljana, Slovenia, and Jožef Stefan International Postgraduate School, Ljubljana, Slovenia
Co-Supervisor: Prof. Dr. Nada Lavrač, Jožef Stefan Institute, Ljubljana, Slovenia, Jožef Stefan International Postgraduate School, Ljubljana, Slovenia, and University of Nova Gorica, Nova Gorica, Slovenia

Evaluation Board:
Asst. Prof. Dr. Tomaž Erjavec, Chair, Jožef Stefan Institute, Ljubljana, Slovenia, and Jožef Stefan International Postgraduate School, Ljubljana, Slovenia

Prof. Dr. Janez Povh, Member, Faculty of Information Studies, Novo mesto, Slovenia

Dr. Indrė Žliobaitė, Member, Aalto University, Finland, and University of Helsinki, Finland

MEDNARODNA PODIPLOMSKA ŠOLA JOŽEFA STEFANA
JOŽEF STEFAN INTERNATIONAL POSTGRADUATE SCHOOL

Jasmina Smailović

SENTIMENT ANALYSIS IN STREAMS OF MICROBLOGGING POSTS

Doctoral Dissertation

ANALIZA SENTIMENTA V TOKOVIH KRATKIH SPLETNIH SPOROČIL

Doktorska disertacija

Supervisor: Asst. Prof. Dr. Martin Žnidaršič

Co-Supervisor: Prof. Dr. Nada Lavrač

Ljubljana, Slovenia, November 2014

To my parents and sister

Acknowledgments

In the first place, I would like to express my gratitude to my supervisor Martin Žnidaršič and co-supervisor Nada Lavrač for all of their support, patience, and inspiring ideas during my doctoral study. Especially I would like to thank them for teaching me how to perform high quality research, and how to think and write like a scientist.

I am thankful to the members of my PhD committee, i.e., Tomaž Erjavec, Janez Povh, and Indrė Žliobaitė for their useful comments and suggestions, which improved the quality of the thesis.

Several funding bodies financially supported the research presented in this dissertation and I would like to thank them: Ad Futura Program of the Slovene Human Resources Development and Scholarship Fund; Department of Knowledge Technologies at the Jožef Stefan Institute; the European Commission through the research projects FIRST, FOC, SIMPOL, MULTIPLEX, and WHIM; and the research voucher funded by the Slovenian Ministry of Education, Science, Culture and Sport.

I would like to thank Sowa Labs GmbH[1] for providing the Goldfinch annotation platform for hand-labeling Twitter messages.

I am thankful to Gama System d.o.o. company[2] for their kind cooperation in the process of incorporating the sentiment analysis methodology for various languages into the PerceptionAnalytics platform.

I am also thankful to many individuals who collaborated with me: Janez Kranjc for the cooperation in implementation of ClowdFlows components and workflows related to my research, for training several language classifiers for distinguishing between tweets written in different, but similar languages, and for cooperation in the Bulgarian elections analysis use case; Dragi Kocev for his help in the statistical evaluation of the results, and inspecting and annotating a sample of Bulgarian political tweets; Vladimir Kuzmanovski for discussions concerning the statistical tests; Ulli Spankowski and Sebastian Schroff for their cooperation as financial experts in the stock analytics application; Martin Saveski for his help regarding Twitter data acquisition and active learning, suggesting the use of sofia-ml library and SWIG, and implementation of several supporting functions for the active learning algorithms; Peter Ljubič for his support in obtaining financial tweets; Matjaž Juršič for the cooperation and support related to the Twitter data manipulation for the annotation platform; Tomaž Erjavec for providing a dataset of smiley-labeled Slovene tweets; Igor Mozetič for providing useful suggestions and comments on my research and written material, and the collaboration in the Bulgarian elections analysis use case. I am particularly grateful to Miha Grčar for formulating the topic of my dissertation, providing many useful ideas (for example, about the concept of the neutral zone), his support and valuable discussions during my research, and for data, code, and advices when performing the analysis of the Slovenian elections use case.

I am grateful to all the colleagues at the Department of Knowledge Technologies for

[1] http://www.sowalabs.com/.
[2] http://www.gama-system.si/en/.

contributing to a positive, pleasant, and friendly working environment, especially to the ones I shared my office with and had many inspiring discussions: Senja, Jan, Vladimir, Borut, Matjaž, and Vid.

Thanks to all my dear longtime friends in Banja Luka and new friends in Ljubljana for always believing in me, making me laugh, and being true friends. Thank you Ana, Sanja, Saška, Milica, Marijana, Bojana, Jelena, and Daniela. Also, I would like to thank my colleagues from the Faculty of Electrical Engineering in Banja Luka for their friendship and encouragement during the years of my PhD study.

Especially, I am grateful to my dear Marko for always being there for me, sincerely celebrating my achievements, and supporting and calming me down in the most difficult moments.

Last, but not least, I am thankful to all the members of my family. Particularly, I am deeply grateful to my parents, Nedim and Besima, and sister Berina, for their endless support and faith in me. I love you so much.

Abstract

Predicting future events has always been an interesting task — from predicting weather and natural catastrophes to predicting sport outcomes, election results, and stock market assets. It seems that it is in the human nature to try to guess or calculate what will happen next. Moreover, with the advancement of computer science and methodologies for data analysis, predicting future trends and events has become easier and more accurate. Motivated by these phenomena and earlier studies, this dissertation investigates whether the opinions expressed in Twitter microblogging posts (tweets), which discuss selected companies, can indicate their future stock price changes. To detect expressed opinions we rely on sentiment analysis, a research area concerned with detecting opinions, attitudes, and emotions in texts.

In order to adjust sentiment analysis to the specific nature of Twitter messages, the thesis first presents the experiments which resulted in selecting the most suitable sentiment analysis algorithm and determining the best Twitter preprocessing setting.

An analysis whether tweets sentiment can be used for predicting stock market prices is presented in two settings: the static and the dynamic setting. In the static setting, the sentiment classifier is trained once and remains unchanged, while the dynamic setting allows its adaptation to streams of continuously arriving Twitter messages. The adaptation is performed using active learning, which periodically asks an oracle to manually label a selected set of instances. The labeled instances are used for updating the classifier.

Due to the fact that tweets do not necessarily express positive or negative opinions, the concept of *neutral zone* was used, which allowed us to employ a binary Support Vector Machine (SVM) classifier to classify tweets into three sentiment categories of positive, negative, and neutral (instead of positive and negative only). The thesis presents two definitions of the neutral zone, i.e., the fixed neutral zone and the relative neutral zone. Moreover, an indicator for predictive tweet sentiment analysis in finance, positive sentiment probability, is formalized. In the static setting, the Granger causality test showed that sentiments in stock-related tweets could be used as indicators of stock price movements a few days in advance, where improved results were achieved by employing the neutral zone, especially for the case of the relative neutral zone. These findings were adopted in the development of a new methodology for stream-based active learning approach to sentiment analysis, applicable in incremental learning from continuously changing streams of Twitter posts. A series of experiments was conducted to determine the best active learning setting for sentiment analysis in streams of tweets discussing finances of a given company.

Selected parts of the study were made publicly available through the ClowdFlows interactive data mining platform. Our sentiment analysis methodology is used also in the PerceptionAnalytics platform, which provides insights into happenings on popular social media Web sites.

The developed sentiment analysis methodology was successfully used in real-world applications for montoring the public sentiment related to Slovenian and Bulgarian elections.

Povzetek

Napovedovanje dogodkov je že od nekdaj zanimiva naloga — od napovedovanja vremena in naravnih katastrof do napovedovanja športnih rezultatov, rezultatov volitev in borznih vrednosti. Ugibanje in napovedovanje, kaj se bo zgodilo, je očitno v človeški naravi. Z napredkom računalništva in metodologij za analizo podatkov je postalo napovedovanje prihodnjih trendov in dogodkov lažje in bolj natančno. Motivirani s temi pojavi in predhodnimi študijami v tej disertaciji raziskujemo, ali izražena mnenja v kratkih spletnih Twitter sporočilih, ki govorijo o določenih podjetjih, lahko nakažejo bodoče gibanje cen njihovih delnic. Pri odkrivanju izraženih mnenj se zanašamo na analizo sentimenta, raziskovalno področje, ki se ukvarja z odkrivanjem mnenj, stališč in čustev v besedilih.

Za prilagoditev analize sentimenta na specifično naravo Twitter sporočil disertacija najprej predstavi poskuse, s katerimi izberemo najprimernejši algoritem za analizo sentimenta in najprimernejšo predobdelavo za tovrstne podatke.

Analiza uporabnosti sentimenta Twitter sporočil za napovedovanje borznih cen je narejena na dva načina: statično in dinamično. Pri statičnem načinu je klasifikator sentimenta naučen enkrat in ostane nespremenjen, medtem ko dinamični način omogoča njegovo prilagoditev tokovom nenehno prihajajočih Twitter sporočil. Prilagoditev je izvedena z uporabo aktivnega učenja, ki redno povprašuje po pravih oznakah za določene primere, ki se nato uporabijo za posodobitev klasifikatorja.

Ker Twitter sporočila ne izražajo vedno pozitivnih ali negativnih mnenj, smo uporabili koncept *nevtralne cone*, kar nam je omogočilo, da uporabimo klasifikator z metodo podpornih vektorjev za razvrščanje Twitter sporočil v pozitivno, negativno in nevtralno kategorijo (namesto le v pozitivno in negativno). Disertacija predstavlja dve definiciji nevtralne cone: fiksno nevtralno cono in relativno nevtralno cono. Formaliziran je tudi indikator verjetnosti pozitivnega sentimenta za napovedno analizo sentimenta na podlagi Twitter sporočil na področju financ. V statičnem načinu je Grangerjev test vzročnosti pokazal, da bi se sentiment v Twitter sporočilih, povezanih z delnicami, lahko uporabljal kot kazalec gibanja cen delnic za nekaj dni vnaprej, pri čemer so bili izboljšani rezultati doseženi z uporabo nevtralne cone, še posebej v primeru relativne nevtralne cone. Te ugotovitve so bile uporabljene pri razvoju nove metodologije za aktivno učenje v analizi sentimenta iz tokov podatkov, ki se lahko uporablja za inkrementalno učenje v nenehno spreminjajočih se tokovih Twitter sporočil. Izvedli smo niz poskusov za določitev najboljšega načina aktivnega učenja za analizo sentimenta v tokovih Twitter sporočil o financah določenega podjetja.

Izbrani deli študije so javno dostopni kot del platforme za interaktivno podatkovno rudarjenje ClowdFlows. Naša metodologija analize sentimenta se uporablja tudi v platformi PerceptionAnalytics, ki omogoča vpogled v dogajanje na priljubljenih spletnih straneh socialnih medijev.

Razvita metodologija analize sentimenta je bila uspešno uporabljena v aplikacijah za spremljanje sentimenta javnosti v zvezi s slovenskimi in bolgarskimi volitvami.

Contents

List of Figures

List of Tables

List of Algorithms

Abbreviations

API	...	Application Programming Interface
BOW	...	Bag-Of-Words
DJIA	...	Dow Jones Industrial Average
EDT	...	Eastern Daylight Time
EMH	...	Efficient Market Hypothesis
EST	...	Eastern Standard Time
GUI	...	Graphical User Interface
HTML	...	HyperText Markup Language
HTTP	...	HyperText Transfer Protocol
ILP	...	Inductive Logic Programming
IP	...	Internet Protocol
IR	...	Information Retrieval
JSON	...	JavaScript Object Notation
k-NN	...	k-Nearest Neighbors
LDA	...	Latent Dirichlet Allocation
MAE	...	Mean Absolute Error
NASDAQ	...	National Association of Securities Dealers Automated Quotations
NB	...	Naive Bayes
NLP	...	Natural Language Processing
POS	...	Part-Of-Speech
ROC	...	Receiver Operating Characteristic
S&P 500	...	Standard & Poor's 500
SVM	...	Support Vector Machine
TF	...	Term Frequency
TF-IDF	...	Term Frequency-Inverse Document Frequency
URL	...	Uniform Resource Locator
UTC	...	Coordinated Universal Time
WSDL	...	Web Services Description Language
XML	...	Extensible Markup Language

Symbols in Twitter Messages

@ ... at symbol. The at symbol is used for referring to a specific Twitter user by writing his username in the form *@username*.

... hash symbol. Writing together the hash symbol and a phrase in a Twitter message represents a topic or a keyword associated with the message.

$ ... dollar sign. Writing the dollar sign and a ticker symbol in Twitter messages is used for discussing stocks of a company (e.g., "$GOOG" for Google stocks).

Chapter 1

Introduction

The topic of this dissertation is the analysis of sentiment in streams of microblogging posts. In this chapter, we first provide a general introduction to social networking and microblogging services and application of sentiment analysis on messages from such services. Moreover, since we address sentiment analysis in data streams, we also introduce the method used to achieve this — the active learning approach. Our analyses are mainly performed in the context of finances, since we apply sentiment analysis on posts discussing the financial aspect of a company and examine whether there is a relationship between the sentiment and the company's future stock prices. In this chapter we explain the motivation for studying this relation. We also state the research hypotheses, the main goals of the dissertation, and its contribution to science. Finally, we describe how the dissertation is organized.

1.1 Sentiment Analysis of Microblogging Posts

In recent years, an extremely large growth of the Internet usage has been observed.[1] This phenomenon is usually attributed to the availability of new technologies and devices, simple usage and a number of benefits it brings (for example, availability of huge amounts of information, data, and fast communication with people from all around the world). Nowadays, the number of Internet users is measured in billions and it is estimated that by the end of year 2014, around 40% of the world population will use the Internet, which is nearly 3 billion people.[2] For comparison, in year 1995, only 0.4% of the population (or 16 million people) was using the Internet[3] (Hubbard, 2011). The Internet has changed the way we communicate, search for information, do our business. It even affects our daily life. A number of new Web sites, technologies, and tools have been developed for use on the Internet. Among them, many social networking and microblogging services have been developed and became very popular.

While microblogging services allow their users to read and post short messages, social network Web sites add a more personal feel. Using social network Web sites the users have the possibility to create personal profiles, post content (personal statuses, texts, pictures, or videos), interact with other users or applications, join different groups, etc. The most

[1] Information retrieved from "World Internet population has doubled in the last 5 years", http://royal.pingdom.com/2012/04/19/world-internet-population-has-doubled-in-the-last-5-years/, the URL accessed on September 4, 2014.

[2] Information retrieved from "The world in 2014: ICT facts and figures", http://www.itu.int/en/ITU-D/Statistics/Documents/facts/ICTFactsFigures2014-e.pdf, the URL accessed on August 6, 2014.

[3] Information retrieved from "Internet growth and stats", http://www.allaboutmarketresearch.com/internet.htm, the URL accessed on August 6, 2014.

popular two social networking Web sites are Facebook and Twitter.[4] Facebook[5] is a social networking Web site which was founded in 2004 and since then it has grown into a global network with 1.23 billion monthly active users (Sedghi, 2014). On the other hand, Twitter[6] is both a social networking and microblogging service, founded in 2006, which rapidly gained global popularity with 271 million monthly active users (Smith, 2014). Twitter allows its users to post short messages consisting of up to 140 characters, via the Twitter website, SMS, or various applications for mobile devices. Twitter messages are known as *tweets*. Besides writing textual messages, the users also have the possibility to upload photos or short videos. On average, around 6,000 tweets are posted every second, leading to 500 million tweets daily, and about 200 billion tweets per year.[7] Besides for the individual usage, Twitter is also particularly interesting for corporations as it allows for viral marketing and news updates (Romero, Galuba, Asur, & Huberman, 2011).

As a consequence of the popularity of the Internet, social networking, and microblogging Web sites, a large increase of on-line user generated content has been observed. More and more people post messages about their observations, opinions, and emotions about various subjects — individuals, companies, political parties, movements, or important events. Consequently, many researchers are interested in analyzing such large amounts of data in order to gain useful knowledge from it. Data from social network and microblogging Web sites is interesting and suitable for analyses because of its large volume, popularity, and capability of near-real-time publishing. This massive amount of data represents a relevant source for gathering people's viewpoints and opinions.

The Twitter data is particularly appropriate for our study since Twitter is the most popular microblogging service in the financial community (Sprenger, Tumasjan, Sandner, & Welpe, 2013), and we are mainly interested in the financial domain. Consequently, the analysis of Twitter data can provide the insights into opinions and discussions about finances. In order to analyze opinions in tweets, we apply sentiment analysis, which is the research area aiming at detecting the author's attitude, emotions, or opinion about a given topic expressed in text (B. Liu, 2010, 2012; Medhat, Hassan, & Korashy, 2014; Pang, Lee, & Vaithyanathan, 2002; Pang & Lee, 2008; Turney, 2002), where the word "sentiment" represents an attitude, view, or opinion caused by emotion.

In the context of analyzing user generated content from the Internet, the task of sentiment analysis is especially challenging since such data often contains slang (Petz et al., 2012). Moreover, messages from social media Web sites are considered noisy also for other reasons; for example, they usually contain grammatical and spelling mistakes (Petz et al., 2013). Therefore, it is important to perform appropriate preprocessing of such data in order to prepare it in the best possible way as input of sentiment analysis algorithms.

A basic task in sentiment analysis is to classify text as being positive, negative, or neutral. While some approaches perform more complex analyses (e.g., determining several dimensions of emotional states, classifying a text on a scale, for example -5 (most negative) to +5 (most positive), detecting the entity or feature discussed in a text to which the sentiment is attached, etc.), in this study we address basic sentiment analysis of Twitter messages, classifying them into two (positive or negative) or three sentiment categories (positive, negative, and neutral).

There exist several approaches which can be applied to perform sentiment analysis, i.e., the machine learning, lexicon-based, and linguistic approach (Thelwall, Buckley, &

[4]Information retrieved from "Top 15 Most Popular Social Networking Sites | August 2014", http://www.ebizmba.com/articles/social-networking-websites, the URL accessed on August 6, 2014.

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

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

[7]Information retrieved from "Internet Live Stats: Twitter Usage Statistics", http://www.internetlivestats.com/twitter-statistics/, the URL accessed on August 7, 2014.

Paltoglou, 2011). The linguistic approach (Thet, Na, Khoo, & Shakthikumar, 2009; Wilson, Wiebe, & Hwa, 2006) analyzes the grammatical structure of the text to determine its sentiment polarity. This approach tends to be computationally demanding, which is a serious drawback for use in a streaming setting. Lexicon-based methods (Smailović, Žnidaršič, & Grčar, 2011; Taboada, Brooke, Tofiloski, Voll, & Stede, 2011) employ sentiment lexicons for determining the sentiment in text. These methods are faster, but they are usually unable to adapt to changes which may occur in data streams. The machine learning approach (Pang et al., 2002) requires a data collection for learning a classifier. In the case of supervised machine learning the data must also be labeled. For Twitter sentiment analysis in data streams we find the supervised machine learning approach to be the most suitable one, thus this is the one we use in our study. This decision is supported also by the experiments presented in Section 3.3.1.

1.2 Twitter Sentiment Analysis in Data Streams

In this dissertation, we are interested also in dynamic tweet analysis, i.e., in adapting the Twitter sentiment analysis to data streams, where a data stream represents a sequence of data elements that continuously arrive from a data source (in our case, the Twitter API). A data stream can be interpreted as a stochastic process where new data elements arrive constantly and individually from each other (Gama, 2010).

In recent years, with the possibility of accessing the information in real-time, extracting knowledge from data streams has gained a high interest of the research community (Gaber, Zaslavsky, & Krishnaswamy, 2005). In a setting where huge amounts of data arrive continuously, the need for real-time analysis poses many challenges (Krempl et al., 2014) — for example, handling of changes in the stream, taking care of incomplete and delayed data instances, proper evaluation of stream-based algorithms, etc. Streaming data and its analysis is in several aspects different from non-streaming data. For example, in the context of data stream mining, an algorithm does not have information about future data instances and the order in which they will arrive from the data stream (Gama & Gaber, 2007; Gama, 2010). Therefore, in streaming scenario, the algorithm has to extract knowledge in the best possible way in real-time from new incoming data instances. After an instance has been used, it can be saved to the memory or discarded. In the case of saving instances to the memory, one should ensure that there are enough resources (in terms of memory space and processing power) for handling potentially huge amounts of data. A previously seen instance can be accessed again only if there exist processes for its storage and retrieval (Gama & Gaber, 2007; Gama, 2010). In contrast to data stream mining, in the non-streaming scenario, one operates on a limited collection of data instances which are already stored in a database or a file and are always accessible.

Using and analyzing stream data for sentiment analysis makes sense when the information about the changes in the sentiment is time-critical and a proper data flow is available. For example, it is beneficial to apply such analysis in streams of financial tweets in order to detect expressed opinions about stocks in real-time. In such a scenario the timing of writing an opinion about a company's finances is important for the investors and the relevant tweets for analysis can be obtained from the Twitter API in real-time. In this study, in order to adapt Twitter sentiment analysis to data streams, we use the active learning approach.

Active learning (Settles & Craven, 2008; Settles, 2009, 2011b) is a well-studied research area, addressing data mining scenarios where a learning algorithm can periodically select new instances to be manually labeled by a human annotator. These labeled instances are then added to the training dataset to improve the learner's performance on new data.

The aim of active learning methods is to maximize the performance of the algorithm and minimize the human labeling effort. In our study, we employ the stream-based active learning approach, meaning that the learning algorithm has to decide in real-time whether to select an incoming instance from the data stream for manual labeling or not. Therefore, the approach which would handle this scenario has to:

- have constant access to a source of data,
- have the ability to quickly and in real-time process each incoming instance and decide whether to request a label for it, and
- periodically update the model and apply it to new instances.

On the one hand, the need for using active learning is a consequence of the scarcity of hand-labeled tweets available for sentiment analysis, which prevents the use of conventional machine learning methods. Namely, it is very difficult and costly to obtain large datasets of hand-labeled tweets, especially if they are domain specific. On the other hand, the static datasets and the resulting models can soon become outdated and, therefore, continuous learning that allows for adaptations to changes with time is inevitable to keep the models up-to-date. Moreover, active learning is typically used in the scenarios where there is a large number of unlabeled data instances available, but their labeling is expensive and time consuming, which is the case for domain specific tweets that can be easily obtained through the Twitter API, but their manual labeling requires domain expertise and specific knowledge.

There are several advantages in employing stream-based active learning to sentiment analysis. First, the approach to sentiment analysis is extended with a capability of continuously adapting the sentiment classifier to changes in a data stream while incorporating selected new hand-labeled instances into the model. By doing so, the sentiment vocabulary is updated with time as new terms occur in the incoming instances. This way, by applying the active learning approach, the sentiment classifier can be made more adaptable and domain specific.

The main challenge of active learning is the selection of the most suitable data instances for hand-labeling in order to achieve the highest prediction accuracy, while knowing that one cannot afford (in terms of time and costs) to label all the available instances (Zhu, Zhang, Lin, & Shi, 2007). There exist several categories of querying strategies for selecting the most suitable instances for hand-labeling (Settles, 2009): uncertainty sampling, query by committee, expected model change, expected error reduction, variance reduction, and density weighted methods. A detailed survey of query strategies and active learning is given in Settles (2009), while we provide only a brief overview of query strategies based on Settles (2009). Query algorithms based on uncertainty sampling select the instances for which the current learner has the highest uncertainty (Lewis & Gale, 1994). Algorithms based on query-by-committee use disagreement among a collection of learners to select new instances for hand-labeling (Freund, Seung, Shamir, & Tishby, 1997; Seung, Opper, & Sompolinsky, 1992). Expected model change query strategy selects the instances which would change the model the most (Settles, Craven, & Ray, 2008). Furthermore, expected error reduction query strategies select the instances which reduce the expected future error (Roy & McCallum, 2001). Variance reduction strategies minimize the model's output variance (Cohn, Ghahramani, & Jordan, 1996). Finally, in the density weighted scenario, the data instances for hand-labeling are chosen from maximal-density regions of the unlabeled instances (Donmez, Carbonell, & Bennett, 2007; Settles & Craven, 2008; Xu, Yu, Tresp, Xu, & Wang, 2003). The active learning approach we apply in this dissertation combines uncertainty sampling and random sampling.

In the context of real-time processing of data stream instances, there are additional challenges when applying the active learning approach, such as the adaptation to changes in data distribution. The changes of statistical properties of data with time, which may occur in the data stream, are called concept drift. Žliobaitė, Bifet, Pfahringer, and Holmes (2011, 2014) propose active learning strategies that explicitly manage concept drift in data streams, and are based on uncertainty, dynamic distribution of labeling effort over time, and randomization of the search space. Our active learning approach also employs randomization of the data stream search space, but unlike Žliobaitė et al. (2011, 2014), we organize the examples into batches.

1.3 Motivation for Stock Market Analysis

In this dissertation we are primarily interested in the financial domain as we address the challenge of predicting the future value of stock market assets in the context of the explosive growth of social media and user-generated content on the Internet. Analysis of user-generated content which discusses finances poses several challenges. On the one hand, such texts are written using the specific vocabulary, and on the other hand, there is need for real-time analysis as the timing of writing an opinion about finances is important for the investors. We apply sentiment analysis and active learning techniques on Twitter microblogging posts that discuss the financial aspects of a company and examine whether there is a relationship between the tweet sentiment and future stock closing prices of a discussed company. With the term *predictive sentiment analysis* we denote an approach in which sentiment analysis is used to predict a specific phenomenon or its changes.

Predicting the future values of stock market assets is a challenge which has been investigated by numerous researchers. One of the reasons for addressing this challenge is the controversy of the efficient market hypothesis (EMH) (Fama, 1965b), which claims that stocks are always traded at their fair value. Based on this market theory all stocks are perfectly priced and it is impossible for traders to constantly outperform the average market returns. This hypothesis is based on the assumption that financial markets are informationally efficient, i.e., that the asset prices always reflect all the relevant public information about an asset at investment time.

The unpredictable nature of stock market prices was first investigated by Regnault (1863) and later by Bachelier (1900). Fama, who proposed the EMH, claimed that stock price movement is unpredictable (Fama, 1965a, 1965b). However, since the EMH is controversial, researchers from various disciplines (including economists, statisticians, finance experts, and data miners) have been investigating the means to predict future stock market prices. The findings vary: from those claiming that stock market prices are not predictable to those presenting opposite conclusions (Butler & Malaikah, 1992; Kavussanos & Dockery, 2001). The indication that there may be a relationship between emotions and stock market could be explained based on two findings (Bollen, Mao, & Zeng, 2011). On the one hand, it has been shown that emotions are crucial to thinking and social behavior (Damasio, 1995), and can influence the choice of actions. On the other hand, given that the opinions and emotions are propagated through social interactions, they can also be transferred through the investors to the stock market and consequently, the general mood of a society can be reflected in stock market values. As a result, the stock market can be considered as a measure of social mood (Nofsinger, 2005). Therefore, based on these findings, it is reasonable to expect that the analysis of the public mood can be used to predict price movements in the stock market, which is the hypothesis that we investigate in this dissertation.

Given that a massive amount of user-generated content became abundant and easily

accessible, besides the stock market applications, many researchers became interested in the predictive power of microblogging messages also in other domains: prediction of election results, prediction of the financial success of movies or books, etc. For example, it was shown that the frequency of blog posts can be used to forecast spikes in on-line consumer purchasing (Gruhl, Guha, Kumar, Novak, & Tomkins, 2005). Moreover, it was shown by Tong (2001) that there is a correlation between references to movies in newsgroups and their sales. Sentiment analysis of weblog data was successfully used to predict the financial success of movies (Mishne & Glance, 2006). Twitter microblogging posts were also shown to be useful for predicting box-office profit of movies before their release (Asur & Huberman, 2010). Even crime prediction can be advanced using Twitter data (Gerber, 2014). Therefore, in addition to the stock market use case, in this dissertation we also present the results on predicting the outcome of Slovenian and Bulgarian elections based on Twitter messages.

1.4 Hypothesis and Goals

The main aim of this dissertation is to develop a methodology for static and dynamic sentiment analysis of Twitter microblogging posts for predicting a phenomenon of interest, providing an answer to a question whether real-time sentiment analysis of tweets can be used for prediction of movements in the stock market. In the static setting, the sentiment classifier is trained using a fixed predefined training dataset and it remains unchanged throughout the experiments, while in the dynamic setting, the sentiment classifier is continuously updated with the new incoming tweets from the data stream using the active learning approach.

We first addressed a static Twitter data analysis problem, which was explored in order to determine the best Twitter-specific text preprocessing setting for training the sentiment classifier. A statistical causality test in the static setting showed that sentiment in stock-related tweets can be used as an indicator of stock price movements a few days in advance, where improved results were achieved by adapting the sentiment classifier to categorize Twitter posts into three sentiment categories of positive, negative, and neutral (instead of positive and negative only). These findings were then used in the development of a stream-based active learning approach to sentiment analysis, applicable in incremental learning from continuously changing financial tweet data streams.

The following hypotheses have been investigated:

1. Appropriate selection of preprocessing steps can improve the classification accuracy.

2. The proposed static methodology for predictive sentiment analysis on tweet data streams is capable of predicting a (financial) phenomenon of interest.

3. Identifying also the non-opinionated tweets improves their predictive power, in terms of forecasting stock market assets, as compared to the approach which assumes that all the tweets are opinionated and categorizes them as positive and negative only.

4. The active learning approach improves upon the static methodology (in terms of adapting the sentiment classifier to a specific domain and improving the F-measure of tweet sentiment classification) and improves its predictive power by adapting to changes in data streams.

5. The developed sentiment analysis methodology is applicable in real-world applications.

The goals of this dissertation can be divided into several groups, most of them consisting of several sub-goals:

1. Provide a critical review of related studies.
   (a) Give an overview of existing sentiment analysis approaches in general and in the domain of stock market and election results prediction in particular. The overview should be concerned with social networking and microblogging messages, emphasizing the existing work which has been done on Twitter data.
   (b) Provide an overview of approaches to active learning, with the emphasis on stream-based active learning.
2. Propose the most suitable approach to sentiment analysis in streams of Twitter microblogging posts in terms of the sentiment analysis algorithm and data preprocessing. In order to do so, discuss the characteristics of available approaches, how they fit into the use case, and perform experiments to show the performance results of the selected approaches. When the approach to sentiment analysis is selected, compare it with the publicly available tools.
   (a) Select the most appropriate classification algorithm for sentiment analysis of Twitter microblogging posts.
   (b) Select the most appropriate preprocessing setting for Twitter microblogging posts.
   (c) Propose an approach to identifying non-opinionated Twitter microblogging posts.
   (d) Provide an overview of a selection of publicly available sentiment classifiers and compare them with the developed classifier.
3. Collect, manually label, and make publicly available large collections of Twitter data.
   (a) Collect the appropriate financial Twitter data from the Twitter API. The queries for the API should be carefully selected in order to retrieve the most relevant results.
   (b) Acquire manual labels for the selected parts of the collected Twitter dataset.
   (c) Provide a public URL to the data, in accordance with the Twitter rules for data distribution.
4. Propose a new static and dynamic stream-based active learning methodology for predictive tweet sentiment analysis in finance.
   (a) Propose an indicator for predictive tweet sentiment analysis in finance.
   (b) Propose, implement, and evaluate static and dynamic methodology for predictive sentiment analysis of Twitter microblogging posts.
   (c) Present a data visualization approach to detect interesting events in classified data time series.
5. Apply the sentiment analysis methodology in real-life situations and examine its suitability for different data and application domains.
6. Enable public availability of selected parts of the research.

1.5 Scientific Contributions

This thesis contributes to the sentiment analysis and active learning research, and partly to better understanding of phenomena in financial stock markets. The scientific contributions (1-6) and contributions to practice (7-9) of this dissertation are the following:

1. Overview of the state of the art of existing sentiment analysis on social networking and microblogging messages, focusing on the research concerned with Twitter data, predicting stock market values and election outcomes.
2. State of the art overview of active learning approaches for data streams.
3. A systematic assessment and proposal of the best Twitter data preprocessing setting.
4. Proposing an approach to identifying non-opinionated Twitter messages.
5. Formalization of an indicator for predictive tweet sentiment analysis in finance.
6. Formalization, implementation, and evaluation of a methodology for static and dynamic predictive analysis of Twitter posts.
7. Collection of manually labeled and publicly available financial Twitter data, as a first large (in the sense of labeling effort) publicly available dataset of its kind.
8. Successful application of the developed sentiment analysis approach in two real-life domains.
9. Development of software tools and components for enabling public availability of selected parts of the research.

In relation to the CRISP-DM process model (Shearer, 2000), the third contribution belongs to the *Data Preparation* phase (and partly also to *Data Understanding*), and contributions 4, 5, and 6 to the *Modeling* phase. The *Deployment* phase of CRISP-DM is covered by contributions 8 and 9. Relations between the CRISP-DM process model phases and our methodological steps are presented in Figure 1.1.

In summary, the main contributions of this thesis are an approach to identifying non-opinionated Twitter messages (aplicable in both the static and dynamic setting) and a new methodology for stream-based active learning for tweet sentiment analysis in finance, which can be used on continuously changing tweet streams. A series of experiments was conducted to determine the best dynamic methodology, which was adapted to sentiment analysis of streams of financial tweets and applied to predictive stream mining in a financial stock market application. As a side effect, since there is no large labeled dataset of financial tweets publicly available, we have labeled and made publicly available a collection of financial tweets, making it the first large publicly available dataset of its kind.

The contributions of the dissertation and related research work were published in the following publications:

- Smailović, J., Žnidaršič, M., & Grčar, M. (2011) Web-based experimental platform for sentiment analysis. In *Proceedings of the 3rd International Conference on Information Society and Information Technologies - ISIT 2011*, 6 pages. Novo Mesto (Slovenia): Faculty of Information Studies.
- Smailović, J., Grčar, M., & Žnidaršič, M. (2012). Sentiment analysis on tweets in a financial domain. In *Proceedings of 4th Jožef Stefan International Postgraduate School Students Conference* (pp. 169–175). Ljubljana (Slovenia): Jožef Stefan International Postgraduate School. (Best ICT student paper award)

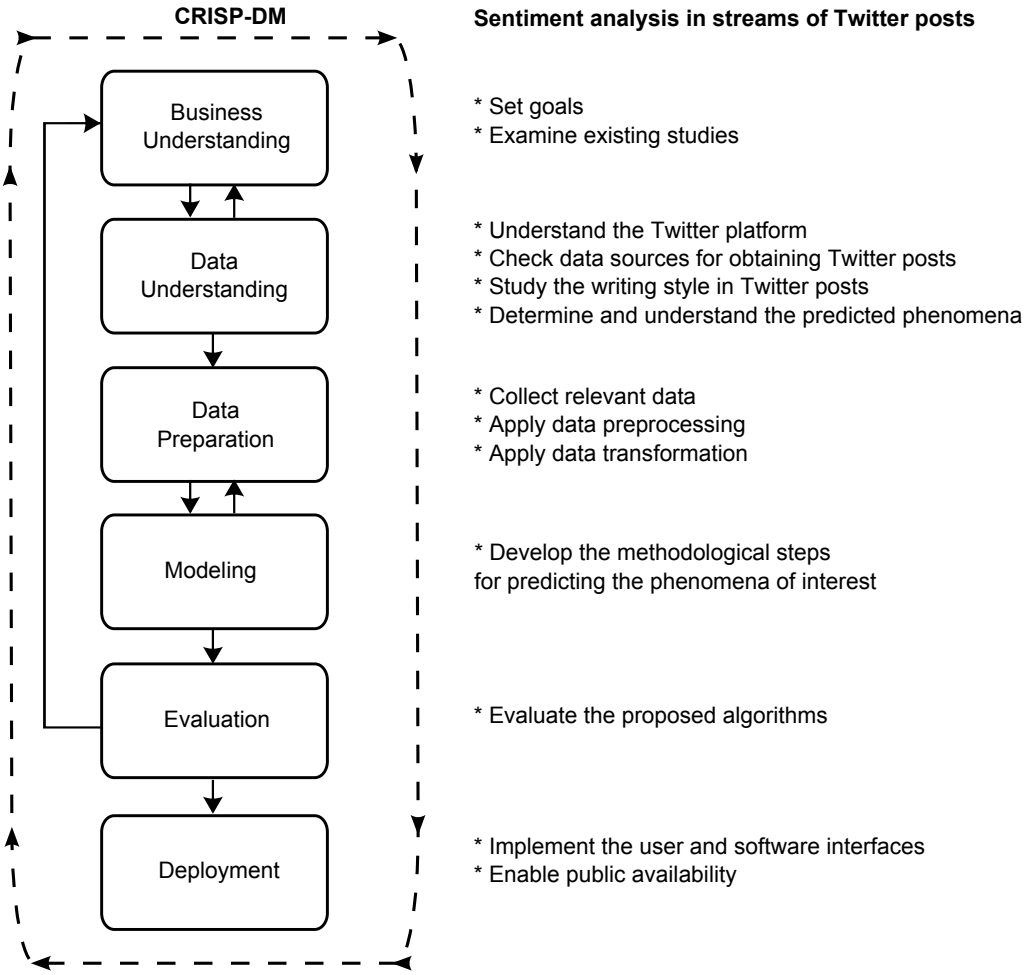

Figure 1.1: The phases of sentiment analysis in streams of Twitter posts related to the CRISP-DM process model.

- Smailović, J., Grčar, M., Lavrač, N., & Žnidaršič, M. (2013). Predictive sentiment analysis of tweets: A stock market application. In *Human-Computer Interaction and Knowledge Discovery in Complex, Unstructured, Big Data* (pp. 77–88). Lecture Notes in Computer Science Volume 7947. Springer Berlin Heidelberg.

- Smailović, J., Grčar, M., Lavrač, N., & Žnidaršič, M. (2014). Stream-based active learning for sentiment analysis in the financial domain. *Information Sciences*, 285, 181-203. Elsevier.

- Kranjc, J., Smailović, J., Podpečan, V., Grčar, M., Žnidaršič, M., & Lavrač, N. (2014). Active learning for sentiment analysis on data streams: Methodology and workflow implementation in the ClowdFlows platform. DOI:10.1016/j.ipm.2014.04.001. Information Processing & Management. Elsevier.

- Sluban, B., Smailović, J., Juršič, M., Mozetič, I., & Battiston, S. (2014). Community sentiment on environmental topics in social networks. In *Proceedings of the 10th International Conference on Signal Image Technology & Internet Based Systems (SITIS), 3rd International Workshop on Complex Networks and their Applications* (pp. 376-382).

1.6 Organization of the Dissertation

The dissertation is structured as follows. Chapter 2 presents an overview of other studies related to the research presented in this dissertation. On the one hand, it discusses the studies which were concerned with sentiment analysis in stock market prediction and in election campaigns, and on the other hand, it considers studies on the active learning approaches applied to data streams.

Chapter 3 describes the developed methodology and experiments in sentiment analysis of Twitter microblogging posts. It presents the datasets, the experiments for selecting the most suitable algorithm (in terms of execution time and classification performance) for training the sentiment classifier, and the experiments that led to the choice of the best preprocessing setting for Twitter data. In this chapter, we also compare the developed static Twitter sentiment classifier with a selection of free publicly available classifiers.

In Chapter 4 we investigate whether the sentiment in Twitter posts that discuss finances of a company can provide predictive information about the future stock prices of the discussed company. Time series data on stock closing prices and the sentiment in relevant tweets were adequately prepared and a statistical test was applied to examine the relationship between the two time series and assess the significance of the results. Moreover, due to the fact that financial tweets do not necessarily express positive or negative opinion, we employed the concept of the *neutral zone*, which allows classification of a tweet also into the neutral category, thus improving the predictive power of the sentiment classifier compared to the classifier classifying Twitter posts into the positive and negative sentiment categories only. We present two definitions of the neutral zone, i.e., the fixed neutral zone and the relative neutral zone, and apply both in our experiments to determine which one provides better results.

Chapter 5 introduces incremental learning of the sentiment classifier on a stream of financial tweets by employing the active learning approach, which periodically asks an oracle (the human annotator) to manually label the examples which it finds most suitable for labeling and updating the classifier. Using this approach the sentiment classifier was made more domain specific and better adapted to a stream-based environment.

Chapter 6 presents selected parts of our work, which were made publicly available through the ClowdFlows interactive data mining platform. Moreover, Chapter 6 presents how our sentiment analysis methodology was incorporated into the PerceptionAnalytics Web platform, which provides an insight into happenings on popular social media Web sites. We also present how the developed sentiment analysis methodology was applied in two elections use cases.

The dissertation concludes in Section 7 with a summary of results, plans for further work, and lessons learned.

Finally, Appendices A, B, and C provide additional experimental results. Since in this dissertation, in several cases we use smiley-labeled tweets for training the sentiment classifier, in Appendix A we present empirical support for considering smiley-labeled tweets as a reasonable approximation for manually labeled tweets of positive/negative sentiment. Appendix B reports experimental results of causality correlation between sentiment in tweets and closing stock price for 8 companies using the fixed neutral zone, while Appendix C provides similar experimental results, but with the application of the relative neutral zone.

Chapter 2

Related Work

This chapter provides an overview of the studies related to the research presented in this dissertation. First, we discuss the studies which are concerned with sentiment analysis. We provide a general introduction to this research area, and then focus on the applications in stock market prediction and in election campaigns, since these are the two main application domains of our work. Second, we discuss the studies interested in the active learning approach, with the emphasis on the cases adjusted to data streams.

2.1 Sentiment Analysis

Sentiment analysis (B. Liu, 2012; Medhat et al., 2014; Pang & Lee, 2008; Turney, 2002) is a research area that aims at detecting the authors sentiment, emotions or opinion about the events, topics or individuals expressed in text. Sentiment analysis is sometimes also referred to as opinion mining, and usually these two terms have identical meaning (Medhat et al., 2014). Nevertheless, some researchers explain that there exist small differences in notions of these two terms (Medhat et al., 2014). For example, Tsytsarau and Palpanas (2012) point out that opinion mining came from the information retrieval (IR) community and aims at first extracting and then processing opinions about an entity, while sentiment analysis originates from the natural language processing (NLP) community and is concerned with detecting the sentiment in given text. Moreover, Pang and Lee (2008) provide information about the terminology and the history of appearance of these two terms, but conclude that in the broad context these two terms represent the same field of study. In this dissertation we use the terms sentiment analysis and opinion mining as synonyms.

Many researchers have been studying sentiment analysis, and consequently there exist many different methods and algorithms for performing it. On the one hand, in applied research, sentiment analysis can be performed using: (i) machine learning, (ii) lexicon-based, or (iii) linguistic methods (Pang & Lee, 2008; Thelwall et al., 2011). We explain these approaches in Section 3.1.1. On the other hand, categorization of sentiment analysis techniques can be performed based on classification levels which can be: (i) document, (ii) sentence, or (iii) entity and aspect-level (B. Liu, 2012; Medhat et al., 2014). Document-level sentiment analysis (also referred to as document-level sentiment classification or sentiment classification (B. Liu, 2010)) is concerned with classifying the entire opinionated document as positive or negative. In this scenario it is assumed that a document contains an opinion about a single entity. Sentence-level sentiment analysis classifies every sentence in a document as subjective or objective, and it classifies subjective sentences as positive or negative. Finally, entity and aspect-level sentiment analysis identifies sentiment related to a particular aspect of a detected entity.

Sentiment analysis techniques can be applied on different kinds of data, such as news,

reviews, blogs, or social networking and microblogging messages. Every data type has its own characteristics, which must be taken into account during data collection, preparation, preprocessing, and feature construction. Additional challenge in sentiment analysis is the analysis of texts containing irony and sarcasm.

Nowadays, sentiment analysis is applied in many situations — tracking and aggregating opinions about a product or a company, detecting opinions for or against some movement or political party, predicting a phenomenon of interest (book or movie profit, financial assets, etc.), and in many other situations. Data from social network and microblogging Web sites (e.g., Twitter) is especially interesting for research and applications because of its large volume, popularity, and capability of near-real-time publishing of individuals' opinions and emotions about any subject. In recent years, many studies have analyzed sentiment expressed in such data in order to describe its content and study its relation to trends. For example, Thelwall et al. (2011) analyzed popular events in Twitter and showed that they are related to an increase in negative sentiment strength. Asur and Huberman (2010) constructed a model based on tweet-rate about particular topics for predicting profit of movies before their release. They further showed how sentiment extracted from Twitter posts can improve their predictive power. Bollen, Mao, and Pepe (2011) found that there is a relationship between the public mood (in terms of tension, depression, anger, vigor, fatigue, and confusion) expressed in Twitter posts and social, political, cultural, and economic events. Sentiment analysis of social network and microblogging messages was also applied in the context of stock markets and political elections. We discuss these cases in more detail in the following sections.

Besides for the research community, sentiment analysis has also been interesting for the industry, and consequently many sentiment analysis services have been developed. For example, in the US there are at least 20-30 companies that provide services for sentiment analysis (B. Liu, 2010).

There exist several survey papers that give an overview of sentiment analysis, discuss approaches and algorithms for sentiment analysis, and review its challenges and applications; for example, the papers by Pang and Lee (2008), B. Liu (2010, 2012), Tsytsarau and Palpanas (2012), and Medhat et al. (2014). In this section, we provide an overview of studies related to this dissertation, which are concerned with sentiment analysis of social media as a predictor of future stock market indicators and election results.

2.1.1 Sentiment Analysis in Stock Market Prediction

In recent years, there has been a lot of research exploring whether sentiment analysis of social media content can be used to predict future stock market indicators. The studies differ based on the collected data, applied techniques, time periods of the data, and financial indicators. In this section we provide an overview of this research.

Antweiler and M. Z. Frank (2004) analyzed messages posted on the *Yahoo! Finance*[1] and Raging Bull[2] Web sites and how they are associated with stock markets. Their experiments showed that the messages do have an impact on stock returns, although it is economically small. Moreover, they showed that the messages have the capability of predicting volatility and that there is a relationship between messages disagreement and trading volume.

Sehgal and Song (2007) analyzed sentiment in messages from the *Yahoo! Finance* Web site and demonstrated that sentiment and stock prices are closely correlated. The authors showed that one can employ sentiment analysis in order to make short-time predictions

[1] http://finance.yahoo.com.
[2] http://ragingbull.com.

about stock price behavior. Their algorithm was also able to calculate the trust value of a message's author based on his past accomplishment in predicting stock values. They also took into account that it is unrealistic for one person to be an expert for all the available stocks and that in real-life scenarios one person is usually knowledgeable only on a certain number of stocks.

Oh and Sheng (2011) analyzed sentiments in posts from Stocktwits.com[3] and *Yahoo! Finance* Web sites over a period of three months and found that stock microblog sentiments may predict future stock price movements. Additionally, the authors reported several interesting observations: for example, they found that pessimistic information has higher predictive power than optimistic information, the majority of messages are posted during trading days (reaching the maximum on Thursdays) and working hours, and that online investors are over-optimistic and over-confident even in the periods of the stock market decline.

X. Zhang, Fuehres, and Gloor (2011) measured positive and negative emotions by counting emotional words in Twitter posts over a period of six months and analyzed the correlation between these measures and stock market indices such as Dow Jones, S&P 500, NASDAQ, and VIX. They noticed that the number of Twitter posts containing positive words (hope and happy) is much higher than the Twitter posts containing negative words (fear, worry, nervous, anxious, and upset). The authors indicated that by inspecting Twitter for any kind of emotional outburst provides a predictor of the stock market performance of the following day.

Bollen, Mao, and Zeng (2011) measured mood in tweets in terms of not only positive and negative mood, but also in terms of six mood dimensions (calm, alert, sure, vital, kind, and happy) and showed that the "calm" mood dimension can predict daily up and down changes in the closing values of the Dow Jones Industrial Average (DJIA) Index. In order to demonstrate this predictive relationship the authors performed two sets of experiments. First, they applied the Granger causality analysis on all moods time series and DJIA closing values time series to check whether one time series contain predictive information about the other. Second, they compared the performance of Self-Organizing Fuzzy Neural Network for predicting the future DJIA values trained on a basis of past DJIA values alone and with the addition of mood dimensions. Chen and Lazer (2011) confirmed the results of Bollen, Mao, and Zeng (2011) and showed that even with a much simpler sentiment analysis approach, one can observe a relationship between tweet sentiment and stock market data. Mittal and Goel (2012) based their work for finding a correlation between public sentiment and the stock market on the approach of Bollen, Mao, and Zeng (2011). Their results are in some agreement with the results of Bollen, Mao, and Zeng (2011), but they indicate that not only the "calm", but also the "happy" mood dimension has a correlation with the DJIA values.

Nann, Krauss, and Schoder (2013) calculated daily sentiment of aggregated data from multiple sources (Twitter, 11 online message boards, and *Yahoo! Finance* news stream), where the data was concerned with stocks of the S&P 500 index during a six-month period. In their experiments, the authors showed that the analyzed data has predictive power for stock price changes on the following day. The authors have also illustrated the possible practical application of their study by describing a trading model, which also takes into account several real-world limitations and characteristics of the stock market (e.g., transaction costs).

Sprenger et al. (2013) analyzed about 250,000 stock-related Twitter microblogging posts over a period of six months and explored association among various values describing tweets and stocks. The authors focused on tweets discussing a selection of S&P 100

[3] http://www.stocktwits.com.

companies. They found a relationship between sentiment in tweets and stock returns, message volume and trading volume, and agreement about stock market information in posts and trading volume. Regarding the predictive abilities, they found that some of the tweet features provide predictive information about the future market features, but their experimental results showed a much stronger predictive power in the opposite direction — that stock market features provide predictive information about future tweet features. Moreover, the authors showed that the users which give high quality investment advice are recognized in the community, since they are retweeted more often and have more followers, which shows their influence and importance in microblogging forums.

Y. Yu, Duan, and Cao (2013) used volume and sentiment to study the effect of social media (blogs, forums, and Twitter) and conventional media (newspapers, television, and magazines) on companies' stock market performance in terms of stock return and risk. The authors found out that social media has a stronger impact, but also that social and conventional media together do have an effect on the stock market. They have also shown that the effect of social media varies depending on its type.

Zheludev, Smith, and Aste (2014) analyzed correlation over a period of three months between the hourly changes in sentiment in Twitter messages obtained using 44 data filters and the hourly returns of 28 financial instruments. Their experimental results showed that tweet sentiment holds statistically-significant information about the future financial returns in cases of twelve Twitter filter/financial instrument combinations. They also demonstrated that sentiment is a better indicator of future prices than volume.

Sul, Dennis, and Yuan (2014) collected tweets discussing companies in the S&P 500, determined overall daily emotional valence using a dictionary approach, and calculated its correlation with the stock market returns of individual stocks. The authors detected a significant relationship between the two. Interestingly, their results showed that the tweets written by the users which had many followers had a greater effect on the same-day stock returns, while the tweets posted by the users which had a small number of followers had an effect on the long-term (10-day) future returns.

Finally, T. Rao and Srivastava (2014) collected tweets posted over a period of 14 months and inspected their correlation with DJIA, NASDAQ-100, and 11 highly traded and discussed companies' stocks. Their experimental results showed strong correlations between various Twitter sentiment and financial features. Moreover, the Granger causality analysis results showed the causal relationship between positive and negative sentiment and stock market returns in several scenarios. The authors have also demonstrated a new way to minimize the risk in a hedged portfolio by making use of tweet sentiment features.

The above literature overview indicates that sentiment analysis of social media may contain predictive information about future stock market indicators, which is also the research topic of this dissertation. Several of the mentioned studies are in some sense similar to ours and apply some of the techniques as we do. For example, close to our research is the work of Sprenger et al. (2013), since the authors also analyzed individual stocks instead of aggregated stock market indices, and used dollar-sign notation for collecting relevant tweets as we do. However, regarding predictive abilities, they found that only some of the tweet features are useful for predicting the future market features, and that there is a much stronger predictive power in the opposite direction — that stock market features predict future tweet features. A similar general idea is used by Nann et al. (2013), however the authors were interested in aggregating data from multiple sources, whereas we are specifically interested in adjusting our approach to Twitter microblogging data. Bollen, Mao, and Zeng (2011) and T. Rao and Srivastava (2014) also applied the Granger causality analysis to check whether the tweet related time series has predictive information about the stock related time series. Nevertheless, Bollen, Mao, and Zeng (2011) collected

all available Twitter messages in a certain time period containing explicitly written mood and try to predict the DJIA index, while we are concerned with the stock-related tweets and predicting prices of individual stocks. T. Rao and Srivastava (2014), on the other hand, provide a study on the relation between various Twitter sentiment and financial features, using tweets for the stock market prediction, and minimizing the risk in a hedged portfolio. However, in our work we are also interested in detecting not only positive and negative opinions, but also the neutral ones and updating the sentiment classifier with time.

2.1.2 Sentiment Analysis in Election Campaigns

Elections are events that usually cause a lot of public interest and (emotional) response and are therefore an interesting topic for various analyses, especially in the context of predicting the final results. A survey about such studies is given in Gayo-Avello (2012, 2013). Findings differ: from those claiming that data from social media is a reliable predictor, to those arguing the opposite. This section provides an overview of studies concerned with applying Twitter data in the domain of politics and elections. A significant amount of work in this field was done on US elections and these studies are introduced first, followed by an overview of work dedicated to Twitter data and elections in several other countries. Among the first papers discussing this issue were papers by O'Connor, Balasubramanyan, Routledge, and Smith (2010) and by Tumasjan, Sprenger, Sandner, and Welpe (2010b), which reported positive results on election predictions.

The majority of existing research is concerned with the US elections. O'Connor et al. (2010) analyzed correlations between public opinion in the US measured from polls and Twitter messages. The authors performed sentiment analysis of tweets by looking at a presence of positive and negative words from the OpinionFinder[4] subjectivity lexicon. On the one hand, they found that sentiment in Twitter messages did not substantially correlate with the US presidential election polls in 2008, but on the other hand, they showed that there was a correlation with the presidential job approval and consumer confidence polls. Moreover, they found that message volume for the "Obama" topic had good correlation to the polls. But, on the other hand, the "McCain" topic also correlated to Obama's ratings in the polls.

Chung and Mustafaraj (2011) used data from the 2010 US Senate special elections in Massachusetts and the Twitter data. They first applied a prediction method which employs the proportion of tweets discussing each candidate, as in Tumasjan et al. (2010b), and then a method which calculates tweet sentiment, as in O'Connor et al. (2010). Based on their experiments, the authors argue that studies which reported positive results on election predictions based on tweet volume or sentiment have many weaknesses and that their approaches are no better than random ones. Gayo-Avello, Metaxas, and Mustafaraj (2011) tested the predictive power of Twitter data on the 2010 US Congressional elections using the slightly changed methods from Tumasjan et al. (2010b) and O'Connor et al. (2010). The authors did not find correlation between the analysis and the election results. Similarly, Metaxas, Mustafaraj, and Gayo-Avello (2011) were concerned with predicting the results of two 2010 US Congressional elections based on the method of Tumasjan et al. (2010b), and the method similar to the one of O'Connor et al. (2010). Their experiments demonstrated that the employed methods showed poor performance when predicting the election outcome. In addition, the authors proposed several standards which should be obeyed when applying methods whose goal is to predict the election outcome using data from social media. Gayo-Avello (2011) provided an analysis why the results of the 2008

[4] http://mpqa.cs.pitt.edu/opinionfinder/.

US presidential elections could not have been predicted from tweets by using the typical approaches at the time. Additionally, the author discussed several lessons that can be learned from the performed analysis.

Livne, Simmons, Adar, and Adamic (2011) analyzed tweets posted by the candidates during the midterm elections in US in 2010. Using different linear regression models, whose independent variables were graph properties, Twitter-derived variables, and candidate's and party's properties, the authors reported 88% prediction accuracy. Without using the Twitter-related variables, the accuracy was 81%.

DiGrazia, McKelvey, Bollen, and Rojas (2013) used Twitter messages which contained a Republican or Democratic candidate name and 2010 US Congressional elections data to examine whether there is any correlation between them. Their results demonstrated that there is a correlation between the proportion of mentions of a Republican candidate name and the Republican vote margin in the consecutive elections.

Finally, Jahanbakhsh and Moon (2014) analyzed correlation between politics-related tweets and the 2012 US presidential elections by employing statistics, text analysis, Latent Dirichlet Allocation (LDA), and machine learning methods. Their results showed that by analyzing Twitter data, popularity of political candidates can be discovered, i.e., Mr. Obama was more popular in the Twitter community than Mr. Romney, and Mr. Obama eventually won the elections. Moreover, by using sentiment analysis and geographical information from tweets, the authors successfully predicted election results for 76% of US states. The LDA algorithm was shown to be useful in detecting popular topics discussed in tweets.

In the following we present related work focused on Twitter data and elections in several other countries: Germany, Ireland, Spain, the Netherlands, Singapore, and Italy. In the context of the 2009 German federal elections, Tumasjan et al. (2010b) showed that people were intensively discussing politics on the Twitter microblogging platform. Moreover, the authors demonstrated that the number of tweets mentioning a certain party reveals the election outcome, while the sentiment in Twitter posts closely corresponds to the real political landscape. To determine the sentiment of tweets, the authors used the LIWC2007[5] software which detects the amount of certain cognitions and emotions presented in a given text. As already indicated above, this research initiated many discussions and new studies which examined whether it is really possible to predict the election outcome based only on the number of tweets. Several studies (for example, Chung and Mustafaraj (2011), Gayo-Avello et al. (2011), Jungherr, Jürgens, and Schoen (2012), Jungherr (2013)) criticized the proposed method. Nevertheless, the approach was also supported by others (Borondo, Morales, Losada, and Benito (2012), Tumasjan, Sprenger, Sandner, and Welpe (2010a, 2012)).

Bermingham and Smeaton (2011) developed a system which provided a real-time monitoring of Twitter messages related to the 2011 Irish general elections. Moreover, the authors used volume and supervised sentiment analysis to examine whether there is a correlation between them, on the one hand, and polls and election outcome, on the other hand. They showed that the two do have predictive power, where the volume was shown to be a better indicator. Nevertheless, the authors report that their approach is not competitive with the classical polling techniques.

Borondo et al. (2012) analyzed 370,000 Twitter messages posted by over 100,000 users and analyzed the user and politicians behavior in the context of the 2011 Spanish presidential elections. The authors found a correlation between the user activity on Twitter and the outcome of the elections. They supported the findings of Tumasjan et al. (2010b), since they showed that votes and tweet volume for each political party correspond quite

[5]Linguistic Inquiry and Word Count, http://www.liwc.net.

precisely. Moreover, they noticed that a small percentage of users attract a lot of attention and that in the Twitter sphere there is a lack of debate among the politicians.

Sang and Bos (2012) analyzed Twitter messages and their relation to the 2011 Dutch Senate elections. The authors applied the prediction method from Tumasjan et al. (2010b) based on tweet volume and applied it to their use case. The result showed that counting tweets that mention political parties is not a good predictor. Moreover, the authors showed that the performance can be improved by additional processing of the collected dataset and by performing sentiment analysis.

Skoric, Poor, Achananuparp, Lim, and Jiang (2012) examined the predictive power of Twitter messages in the context of the 2011 Singapore general elections. The authors showed that there exists a relationship between the share of tweets and the share of votes. Nevertheless, this relationship was moderately strong at the national level and much weaker at the local level. The reported mean absolute error was much higher than the one obtained by Tumasjan et al. (2010b). Additionally, Skoric et al. (2012) discuss that the Twitter prediction power depends on political and general situation in a country — democracy level, media freedom, and competitiveness of elections.

Caldarelli et al. (2014) analyzed 3.4 million Twitter posts and their volume per political party in relation to the 2013 Italian national elections. The authors conduced the analysis at the national level, three macro geographical levels, and at a smaller Italian regional scale. The results indicated that the tweet volume and its time-change are good indicators of the final election results for the national level and macro Italian areas. Additionally, based on joint mentions of various political candidates in same tweets, the authors analyzed how similar are the candidates.

The presented overview of related work indicates that a lot of research was dedicated to the analysis of correlation between the volume and sentiment of Twitter posts and election outcomes in various scenarios. The conclusions are mixed, sometimes even conflicting.

2.2 Active Learning

In active learning (Settles & Craven, 2008; Settles, 2009, 2011b) a learning algorithm has a collection of unlabeled instances at its disposal, but it can select only a limited number of them for hand-labeling. The labeled instances are then used for learning. The goal of active learning is to select instances in the best possible way so that the performance of the algorithm is maximized and the hand-labeling effort is minimized. Active learning is particularly useful in situations when labeling of the instances is difficult, time-consuming, or expensive (Settles, 2009) and one cannot afford to label all the obtained instances. Such a situation can be observed in the case of domain specific Twitter microblogging posts which can be easily obtained using the Twitter API, but their hand-labeling requires domain expertise and special knowledge.

Active learning has been applied in several domains. For example, in spam filtering (Chu, Zinkevich, Li, Thomas, & Tseng, 2011; Sculley, 2007), part-of-speech tagging (Argamon-Engelson & Dagan, 1999), text classification (Settles et al., 2008; P. Wang, Zhang, & Guo, 2012), classification of medical images (Hoi, Jin, Zhu, & Lyu, 2006), and image retrieval (Settles et al., 2008). Settles (2011b) reports that big companies such as CiteSeer, Google, IBM, Microsoft, and Siemens all apply active learning.

Active learning has been studied in three different scenarios: (i) membership query synthesis, (ii) pool-based sampling, and (iii) stream-based selective sampling (Settles, 2008, 2009). In the membership query synthesis scenario, the learner can select new examples for hand-labeling from a collection of unlabeled instances or it can generate completely new instances by itself. In the pool-based scenario, the learner has at its disposal a large

pool of historical data. Based on the analysis of the whole data pool the learner requests labeling for the examples which it finds the most suitable. Finally, in the stream-based active learning scenario, examples are made available constantly from a data stream and the learner has to decide in real-time whether to request a label for a new example or not.

In this dissertation, we are interested in the stream-based active learning scenario applied in a stream of Twitter microblogging messages. Therefore, in this section we provide an overview of the studies related to this topic.

2.2.1 Stream-based Active Learning

Nowadays, different kinds of data can be obtained constantly in real-time using APIs, sensors, cameras, and other sources. Consequently, an algorithm which would analyze such constantly incoming data has to be up-to-date with changes in the data stream, provide fast data processing, and deal with large data volumes in real-time. Classical supervised machine learning techniques are not suitable for this task, since they require a large labeled dataset, which is particularly difficult to obtain in a real-time streaming setting due to large volumes and constantly changing data. Moreover, even if a suitable dataset for training the model is obtained, the model can soon become outdated due to the changes in the data stream. Stream-based active learning is a reasonable solution to this problem, since its main task is to continuously select the most suitable examples from the data stream in order to update the model.

Active learning on data streams has been analyzed in several studies. One of the simplest ways to select the examples to be labeled is based on maximizing the expected informativeness of labeled examples. For example, the learner may find the examples with the highest uncertainty to be the most informative and request them to be labeled. Zhu et al. (2007) applied (among other strategies) uncertainty sampling in two scenarios: (i) local uncertainty sampling which selects instances for labeling based only on a current batch of data from the data stream, and (ii) global uncertainty sampling which in the selection process takes into account also previous classifiers and thus forms a classifier ensemble.

Žliobaitė et al. (2011, 2014) proposed strategies that extend the fixed uncertainty strategy with dynamic allocation of labeling efforts over time and randomization of the search space. The latter approach was used also in some of our active learning strategies. The active learning strategies proposed in Žliobaitė et al. (2011, 2014) explicitly handle concept drift, i.e., adapt the classifier to data distribution changes in data streams over time. The authors do not consider batches as we do, but perform labeling decisions on every encountered data instance. Sculley (2007) and Chu et al. (2011) also examine instances which come from a data stream one by one, while Zhu et al. (2007) partition instances from data stream into batches. Furthermore, labeling budget management in the study of Žliobaitė et al. (2011, 2014) is different, as it uses a fixed overall budget within which the active learning rate is dynamically adapted. We opted for a fixed budget per batch, which enables the labeling effort to remain constant in each time period. This was perceived as a favorable approach from the user's point of view, as in our case the labeling cost is measured in human time, which is difficult to provide in unevenly dispersed bursts.

Deciding which instances are the most suitable for labeling can be made by a single evolving classifier (Žliobaitė et al., 2011, 2014) or by a classifier ensemble (P. Wang et al., 2012; Zhu et al., 2007, 2010). In classifier-ensemble-based active learning frameworks, a number of classifiers are trained from small portions of stream data. These classifiers construct an ensemble classifier for predictions (P. Zhang, Zhu, & Guo, 2009). Our work is concerned with the development of a single evolving sentiment classifier for Twitter posts.

Active learning on data streams for sentiment analysis of Twitter posts is still insufficiently explored and represents a significant challenge. Preliminary work of our research group at the Department of Knowledge Technologies of the Jožef Stefan Institute is presented in Saveski and Grčar (2011). We provide a study on stream-based active learning for sentiment analysis in the financial domain in Smailović, Grčar, Lavrač, and Žnidaršič (2014), where we develop a single sentiment classifier, organize data from a data stream into batches, and employ active learning query strategies based on uncertainty sampling, random sampling, and combination of both in order to select the most suitable examples for hand-labeling and incrementally updating the classifier. The developed methodology was applied to predict future stock price changes of a company that was discussed in tweets. The active learning methodology described in this dissertation is improved compared to the one in Smailović et al. (2014), since in the current version of the active learning methodology, not only the sentiment classifier, but also the neutral zone is dynamically updated with time (see Section 5.1.2). Finally, in Kranjc et al. (2014) we present an implementation of active learning on data streams for sentiment analysis in the ClowdFlows data mining platform.

We are not aware of any study that is concerned with the specific task of stream-based active learning for sentiment analysis of Twitter posts. There exist several papers which are related to it. For example, Bifet and E. Frank (2010) discussed challenges which arise when analyzing Twitter data streams in the classification setting, and consider such streams for sentiment analysis. The authors did not apply the active learning approach. Furthermore, Bifet, Holmes, and Pfahringer (2011) presented MOA-TweetReader, a system for real-time analysis of Twitter messages, which detects changes in word frequencies and performs sentiment analysis. Settles (2011a) developed an active learning annotation tool, DUALIST; while the author showed its potential by applying it to sentiment analysis of general tweets, his tool is not specifically adjusted to tweet analysis or data streams. Finally, H. Wang, Can, Kazemzadeh, Bar, and Narayanan (2012) developed a system for Twitter sentiment analysis in real-time for the US elections use case. In the system the user can observe aggregated sentiment, volume, statistics, trending words, tag clouds, or individual tweets. It also offers the possibility for a user to label an arbitrarily chosen tweet, but in the described implementation the system does not suggest tweets for labeling, or use labeled tweets for updating the sentiment model.

Chapter 3

Sentiment Classification

Sentiment analysis, which is also referred to as opinion mining (B. Liu, 2010, 2012; Medhat et al., 2014; Pang et al., 2002; Pang & Lee, 2008; Turney, 2002), is a research area, which aims at detecting the author's attitude, emotions, or opinion about a given topic expressed in text. Nowadays, with the massive use of the Internet, more and more people share their emotions and personal opinions about individuals, companies, products, movements, or important events on blogs, forums, and social networking Web sites. Such written opinions do not only express, but also evoke sentiments in readers of the written material (Y. Rao, Li, Mao, & Wenyin, 2014). The analysis of this huge amount of data can provide an insight into the current mood of the society about different topics or even help predict future trends and events.

In this dissertation we are interested in sentiment analysis of Web posts from the Twitter microblogging service. This service is suitable for our research because of its popularity and the possibility to access its microblogging posts through the Twitter API, which consequently means that we can obtain a large amount of Twitter messages about any subject for analysis. Furthermore, frequent posting of messages allows us to analyze not only a fixed amount of pre-collected Twitter messages, but also to apply the algorithms in real-time to streams of Twitter data as will be demonstrated in the subsequent chapters of this thesis.

This chapter describes the methodology and the experiments for sentiment analysis of Twitter microblogging posts. We describe the Twitter dataset used, the experiments for selecting the most suitable sentiment analysis algorithm, and the experiments that led to the choice of the best preprocessing setting for Twitter data. We also compare the developed sentiment classifier with several classifiers, which are publicly available. The experiments on Twitter data preprocessing are based on our studies presented in Smailović, Grčar, and Žnidaršič (2012, 2013, 2014), with the preprocessing settings presented in this chapter being further refined compared with our already published work. Preliminary experiments on selecting the sentiment analysis algorithm are provided in Kranjc et al. (2014). Comparison with publicly available sentiment classifiers has not been published yet.

3.1 Methodology

Sentiment analysis is a very popular and actively explored research area, and therefore, there exists a range of different methods and algorithms for applying sentiment classification on texts. Each of these methods and algorithms has its advantages and weaknesses, and the final choice depends on a domain of interest, available datasets, tools for text processing and analysis, researcher's experience, and the results of empirical assessment.

First, one needs to decide which general approach she will use: the machine learning, the lexicon-based, or the linguistic approach (Thelwall et al., 2011). After an approach is selected, a specific algorithm needs to be chosen. For example, for the machine learning approach, there are many possible algorithms that can be applied to sentiment analysis: Support Vector Machine (SVM), Naive Bayes (NB), k-Nearest Neighbors (k-NN), etc. Besides the approach and the algorithm, an important step is also to decide which text preprocessing techniques should be applied on the data in order to improve its quality and prepare it for the algorithm. This section discusses these steps in the light of the general task of the dissertation — sentiment analysis in streams of Twitter microblogging posts.

3.1.1 Sentiment Analysis Algorithm

There are three common approaches to sentiment analysis (Thelwall et al., 2011):

- machine learning,
- lexicon-based, and
- linguistic approach.

There exist several machine learning algorithm types which can be used in the context of sentiment analysis. Most of the approaches to determining the sentiment polarity of a document are based on supervised learning, but some of them employ also unsupervised learning methods (B. Liu, 2010). Standard supervised machine learning methods require a class-labeled collection of documents, which have been pre-categorized manually or in some other presumably reliable way. A labeled dataset is used to train a model which is able to classify new unlabeled documents. Documents are usually described by vectors of features where the length of the feature vectors corresponds to the number of features. The calculation of feature vector values can be arbitrarily complex, but they are usually a function of frequencies of single or multiple terms detected in the document or document collection. On the other hand, unsupervised machine learning methods operate on unlabeled documents. These methods analyze documents' patterns and features, and organize the documents into groups of similar ones. In the context of a data stream and changes which can occur in it, the supervised machine learning approach is highly suitable since the classification model can be flexible, as it allows to be updated over time using new labeled documents.

Lexicon-based methods use predefined collections of sentiment words and predict overall sentiment polarity of an analyzed text based on the occurrences of the sentiment words in it (Smailović et al., 2011; Taboada et al., 2011). Lexicon-based methods are fast, but they are usually unable to adapt to changes in data stream, which can be a serious drawback in the continuous real-time analysis of public sentiment. Moreover, lexicon-based methods usually rely on explicitly expressed sentiment in the analyzed text and do not take into account the context and the terminology that may implicitly carry sentiment.

Linguistic methods analyze the grammatical structure of the text in order to determine the sentiment polarity (Thet et al., 2009; Wilson et al., 2006). They can be combined with a lexicon (Thelwall et al., 2011). In the context of sentiment analysis on data streams, the linguistic methods pose several challenges, as they tend to be too computationally demanding for the use in a streaming near real-time setting. Also, additional challenge in the context of Twitter sentiment analysis is that there is a lack of readily available tools for parsing tweets.

Therefore, based on the observed challenges of lexicon-based and linguistic methods in the context of real-time sentiment analysis of tweets, and the experiments described in

Section 3.3.1, we applied a supervised machine learning approach in our study. In order to select the most suitable machine learning algorithm for sentiment analysis of Twitter data, we experimented with several algorithms, i.e., with the Support Vector Machine (Boser, Guyon, & Vapnik, 1992; Cortes & Vapnik, 1995; Vapnik, 1995, 1998), Naive Bayes (N. Friedman, Geiger, & Goldszmidt, 1997) and the k-Nearest Neighbors algorithm (Cover & Hart, 1967; Duda & Hart, 1973), which are all potentially applicable to the Twitter sentiment analysis task. Based on the empirical assessment (see Section 3.3.1) we opted for the linear SVM algorithm.

The SVM algorithm (Boser et al., 1992; Cortes & Vapnik, 1995; Vapnik, 1995, 1998) is theoretically well motivated and has been often successfully applied. Moreover, it has several advantages, which are important for learning a sentiment classifier from a large dataset; for example, it is fairly robust to overfitting and it can handle large feature spaces (Chang & Lin, 2011; Joachims, 1998; Sebastiani, 2002). A linear SVM algorithm represents the positively and negatively labeled training examples as points in the high-dimensional space and separates them by a hyperplane. There are many possible hyperplanes which could separate the training examples that belong to different classes, but the aim of the SVM algorithm is to choose the one which separates them with the largest possible gap, i.e., for which the margin between the training examples of both classes and the SVM hyperplane is as large as possible (Guyon, Weston, Barnhill, & Vapnik, 2002). In general, the larger margin indicates the lower classification error of the new unseen examples. The training examples which are closest to the SVM hyperplane, i.e., which are positioned on the margin, are called support vectors. New unlabeled examples, for which a label should be determined, are mapped into the same space as the training examples and are classified based on the side of the hyperplane in which they reside. The linear SVM decision function used for the classification of a new example has the following form (Guyon et al., 2002):

$$D(x) = x \times w + b \tag{3.1}$$

where, x is the feature vector of the new input example to be classified, w is the SVM weight vector, and b is the hyperplane bias. The SVM weight vector w is a function of support vectors (Guyon et al., 2002). If a value of $D(x)$ is positive, the example is classified as positive, and if it is negative, the example is classified as negative. Every feature from the feature vector x and a corresponding value from the SVM weight vector contribute to the final classification decision. A positive product between the two means that the corresponding feature contributes to classifying the example as positive. Similarly, a negative product means that the corresponding feature contributes to classifying the example as negative.

For training the tweet sentiment classifier, we used the SVM^{perf} (Joachims, 2005, 2006; Joachims & C.-N. J. Yu, 2009) implementation of the SVM algorithm.

3.1.2 Data Preprocessing

Data preprocessing is an important step in sentiment analysis since with the appropriate selection of preprocessing techniques, the classification accuracy can be improved (Haddi, Liu, & Shi, 2013; Parikh & Movassate, 2009). We apply both Twitter-specific and standard preprocessing on the data. The specific preprocessing is especially important for Twitter messages, since the Twitter community has created its own unique phrases and forms to write messages. Moreover, user-generated content in social media often contains slang (Petz et al., 2012) and frequent grammatical and spelling mistakes (Petz et al., 2013). Therefore, with Twitter-specific text preprocessing we try to handle these proper-

ties of the Twitter language and improve the quality of features. We consider the following options (Agarwal, Xie, Vovsha, Rambow, & Passonneau, 2011; Go, Bhayani, & Huang, 2009; Smailović et al., 2012, 2013, 2014) for Twitter-specific preprocessing to better define the feature space:

- **Usernames**: Mentioning of other users in a tweet in the form *@TwitterUser* is transformed to *atttTwitterUser*. The motivation for this transformation lies in the fact that the tokenizer we use, considers the *at* simbol ("@") and an username as two separate words. By replacing the *at* symbol by the *attt* word, we force the tokenizer to consider a username as one single word.

- **Stock symbols**: In Twitter community, the dollar-sign notation (e.g., "$GOOG" for Google stocks) is used for discussing stock symbols. We transform stock symbols in a form *$Symbol* to *stockSymbol*. The motivation for this transformation is the same as for the usernames.

- **Usage of Web links**: Web links pointing to different Web pages are replaced by a single token named *URL*. We consider a Web link to be every character sequence starting either with *http* or *www*.

- **Hashtags**: A hashtag is a word which consists of the hash symbol ("#") and a phrase. It represents a group or a topic associated with a tweet. We replace hash symbols in hashtags in a tweet by the *hash* word. For example, a hashtag *#bowling* is transformed to *hashbowling*. The motivation for this transformation is the same as for the usernames.

- **Exclamation and question marks**: Successive occurrences of exclamation and question marks are replaced by a *MULTIMIX* token (for example, *?!??!!?* is replaced by *MULTIMIX*). Two or more exclamation marks are replaced by a single token *MULTIPLEEXCLAMATION* and a single exclamation mark by a token *SINGLEEXCLAMATION*. Similarly, two or more question marks are replaced by a single token *MULTIPLEQUESTION* and a single question mark by a token *SINGLEQUESTION*.

- **Letter repetition**: repetitive letters with more than three occurrences in a word are replaced by a word with three occurrences of this letter (e.g., word *goooooooooood* is replaced by *goood*). With this transformation we preserve information about overemphasized repetitive letters.

- **Negations**: we replace negation words (*not, isn't, aren't, wasn't, weren't, hasn't, haven't, hadn't, doesn't, don't, didn't*) with a unique token *NEGATION*. Using this approach, we do not lose information about a negation, but treat all negation expressions in the same way.

Besides the Twitter-specific text preprocessing, we also consider standard text preprocessing techniques (Feldman & Sanger, 2007) in order to better define and reduce the feature space. These involve applying text tokenization (text splitting into individual words/terms), testing whether stop word removal (removing words which do not contain relevant information, e.g., *a, an, the, and, but, if, or*, etc.) is beneficial, performing stemming (converting words into their base or root form), and n-gram construction (concatenating 1 to n stemmed words appearing consecutively in a tweet.). We set the value of n for n-gram construction to 2, meaning that we construct unigrams (an n-gram of size 1,

which is a single stemmed word) and bigrams (an n-gram of size 2, made by concatenating two stemmed words which appear successively in a tweet). Examples of unigrams and bigrams are provided in Section 3.3.2. The resulting terms are used in the construction of feature vectors that represent the tweets.

The standard approach to feature vector construction is TF-IDF-based, where TF-IDF stands for the "term frequency-inverse document frequency" feature weighting scheme (Joachims, 1998; Yang & X. Liu, 1999). In the TF-IDF scheme a weight reflects how important a word is to a document in a document collection (TF-IDF increases proportionally to the number of times a word is present in the document, but decreases with respect to the number of documents in which the word occurs). In the classification setting, however, TF (term frequency)-based approach, where a weight reflects how often a word is found in a document, performs better than the TF-IDF-based approach (Martineau & Finin, 2009). Moreover, in our study in (Smailović et al., 2014), we showed that the TF-based approach is statistically significantly better than the TF-IDF approach in the Twitter sentiment classification setting. Therefore, the feature vector construction in our experiments is based on the TF weighting scheme.

3.2 Experimental Setting

In this section we introduce the datasets used in our experiments and provide details about experimental settings for two series of experiments: (i) the ones concerned with the selection of the base sentiment analysis algorithm, and (ii) the assessment of various text preprocessing settings.

3.2.1 Datasets

For training the sentiment classifier and evaluation of the results we used two publicly available Twitter datasets (Go et al., 2009).[1]

The first one is a large collection of 1,600,000 (800,000 positive and 800,000 negative) tweets collected and prepared by Stanford University (Go et al., 2009), where the tweets were labeled based on a presence of positive and negative emoticons. In this dataset, the emoticons approximate the actual positive and negative sentiment labels. This approach was proposed by Read (2005). For example, if a tweet contains the ":)" emoticon, it is labeled as positive, and if it contains the ":(" emoticon, it is labeled as negative. The full list of emoticons used for labeling can be found in Table 3.1. Inevitably, this approach causes only partially correct (noisy) labeling. However, in Appendix A, we illustrate that smiley-labeled tweets are still a reasonable approximation for manually-annotated positive/negative sentiments of tweets. In the training data, the tweets containing both positive and negative emoticons, retweets and duplicate tweets were already removed (Go et al., 2009). The emoticons, which approximate sentiment labels, were also already removed from the tweets in order not to put too much weight on them in the training phase. Therefore, the classifier was learned from the other features of tweets. The Twitter messages in this collection are from the time period between April and June, 2009. They are general and do not belong to any particular domain.

The second dataset is a test dataset collected and manually labeled also by Go et al. (2009). This dataset contains tweets belonging to different domains (companies, people, products, etc.). It consists of 498 hand-labeled tweets, of which 182 were labeled as positive, 177 as negative, and 139 as neutral. The tweets were manually labeled based on

[1]The datasets were obtained from "For Academics" section on the Sentiment140 Web page, available at the following URL: http://help.sentiment140.com/for-students.

Table 3.1: List of emoticons used for labeling the training set.

Positive emoticons	Negative emoticons
:)	:(
:-)	:-(
:)	: (
:D	
=)	

their sentiment, regardless of the presence of emoticons in the tweets. The tweets in this collection are from May and June, 2009.

3.2.2 Sentiment Analysis Algorithm Selection

In the experiments for determining the best approach for sentiment classification, we used the 177 negative and 182 positive hand-labeled tweets from the Stanford test set (Go et al., 2009) for performance evaluation. For the machine learning approach the collection of 1,600,000 smiley-labeled tweets (Go et al., 2009) was used for training a selection of classifiers. We calculated the accuracy on the test set, in the same way as Go et al. (2009) for the results to be comparable. The accuracy is the fraction of all examples which are correctly classified. It is calculated as:

$$Accuracy = \frac{\text{Number of Correctly Classified Examples}}{\text{Number of All Examples}} \times 100\% \tag{3.2}$$

After the experiments for selecting the best sentiment classification approach (see Table 3.2), which showed that the machine learning approach is more suitable, we employed an additional performance evaluation of the selected machine learning algorithms (SVM, Naive Bayes, and k-NN). The best algorithm was determined according to the ten-fold cross-validation method (Kohavi, 1995) on the larger dataset of 1,600,000 smiley-labeled tweets (Go et al., 2009). When using ten-fold cross-validation method for evaluation, the model is trained and tested ten times. The dataset is randomly split into ten approximately equal size subsets and, in each iteration, one subset is used for testing the model, while the remaining nine subsets are used for training the model. Therefore, each of the ten subsets is used nine times for training and once for testing. Finally, the average performance can be determined. We applied the stratified ten-fold cross-validation method which ensures that each subset contains roughly the same label distribution as the original dataset (Kohavi, 1995).

The performance of the selected machine learning algorithms was measured using the accuracy and the F-measure (Van Rijsbergen, 1974) of positive examples. The F-measure (also known as F-score or F_1 score) is a harmonic mean of precision and recall, whose best value is 1 and worst is 0. It is calculated as:

$$F\text{-}measure = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{3.3}$$

where, when calculating the F-measure of positive examples, precision is the fraction of all the examples classified as positive which are correctly classified as positive, while recall is the fraction of all the positive examples that are correctly classified as positive.

3.2.3 Preprocessing Experiments

In the experiments aimed at the selection of the best Twitter preprocessing setting, the best setting was determined using the stratified ten-fold cross-validation method on 1,600,000 smiley-labeled tweets (Go et al., 2009). The performance of the different preprocessing settings was measured using both the accuracy and the F-measure of positive examples, but the overall best setting was determined based on the best F-measure result.

The approach to tweet preprocessing and classifier training was implemented using the LATINO[2] software library of text processing and data mining algorithms.

3.3 Experimental Results

This section presents the results of empirical evaluation for determining the best approach to Twitter sentiment analysis in terms of classification algorithm and data preprocessing. We also provide a comparison of the developed sentiment classifier with a selection of publicly available sentiment classifiers. The experiments were performed using the datasets and the methods described in Section 3.2.

3.3.1 Selection of the Sentiment Analysis Algorithm

We evaluated several algorithms for Twitter sentiment analysis in order to select the most suitable one for our study. In a streaming near real-time setting, which is a general context of this dissertation, the linguistic approach poses several challenges (as explained in Section 3.1.1). As a result, we found this approach inadequate for our study, and therefore we consider only the two other approaches: the machine learning and the lexicon-based approach.

We performed two experiments. In the first experiment, we tested a selection of machine learning algorithms and a lexicon-based method on 177 negative and 182 positive hand-labeled tweets (Go et al., 2009). The results in Table 3.2 show that, in this setting, the machine learning approach is more appropriate. In the second experiment, we conducted additional testing of the machine learning algorithms, to further compare their performances using the stratified ten-fold cross-validation method on 1,600,000 smiley-labeled tweets (Go et al., 2009) in order to choose the best performing one for the use in the rest of our studies and analyses.

Therefore, in the first experiment for the machine learning approach we tested the linear SVM (Boser et al., 1992; Cortes & Vapnik, 1995; Vapnik, 1995, 1998), Naive Bayes (N. Friedman et al., 1997) and k-Nearest Neighbors algorithm (Cover & Hart, 1967; Duda & Hart, 1973). For the linear SVM we used the SVM^{perf} (Joachims, 2005, 2006; Joachims & C.-N. J. Yu, 2009) implementation, and for Naive Bayes and k-NN we used the implementations from the LATINO library. We trained the sentiment classifier on the collection of 1,600,000 smiley-labeled tweets (Go et al., 2009) and tested it on 177 negative and 182 positive hand-labeled tweets (Go et al., 2009). We applied standard text preprocessing (text tokenization, stop word removing, stemming, unigram and bigram construction) on the tweets. The resulting terms were used as features in the construction of feature vectors representing the tweets, where feature vector construction was based on the TF feature weighting scheme. We also added the condition that a given term has to appear at least twice in the entire corpus, either twice in a given tweet or in two different tweets. In these experiments, we achieved the accuracies of 79.11% for SVM, 75.21% for Naive Bayes, and 72.98% for the k-NN classifier with k=10 on the test set.

[2]LATINO (Link Analysis and Text Mining Toolbox) software library is available at http://source.ijs.si/mgrcar/latino.

Table 3.2: Evaluation performance (accuracy) for different approaches to sentiment analysis on the test set.

SVM	NB	k-NN	Lexicon method
79.11%	75.21%	72.98%	73.54%

We have also tested a lexicon method on the same hand-labeled test set as for the machine learning approach. In the lexicon-based method, we used the opinion lexicon containing 2,006 positive and 4,783 negative words[3] (Hu & B. Liu, 2004; B. Liu, Hu, & Cheng, 2005). The lexicon is adjusted to social media content, as it also contains many misspelled words, which are often used in the social media language. We calculated the positive and negative score for each tweet, based on the occurrences of positive and negative lexicon words in them. For example, if a tweet contains the word 'love' from the positive lexicon list, the positive score will increase by one. The score will not increase if the currently observed lexicon word contains or is contained in some of the previously seen lexicon words for that specific class in the observed tweet. If the resulting positive score for a tweet is the same or higher than the negative score, the tweet is labeled as positive. If it is lower, it is labeled as negative. The tweets with equal positive and negative scores are labeled as positive, since the positive lexicon list contains less words. In this experiment we achieved the accuracy of 73.54% on the test set.

Accuracies for all the tested approaches are presented in Table 3.2. As can be seen from the table, the two machine learning approaches (SVM and NB) outperformed the lexicon-based approach, while the third machine learning approach (k-NN) achieved lower, but comparable, performance result as the lexicon-based method. Therefore, based on the results and the nature of machine learning approach, which suits our needs for the real-time Twitter sentiment analysis, the machine learning approach was identified to be the most suitable one for our study. Moreover, the results also indicate that the best machine learning algorithm is the SVM. Nevertheless, we conducted an additional experiment to further test the performances of the selected machine learning algorithms.

In the second experiment, we employed stratified ten-fold cross-validation on 1,600,000 smiley-labeled tweets and calculated average accuracy and F-measure for the selected machine learning algorithms. We applied standard text preprocessing (text tokenization, stop word removing, stemming, unigram and bigram construction) on the tweets. In this setting, the k-NN algorithm proved to be too slow (in ten-fold cross-validation experiments for k=5 and k=10, the one-fold experiment took more than 24 hours on a standard desktop computer), Naive Bayes achieved an average F-measure of 0.7586 and the accuracy of 75.84%, and SVM achieved an average F-measure of 0.7850 and the accuracy of 78.55%. Therefore, the SVM had higher performance compared to the Naive Bayes, and the k-NN algorithm proved to be too slow in this particular setting. Also the results of the first experiment indicated that the best performing machine learning algorithm was the SVM (see Table 3.2). We, thus, used the SVM approach in the rest of our studies and analyses.

3.3.2 Preprocessing Experiments

In Section 3.1.2 we discussed several preprocessing options for improving the data quality and constructing a better feature space. In this section, we present the experiments for determining which combination of the discussed preprocessing options is the best one

[3]The opinion lexicon was obtained from the following URL: http://www.cs.uic.edu/~liub/FBS/sentiment-analysis.html#lexicon.

for the Twitter sentiment analysis use case. The first two Twitter-specific preprocessing options (usernames and stock symbols transformations) were always applied and we experimented with all the combinations of the rest of the discussed transformations. The text of the Twitter messages was first converted to lowercase, and usernames and stock symbols transformations were applied. Then, we experimented with different combinations of the optional transformations in order to find the best combination. The transformations were applied in the order as listed in Section 3.1.2.

In addition to Twitter-specific text preprocessing, other standard preprocessing steps were applied (Feldman & Sanger, 2007) to define the feature space for tweet feature vector construction. These included text tokenization,[4] stemming,[5] and n-gram construction for feature space reduction. In our experiments n had a value of 2, meaning that we were constructing both unigrams and bigrams. We experimented with the removal of stop words in order to test if it is beneficial. We also added the condition that a given term has to appear at least twice in the entire corpus, either twice in a given tweet or in two different tweets. This is an appropriate condition since Saif, Fernandez, He, and Alani (2014) showed that removing words which occur only once in the corpus reduces the feature space by nearly 65%, while preserving the classification performance. We did not employ cutting low weight features. The resulting terms were used as features in the construction of TF feature vectors representing the documents (tweets). We did not use a part-of-speech (POS) tagger, since it was indicated by Go et al. (2009) and Pang et al. (2002) that POS tags are not useful when using SVMs for sentiment analysis. Moreover, Kouloumpis, Wilson, and Moore (2011) showed that POS features may not be useful for sentiment analysis in the microblogging domain.

The results of preprocessing experiments are shown in Table 3.3. The best preprocessing setting is Setting 1, which is shown in the first row of Table 3.3. It performs tokenization and stemming, it is TF-based, uses maximum n-grams of size 2, words which appear at least two times in the corpus, employs usernames and stock symbols transformations, replaces hash symbols in hashtags in a tweet by the *hash* word, and replaces repetitive letters with more than three occurrences in a word by a word with three occurrences of this letter.

Using the best Twitter preprocessing setting (without the condition that a given term has to appear at least twice in the entire corpus, since we were processing a single tweet in this experiment) a tweet *"@jenny I am with my Sisterrrrrrr and we are buying $aapl stocks #happy !"* would be preprocessed into : *"atttjenny i am with my sisterrr and we are buying stockaapl stocks hashhappy !"*' and the resulting features would be <atttjenni, atttjenni i, i, i am, am, am with, with, with my, my, my sisterrr, sisterrr, sisterrr and, and, and we, we, we are, are, are buy, buy, buy stockaapl, stockaapl, stockaapl stock, stock, stock hashhappi, hashhappi, hashhappi !, !>. It can be observed that the features are a combination of unigrams and bigrams. For example <atttjenni, i, am, with, my> are unigrams represented by a single stemmed word, while <atttjenni i, i am, am with, with my, my sisterrr> are bigram examples made by concatenating two stemmed words.

The best tweet preprocessing setting resulted in construction of 1,198,302 features that were used for sentiment classifier training from smiley-labeled tweets and achieved the accuracy of 80.22% on the test dataset, which contained 177 negative and 182 positive hand-labeled tweets. This result is comparable with the results of Go et al. (2009) who used the same test set and employed several classification algorithms. Their best reported accuracy was 83% while specifically for the SVM classifier they reported 82.2% (unigrams), 78.8% (bigrams), 81.6% (unigrams + bigrams), and 81.9% (unigrams + POS)

[4]We used the Regex tokenizer from the LATINO library, which is based on regular expressions.
[5]The LATINO library uses Snowball word stemmer.

Table 3.3: SVM classifier performance for various preprocessing settings measured by applying the stratified ten-fold cross-validation method on 1,600,000 smiley-labeled tweets. Applied settings are marked with the "X" sign.

ID	Web links	Hashtags	Ex. and q. marks	Letter repet.	Neg.	Stop words	Avg. accuracy ± std. dev.	Avg. F-measure ± std. dev.
1		X		X			81.23% ± 0.16%	0.8143 ± 0.0046
2	X		X				81.07% ± 0.33%	0.8127 ± 0.0049
3			X		X		81.09% ± 0.20%	0.8125 ± 0.0067
4	X						81.26% ± 0.21%	0.8123 ± 0.0047
5					X		81.18% ± 0.21%	0.8121 ± 0.0041
6	X			X			81.24% ± 0.19%	0.8121 ± 0.0047
7	X	X	X	X			81.25% ± 0.31%	0.8116 ± 0.0073
8	X	X		X	X		81.23% ± 0.33%	0.8113 ± 0.0064
9		X	X	X	X		81.16% ± 0.24%	0.8110 ± 0.0080
10	X			X	X		81.24% ± 0.22%	0.8110 ± 0.0049
11	X		X	X			81.12% ± 0.20%	0.8109 ± 0.0046
12	X	X	X	X	X		81.15% ± 0.20%	0.8109 ± 0.0042
13	X				X		81.08% ± 0.20%	0.8109 ± 0.0060
14	X	X	X		X		81.19% ± 0.19%	0.8108 ± 0.0044
15	X		X	X	X		81.21% ± 0.26%	0.8106 ± 0.0065
16		X	X	X			81.14% ± 0.20%	0.8105 ± 0.0065
17		X	X		X		81.19% ± 0.16%	0.8104 ± 0.0048
18		X					81.04% ± 0.30%	0.8103 ± 0.0077
19							81.13% ± 0.24%	0.8100 ± 0.0056
20		X	X				81.13% ± 0.15%	0.8099 ± 0.0048
21				X			81.15% ± 0.32%	0.8099 ± 0.0078
22			X	X	X		81.12% ± 0.34%	0.8096 ± 0.0080
23	X	X	X				81.04% ± 0.20%	0.8093 ± 0.0064
24			X				81.15% ± 0.25%	0.8089 ± 0.0062
25				X	X		81.10% ± 0.20%	0.8086 ± 0.0057
26	X	X		X			81.15% ± 0.17%	0.8086 ± 0.0047
27	X		X		X		81.09% ± 0.28%	0.8086 ± 0.0054
28		X		X	X		81.08% ± 0.20%	0.8079 ± 0.0051
29	X	X			X		81.10% ± 0.24%	0.8077 ± 0.0062
30			X	X			81.26% ± 0.22%	0.8076 ± 0.0033
31	X	X					81.10% ± 0.21%	0.8074 ± 0.0061
32		X			X		81.09% ± 0.30%	0.8072 ± 0.0060
33	X	X		X	X	X	79.16% ± 0.23%	0.7962 ± 0.0050
34		X			X	X	79.11% ± 0.24%	0.7942 ± 0.0045
35	X		X		X	X	79.22% ± 0.20%	0.7936 ± 0.0057
36		X	X	X	X	X	79.20% ± 0.18%	0.7931 ± 0.0067
37	X		X	X	X	X	79.22% ± 0.18%	0.7930 ± 0.0066
38	X	X	X	X	X	X	79.21% ± 0.24%	0.7930 ± 0.0072
39	X	X	X		X	X	79.21% ± 0.21%	0.7926 ± 0.0054
40				X	X	X	79.26% ± 0.23%	0.7926 ± 0.0061
41		X	X		X	X	79.20% ± 0.20%	0.7925 ± 0.0066
42			X	X	X	X	79.23% ± 0.24%	0.7923 ± 0.0038
43		X		X	X	X	79.23% ± 0.23%	0.7923 ± 0.0044
44	X			X	X	X	79.18% ± 0.22%	0.7905 ± 0.0054
45	X	X			X	X	79.21% ± 0.19%	0.7904 ± 0.0044
46			X		X	X	79.21% ± 0.21%	0.7903 ± 0.0058
47					X	X	79.17% ± 0.24%	0.7902 ± 0.0059
48	X				X	X	79.18% ± 0.20%	0.7902 ± 0.0056
49	X	X				X	78.48% ± 0.21%	0.7900 ± 0.0060
50	X					X	78.45% ± 0.22%	0.7882 ± 0.0061
51	X			X		X	78.58% ± 0.21%	0.7881 ± 0.0067
52		X		X		X	78.53% ± 0.19%	0.7878 ± 0.0066
53			X	X		X	78.62% ± 0.25%	0.7878 ± 0.0052
54	X		X	X		X	78.63% ± 0.15%	0.7877 ± 0.0060
55	X		X			X	78.53% ± 0.20%	0.7870 ± 0.0057
56		X				X	78.54% ± 0.14%	0.7868 ± 0.0052
57		X	X	X		X	78.60% ± 0.25%	0.7867 ± 0.0047
58				X		X	78.52% ± 0.17%	0.7865 ± 0.0074
59						X	78.45% ± 0.24%	0.7863 ± 0.0060
60	X	X	X	X		X	78.49% ± 0.16%	0.7860 ± 0.0053
61	X	X		X		X	78.50% ± 0.21%	0.7860 ± 0.0066
62	X	X	X			X	78.55% ± 0.21%	0.7858 ± 0.0070
63		X	X			X	78.49% ± 0.15%	0.7855 ± 0.0068
64			X			X	78.43% ± 0.20%	0.7823 ± 0.0063

accuracies for different settings. Moreover, our result is also comparable with the general accuracy of human annotators. Namely, in a process of manually labeling texts based on its sentiment, human annotators usually agree 79% of the time (Ogneva, 2010). Therefore, a sentiment classifier which achieves performance accuracy of around 80% is comparable to the performance of human annotators.

From Table 3.3 several additional observations can be made. For example, the results indicate that it is beneficial not to remove stop words[6] from tweets, since all the settings which do not apply this preprocessing option are placed in the upper part of the table. In (Saif, He, & Alani, 2012, 2014) the authors also observed that removing pre-compiled stop words decreases accuracy in the sentiment classification setting. However, if one decides to remove stop words after all, our results showed that it is beneficial to employ also the replacement of the negation words with a unique token *NEGATION*. This is due to the fact that the used stop word list contains the negation words, and with the replacement of the negation words with the token, the information about it is preserved in a tweet, even though the stop words are removed. Regarding the rest of the preprocessing settings, one cannot draw a general conclusion, since these preprocessing options are dispersed across the table. Nevertheless, the Setting 1 shown in the first row of Table 3.3 performed best, and, therefore, we used it in the rest of our studies and analyses.

In the preprocessing experiments the original dataset was pre-filtered and did not contain tweets with both positive and negative emoticons, thus the reported results in Table 3.3 may be somewhat overoptimistic (i.e., if the data were not pre-filtered and contained also tweets with mixed emoticons, the performance results would probably be somewhat lower. Unfortunately, we did not have an access to unfiltered data to test this hypothesis). Nevertheless, even if the reported results are overoptimistic, this property of the dataset does not affect the general conclusions concerning the choice of preprocessing settings, given that in all the settings assessed the dataset was preprocessed in the same way.

3.3.3 Comparison with Publicly Available Sentiment Classifiers

There are some sentiment analysis tools that are publicly available on the Internet. This section presents the evaluation of several free versions of them on 177 negative and 182 positive hand-labeled tweets. We discuss their characteristics and compare their performance results with the results of our sentiment classifier on the same test dataset.[7]

AlchemyAPI[8] is a service, which performs sentiment analysis, keyword extraction, entity extraction, image extraction and tagging, etc. It employs natural language processing techniques and machine learning algorithms. Concerning sentiment analysis, this service is able to perform document-level, user-specified target, entity-level, and keyword-level sentiment analysis from a public webpage, HTML, or text document. It can perform sentiment analysis for a text written in English or German. To test this service, we used their free starter API, which offers all service's functions, but it is limited to 1,000 transactions per day and 5 concurrent requests. We used the Text API for sentiment analysis and classified tweets from the test set. The API returns, among other information, a polarity result of sentiment classification in the form "positive", "negative," or "neutral", and a

[6]The LATINO library uses the Snowball English stop word list, which is available at http://snowball.tartarus.org/algorithms/english/stop.txt.

[7]Since the tweets in the test dataset were already HTML encoded, we decoded them in order to give the original texts of the tweets to the tested tools. For example, '&' is transformed to the '&' sign. We tested several public tools also without the HTML decoding and differences in the performance were negligible.

[8]http://www.alchemyapi.com/.

numerical sentiment strength value. We classified the tweets from the test dataset and achieved the accuracy of 83.57%.

Repustate[9] provides sentiment analysis and text analytics for several languages, i.e., English, French, Spanish, Italian, German, Chinese, and Arabic. Sentiment analysis can be performed at the document, topic, or sentence level from a given text, an URL, or a list of URLs. The free usage enables 1,000 API calls per month. API response provides a numerical score of the sentiment classification, where a positive number represents positive sentiment and a negative number represents negative sentiment. In the documentation it is written that a score close to zero can be interpreted as the neutral sentiment. In our experiments, we interpreted exact zero score as the neutral sentiment. To test this service we used version 2 of the API and classified the tweets from the test dataset. In this experiment, we achieved the accuracy of 66.57%.

Text-processing[10] is a free and open public service for text mining and natural language processing. It provides sentiment analysis, stemming, lemmatization, POS tagging, chunking, phrase extraction, and named entity recognition. The public API is intended for non-commercial usage, and it is limited to 1,000 calls per day per IP (Internet Protocol) address. For higher limits or commercial usage, one should use the Mashape Text-Processing API.[11] We used the free public Sentiment API, which returns the sentiment label of an analyzed text in the form "pos", "neg," or "neutral". It also returns probabilities for each of the sentiment labels. Using hierarchical classification, the algorithm first determines whether the analyzed text is neutral. If it is not neutral, it continues to the next step to determine the positive or negative sentiment polarity of the text. The sentiment API can perform sentiment analysis for text written in English, Dutch, or French. We classified the tweets from the test dataset using this service and used only the information about the final sentiment label. We achieved the accuracy of 61.00%.

Sentiment140[12] is a sentiment analysis tool by Go et al. (2009). The service can perform sentiment analysis for text written in English or Spanish. We used their Simple Classification Service API and classified tweets from the test dataset using the HTTP GET requests. The API response returns sentiment polarity values in the form: "0" for negative, "2" for neutral, and "4" for the positive sentiment. On the test set we achieved the accuracy of 45.96%.

Finally, SentiStrength[13] determines the strength of the positive (on the scale from 1 to 5) and negative (on the scale from -1 do -5) sentiment in short texts. The SentiStrength algorithm uses a sentiment word strength dictionary and handles a range of non-standard spellings and approaches for expressing sentiment in text (Thelwall, Buckley, Paltoglou, Cai, & Kappas, 2010). Initially, it was implemented for English texts and optimized for short social Web contents, but it can be used also for a number of other languages, i.e., Finnish, German, Dutch, Spanish, Russian, Portuguese, French, Arabic, Polish, Persian, Swedish, Greek, Welsh, Italian, or Turkish. We registered and obtained the free version of SentiStrength in order to classify tweets from the test dataset. If the output sentiment strengths for the positive and negative class were the same, we classified the corresponding tweet as neutral. If the strengths were not equal, the tweet was classified as positive or negative, depending on which class got the higher score. We achieved the accuracy of 69.92% on the test set.

Summary of the evaluation performance of the tested publicly available sentiment analysis tools is given in Table 3.4. As can be seen from the table, the best result (the accuracy

[9] https://www.repustate.com/.
[10] http://text-processing.com/.
[11] https://www.mashape.com/japerk/text-processing/pricing#!.
[12] http://www.sentiment140.com/.
[13] http://sentistrength.wlv.ac.uk/.

Table 3.4: Evaluation performance (accuracy) on the test set for the developed sentiment classifier and several publicly available sentiment analysis tools.

Sentiment tool	Accuracy on the test set
Our approach	80.22%
AlchemyAPI	83.57%
Repustate	66.57%
Text-processing	61.00%
Sentiment140	45.96%
SentiStrength	69.92%

of 83.57%) was achieved with the AlchemyAPI service. The second best sentiment classifier is ours, while all the other classifiers achieved much lower performance. Therefore, the results indicate that our sentiment classifier is better than most of the tested public classifiers on the hand-labeled tweets test dataset. Moreover, given that its performance is not substantially lower than the best publicly available sentiment classifier, and given that we can employ it for classification of much larger sets of tweets than using the standard free version of the Alchemy tool,[14] the choice of using the developed SVM classifier with the selected tweet preprocessing setting was a reasonable choice made for further experimentation, as presented in the next chapters of the thesis.

3.4 Methodology and Results Summary

In this chapter, we presented the experiments based on which we determined the most suitable approach for Twitter sentiment analysis in terms of the algorithm for sentiment classification and the text preprocessing setting.

Regarding the algorithm for sentiment classification, we conclude that the SVM algorithm is the best choice for Twitter sentiment analysis. The selection is based on its nature which suits the needs for the Twitter sentiment analysis in data streams and on the empirical evaluation results which show its superiority over several other algorithms.

The proposed Twitter-specific and standard preprocessing steps are presented in Figure 3.1. The steps are presented in order of their execution. The experiments show that the best preprocessing setting employs usernames, stock symbols, and hashtags transformations, replaces repetitive letters with more than three occurrences in a word by a word with three occurrences of this letter, performs tokenization and stemming, constructs unigrams and bigrams, uses terms which appear at least two times in the corpus, and it is TF-based. After the preprocessing and feature construction, the resulting features and their values are used for training the SVM sentiment classifier. The same preprocessing is used also for the tweets which are classified with the developed SVM sentiment classifier.

By performing the preprocessing experiments, we also tested the hypothesis which states that with the appropriate selection of preprocessing steps the classification accuracy can be improved. We set this hypothesis in the Introduction (see Section 1.4) of this dissertation. By observing Table 3.3 we can accept this hypothesis, as the results from the table indicate that with the appropriate selection of the preprocessing options one can influence and improve the classification accuracy of the sentiment classification algorithm. From the table it follows that the accuracy and the F-measure values for different

[14] As already mentioned, the standard free version of the AlchemyAPI service offers 1,000 API calls per day. Nevertheless, the service also offers 30,000 API calls per day for approved academic users, but for all of our experiments and analyses we would prefer even larger number of API calls.

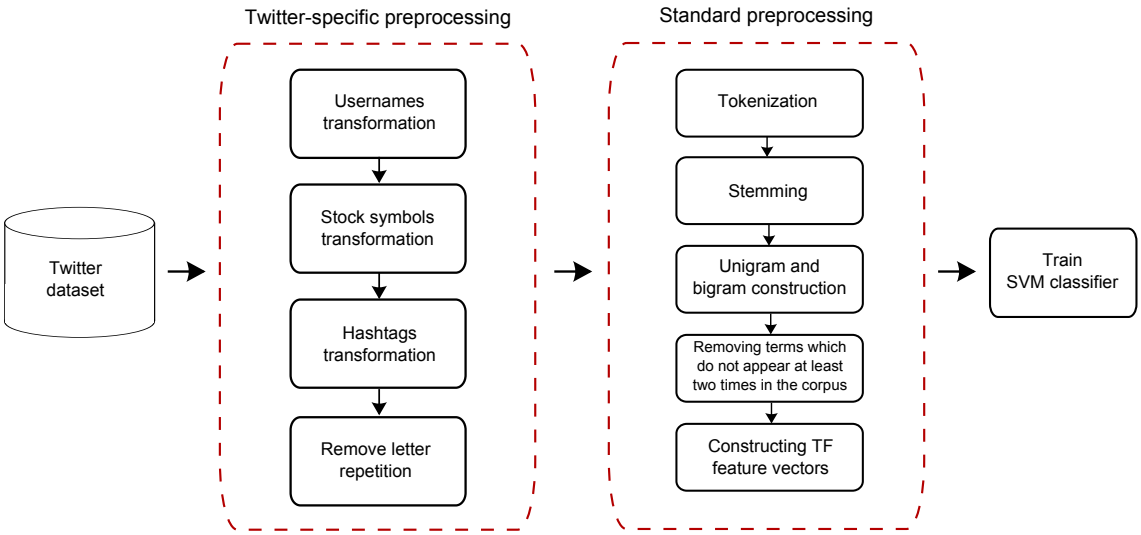

Figure 3.1: Methodological steps for Twitter-specific and standard preprocessing for Twitter microblogging posts.

preprocessing settings differ in terms of even several percents, depending on the applied preprocessing setting.

Chapter 4

Static Predictive Twitter Sentiment Analysis

This chapter is concerned with finding a relationship between the sentiment expressed in Twitter posts discussing finances and the future stock market prices. The motivation for this study lies, on the one hand, in observations that emotions are crucial to social behavior (Damasio, 1995) and that the stock market itself can be considered as a measure of social mood (Nofsinger, 2005), and on the other hand, in earlier studies which showed that sentiment expressed in social media could contain predictive information about future stock market assets (see Section 2.1.1). Since our research is based on Twitter data, in this chapter we investigate whether sentiment analysis on Twitter posts can provide predictive information about the value of future stock closing prices. The experiments in this chapter are based on our previously published work (Smailović et al., 2012, 2013, 2014), while the relative neutral zone is introduced here for the first time.

4.1 Methodology

In order to perform sentiment analysis on Twitter data, we employed a supervised machine learning approach to train a sentiment classifier, using the Support Vector Machine (SVM) algorithm. For training the sentiment classifier we used a collection of 1,600,000 smiley-labeled tweets (Go et al., 2009) which was preprocessed as explained in Section 3.3.2. In addition to that, we collected a dataset of financial tweets, discussing various companies. The sentiment classifier was trained on the first dataset and applied to the latter.

By applying the best setting for tweet preprocessing on the gathered financial tweets, two sets of experiments were performed. In the first set of experiments, the financial tweets were classified into two sentiment categories, positive or negative. In the second set of experiments, the SVM classification approach was advanced by enabling it to identify also the neutral tweets (not clearly expressing positive or negative sentiment), by employing the "neutral zone" concept: (i) the basic definition, and (ii) the improved and better formalized definition of the neutral zone.

Finally, given the adequately prepared time series data of sentiment and the stock prices, a statistical test was performed to test whether tweet sentiment is useful for forecasting the movement of prices on the stock market and how significant the results are.

4.1.1 The Neutral Zone

In this section, we present the concept of the neutral zone, which enables identification of tweets which do not fall into the positive or negative category. The motivation for this

concept lies in the fact that classifying tweets into two sentiment categories, positive or negative, is sometimes unrealistic, since a tweet can be objective and without any opinion (i.e., without an expressed sentiment). Considering this, a tweet should also have the possibility of being classified as neutral or weakly opinionated. Therefore, in this section, we present a mechanism for classifying tweets into the positive, negative, and neutral categories.

We present two ways of identifying non-opinionated tweets: (i) the basic one, where we define the fixed neutral zone for classifying neutral tweets, and (ii) the improved one, which employs the relative neutral zone, which is more intuitive and better formalized.

4.1.1.1 Fixed Neutral Zone

The training smiley-labeled dataset (Go et al., 2009) does not contain any neutral tweets for the classifier to learn from. Therefore, in the classification phase, we label a tweet as neutral, if it is projected into an area close to the SVM model's hyperplane. We define this area as the neutral zone, which is parameterized by value t, where t represents the positive and $-t$ the negative border of the neutral zone. If a tweet x, with a distance from the SVM hyperplane $d(x)$, is projected into this zone, i.e., $-t < d(x) < t$, then instead of being classified into the positive or negative category, it is classified as neutral. With a larger size of the neutral zone, the classifier can be more confident in its classification decision for a classified tweet being positive or negative.

Note that this "neutral zone" does not denote only the "neutral tweets", such as tweets which would be labeled as neutral by a human annotator. Instead, the neutral zone contains also the tweets which are either positive or negative but close to the SVM hyperplane, which separates the positives from the negatives. Thus, the neutral zone includes tweets containing mixed sentiments, weakly opinionated positive/negative tweets, as well as tweets containing terms which were not observed during the training phase (if human annotated neutral tweets were available, they would have been included in the neutral zone as well). From now on, we refer to all these kinds of tweets as being neutral.

4.1.1.2 Relative Neutral Zone

The fixed neutral zone definition is rather simple, but allows for fast computation. Nevertheless, we further improved it in order to be more intuitive, better formalized, and to be theoretically grounded, which resulted in the concept of a relative neutral zone. The idea of the relative neutral zone is based on geometric interpretation of the classification scores given by linear classifiers, and calculating probabilities from these values (Flach, 2012).

The idea is to calculate the classification reliability of an example to be classified based on its distance from the SVM hyperplane. The example is classified as being neutral if the classification reliability is below a given threshold. Otherwise, the example is classified as positive or negative, depending on the side of the SVM hyperplane in which it is projected.

Let d_A be the average distance of training examples (positive or negative) from the SVM hyperplane. Let d be the distance from the SVM hyperplane of an example to be classified. The reliability R of the classification is calculated as:

$$R = \begin{cases} \frac{d}{2 \times d_A} & \text{if } d < 2 \times d_A \\ 1 & \text{if } d \geq 2 \times d_A \end{cases} \tag{4.1}$$

When the classified example is more than two average distances away from the SVM hyperplane, the reliability of classification is $R = 1$. Examples which lie on the hyperplane have classification reliability $R = 0$. Therefore, reliability ranges between 0 and 1, where the higher values correspond to more confident classifications.

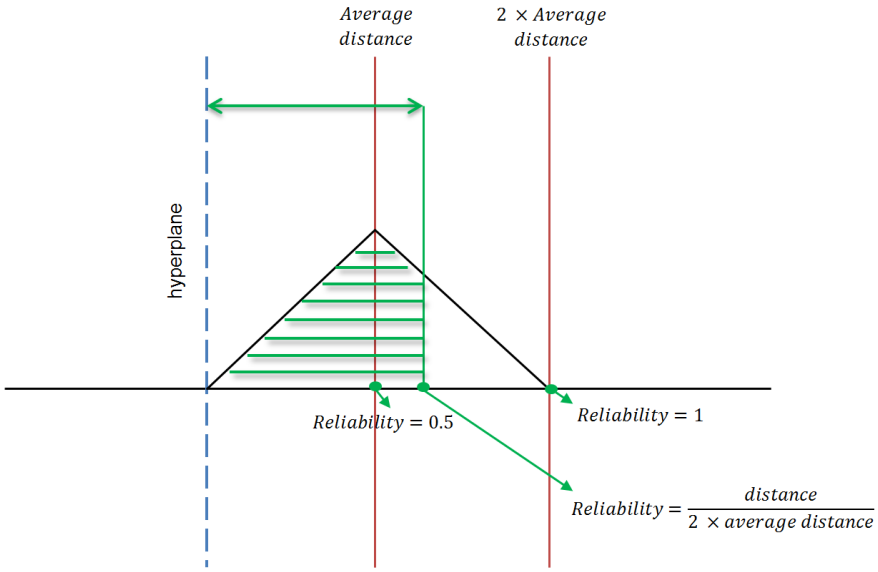

Figure 4.1: Reliability as a function of the distance from the SVM hyperplane.

The definition of the relative neutral zone is based on the assumption that for the whole population of positive examples, distances from the SVM hyperplane are normally distributed around the average distance of the positive training examples (Flach, 2012). Also, the distances of negative examples are assumed to be normally distributed around the average distance of the negative training examples (Flach, 2012). For simplification, the normal distribution is approximated with a triangular one, as illustrated in Figure 4.1.

4.2 Experimental Setting

For the purpose of static predictive Twitter sentiment analysis, we collected a number of financial tweets, discussing several companies. Moreover, we collected also the stock closing prices of these companies for the corresponding time period. The time series data of tweet sentiments and stock closing prices were adequately prepared and a statistical test for determining the correlation between these two time series was applied. This section discusses the financial data and the statistical test.

4.2.1 Financial Dataset

A tweet dataset and stock closing prices of several companies were collected for our experiments. On the one hand, we collected 152,570 tweets written in English language discussing relevant stock information concerning eight companies (Apple, Amazon, Baidu, Cisco, Google, Microsoft, Netflix, and Research In Motion Limited)[1] in the nine-month time period from March 11 to December 9, 2011. On the other hand, we collected stock closing prices of these companies for the same time period.

The data source for collecting financial Twitter posts was the Twitter API[2] (i.e., the Twitter Search API), which returns tweets that match a specified query. By informal

[1]Tweet IDs of our datasets are available on: http://streammining.ijs.si/TwitterStockSentimentDataset/TwitterStockSentimentDataset.zip.

[2]https://dev.twitter.com/.

Twitter conventions, the dollar-sign notation is used for discussing stocks of companies. For example, the $BIDU tag indicates that the user discusses stocks of the company Baidu. This convention was used for the retrieval of financial tweets.[3] The stock closing prices of the companies for each day were obtained from the *Yahoo! Finance* website.

The time of tweets in our dataset is presented in UTC (Coordinated Universal Time) since the Twitter API stores and returns dates and times in UTC. On the other hand, Baidu is included in the NASDAQ-100 index, and this stock exchange works in the EST (Eastern Standard Time)/EDT (Eastern Daylight Time) timezone which is four to five hours behind UTC. Therefore, compared to EST/EDT, there is an additional shift of four to five hours; thus, there is even a larger time lag between the tweets of a previous day and the stock market activity and closing prices of the current day.

In the entire study, we focused on the analysis of financial tweets on the stocks of Chinese web search engine provider, Baidu,[4] in order to investigate relationships between the observed sentiments in the stock-related tweets and the corresponding stock price movements. The data of this Chinese web search engine provider was chosen for hand-labeling since the set of tweets related to Baidu was of a manageable size given the resources available (we collected and labeled approximately 11,000 tweets, compared to, for example, approximately 40,000 tweets that we collected for the Apple company). The collection of Baidu tweets was manually labeled by two annotators, each labeling every second tweet ordered by time. This hand-labeling effort took over three months to ensure good quality of the labeled data.

We were interested in manual labeling of the tweets from the point of view of a particular company and not mainly on the sentiment-carrying words used. The reason for this decision was that our long-term intention was to construct classifiers that should distinguish between sentiments of tweets of different companies; hence, a company-focused view was a necessity. The labels were given to instances according to their financial sentiment; that is, their impact on the perception of the company, its products, or its stock. For example, a tweet: "*I just love shorting CompanyX. What a nice day of profits, first of many...*" would be labeled as negative, since shorting means betting that the value of the stock will drop. Despite containing many positive sentiment words, such a tweet would be providing a message of a negative financial prospect for CompanyX. Another issue was that in the dataset there are many tweets that do not discuss Baidu stocks, although they do contain the $BIDU tag. These tweets may actually express an opinion about another company or issue. For example, a tweet like "*Apple is great $BIDU*" reflects a positive tweet sentiment, but does not discuss the Baidu company at all. Again, these kinds of tweets were labeled from the point of view of the Baidu company, and not mainly on the sentiment-carrying words used. Therefore, the mentioned tweet would be labeled as neutral.

Therefore, in Baidu sentiment labeling, the annotator was instructed to focus on the following question:

- "What would someone who knows what Baidu is and shares in general, think of Baidu and its shares after he read this tweet?", or in other words,
- "Is this tweet positive, negative, or neutral concerning Baidu and/or the price of its shares?"

[3]To deal with spam (writing nearly identical messages from different accounts), the algorithm based on the work of Broder, Glassman, Manasse, and Zweig (1997) was employed to discard tweets that were detected as near duplicates. This was done as part of the study of our research group at the Department of Knowledge Technologies of the Jožef Stefan Institute, and it is presented in (Saveski & Grčar, 2011).

[4]www.baidu.com.

The resulting hand-labeled dataset consists of 11,389 Baidu financial tweets (4,861 positive, 1,856 negative, and 4,672 neutral).[5] In this dataset, neutral tweets are those that contain no sentiment about Baidu, contain both positive and negative sentiments about Baidu, as well as those that do not discuss Baidu even if they are positively or negatively oriented (as discussed above).

4.2.2 Correlation Between Tweet Sentiment and Stock Closing Price

Given the time series of tweet sentiments as classified by our classifier and the time series of stock closing prices, the question addressed is whether one time series is useful in forecasting another. We applied a statistical test to determine whether sentiments expressed in tweets contain predictive information about the future values of stock closing prices. To this end, we employed the Granger causality analysis test (Granger, 1969), which is a statistical hypothesis test used for discovering whether one time series is effective for forecasting another time series. Since we have the tweet time series on the one hand and the stock closing price time series on the other hand, this test suits our needs to check whether there is a predictive relationship between sentiments in tweets and stock closing prices. Granger causality analysis has been a popular method for revealing dependences between time series and has been successfully applied in various domains (Y. Liu & Bahadori, 2014).

If time series X is said to Granger-cause time series Y, then the information in past values of X helps predict values of Y better than only the information in past values of Y (Seth, 2007). Therefore, time series X contains unique and statistically significant information about the future values of time series Y. Granger causality analysis is usually done by employing statistical tests on lagged values of X combined also with lagged values of Y. The Granger causality test expects that the time series is stationary and that it can be represented by a linear model. Complex implementations for nonlinear scenarios exist; nevertheless, they are often more challenging to apply in practice (Seth, 2007).

The output of the Granger causality test is the p-value, which takes values in the [0,1] interval. In statistical hypothesis testing, the p-value is a measure of how much evidence we have against the null hypothesis. In the context of Granger causality, the null hypothesis states that time series X does not Granger-cause time series Y. If the p-value is lower than the selected significance level, for example 5% ($p < 0.05$), the null hypothesis is rejected and the result can be considered statistically significant. On the other hand, a large p-value represents weak evidence against the null hypothesis; thus, the null hypothesis cannot be rejected. In our experiments, we used the Granger causality test implementation from Free Statistics Software (Wessa, 2013).

We applied the Granger causality analysis on Baidu tweet sentiment time series data, on the one hand, and the corresponding stock closing prices time series data, on the other hand. For representation of the tweet sentiment we propose a sentiment indicator for predictive tweet sentiment analysis in finance, named *positive sentiment probability*: p_{sp} (Smailović et al., 2013). Positive sentiment probability is computed for a day d of a time series by dividing the number of positive tweets N_{pos} by the number of all tweets on that day N_t.

$$p_{sp}(d) = \frac{N_{pos}(d)}{N_t(d)} \tag{4.2}$$

This ratio is used to estimate the probability that the sentiment of a randomly selected tweet on a given day is positive.

[5]The Baidu tweet IDs and manual labels are publicly available on: http://streammining.ijs.si/TwitterStockSentimentDataset/LabeledTwitterDataset.zip, file BIDU.txt.

To test whether one time series is useful in forecasting another, using the Granger causality test, we first calculated positive sentiment probability for each day when the stock market was open. We then calculated two ratios[6] to meet the Granger causality test condition that the time series data needs to be stationary:

- Daily change of the positive sentiment probability D_{sent}: *positive sentiment probability today – positive sentiment probability yesterday.*

$$D_{sent}(d) = p_{sp}(d) - p_{sp}(d-1) \tag{4.3}$$

- Daily return in stock closing price D_{price}: *(closing price today – closing price yesterday)/closing price yesterday.*[7]

$$D_{price}(d) = \frac{price(d) - price(d-1)}{price(d-1)} \tag{4.4}$$

We applied the Granger causality test on D_{sent} and D_{price} time series to test the following null hypothesis: "sentiment in tweets does not predict stock closing prices". If this hypothesis is rejected, it means that the sentiment in tweets Granger-causes the values of stock closing prices.

We performed statistical tests on the entire nine-month time period of Baidu data (from March 11 to December 9, 2011), as well as on individual three-month time periods (corresponding approximately to March to May, June to August, and September to November). In Granger causality testing, we considered lagged values of time series for one, two, and three days.

4.3 Experimental Results

In this section, we present the experimental results of Granger causality testing applied to two adequately transformed time series data: sentiment in tweets and stock closing prices. The Granger causality test was employed in order to determine whether there is predictive information between sentiments expressed in tweets discussing a company and values of stock closing prices of the same company.

Two sets of experiments were performed. In the first set of experiments, the financial tweets were classified into two sentiment categories: positive or negative, while in the second set of experiments, the approach was advanced by enabling it to identify also the neutral tweets, by employing the concept of the neutral zone. We experimented with the two definitions of the neutral zone (see Sections 4.1.1.1 and 4.1.1.2) and observed how they affected the results. The text of the financial tweets was preprocessed as explained in Section 3.3.2.

4.3.1 Two-class Sentiment Classification

In the first experiment, the Baidu financial tweets were classified only as positive or negative with the developed SVM sentiment classifier. We applied the data transformation as described in Section 4.2.2 (counted the number of positive, negative, and neutral tweets, calculated positive sentiment probability, calculated daily changes of the positive sentiment probability, and the daily return of the stocks' closing price) and performed the

[6]The ratios were defined in collaboration with the domain experts from the Stuttgart Stock Exchange.

[7]The same transformation of the price time series was also used in (Ruiz, Hristidis, Castillo, Gionis, & Jaimes, 2012).

Table 4.1: Statistical significance (p-values) of Granger causality correlation between positive sentiment probability and closing stock price for Baidu, while changing the size of the fixed neutral zone (i.e., the t value) from 0 to 1. Values which are lower than the p-value of 0.1, after applying the Bonferroni correction, are marked in bold.

Size of the neutral zone		0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
Time period	Lag											
9 months	1	0.587	0.762	0.604	0.797	0.758	0.877	0.650	0.471	0.388	0.743	0.683
Mar.-May	1	0.824	0.698	0.780	0.676	0.645	0.733	0.502	0.497	0.777	0.808	0.828
June-Aug.	1	0.995	0.865	0.812	0.863	0.920	0.347	0.514	0.622	0.585	0.347	0.368
Sept.-Nov.	1	0.298	0.452	0.555	0.576	0.620	0.540	0.335	0.216	0.140	0.288	0.251
9 months	2	0.594	0.618	0.441	0.347	0.312	0.300	0.453	0.225	0.066	0.063	**0.019**
Mar.-May	2	0.624	0.699	0.766	0.822	0.785	0.894	0.717	0.755	0.949	0.943	0.735
June-Aug.	2	0.993	0.963	0.974	0.859	0.605	0.347	0.574	0.342	0.178	0.151	0.067
Sept.-Nov.	2	**0.017**	**0.020**	**0.020**	0.037	0.054	0.068	0.082	0.039	**0.020**	0.025	**0.021**
9 months	3	0.795	0.813	0.705	0.599	0.575	0.522	0.733	0.459	0.165	0.166	0.051
Mar.-May	3	0.311	0.379	0.509	0.728	0.705	0.705	0.777	0.767	0.924	0.938	0.836
June-Aug.	3	0.915	0.684	0.924	0.791	0.574	0.309	0.456	0.405	0.255	0.195	0.106
Sept.-Nov.	3	0.026	0.035	0.039	0.075	0.080	0.095	0.160	0.077	**0.024**	0.034	**0.022**

Granger analysis test. Results (i.e., p-values which were obtained as output from the Granger causality test) are shown in Table 4.1, in the column where the size of the neutral zone is 0.

Since in the Granger causality experiments we computed the p-value repetitively and performed multiple comparisons of p-values for different experimental settings, there was a possibility of making a "Type I" error, i.e., to reject a true null hypothesis by chance. Therefore, we used the Bonferroni correction (Abdi, 2007) to neutralize the problem of multiple comparisons. This correction is considered very conservative. It makes adjustments to a critical p-value by dividing it by the number of tests being made. In our case, we divided the critical p-value of 0.1 by 4, as this was the number of time periods (whole nine months and three three-month periods), which we considered to be a family of tests, where a family of tests represents a series of tests executed on a dataset (Abdi, 2007). We compared the p-values, which came from the Granger causality test with 0.1/4=0.025 and rejected the null hypothesis if the value was lower than 0.025. After applying the Bonferroni correction, the results of Granger analysis indicated that the best correlation was obtained for the September-November time period for the 2 days time lag (see Table 4.1 and the column where the size of the neutral zone is 0), meaning that for this specific time period and time lag there is evidence that sentiment in tweets discussing the Baidu company Granger-causes Baidu stock closing prices.

4.3.2 Three-class Sentiment Classification

In the previous experiment, we classified financial tweets into one of the two sentiment categories, positive or negative, and therefore assumed that every tweet contained an opinion. In the following experiments, we took into account the concept of the neutral zone, in order to classify tweets into three sentiment categories: positive, negative, or neutral. Our aim was to investigate whether the introduction of the neutral zone would improve the predictive capabilities of tweets. First, we experimented with the simple

version of the neutral zone, i.e., the fixed neutral zone (see Section 4.1.1.1), and then, in the second experiment, we applied the improved neutral zone, i.e., the relative neutral zone (see Section 4.1.1.2).

In the first experiment, we employed the fixed neutral zone and, therefore, a tweet was classified as neutral if its distance $d(x)$ from the SVM hyperplane was in the boundaries of the neutral zone, that is, $-t < d(x) < t$. We applied the same transformation of data as before (see Section 4.2.2) and performed the Granger analysis test on tweet sentiment time series data and the corresponding stock closing prices time series data. We varied the size of the neutral zone (i.e., the t value) from 0 to 1 (where t=0 corresponds to classification without the neutral zone) and calculated the p-value for the separate day lags (1, 2, and 3). The results are shown in Table 4.1. The column where the size of the neutral zone is 0, represents the results of the Granger analysis test without the neutral zone, where financial tweets were classified into one of the two categories, positive or negative. All the remaining columns contain p-values for various sizes of the neutral zone. Values which are lower than the p-value of 0.1, after applying the Bonferroni correction, are marked in bold. The highest number of significant values was obtained with t values of 0.8 and 1 for the border distance of the neutral zone from the SVM hyperplane and for the September-November time period. In the same time period significant results were obtained also in the two-class setting without the neutral zone (see Section 4.3.1). This may be due to more active discussions on Twitter given high volatility of the stock price in this period. The results in Table 4.1 therefore indicate that by introducing the neutral zone the predictive power (in terms of Granger causality testing) of the sentiment classifier was improved.

In Appendix B, we report the results of the Granger causality correlation between the values of daily changes of positive sentiment probability and daily returns of the stock closing prices, by using the fixed neutral zone, also for several other companies (Apple, Amazon, Cisco, Google, Microsoft, Netflix, and Research In Motion Limited), whose tweets we collected. The results show that also for several other companies, the learned classifier has the potential to be useful for stock price prediction in terms of Granger causality.

Additionally, we explored whether there is evidence for the reversed causality (that the price movements may influence the public sentiment in tweets) for the Baidu company. The results showed that there was some causality in that direction, but after making adjustments to the critical p-value by applying the Bonferroni correction, no significant results were left.

In the second experiment, using the relative neutral zone and calculation of classification reliability (see Section 4.1.1.2), we repeated our experiments on classifying Baidu financial tweets and calculating Granger causality correlation between the values of daily changes of positive sentiment probability and daily returns of the stock closing prices. In this setting, a tweet was classified as being neutral if the classification reliability was below a given threshold. As the first step, we calculated the average distances of training examples (positive and negative) from the SVM hyperplane. The average distance of positive training examples was 1.7922 and the average distance of negative training examples was -1.7069. These distances indicate that the classifier was more certain in classifying positive examples, since the absolute value of positive average distance was higher than the absolute value of the negative one. We conducted a series of experiments by varying the reliability threshold R and calculating Granger causality correlation. The reliability threshold value was varied from 0 to 1, with an increase of 0.1. As in the previous experiments, we adjusted the critical p-value by applying the Bonferroni correction. The results are shown in Table 4.2. As can be seen from the table, the relative neutral zone further improved the results, i.e., that the positive sentiment probability Granger-causes closing stock price for Baidu. The best results were obtained with the values of 0.4 and

Table 4.2: Statistical significance (p-values) of Granger causality correlation between positive sentiment probability and closing stock price for Baidu, while changing the value of reliability threshold. Values which are lower than the p-value of 0.1, after applying the Bonferroni correction, are marked in bold.

Reliability threshold		0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
Time period	Lag											
9 months	1	0.587	0.762	0.446	0.423	0.791	0.356	0.488	0.590	0.523	0.904	0.552
Mar.-May	1	0.824	0.778	0.476	0.720	0.471	0.760	0.910	0.920	0.917	0.928	0.667
June-Aug.	1	0.995	0.894	0.508	0.718	**0.009**	**0.022**	0.519	0.639	0.750	0.567	0.863
Sept.-Nov.	1	0.298	0.520	0.181	0.212	0.032	**0.007**	0.053	0.035	**0.016**	0.089	0.063
9 months	2	0.594	0.249	0.200	**0.012**	**0.007**	**0.003**	0.163	0.717	0.735	0.829	0.168
Mar.-May	2	0.624	0.788	0.733	0.642	0.298	0.406	0.566	0.681	0.996	0.517	0.218
June-Aug.	2	0.993	0.496	0.292	0.227	**0.014**	0.053	0.775	0.683	0.803	0.197	0.442
Sept.-Nov.	2	**0.017**	0.039	0.029	**0.010**	**<0.001**	**<0.001**	**0.001**	**0.023**	**0.017**	0.075	0.119
9 months	3	0.795	0.485	0.419	0.032	**0.023**	**0.008**	0.215	0.868	0.845	0.908	0.296
Mar.-May	3	0.311	0.652	0.770	0.762	0.507	0.492	0.738	0.647	0.989	0.478	0.382
June-Aug.	3	0.915	0.400	0.337	0.354	0.031	0.030	0.389	0.821	0.648	0.149	0.530
Sept.-Nov.	3	0.026	0.056	0.058	**0.004**	**<0.001**	**<0.001**	**0.004**	0.051	**0.003**	**0.012**	0.038

0.5 for the reliability threshold R. In comparison with Table 4.1, new results in Table 4.2 show more significant results. Also, it can be observed that for the same time period of September-November the p-values are lower than in the experiments with the fixed neutral zone.

In Appendix C, we report the results of the Granger causality correlation between the values of daily changes of positive sentiment probability and daily returns of the stock closing prices, by using the relative neutral zone, also for the other companies (Apple, Amazon, Cisco, Google, Microsoft, Netflix, and Research In Motion Limited), whose tweets we collected. Again, the results indicate that also for several other companies, the learned classifier and the improved relative neutral zone have the potential to be useful for stock price prediction in terms of Granger causality.

In addition, we explored whether there is evidence for the reversed causality (that the price movements may influence the public sentiment) for the Baidu company. In this experiment we obtained only one significant result for the time period June-August, 3-days lag and the value of 0.5 for the reliability threshold R.

4.3.3 Comparison of the Developed Sentiment Classifier with the Publicly Available Sentiment Classifiers in the Three-class Setting

In Section 3.3.3 we presented the performance results of experiments on comparing the developed smiley sentiment classifier with several publicly available sentiment analysis tools on the Internet. The results indicated that our sentiment classifier is better than most of the tested public classifiers. But, the testing in Section 3.3.3 was performed using the test dataset which contained only positive and negative tweets, although the publicly available sentiment tools had also the possibility to classify a given text into the neutral category. Our sentiment classifier, on the other hand, in that particular setting, had possibility to classify texts only as positive or negative.

With the introduction of the neutral zone in this chapter, the developed sentiment classifier was extended with the possibility to classify text also into the neutral category. Therefore, this section presents the experiments on testing the same publicly available

Table 4.3: Evaluation performance (accuracy) on the test set in the three-class setting for the developed sentiment classifier and several publicly available sentiment analysis tools.

Sentiment tool	Accuracy on the test set
Our approach	56.63%
AlchemyAPI	65.26%
Repustate	66.87%
Text-processing	61.65%
Sentiment140	56.83%
SentiStrength	68.67%

sentiment tools, as in Section 3.3.3, but in the three-class setting, by using the test dataset containing 177 negative, 182 positive, and 139 neutral hand-labeled tweets (Go et al., 2009), explained in Section 3.2.1. This test set is the same as in Section 3.3.3, but extended with the set of neutral tweets. Despite the fact that we did not have any neutral tweets in our learning set, we tried to detect the neutral tweets, by employing the relative neutral zone, as presented in Section 4.1.1.2. The reliability threshold set was to 0.1. Results, in terms of accuracy, for the developed sentiment classifier and selected publicly available sentiment analysis tools are presented in Table 4.3.

From Table 4.3 it follows that the best performance result in the three-class setting was achieved with the SentiStrength tool. Our sentiment classifier achieved the accuracy comparable with the performance of the Sentiment140 analysis tool created by Go et al. (2009), whose training and test set we used. This performance result is therefore reasonable and expected, particularly as we did not have any neutral tweets in our learning dataset to train on.

4.4 Methodology and Results Summary

We proposed a new methodology for determining whether sentiments in tweets that discuss a company's relevant stock information contain predictive information about the future stock closing prices of the discussed company. The methodological steps are summarized in Figure 4.2.

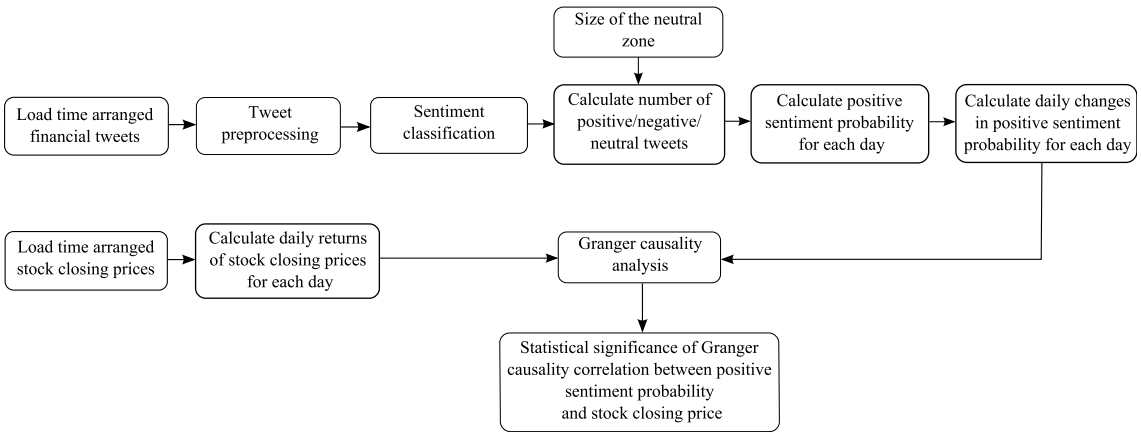

Figure 4.2: Methodological steps for predictive sentiment analysis applied to determine the correlation between tweets sentiment and stock closing prices.

As follows from Figure 4.2, one should first provide a time series collection of tweets discussing relevant stock information concerning a company of interest. Tweets are then adequately preprocessed (see Section 3.3.2). Furthermore, tweets are classified as being positive, negative, or neutral, where tweet labels depend both on the output of the sentiment classifier, and type and the parameters of the neutral zone. Moreover, for every day of a time series, positive sentiment probability is computed by dividing the number of positive tweets per day by the total number of tweets in that day. Last, daily changes of the positive sentiment probability are calculated.

On the other hand, the stock closing prices of the selected company for each day should be collected, and the daily returns in the stock closing price are to be calculated.

Given the daily changes of the positive sentiment probability time series and the daily returns in the stock closing price time series, Granger causality analysis is performed (considering lagged values of time series for one, two, and three days) to test whether tweet sentiment is useful for forecasting the movement of prices on the stock market and how significant the results are.

By performing the experiments in this chapter, we also tested two hypotheses stated in the Introduction (Section 1.4) of this dissertation. In the first one we stated that the static methodology for predictive sentiment analysis on tweet data streams is capable of predicting future financial assets. The results in Section 4.3.1, Section 4.3.2, Appendix B, and Appendix C indicate that, indeed, sentiment in Twitter messages has a potential to be useful for stock price prediction in terms of Granger causality analysis. Our experiments were performed for all the companies of the NASDAQ-100 index that have at least one tweet per day in the experimental time period, thus our claims are valid primarily for these kinds of companies and might not hold for companies with lower frequencies of tweets. Moreover, we showed that by introducing the concept of the neutral zone, the predictive power, in terms of predicting stock market assets, of the sentiment classifier was improved (see Section 4.3.2), particularly in the case of the relative neutral zone. Therefore, we can accept the other hypothesis, which states that identifying also the non-opinionated tweets improves their predictive power, in terms of forecasting stock market assets, as compared to the approach which assumes that all the tweets are opinionated and categorizes them as positive and negative only.

Chapter 5

Dynamic Predictive Twitter Sentiment Analysis

In the previous chapter, we presented a methodology for classifying financial tweets by using a static sentiment classifier, which was learned from the smiley-labeled general purpose tweets and was unchanged over time. Significant correlations between the sentiment in financial tweets and the stock closing prices were obtained, which motivated further advances. Therefore, in this chapter we focus on two goals: (i) making the classifier more domain specific to be able to better classify financial tweets by using also hand-labeled financial tweets in the training phase; and (ii) extending the approach with a capability of continuous updating of the classifier in order to adapt to sentiment vocabulary changes in a data stream. The crucial element of addressing these challenges is the use of active learning (Settles & Craven, 2008; Settles, 2009, 2011b).

The active learning methodology described in this chapter is based on our published study in Smailović et al. (2014). The methodology presented in this chapter is improved compared to the one in our published study, as we added the possibility that not only the sentiment classifier, but also the neutral zone is dynamically updated with time.

5.1 Methodology

In active learning, the learning algorithm periodically asks an oracle (e.g., a human annotator) to manually label the examples which he finds the most suitable for labeling. Using this approach and an appropriate labeling strategy, the number of examples that need to be manually labeled is largely decreased. Typically, the active learning algorithm first learns from an initially labeled collection of examples. Based on the initial model and the characteristics of the newly observed unlabeled examples, the algorithm selects new examples for manual labeling. After the labeling is finished, the model is updated and the process is repeated for the new examples. This process is repeated until some threshold (e.g., time limit, labeling quota, or target performance) is reached or, as it is the case in stream data, it continues as long as the application is active or as long as it is not terminated by the user. In our study, we are interested in applying the active learning approach for sentiment analysis on a stream of financial tweets. Since it is very difficult and costly to obtain hand-labeled datasets of tweets, especially if they are domain dependent, an active learning approach is highly suitable for such a scenario. The learning algorithm is able to interactively query the expert to obtain the manual labels as new financial tweets come from a data stream. Consequently, the sentiment classifier becomes more domain specific, it is updated with time in order to detect the changes in sentiment vocabulary, and it is improved using reliable hand-labeled tweets.

Since we use the machine learning approach, the sentiment classifier is adaptable and can be updated with time using the new training tweets, which come from the data stream. If we used an approach based on a sentiment lexicon, incremental active learning would not make sense since sentiment-bearing words (senti-words) typically do not change over time (e.g., the word "excellent" surely would not change its sentiment over time). However, taking as a basis the n-gram representation (we use unigrams and bigrams), our approach is different, and not based solely on words that explicitly bear sentiment. The classifier takes into account all the terms appearing in tweets including those representing names of people, products, technologies, countries, etc., whose impact on sentiment may change with time and can even completely shift their sentiment polarity (consider terms like "Ireland," whose sentiment has changed in recent history due to the developments of the financial crisis).

In the proposed dynamic methodology, the active learning algorithm first learns from the Stanford smiley-labeled dataset (Go et al., 2009). According to the current sentiment model, the algorithm selects financial tweets from a first batch of data from a data stream to query for their manual labels and update the model. The process is repeated for the incoming batches of financial tweets until the end of the data stream is reached (in our experiments, the data stream is simulated and it consists of previously collected Baidu financial tweets). In this way, with time and by updating the model with hand-labeled tweets, the sentiment classifier is improved and made more domain specific.

Regarding the correlation between sentiments in tweets that discuss a company's relevant stock information and stock closing prices of the company, as in the static setting, for every day of a time series, positive sentiment probability and its daily changes are calculated. On the other hand, the stock closing prices of the selected company for each day are also collected and daily returns in the stock closing price are calculated. Given the daily changes of the positive sentiment probability time series and the daily returns time series, Granger causality analysis is performed to test whether sentiments in tweets are useful for forecasting the movement of prices on the stock market and assess the significance of the results.

In Section 5.1.1 we present the approach to measuring the classifier's performance, while dynamic neutral zone for detecting the non-opinionated tweets in the active learning setting is explained in Section 5.1.2.

The general idea of our active learning approach is presented as Algorithm 5.1.

5.1.1 Measuring Performance in a Streaming Setting

In the stream-based environment there exist two main approaches for measuring the classifier's performance (Bifet & Kirkby, 2009; Bifet, Holmes, Kirkby, & Pfahringer, 2010; Ikonomovska, Gama, & Džeroski, 2011; Ikonomovska, 2012):

- Holdout evaluation, where the classifier performance is measured using a single unseen holdout set of test examples.
- Interleaved test-then-train or prequential, where each example from a data stream can be used for testing the classifier performance, and then it can be used for training.

For the evaluation of the active learning algorithms, we used the first approach - the holdout evaluation approach. In dynamic environments, where new examples come constantly from a data stream, an algorithm can collect a batch of examples from the stream and use them as a holdout set to evaluate the model (Bifet & Kirkby, 2009; Ikonomovska et al., 2011). This approach is especially suitable for environments where concept drift is assumed, since it allows measuring of how much the algorithm is adaptable to changes

Algorithm 5.1: The active learning approach for the Twitter sentiment analysis.

initialization;
calculate average distance of positive training tweets A_p;
calculate average distance of negative training tweets A_n;
while *not at the end of the data stream* **do**
  collect a batch of examples b from the data stream;
  classify all tweets from b as being positive, negative, or neutral;
  calculate F-measure of the positive class for b;
  for *fixed number of steps* **do**
    $t = select\_tweet(b, strategy)$;
    $l = hand\_label(t)$;
    if l = *'positive' or* l = *'negative'* **then**
      $update\_model(t, l)$;
    end
    if l = *'positive'* **then**
      save SVM distance in a list of distances for positive training tweets L_p;
    end
    if l = *'negative'* **then**
      save SVM distance in a list of distances for negative training tweets L_n;
    end
  end
  if L_p *not empty* **then**
    calculate average distance A_{dp} from L_p ;
    calculate new A_p: $A'_p = (1 - \alpha) * A_p + \alpha * A_{dp}$;
  else
    $A'_p = A_p$;
  end
  if L_n *not empty* **then**
    calculate average distance A_{dn} from L_n ;
    calculate new A_n: $A'_n = (1 - \alpha) * A_n + \alpha * A_{dn}$;
  else
    $A'_n = A_n$;
  end
  clear L_p;
  clear L_n;
end

in the data stream (Bifet & Kirkby, 2009). The evaluation is repeated periodically for new batches of examples which come from the data stream. After the testing of a batch is completed, the active learning algorithm is employed, i.e., the algorithm asks for hand-labels for the most suitable examples from the batch based on a specific active learning strategy. The labeled examples are used for additional training of the algorithm, i.e., to update the sentiment model (see Algorithm 5.1).

We evaluated our classification model by calculating the F-measure of the positive class in two different settings: after every 50 and after every 100 tweets which come from the data stream and represent a batch. After the testing of a batch is completed, the algorithm selects 10 tweets for hand-labeling. Only positive and negative labeled tweets are used for additional training of the sentiment classifier, while the neutral ones are discarded.

5.1.2 Dynamic Neutral Zone

In the static predictive Twitter sentiment analysis, the results (see Section 4.3) showed that it is beneficial to detect also the neutral tweets (besides the positive and negative ones) using the neutral zone. The results also demonstrated the superiority of the relative neutral zone over the fixed neutral zone. Therefore, also in the dynamic setting, we apply the concept of the relative neutral zone for classifying tweets as being neutral.

The first step is to calculate the average positive and negative distances of training examples from the SVM hyperplane. As with every incoming batch from the data stream, the sentiment classifier is updated with a selection of new training tweets, the average distances of positive and negative training examples can also be dynamically updated. Consequently, since the classification reliability and the relative neutral zone are a function of average distances of training examples (see Section 4.1.1.2), the calculation of the classification reliability and the relative neutral zone are made dynamic. Therefore, after the processing of a batch is finished, if there are some additional positive/negative training tweets, the average positive/negative distance is updated following the formula:

$$A' = (1 - \alpha) * A + \alpha * A_d \tag{5.1}$$

where A' is a new average positive/negative distance, A is the current average positive/negative distance, and A_d is the average distance of the additional positive/negative training tweets in the current processed batch. Parameter α controls the influence of the additional training tweets on the new average positive/negative distance. This allows the new training examples to have more influence on the neutral zone than the previous ones. If $\alpha = 0$, the average positive/negative distance is constant and is not changing with time. If $\alpha = 0.1$, the training tweets from a processed batch have influence of 10% on the overall average distance. If there are no positive/negative training tweets in a processed batch, the value of average positive/negative distance remains unchanged regardless of the value of α.

5.2 Experimental Setting

Here we present the active learning implementation, data preparation, query strategies for selecting the tweets for hand-labeling, and statistical tests for measuring the significance of the differences between the multiple active learning settings.

5.2.1 Implementation

In the active learning implementation we used the Pegasos SVM (Shalev-Shwartz, Singer, & Srebro, 2007) learning algorithm from the sofia-ml (Suite of Fast Incremental Algorithms for Machine Learning) library[1] (Sculley, 2009, 2010a, 2010b) of fast incremental algorithms for machine learning. Moreover, we used SWIG (Simplified Wrapper and Interface Generator)[2] to connect sofia-ml (written in C++ programming language) with our implementation of active learning in C# programming language.

For the active learning experiments, we found SVM^{perf} (Joachims, 2005, 2006; Joachims & C.-N. J. Yu, 2009) to be too slow and therefore we used the Pegasos SVM implementation, which is faster than SVM^{perf}, and is adapted for learning from large datasets (Shalev-Shwartz et al., 2007). These features of the Pegasos SVM are beneficial for usage in the stream-based setting. We adjusted the learning algorithm in sofia-ml to our active learning

[1] https://code.google.com/p/sofia-ml/.
[2] http://www.swig.org/.

experiments by implementing sampling which takes the training examples one by one for learning the initial model.

5.2.2 Data Preparation

The initial sentiment model for the active learning experiments was built from the 1,600,000 Stanford general smiley-labeled tweets (Go et al., 2009). The Baidu financial dataset, described in Section 4.2.1, was used for simulating the data stream.

The features for the active learning experiments were prepared by using the LATINO library. For building the feature space for initial smiley-labeled dataset and incoming financial tweets from the data stream, we used an Incremental Bag-of-Words (IncrementalBOW) construction mechanism in the LATINO library. The IncrementalBOW allows fast and incremental updates of the feature space as new examples come from the data stream. One or a collection of new examples can be added to the existing feature space in order to update it. Using the LATINO library we prepared feature sets for the initial sentiment model and the simulated financial data stream. We first read and preprocessed (using the best preprocessing setting, as explained in Section 3.3.2) all Stanford smiley-labeled tweets and initialized the BOW model with them. The resulting features and their values were used by sofia-ml for the initial Pegasos SVM training. Furthermore, in order to prepare the Baidu feature set, the initial feature space was incrementally updated with every Baidu tweet which came from simulated data stream and was adequately preprocessed. After every update, the feature values for the current tweet were saved. The Baidu features and corresponding values were used as input simulated data stream in the implementation of the active learning approach which uses the sofia-ml library.

5.2.3 Active Learning Query Strategies

In the context of active learning, a query strategy is an approach to select the most informative examples which should be given to an oracle (e.g., human annotator) for labeling (Settles, 2009). Based on the new labeled examples the model is updated. Therefore, it is important to choose a good query strategy, which would select the most suitable examples for updating the model and improving its performance. There exist several categories of query strategies (Settles, 2009): uncertainty sampling, query by committee, expected model change, expected error reduction, variance reduction, and density weighted methods. To address the challenging task of stream-based sentiment analysis, we employed and tested several active learning query strategies for selecting the most suitable tweets from a batch for hand-labeling. Our strategies apply uncertainty sampling, random sampling and combinations of both. We implemented the following active learning strategies:

- Active learning closest to the neutral zone (AL closest to NZ): The algorithm inspects tweets in the current processed batch and selects 10 tweets for hand-labeling whose classification reliability is closest to the reliability threshold. Out of the selected ten tweets, at most, five are positive and five are negative, based on positive/negative labeling by the classifier.

- Active learning random 100% (AL rand. 100%): The algorithm randomly selects 10 tweets for hand-labeling from a batch of tweets from the data stream.

- Active learning combination (AL comb.): This strategy combines the other two strategies in order to better explore the SVM space. We experimented with two combinations: 80% "AL closest to NZ" and 20% random strategy (AL comb. 20% rand.) and 50% "AL closest to NZ" and 50% random strategy (AL comb. 50%

rand.). In the combination strategies, the maximum number of positive/negative tweets selected for hand-labeling from a batch with "Closest to the neutral zone" is five.

In order to detect the neutral tweets, we used the relative neutral zone as described in Section 4.1.1.2, since the experimental results showed that this type of neutral zone yielded better results in terms of Granger causality relationship between tweet sentiment and stock closing prices (see Section 4.3). The relative neutral zone was adapted to dynamic stream-based setting, as explained in Section 5.1.2. We experimented with different values of the reliability threshold for determining the neutral tweets in the process of testing the active learning query strategies.

For the evaluation and comparison of different active learning strategies, we used the holdout evaluation approach and calculated the F-measure of positive class for every batch of financial tweets which came from the simulated data stream. The reason for using the F-measure was the unbalanced class distribution in batches. As presented in Algorithm 5.1, the evaluation of a batch was performed first, and then the tweet selection, hand-labeling and updating the sentiment model.

In order to perform comparison of multiple active learning settings, we followed the procedure recommended by Demšar (2006). Namely, we first used the Friedman statistical test (M. Friedman, 1937, 1940) with the Iman-Davenport improvement (Iman & Davenport, 1980) to rank the tested active learning settings and to check whether the difference in performance of the settings was statistically significant. Then, we applied the Nemenyi post-hoc test (Nemenyi, 1963) in order to find where the significant differences occur. These statistical tests helped us to select the best active learning setting for the Twitter sentiment analysis use case.

The Friedman test (M. Friedman, 1937, 1940) ranks the performance results of the tested algorithms for each dataset individually, where the best performing result gets rank 1, the second best rank 2, etc. If the identical performance results are observed (we used the F-measure computed to a precision of four decimal points), average rank values are assigned to the tied algorithms for the specific set. The Friedman test then compares the mean ranks of the algorithms. The null hypothesis states that there is no difference in performances of the tested algorithms and, therefore, their assigned ranks should be the same. If the test rejects the null hypothesis, one can continue with a post-hoc test. We used the Nemenyi post-hoc test (Nemenyi, 1963) in order to make pair-wise comparisons of the algorithms' performances. The test shows where the significant differences in algorithms' performance occur by following the rule that differences are statistically significant if the average ranks of the compared settings vary by at least the critical distance.

Finally, having applied the best active learning setting and performed sentiment analysis on a stream of Baidu financial tweets, we employed the Granger causality analysis (as in Section 4.3) in order to verify if the active learning approach improved the capability of sentiment in tweets to predict the future stock closing prices.

5.3 Experimental Results

In this section, we present the results of experiments for determining the best setting for learning from financial Twitter stream data. Having selected the best active learning setting, we employed the Granger causality analysis to examine the correlation between sentiment in tweet streams and the prices on the stock market.

5.3.1 Selecting the Active Learning Strategy

Having adopted the relative neutral zone from Section 4.1.1.2 for use in the dynamic active learning setting, the first step was to calculate the average distances of training examples from the SVM hyperplane. The average distance of positive training examples was 0.0097 and the average distance of negative training examples was -0.0169.[3] In order to adjust the neutral zone, i.e., the average distances, to the changes in the data stream, the formula for calculating adaptive average distances of positive and negative training examples, given in Equation 5.1, was applied after processing every batch from the Twitter data stream, under the condition that there were some positive/negative training tweets in the processed batch (see Algorithm 5.1).

We examined the impact of the α parameter from Equation 5.1, which controls the influence of additional training tweets on the positive and negative average distances. First, we experimented with $\alpha = 0$, meaning that positive and negative average distances of training examples were constant throughout the whole experiment. In the second experiment the value of α was set to 0.1, meaning that examples from a processed batch had influence of 10% on the average distances. Tables 5.1 and 5.2 show the results for $\alpha = 0$ and $\alpha = 0.1$, respectively, presented in terms of average F-measure values ($\pm$ std. deviation) over all batches from the tweet data stream. In both experiments we employed all the active learning strategies discussed in Section 5.2.3 and the strategy which did not employ active learning (in tables marked as "No AL"), meaning that the sentiment classifier was not updated with time. In the experiments we varied the reliability threshold from 0 to 0.5, in steps of 0.1.

The results in Table 5.1, where $\alpha = 0$, indicate that, in general, active learning improves the performance of the classifiers compared to the strategy which does not employ the active learning approach. Moreover, for all the active learning strategies, the results show that in terms of the F-measure, it is better to have a very small value for reliability threshold. Also, it can be observed, that values of the F-measure are very similar throughout the different active learning strategies for the same reliability threshold value. This could be a consequence of an unchanged average positive/negative distance of training tweets, although new training tweets were used for periodically updating the sentiment model.

On the other hand, the results for $\alpha = 0.1$, in Table 5.2, are more diverse and show bigger differences between the F-measures of the active learning strategies, which leads to the conclusion that, by dynamically adjusting the neutral zone through the average positive/negative distances of training tweets, the approach becomes more sensitive regarding the query strategy. It is interesting to notice that the strategy which did not employ active learning was in general better than the Active random 100% strategy. This implies that the complete random choice of tweets for manual labeling is not a good one in this setting, since the randomly chosen tweets for hand-labeling and their distances from the SVM hyperplane worsen the classifier performance. Recall that when the average distances of training tweets were unchanged, for $\alpha = 0$ (see Table 5.1), the Active random 100% strategy was better than the strategy without active learning. Moreover, it can be noticed that the values of the F-measures in Table 5.2 for the Active learning combination strategies are highest compared both to values in Table 5.2 and Table 5.1. Therefore, if the proper active learning strategy is applied, it is beneficial to dynamically

[3]Note that the values for average distances of training examples are different from the ones in Section 4.3.2. This is a consequence of using a different SVM implementation in the active learning experiments, i.e., we used the Pegasos SVM implementation, instead of SVM^{perf}, which we used in the static setting. We found the Pegasos SVM better for use in the stream-based environment since it is faster, allows incremental learning, and it is adapted for learning from large datasets. See discussion in Section 5.2.1.

Table 5.1: Values of average F-measure ± std. deviation for different strategies, while changing the size of the reliability threshold for $\alpha = 0$.

Reliability threshold	0	0.1	0.2	0.3	0.4	0.5
Select 10 of 100						
AL closest to NZ	0.5512±0.12	0.5396±0.12	0.5282±0.11	0.5165±0.11	0.5018±0.11	0.4802±0.11
AL comb. 20% rand.	0.5512±0.12	0.5396±0.12	0.5281±0.11	0.5164±0.11	0.5017±0.11	0.4803±0.11
AL comb. 50% rand.	0.5513±0.12	0.5398±0.12	0.5283±0.11	0.5165±0.11	0.5016±0.11	0.4803±0.11
AL rand. 100%	0.5514±0.12	0.5399±0.12	0.5281±0.11	0.5169±0.11	0.5017±0.11	0.4804±0.11
No AL	0.5500±0.12	0.5389±0.12	0.5277±0.11	0.5162±0.11	0.5004±0.11	0.4787±0.10
Select 10 of 50						
AL closest to NZ	0.5466±0.14	0.5342±0.14	0.5221±0.14	0.5103±0.14	0.4956±0.13	0.4756±0.13
AL comb. 20% rand.	0.5466±0.14	0.5339±0.14	0.5220±0.14	0.5103±0.14	0.4957±0.13	0.4757±0.13
AL comb. 50% rand.	0.5465±0.14	0.5340±0.14	0.5219±0.14	0.5104±0.14	0.4957±0.13	0.4758±0.13
AL rand. 100%	0.5466±0.14	0.5341±0.14	0.5222±0.14	0.5109±0.14	0.4963±0.13	0.4762±0.13
No AL	0.5444±0.14	0.5329±0.14	0.5213±0.14	0.5094±0.14	0.4938±0.13	0.4731±0.13

Table 5.2: Values of average F-measure ± std. deviation for different strategies, while changing the size of the reliability threshold for $\alpha = 0.1$. Significance of differences in performance of the strategies can be observed in Figures 5.1, 5.2, and 5.3.

Reliability threshold	0	0.1	0.2	0.3	0.4	0.5
Select 10 of 100						
AL closest to NZ	0.5512±0.12	0.5808±0.12*	0.5800±0.10*	0.5923±0.10*	0.5356±0.11*	0.5765±0.10*
AL comb. 20% rand.	0.5530±0.12	0.5463±0.12	0.5432±0.11	0.5375±0.11	0.5289±0.11	0.5102±0.11
AL comb. 50% rand.	0.5513±0.12	0.5415±0.12	0.5320±0.12	0.5246±0.11	0.5116±0.11	0.4831±0.11
AL random 100%	0.5514±0.12	0.5335±0.11	0.5164±0.11	0.4961±0.11	0.4638±0.11	0.4323±0.11
No AL	0.5500±0.12	0.5389±0.12	0.5277±0.11	0.5162±0.11	0.5004±0.11	0.4787±0.10
Select 10 of 50						
AL closest to NZ	0.5766±0.15*	0.5682±0.14	0.6349±0.12*	0.6348±0.11*	0.6250±0.11*	0.5114±0.14
AL comb. 20% rand.	0.5466±0.14	0.5398±0.14	0.5382±0.14	0.5328±0.14	0.5237±0.14	0.5172±0.14
AL comb. 50% rand.	0.5464±0.14	0.5359±0.14	0.5262±0.14	0.5173±0.14	0.4957±0.14	0.4690±0.13
AL random 100%	0.5466±0.14	0.5299±0.14	0.5153±0.14	0.4967±0.14	0.4751±0.13	0.4521±0.13
No AL	0.5444±0.14	0.5329±0.14	0.5213±0.14	0.5094±0.14	0.4938±0.13	0.4731±0.13

* sample contains less than 50% of all data[4]

update the neutral zone through the average positive/negative distances. For that reason, for the rest of the experiments, we focused mainly on the setting where the neutral zone was dynamically adjusted, i.e., where $\alpha = 0.1$.

The results of the Friedman test (M. Friedman, 1937, 1940) with the Iman-Davenport improvement (Iman & Davenport, 1980) and its corresponding post-hoc Nemenyi test (Nemenyi, 1963) for different active learning strategies are graphically represented using critical diagrams. Figure 5.1 shows the results of the analysis of the F-measures from Table 5.2. The diagram presents the mean ranks of the active learning settings, having the lowest (best) ranks on the right side. The critical distance, which connects the settings that are not significantly different, is shown on the top of the graph. From these results we can draw several conclusions. Overall, the best setting for active learning is to choose 10 tweets in each batch of 100 tweets and use the querying strategy "Active learning combination 20% random". This setting is significantly better than "Select 10 of 50 Active

[4]"AL closest to NZ" strategy for $\alpha = 0.1$ proved to be unreliable, since many batches did not contain tweets classified as positive, leading to missing F-measure values for such batches. After examining this phenomenon, we found out that this was a consequence of inverting the average positive distance to a negative number, which means that even though a tweet was positioned on a positive distance from the SVM hyperplane in the classification process, when calculating its classification reliability, the reliability was a negative number, and consequently the tweet was classified as being neutral. Such values for the average F-measure values, which were calculated on an insufficient sample size, are marked with an asterisk in Table 5.2. As a consequence of this phenomenon, we did not use "AL closest to NZ" strategy in the following experiments.

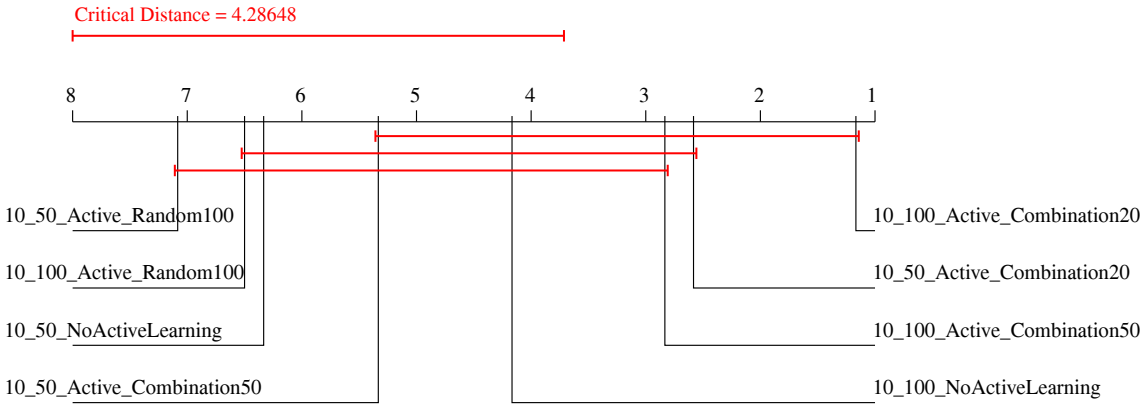

Figure 5.1: Visualisation of Nemenyi post-hoc tests for the active learning strategies on data from Table 5.2.

Random 100%", "Select 10 of 100 Active Random 100%" and "Select 10 of 50 no active learning". Moreover, it seems that in general active combination strategies are the best ones. Also, from the figure it can be observed that "Select 10 of 100" batch selection is in general better than "Select 10 of 50" batch selection, since most of the strategies on the right-hand side of the figure employ "Select 10 of 100" batch selection.

Next, we applied the Friedman test with the Iman-Davenport improvement and the significance post-hoc test on F-measure values of individual batches for the two analyzed batch selection strategies for $\alpha = 0.1$. In Figure 5.2, the results of the test on the case "Select 10 of 50" batch selection can be seen. Similarly, Figure 5.3 shows the results of the "Select 10 of 100" batch selection. From both figures it follows that strategies with the active combination approach are better than the strategies without the active learning approach or the active random strategy. In most of the cases the results are also significant, as can be seen from the figures.

In Smailović et al. (2014) we performed an additional experiment where incremental active learning was performed from Baidu Twitter data only; that is, the active learning algorithm, instead from smiley-labeled dataset, conducted the initial learning from 100 positive and 100 negative tweets chosen from the first 1,000 hand-labeled financial tweets from the Baidu dataset. According to this initial model, the algorithm selected a set of financial tweets from a first batch of data from the Baidu tweet data stream to query for their labels. Based on these hand-labeled financial tweets, the model was updated and the process was repeated for the next batch of Baidu tweets. This process was repeated until the end of the simulated data stream was reached. The results indicated that the classifier learned on such a small initial dataset, although hand-labeled and specific for the financial domain, was highly unstable. The sentiment classifier learned on this dataset classified all tweets at the beginning of the data stream as negative. Then, as a consequence of active learning and improving the classifier with new labeled tweets, the classifier improved and started to classify new tweets as positive or negative. This improvement lasted for several batches, and then the classifier classified all the newcoming tweets as positive. This behavior indicated that the classifier was highly unstable since incremental learning introduced significant changes into the model with the occurrence of every new labeled tweet. Although this experiment in Smailović et al. (2014) was performed using a different active learning setting, it showed that in general sentiment classifier is unstable if it is trained on a small dataset, even if such a dataset is manually labeled and domain specific.

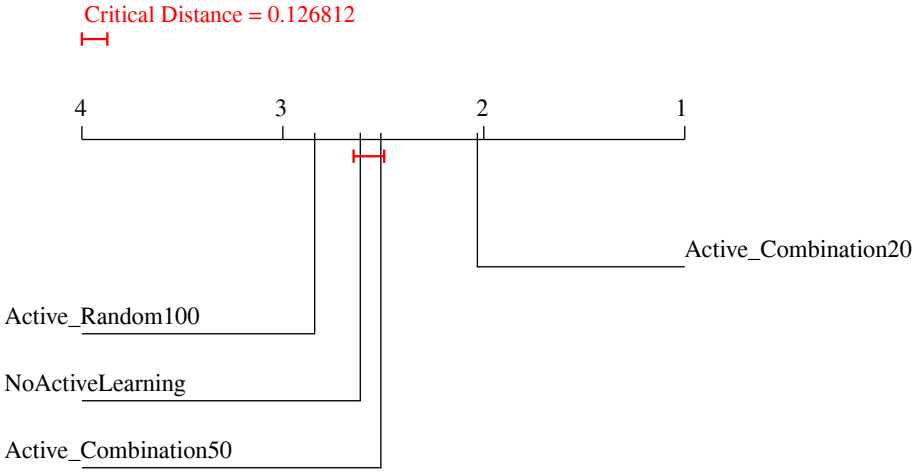

Figure 5.2: Visualization of Nemenyi post-hoc tests for the "Select 10 of 50" batch selection for $\alpha = 0.1$.

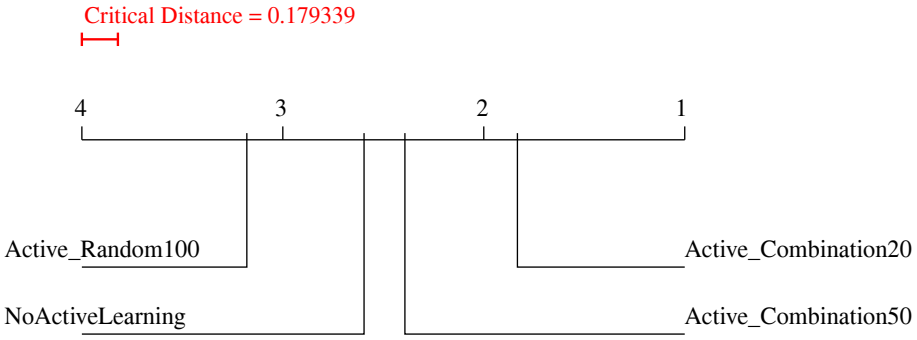

Figure 5.3: Visualization of Nemenyi post-hoc tests for the "Select 10 of 100" batch selection for $\alpha = 0.1$.

5.3.2 Stock Market Analysis

The active learning experiments from Section 5.3.1 indicate that the best setting for learning from financial Twitter stream data is to divide tweets from the data stream into batches of 100 tweets out of which 10 tweets from each batch are selected for hand-labeling using the active combination querying strategies. Therefore, we selected the strategies which combine 20% and 50% random with the "Active learning closest to the neutral zone" strategy. We repeated the Granger causality analysis in order to examine if by using these strategies the predictive power of financial tweets to predict the stock closing price of the Baidu company would improve. The results are shown in Table 5.3.

Since in our experiments we computed the p-value repetitively and made multiple comparisons of p-values, we applied the Bonferroni correction (Abdi, 2007) to neutralize the problem of multiple comparisons as we did in the static part of our study (Section 4.2.2). As can be seen from Table 5.3, the best correlations were obtained for reliability threshold value 0.5 where we obtained three significant correlations for the June-August time period for the both tested active learning strategies. In comparison with the static approach (see Table 4.2), where the significant results were dispersed across different time periods, in the dynamic setting, all the significant results between sentiments in tweets and stock closing prices in Table 5.3 were for the June-August time period. Since the sentiment classifier in the dynamic setting was more domain specific (as a consequence of using the

Table 5.3: Statistical significance (p-values) of Granger causality correlation between positive sentiment probability and the closing stock price for Baidu using active learning, while changing the value of the reliability threshold. Two combined strategies for selecting 10 of 100 tweets for labeling are presented. Values which are lower than a p-value of 0.1, after applying the Bonferroni correction, are marked in bold.

Reliability threshold		0	0.1	0.2	0.3	0.4	0.5
Select 10 of 100, combined 20% random	Lag						
9 months	1	0.236	0.126	0.133	0.116	0.189	0.043
Mar.-May	1	0.642	0.870	0.949	0.968	0.895	0.952
June-Aug.	1	0.088	0.059	0.070	0.049	0.056	**0.017**
Sept.-Nov.	1	0.812	0.764	0.778	0.766	0.882	0.541
9 months	2	0.278	0.211	0.222	0.195	0.318	0.059
Mar.-May	2	0.375	0.493	0.437	0.470	0.521	0.117
June-Aug.	2	0.108	0.068	0.076	0.036	0.055	**0.021**
Sept.-Nov.	2	0.512	0.644	0.755	0.866	0.941	0.665
9 months	3	0.180	0.251	0.281	0.226	0.271	0.083
Mar.-May	3	0.535	0.647	0.657	0.711	0.720	0.259
June-Aug.	3	0.136	0.071	0.073	**0.023**	0.060	**0.024**
Sept.-Nov.	3	0.254	0.261	0.347	0.409	0.344	0.166
Select 10 of 100, combined 50% random	Lag						
9 months	1	0.221	0.115	0.086	0.324	0.048	0.138
Mar.-May	1	0.642	0.963	0.859	0.620	0.836	0.713
June-Aug.	1	0.088	0.059	0.035	0.081	**0.009**	**0.007**
Sept.-Nov.	1	0.794	0.785	0.631	0.963	0.626	0.931
9 months	2	0.347	0.171	0.152	0.420	0.039	0.033
Mar.-May	2	0.375	0.516	0.298	0.263	0.191	0.266
June-Aug.	2	0.108	0.049	0.034	0.084	**0.013**	**0.004**
Sept.-Nov.	2	0.662	0.722	0.764	0.962	0.733	0.570
9 months	3	0.367	0.224	0.189	0.430	0.040	0.052
Mar.-May	3	0.535	0.724	0.523	0.465	0.371	0.344
June-Aug.	3	0.136	0.042	**0.022**	0.102	0.031	**0.010**
Sept.-Nov.	3	0.270	0.378	0.297	0.512	0.191	0.108

financial tweets in the training phase), this could be an indicator of important events for the Baidu finances occurring in this time period. Therefore, we investigated in more detail the public web media for Baidu from this time period. Figure 5.4 shows a screenshot from the Google Finance[5] web page displaying stock price and news media coverage for Baidu in 2011. From the figure, it can be observed that most of the key events in 2011 happened in the period from June to August. Note that this period is also characterized by the highest number of press releases[6] for Baidu in 2011. We hypothesize that this resulted in higher media exposure and, consequently, caused discussions and speculations in social media about future price movements. However, further studies are required to confirm or reject this claim.

Additionally, we explored whether there is evidence for the reversed causality (that the price movements may influence the public sentiment). The results show that after making adjustments to the critical p-value by applying the Bonferroni correction, there were no significant results of this kind.

In Smailović et al. (2014) we presented initial experimental results of a simulation where we tested whether the sentiments in tweets can predict values of future stock prices in real-time and, consequently, provide returns. Initial results indicated that by augmenting a

[5]https://www.google.com/finance

[6]http://phx.corporate-ir.net/phoenix.zhtml?c=188488&p=irol-news&nyo=3

Figure 5.4: Screenshot from the Google Finance web page showing stock prices and key events. It can be observed that most of the key events in 2011 happened in the period from June to August. We hypothesize that this resulted in a higher media exposure and, consequently, enabled discussions and speculations about price movements in social media.

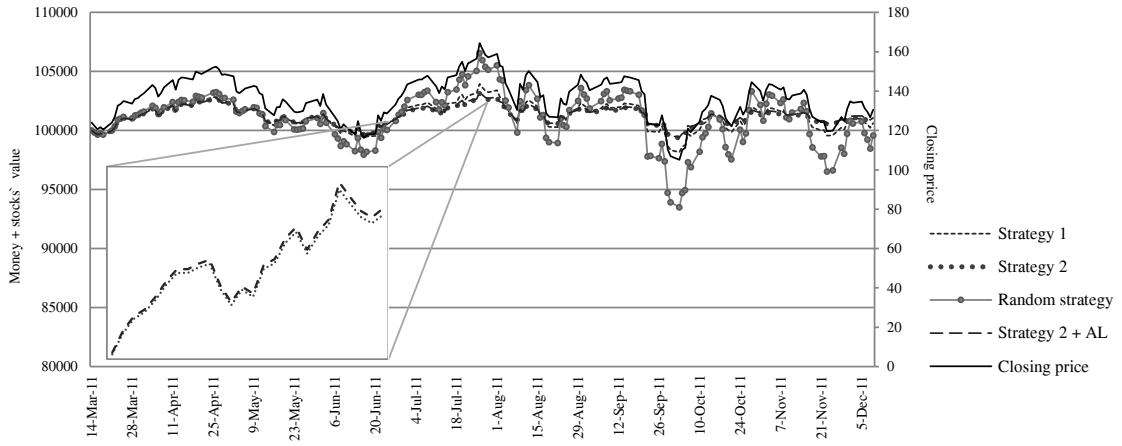

Figure 5.5: Simulation of online experiments from Smailović, Grčar, Lavrač, and Žnidaršič (2014) for predicting values of future stock prices in real-time. The x-axis presents the dates, while the y-axis shows the sum of money and stocks values.

trading strategy with consideration of the changes in the values of positive sentiment probability one could improve the returns. In the experiments, the lowest performance was observed with the random strategy, while the best performance was obtained with the strategy which used the active learning approach. In Figure 5.5 a graphical simulation of several tested strategies from the paper is presented. The time period between June 24 and August 1 for a strategy without the active learning and the strategy which employs the active learning algorithm is zoomed in, since in this time period, the strategy which employs the active learning algorithm started outperforming the strategy without active learning and remained better until the end of the simulation. Although online experiments in (Smailović et al., 2014) were performed in a different active learning setting, they showed the potential that tweet sentiment can predict stock prices in real-time and provide returns. Moreover, the experiments showed the superiority of the active learning approach in this setting.

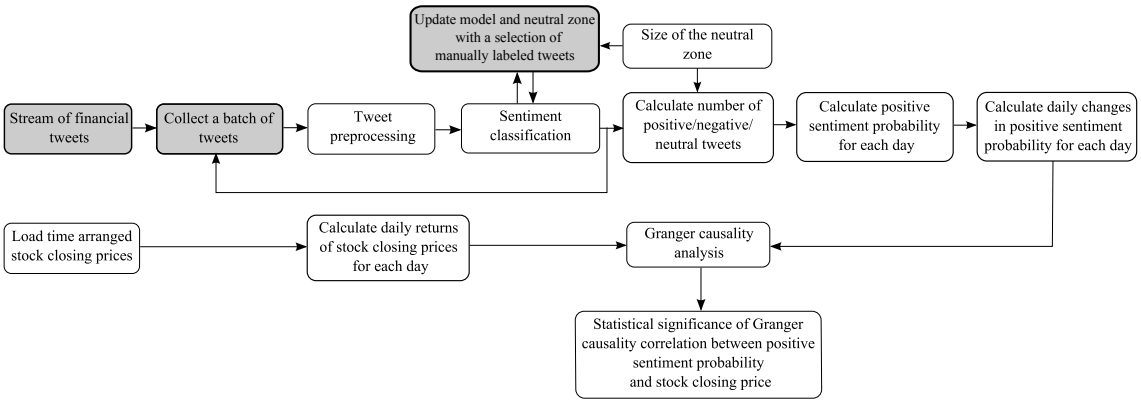

Figure 5.6: Methodological steps for stream-based active learning for Twitter sentiment analysis in finance. Components which are specific to the stream-based setting and are not present in the static setting (Figure 4.2) are colored gray.

5.4 Methodology and Results Summary

The proposed methodology for stream-based active learning for Twitter sentiment analysis in finance consists of the sequence of steps presented in Figure 5.6. Components which are specific to the stream-based setting and not present in the static setting (Figure 4.2) are emphasized with a gray background.

As can be seen from the figure, one should first provide a stream of financial tweets discussing stock relevant information concerning a company of interest. The algorithm then collects and preprocesses a batch of examples from the stream. After classification of tweets as positive, negative, or neutral, based on a querying strategy, the algorithm selects tweets for hand-labeling. With the new labeled data, the sentiment model and the neutral zone are updated. These steps are repeated for all batches in the data stream. For every day of the time series, the positive sentiment probability is computed by dividing the number of positive tweets per day by the total number of tweets in that day. Lastly, daily changes of the positive sentiment probability are calculated.

On the other hand, the stock closing prices of a selected company for each day should be collected. The daily returns in the stock closing price are then calculated in order to satisfy stationary conditions demanded by the Granger causality test.

Given the daily changes of the positive sentiment probability time series and the daily returns in the stock closing price time series, Granger causality analysis is performed (considering lagged values of time series for one, two, and three days) to test whether tweet sentiment is useful for forecasting the movement of prices in the stock market and the significance of the results.

By performing the experiments in this chapter, we tested the hypothesis from the Introduction, which states that the active learning approach improves upon the static methodology (in terms of adapting the sentiment classifier to a specific domain and improving the F-measure of tweet sentiment classification) and improves its predictive power by adapting to changes with time in data streams (see Section 1.4). On the one hand, active learning did indeed make the sentiment classifier more domain specific and extend the approach with a capability of continuous updating of the classifier by adding hand-labeled financial tweets in the training dataset. Regarding the improvement of the F-measure of tweet sentiment classification, the experiments showed that, when the relative neutral zone is not dynamic ($\alpha = 0$), active learning improves the performance of the classifiers compared to the strategy which does not employ the active learning approach (see Ta-

ble 5.1). When the relative neutral zone is dynamic, the improvement is visible for selected active learning strategies (see Table 5.2). On the other hand, in terms of Granger causality and a relationship between tweet sentiment and future stock closing prices, we obtained significant results in specific periods, but their number was lower in comparison with the static approach. The experiments in the dynamic setting indicated that the significant results were more focused on a specific time period which was characterized by a large number of key events and press releases for the analyzed company, which could be possible added value of the dynamic approach, but further experiments are needed in order to fully accept this claim. Therefore, based on the experiments in this chapter we cannot accept the above mentioned hypothesis. It should be noted, however, that our preliminary results on this topic showed that the active learning approach improved the static methodology, but with the advances of the static approach in terms of new relative neutral zone and better preprocessing, the static approach started providing extremely good results, and consequently the improvement of the active learning vanished.

Chapter 6

Implementations and Applications

Many researchers conduct studies and publish written material (papers, books, theses, etc.) about their work. However, making the developed methodologies and implementations publicly available adds more value and allows other researchers to replicate, validate, or improve the proposed methods and results. Also, by making the research available for others, its impact is potentially increased. For these reasons, we made selected parts of the work, described in this dissertation, publicly available through an interactive data mining platform. Namely, the workflows and the individual components that perform sentiment analysis, apply active learning, and monitor tweets discussing politics have been made publicly available in ClowdFlows (Kranjc, Podpečan, & Lavrač, 2012) data mining platform (see Section 6.1). Furthermore, the developed sentiment methodology for various languages has been used in the PerceptionAnalytics platform, which in real-time analyzes messages posted on popular social media Web sites (see Section 6.2).

Moreover, additional value and impact can be achieved if research methods are applied in real-life situations. In our case, the sentiment analysis methodology was successfully applied to monitoring of sentiment in Twitter messages discussing the Slovenian and Bulgarian elections, as described in Sections 6.3 and 6.4. By doing so, we tested and confirmed a hypothesis from the Introduction (Section 1.4) of this dissertation which states that the developed sentiment analysis methodology is applicable in real-world applications.

In our published work in Kranjc et al. (2014) we describe the implementation of sentiment analysis with active learning in ClowdFlows, while in this chapter we provide also the details about a new version of active learning in ClowdFlows, which is based on the experiments presented in this dissertation. We have not published yet our research work related to the two elections use cases and the PerceptionAnalytics platform.

6.1 Implementations in the ClowdFlows Platform

ClowdFlows (Kranjc et al., 2012) is an open source interactive data mining platform, which allows its users to create new or explore existing data mining workflows, execute them, and share the workflows and the results. The platform is free, cloud-based, requires no installation, and runs on major Web browsers and mobile devices. In order to start creating a new workflow or manipulating an existing one, one should first register and log in. The ClowdFlows platform is available at http://clowdflows.org/.

In the process of creating a workflow, the user has the possibility to use the existing workflow components (Big data, ILP, NLP, etc.) available in ClowdFlows or to import an additional component in the form of a Web service by entering a WSDL URL. Workflow components are called widgets. Typical widget's task in ClowdFlows is to load data, preprocess it, apply an algorithm, provide a visualization of the results, interact with the

user, or even create a sub-workflow inside of it. The widget types and their functions are described in detail in (Kranjc, Podpečan, & Lavrač, 2013). The widgets in ClowdFlows can be combined with each other using the graphical user interface (GUI) in order to construct a workflow.

Currently, the ClowdFlows platform offers a number of publicly available data mining workflows and individual widgets created by various researchers. Among them there are also components and workflows which allow the user to perform sentiment analysis and active learning on Twitter messages. These are implemented as a result of the research described in this dissertation and in collaboration with the lead developer of the platform, Janez Kranjc. We describe them in the following subsections. Specifically, in Section 6.1.1, we describe a sentiment analysis widget in Clowdflows. Section 6.1.2 presents a workflow, which performs Twitter sentiment analysis with active learning. Finally, Section 6.1.3 presents a workflow which was constructed for monitoring tweets discussing the Bulgarian parliamentary elections.

6.1.1 Sentiment Analysis Widget

The Twitter sentiment classifier, described in Chapter 3, is available in the ClowdFlows data mining platform as a workflow component, i.e., widget. The choice of the training dataset, preprocessing setting, and the algorithm for implementing the sentiment analysis classifier used in the platform was based on the discussions and experimental results presented in this dissertation. The training dataset for learning the classifier consisted of 1,600,000 smiley-labeled tweets (Go et al., 2009), as explained in Section 3.2.1. We applied the best standard and Twitter-specific preprocessing to the training tweets, following the lessons learned from our experiments described in Section 3.3.2. For building the sentiment classifier the SVM^{perf} (Joachims, 2005, 2006; Joachims & C.-N. J. Yu, 2009) implementation of the linear Support Vector Machine (SVM) (Boser et al., 1992; Cortes & Vapnik, 1995; Vapnik, 1995, 1998) algorithm was used, which was the choice based on the experiments described in Sections 3.1.1 and 3.3.1. The reliability of the sentiment classification is calculated as described in Section 4.1.1.2. The developed sentiment classifier is static, i.e., unchanged with time.

The developed Twitter sentiment classifier was exposed as a Web service and this service was imported in ClowdFlows in order to create a sentiment analysis widget. The widget receives a list of tweets on the input, which can be obtained from the Twitter API or an uploaded file. The tweets are then preprocessed and sentiment analysis is performed. Finally, information about a tweet, sentiment class, and classification reliability for every tweet are sent to the output of the widget. The sentiment class provided by the sentiment analysis widget can be positive or negative. If the user wants to detect also the neutral tweets, the "Add neutral zone" widget should be connected to the output of the tweet sentiment analysis widget. If a tweet, sent from the sentiment analysis widget, has a classification reliability below the threshold (which is a parameter of the "Add neutral zone" widget), sentiment class of the tweet is changed to neutral. An example of several connected widgets for obtaining Twitter messages, filtering the messages by language (based on information about language provided by Twitter), and performing sentiment analysis is presented in Figure 6.1. In the figure we show also a selected input for the first widget of the workflow, i.e., a "Twitter" widget input, and selection of outputs of the last widget of the workflow, i.e., "Add neutral zone" widget outputs. Note that, if the user wants to inspect the results over time, additional widgets, which provide viewing of all the collected results, can also be added. Otherwise, the user can only observe the results of the last run.

An example of a workflow which employs sentiment analysis and visualizes the results is

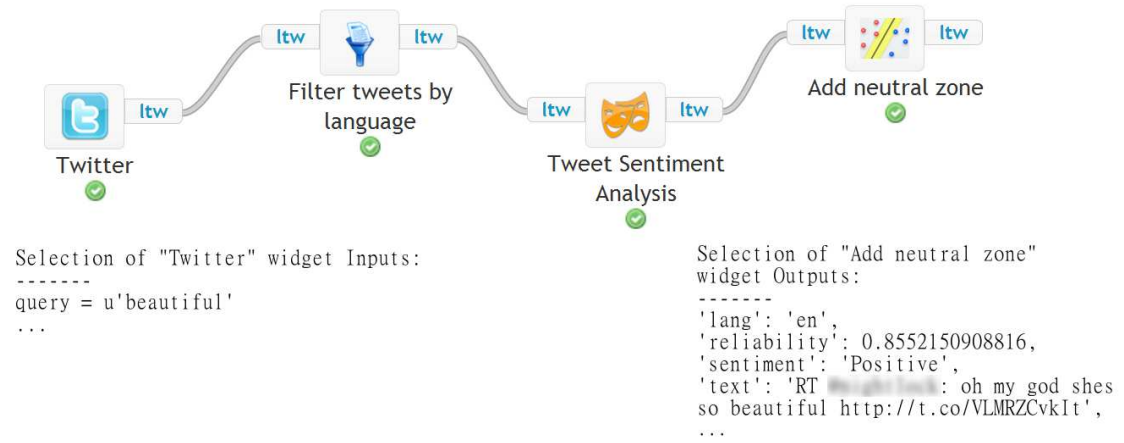

Figure 6.1: An example of several connected widgets for obtaining Twitter messages, filtering them by language and performing sentiment analysis. A selected input for the first widget of the workflow, i.e., a "Twitter" widget input, and selection of outputs of the last widget of the workflow, i.e., "Add neutral zone" widget outputs, are also presented.

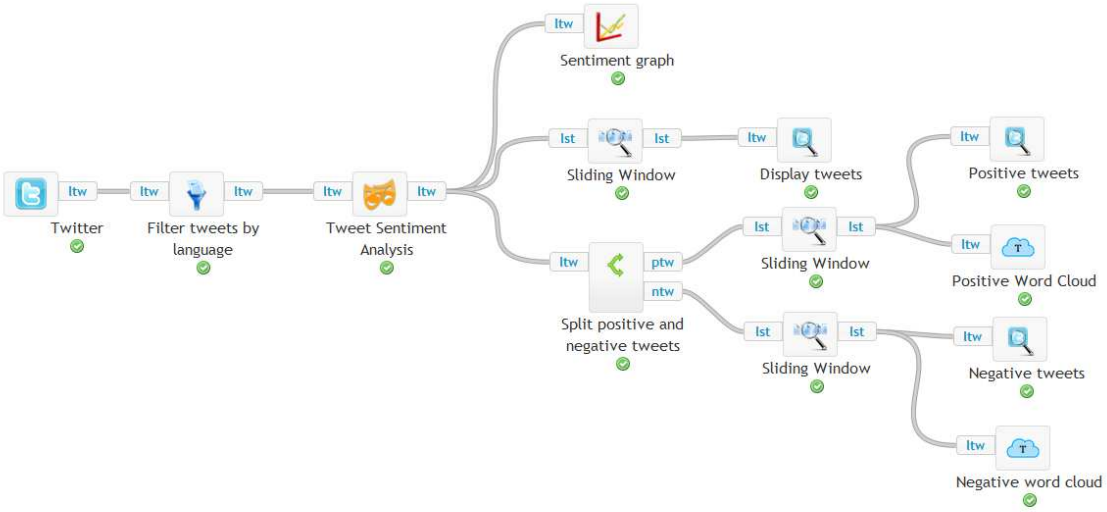

Figure 6.2: The workflow for Twitter sentiment analysis which collects tweets from the Twitter API, filters them by language, performs sentiment analysis, and shows the results in the form of a sentiment graph, word clouds, and most recent individual tweets.

the one constructed in Kranjc et al. (2013) and presented in Figure 6.2. Here the sentiment classifier widget[1] is combined with several other components to create a workflow which collects tweets from the Twitter API in real-time, filters them by language, performs sentiment analysis, and eventually shows the results in the form of a sentiment graph, positive and negative word clouds, and most recent individual tweets. More details about this workflow can be found in Kranjc et al. (2013). The workflow is available at: http://clowdflows.org/workflow/1041/.

[1]Note that the version of the sentiment classifier in (Kranjc et al., 2013) is older than the one we are discussing in this section.

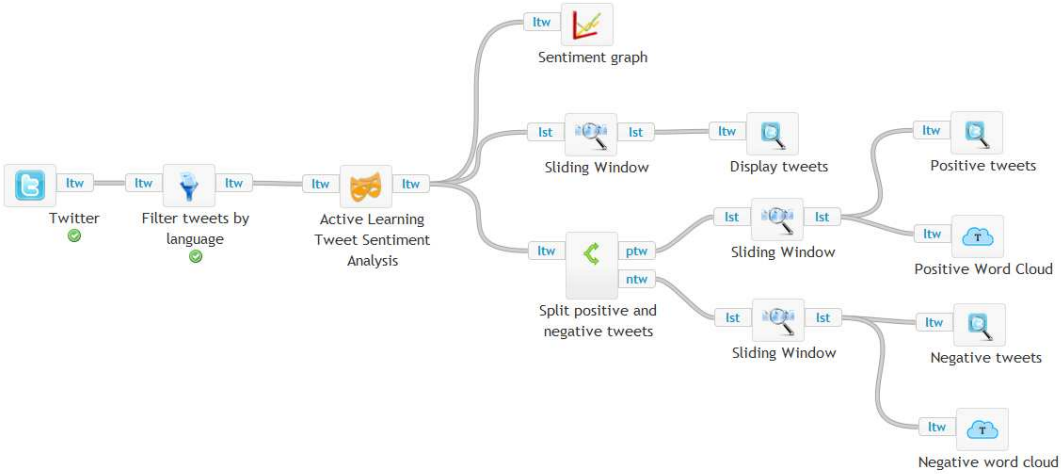

Figure 6.3: The workflow in the ClowdFlows platform for Twitter sentiment analysis with active learning from Kranjc et al. (2014).

6.1.2 Sentiment Analysis with Active Learning Workflow

In this section we present a workflow in the ClowdFlows platform, which performs Twitter sentiment analysis with active learning. The workflow is available at: http://clowdflows.org/workflow/1422/ and is presented in Figure 6.3.

As can be seen from Figure 6.3, the workflow is similar to the one in Figure 6.2 with the only difference being that instead of the "Tweet Sentiment Analysis" widget there is the "Active Learning Tweet Sentiment Analysis" widget. The name of the new widget indicates that it performs sentiment analysis with the enhancement of the active learning approach. The workflow collects Twitter messages based on a user specified query, filters them by language, applies selected preprocessing techniques, performs sentiment analysis, and presents the results in form of individual tweets, aggregated sentiment graphs, and word clouds of positive and negative tweets. Additionally, as a part of the active learning process, the user has the possibility to manually label a selection of tweets given by the algorithm in order to improve the performance of the sentiment classifier. The labeling interface is shown in Figure 6.4. Tweets are initially labeled as neutral, while the user can manually label them also as positive or negative.

The workflow from Figure 6.3 is a result of the study presented in Kranjc et al. (2014). It can be observed that there is no widget for employing the neutral zone, but it can be easily added and applied, as described in Section 6.1.1.

As a result of new experiments and results in the dissertation, we have implemented the second version of sentiment analysis and active learning widget in ClowdFlows (Active Learning Tweet Sentiment Analysis v2), which is different than the one described by Kranjc et al. (2014). Regarding the sentiment analysis in the new version of the active learning widget, the choice of the training dataset, standard and Twitter-specific preprocessing setting, and the algorithm for implementing the Twitter sentiment classifier are based on the discussions and experimental results presented in Chapter 3 of this dissertation. On the other hand, the implementation of the active learning approach is based on the experimental results in Section 5.3. More details are provided below.

The approach to sentiment analysis and active learning was developed as a Web service and imported in ClowdFlows. The service offers operations for classifying a collection of

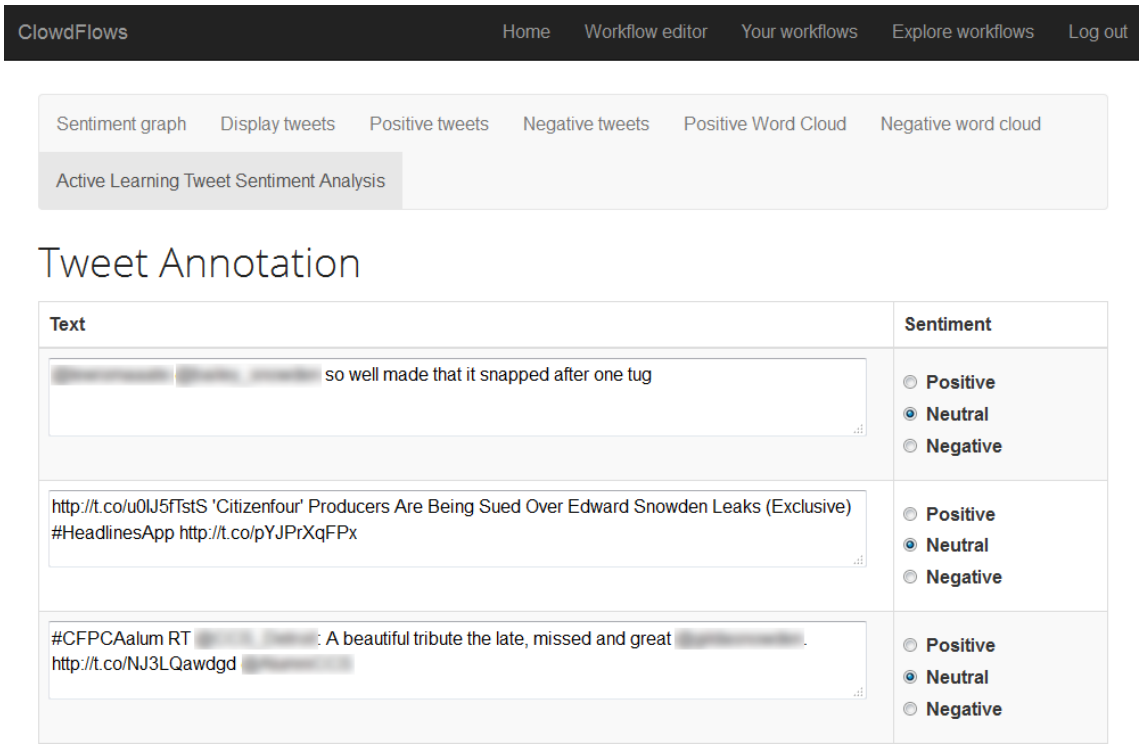

Figure 6.4: The labeling interface in the ClowdFlows platform as a part of active learning. Tweets are initially labeled as neutral, while the user can manually label them also as positive or negative. Twitter usernames are blurred in order to hide personal information.

tweets, providing a set of tweets for hand-labeling as a part of the active learning process, and updating the sentiment classifier with obtained hand-labeled tweets. It supports multiple workflows and builds a separate sentiment model for each of them. The initial sentiment model is the same for all workflows, but differences occur later, when active learning is applied. Namely, the updated models for workflows are different since every workflow obtains a distinct collection of additional hand-labeled training tweets. Differences between the hand-labeled collections may arise as a consequence of different queries for collecting the tweets, or, if the queries are the same, a different amount of hand-labeled tweets or different hand-labels of the same tweets in the collections. In order to distinguish between the different workflows, they are given unique identifiers.

The default query strategy in the active learning workflow is the one that performed best in the experiments in Section 5.3, i.e., the strategy that divides tweets from the data stream into batches of 100 tweets out of which 10 tweets from each batch are selected for hand-labeling using the active combination querying strategy (combination of 80% "Closest to the neutral zone" query strategy with the 20% random strategy). Therefore, out of 10 tweets for hand-labeling the algorithm selects 8 tweets whose classification reliability is closest to the reliability threshold and puts them into the pool of query tweets, so that the top-most are the ones which are closest to the reliability threshold. The value for the reliability threshold is set to 0.1. The other two tweets for hand-labeling are chosen randomly from the batch and put into a separate pool of random tweets. With time, as new tweets arrive, the pools are updated. Whenever the user decides to conduct manual labeling of the tweets, she is presented with a set of tweets to label, which contains 8 that are closest to the reliability threshold and 2 random ones from the pool of random tweets. The hand-labeled tweets are placed in the pool of labeled tweets. Once a day, using the

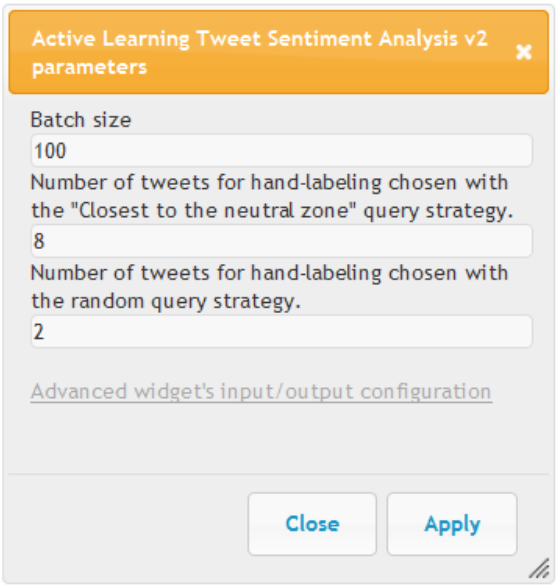

Figure 6.5: Default parameters for the active learning widget in ClowdFlows, which can be changed by the user.

initial set of training data and manually labeled positive and negative tweets from the pool of labeled tweets, the sentiment models are retrained in every workflow which obtained new hand-labeled tweets.

Besides using the default setting for the active learning query strategy, we enabled the user also with the possibility to change the following parameters:

- batch size,
- number of tweets for hand-labeling chosen with the "Closest to the neutral zone" query strategy, and
- number of tweets for hand-labeling chosen with the random strategy.

Figure 6.5 presents a screenshot of a window obtained with double-click on the newest version of the active learning widget. There one can observe several default active learning parameters, which can be changed.

We applied the relative neutral zone from Section 4.1.1.2 in order to calculate classification reliability and to detect the neutral tweets. In order to adjust the neutral zone, i.e., the average distances of positive and negative training examples, to the changes in the data stream, the formula for calculating adaptive average distances of positive and negative training examples, given in Equation 5.1, is applied every time after the sentiment model is updated. We set α parameter from the equation to 0.1, meaning that new manually labeled tweets for a specific workflow have influence of 10% on the average distances.

6.1.3 Bulgarian Parliamentary Elections Workflow

Section 6.4 presents a study on monitoring tweets discussing the Bulgarian parliamentary elections held in 2013. The developed methodology is available in the ClowdFlows platform as a standalone workflow which was executed in real-time during the election campaign.

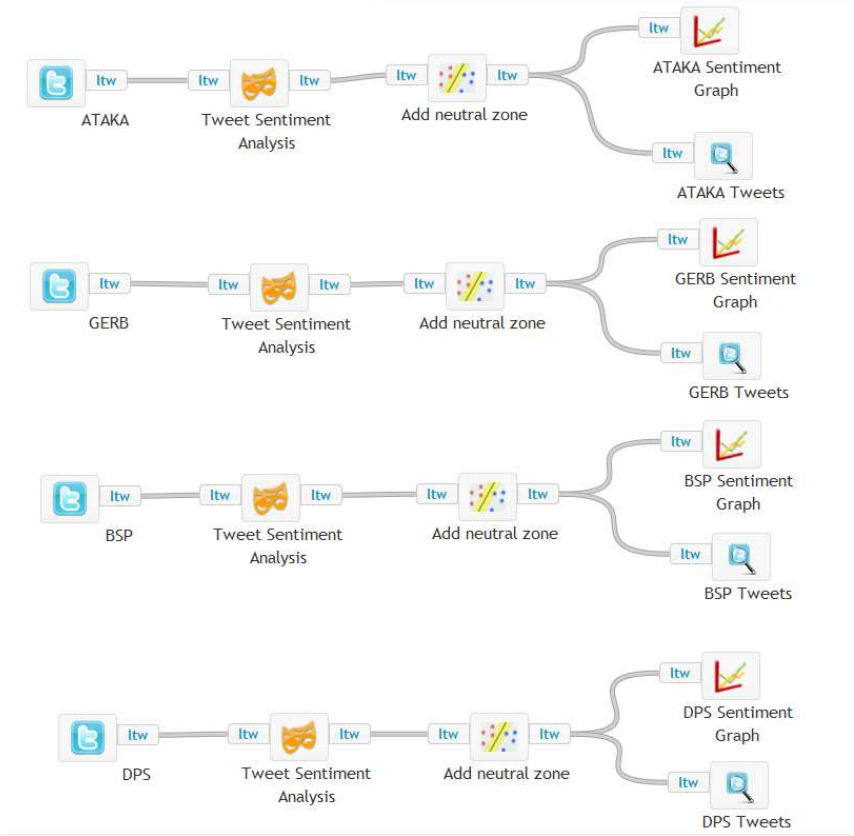

Figure 6.6: The workflow for Twitter sentiment analysis in the Bulgarian elections use case.

The workflow is publicly available online at: http://clowdflows.org/workflow/1115/ and it is shown in Figure 6.6. As can be seen from the figure, for every analyzed political party, we collected the relevant tweets and performed Twitter sentiment analysis by employing the sentiment classifier and the concept of neutral zone. Also, on the one hand, the user had the possibility to go through the results by inspecting the individual tweets, their sentiment, and classification reliabilities, and on the other hand, by examining the aggregated sentiment graphs over time. The resulting sentiment graphs and individual tweets for every political party are available at: http://clowdflows.org/streams/data/10/9691/. For the purpose of reproducibility of the results, the Twitter data streams in the workflow are simulated from static uploaded data of tweets collected during the elections. Details about the Twitter sentiment classifier used in this setting are given in Section 6.4.

6.2 Implementations in the PerceptionAnalytics Platform

PerceptionAnalytics is a Web platform which provides an insight into public discussions, opinions, and trends on popular social media Web sites, like Twitter or Facebook. It is available online at: http://www.perceptionanalytics.net/. The platform is a product of a Slovenian company Gama System.[2]

Based on user specified search criteria, which can include a specific keyword, location, username, or an account on a social network Web site, the PerceptionAnalytics platform collects messages that match the defined criteria and analyzes them in real-time. The

[2]http://www.gama-system.si/en/.

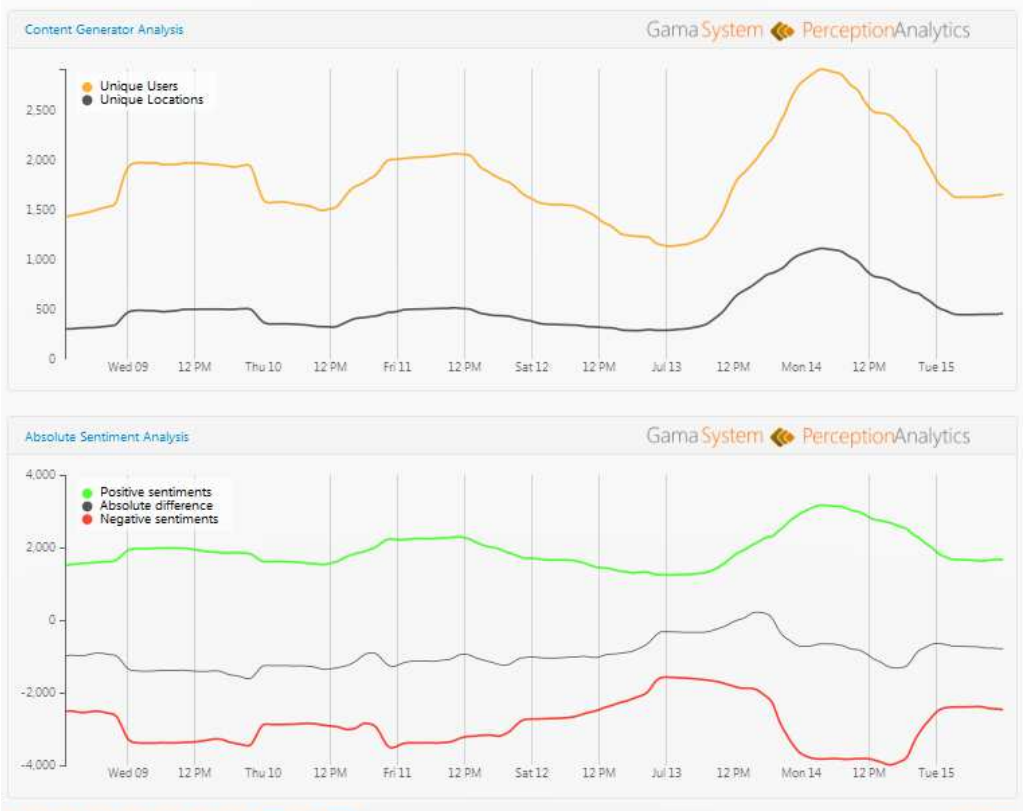

Figure 6.7: A part of an analysis report in the PerceptionAnalytics platform.

platform provides various analyses: hit count, sentiment, content, top user, word cloud, tag cloud, communication cloud analysis, etc. All analyses can be combined in an analysis report, whose part is shown in Figure 6.7. Our sentiment analysis methodology was incorporated into this platform.[3]

6.2.1 Sentiment Analysis for Multiple Languages

For the PerceptionAnalytics platform we have provided sentiment analysis for a number of languages: English, Slovenian, Spanish, German, Russian, Hungarian, Polish, Portuguese, Bulgarian, Arabic, Albanian, and the unified sentiment analysis for Croatian, Serbian, Montenegrin, and Bosnian languages. The collaboration with the Gama System company is still active, and we intend to expand the ability of performing sentiment analysis for several new languages.

As we use the machine learning approach for training the sentiment classifiers, datasets of manually labeled language-specific tweets have to be obtained. For every language one or more native speakers manually label a set of tweets. Manual annotations are carried out using the Goldfinch annotation platform.[4] Collections of tweets given to the annotators contain approximately 20% of duplicate tweets which can be labeled by the same or different annotators. The duplicate tweets offer a possibility to compute the inter-rater agreement, which is a measure that shows the degree of agreement in independent judgments among annotators (Gwet, 2001; Jelles, Van Bennekom, Lankhorst, Sibbel, &

[3]With the Gama System company we have had a noncommercial cooperation within a research project funded by the Slovenian Ministry of Education, Science, Culture and Sport.

[4]The Goldfinch annotation platform is developed by the Sowa Labs company, http://sowalabs.com/.

Bouter, 1995; Nowak & Rüger, 2010) or intra-rater agreement, which shows how reproducible are the judgments provided by one annotator for the same instance (Jelles et al., 1995). At the time of writing this dissertation we do not have the results of inter- or intra-annotator agreements, but we plan to conduct such analyses as part of the future work.

After a sufficient amount of hand-labeled tweets is obtained for a specific language, a sentiment model is built. For training the model we use all the unique hand-labeled tweets and a subset of duplicate tweets.[5] Namely, if the annotators of two duplicated tweets are different, both tweets are used to train the model. On the other hand, if the annotator is the same, only one randomly chosen tweet from two duplicated tweets is selected to be in the training dataset. The hand-labeled tweets used for training the sentiment classifier are then preprocessed using standard and Twitter-specific text-processing.[6] The preprocessed tweets are used for training the linear SVM (Boser et al., 1992; Cortes & Vapnik, 1995; Vapnik, 1995, 1998) language-specific sentiment models, which are then used in classifying new tweets from the data stream in real-time.

The neutral tweets in the platform are detected by employing the relative neutral zone, as described in Section 4.1.1.2. For every language we calculate average positive and negative distances of training tweets from the SVM hyperplane and apply Equation 4.1 to calculate the reliability of sentiment classification. In real-time, a tweet is classified as neutral if its classification reliability is below a predefined threshold.

Figure 6.8 presents a screenshot of the PerceptionAnalytics platform, showing individual tweets and the attached sentiment. Every tweet has attached a sentiment label (positive, negative, or neutral), a percentage of estimated reliability of sentiment classification, and a selection of words which influenced the classification. The last two features are visible when hovering with mouse pointer over the sentiment label.

6.3 Real-time Opinion Monitoring: Slovenian Presidential Elections Use Case

In this section we present the approach to constructing the Slovene Twitter sentiment classifier and its use on political tweets written in Slovene language. In Section 6.3.1 we present the approach which initially used a collection of Slovene smiley-labeled tweets as a training dataset, followed by the approach which employed high quality hand-labeled tweets. Moreover, in Section 6.3.2, we present the approach to predicting the results of the Slovenian presidential elections, held on November 11, 2012. Finally, Section 6.3.3 presents real-time monitoring of sentiment in Twitter messages during the elections.

6.3.1 Twitter Sentiment Analysis

This section presents the construction of the sentiment classifier and its use on Slovene political tweets. We performed several experiments. In the first experiment we trained the sentiment classifier on smiley-labeled Slovene tweets and tested it in the two-class setting

[5] In this context, duplicate tweets are the ones with the same IDs. Therefore, if two tweets have identical content, but different IDs, they are not considered to be duplicates.

[6] The Twitter data processing was implemented using the LATINO software library. It was performed according to our experience and experimental results at the time. The approach was improved with new knowledge, experience, and experiments, and applied to new sentiment models. For example, at the beginning, for the first analyzed languages, we applied the TF-IDF scheme for feature vector construction. Later, after showing that the TF-based approach is statistically significantly better than the TF-IDF approach in the Twitter sentiment classification setting (Smailović et al., 2014), we used only the TF scheme for subsequent sentiment models.

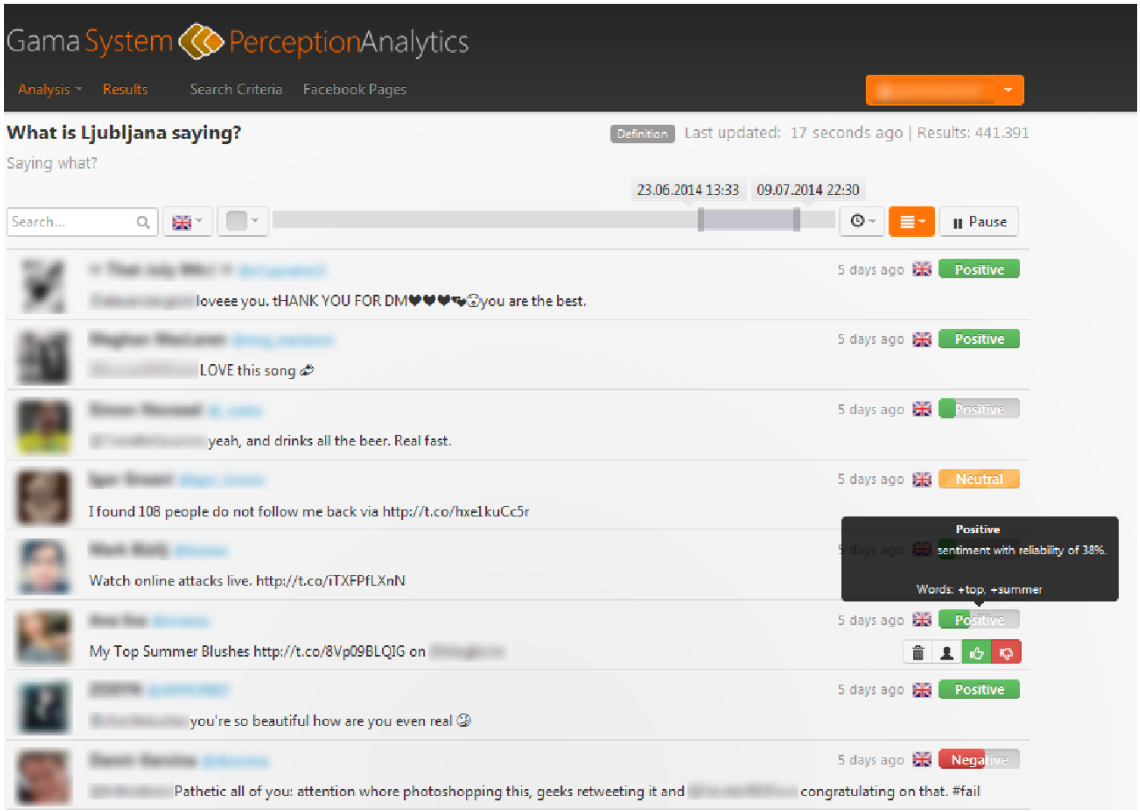

Figure 6.8: A screenshot of the PerceptionAnalytics platform showing individual tweets and the attached sentiment. Usernames and images are blurred in order to hide personal information.

on hand-labeled tweets. Furthermore, in the second experiment, we employed the concept of the neutral zone to perform the experiment in the three-class setting. Finally, in the third experiment, we trained the sentiment classifier on hand-labeled Slovene tweets and tested it in the three-class setting on a separate set of Slovene hand-labeled tweets.

The initial sentiment classifier used for the Slovenian presidential elections use case was developed using a collection of general 148,194 (142,671 positive and 5,523 negative) smiley-labeled Slovene tweets. The test set was a collection of manually labeled Slovene political tweets discussing the candidates in the first and the second round of the Slovene presidential elections, held in 2012. It consisted of 11,253 tweets manually labeled as "positive", "positive, but not sure", "negative", "negative, but not sure," or "neutral". The political tweets were labeled based on a candidate that was mentioned in the tweet and on the expressed opinion about the candidate. If the tweet was discussing more than one candidate, it was presented in the dataset as many times as there were candidates in it. For every instance of the tweet, the labeling was focused on a single candidate and the opinion about him. The tweets were preprocessed using both standard and Twitter-specific preprocessing techniques as proposed in Section 3.3.2.

In the first experiment, we trained the SVM classifier[7] on smiley-labeled tweets and

[7]Since the training data was highly unbalanced we used the SVM^{light} (Joachims, 1999) implementation of the SVM algorithm, which allows handling the unbalanced data by adjusting certain training parameters. Therefore, we trained the SVM with the parameters "-b 0 -j 0.038711441", where parameter "-b 0" means that we used unbiased hyperplane, while -j is a cost-factor which penalizes misclassified training examples and its value was the number of negative tweets divided by the number of positive tweets.

tested it in the two-class setting on 3,323 positive and 5,050 negative manually labeled Slovene political tweets. In this experiment positive tweets were the ones labeled as "positive" or "positive, but not sure" and negative tweets were the ones labeled as "negative" and "negative, but not sure". In this setting we achieved the accuracy of 49.24% on the test set.

According to the findings in Section 4.3 better results can be expected in a three-class setting by including a neutral zone. Therefore, in the second experiment, we employed also the concept of the relative neutral zone (see Section 4.1.1.2). The reliability threshold was set to 0.1. In this setting, the test set contained 3,323 positive, 5,050 negative, and 2,880 neutral manually labeled tweets. We achieved the accuracy of 35.16%.

These experimental results indicate that the sentiment classifier learned on smiley-labeled Slovene tweets is not suitable for classification of the Slovene political tweets, either in the two-class or in the three-class setting. Nevertheless, in these particular experiments, this can be a consequence of highly unbalanced training data where around 96% of data consisted of positive tweets and only a small proportion of data were negative tweets. Consequently, the sentiment classifier had knowledge about many positive terms and mostly classified test tweets as being positive. On the other hand, its vocabulary for determining the negative class was deficient and consequently it did not classify many tweets as negative, even though we used a classifier that is suitable for unbalanced data.

In the next experiment, we trained the sentiment classifier[8] by using a collection of hand-labeled Slovene tweets. For the training dataset we used 2,693 positive and 3,800 negative Slovene political tweets discussing the candidates in the first round of the elections, while for the test set we used the 630 positive, 1,250 negative, and 690 neutral Slovene political tweets discussing the candidates in the second round of the elections. In order to determine the neutral tweets, we used the relative neutral zone with the reliability threshold set to 0.1. In this experiment, we achieved the accuracy of 54.82%, which represented a considerable improvement in performance compared to the three-class sentiment classifier trained on smiley-labeled tweets and tested on manually labeled Slovene political tweets discussing the candidates in both the first and the second round of the elections. Therefore, since the hand-labeled Slovene training dataset proved to contain qualitative information about opinions on Slovene politics, we used it in the next experiment to predict the election results.

6.3.2 Analysis of the Election Results

In this section we present the experiments on predicting the outcome of the Slovenian presidential elections, which were held on November 11, 2012. The voters had a possibility to choose between the following three presidential candidates: Mr. Danilo Türk, Mr. Milan Zver, and Mr. Borut Pahor. Mr. Borut Pahor won both rounds of the elections.

Our approach to predicting the election results was based on employing the dataset of hand-labeled Slovene political tweets, which is explained in Section 6.3.1. We used the hand-labeled tweets written on October 25, November 2, and November 8, 2012 discussing the candidates in the first round in order to predict the election outcome. For every Twitter user we identified the tweets he wrote, which candidates he mentioned, and the expressed sentiment about the mentioned candidates. Based on this analysis, for every user, we determined which candidate he prefers and gave a vote to the favored candidate.[9] There were 1,483 Twitter users in the dataset. The actual valid votes from the first round, our

[8]The input parameter for the SVM was "-b 0".

[9]Note that one user can prefer more than one candidate. In that case, his vote is split with two or three and it is given to each of the preferred candidate.

Table 6.1: Results of the first round of the 2012 Slovenian presidential elections, predicted results, and number of tweets per candidate.

Candidate	Actual votes		Predicted votes		Tweet volume	
Borut Pahor	326,006	(39.87%)	600	(47.62%)	3,261	(37.56%)
Danilo Türk	293,429	(35.88%)	402	(31.90%)	2,645	(30.46%)
Milan Zver	198,337	(24.25%)	258	(20.48%)	2,777	(31.98%)
Total	817,772	(100%)	1,260	(100%)	8,683	(100%)

Table 6.2: Number of positive, negative and neutral tweets, and volume of tweets per presidential candidate of the 2012 Slovenian presidential elections.

Candidate	Positive tweets	Neutral tweets	Negative tweets	Tweet volume
Borut Pahor	1,732 (53.11%)	720 (22.08%)	809 (24.81%)	3,261 (100%)
Danilo Türk	559 (21.13%)	781 (29.53%)	1,305 (49.34%)	2,645 (100%)
Milan Zver	402 (14.48%)	689 (24.81%)	1,686 (60.71%)	2,777 (100%)

predicted results based on analysis of hand-labeled Slovene political tweets, and number of tweets per candidate are shown in Table 6.1.

As can be seen from Table 6.1, predicted votes indicated that Mr. Pahor would win the elections, followed by Mr. Türk, and Mr. Zver, which was actually the case. Namely, the first round of the presidential elections was won by Mr. Pahor, followed by Mr. Türk and Mr. Zver. Mr. Pahor and Mr. Türk proceeded to the second round. Eventually, Mr. Pahor won the elections. On the other hand, the volume of tweets per candidate showed different ranking of the candidates, i.e., it demonstrated that the most discussed candidate was Mr. Pahor, the second one was Mr. Zver, and the least discussed candidate was Mr. Türk.

Besides predicted votes and tweet volume, we also analyzed the number of manually labeled positive, negative, and neutral tweets per presidential candidate in the dataset. Table 6.2 shows the results. From the table it follows that the positive and the negative sentiment reflect the actual ranking of the candidates. Namely, Mr. Pahor received the highest number and percentage of positive tweets, followed by Mr. Türk, and Mr. Zver. On the other hand, the lowest number of negative opinions were about Mr. Pahor (in terms of number and percentage of negative tweets) followed by Mr. Türk, and Mr. Zver.

Therefore, the experimental results indicate that tweets could contain predictive information about election results. Interestingly, the first round election results were in conflict with the opinion poll predictions of the main polling agencies as their results indicated that the current Slovenian president at the time (Mr. Danilo Türk) would win the first round.

6.3.3 Social Media Analysis Platform

For the purpose of monitoring public opinion about the three Slovenian presidential candidates, a social media analysis platform was developed in collaboration with the Gama System company and the POP TV broadcasting company, which organized (among other TV stations) the TV debates of the presidential candidates. The platform was collecting and analyzing Twitter messages about the candidates (see Figure 6.9). It displayed the names and pictures of the candidates and a corresponding sentiment chart. A positive number on a chart and below the picture of a candidate meant that there were more positive than negative tweets about the candidate. A negative number meant the opposite —

that there were more negative tweets than the positive ones. The values were refreshed every few minutes. There was also a chart that presented the absolute number of positive, negative, and neutral tweets. During live TV debates on POP TV, tweets discussing the candidates were shown in the lower part of the screen and occasionally the overall aggregated results from the platform were also shown and discussed.

During the television debates on POP TV several phenomena were observed. For example, it was noticed that the number of tweets discussing the candidates was almost ten times higher than in the same time period any other day during the campaign (except for the two election days). Moreover, in comparison with the tweet volume during the debates on the Slovene national television, the tweet volume during the debates on POP TV was roughly two times higher. Also, POP TV noticed that the ratings of their shows were three to four times higher than the competitors'. These observations indicate that, nowadays, the general public can be engaged through modern technologies (like Twitter) in TV debates, by posting messages on the Internet and interactively contributing to the show. Consequently, this kind of interaction encourages the public to openly express their opinions and makes the shows and political debates more interesting and popular. On the other hand, the analyzed Twitter data can provide insights into the public opinions and emotions, which can help to predict the outcomes, possibly even better than the opinion polls.

Figure 6.9: The social media analysis platform for monitoring sentiment in Twitter messages discussing candidates of the 2012 Slovenian presidential elections.

6.4 Real-time Opinion Monitoring: Bulgarian Parliamentary Elections Use Case

This section presents the approach to real-time monitoring of public opinion expressed in Twitter messages on the Bulgarian parliamentary elections, which were held on May 12, 2013. The approach is based on the construction of a sentiment classifier from high quality manually annotated and preprocessed Twitter data, extensive evaluation, and application of the sentiment classifier in real-time on tweets discussing the elections. We also examine the results of tweet sentiment analysis and compare them with the actual outcome of the elections in order to inspect whether there exists a relationship between the two.

6.4.1 Overview of the Approach

The approach to monitoring sentiment in tweets on the parliamentary elections in Bulgaria is presented in Figure 6.10. At the highest level, the process is divided into two phases: (i) obtaining the hand-labeled training dataset and training the Twitter sentiment classifier (the top row of the figure), and (ii) applying the developed classifier to real-time Twitter data (the bottom row of the figure).

In the first phase, the main idea was to automatically learn sentiment classification models from manually annotated and preprocessed Twitter data. First, the Twitter data was collected, which, for our study, was a collection of Twitter messages written in Bulgarian language. Since Bulgarian and Macedonian languages are very similar, a language classifier was trained and applied in order to better distinguish between the tweets written in these two languages.[10] The data was manually labeled, preprocessed and finally, the sentiment classifier was trained.

In the second phase of the approach, we applied the developed sentiment classifier in real-time to each new incoming tweet from a data stream. Language of a tweet was verified, the tweet was preprocessed and classified with the sentiment classifier in real-time.

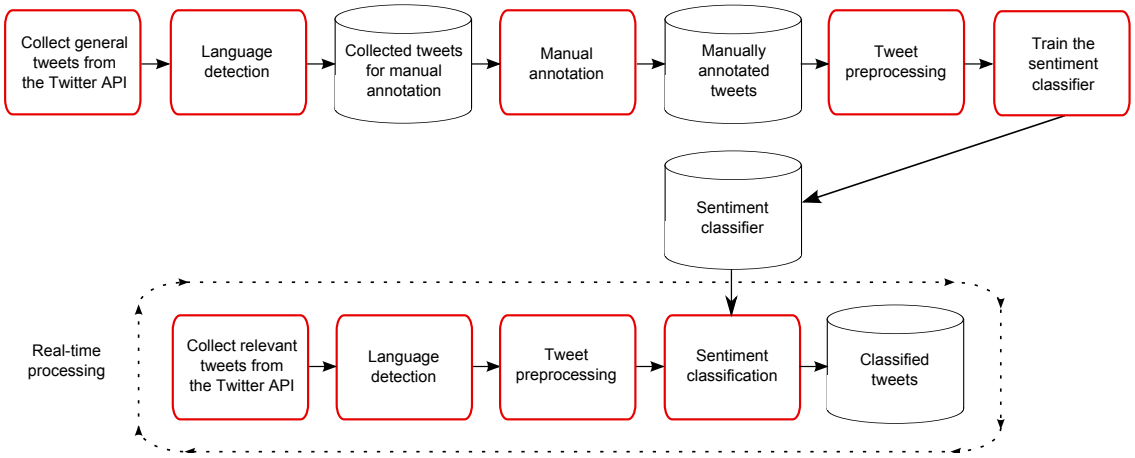

Figure 6.10: A flowchart of obtaining the hand-labeled training dataset, training the Twitter sentiment classifier, and applying it to real-time Twitter data.

[10]The language classification was developed by Janez Kranjc.

6.4.2 Twitter Sentiment Analysis

In this section, we present the developed approach to Twitter sentiment analysis in the Bulgarian parliamentary elections use case. We present the Twitter data used for the study, the algorithm used for training the sentiment classifier, preprocessing of tweets, and the evaluation experiments.

For collecting the relevant tweets, we used the PerceptionAnalytics platform and collected two sets of Twitter data:

1. General Bulgarian tweets (29,433) of the period from April 16 to April 29, 2013. The query for acquiring the tweets were the geolocations of large Bulgarian cities. These tweets were manually annotated, preprocessed and then used for training the sentiment classifiers. Moreover, they were used for the evaluation by cross validation.

2. Political Bulgarian tweets (10,300) of the period from April 29 to May 15, 2013. These tweets were the result of real-time monitoring of sentiment in political tweets before and after the Bulgarian elections, which were held on May 12, 2013. The search criteria were the names of major Bulgarian political parties and their leaders. The tweets were used in the experiments to compare their volume and sentiment to the actual results of the elections. A small subset of 90 tweets from this dataset was also manually labeled and used as a gold standard for the evaluation in the three-class classification setting.

For the purpose of training the sentiment classifier, the general Bulgarian tweets were manually annotated by twelve annotators.[11] Since we were interested in labels from the point of view of sentiment analysis, the annotators were engaged to label the tweets as being positive, negative, or neutral. They also had the possibility to exclude or skip a tweet if they, for example, considered the tweet to be inappropriate or irrelevant. If the tweets were specific to politics, even a smaller amount of tweets would have been acceptable (as observed when performing experiments for the Slovenian presidential elections case study). But, since a relatively small amount of tweets discussing politics was available before the elections, we trained the sentiment classifiers using the general Bulgarian tweets. The 29,433 general tweets were annotated in four consecutive phases. Numbers of annotated tweets for each phase is shown in Figure 6.11, while the numbers of positive, negative, and neutral tweets for each phase are shown in Figure 6.12. As can be seen, most of the annotations were performed in the first ten days. After each annotation phase, a new sentiment classifier was trained and evaluated.

For training the sentiment classifier, the training set consisted of only positive and negative tweets and a linear SVM (Boser et al., 1992; Cortes & Vapnik, 1995; Vapnik, 1995, 1998) classifier was used. The Twitter data processing was implemented using the LATINO software library, while for training the sentiment classifier, we used the SVM implementation in LATINO.

Data Preprocessing As a part of the preprocessing step, we experimented with standard text preprocessing (Feldman & Sanger, 2007) and Twitter-specific text preprocessing to better define the feature space. Standard text preprocessing included text tokenization, removal of stop words, lemmatization, and n-gram construction (we varied n from 1 to 3). The Twitter-specific preprocessing included usernames, Web links, stock symbols, exclamation and question marks transformations, and removing letter repetition. The resulting terms were used as features in the construction of the TF-IDF feature vectors

[11] For annotating the tweets the Goldfinch annotation platform was used.

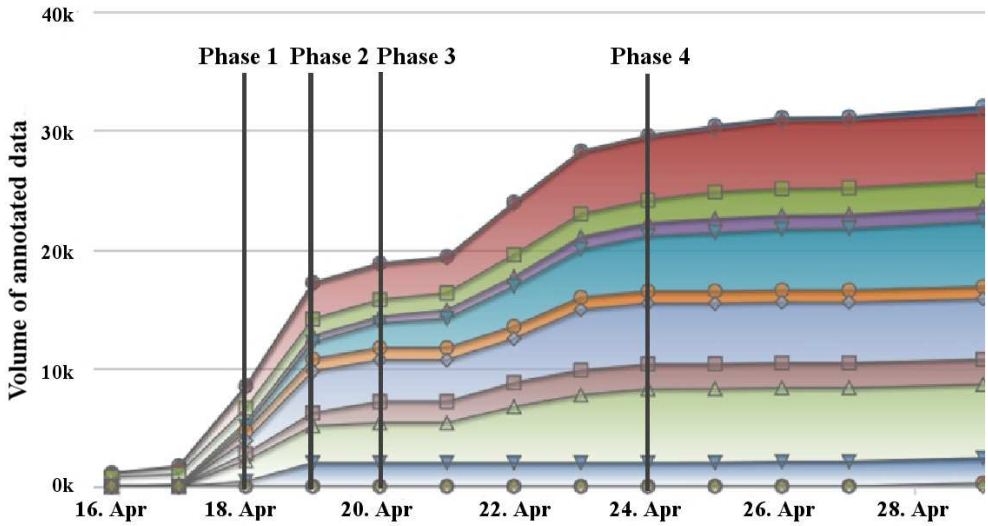

Figure 6.11: The graph from the annotation platform presenting the number of annotations of general Bulgarian tweets performed by twelve annotators between April 16 and April 29, 2013. Phases after which new sentiment classifiers were trained are additionally marked with vertical lines.

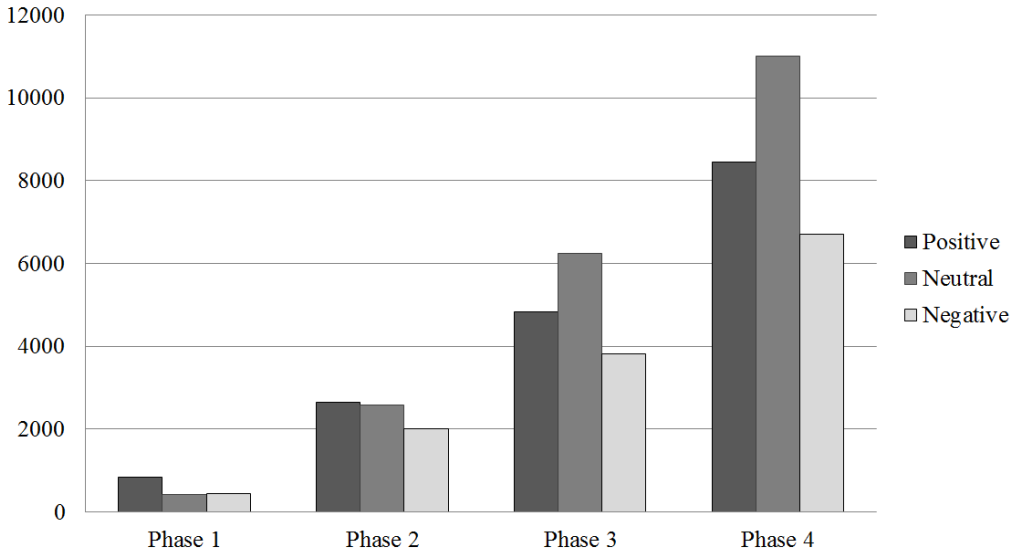

Figure 6.12: The number of positive, neutral, and negative hand-labeled general Bulgarian tweets in each annotation phase.

representing the tweets. The preprocessing experiments were performed on about 20,000 hand-labeled tweets after the third phase of the annotation process, since in this phase we succeeded to collect a reasonable amount of hand-labeled tweets for preprocessing experiments. Then, after the last annotation phase, when training the new sentiment classifier, instead of repeating the whole process of experimenting with different preprocessing settings, we applied the best preprocessing experiment from the previous (third) phase. We experimented with different combinations of preprocessing settings and the best one was determined according to the average accuracy achieved in ten-fold cross validation. It

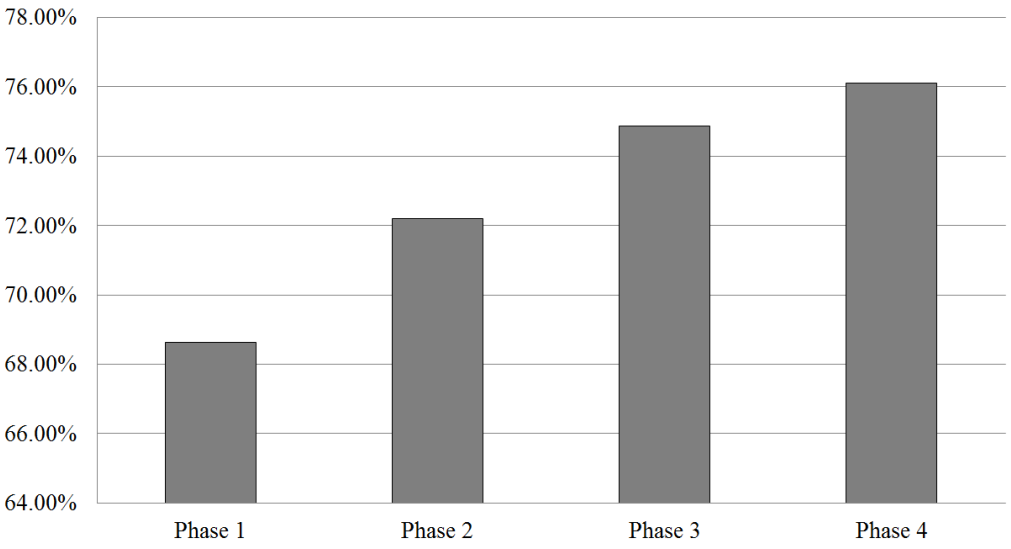

Figure 6.13: The 10-fold cross validation accuracy on positive and negative tweets after each annotation phase.

resulted in 74.88% accuracy and its characteristics were: it uses maximum n-gram length of size 2, uses terms with minimum frequency 1 in the corpus, replaces usernames with a token, replaces Web links with a token, removes stock symbols, and removes repetitive letters.[12]

Evaluation of the Models Learned from Tweets from Next Annotation Phases Using the best preprocessing setting, we calculated the ten-fold cross validation accuracy on positive and negative tweets also for the sentiment models learned in the next annotation phases in order to compare their performances. The results are shown in Figure 6.13. The highest accuracy (76.11%) was achieved in the last annotation phase where, for the classifier training, the largest number of hand-labeled tweets was available (29,433), which is an expected result. Also, it can be observed that the accuracy improved with larger training sets, but with increasingly smaller improvements. Consequently, after the fourth phase we stopped with the manual annotation of tweets.

Three-class Sentiment Classification In order to extend the binary SVM classifier by classifying tweets into three sentiment classes (positive, negative, and neutral) we used the relative neutral zone, as described in Section 4.1.1.2. A tweet was classified as neutral if the reliability of the binary classification was below a predefined threshold. In order to get an insight into which reliability threshold values were optimal, a series of experiments were conducted by changing the value of the reliability threshold from 0 to 0.5 (with an increase of 0.05), testing using the ten-fold cross validation, and plotting the receiver operating characteristic (ROC) curves, which illustrate a compromise between the benefits (true positive rate) and the costs (false positive rate) (Fawcett, 2006). For these experiments we used the complete Twitter dataset, i.e., the 29,433 annotated general

[12]Note that this preprocessing setting is different from the one used in the Slovenian presidential elections use case in Section 6.3.1, and from the preprocessing technique proposed in Section 3.3.2 of this dissertation. The reason for the differences is that for the Bulgarian parliamentary elections use case we performed separate preprocessing experiments with available hand-labeled Bulgarian Twitter data and therefore obtained a different set of preprocessing steps.

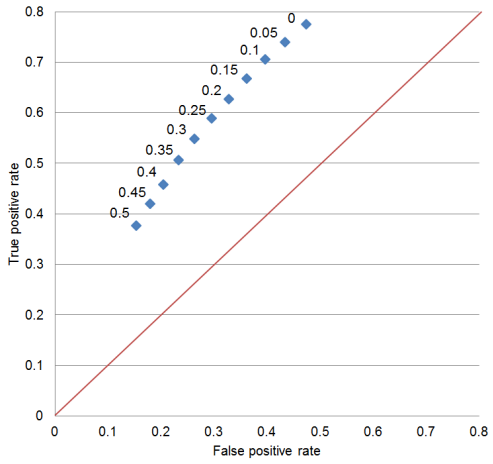

Figure 6.14: The ROC points for "positive vs. negative and neutral tweets" by varying the reliability R from 0 to 0.5.

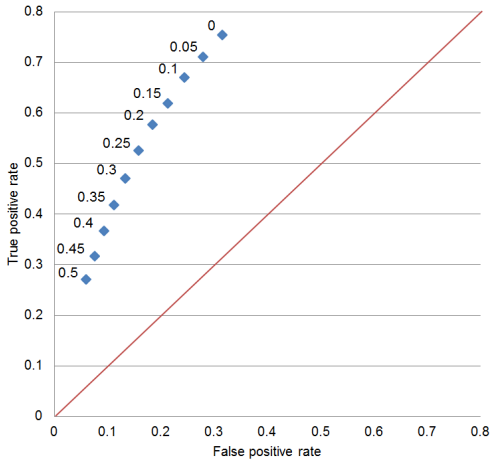

Figure 6.15: The ROC points for "negative vs. positive and neutral tweets" by varying the reliability R from 0 to 0.5.

Bulgarian tweets (8,444 positive, 6,716 negative, and 11,015 neutral). Only positive and negative tweets were used for training, but in the classification phase, the classifier had the ability to classify tweets as being positive, negative, or neutral. We were interested in the separation of positive tweets from the union of negative and neutral tweets, and on the other hand, negative tweets from the union of positive and neutral tweets. Therefore, we present the sentiment classifier performance by plotting "positive vs. negative and neutral tweets" and "negative vs. positive and neutral tweets" ROC graphs. The graphs are shown in Figures 6.14 and 6.15. The settings which achieve performances above the diagonal are better than random, where the ones closest to the left upper corner are the best. From the figures it follows that the reliability threshold $R = 0.2$ is a good choice since this value represents a good compromise between the true and false positive rate.

Evaluation on the Gold Standard Dataset We used a small dataset consisting of 90 manually labeled political tweets as a gold standard for three-class classification. First, the reliability threshold was set to $R = 0$, i.e., tweets were classified just as being positive or negative. The resulting accuracy on positive and negative tweets was 80.3%. In the second experiment, the reliability threshold was set to $R = 0.2$ and therefore the sentiment classifier was able to predict the three sentiment classes. In this setting, the three-class accuracy was 53.3% (the baseline was 33.3%). True positive rates for positive, neutral, and negative class were 0.63, 0.38, and 0.56, respectively. Detailed inspection of prediction errors showed that the classifier could not recognize sarcasm and therefore labeled such tweets with incorrect sentiment. Also, determining the sentiment was often difficult due to the unknown context of the tweet, as a consequence of writing compact and short messages. Moreover, we noticed that most of the features related to the names of the parties and politicians carried negative sentiment which might negatively bias the overall sentiment of a tweet, although the tweet contained positive or neutral opinion.

6.4.3 Analysis of Election Results

Bulgarian parliamentary elections were held on May 12, 2013. In the study, we considered only the four political parties which received more than 4% of the votes, which was a minimum percentage of votes to win a seat in the parliament. Therefore, we considered

GERB (Citizens for European Development of Bulgaria), BSP (Bulgarian Socialist Party), DPS (Movement for Rights and Freedoms), and ATAKA (Attack). We analyzed 10,300 Bulgarian political tweets, obtained from Twitter between April 29, 2013 and May 15, 2013 which discussed these four political parties and their political leaders.

For the analysis we used the sentiment classifier trained after the fourth phase of the annotation process. The training set consisted of the positive and negative general Bulgarian tweets, without the neutral ones. We applied the best preprocessing setting, and also removed all the punctuation symbols. Reliability threshold R was set to 0.2, and, therefore, all tweets with classification reliability below 0.2 were classified as being neutral. The classification results showed that the majority of tweets were classified as being negative and that the tweet volume reached its maximum on the election day. The prevailing negative sentiment in Twitter messages was probably the result of scandals, disappointment with politicians, and the discovery of 350,000 alleged illegally printed ballots one day before the elections.

Two sets of analyses were performed based on the time period: before the elections (April 29 to May 11) and after the elections (May 12 to May 15). Tweets written on May 12 were assigned to the post-elections analysis, since we noticed that already early in the day some preliminary election results (or exit polls) were published.

Pre-elections Analysis

In the first part of the analysis we considered Bulgarian political tweets which were posted before the parliamentary elections, i.e., between April 29 and May 11, 2013.

The number and the percentage of positive, neutral, and negative tweets for each political party are shown in Table 6.3. The table indicates that before the elections people expressed mostly negative opinions: for the two major parties the percentage of negative tweets was almost 75%, while the tweets discussing the DPS party were somewhat less negative, but still over 50%.

Table 6.4 presents the actual election results (as the number and proportion of parliamentary seats won by each of the analyzed parties), the volume of tweets, and differences between the negative and positive number of tweets, before the elections. As can be observed from the table, most of the tweets (63.40%) before the elections were discussing the GERB party or its leader, with a 66.69% proportion of the negative−positive tweets. GERB actually received the largest number of parliamentary seats (40.4%). Moreover, the ranking of the analyzed four parties, regarding the tweet volume and the negative−positive differences, corresponds to their ranking in terms of final election results.

Post-elections Analysis

In the second part of the analysis we considered Bulgarian political tweets which were posted between May 12 and May 15, 2013. The goal was to inspect the correlation between the election results and Twitter sentiment on the election day and a few days after the elections.

Table 6.5 shows the number and the percentage of positive, neutral, and negative tweets for each party after the elections. The negative sentiment is even higher than in the pre–elections time period.

In Table 6.6 the number and the proportion of tweets per party for the post-elections time period are shown. As in the pre-elections analysis, most of the tweets discussed the GERB political party or its leaders. However, the percentage (43.07% of the volume) is lower than before the elections (63.40%), which indicates that after the elections the discussions were more uniformly spread among the parties. Furthermore, the ranking

Table 6.3: The number and the percentage of positive, neutral, and negative tweets per party before the 2013 Bulgarian parliamentary elections.

Party	Positive tweets	Neutral tweets	Negative tweets	Volume of tweets
GERB	174 (4.72%)	771 (20.91%)	2,743 (74.38%)	3,688 (100%)
BSP	82 (5.64%)	284 (19.55%)	1,087 (74.81%)	1,453 (100%)
DPS	71 (15.01%)	158 (33.40%)	244 (51.59%)	473 (100%)
ATAKA	18 (8.87%)	62 (30.54%)	123 (60.59%)	203 (100%)

Table 6.4: Actual election results, tweet volume, and difference between the negative and positive tweets per party before the 2013 Bulgarian parliamentary elections.

Party	Parliamentary seats	Volume of tweets	Negative−Positive tweets
GERB	97 (40.4%)	3,688 (63.40%)	2,569 (66.69%)
BSP	84 (35.0%)	1,453 (24.98%)	1,005 (26.09%)
DPS	36 (15.0%)	473 (8.13%)	173 (4.49%)
ATAKA	23 (9.6%)	203 (3.49%)	105 (2.73%)
Total	240 (100%)	5,817 (100%)	3,852 (100%)

of the election results and the proportion of the parliamentary seats closely correspond to the tweet volume and the negative−positive sentiment differences. We calculated the mean absolute error (MAE) which measures how close the predictions (the proportion of tweet volume or sentiment indicator) are to the actual election results (the proportion of the parliamentary seats won). The MAE for the tweet volume was 1.88%, and for the negative−positive sentiment differences was 2.09%. In comparison, MAE for professional polling services is usually about 2-3% (Gayo-Avello et al., 2011). This leads to the conclusion that both volume and sentiment of tweets written on the election day and a few days after the elections closely corresponded to the proportion of parliament seats, with the volume correlated slightly better.

6.4.4 Public Availability of the Implemented Methodology and Results

Our approach and the sentiment analysis results for the Bulgarian elections use case have been made publicly available online. The real-time monitoring of the Bulgarian political sentiment in tweets was available in the PerceptionAnalytics platform, which is described in Section 6.2. Moreover, the results were also presented at the Bulgarian Re:Action Web page.[13] On the other hand, our implementation is available in ClowdFlows data mining platform (Kranjc et al., 2012) as a workflow which was executed in real-time during the elections. The workflow is shown in Figure 6.6. For the purposes of repeatability of this study and exploration of results, a data stream in the workflow was simulated from static uploaded data of a tweet collection written during the elections. The workflow is publicly available online at: http://clowdflows.org/workflow/1115/. Moreover, the results of the stream mining process are available at: http://clowdflows.org/streams/data/10/9691/.

[13] http://reaction.bia-bg.com.

Table 6.5: The number and the percentage of positive, neutral, and negative tweets per party after the 2013 Bulgarian parliamentary elections.

Party	Positive tweets	Neutral tweets	Negative tweets	Volume of tweets
GERB	54 (2.80%)	367 (19.01%)	1,510 (78.20%)	1,931 (100%)
BSP	33 (2.30%)	262 (18.27%)	1,139 (79.43%)	1,434 (100%)
DPS	31 (4.30%)	168 (23.30%)	522 (72.40%)	721 (100%)
ATAKA	40 (10.08%)	102 (25.69%)	255 (64.23%)	397 (100%)

Table 6.6: Election results, volume of tweets, and difference between the negative and positive tweets per party after the 2013 Bulgarian parliamentary elections.

Party	Parliamentary seats	Volume of tweets	Negative−Positive tweets
GERB	97 (40.4%)	1,931 (43.07%)	1,456 (44.55%)
BSP	84 (35.0%)	1,434 (31.99%)	1,106 (33.84%)
DPS	36 (15.0%)	721 (16.08%)	491 (15.02%)
ATAKA	23 (9.6%)	397 (8.86%)	215 (6.58%)
Total	240 (100%)	4,483 (100%)	3,268 (100%)

Chapter 7

Conclusions, Further Work, and Lessons Learned

In this chapter, we first provide a summary of the work presented in this dissertation and the conclusions we reached. Moreover, since in every study there is room for extension and improvement of the applied techniques and methodologies, we discuss several ideas for future work. Finally, we review the lessons learned while performing the research.

7.1 Conclusions

Predicting future trends, events, and phenomena is an interesting task, commonly connected to the analysis of public mood. Given that more and more people write comments, observations, and personal opinions on different Web sites and services on the Internet, various studies indicate that analysis of such texts can be automated and can produce useful results. This dissertation presents the study whose main task is to investigate whether sentiment analysis of Twitter microblogging posts (tweets) is a suitable data source for predicting future stock market values.

The performed study indicates that sentiment analysis of public mood derived from Twitter feeds could indeed be used to forecast movements of individual stock prices. The relationship was confirmed by employing a statistical test, i.e., the Granger causality test, to show whether one time series is useful in predicting the other time series. This confirms the hypothesis from Section 1.4 which states that the proposed static methodology for predictive sentiment analysis on tweet data streams is capable of predicting a (financial) phenomenon of interest. It should be noted, however, that the hypothesis is tested only on a selection of companies (see the discussion in Section 4.4).

In order to perform high quality sentiment analysis of Twitter messages, which have a specific nature, we conducted a series of experiments to determine the most suitable Twitter sentiment analysis algorithm and the best text preprocessing setting. The experiments showed that, by applying different combinations of preprocessing settings, the performance results (in terms of the accuracy and the F-measure) varied even by several percents. This confirms the hypothesis from Section 1.4 which states that appropriate selection of preprocessing steps can improve the classification accuracy.

Moreover, we presented the concept of the neutral zone, which allowed classification of tweets also into the neutral category (instead of positive and negative only). Two definitions of the neutral zone were introduced, i.e., the fixed neutral zone and the relative neutral zone. The neutral zone proved to be useful for improving the predictive power of tweets by strengthening the correlation between the tweet sentiment and the stock closing price. The correlation was especially strong when the relative neutral zone was

applied. Both results confirm the hypothesis, which states that identifying also the non-opinionated tweets improves their predictive power, in terms of forecasting stock market assets, as compared to the approach which assumes that all the tweets are opinionated and categorizes them as positive and negative only.

Furthermore, the methodology was adapted to a stream-based setting using the incremental active learning approach, which provides the algorithm with the ability to choose new training data from a data stream to be manually labeled. A number of experiments was conducted to find the best active learning approach for financial Twitter data. Definition of the (adaptive) neutral zone and the introduced stream-based active learning for sentiment analysis of microblogging messages in the financial domain contribute both to the sentiment analysis and the active learning research area and are the main contributions of this dissertation. By performing the active learning experiments, we tested the hypothesis from Section 1.4 which states that the active learning approach improves upon the static methodology (in terms of adapting the sentiment classifier to a specific domain and improving the F-measure of tweet sentiment classification) and improves its predictive power by adapting to changes in data streams. The performed experiments provided mixed conclusions regarding this hypothesis and therefore it cannot be fully confirmed. Discussion on this is provided in Section 5.4.

Selected parts of the study were made publicly available. Namely, the sentiment analysis widget, the workflow for performing sentiment analysis with active learning, and the workflow for Twitter sentiment analysis in the Bulgarian elections use case are available for public use through the ClowdFlows interactive data mining platform. Moreover, our sentiment analysis methodology was incorporated into the PerceptionAnalytics Web platform, which provides insights into happenings on popular social media Web sites.

Finally, the developed sentiment analysis methodology was successfully applied in two real-life scenarios — monitoring of sentiment in Twitter messages discussing the Slovenian and Bulgarian elections. By doing so, we confirmed the hypothesis from the Introduction (Section 1.4) which states that the developed sentiment analysis methodology is applicable in real-world applications.

7.2 Further Work

The planned future work is concerned with, on the one hand, improving the methodology for collecting, manual labeling, preprocessing, and analysis of Twitter messages, and on the other hand, advancing the application of the study in the financial domain and study the possibilities for applications in other domains.

First, we plan to take special care on collecting the tweets and their manual labeling. Namely, we intend to examine whether the approach to collecting relevant tweets from the Twitter API and composing a dataset for hand-labeling can be improved in terms of better preparing the queries for the Twitter API and additional processing of the collected data. We also plan to analyze the process of performing manual annotations and the obtained hand-labels (analyze annotation speed, annotator agreement, etc.). Preprocessing of Twitter data can also be advanced by, for example, employing a different tokenizer or varying the maximum n-gram length for features vectors construction (see Section 7.3 for further explanation of these ideas). Also, our work could benefit from using other (preferably hand-labeled) datasets in order to obtain more realistic performance estimates in tests of preprocessing settings and an even better initial sentiment classifier. Finally, we plan to improve our sentiment analysis methodology by developing techniques for the detection of irony and sarcasm (González-Ibáñez, Muresan, & Wacholder, 2011; Reyes, Rosso, & Veale, 2013).

We plan to expand the number of companies to further test our static and dynamic methodologies for sentiment analysis of microblogging messages in the financial domain. Moreover, we intend to consider several other Twitter indicators for predicting future financial assets (e.g, tweet volume, negative and neutral sentiment probability) and also several other stock market indicators (e.g., opening price and trading volume). Finally, we plan to adapt and use the sentiment analysis methodology also in other domains, such as the environmental domain for which we already started with preliminary work (Sluban, Smailović, Juršič, Mozetič, & Battiston, 2014).

7.3 Lessons Learned

In the course of the work on this dissertation we performed a number of experiments in order to select the most appropriate sentiment approach, text preprocessing setting, active learning parameters, etc. Some of the experiments provided acceptable performance results and some of them were less satisfying, but throughout this process we learned a lot, found out what are the good and bad practices, which parameters and algorithms most influence the performance results, and how to implement the proposed methods in the best way. In this section we summarize and discuss several practical issues which are concerned with the Twitter data preprocessing.

Feature Vector Construction

In our study (Smailović et al., 2014), the experiments showed that the TF-based approach for the feature vector construction is statistically significantly better than the TF-IDF approach in the Twitter sentiment classification setting. Moreover, Martineau and Finin (2009) also showed that in the sentiment classification setting, the TF-based approach performs better than the TF-IDF-based approach.

This phenomenon can be explained by taking into account the nature and aim of both approaches. The TF-based approach takes into account the number of occurrences of a term in a document, regardless of its occurrences in the whole document corpus. On the other hand, the TF-IDF approach takes into account both the number of occurrences of a term in a document and in the whole document corpus by increasing the feature value proportionally to the number of times a term is present in the document, and decreasing it with respect to the number of documents in which the term occurs. Therefore, terms which are present in many documents have lower TF-IDF feature values, and rare terms have higher TF-IDF feature values, which is beneficial for, for example, information retrieval, but for the sentiment analysis use case it may not be so useful.

Stop Word Removal

Our results in Section 3.3.2 showed that it is beneficial not to remove classical pre-compiled stop words from tweets in the sentiment classification setting. This phenomenon was also observed by Saif et al. (2012, 2014).

Moreover, this claim can be further confirmed by observing Table A.1 in Appendix A. The table presents the top most relatively positive and negative sentiment-bearing terms from the smiley-labeled Stanford dataset (Go et al., 2009) and several stop words can be found there. Therefore, although by definition stop words are words that do not carry relevant information, our results and the results of the mentioned studies indicate that in the sentiment classification setting they do carry sentiment information and should not be removed.

Tokenizer

In the experiments in Chapter 3 we employed the Regex tokenizer from the LATINO library for splitting tweets into separate elements called tokens. The Regex tokenizer is based on regular expressions and we used its default set of parameters. For example, the text "*@jenny we are buying $aapl stocks #happy ! https://www.apple.com*", without applying Twitter-specific preprocessing, would be split with the Regex tokenizer into the following tokens: <"@", "jenny", "we", "are", "buying", "$", "aapl", "stocks", "#", "happy", "!", "https", "://", "www", ".", "apple", ".", "com">. From this example it can be observed that, if the Regex tokenizer is used, the punctuation marks can be individual tokens.

Our initial experiments showed that the punctuation marks are important features in the context of sentiment analysis and therefore they should not be removed from the features. In Table A.1 in Appendix A, which presents the top most relatively positive and negative sentiment-bearing terms from the smiley-labeled Stanford dataset (Go et al., 2009), it can be observed that the most relatively positive word is actually the exclamation mark.

In order to further investigate this phenomenon, we repeated the ten-fold cross-validation experiment for the best preprocessing setting from Section 3.3.2, but using a different tokenizer, i.e., the Simple tokenizer from the LATINO library.[1] The Simple tokenizer does not take into account the punctuation marks and would split the above example into the following tokens: <"jenny", "we", "are", "buying", "aapl", "stocks", "happy", "https", "www", "apple", "com">.

In the ten-fold cross-validation experiment with the Simple tokenizer we achieved lower performance results than in the same experiment with the Regex tokenizer. Namely, with the Simple tokenizer the average value of F-measure was 0.8049 ± 0.0062 and the average accuracy was 80.48% ± 0.23%, while with the Regex tokenizer we achieved the average F-measure of 0.8143 ± 0.0046 and the average accuracy of 81.23% ± 0.16%, as presented in the first row of Table 3.3. Therefore, the initial experimental results indicate that, in the context of Twitter sentiment analysis, it is beneficial to leave the punctuation marks in the tweet feature construction.

N-Gram Construction

In the experiments for determining the best preprocessing setting (see Chapter 3) maximum n-gram length was set to 2, meaning that we constructed unigrams and bigrams for feature vectors. In order to check whether the obtained performance result can be improved by varying the length of n-grams in the feature vector construction, we repeated the ten-fold cross-validation experiments for the best preprocessing setting, where maximum n-gram length was set to 1, 2, and 3.

Classifier performance results and average number of features for two new (1 and 3) and already used (2) maximum n-gram lengths are presented in Table 7.1. As can be seen from the table, there is quite a large improvement in accuracy and F-measure between the settings with maximum n-gram length 1 and 2, which indicates that it is beneficial to construct not only unigrams, but both unigrams and bigrams for feature vectors. Naturally, the number of features was also increased, but given a considerable increase in performance, the larger number of features is acceptable. On the other hand, the improvement in accuracy and F-measure is rather small between the settings with maximum n-gram length 2 and 3, while the average number of constructed features again increased (from 1.1 million to 2.7 million). Therefore, the initial experiments showed that,

[1] The parameters for the Simple tokenizer: the tokenizer type was AlphaOnly and the minimum token length was set to 2.

Table 7.1: SVM sentiment classifier performance results and average number of features for several values of the maximum n-gram length.

Max. n-gram length	Avg. accuracy ± std. dev.	Avg. F-measure ± std. dev.	Avg. number of features ± std. dev.
1	78.66% ± 0.20%	0.7875 ± 0.0077	197,466.60 ± 189.34
2	81.23% ± 0.16%	0.8143 ± 0.0046	1,103,850.70 ± 341.16
3	81.42% ± 0.28%	0.8157 ± 0.0049	2,679,897.40 ± 785.10

in terms of performance results and the size of the feature space, the maximum n-gram length of 2 is a reasonable trade-off. As a part of the future work, we plan to expand the experiments on employing various n-gram lengths in different settings for sentiment analysis.

Appendix A

Assessment of the Smiley-Labeled Approximation

Manual data labeling is a time-consuming and expensive task, especially if the task is to label domain specific data, which requires a domain expert. Moreover, if the data to be labeled is user-generated content from the Internet, it often contains slang, many grammatical and spelling mistakes, which makes the labeling task even harder. Consequently, currently there is a lack of large publicly available hand-labeled datasets.

Instead of using the real manual labels, the labels can be approximated using the presence of positive and negative emoticons in a text. This approach was proposed by Read (2005). In our experiments, we used a large collection of 1,600,000 (800,000 positive and 800,000 negative) smiley-labeled tweets collected and prepared by Stanford University (Go et al., 2009). This simple approach causes partially correct or noisy labeling. However, in this appendix, we present empirical support for considering smiley-labeled tweets as a reasonable approximation for manually-annotated positive/negative sentiments of tweets.

Table A.1 presents the most relatively positive and negative sentiment-bearing terms from the smiley-labeled Stanford dataset after transforming mentioning of other users in a tweet in the form *@TwitterUser* to *atttTwitterUser*, transforming stock symbols in a form *$Symbol* to *stockSymbol*, replacing hash symbols in hashtags by the *hash* word, and replacing repetitive letters with more than three occurrences in a word by a word with three occurrence of such a letter. Since some terms were presented in both positive and negative tweets, we took the 1,000 terms with the highest document frequency in positive tweets, and the 1,000 terms with the highest document frequency in negative tweets and calculated the difference between document frequencies of individual terms. From the table, it follows that positive/negative smiley-labeled tweets contain numerous common positive/negative sentiment-bearing words, such as “thanks,” “love,” and “good” for positive sentiments, and “miss,” and “sad” for negative sentiments.

Interestingly, from the table it follows that writing exclamation marks and web URLs (as can be seen from the different parts of URLs presented in the table) is associated with positive emoticons, and probably positive feelings. On the other hand, it seems that writing about yourself by using personal pronouns ‘i’ or ‘my’, about work and writing question marks is associated with negative emoticons, and probably negative feelings.

To address the concern of assessment of the smiley-labeled approximation even further, we conducted an additional experiment: we manually labeled a subset of 2,000 randomly chosen tweets from the collection of 1,600,000 smiley-labeled tweets (Go et al., 2009) and computed how accurate emoticons are as labels. Their accuracy on 1,500 positive and negative manually labeled tweets was 86.40%. This result provides an error estimate and illustrates that smiley-labeled tweets are a reasonable approximation for manually-

Table A.1: The most relatively positive and negative sentiment-bearing terms from the smiley-labeled Stanford dataset.

Positive terms	Document frequency difference	Negative terms	Document frequency difference
'!'	77,648	'i'	-111,054
'you'	75,532	'go'	-74,197
'going'	55,789	't'	-61,790
'thanks'	41,668	'' t'	-61,605
'love'	36,619	'?'	-59,114
'good'	31,164	'my'	-54,526
'your'	22,638	'not'	-39,649
'://'	22,201	'"	-39,461
'http'	22,166	'to'	-38,326
'http ://'	22,141	'miss'	-37,130
'look'	21,278	'but'	-35,060
','	21,079	'sad'	-29,844
'com'	17,783	'no'	-29,739
'. com'	17,725	'...'	-29,702
'for'	17,140	'work'	-28,675

annotated positive/negative sentiment of tweets.

Furthermore, using the best preprocessing setting, as explained in Section 3.3.2, we trained a sentiment classifier on 1,600,000 smiley-labeled tweets (Go et al., 2009) and classified a collection of financial tweets, discussing the Chinese web search engine provider Baidu (the Baidu dataset is described in Section 4.2.1). With the learned SVM sentiment classifier, we classified the tweets into one of two categories (positive or negative), counted the number of positive and negative tweets for each day of the time series, and plotted them together with their difference, the moving average of the difference (averaged over five days), and the stock closing price per day. The visual presentation of the sentiment time series for the Baidu dataset can be seen in Figure A.1. The peaks show the days when people intensively tweeted about the stocks and these are indication of interesting events for a specific company. In the experiments for the visualization, we classified only the tweets whose dates corresponded to the dates when the stock market was open. The rest of the tweets (e.g., tweets which were written on weekends or non-working days of the stock market) were discarded and not analyzed. In Smailović et al. (2013) we present the visual presentation of the sentiment time series for tweets discussing two other companies: Google and Netflix.

In order to inspect how accurate the smiley-based sentiment classifier is, we calculated the F-measure on all of the hand-labeled tweets from the Baidu dataset using the sentiment classifier learned on the Stanford smiley-labeled dataset. In this experiment we achieved an F-measure of 0.5973. We have also calculated the F-measure on the positive and negative hand-labeled Baidu tweets and achieved an F-measure of 0.8032. Additionally, we employed ten-fold cross-validation experiments, using SVM on all (positive, negative and neutral) hand-labeled tweets from the Baidu dataset and achieved average F-measure value of 0.6550.

Given that the collection of 11,389 Baidu tweets was manually labeled by a domain expert (see Section 4.2.1), we were able to plot a figure also for this data (see Figure A.2) based on true positive and negative labels. Additionally, we plotted a graph (see Figure A.3) which presents the moving average of the sentiment difference (averaged over

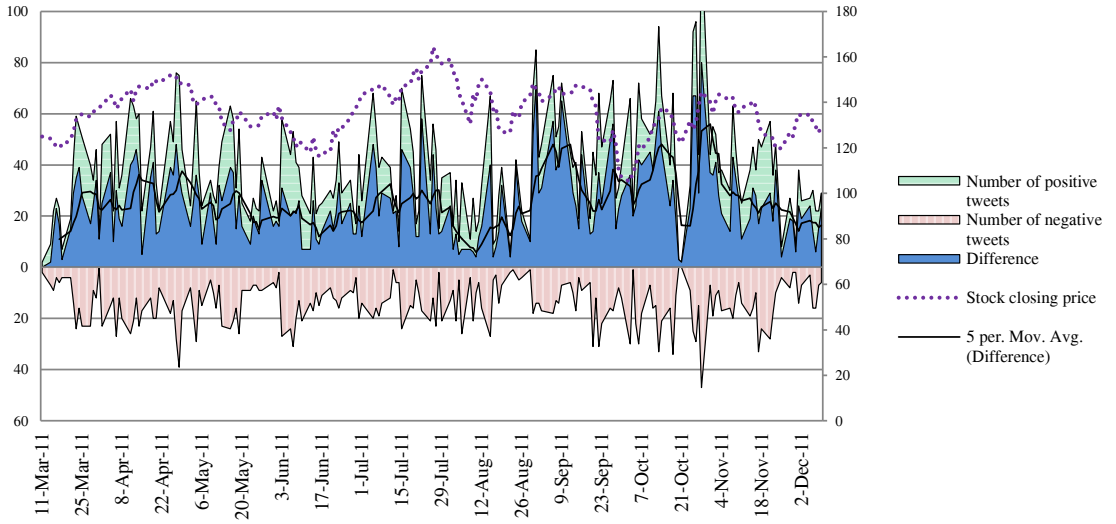

Figure A.1: Number of tweet posts classified as positive or negative, their difference, the moving average of the difference (averaged over 5 days), and the stock closing price per day for Baidu.

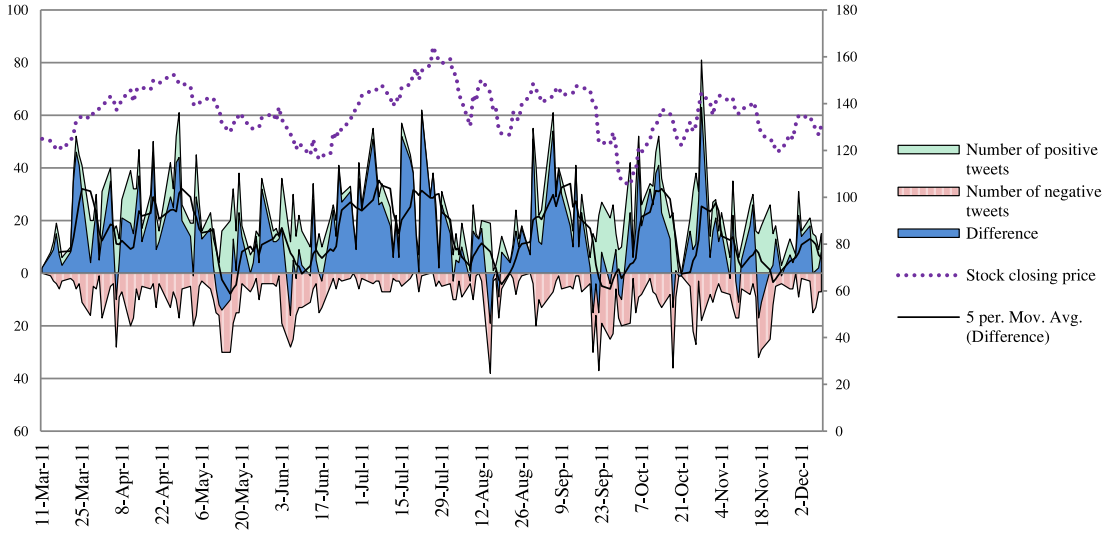

Figure A.2: Number of hand-labeled positive and negative tweet posts, their difference, the moving average of the difference (averaged over 5 days), and the stock closing price per day for Baidu.

five days) for hand-labeled positive and negative tweets and the ones classified as positive or negative by our SVM sentiment classifier. As it can be seen from the figure, the biggest peaks of differences and general trend remain basically the same, which leads to the conclusion that the performance of the classifier learned on smiley-labeled tweets is comparable to the performance of the human annotators.

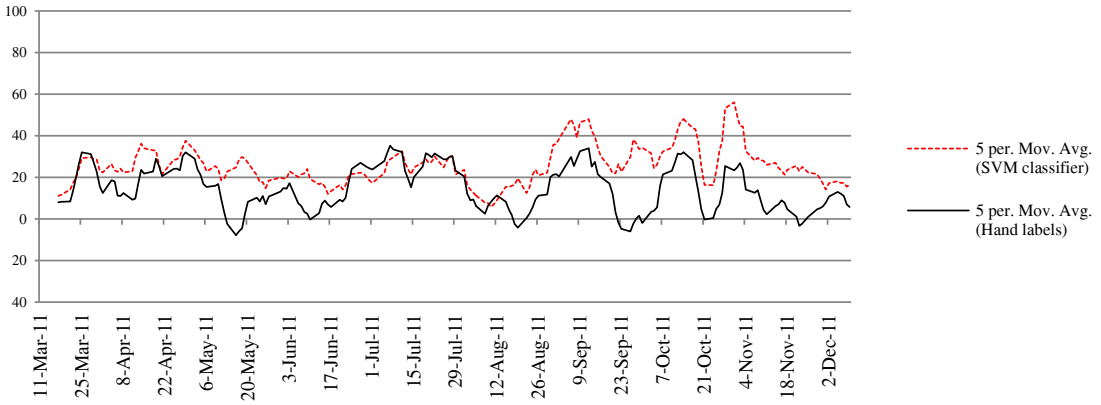

Figure A.3: The moving average of the difference (averaged over 5 days) for hand-labeled positive and negative tweets and the ones classified as positive or negative by the SVM sentiment classifier.

Appendix B

Granger Causality Correlation Between Tweet Sentiment and Stock Prices Using the Fixed Neutral Zone

This appendix reports experimental results of Granger causality correlation between daily changes of positive sentiment probability and daily returns of closing stock price for 8 companies (Apple, Amazon, Baidu, Cisco, Google, Microsoft, Netflix, and Research In Motion) using the fixed neutral zone, described in Section 4.1.1.1. Results are shown in Table C.1.

As can be seen from the table, for several companies (especially for Baidu, Microsoft, Netflix, and Research In Motion), the learned sentiment classifier has the potential to be useful for stock price prediction in terms of Granger causality.

Table B.1: Statistical significance (p-values) of Granger causality correlation between positive sentiment probability and closing stock price for 8 companies, while changing the size of the fixed neutral zone, i.e., the t value from 0 to 1. Values which are lower than a p-value of 0.1, after applying the Bonferroni correction, are marked in bold.

Size of the neutral zone		0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
Time period	Lag											
Apple												
9 months	1	0.402	0.539	0.419	0.376	0.496	0.837	0.495	0.643	0.653	0.567	0.343
Mar.-May	1	0.710	0.725	0.377	0.345	0.307	0.252	0.251	0.761	0.884	0.515	0.381
June-Aug.	1	0.302	0.415	0.564	0.650	0.979	0.689	0.837	0.877	0.872	0.790	0.866
Sept.-Nov.	1	0.786	0.772	0.943	0.657	0.495	0.641	0.324	0.334	0.219	0.174	0.108
9 months	2	0.720	0.812	0.766	0.691	0.833	0.757	0.676	0.925	0.973	0.890	0.758
Mar.-May	2	0.731	0.304	0.287	0.298	0.384	0.200	0.073	0.075	0.185	0.031	0.028
June-Aug.	2	0.406	0.433	0.555	0.907	0.980	0.841	0.924	0.722	0.600	0.441	0.395
Sept.-Nov.	2	0.946	0.813	0.882	0.888	0.782	0.858	0.622	0.661	0.502	0.404	0.272
9 months	3	0.935	0.969	0.855	0.876	0.951	0.861	0.882	0.855	0.927	0.984	0.925
Mar.-May	3	0.993	0.958	0.955	0.801	0.583	0.527	0.319	0.151	0.263	0.134	0.207
June-Aug.	3	0.737	0.788	0.817	0.985	0.919	0.802	0.898	0.727	0.583	0.436	0.446
Sept.-Nov.	3	0.986	0.907	0.888	0.882	0.779	0.854	0.680	0.627	0.517	0.531	0.297
Amazon												
9 months	1	0.193	0.619	0.571	0.564	0.685	0.678	0.524	0.649	0.775	0.778	0.941

Mar.-May	1	0.972	0.856	0.458	0.323	0.268	0.510	0.498	0.436	0.403	0.266	0.171
June-Aug.	1	0.130	0.860	0.734	0.841	0.992	0.826	0.860	0.865	0.997	0.600	0.913
Sept.-Nov.	1	0.495	0.372	0.235	0.113	0.084	0.128	0.124	0.083	0.177	0.258	0.222
9 months	2	0.386	0.877	0.661	0.562	0.788	0.908	0.804	0.893	0.958	0.937	0.836
Mar.-May	2	0.961	0.955	0.793	0.616	0.560	0.832	0.806	0.687	0.654	0.434	0.218
June-Aug.	2	0.480	0.983	0.823	0.806	0.874	0.941	0.994	0.885	0.675	0.777	0.823
Sept.-Nov.	2	0.663	0.776	0.572	0.340	0.313	0.406	0.375	0.266	0.176	0.146	0.085
9 months	3	0.558	0.974	0.836	0.766	0.886	0.922	0.864	0.969	0.992	0.986	0.949
Mar.-May	3	0.701	0.734	0.779	0.743	0.672	0.892	0.913	0.661	0.505	0.301	0.204
June-Aug.	3	0.747	0.899	0.862	0.835	0.844	0.914	0.997	0.908	0.754	0.872	0.922
Sept.-Nov.	3	0.496	0.605	0.497	0.436	0.525	0.273	0.481	0.386	0.149	0.211	0.150
Baidu												
9 months	1	0.587	0.762	0.604	0.797	0.758	0.877	0.650	0.471	0.388	0.743	0.683
Mar.-May	1	0.824	0.698	0.780	0.676	0.645	0.733	0.502	0.497	0.777	0.808	0.828
June-Aug.	1	0.995	0.865	0.812	0.863	0.920	0.347	0.514	0.622	0.585	0.347	0.368
Sept.-Nov.	1	0.298	0.452	0.555	0.576	0.620	0.540	0.335	0.216	0.140	0.288	0.251
9 months	2	0.594	0.618	0.441	0.347	0.312	0.300	0.453	0.225	0.066	0.063	**0.019**
Mar.-May	2	0.624	0.699	0.766	0.822	0.785	0.894	0.717	0.755	0.949	0.943	0.735
June-Aug.	2	0.993	0.963	0.974	0.859	0.605	0.347	0.574	0.342	0.178	0.151	0.067
Sept.-Nov.	2	**0.017**	**0.020**	**0.020**	0.037	0.054	0.068	0.082	0.039	**0.020**	0.025	**0.021**
9 months	3	0.795	0.813	0.705	0.599	0.575	0.522	0.733	0.459	0.165	0.166	0.051
Mar.-May	3	0.311	0.379	0.509	0.728	0.705	0.705	0.777	0.767	0.924	0.938	0.836
June-Aug.	3	0.915	0.684	0.924	0.791	0.574	0.309	0.456	0.405	0.255	0.195	0.106
Sept.-Nov.	3	0.026	0.035	0.039	0.075	0.080	0.095	0.160	0.077	**0.024**	0.034	**0.022**
Cisco												
9 months	1	0.131	0.105	0.193	0.093	0.106	0.080	0.200	0.093	0.070	0.034	0.036
Mar.-May	1	0.950	0.710	0.943	0.680	0.549	0.458	0.315	0.198	0.118	0.063	0.073
June-Aug.	1	0.280	0.200	0.273	0.159	0.255	0.171	0.269	0.140	0.075	0.050	0.078
Sept.-Nov.	1	0.485	0.649	0.831	0.892	0.679	0.726	0.929	0.905	0.923	0.910	0.725
9 months	2	**0.023**	**0.019**	0.086	0.049	0.051	0.068	0.182	0.150	0.149	0.137	0.144
Mar.-May	2	0.844	0.704	0.604	0.497	0.669	0.619	0.469	0.394	0.294	0.164	0.178
June-Aug.	2	0.063	0.030	0.153	0.116	0.203	0.184	0.205	0.170	0.100	0.085	0.106
Sept.-Nov.	2	0.652	0.787	0.736	0.713	0.526	0.693	0.826	0.873	0.931	0.977	0.934
9 months	3	0.051	0.050	0.165	0.104	0.096	0.156	0.320	0.259	0.296	0.237	0.250
Mar.-May	3	0.846	0.606	0.560	0.429	0.604	0.381	0.184	0.168	0.252	0.220	0.250
June-Aug.	3	0.144	0.079	0.301	0.219	0.267	0.243	0.177	0.215	0.174	0.159	0.168
Sept.-Nov.	3	0.806	0.904	0.831	0.845	0.633	0.527	0.595	0.387	0.784	0.835	0.785
Google												
9 months	1	0.287	0.422	0.321	0.607	0.951	0.970	0.975	0.636	0.589	0.468	0.220
Mar.-May	1	0.395	0.284	0.438	0.283	0.193	0.191	0.306	0.500	0.493	0.741	0.639
June-Aug.	1	0.236	0.256	0.152	0.470	0.797	0.693	0.858	0.492	0.481	0.455	0.222
Sept.-Nov.	1	0.231	0.297	0.478	0.297	0.428	0.480	0.417	0.486	0.444	0.523	0.312
9 months	2	0.636	0.757	0.713	0.920	0.900	0.943	0.967	0.864	0.777	0.627	0.456
Mar.-May	2	0.702	0.579	0.738	0.556	0.398	0.385	0.502	0.700	0.675	0.875	0.812
June-Aug.	2	0.397	0.412	0.331	0.529	0.663	0.498	0.658	0.404	0.418	0.484	0.306
Sept.-Nov.	2	0.507	0.646	0.728	0.540	0.778	0.751	0.685	0.839	0.762	0.692	0.625
9 months	3	0.832	0.854	0.730	0.872	0.696	0.549	0.712	0.634	0.660	0.718	0.584
Mar.-May	3	0.474	0.476	0.476	0.309	0.278	0.282	0.283	0.386	0.346	0.209	0.137
June-Aug.	3	0.640	0.648	0.566	0.740	0.700	0.544	0.631	0.372	0.405	0.567	0.450
Sept.-Nov.	3	0.454	0.477	0.424	0.286	0.352	0.156	0.261	0.529	0.569	0.736	0.831
Microsoft												
9 months	1	0.043	0.092	0.036	0.043	0.041	0.039	0.030	**0.023**	**0.005**	**0.006**	**0.004**
Mar.-May	1	0.563	0.609	0.516	0.339	0.194	0.217	0.128	0.115	0.067	0.068	0.096
June-Aug.	1	0.076	0.191	0.070	0.080	0.059	0.065	0.085	0.063	0.037	0.056	0.037
Sept.-Nov.	1	0.599	0.614	0.860	0.837	0.867	0.749	0.556	0.676	0.461	0.444	0.481
9 months	2	0.067	0.239	0.142	0.154	0.138	0.125	0.068	0.043	**0.008**	**0.004**	**0.004**
Mar.-May	2	0.177	0.248	0.293	0.177	0.503	0.445	0.237	0.206	0.095	0.097	0.122
June-Aug.	2	0.200	0.444	0.248	0.275	0.178	0.137	0.139	0.122	0.071	0.055	0.071
Sept.-Nov.	2	0.787	0.715	0.595	0.741	0.984	0.861	0.795	0.807	0.532	0.477	0.574
9 months	3	0.028	0.093	0.064	0.160	0.116	0.089	0.046	0.036	**0.004**	**0.002**	**0.003**
Mar.-May	3	0.268	0.321	0.152	0.065	0.057	0.055	0.049	0.051	0.029	0.132	0.214

June-Aug.	3	0.117	0.314	0.212	0.478	0.393	0.343	0.302	0.258	0.146	0.084	0.074
Sept.-Nov.	3	0.400	0.326	0.222	0.270	0.200	0.249	0.131	0.289	0.181	0.200	0.315
Netflix												
9 months	1	0.570	0.544	0.444	0.673	0.524	0.439	0.662	0.694	0.724	0.513	0.446
Mar.-May	1	0.512	0.610	0.466	0.529	0.608	0.878	0.734	0.804	0.691	0.243	0.595
June-Aug.	1	0.453	0.415	0.295	0.195	0.253	0.300	0.141	0.052	0.131	0.209	0.280
Sept.-Nov.	1	0.106	0.072	0.108	0.172	0.085	0.042	0.081	0.316	0.389	0.485	0.981
9 months	2	**0.002**	**0.006**	**0.007**	0.054	0.075	0.049	0.177	0.259	0.375	0.496	0.795
Mar.-May	2	0.233	0.192	0.328	0.213	0.257	0.643	0.632	0.597	0.524	0.469	0.608
June-Aug.	2	0.077	0.065	0.069	0.096	0.101	0.093	0.056	**0.017**	0.027	0.056	0.183
Sept.-Nov.	2	**<0.001**	**<0.001**	**<0.001**	**0.003**	**0.005**	**0.002**	0.035	0.162	0.392	0.485	0.904
9 months	3	**0.002**	**0.008**	**0.017**	0.099	0.079	0.053	0.165	0.288	0.324	0.367	0.610
Mar.-May	3	0.364	0.303	0.302	0.208	0.277	0.742	0.653	0.701	0.676	0.664	0.683
June-Aug.	3	0.124	0.158	0.198	0.218	0.245	0.248	0.161	0.066	0.106	0.194	0.394
Sept.-Nov.	3	**<0.001**	**0.001**	**0.001**	**0.005**	**0.007**	**0.003**	0.040	0.169	0.330	0.309	0.696
Research In Motion												
9 months	1	**0.005**	**0.008**	**0.019**	**0.020**	0.028	0.065	0.080	0.079	0.087	**0.025**	**0.025**
Mar.-May	1	0.344	0.373	0.252	0.245	0.374	0.336	0.258	0.203	0.165	0.173	0.115
June-Aug.	1	0.036	0.042	0.156	0.182	0.218	0.322	0.432	0.494	0.721	0.435	0.343
Sept.-Nov.	1	0.238	0.339	0.244	0.196	0.175	0.257	0.273	0.244	0.107	0.064	0.140
9 months	2	**0.019**	**0.025**	0.050	0.054	0.063	0.105	0.121	0.170	0.240	0.086	0.079
Mar.-May	2	0.697	0.766	0.513	0.569	0.494	0.594	0.439	0.473	0.462	0.483	0.338
June-Aug.	2	0.069	0.089	0.282	0.318	0.321	0.511	0.619	0.457	0.442	0.473	0.513
Sept.-Nov.	2	0.082	0.046	0.125	0.078	0.074	0.077	0.104	0.080	0.169	0.125	0.213
9 months	3	**0.001**	**0.001**	**0.006**	**0.008**	**0.011**	0.030	0.035	0.040	0.151	**0.023**	**0.024**
Mar.-May	3	0.880	0.895	0.775	0.847	0.712	0.816	0.501	0.390	0.605	0.381	0.383
June-Aug.	3	0.063	0.075	0.233	0.245	0.219	0.459	0.571	0.412	0.467	0.502	0.535
Sept.-Nov.	3	0.054	0.054	0.093	0.084	0.111	0.100	0.149	0.136	0.310	0.159	0.215

Appendix C

Granger Causality Correlation Between Tweet Sentiment and Stock Prices Using the Relative Neutral Zone

This appendix reports experimental results of Granger causality correlation between daily changes of positive sentiment probability and daily returns of closing stock price for 8 companies (Apple, Amazon, Baidu, Cisco, Google, Microsoft, Netflix, and Research In Motion) using the relative neutral zone, described in Section 4.1.1.2. Results are shown in Table C.1.

As can be seen from the table, for several companies (especially for Baidu, Cisco, Microsoft, Netflix, and Research In Motion), the learned sentiment classifier and the relative neutral zone have the potential to be useful for stock price prediction in terms of Granger causality.

Table C.1: Statistical significance (p-values) of Granger causality correlation between positive sentiment probability and closing stock price for 8 companies, while changing the reliability threshold from 0 to 1. Values which are lower than a p-value of 0.1, after applying the Bonferroni correction, are marked in bold.

Reliability threshold		0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
Time period	Lag											
Apple												
9 months	1	0.402	0.322	0.671	0.451	0.425	0.946	0.995	0.357	0.937	0.222	0.147
Mar.-May	1	0.710	0.124	0.761	0.355	0.691	0.354	0.444	0.096	0.094	0.972	0.409
June-Aug.	1	0.302	0.855	0.894	0.648	0.820	0.679	0.748	0.716	0.734	0.477	0.641
Sept.-Nov.	1	0.786	0.471	0.368	0.087	0.114	0.727	0.890	0.945	0.707	0.292	0.232
9 months	2	0.720	0.659	0.948	0.747	0.863	0.438	0.705	0.496	0.793	0.171	0.054
Mar.-May	2	0.731	0.206	0.104	0.051	0.134	0.030	0.261	0.147	0.297	0.652	0.034
June-Aug.	2	0.406	0.967	0.753	0.370	0.106	0.505	0.900	0.863	0.930	0.939	0.892
Sept.-Nov.	2	0.946	0.746	0.701	0.203	0.264	0.475	0.921	0.906	0.883	0.350	0.141
9 months	3	0.935	0.898	0.902	0.962	0.893	0.616	0.855	0.702	0.992	0.566	0.509
Mar.-May	3	0.993	0.691	0.205	0.367	0.220	0.179	0.373	0.295	0.644	0.989	0.708
June-Aug.	3	0.737	0.953	0.754	0.323	0.058	0.365	0.908	0.942	0.884	0.972	0.796
Sept.-Nov.	3	0.986	0.748	0.697	0.173	0.187	0.386	0.820	0.719	0.666	0.502	0.254
Amazon												

9 months	1	0.193	0.606	0.740	0.755	0.866	0.688	0.936	0.647	0.582	0.659	0.530
Mar.-May	1	0.972	0.310	0.467	0.199	0.115	0.345	0.655	0.126	0.751	0.248	0.790
June-Aug.	1	0.130	0.955	0.792	0.664	0.592	0.438	0.509	0.874	0.403	0.452	0.165
Sept.-Nov.	1	0.495	0.066	0.108	0.169	0.621	0.523	0.582	0.495	0.806	0.182	0.374
9 months	2	0.386	0.609	0.947	0.858	0.707	0.606	0.779	0.870	0.688	0.895	0.477
Mar.-May	2	0.961	0.609	0.728	0.259	0.211	0.456	0.606	0.354	0.965	0.270	0.460
June-Aug.	2	0.480	0.737	0.816	0.713	0.787	0.515	0.533	0.615	0.545	0.755	0.397
Sept.-Nov.	2	0.663	0.241	0.285	0.091	**0.020**	**0.004**	0.073	**0.023**	0.110	0.061	0.527
9 months	3	0.558	0.802	0.988	0.958	0.856	0.758	0.675	0.558	0.706	0.726	0.690
Mar.-May	3	0.701	0.728	0.616	0.280	0.302	0.535	0.696	0.524	0.540	0.143	0.476
June-Aug.	3	0.747	0.803	0.851	0.820	0.905	0.622	0.576	0.397	0.597	0.920	0.476
Sept.-Nov.	3	0.496	0.352	0.454	0.157	0.046	**0.008**	0.027	0.030	0.059	**0.002**	0.193
Baidu												
9 months	1	0.587	0.762	0.446	0.423	0.791	0.356	0.488	0.590	0.523	0.904	0.552
Mar.-May	1	0.824	0.778	0.476	0.720	0.471	0.760	0.910	0.920	0.917	0.928	0.667
June-Aug.	1	0.995	0.894	0.508	0.718	**0.009**	**0.022**	0.519	0.639	0.750	0.567	0.863
Sept.-Nov.	1	0.298	0.520	0.181	0.212	0.032	**0.007**	0.053	0.035	**0.016**	0.089	0.063
9 months	2	0.594	0.249	0.200	**0.012**	**0.007**	**0.003**	0.163	0.717	0.735	0.829	0.168
Mar.-May	2	0.624	0.788	0.733	0.642	0.298	0.406	0.566	0.681	0.996	0.517	0.218
June-Aug.	2	0.993	0.496	0.292	0.227	**0.014**	0.053	0.775	0.683	0.803	0.197	0.442
Sept.-Nov.	2	**0.017**	0.039	0.029	**0.010**	**<0.001**	**<0.001**	**0.001**	**0.023**	**0.017**	0.075	0.119
9 months	3	0.795	0.485	0.419	0.032	**0.023**	**0.008**	0.215	0.868	0.845	0.908	0.296
Mar.-May	3	0.311	0.652	0.770	0.762	0.507	0.492	0.738	0.647	0.989	0.478	0.382
June-Aug.	3	0.915	0.400	0.337	0.354	0.031	0.030	0.389	0.821	0.648	0.149	0.530
Sept.-Nov.	3	0.026	0.056	0.058	**0.004**	**<0.001**	**<0.001**	**0.004**	0.051	**0.003**	**0.012**	0.038
Cisco												
9 months	1	0.131	0.102	0.088	0.048	**0.018**	0.080	0.565	0.835	0.449	0.519	0.267
Mar.-May	1	0.950	0.579	0.185	0.058	0.095	0.177	0.970	0.348	0.526	0.777	0.411
June-Aug.	1	0.280	0.235	0.093	0.106	0.070	0.104	0.523	0.996	0.491	0.575	0.394
Sept.-Nov.	1	0.485	0.715	0.959	0.722	0.997	0.515	0.624	0.720	0.989	0.641	0.624
9 months	2	**0.023**	0.060	0.135	0.172	0.048	0.047	**0.016**	**0.007**	**<0.001**	**<0.001**	**0.003**
Mar.-May	2	0.844	0.690	0.409	0.233	0.556	0.645	0.303	0.354	0.376	0.833	0.565
June-Aug.	2	0.063	0.214	0.134	0.116	0.055	**0.017**	**0.003**	**<0.001**	**<0.001**	**0.001**	**0.014**
Sept.-Nov.	2	0.652	0.605	0.744	0.933	0.983	0.774	0.882	0.710	0.993	0.901	0.698
9 months	3	0.051	0.113	0.227	0.283	0.046	**0.020**	**0.006**	**0.003**	**<0.001**	**<0.001**	**0.012**
Mar.-May	3	0.846	0.597	0.172	0.285	0.823	0.930	0.647	0.337	0.345	0.504	0.172
June-Aug.	3	0.144	0.291	0.206	0.198	0.114	0.037	**0.006**	**0.001**	**0.001**	**0.006**	0.072
Sept.-Nov.	3	0.806	0.711	0.429	0.894	0.852	0.841	0.938	0.427	0.342	0.052	0.336
Google												
9 months	1	0.287	0.769	0.559	0.369	0.592	0.618	0.798	0.914	0.617	0.264	0.063
Mar.-May	1	0.395	0.253	0.499	0.665	0.824	0.374	0.550	0.963	0.719	0.805	0.896
June-Aug.	1	0.236	0.566	0.395	0.467	0.729	0.682	0.721	0.852	0.743	0.476	0.100
Sept.-Nov.	1	0.231	0.385	0.514	0.266	0.604	0.767	0.675	0.833	0.993	0.584	0.398
9 months	2	0.636	0.972	0.848	0.672	0.825	0.549	0.841	0.949	0.723	0.669	0.278
Mar.-May	2	0.702	0.496	0.672	0.770	0.947	0.622	0.822	0.863	0.622	0.642	0.203
June-Aug.	2	0.397	0.630	0.407	0.533	0.938	0.564	0.691	0.824	0.936	0.824	0.392
Sept.-Nov.	2	0.507	0.656	0.861	0.540	0.522	0.945	0.852	0.558	0.942	0.848	0.636
9 months	3	0.832	0.849	0.580	0.774	0.833	0.821	0.750	0.744	0.959	0.459	0.616
Mar.-May	3	0.474	0.229	0.351	0.183	0.303	0.485	0.573	0.995	0.643	0.255	0.189
June-Aug.	3	0.640	0.739	0.348	0.688	0.971	0.661	0.383	0.416	0.920	0.777	0.602
Sept.-Nov.	3	0.454	0.387	0.495	0.736	0.734	0.968	0.951	0.733	0.987	0.889	0.538
Microsoft												
9 months	1	0.043	0.052	**0.023**	**0.007**	**0.022**	0.046	0.776	0.603	0.689	0.747	0.568
Mar.-May	1	0.563	0.282	0.101	0.131	0.186	0.250	0.624	0.675	0.596	0.628	0.808
June-Aug.	1	0.076	0.085	0.105	0.049	0.114	0.055	0.513	0.699	0.285	0.493	0.796
Sept.-Nov.	1	0.599	0.999	0.412	0.569	0.616	0.545	0.305	0.598	0.253	0.562	0.944
9 months	2	0.067	0.213	0.038	**0.008**	**0.010**	0.040	0.303	0.667	0.789	0.306	0.368
Mar.-May	2	0.177	0.608	0.189	0.204	0.355	0.484	0.677	0.810	0.515	0.300	0.767
June-Aug.	2	0.200	0.286	0.148	0.137	0.238	0.152	0.656	0.863	0.528	0.279	0.294
Sept.-Nov.	2	0.787	0.900	0.612	0.439	0.093	0.214	0.199	0.371	0.372	0.831	0.565

9 months	3	0.028	0.189	**0.024**	**0.008**	**0.011**	0.067	0.236	0.851	0.740	0.396	0.498
Mar.-May	3	0.268	0.106	0.045	0.309	0.456	0.930	0.766	0.748	0.657	0.387	0.112
June-Aug.	3	0.117	0.509	0.268	0.220	0.344	0.276	0.613	0.962	0.619	0.346	0.345
Sept.-Nov.	3	0.400	0.211	0.194	0.275	0.190	0.367	0.292	0.658	0.598	0.944	0.616
Netflix												
9 months	1	0.570	0.625	0.750	0.437	0.122	0.411	0.363	0.702	0.545	0.465	0.103
Mar.-May	1	0.512	0.524	0.720	0.474	0.907	0.566	0.938	0.364	0.335	0.167	0.190
June-Aug.	1	0.453	0.267	0.050	0.393	0.342	0.338	0.217	0.420	0.878	0.587	0.541
Sept.-Nov.	1	0.106	0.152	0.309	0.917	0.146	0.511	0.758	0.822	0.388	0.260	0.103
9 months	2	**0.002**	0.062	0.306	0.894	0.035	0.191	0.444	0.634	0.575	0.388	0.069
Mar.-May	2	0.233	0.154	0.537	0.490	0.251	0.502	0.550	0.651	0.448	0.314	0.258
June-Aug.	2	0.077	0.109	**0.016**	0.212	0.400	0.565	0.513	0.577	0.964	0.424	0.108
Sept.-Nov.	2	**<0.001**	**0.003**	0.208	0.974	**0.004**	0.028	0.186	0.099	0.050	0.239	0.247
9 months	3	**0.002**	0.087	0.283	0.704	0.087	0.321	0.628	0.404	0.488	0.594	0.149
Mar.-May	3	0.364	0.154	0.649	0.628	0.181	0.436	0.678	0.664	0.612	0.443	0.354
June-Aug.	3	0.124	0.227	0.060	0.455	0.659	0.495	0.634	0.318	0.989	0.630	0.194
Sept.-Nov.	3	**<0.001**	**0.005**	0.200	0.823	**0.013**	0.056	0.294	0.072	0.057	0.395	0.355
Research In Motion												
9 months	1	**0.005**	**0.022**	0.067	**0.012**	**0.010**	0.042	**0.006**	0.115	0.215	0.407	0.044
Mar.-May	1	0.344	0.285	0.257	**0.051**	**0.007**	0.057	**0.014**	0.041	0.116	0.378	0.926
June-Aug.	1	0.036	0.181	0.604	0.264	0.191	0.239	0.071	0.299	0.504	0.939	**0.023**
Sept.-Nov.	1	0.238	0.212	0.095	0.147	0.392	0.728	0.602	0.607	0.833	0.377	0.251
9 months	2	**0.019**	0.056	0.190	0.043	**0.023**	0.077	**0.024**	0.248	0.454	0.704	0.136
Mar.-May	2	0.697	0.471	0.583	0.178	0.033	0.170	0.054	0.125	0.219	0.530	0.282
June-Aug.	2	0.069	0.276	0.446	0.481	0.346	0.472	0.118	0.396	0.626	0.929	0.060
Sept.-Nov.	2	0.082	0.098	0.062	0.240	0.075	0.365	0.371	0.822	0.957	0.687	0.453
9 months	3	**0.001**	**0.004**	0.074	0.031	**0.015**	0.098	0.038	0.296	0.586	0.813	0.285
Mar.-May	3	0.880	0.683	0.714	0.345	0.082	0.231	0.057	0.197	0.256	0.569	0.411
June-Aug.	3	0.063	0.155	0.377	0.572	0.409	0.456	0.147	0.575	0.682	0.961	0.139
Sept.-Nov.	3	0.054	0.109	0.135	0.277	0.101	0.444	0.523	0.781	0.980	0.818	0.494

References

Abdi, H. (2007). Bonferroni and Šidák corrections for multiple comparisons. In N. Salkind (Ed.), *Encyclopedia of Measurement and Statistics* (pp. 103–107). Thousand Oaks (CA): Sage.

Agarwal, A., Xie, B., Vovsha, I., Rambow, O., & Passonneau, R. (2011). Sentiment analysis of Twitter data. In *Proceedings of the Workshop on Languages in Social Media* (pp. 30–38). Association for Computational Linguistics.

Antweiler, W. & Frank, M. Z. (2004). Is all that talk just noise? The information content of internet stock message boards. *The Journal of Finance*, *59*(3), 1259–1294.

Argamon-Engelson, S. & Dagan, I. (1999). Committee-based sample selection for probabilistic classifiers. *Journal of Artificial Intelligence Research*, *11*, 335–360.

Asur, S. & Huberman, B. A. (2010). Predicting the future with social media. In *Proceedings of the 2010 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT)* (Vol. 1, pp. 492–499).

Bachelier, L. (1900). *Théorie de la spéculation*. Gauthier-Villars.

Bermingham, A. & Smeaton, A. (2011). On using Twitter to monitor political sentiment and predict election results. In *Proceedings of the Workshop on Sentiment Analysis where AI meets Psychology (SAAIP)* (pp. 2–10). Chiang Mai, Thailand: Asian Federation of Natural Language Processing.

Bifet, A. & Frank, E. (2010). Sentiment knowledge discovery in Twitter streaming data. In *Proceedings of the 13th International Conference on Discovery Science (DS)* (pp. 1–15). Springer.

Bifet, A., Holmes, G., Kirkby, R., & Pfahringer, B. (2010). MOA: Massive Online Analysis. *The Journal of Machine Learning Research*, *11*, 1601–1604.

Bifet, A., Holmes, G., & Pfahringer, B. (2011). MOA-TweetReader: Real-time analysis in Twitter streaming data. In *Discovery Science* (Vol. 6926, pp. 46–60). Lecture Notes in Computer Science. Springer.

Bifet, A. & Kirkby, R. (2009). Data stream mining: A practical approach.

Bollen, J., Mao, H., & Pepe, A. (2011). Modeling public mood and emotion: Twitter sentiment and socio-economic phenomena. In *Proceedings of the 5th International AAAI Conference on Weblogs and Social Media (ICWSM)*.

Bollen, J., Mao, H., & Zeng, X. (2011). Twitter mood predicts the stock market. *Journal of Computational Science*, *2*(1), 1–8.

Borondo, J., Morales, A., Losada, J., & Benito, R. (2012). Characterizing and modeling an electoral campaign in the context of Twitter: 2011 Spanish presidential election as a case study. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, *22*(2), 023138.

Boser, B. E., Guyon, I. M., & Vapnik, V. N. (1992). A training algorithm for optimal margin classifiers. In *Proceedings of the 5th Annual Workshop on Computational Learning Theory* (pp. 144–152). ACM.

Broder, A. Z., Glassman, S. C., Manasse, M. S., & Zweig, G. (1997). Syntactic clustering of the web. *Computer Networks and ISDN Systems*, *29*(8), 1157–1166.

Butler, K. C. & Malaikah, S. J. (1992). Efficiency and inefficiency in thinly traded stock markets: Kuwait and saudi arabia. *Journal of Banking & Finance*, *16*(1), 197–210.

Caldarelli, G., Chessa, A., Pammolli, F., Pompa, G., Puliga, M., Riccaboni, M., & Riotta, G. (2014). A multi-level geographical study of Italian political elections from Twitter data. *PloS one*, *9*(5), e95809.

Chang, C.-C. & Lin, C.-J. (2011). LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, *2*(3), 27.

Chen, R. & Lazer, M. (2011). Sentiment analysis of Twitter feeds for the prediction of stock market movement. *CS 229 Machine Learning : Final Project*.

Chu, W., Zinkevich, M., Li, L., Thomas, A., & Tseng, B. (2011). Unbiased online active learning in data streams. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 195–203). ACM.

Chung, J. E. & Mustafaraj, E. (2011). Can collective sentiment expressed on Twitter predict political elections? In *Proceedings of the 25th AAAI Conference on Artificial Intelligence*.

Cohn, D. A., Ghahramani, Z., & Jordan, M. I. (1996). Active learning with statistical models. *arXiv preprint cs/9603104*.

Cortes, C. & Vapnik, V. (1995). Support-vector networks. *Machine Learning*, *20*(3), 273–297.

Cover, T. & Hart, P. E. (1967). Nearest neighbor pattern classification. *Information Theory, IEEE Transactions on*, *13*(1), 21–27.

Damasio, A. (1995). *Descartes error: Emotion, reason, and the human brain*. Harper Perennial.

Demšar, J. (2006). Statistical comparisons of classifiers over multiple data sets. *The Journal of Machine Learning Research*, *7*, 1–30.

DiGrazia, J., McKelvey, K., Bollen, J., & Rojas, F. (2013). More tweets, more votes: Social media as a quantitative indicator of political behavior. *PloS one*, *8*(11), e79449.

Donmez, P., Carbonell, J. G., & Bennett, P. N. (2007). Dual strategy active learning. In *Proceedings of the 18th European Conference on Machine Learning (ECML)* (pp. 116–127). Springer.

Duda, R. O. & Hart, P. E. (1973). *Pattern classification and scene analysis*. Wiley New York.

Fama, E. F. (1965a). Random walks in stock market prices. *Financial Analysts Journal*, 55–59.

Fama, E. F. (1965b). The behavior of stock-market prices. *Journal of Business*, 34–105.

Fawcett, T. (2006). An introduction to ROC analysis. *Pattern Recognition Letters*, *27*(8), 861–874.

Feldman, R. & Sanger, J. (2007). *The text mining handbook: Advanced approaches in analyzing unstructured data*. Cambridge University Press.

Flach, P. (2012). *Machine learning: The art and science of algorithms that make sense of data*. Cambridge University Press.

Freund, Y., Seung, H. S., Shamir, E., & Tishby, N. (1997). Selective sampling using the query by committee algorithm. *Machine Learning*, *28*(2-3), 133–168.

Friedman, M. (1937). The use of ranks to avoid the assumption of normality implicit in the analysis of variance. *Journal of the American Statistical Association*, *32*(200), 675–701.

Friedman, M. (1940). A comparison of alternative tests of significance for the problem of m rankings. *The Annals of Mathematical Statistics*, *11*(1), 86–92.

Friedman, N., Geiger, D., & Goldszmidt, M. (1997). Bayesian network classifiers. *Machine Learning*, *29*(2-3), 131–163.

Gaber, M. M., Zaslavsky, A., & Krishnaswamy, S. (2005). Mining data streams: A review. *ACM Sigmod Record*, *34*(2), 18–26.

Gama, J. (2010). *Knowledge discovery from data streams*. Chapman & Hall/CRC Press.

Gama, J. & Gaber, M. M. (2007). *Learning from data streams*. Springer-Verlag Berlin Heidelberg.

Gayo-Avello, D. (2011). Don't turn social media into another 'Literary Digest' poll. *Communications of the ACM*, *54*(10), 121–128.

Gayo-Avello, D. (2012). "I wanted to predict elections with Twitter and all i got was this lousy paper"–a balanced survey on election prediction using Twitter data. *arXiv preprint arXiv:1204.6441*.

Gayo-Avello, D. (2013). A meta-analysis of state-of-the-art electoral prediction from Twitter data. *Social Science Computer Review*.

Gayo-Avello, D., Metaxas, P. T., & Mustafaraj, E. (2011). Limits of electoral predictions using Twitter. In *Proceedings of the International Conference on Weblogs and Social Media (ICWSM) 2011*.

Gerber, M. S. (2014). Predicting crime using Twitter and kernel density estimation. *Decision Support Systems*, *61*, 115–125.

Go, A., Bhayani, R., & Huang, L. (2009). Twitter sentiment classification using distant supervision. *CS224N Project Report, Stanford*, 1–12.

González-Ibáñez, R., Muresan, S., & Wacholder, N. (2011). Identifying sarcasm in Twitter: A closer look. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers-Volume 2* (pp. 581–586). Association for Computational Linguistics.

Granger, C. W. (1969). Investigating causal relations by econometric models and cross-spectral methods. *Econometrica: Journal of the Econometric Society*, 424–438.

Gruhl, D., Guha, R., Kumar, R., Novak, J., & Tomkins, A. (2005). The predictive power of online chatter. In *Proceedings of the 11th ACM SIGKDD International Conference on Knowledge Discovery in Data Mining* (pp. 78–87). ACM.

Guyon, I., Weston, J., Barnhill, S., & Vapnik, V. (2002). Gene selection for cancer classification using support vector machines. *Machine Learning*, *46*(1-3), 389–422.

Gwet, K. (2001). Handbook of inter-rater reliability. *Gaithersburg, MD: STATAXIS Publishing Company*, 223–246.

Haddi, E., Liu, X., & Shi, Y. (2013). The role of text pre-processing in sentiment analysis. *Procedia Computer Science*, *17*, 26–32.

Hoi, S. C., Jin, R., Zhu, J., & Lyu, M. R. (2006). Batch mode active learning and its application to medical image classification. In *Proceedings of the 23rd International Conference on Machine Learning* (pp. 417–424). ACM.

Hu, M. & Liu, B. (2004). Mining and summarizing customer reviews. In *Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 168–177). ACM.

Hubbard, D. W. (2011). *Pulse: The new science of harnessing internet buzz to track threats and opportunities*. John Wiley & Sons.

Ikonomovska, E. (2012). *Algorithms for learning regression trees and ensembles on evolving data streams* (Doctoral dissertation, Jožef Stefan International Postgraduate School, Ljubljana, Slovenija).

Ikonomovska, E., Gama, J., & Džeroski, S. (2011). Learning model trees from evolving data streams. *Data Mining and Knowledge Discovery*, *23*(1), 128–168.

Iman, R. L. & Davenport, J. M. (1980). Approximations of the critical region of the friedman statistic. *Communications in Statistics-Theory and Methods*, *9*(6), 571–595.

Jahanbakhsh, K. & Moon, Y. (2014). The predictive power of social media: On the predictability of US presidential elections using Twitter. *arXiv preprint arXiv:1407.0622*.

Jelles, F., Van Bennekom, C. A., Lankhorst, G. J., Sibbel, C. J., & Bouter, L. M. (1995). Inter-and intra-rater agreement of the rehabilitation activities profile. *Journal of Clinical Epidemiology*, *48*(3), 407–416.

Joachims, T. (1998). Text categorization with support vector machines: Learning with many relevant features. In *Proceedings of the European Conference on Machine Learning* (pp. 137–142).

Joachims, T. (1999). Making large-scale SVM learning practical. In B. Schölkopf, C. Burges, & A. Smola (Eds.), *Advances in kernel methods - support vector learning* (Chap. 11, pp. 169–184). Cambridge, MA: MIT Press.

Joachims, T. (2005). A support vector method for multivariate performance measures. In *Proceedings of the 22nd International Conference on Machine Learning* (pp. 377–384). ACM.

Joachims, T. (2006). Training linear SVMs in linear time. In *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 217–226). ACM.

Joachims, T. & Yu, C.-N. J. (2009). Sparse kernel SVMs via cutting-plane training. *Machine Learning*, *76*(2-3), 179–193.

Jungherr, A. (2013). Tweets and votes, a special relationship: The 2009 federal election in Germany. In *Proceedings of the 2nd workshop on Politics, Elections and Data* (pp. 5–14). ACM.

Jungherr, A., Jürgens, P., & Schoen, H. (2012). Why the pirate party won the German election of 2009 or the trouble with predictions: A response to Tumasjan, A., Sprenger, T. O., Sander, P. G., & Welpe, I. M. "Predicting elections with Twitter: What 140 characters reveal about political sentiment". *Social Science Computer Review*, *30*(2), 229–234.

Kavussanos, M. G. & Dockery, E. (2001). A multivariate test for stock market efficiency: The case of ASE. *Applied Financial Economics*, *11*(5), 573–579.

Kohavi, R. (1995). A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence (IJCAI) - Volume 2* (pp. 1137–1143).

Kouloumpis, E., Wilson, T., & Moore, J. (2011). Twitter sentiment analysis: The good the bad and the OMG! In *Proceedings of The International Conference on Weblogs and Social Media (ICWSM)* (Vol. 11, pp. 538–541).

Kranjc, J., Podpečan, V., & Lavrač, N. (2012). Clowdflows: A cloud based scientific workflow platform. In *Machine Learning and Knowledge Discovery in Databases* (pp. 816–819). Springer.

Kranjc, J., Podpečan, V., & Lavrač, N. (2013). Real-time data analysis in ClowdFlows. In *Proceedings of The 2013 IEEE International Conference on Big Data* (pp. 15–22). IEEE.

Kranjc, J., Smailović, J., Podpečan, V., Grčar, M., Žnidaršič, M., & Lavrač, N. (2014). Active learning for sentiment analysis on data streams: Methodology and workflow implementation in the ClowdFlows platform. *Information Processing & Management*. doi:http://dx.doi.org/10.1016/j.ipm.2014.04.001

Krempl, G., Žliobaitė, I., Brzeziński, D., Hüllermeier, E., Last, M., Lemaire, V., ... Stefanowski, J. (2014). Open challenges for data stream mining research. *ACM SIGKDD Explorations Newsletter*, *16*(1), 1–10.

Lewis, D. D. & Gale, W. A. (1994). A sequential algorithm for training text classifiers. In *Proceedings of the ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 3–12). Springer-Verlag New York, Inc.

Liu, B. (2010). Sentiment analysis and subjectivity. *Handbook of Natural Language Processing*, *2*, 627–666.

Liu, B. (2012). Sentiment analysis and opinion mining. *Synthesis Lectures on Human Language Technologies*, *5*(1), 1–167.

Liu, B., Hu, M., & Cheng, J. (2005). Opinion observer: Analyzing and comparing opinions on the web. In *Proceedings of the 14th International Conference on World Wide Web* (pp. 342–351). ACM.

Liu, Y. & Bahadori, M. T. (2014). A survey on Granger causality.

Livne, A., Simmons, M. P., Adar, E., & Adamic, L. A. (2011). The party is over here: Structure and content in the 2010 election. In *Proceedings of the 5th International AAAI Conference on Weblogs and Social Media*.

Martineau, J. & Finin, T. (2009). Delta TFIDF: An improved feature space for sentiment analysis. In *Proceedings of the Third AAAI Internatonal Conference on Weblogs and Social Media*.

Medhat, W., Hassan, A., & Korashy, H. (2014). Sentiment analysis algorithms and applications: A survey. *Ain Shams Engineering Journal*.

Metaxas, P. T., Mustafaraj, E., & Gayo-Avello, D. (2011). How (not) to predict elections. In *2011 IEEE 3rd International Conference on Privacy, Security, Risk and Trust (PASSAT) and 2011 IEEE 3rd International Conference on Social Computing (SocialCom)* (pp. 165–171). IEEE.

Mishne, G. & Glance, N. S. (2006). Predicting movie sales from blogger sentiment. In *AAAI Spring Symposium: Computational Approaches to Analyzing Weblogs* (pp. 155–158).

Mittal, A. & Goel, A. (2012). Stock prediction using Twitter sentiment analysis. *Standford University, CS229 (http:// cs229.stanford.edu/ proj2011/ GoelMittal-StockMarket PredictionUsingTwitterSentimentAnalysis.pdf)*.

Nann, S., Krauss, J., & Schoder, D. (2013). Predictive analytics on public data - the case of stock markets. In *Proceeding of 21st European Conference on Information Systems* (paper 116).

Nemenyi, P. B. (1963). *Distribution-free multiple comparisons* (Doctoral dissertation, Princeton University).

Nofsinger, J. R. (2005). Social mood and financial economics. *The Journal of Behavioral Finance*, *6*(3), 144–160.

Nowak, S. & Rüger, S. (2010). How reliable are annotations via crowdsourcing: a study about inter-annotator agreement for multi-label image annotation. In *Proceedings of the International Conference on Multimedia Information Retrieval* (pp. 557–566). ACM.

O'Connor, B., Balasubramanyan, R., Routledge, B. R., & Smith, N. A. (2010). From tweets to polls: linking text sentiment to public opinion time series. *Proceedings of the International AAAI Conference on Weblogs and Social Media*, *11*, 122–129.

Ogneva, M. (2010). How companies can use sentiment analysis to improve their business. Retrieved June 26, 2014, from http://mashable.com/2010/04/19/sentiment-analysis/

Oh, C. & Sheng, O. (2011). Investigating predictive power of stock micro blog sentiment in forecasting future stock price directional movement. In *Proceedings of the International Conference on Information Systems*.

Pang, B. & Lee, L. (2008). Opinion mining and sentiment analysis. *Foundations and Trends in Information Retrieval*, *2*(1-2), 1–135.

Pang, B., Lee, L., & Vaithyanathan, S. (2002). Thumbs up? Sentiment classification using machine learning techniques. In *Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing - Volume 10* (pp. 79–86). Association for Computational Linguistics.

Parikh, R. & Movassate, M. (2009). Sentiment analysis of user-generated Twitter updates using various classification techniques. *CS224N Final Report*, 1–18.

Petz, G., Karpowicz, M., Fürschuß, H., Auinger, A., Stříteský, V., & Holzinger, A. (2013). Opinion mining on the web 2.0–characteristics of user generated content and their impacts. In *Human-Computer Interaction and Knowledge Discovery in Complex, Unstructured, Big Data* (pp. 35–46). Springer.

Petz, G., Karpowicz, M., Fürschuß, H., Auinger, A., Winkler, S. M., Schaller, S., & Holzinger, A. (2012). On text preprocessing for opinion mining outside of laboratory environments. In *Active Media Technology* (pp. 618–629). Springer.

Rao, T. & Srivastava, S. (2014). Twitter sentiment analysis: How to hedge your bets in the stock markets. In *State of the Art Applications of Social Network Analysis* (pp. 227–247). Springer.

Rao, Y., Li, Q., Mao, X., & Wenyin, L. (2014). Sentiment topic models for social emotion mining. *Information Sciences*.

Read, J. (2005). Using emoticons to reduce dependency in machine learning techniques for sentiment classification. In *Proceedings of the ACL Student Research Workshop* (pp. 43–48). Association for Computational Linguistics.

Regnault, J. (1863). *Calcul des chances et philosophie de la bourse*. librairie Castel.

Reyes, A., Rosso, P., & Veale, T. (2013). A multidimensional approach for detecting irony in Twitter. *Language Resources and Evaluation*, *47*(1), 239–268.

Romero, D. M., Galuba, W., Asur, S., & Huberman, B. A. (2011). Influence and passivity in social media. In *Machine Learning and Knowledge Discovery in Databases* (pp. 18–33). Springer.

Roy, N. & McCallum, A. (2001). Toward optimal active learning through monte carlo estimation of error reduction. In *Proceedings of the international conference on machine learning (icml)*.

Ruiz, E. J., Hristidis, V., Castillo, C., Gionis, A., & Jaimes, A. (2012). Correlating financial time series with micro-blogging activity. In *Proceedings of the 5th ACM International Conference on Web Search and Data Mining* (pp. 513–522). ACM.

Saif, H., Fernandez, M., He, Y., & Alani, H. (2014). On stopwords, filtering and data sparsity for sentiment analysis of Twitter. In *Proceedings of the 9th International Conference on Language Resources and Evaluation (LREC)* (pp. 810–817).

Saif, H., He, Y., & Alani, H. (2012). Semantic sentiment analysis of Twitter. In *The Semantic Web–ISWC 2012* (Vol. 7649, pp. 508–524). Lecture Notes in Computer Science. Springer.

Sang, E. T. K. & Bos, J. (2012). Predicting the 2011 Dutch Senate election results with Twitter. In *Proceedings of the Workshop on Semantic Analysis in Social Media* (pp. 53–60). Association for Computational Linguistics.

Saveski, M. & Grčar, M. (2011). Web services for stream mining: A stream-based active learning use case. *Proceedings of the PlanSoKD Workshop at ECML PKDD 2011*, 36.

Sculley, D. (2007). Online active learning methods for fast label-efficient spam filtering. In *Proceedings of the 4th Conference on Email and AntiSpam (CEAS)*.

Sculley, D. (2009). Large scale learning to rank. In *NIPS 2009 Workshop on Advances in Ranking* (pp. 1–6).

Sculley, D. (2010a). Combined regression and ranking. In *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 979–988). ACM.

Sculley, D. (2010b). Web-scale k-means clustering. In *Proceedings of the 19th International Conference on World Wide Web* (pp. 1177–1178). ACM.

Sebastiani, F. (2002). Machine learning in automated text categorization. *ACM computing surveys (CSUR)*, *34*(1), 1–47.

Sedghi, A. (2014). Facebook: 10 years of social networking, in numbers. Retrieved August 6, 2014, from http://www.theguardian.com/news/datablog/2014/feb/04/facebook-in-numbers-statistics

Sehgal, V. & Song, C. (2007). SOPS: Stock prediction using web sentiment. In *Proceedings of the 7th IEEE International Conference on Data Mining Workshops* (pp. 21–26). IEEE.

Seth, A. (2007). Granger causality. *Scholarpedia*, *2*(7), 1667.

Settles, B. (2008). *Curious machines: Active learning with structured instances*. ProQuest.

Settles, B. (2009). *Active learning literature survey* (Computer Sciences Technical Report No. 1648). University of Wisconsin–Madison.

Settles, B. (2011a). Closing the loop: Fast, interactive semi-supervised annotation with queries on features and instances. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing* (pp. 1467–1478). Association for Computational Linguistics.

Settles, B. (2011b). From theories to queries: Active learning in practice. (pp. 1–18).

Settles, B. & Craven, M. (2008). An analysis of active learning strategies for sequence labeling tasks. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing* (pp. 1070–1079). Association for Computational Linguistics.

Settles, B., Craven, M., & Ray, S. (2008). Multiple-instance active learning. In *Advances in Neural Information Processing Systems* (pp. 1289–1296).

Seung, H. S., Opper, M., & Sompolinsky, H. (1992). Query by committee. In *Proceedings of the 5th Annual Workshop on Computational Learning Theory* (pp. 287–294). ACM.

Shalev-Shwartz, S., Singer, Y., & Srebro, N. (2007). Pegasos: Primal estimated sub-gradient solver for SVM. In *Proceedings of the 24th International Conference on Machine Learning* (pp. 807–814).

Shearer, C. (2000). The CRISP-DM model: The new blueprint for data mining. *Journal of Data Warehousing*, *5*(4), 13–22.

Skoric, M., Poor, N., Achananuparp, P., Lim, E.-P., & Jiang, J. (2012). Tweets and votes: A study of the 2011 Singapore general election. In *Proceedings of the 45th Hawaii International Conference on System Science (HICSS)* (pp. 2583–2591). IEEE.

Sluban, B., Smailović, J., Juršič, M., Mozetič, I., & Battiston, S. (2014). Community sentiment on environmental topics in social networks. In *Proceedings of the 10th International Conference on Signal Image Technology & Internet Based Systems (SITIS), 3rd International Workshop on Complex Networks and their Applications* (pp. 376–382).

Smailović, J., Grčar, M., Lavrač, N., & Žnidaršič, M. (2013). Predictive sentiment analysis of tweets: A stock market application. In *Human-Computer Interaction and Knowledge Discovery in Complex, Unstructured, Big Data* (pp. 77–88). Lecture Notes in Computer Science Volume 7947. Springer Berlin Heidelberg.

Smailović, J., Grčar, M., Lavrač, N., & Žnidaršič, M. (2014). Stream-based active learning for sentiment analysis in the financial domain. *Information Sciences*, *285*, 181–203. [IF = 3.893].

Smailović, J., Grčar, M., & Žnidaršič, M. (2012). Sentiment analysis on tweets in a financial domain. In *Proceedings of 4th Jožef Stefan International Postgraduate School Students Conference* (pp. 169–175).

Smailović, J., Žnidaršič, M., & Grčar, M. (2011). Web-based experimental platform for sentiment analysis. In *Proceedings of the 3rd International Conference on Information Society and Information Technologies (ISIT)*.

Smith, C. (2014). By the numbers: 215 amazing Twitter statistics. Retrieved August 6, 2014, from http://expandedramblings.com/index.php/march-2013-by-the-numbers-a-few-amazing-twitter-stats/#.U-H40GOftkI

Sprenger, T. O., Tumasjan, A., Sandner, P. G., & Welpe, I. M. (2013). Tweets and trades: The information content of stock microblogs. *European Financial Management*.

Sul, H., Dennis, A. R., & Yuan, L. I. (2014). Trading on Twitter: The financial information content of emotion in social media. In *47th Hawaii International Conference on System Sciences (HICSS)* (pp. 806–815). IEEE.

Taboada, M., Brooke, J., Tofiloski, M., Voll, K., & Stede, M. (2011). Lexicon-based methods for sentiment analysis. *Computational Linguistics*, *37*(2), 267–307.

Thelwall, M., Buckley, K., & Paltoglou, G. (2011). Sentiment in Twitter events. *Journal of the American Society for Information Science and Technology*, *62*(2), 406–418.

Thelwall, M., Buckley, K., Paltoglou, G., Cai, D., & Kappas, A. (2010). Sentiment strength detection in short informal text. *Journal of the American Society for Information Science and Technology*, *61*(12), 2544–2558.

Thet, T. T., Na, J.-C., Khoo, C. S., & Shakthikumar, S. (2009). Sentiment analysis of movie reviews on discussion boards using a linguistic approach. In *Proceedings of the 1st International CIKM Workshop on Topic-Sentiment Analysis for Mass Opinion* (pp. 81–84). ACM.

Tong, R. M. (2001). An operational system for detecting and tracking opinions in on-line discussion. In *Working Notes of the ACM SIGIR 2001 Workshop on Operational Text Classification* (Vol. 1, p. 6).

Tsytsarau, M. & Palpanas, T. (2012). Survey on mining subjective data on the web. *Data Mining and Knowledge Discovery*, *24*(3), 478–514.

Tumasjan, A., Sprenger, T. O., Sandner, P. G., & Welpe, I. M. (2010a). Election forecasts with Twitter: How 140 characters reflect the political landscape. *Social Science Computer Review*.

Tumasjan, A., Sprenger, T. O., Sandner, P. G., & Welpe, I. M. (2010b). Predicting elections with Twitter: What 140 characters reveal about political sentiment. *Proceedings of the 4th International AAAI Conference on Weblogs and Social Media*, *10*, 178–185.

Tumasjan, A., Sprenger, T. O., Sandner, P. G., & Welpe, I. M. (2012). Where there is a sea there are pirates: Response to Jungherr, Jürgens, and Schoen. *Social Science Computer Review*, *30*(2), 235–239.

Turney, P. D. (2002). 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* (pp. 417–424). Association for Computational Linguistics.

Van Rijsbergen, C. J. (1974). Foundation of evaluation. *Journal of Documentation*, *30*(4), 365–373.

Vapnik, V. (1995). *The nature of statistical learning theory*. Springer.

Vapnik, V. (1998). *Statistical learning theory*. Wiley.

Wang, H., Can, D., Kazemzadeh, A., Bar, F., & Narayanan, S. (2012). A system for real-time Twitter sentiment analysis of 2012 U.S. presidential election cycle. In *Proceedings of the ACL 2012 System Demonstrations* (pp. 115–120). Association for Computational Linguistics.

Wang, P., Zhang, P., & Guo, L. (2012). Mining multi-label data streams using ensemble-based active learning. In *Proceedings of SDM* (pp. 1131–1140). SIAM.

Wessa, P. (2013). Bivariate Granger Causality (v1.0.3) in Free Statistics Software (v1.1.23-r7), Office for Research Development and Education. Retrieved April 10, 2014, from http://www.wessa.net/rwasp_grangercausality.wasp/

Wilson, T., Wiebe, J., & Hwa, R. (2006). Recognizing strong and weak opinion clauses. *Computational Intelligence*, *22*(2), 73–99.

Xu, Z., Yu, K., Tresp, V., Xu, X., & Wang, J. (2003). *Representative sampling for text classification using support vector machines*. Springer.

Yang, Y. & Liu, X. (1999). A re-examination of text categorization methods. In *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 42–49). ACM.

Yu, Y., Duan, W., & Cao, Q. (2013). The impact of social and conventional media on firm equity value: A sentiment analysis approach. *Decision Support Systems*, *55*(4), 919–926.

Zhang, P., Zhu, X., & Guo, L. (2009). Mining data streams with labeled and unlabeled training examples. In *Proceedings of the 9th IEEE International Conference on Data Mining (ICDM'09)* (pp. 627–636). IEEE.

Zhang, X., Fuehres, H., & Gloor, P. A. (2011). Predicting stock market indicators through Twitter "I hope it is not as bad as I fear". *Procedia-Social and Behavioral Sciences*, *26*, 55–62.

Zheludev, I., Smith, R., & Aste, T. (2014). When can social media lead financial markets? *Scientific Reports*, *4*.

Zhu, X., Zhang, P., Lin, X., & Shi, Y. (2007). Active learning from data streams. In *Proceedings of the 7th International Conference on Data Mining (ICDM)* (pp. 757–762). IEEE.

Zhu, X., Zhang, P., Lin, X., & Shi, Y. (2010). Active learning from stream data using optimal weight classifier ensemble. *IEEE Transactions on Systems, Man, and Cybernetics:Part B*, *40*(6), 1607–1621.

Žliobaitė, I., Bifet, A., Pfahringer, B., & Holmes, G. (2011). Active learning with evolving streaming data. In *Machine Learning and Knowledge Discovery in Databases* (pp. 597–612). Springer.

Žliobaitė, I., Bifet, A., Pfahringer, B., & Holmes, G. (2014). Active learning with drifting streaming data. *IEEE Transactions on Neural Networks and Learning Systems*, *25*(1), 27–39.

Bibliography

Publications Related to the Thesis

Original scientific articles

Kranjc, J., Smailović, J., Podpečan, V., Grčar, M., Žnidaršič, M., & Lavrač, N. (2014). Active learning for sentiment analysis on data streams: Methodology and workflow implementation in the ClowdFlows platform. *Information Processing & Management*. doi:http://dx.doi.org/10.1016/j.ipm.2014.04.001

Smailović, J., Grčar, M., Lavrač, N., & Žnidaršič, M. (2013). Predictive sentiment analysis of tweets: A stock market application. In *Human-Computer Interaction and Knowledge Discovery in Complex, Unstructured, Big Data* (pp. 77–88). Lecture Notes in Computer Science Volume 7947. Springer Berlin Heidelberg.

Smailović, J., Grčar, M., Lavrač, N., & Žnidaršič, M. (2014). Stream-based active learning for sentiment analysis in the financial domain. *Information Sciences*, *285*, 181–203. [IF = 3.893].

Conference papers

Sluban, B., Smailović, J., Juršič, M., Mozetič, I., & Battiston, S. (2014). Community sentiment on environmental topics in social networks. In *Proceedings of the 10th International Conference on Signal Image Technology & Internet Based Systems (SITIS), 3rd International Workshop on Complex Networks and their Applications* (pp. 376–382).

Smailović, J., Grčar, M., & Žnidaršič, M. (2012). Sentiment analysis on tweets in a financial domain. In *Proceedings of 4th Jožef Stefan International Postgraduate School Students Conference* (pp. 169–175).

Smailović, J., Žnidaršič, M., & Grčar, M. (2011). Web-based experimental platform for sentiment analysis. In *Proceedings of the 3rd International Conference on Information Society and Information Technologies (ISIT)*.

Biography

Jasmina Smailović was born on May 22, 1986 in Banja Luka, Bosnia and Herzegovina. In 2009, she graduated in Computer Science and Information Technologies at the Faculty of Electrical Engineering, University in Banja Luka. During the study, she received four awards for the best student of the generation at the faculty, and after graduation she received the award for the best diploma paper of the Computer Science and Information Technologies program in the academic year 2009/2010, the award for the best graduate engineer at the Faculty of Electrical Engineering in the academic year 2009/2010, and the gold plaque from the University in Banja Luka for the achieved success at the faculty and completing the study on time. In 2009 and 2010 Jasmina worked as a software engineer at ComTrade Group Banja Luka. In 2010, she started her doctoral studies at the Jožef Stefan International Postgraduate School in Ljubljana, Slovenia. She enrolled in the Information and Communication Technologies PhD program under the supervision of Asst. Prof. Dr. Martin Žnidaršič and Prof. Dr. Nada Lavrač. Her research concerns sentiment analysis of social media contents in stream-based environments. Jasmina was involved in European projects FIRST, FOC, SIMPOL, MULTIPLEX, and WHIM of the 7th Framework Programme for Research and Technological Development. At the 4th Jožef Stefan International Postgraduate School Student Conference she received the award for the best student paper in the field of Information and Communication Technologies. She also successfully collaborated the Gama System company in the development of the sentiment analytics for various languages.