UNIVERSIDAD DE CHILE
FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN

ANÁLISIS ESTÁTICO Y DINÁMICO DE OPINIONES EN TWITTER

TESIS PARA OPTAR AL GRADO DE
MAGÍSTER EN CIENCIAS, MENCIÓN COMPUTACIÓN

FELIPE JOSÉ BRAVO MÁRQUEZ

PROFESOR GUÍA:
BÁRBARA POBLETE LABRA
PROFESOR CO-GUÍA:
MARCELO MENDOZA ROCHA

MIEMBROS DE LA COMISIÓN :
CLAUDIO GUTIÉRREZ GALLARDO
ROMAIN ROBBES
DIEGO ARROYUELO BILLIARDI

SANTIAGO, CHILE
AGOSTO 2013

Este trabajo fue parcialmente financiado por la beca de magíster CONICYT y por el proyecto FONDEF D09I1185

RESUMEN DE LA TESIS
PARA OPTAR AL GRADO DE
MAGÍSTER EN CIENCIAS,
MENCIÓN COMPUTACIÓN
POR : FELIPE BRAVO MÁRQUEZ
FECHA: 26/08/2013
PROF. GUÍA: BÁRBARA POBLETE

Resumen

Los medios de comunicación social y en particular las plataformas de Microblogging se han consolidado como un espacio para el consumo y producción de información. Twitter se ha vuelto una de las plataforma más populares de este estilo y hoy en día tiene millones de usuarios que diariamente publican millones de mensajes personales o "twiits". Una parte importante de estos mensajes corresponden a opiniones personales, cuya riqueza y volumen ofrecen una gran oportunidad para el estudio de la opinión pública. Para tabajar con este alto volumen de opiniones digitales, se utilizan un conjunto de herramientas computacionales conocidas como métodos de análisis de sentimiento o minería de opinión.

La utilidad de evaluar la opinión pública usando análisis de sentimiento sobre opiniones digitales genera controversia en la comunidad científica. Mientras diversos trabajos declaran que este enfoque permite capturar la opinión pública de una manera similar a medios tradicionales como las encuestas, otros trabajos declaran que este poder esta sobrevalorado. En este contexto, estudiamos el comportamiento estático y dinámico de las opiniones digitales para comprender su naturaleza y determinar las limitaciones de predecir su evolución en el tiempo.

En una primera etapa se estudia el problema de identificar de manera automática los tuits que expresan una opinión, para luego inferir si es que esa opinión tiene una connotación positiva o negativa. Se propone una metodología para mejorar la clasificación de sentimiento en Twitter usando atributos basados en distintas dimensiones de sentimiento. Se combinan aspectos como la intensidad de opinión, la emoción y la polaridad, a partir de distintos métodos y recursos existentes para el análisis de sentimiento. La investigación muestra que la combinación de distintas dimensiones de opinión permite mejorar significativamente las tareas de clasificación de sentimientos en Twitter de detección de subjetividad y de polaridad.

En la segunda parte del análisis se exploran las propiedades temporales de las opiniones en Twitter mediante el análisis de series temporales de opinión. La idea principal es determinar si es que las series temporales de opinión pueden ser usadas para crear modelos predictivos confiables. Se recuperan en el tiempo mensajes emitidos en Twitter asociados a un grupo definido de tópicos. Luego se calculan indicadores de opinión usando métodos de análisis de sentimiento para luego agregarlos en el tiempo y construir series temporales de opinión. El estudio se basa en modelos ARMA/ARIMA y GARCH para modelar la media y la volatilidad de las series. Se realiza un análisis profundo de las propiedades estadísticas de las series temporales encontrando que éstas presentan propiedades de estacionalidad y volatilidad. Como la volatilidad se relaciona con la incertidumbre, se postula que estas series no debiesen ser usadas para realizar pronósticos en el largo plazo.

Los resultados experimentales obtenidos permiten concluir que las opiniones son objetos multidimensionales, donde las distintas dimensiones pueden complementarse para mejorar la clasificación de sentimiento. Por otro lado, podemos decir que las series temporales de opinión deben cumplir con ciertas propiedades estadísticas para poder realizar pronósticos confiables a partir de ellas. Dado que aún no hay suficiente evidencia para validar el supuesto poder predictivo de las opiniones digitales, nuestros resultados indican que una validación más rigurosa de los modelos estáticos y dinámicos que se constuyen a partir de estas opiniones permiten establecer de mejor manera los alcances de la minería de opinión.

Agradecimientos

Al concluir este trabajo se cierra un largo período de más de 10 años en la Facultad de Ciencias Físicas y Matemáticas de la Universidad de Chile. Un período donde viví muchas experiencias de las cuales me gustaría destacar las que me entregaron mayores aprendizajes; haber pasado por dos Departamentos, haber trabajado en distintos grupos de investigación, haber asistido a varios cursos interesantes, haber participado en congresos internacionales, haber ejercido la docencia tanto como profesor auxiliar como profesor de cátedra y lejos lo más importante, haber conocido a muchas personas valiosas.

Claramente, mi vida no comienza ni termina en Beauchef. Existe otro grupo de personas muy valiosas para mí que no tienen nada que ver con mi historia en la Facultad. Quiero dedicar este trabajo y agradecer a todo este conjunto de personas "valiosas" que con sus cualidades y curiosidades me han nutrido para ser quien soy.

Comienzo por mi madre, gracias por tu calidez y disposición. Tus palabras y acciones han sido luz infinita en momentos oscuros. A mi padre, que por medio del ejemplo me ha enseñado a ser perseverante y a querer el trabajo propio. Sin tu apoyo y tus consejos me hubiese retirado a mitad de camino. A mis hermanos. Gracias Cristóbal por motivarme desde chico a desafiarme intelectualmente y a encontrar "mi verdad". Gracias Cristián por ser como eres, una persona sencilla y buena de alma. Al resto de mi familia cercana. En particular quiero agradecer a mi abuelo Norman, mi abuela Ruth, mi primos Sebastián Bravo y Álvaro Márquez y mi cuñada María de los Ángeles.

A Constanza Lledó por haber aparecido en mi vida. Por aceptarme y quererme tal cual soy. Por ayudarme a tomar las cosas con más calma y ser más tolerante a la frustración. Todo sería mucho más aburrido y carente de sentido sin tu compañía. Construir una vida juntos es definitivamente la tarea más desafiante y emocionante para el futuro. Agradezco también a toda la familia de Constanza por haberme hecho sentir todos estos años como en casa.

A los grandes amigos de la vida: Alex Klotz, Patricia Wagner, Germán Johannsen, Pedro Gil, Rodrigo Born, María Ignacia Magofke, Christian Schüler, Nicolás Saul, Alan Breinbauer, Cristóbal Peters, Tomás Vega, Pablo Vega, Andrés Schmidt-Hebbel, Renato Hermosilla, Roberto Piel, Sabina García, Sebastián Martinez, Gian Paolo Gritti, Mathias Klapp, Matías Melkonian, Víctor Riveros, Ernesto Bentjerodt y Juan Cristóbal Bartelsman. A José Antonio Gil por haberme enseñado a tocar guitarra. Ese instrumento de madera se ha convertido en un fiel compañero en tiempos de ocio y relajación.

A los buenos amigos de Beauchef: Thomas Booth, Daniel Sabbatino, David Bustos, Mauricio Solervicens, Marco Ruiz, Marcos Himmer y Cristian Ludwig.

Me disculpo de todas las personas que de alguna manera me han aportado y olvidé mencionar. De verdad se me hizo imposible agregarlos a todos. A todas las personas con las que he compartido una cerveza, una buena charla, un paseo, o cantado una canción, quiero que sepan que ese tipo de compañía es lo que me hace más feliz.

Entrando en el plano académico quiero comenzar agradeciendo a las personas que ayudaron a darme cuenta que la investigación era lo mío en mis últimos años de pregrado. Muchas gracias a todo el grupo Web Intelligence y al proyecto DOCODE del departamento

de Ingeniería Industrial. Gracias Sebastián Ríos por haber creído en mí sin conocerme y haberme dado el espacio para aprender a desarrollar mis propias ideas. Infinitas gracias a Gastón L'huillier por haberme ayudado a concretar mis primeros artículos científicos, por tu amistad y tu inmensa calidad humana. Gracias a Juan Velásquez por haberme dado el espacio de dictar mis primeras clases y por todos los momentos de camadería entregados. Probablemente no hubiese tenido la inquietud de continuar desarrollándome en el plano académico si no hubiese podido saborear la investigación en este grupo. Otras personas muy valiosas con las que me tocó compartir del mismo grupo son: Gabriel Oberreuter, Patricio Moya y Edison Maresse Taylor. ¡Gracias a ellos también!

Mi segunda fase en el mundo de la investigación comienza el año 2011 al ingresar al programa de magíster en el Departamento de Ciencias de la Computación (DCC) trabajando en el laboratorio de investigación Yahoo! Labs Latin America.

Quiero agradecer a mis dos profesores guía Marcelo Mendoza y Bárbara Poblete. A Marcelo por el tiempo dedicado en plantear el problema, las discusiones técnicas y por mostrarme una manera de investigar centrada en las ideas. A Bárbara, por ayudarme a estructurar las ideas, por los múltiples consejos y correcciones en la escritura de los artículos y por la disponibilidad de ayudarme y orientarme cuando lo requerí.

Existe una persona en el DCC que siempre tuvo la puerta de su oficina abierta. Muchas gracias Claudio Gutiérrez por el espacio de reflexión, por todas las conversaciones y por los consejos entregados. Otras personas del DCC a las que me gustaría agradecer son Angélica Aguirre, Nelson Baloian, Benjamín Bustos y a la gente del grupo PRISMA.

Agradezco a Mauricio Marín por haberme invitado a desarrollar mi tesis en las dependencias de Yahoo! Labs. Valoro además de Mauricio el haberme incentivado a apuntar mi trabajo a estándares internacionales. Destaco además el grato ambiente de trabajo en Yahoo! donde agradezco a: Pablo Torres, Emir Muñoz, Carlos Gomez, Alonso Inostrosa, Carolina Bonacic, Manuel Manriquez, Juan Zamora, Alejandro Figueroa, Diego Arroyuelo, Roberto Solar, Sara Quiñones, Adreas Eiselt, Francia Jiménez, Claudio Sanhueza y Rodrigo Scheihing.

Agradezco la cooperación de Daniel Gayo-Avello de la Universidad de Oviedo, a los miembros de la comisión por sus valiosos comentarios, a Renato Cerro por haber revisado el inglés de la tesis y a todos los revisores de los artículos enviados cuyos comentarios anónimos permitieron estructurar el trabajo de una mejor manera.

Para finalizar, agradezco el financiamiento de la beca de magíster CONICYT y del proyecto FONDEF D09I1185 titulado "Observatorios Escalables de la Web en Tiempo Real" que hicieron este trabajo posible.

Felipe Bravo Márquez
26 de Agosto 2013
Santiago, Chile

Abstract

Social media, and in particular Microblogging platforms have opened new possibilities for people to consume and produce information. Twitter has become one of the most popular microblogging platforms and has millions of users which daily spread millions of personal posts or tweets. An important part of the messages propagated through social media correspond to personal opinions. The rich and enormous volume of these opinions offers great opportunities for the study of public opinion. Due to the volume of this data, social media opinions are being studied using a set of computational tools referred to as opinion mining or sentiment analysis techniques.

The usefulness of assessing public opinion from social media using opinion mining methods is a controversial topic among researchers. As several works claim that the approach is able to capture the public opinion in a similar manner than traditional polls, other works state that this power is greatly exaggerated. In this context, we study the static and dynamic behavior of social media opinions aimed to understand the nature of them, and also to determine the limitations of predicting how they evolve in time.

In the first stage we study the static properties of Twitter opinions using sentiment analysis tools. The main goals of this analysis are to detect automatically tweets that express an opinion, and to infer whether these opinionated tweets have a positive or negative connotation. We propose a methodology for boosting Twitter sentiment classification using different sentiment dimensions as input features. We combine aspects such as opinion strength, emotion and polarity indicators, generated by existing sentiment analysis methods and resources. Our research shows that the combination of different sentiment dimensions provides significant improvement in Twitter sentiment classification tasks such as polarity and subjectivity.

In the second stage we explore the temporal properties of opinions or "opinion dynamics" by analyzing opinion time series created from Twitter data. The focus of this step is to establish whether an opinion time series is or not appropriate for generating a reliable predictive model. Twitter messages related to certain topics are tracked through time and evaluated using sentiment analysis methods. Afterwards, evaluated tweets are aggregated over time and used to build opinion time series. Our study rely on ARMA/ARIMA and GARCH models to assess the conditional mean and the volatility of the process respectively. We present an in-depth analysis of the statistical properties of the time series finding that they present important seasonality and volatility factors. As volatility is related to uncertainty, we state that these time series should not be used for long-term forecasting purposes.

The experimental results allow us to conclude, on the one hand, that social media opinions are multidimensional objects, in which the different dimensions complement each other for sentiment classification purposes. On the other hand, we can say that there are certain statistical conditions that opinion time series must satisfy in order to derive accurate forecasts from them. Due to the fact that researchers have not provided enough evidence to support the alleged predictive power of social media opinions, our results indicate that a more rigorous validation of static and dynamic models generated from this data could benefit the opinion mining field.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

When people are exposed to information regarding an event, they normally respond to this external stimuli by acquiring an orientation or personal opinion. Public opinion is defined as the aggregation of individual attitudes or beliefs held by the adult population about specific subjects [Wik11b]. An adequate understanding of public opinion in relation to different subjects or entities has long been a topic of interest for several institutions. For example, private companies conduct public opinion surveys to know preferences for products and brands in their marketing studies. Likewise, public institutions and research centers conduct public opinion surveys to measure the approval rating of a president or political party.

Public opinion has been traditionally evaluated using polls. An opinion poll is defined as a survey of public opinion from a particular population sample [Wik11a]. Polls are composed by a set of questions related to the subject to be evaluated and results obtained from them are used to extrapolate generalities of the target population. Nevertheless, this approach has several drawbacks which are presented below:

- Polls need to be conducted periodically in order to track a target topic over time.
- Polls are unable to detect instantaneous changes in public opinion [ABDF10].
- It is hard to evaluate a set of different topics in a single poll.
- The way that polls are worded and ordered can influence the response of the respondents and therefore bias the overall evaluation (cf. [SP96]).

Social media platforms and, in particular, microblogging services such as **Twitter**[1], **Tumblr**[2], and **Weibo**[3] are increasingly being adopted by people in order to access and publish information about a great variety of topics. These new mediums of expression enable people to connect to each other, and voice their opinion in a

[1] `http://www.twitter.com`
[2] `http://www.twitter.com`
[3] `http://www.weibo.com`

simple manner. Twitter in particular, is a service in which users post messages or **tweets** limited to 140 characters and may also subscribe to the tweets posted by other users. The service can be accessed through the Twitter Website or through external applications such as smartphones. Twitter users have adopted different conventions such as replies, retweets, and hashtags in their tweets. Twitter replies, denoted as `@username`, indicate that the tweet is a response to a tweet posted by another user. Retweets are used to re-publish the content of another tweet using the format `RT @username`. Hashtags are used to denote the context of the message by prefixing a word with a hash symbol e.g. #obama, #elections, etc. By April of 2010, Twitter had around 106 million registered users according to the presentation given by the company in the Twitter Chirp Developer Conference[4].

According to [JSFT07], Twitter users can express the following intentions in their *tweets*[5]: daily chatter, conversations, sharing information, and reporting news. Additionally, Twitter users tend to publish personal opinions regarding certain topics and news events.

A great advantage of these opinions is that they are provided freely and voluntarily by the users. Therefore, textual data from posted opinions could be aggregated and used to measure the public opinion implicitly. Nevertheless, the increasing amount of opinions generated daily on social media applications makes a human evaluation of this content impossible to achieve. For this reason, these textual opinions are commonly assessed using computational methods.

Opinion mining or *sentiment analysis* refers to the application of techniques from fields such as natural language processing, information retrieval and text classification, to identify and extract subjective information from textual datasets [PL08]. Some of the most important tasks of the field are:

- Subjectivity classification: to distinguish between objective information and opinions in textual data sources.
- Polarity classification: to detect feelings in opinionated texts by identifying whether an opinion has a positive or negative connotation about the topic being addressed.

Manual classification of thousands of posts for opinion mining tasks is an unfeasible effort at human scale. Several methods have been proposed to automatically infer human opinions from natural language texts. Due to the inherent subjectivity of the data, this problem is still an open problem in the field.

Undoubtedly, there are limitations to evaluating public opinion using opinion mining methods applied to social media data. The main limitation is that the population which uses social media platforms is not necessarily a representative sample of the entire population. Therefore, the conclusions obtained from that kind of analysis will only reflect the public opinion regarding a particular fraction of

[4] `http://www.businessinsider.com/twitter-stats-2010-4/`

[5] A tweet is a message posted in Twitter

the population. Nevertheless, opinions extracted from social media provide some benefits in comparison to traditional polling.

First of all, this approach allows for cheaply processing greater amounts of data [YK12]. Secondly, as social media opinions become available in continuous time streams, we believe that social media is more suitable for studying the temporal properties of public opinion. These properties, which are discussed later, include stationarity, trends, seasonality, and volatility.

The main motivation of this work is to understand the nature of opinions in Twitter. In the first part we study the sentiment classification of tweets combining different sentiment analysis approaches in a supervised learning framework. Then, we study the dynamics of opinions expressed in social media by studying how they evolve over time. The focus is on the predictability of opinion time series and evaluating the possibility of forecasting future outcomes.

1.1 Research Problem

Before presenting our research problem, it is appropriate to give a definition of *topic* in the context of social media. According to [All02], a topic is "a seminal event or activity, along with directly related events and activities". Additionally, in [ZJ11] the following definition of topic was given: "A topic is a subject discussed in one or more documents". Furthermore, in that work three categories of topics are proposed:

1. **Event-oriented topics**: which include news events such as natural disasters, and elections (e.g. "Chilean Earthquake").
2. **Entity-oriented topics**: include entities such as public people and brands (e.g. "Barack Obama", "Coca-Cola").
3. **Long-standing topics**: include global subjects such as "music" and "global warming".

In this work, we study two problems of Twitter opinions which are closely related to each other. The first problem consists in boosting the sentiment classification of Twitter opinions by combining different sentiment dimensions of the problem: polarity, emotions, and strength.

In the second problem, we propose a methodology for creating and analyzing Twitter opinion time series regarding *topics* of interest. An appropriate understanding of these series will allow to determine the feasibility of predicting future outcomes of public opinion regarding those topics from Twitter data.

To address these problems we will track a list of topics retrieving periodically their related messages from Twitter. Here, the hypothesis is that a topic can be represented by a set of keywords, and that a tweet containing these keywords will be associated with it.

Microblogging platforms usually provide an API[6] through which to search and extract posts. In particular, Twitter has a streaming API from which a real time sample of public posts can be retrieved and a REST API, which allows the submission of queries composed by key-terms. Therefore, if we model our topics as sets of descriptive words, we could use the Twitter REST API to retrieve messages addressing the topics. Unfortunately, this API does not allow retrieval of tweets older than about a week.

A sentiment evaluation will be performed to each tweet retrieved using opinion mining techniques. We propose to efficiently combine existing methods and resources to improve two sentiment analysis tasks: 1) Subjectivity classification, and 2) Polarity classification. These approaches are focused on three different sentiment dimensions: polarity, strength and emotion. We combine these aspects as input features in a sentiment classifier using supervised learning algorithms.

Due to the fact that tweets are timestamped, they can easily be aggregated by time periods and used to build opinion time series. For example, we could create different time series for each topic counting the number of positive, negative and neutral tweets per day over a period of several months. Furthermore, we will use different opinion mining methods in order to create the time series, and compare the differences and similarities of the resulting time series. The state of the art on opinion mining methods is presented in Section 2.1.

Once several opinion time series have been created, we will analyze them with the aim of determining whether reliable predictive models can be generated from Twitter data. For example, suppose we have collected the number of positive tweets per day related to *Obama* for a period of two months, the idea of our analysis is to determine the feasibility of predicting the number of positive tweets per day for the following days given the previous observations.

In recent years, several works claimed impressive forecasting abilities from social media. Nevertheless, as it is discussed in Section 2.3 most of these works do not perform deep statistical analysis of the time series. We believe that at least a minimal set of statistical tests must be applied in order to determine if a predictive model can be generated.

Our analysis will include different aspects of the time series, such as the identification of trends, seasonal patterns and especially, on the concept of *volatility*. As we show in Section 3.2, volatility clustering is a pattern of time series which is closely related to uncertainty and is commonly found in financial time series. Time series that exhibit volatility clustering tend to have periods of swings followed by periods of relative calm. Due to this, volatile time series are not appropriate for making long-term predictions. We hypothesize that opinion time series will also exhibit volatility patterns, and that their levels of volatility will reflect how the information spreads in social media. Moreover, we believe that when the population is more open to information, the resulting opinion time series will be more volatile.

[6]Application Programming Interface

In summary, we intend in this thesis to identify the static and dynamic properties of opinions in Twitter. Firstly, we expect that an adequate understanding of the different dimensions of Twitter opinions can lead to improvements for the sentiment classification of tweets. Then, in the dynamic analysis we will try to determine whether reliable predictive models can be built from opinion time series created from Twitter data. Additionally, we will compare these properties between different topics with the aim of determining if topics with similar characteristics share temporal properties or not.

1.2 Research Hypothesis

There is empirical evidence that social phenomena are reflected in some manner by social media data [BMZ11, AH10, Lee11]. Furthermore, there is also evidence that opinions expressed in social media can be used in some cases to assess public opinion indirectly [OBRS10]. Due to this, we propose the following research hypothesis:

> "The properties of the public opinion regarding social phenomena that are discussed in social media, can be determined using data-driven models."

The hypothesis is composed of the following two subordinate hypotheses, each of which is concerned with a different aspect of social media opinions.

1. The sentiment evaluation of social media messages can be outperformed by combining different sentiment dimensions into a supervised learning system.
2. The evolution of social media opinions can be modeled mathematically using a time series approach.

As the former hypothesis concerns with the static properties of opinions, the latter is concerned with the evolution of them. Furthermore, they both focus on the study of social media opinions using data-driven models: supervised learning and time series models. In this manner, the two hypotheses are subordinate to the main hypothesis, and hence, if one of them is refuted, the main hypothesis should also be refuted.

1.3 Objectives

1.3.1 General Objective

The main objective of this thesis is to design and develop a methodology to understand the static and dynamic properties of opinions in Twitter.

1.3.2 Specific Objectives

1. To implement "state of the art" opinion mining methods to extract sentiment indicators from a tweet.
2. To create a new sentiment classification approach using different sentiment indicators as input features for supervised learning algorithms.
3. To develop a topic-tracking tool for Twitter topics. Here, the idea is to periodically retrieve tweets which discuss a list of topics chosen to be tracked.
4. To implement a tool to build opinion time series from temporal collections of tweets. The tool should convert each tracked topic into sequences of sentiment values computed with the opinion mining methods.
5. To identify and apply statistical methods to opinion time series. Here we expect to extract temporal properties of the series. Moreover, we intend to fit the most appropriate function for each time series, and to evaluate the forecasting abilities of the resulting functions.
6. To compare temporal properties of different opinion time series and to discover if there are common temporal properties between the topics.

1.4 Methodology

In order to accomplish the specific objectives described above, a methodology composed of the following steps is proposed. The methodology is illustrated in Figure 1.1.

1. **Topic Tracking**
   A list of trackable topics will be developed, in which each topic will be represented by a set of key terms. These topics will be selected from different types of entities such as politicians, companies, countries, etc. A message retrieval tool responsible for tracking the topics on Twitter will be implemented. The tool will submit the key terms of the topics as queries through the Twitter API periodically. Additionally, the time period of the post will be obtained directly from the API.
2. **Static Analysis of Twitter Opinions**
   A text processing tool will be developed in order to implement opinion mining methods presented in Section 2.1. Mainly, we will implement unsupervised methods based on lexical resources, and supervised methods based on supervised machine learning methods applied to training corpora. The tool will also implement text preprocessing tasks, such as those proposed in [BYRN99, MRS08], like stopwords removal, stemming, tokenization, etc. Nevertheless, it is important to consider that many of these tasks are dependent on the language and require therefore, a proper implementation for each language to be included. Due to the fact that most opinion mining resources are made for English, we will mainly consider tweets written in that language.
3. **Boosting Twitter Sentiment Classification**

In this part, we will combine the methods developed in the previous part for boosting Twitter sentiment analysis. We will use the outcomes of opinion strength, emotion and polarity-based methods as the inputs of a sentiment classifier. To validate our approach we will evaluate our methodology on two existing datasets.

4. **Opinion Time Series Construction**
   At the end of each tracking period (that could be one full day), all retrieved posts will be aggregated by topic and time period into sequences of sets of tweets. All tweets will receive a sentiment value computed with a certain opinion mining technique. For each period, each set of posts will receive a public opinion value computed by the aggregation of the sentiment values within the set. Operations such as the summation or the average value could be used as aggregation criteria. Finally, the resulting sequences of public opinion values will form the opinion time series.
5. **Opinion Time Series Analysis**
   Opinion time series will be analyzed to discover patterns of variability in the historical data and used to forecast future values. We will concentrate on the conditional mean and the conditional variance of the series. As the conditional mean focuses on the expected outcome, the conditional variance is centred on how the variance changes over time. Furthermore, volatility aspects discussed before are included in the study of the conditional variance of a time series. In this work, we will mainly study the conditional mean using ARIMA models, and the conditional variance using GARCH models. These methods are described in Section 3.2.

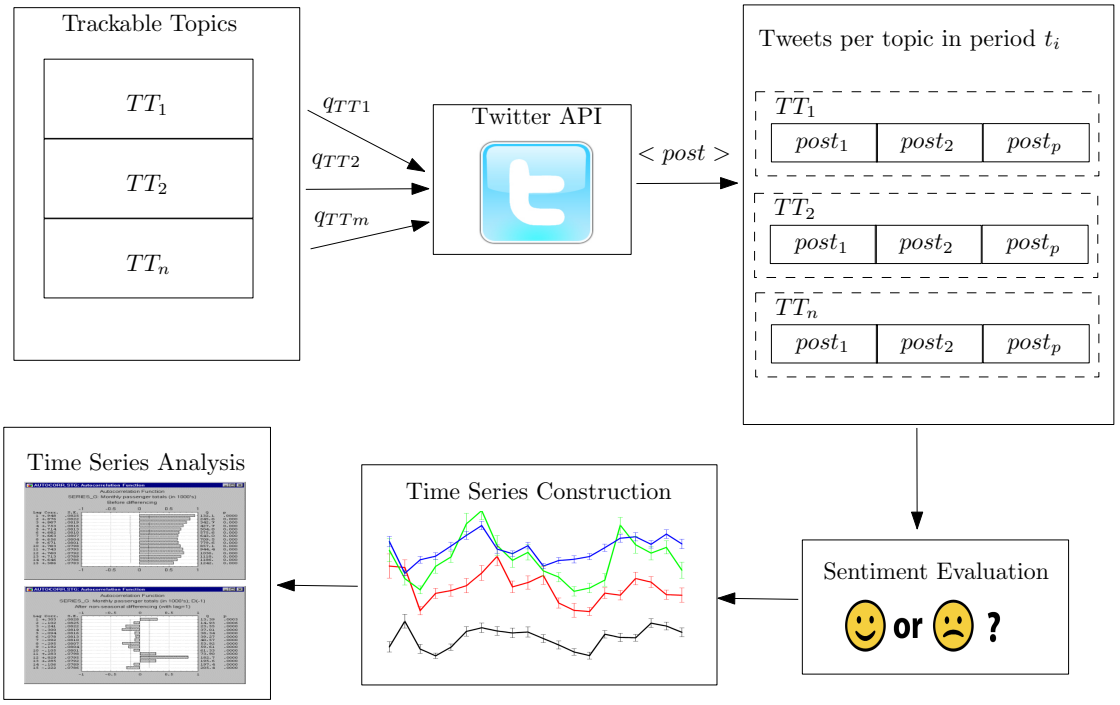


Figure 1.1: Opinion Time Series Analysis Process.

1.5 Thesis Outline

This thesis is organized as follows. In Chapter 2 we describe the related work of this thesis. The chapter covers state-of-the-art sentiment analysis approaches and also presents works that study the dynamics of opinions in social media.

The most relevant data analysis tools considered in this work are presented in Chapter 3. We introduce some of most popular supervised machine learning methods such as Naive Bayes, Support Vector Machines, Decision Trees, among others. Afterwards, we discuss time series analysis methods to study the conditional mean and the conditional variance of a time series. We present ARMA and ARIMA together with the Box and Jenkins methodology for the conditional mean and GARCH models for the conditional variance.

In Chapter 4 we present a Twitter sentiment analysis approach for polarity and subjectivity classification. The proposed approach is based on supervised learning, and uses different sentiment analysis methods and lexical resources as features.

The process on which opinion time series are created from Twitter regarding a list of defined topics is detailed in Chapter 5. We describe a process composed of three steps: topic tracking, sentiment evaluation, time series building. In the first step we present a tool that tracks topics on Twitter by retrieving tweets associated with the topics periodically using the Twitter API. Then, in the second and the third steps we extract sentiment information from the tracked tweets and create opinion time series by aggregating them over time.

In Chapter 6 we analyze opinion time series according to different statistical properties. We focus both on the conditional mean and the conditional variance of the series. The chapter is separated in two parts. In the first part we present a case study in which we analyze opinion time series related to the U.S 2008 elections. Then, in the second part we compare opinion time series related to a number of topics created with the tools presented in the previous chapter.

Finally, in Chapter 7 we discuss the main conclusions of this thesis. Additionally, we propose a methodology aimed to classify opinions from different domains in a stream data model to be developed as future work.

1.6 Publications

The main results of this thesis were published in the following international conferences and workshops:

- F. Bravo-Marquez, D. Gayo-Avello, M. Mendoza and B. Poblete *Opinion Dynamics of Elections in Twitter*, In *LA-WEB '12: 8th Latin American Web*

Congress. Cartagena de Indias, Colombia, 2012. IEEE Computer Society's Conference Publishing Services (CPS).

- F. Bravo-Marquez, M. Mendoza and B. Poblete *Combining Strengths, Emotions and Polarities for Boosting Twitter Sentiment Analysis*, In *KDD-WISDOM '13: 2nd Workshop on Issues of Sentiment Discovery and Opinion Mining*. Chicago, USA 2013.

In addition, the following publication was made from contributions partially related to this work:

- M. Mendoza, F. Bravo-Marquez, B. Poblete, and D. Gayo-Avello *Long-memory Time Series Ensembles for Concept Shift Detection*, In *KDD-BigMine '13 2nd International Workshop on Big Data, Streams and Heterogeneous Source Mining: Algorithms, Systems, Programming Models and Applications*. Chicago, USA 2013.

Chapter 2

Related Work

In this chapter, we give an introduction to the field of opinion mining and sentiment analysis. The definition of the opinion mining problem is presented, and further, state-of-the-art works in the field are reviewed. We present previous work on sentiment analysis applied to Twitter data. Finally, we discuss related work that performs temporal analysis of opinions and other works aimed at making predictions using social media.

2.1 Opinion Mining and Sentiment Analysis

There are two main types of textual information on the web: facts and opinions. On one hand, facts are assumed to be true and on the other, opinions express subjective information about a certain entity or topic. While information extraction systems like search engines are focused on solving information requirements associated with factual information, opinion mining applications focus on the processing of subjective texts. Opinion mining or sentiment analysis has become a popular discipline due to the increasing amount of user-generated content on the web from sources like forum discussions, blogs and microblogging services.

Let d be an opinionated document (e.g., a product review) composed of a list of sentences $s_1, \ldots, s_n$. As stated in [Liu09], the basic components of an opinion expressed in d are:

- **Entity**: can be a product, person, event, organization, or topic on which an **opinion** is expressed (**opinion target**). An entity is composed of a hierarchy of components and sub-components where each **component** can have a set of **attributes**. For example, a cell phone is composed of a screen, a battery among other components, the attributes of which could be the size and the weight. For simplicity, components and attributes are named together as **aspects**.

- **Opinion holder:** the person or organization that holds a specific opinion on a particular **entity**. While in reviews or blog posts the holders are usually the authors of the documents, in news articles the holders are commonly indicated explicitly [BYT+04].
- **Opinion**: a view, attitude, or appraisal of an **object** from an **opinion holder**. An opinion can have a positive, negative or neutral **orientation**, where the neutral orientation is commonly interpreted as no opinion. The orientation is also named as **sentiment orientation**, **semantic orientation** [Tur02], and **polarity**.

Considering the components of the opinions presented above, an opinion is defined as a quintuple $(e_i, a_{ij}, oo_{ijkl}, h_k, t_l)$ [Liu10]. Where e_i is an entity, a_{ij} is an aspect of e_i and oo_{ijkl} is the opinion orientation of a_{ij} expressed by the holder h_k on the time period t_l. Possible values for oo_{ijkl} are the categories positive, negative and neutral or different strength/intensity levels. In cases when the opinion refers to the whole entity, a_{ij} takes a special value named GENERAL.

In addition to the orientation, there are other criteria by which opinions can be evaluated like **subjectivity** and **emotion**. A sentence of a document is defined as subjective when it expresses personal feelings, views or beliefs. The emotions are subjective feelings and thoughts. According to [Par01] people have six primary emotions, which are: love, joy, surprise, anger, sadness, and fear. Though subjectivity and emotion are concepts which are strongly related to opinions, they are not equal. A commonly assumed fact is that opinionated sentences are subjective, although it is possible to find some exceptions [ZL11].

2.1.1 Opinion Mining Problems

It is important to consider that within an opinionated document, several opinions about different entities and also different holders can be found. The global objective of opinion mining proposed in [Liu09] consists in discovering all opinion quintuples $(e_i, a_{ij}, oo_{ijkl}, h_k, t_l)$ from a collection D of opinionated documents. Hence, working with opinionated documents involves tasks such as identifying entities, extracting aspects from the entities, the identification of opinion holders [BYT+04], and the sentiment evaluation of the opinions [PL08]. According to [Liu09], none of this problems is solved.

Sentiment or Polarity Classification

The most popular task from opinion mining is document sentiment classification. This task consists in determining the opinion orientation of a document. In order to simplify the problem, it is assumed that the document expresses opinions about one single entity from one opinion holder [Liu09]. When the sentiment classification

is applied to a single sentence instead of to a whole document, the task is named **sentence-level sentiment classification** [WWH05]. In the following, we present different approaches for solving the sentiment classification task.

The first approach is to model it as a supervised learning problem. The idea behind this approach is to train a function capable of determining the sentiment orientation of an unseen document using a corpus of sentiment labeled documents. The training and testing data can be obtained from websites of product reviews where each review is composed by a free text comment and a reviewer-assigned rating. A list of available training corpora from opinion reviews can be found in [PL08]. Afterwards, the text data and the ratings are transformed into a feature-vector and a target value respectively. For example, if the rating is a 1-5 star scale, high-starred reviews can be labeled as positive opinions and low-starred reviews as negative in the same way. The problem can also be formulated as an ordinal regression problem using the number of stars as the target variable directly [PL05]. In [PLV02], the authors trained a binary classifier (positive/negative) over movie reviews from the Internet Movie Database (IMDb). They used as features: unigrams, bigrams, and part of speech tags, and as learning algorithms: Support Vector Machines (SVM), Maximum Entropy, and Naive Bayes. The best average accuracy obtained through a three-fold cross-validation was of 82.9%. This result was achieved using purely unigrams as features and a SVM as learning algorithm.

Sentiment classification can also be perfomed using an unsupervised approach. The idea of an unsupervised approach is to be able to infer the opinion orientation of a document without a labeled corpus. In [Tur02], a method is proposed based on the identification and evaluation of words and phrases that are likely to express opinions. First of all, a part-of-speech (POS) tagging is applied to all words of the document. POS tagging consists in identifying automatically the linguistic category to which a word belongs within a sentence. Common POS categories are: noun, verb, adjective, adverb, pronoun, preposition, conjunction and interjection. The hypothesis of this work is that phrases containing a sequence of an adjective or an adverb as adjective followed by an adverb, probably express an opinion. Therefore, they extract all sentences having a sequence of words that satisfy the pattern described above. The semantic orientation of each selected sentence is estimated using the pointwise mutual information (PMI) that gives a measure of statistical independence between two words:

$$\mathrm{PMI}(term_1, term_2) = \log_2 \left(\frac{Pr(term_1 \wedge term_2)}{Pr(term_1) Pr(term_2)} \right). \tag{2.1}$$

In order to compute the semantic orientation, the PMI values of each sentence are calculated against the word “poor” and the word “excellent”. Then, as is shown in Equation 2.2, the first value is subtracted from the second one.

$$\mathrm{SO}(phrase) = \mathrm{PMI}(\mathrm{phrase}, \text{“excellent”}) - \mathrm{PMI}(\mathrm{phrase}, \text{“poor”}). \tag{2.2}$$

Both PMI values are estimated using the number of hits returned by a search engine in response to a query composed of the sentence and the word "excellent" and other query using the word "poor" in the same way. Finally, the SO of a document is calculated as the avergae SO of the sentences within it. If this value is positive, the sentiment orientation of the document is labeled with the tag "positive", otherwise it is labeled with a "negative" tag. This approach achieved an accuracy of 84% for automobile reviews and of 66% for movie reviews.

Another possible unsupervised approach for sentiment classification is to create an opinion lexicon. An opinion or sentiment lexicon is a collection of opinion words where the sentiment orientation is given. The idea is to compute the sentiment orientation of a document through a score function that uses the orientation of the words obtained from the lexicon [HW00]. As stated in [Liu09], there are two approaches to building an opinion lexicon in an unsupervised manner: the former is a dictionary-based approach and the latter is a corpus-based approach. In [HL04, KH04] a bootstrapping dictionary-based approach is proposed. The method starts by collecting a set of labeled opinion words, then the set is expanded using synonyms and antonyms of the words obtained from WordNet dictionary[1] [Fel98]. Then, the process is repeated until converge. A problem of this approach is its inability to capture domain dependent words. In a corpus-based approach, in addition to a set of seed labeled opinion words, syntactic or co-occurrence patterns are used in order to expand the set. In [HM97] the authors started with a set of adjectives whose semantic orientation was known and then discovered new adjectives with their semantic orientations using some linguistic conventions from a corpus. They show, using a log-linear regression, that conjunctions between adjectives provide indirect information about the orientation. For example, adjectives connected with the conjunction "and" tend to have the same orientation and adjectives connected with conjunction "but" tend to have the opposite orientation. This approach allows the extraction of domain-dependent information and the adaption to new domains when the corpus of documents is changed.

It is important to remark that all these sentiment evaluation approaches discussed below are still rather far from the performance usually achieved in other Natural Language Processing (NLP) tasks such as topic classification. Moreover, the performance of this methods tends to vary across different domains [PL08]. For instance, a sentiment evaluator that works properly for movie reviews will not necessarily perform well in a political context.

Subjectivity Detection and Opinion Identification

As has been seen above, sentiment classification usually assumes that documents are opinionated. However, in many cases a document within a collection contains only factual information, e.g. news sources. Furthermore, an opinionated document may contain several non-opinionated sentences. Hence, identifying the subjective

[1] WordNet is a lexical database for the English langauge.

sentences from a document is a relevant task and commonly used before the sentiment orientation evaluation. The problem of determining whether a sentence is subjective or objective is called **subjectivity classification** [WR05]. This problem can also be formulated as a supervised learning problem. In [WBO99] and [YH03] a subjectivity classifier was trained using Naive Bayes where an accuracy of 97% was achieved on a corpus of journal articles.

2.1.2 Lexical Resources for Sentiment Analysis

The computational linguistic community has paid attention to the development of lexical resources for sentiment analysis. Wilson et al. [WWH05] labeled a list of English words in positive and negative categories, releasing the Opinion Finder lexicon. Bradley and Lang [BL] released ANEW, a lexicon with affective norms for English words. The application of ANEW to Twitter was explored by Nielsen [Nie11], leveraging the AFINN lexicon. Esuli and Sebastiani [ES06] and later Baccianella et al. [BES10] extended the well known Wordnet lexical database [MBF$^+$90] by introducing sentiment ratings to a number of synsets according to three sentiment scores: positive, negative, or objective, in turn, creating SentiWordnet.

The development of lexicon resources for strength estimation was addressed by Thelwall et al. [TBP12], leveraging SentiStrength. Finally, NRC, a lexicon resource for emotion estimation was recently released by Mohammad and Turney [MT12], where a number of English words were tagged with emotion ratings, according to the emotional wheel taxonomy introduced by Plutchik [Plu01].

Besides the syntactic-level resources for sentiment analysis presented above, other types of resources have been elaborated for a semantic-level analysis such as concept-based methods. These Concept-based approaches conduct a semantic analysis of the text using semantic knowledge bases such as web ontologies [GCHP11] and semantic networks [Ols12]. In this manner, concept-based methods allow the detection of subjective information which can be expressed implicitly in a text passage. A publicly available concept-based resource to extract sentiment information from common sense concepts is SenticNet[2]. SenticNet was built using both graph-mining and dimensionality-reduction techniques [CSHH10].

Summarization and Others Tasks

In order to present all opinions related to a certain entity in a condensed form when different opinions from different documents are being analyzed, a summarization task is required. In [HL04] a structured summary is proposed. This summary counts negative and positive opinions of all aspects (including GENERAL) associated with each entity evaluated. Furthermore, the opinions are displayed as bar

[2] `http://sentic.net/`

charts. Another possible approach is using automatic summarization techniques to produce a short text summary of the opinions. In [CNP06] two text summarization techniques are compared. The first one is a sentence extraction approach, and the second one is a language generation-based approach. Experimental results show that summaries created with the former approach tend to express a general overview, and summaries created with the latter approach tend to produce a variety of expressions. Finally, there also other tasks related to opinion mining like the search and retrieval of opinions in document collections or the detection of opinion spam. These tasks are discussed in [Liu09].

2.2 Opinion Mining on Twitter

Twitter users tend to write about products or services or to discuss political views [PP10]. Tweets (posts on Twitter) are usually straight to the point and therefore are an appropriate source for sentiment analysis. Common tasks of opinion mining that can be applied to Twitter data are "sentiment classification" and "opinion identification". As Twitter messages are relatively short, a sentence-level classification approach can be adopted, assuming that tweets express opinions about one single entity. Furthermore, retrieving messages from Twitter is a straightforward task, through the use of the Twitter API. Some approaches for sentiment classification on Twitter data are discussed below.

In [GBH09], due to the difficulty of obtaining large corpora of hand-labeled data for sentiment classification, a distant supervised approach was proposed using emoticons as noisy labels. The distant supervision paradigm [MBSJ09] consists of using a weakly labeled training dataset based on a heuristic labeling function for supervised learning. Smileys or emoticons are visual cues that are associated with emotional states [CSSd09]. The idea of using emoticons as labels was proposed in [Rea05] and is based on the idea that a tweet containing a positive emoticon should have a positive orientation as the same way that the presence of a negative emoticon should indicate a negative orientation. The Twitter API was used to retrieve tweets containing positive and negative emoticons, and hence, building a training dataset of $1,600,000$ tweets. Some emoticons which could be associated with positive and negative classes are presented in table 2.1. They used similar features and learning algorithms as those described in section 2.1.1 and created a manually annotated test dataset of 177 negative tweets and 182 positive ones[3]. The best accuracy obtained was of 83.7 using a maximum entropy classifier and a feature set composed by unigrams and bigrams selected by the mutual information criterion. Furthermore, feature reduction tasks were performed such as the replacement of repeated letters (e.g., huuungry to huungry, loooove to loove) and the replacement of all mentions of Twitter users prefixed by the '@' symbol, to a generic token named as "USER". In the same way URLs were replaced to a special token with the name "URL". Pak and

[3]The training/testing corpus is available for download at `http://www.stanford.edu/~alecmgo/cs224n/trainingandtestdata.zip`

Paraoubek conducted similar work in [PP10]. They included, in addition to positive and negative classes obtained from emoticons, an objective class obtained from factual messages posted by Twitter accounts of popular newspapers and magazines.

positive	negative
:)	:(
:-)	:-(
:D	=(
=)	:'(

Table 2.1: Positive and negative emoticons.

Sentiment Lexical resources were used as features in a supervised classification scheme in [KWM11, JYZ+11, ZNSS11] among other works. In [KWM11] a supervised approach for Twitter sentiment classification based on linguistic features was proposed. In addition to using n-grams and part-of-speech tags as features, the authors used sentiment lexical resources and aspects particular from microblogging platforms such as the presence of emoticons, abbreviations and intensifiers. A comparison of the different types of features was carried out, showing that although features created from the opinion lexicon are relevant, microblogging-oriented features are the most useful.

In [BF10], authors applied data stream mining methods over Twitter data using the same corpus as the one mentioned above and also used the Edinburgh corpus[4] (97 million Twitter posts) described in [POL10]. Methods were evaluated using metrics for data streams such as the sliding window Kappa statistic. Bifet et al. developed MOA-TweetReader in [BHPG11] as an extension of the MOA framework. This extension allows the reading tweets in real time, to store the frequency of the most frequent terms, and detect change in the frequency of words.

In [ZGD+11], Zhang et al. non-supervised technique was proposed based on an augmented lexicon of opinion words. This technique does not depend on supervision or manually labeled training data and is able to capture domain opinion words regarding a topic. Liu et al. [LLG12] explored the combination of emoticon labels and human labeled tweets in language models, outperforming previous approaches.

Recently, the Semantic Evaluation (SemEval) workshop has organized a Sentiment Analysis in Twitter task (SemEval-2013)[5]. This task provides training and testing datasets for Twitter sentiment classification at both expression and message levels [WKN+13].

There are some Twitter sentiment analysis applications available on the Web.

[4]This corpus is no longer available

[5]`http://www.cs.york.ac.uk/semeval-2013/task2/`

Some of them based on supervised learning are: *Twendz*[6] , *Twitter Sentiment* [7] which is based on the work in [GBH09], that also provides an API[8], and *TweetFeel*[9]. *Twittratr*[10] performs sentiment analysis using a list of positive and negative opinion words. Finally, *Socialmention*[11] is a social media search and analysis platform that aggregates user-generated content from different social media sources. The application also performs a sentiment evaluation of the content.

2.3 Opinion Dynamics

As discussed above, opinion mining applied to social media has become an emerging field of research. Several works have been developed for different contexts, including marketing studies [JZSC09] and social sciences studies [DD10]. Most recently, there has emerged an interest in understanding temporal aspects of opinions and further, in predicting future events from social media.

2.3.1 Temporal Aspects of Opinions

A tool called *Moodviews*[12] was proposed in [Md06] to analyze temporal change of sentiment from LiveJournal[13] blogs. The tool tracks the stream of 132 mood-annotated text in the blogs and allows the visualization of mood changes through time. A temporal analysis of sentiment events using a method based on Conditional Random Field (CRF) was performed in [DKEB11]. The authors included sentiment features to events in order to identify temporal relation between different events from text sources. In [MdR06], it was showed that opinions exhibit a certain degree of seasonality in Twitter. They found that people tend to awake in a good mood that decays during the day demostrating that people are happier on weekends than weekdays.

The online detection of temporal changes in the public opinion is studied in [ABDF10]. They state that a breakpoint in the public opinion is formed both by a change in the emotion pattern and the word pattern of Twitter messages. The tweets on a certain topic TT in a time period T are used to create both a vector of sentiment dimensions $\overrightarrow{v}$ and a set formed by the words within the tweets $\mathrm{Set}(T_{\mathrm{i}})$, where the vector represents the sentiment pattern, and the set represents the word pattern. Similarity measures are used to compare the word and the sentiment

[6] http://twendz.waggeneredstrom.com
[7] http://twittersentiment.appspot.com/
[8] http://sites.google.com/site/twittersentimenthelp/api
[9] http://www.tweetfeel.com/
[10] http://twitrratr.com/
[11] http://www.socialmention.com
[12] http://moodviews.com/
[13] http://www.livejournal.com/

patterns between different periods of tweets. Thus, a period T_n must satisfy the following conditions in the two patterns in order to be considered as a breakpoint:

$$\text{Sim}(T_{n-1}, T_n) < \text{Sim}(T_{n-2}, T_{n-1}) \tag{2.3}$$

$$\text{Sim}(T_{n-1}, T_n) < \text{Sim}(T_n, T_{n+1}). \tag{2.4}$$

In [OBRS10], two mechanisms for measuring the public opinion were compared; polls and opinions extracted from Twitter data. The authors compared several surveys on consumer confidence and political opinion, like the Gallup Organization's Economic Confidence Index and the Index of Consumer Sentiment (ICS), with sentiment ratio time series. The series were created from Twitter messages by counting positive and negative words from an opinion lexicon according to the following expression:

$$x_t = \frac{\text{count}_t(\text{pos. word} \wedge \text{topic word})}{\text{count}_t(\text{neg. word} \wedge \text{topic word})} \tag{2.5}$$

Furthermore, the series were smoothed using a moving average smoothing technique in order to reduce the volatility and derive a more consistent signal. The correlation analysis between the polls and the sentiment ratio series showed that the sentiment series are able to capture broad trends in the survey data. Nevertheless, the results showed great variation among different datasets. For example, while a high correlation between the sentiment ratio series and the index of Presidential Job Approval was observed, the correlation was non-significant between the sentiment series and the pre-electoral polls for the U.S. 2008 Presidential elections.

Opinion time series created from Twitter data were also explored in [LP11]. The authors sampled around $40,000,000$ tweets in a period of 510 days using the Twitter streaming API. The sentiment evaluation of the tweets was conducted according to four emotion states: happy, sad, very happy, and very sad. A number of emoticons was mapped to each emotion state, assuming that a tweet with one of these emoticons will be associated with the corresponding emotion state. In this manner, emotion-oriented time series were calculated according to the proportion of tweets associated to each emotion state over the total number of messages in a day. The resulting time series were analyzed focusing on the study of seasonal and volatility patterns. The experimental results indicated the presence of significant weekly seasonality factors and also the presence of a significant level of volatility clustering in the time series.

In the opinion mining problem discussed in Section 2.1, the main goal is to evaluate the opinions of one single document. In contrast to this, the temporal study of opinions is concerned in evaluating the *aggregated sentiment* of a target population for a certain time period. According to [HK10], it can be inaccurate to use standard text analysis methods for document classification when the goal is to assess

aggregate population. Therefore, it is important to remark that effective methods for per-document sentiment classification will not necessarily result in the same effectiveness in a context of aggregate analysis.

2.3.2 Predictions using Social Media

As stated in [YK12], not all topics or subjects are well suited for making predictions from social media. First of all, the topic must be related to a human event, that means, that social media cannot be used to predict events whose development is independent of human actions (e.g., eclipse, earthquake). Secondly, there are some topics in which it is considered impolite to express opinions with a certain orientation. Therefore, the topics should be easy to be discussed by people in public, otherwise the content will be biased. In the following, we present some works of predictions based on social media.

Stock Market

Stock market prediction has been traditionally addressed through the random walk theory and the Efficient Market Hypothesis (EMH). This approach states that stock market prices reflect all public available information and adjust rapidly to the arrival of new information. Moreover, due to the fact that the arrival of information is unpredictable, stock prices follow a random walk process and cannot be predicted. In contrast to the approach discussed above, we discuss in the following some works claiming that social data can be used to predict stock markets. In [DCSJS08] the communication dynamics in the blogosphere was studied, showing a considerable correlation between social data and stock market activity. An SVM regressor was trained using contextual properties of communications for a particular company as features and the stock market movement of the company as target variable. Some of the features considered were: the number of posts, the number of comments, the length and response time of comments, among others. An accuracy of 78% was obtained for predicting the magnitude of movement and 87% for the direction of movement. In [BMZ11], it was investigated whether public moods extracted from Twitter data can be used to predict the stock market. Two methods were used to create mood time series from a collection of $9,853,498$ tweets from February 28 to December 19th. The former method uses *OpinionFinder*[14] to create a positive vs. negative daily time series, and the latter uses Google-Profile of Mood States (GPOMS) to create a six-dimensional daily time series based on the following mood states: Calm, Alert, Sure, Vital, Kind, and Happy. In order to asses the ability of these time series to predict stock market changes, they compared them with the Dow Jones Industrial Average (DJIA) using the econometric technique of Granger causality analysis. The results obtained indicate that the prediction of stock market can be significantly improved when mood dimensions Calm and Happiness are

[14] `http://www.cs.pitt.edu/mpqa/opinionfinder_1.html`

considered, but not others.

Movie Box-Office

"Movie box-office", is a concept used to describe how successful a movie is. There are a number of works that use social media to predict movie performance (e.g. [AH10, LHAY07, MG06]). According to [YK12], there are several reasons why predicting movie box-office is a good subject of research. The first reason is the availability of large volumes of data about movies and the easy access to them. *The Internet Movie Database*[15] (IMDB) provides box-office indicators such as the gross income of released movies. Furthermore, social media users that are interested in a movie tend to post about it and hence watch the movie. Therefore, it is a clear correlation between social media and movie box-office. For example, in [AH10], authors found more than $100,000$ tweets for each monitored movie. In that work, tweets were used to forecast box-office revenues for movies using properties such as the rate at which tweets are created and sentiment indicators. [LHAY07] proposed an Autoregressive Sentiment Aware model (ARSA) to predict box office performance from blogs. The model assumes that each blog document is generated by a number of hidden sentiment factors which are estimated using the Expectation Maximization algorithm (EM). Then, movie box revenues are predicted by combining an autoregressive model of past revenues with sentiment factors extracted from blogs.

Politics

In the context of politics, predicting elections with social media has become an active area of research. The result of an elections has been traditionally predicted through public opinion surveys such as telephone surveys or polls. There is not clear consensus about the predictive power of election predictions based on social media and opinion mining. For example, as [TSSW10] argues that the predictive power of this approach is "close to traditional election polls", [GA11] states that this power is greatly exaggerated. Furthermore, there are cases in which different social media predictions for a same event give contrary results. While [TSSW10] claims that German elections of 2009 could have been predicted using Twitter, [JJS12] states the opposite.

2.4 Discussions

Opinion mining and sentiment analysis methods applied to social media is an emerging field of research. We have reviewed several works that study how to classify

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

Twitter opinions. We also presented works that study the temporal properties of opinions, addressing the problem of predicting the *future* with social media. To the best of our knowledge, no other research work has combined different sentiment analysis resources and methods as meta-level features to enhance the sentiment classification of tweets.

Regarding the temporal analysis of Twitter opinions, we strongly believe that the study of the volatility and other aspects of opinion time series, like seasonality and stationarity, will allow us to determine the limitations of assessing public opinion from social media. Although many of these aspects were studied before in [LP11], that work has two major limitations. To begin with, the sentiment evaluation approach which is based on emoticon-based states is naive. In our work, we will create more robust opinion time series using twitter-focused sentiment analysis methods. Secondly, considering that the time series are created from the whole Twitter stream, we believe that they contain too much noise. Instead of aggregating tweets from the whole stream, we will create opinion time series associated with different topics. Therefore, the tweets from our time series will be more related to each other, and in this way we expect to reduce the level of noise in the series.

Chapter 3

Data Analysis Tools

Data mining, machine learning, statistics, and econometrics are all data analysis fields focused on acquiring knowledge from data. In order to study both the static and dynamic properties of Twitter opinions, techniques from all these fields are required, as for instance, classification, hypothesis testing, clustering, and forecasting. The most important methods to be used in this thesis are described in this chapter. The chapter is divided into two major parts. In the first part we introduce the classification problem together with the most common machine learning supervised algorithms. Then, in the second part, we present the "Box-Jenkins" time series analysis methodology together with time series analysis methods such as ARMA and GARCH models.

3.1 Classification

Classification is the task of predicting a discrete variable y using a set of features $x_1, x_2, \ldots, x_n$ as independent variables. In order to train a classifier we need to learn a hypothesis function h from a collection of training examples as shown in Figure 3.1. This collection of records, has the form $(\mathcal{X}, \mathcal{Y})$, and is usually referred to as **dataset**. Each entry of the dataset is a tuple (x, y), where x is the feature set and y is the class or target label. As was mentioned before, the target label y is a discrete variable with c possible categories. When the possible outcomes are restricted to binary values, $y_{\text{i}} \in \{+1, -1\}$, $\forall \text{i} \in \{1, \ldots, N\}$, the classification problem is referred to as binary classification problem. The different classification learning algorithm to be considered in this thesis are presented in the following sections. For further information about these algorithms please refer to [WFH11].

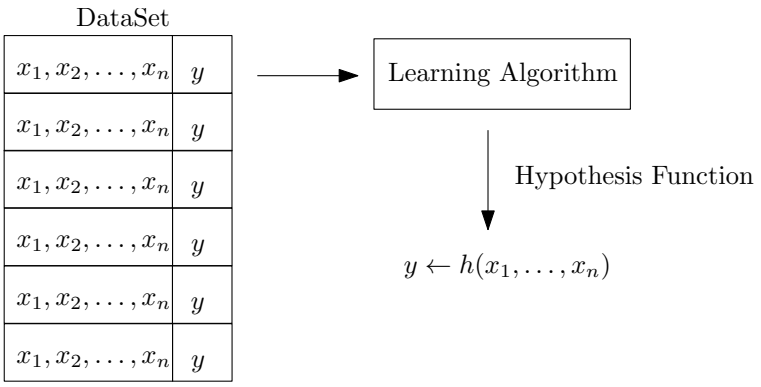


Figure 3.1: Learning Task.

3.1.1 Decision Trees

A Decision Tree is a discriminant classifier represented by a tree data structure. Each internal node from the tree corresponds to a condition that splits the data into different groups according to a specific feature. Then, each branch of the tree represents a subgroup of the data conditioned by the parent node. Finally, leaf nodes correspond to a value of the target variable given the values of the features along the path from the root of the tree to the leaf node.

Trees are constructed by repeated splits of subsets of data based on the selection of features and split conditions. There are a number of algorithms for inducing decision trees from data, e.g. ID3, C4.5, CART, in which features are selected according to an impurity measure. The most common criterion used as impurity measure for splitting the data at each node is the information gain criterion. This criterion is based on the concept of entropy that comes from information theory.

Let c be the number of classes of the target variable y and S be the dataspace associated with a node of the tree, the entropy of S is calculated as follows:

$$\text{Entropy}(S) = -\sum_{\text{i}=1}^{c} P(y = c_{\text{i}}|S) \cdot \log_2 P(y = c_{\text{i}}|S). \tag{3.1}$$

The entropy represents the impurity or homogeneity of the target variable in region S. Note that if S corresponds to the root of the tree, the entropy is calculated over the entire dataset. Let Values(A) be the possible values that variable A can take, the information gain for a feature A in the region S represents the reduction of the impurity in the dataspace induced by the feature which is calculated as follows:

$$\text{Info. Gain}(S, A) = \text{Entropy}(S) - \sum_{v \in \text{Values}(A)} \frac{[S_v]}{[S]} \text{Entropy}(S_v). \tag{3.2}$$

When A is a numerical variable, it is discretized as part of the learning algorithm.

The learning algorithm takes the feature A that maximizes the information gain to select the internal nodes of the tree.

3.1.2 Logistic Regression

This classification algorithm estimates the posterior probability $P(y|x)$ of the target variable y given the observed values of x by fitting a linear model to the data. The parameters of the model are formed by a vector of parameters β which are related to the feature space x by a linear function. Considering that the intercept term is zero $x_0 = 1$, the linear function has the following form:

$$h_\beta(x) = \sum_{\mathrm{i}=0}^{n} \beta_\mathrm{i} x_\mathrm{i} = \beta^T x. \tag{3.3}$$

The linear function $h_\beta(x)$ is mapped into the interval $[0, 1]$ using the logistic or sigmoid function:

$$g(z) = \frac{1}{1 + \mathrm{e}^{-z}}. \tag{3.4}$$

Parameters β are determined by maximizing the conditional likelihood on the dataset. Once the parameters are estimated, the predictions are made by taking the value of y that maximizes the posterior probability:

$$P(y|x) = \frac{1}{1 + \mathrm{e}^{-\beta^T \cdot x}}. \tag{3.5}$$

3.1.3 Naive Bayes

This is a probabilistic classifier that uses the Bayes theorem to estimate the posterior probability $P(y|x)$ of the class y given the observed variables x:

$$P(y|x) = \frac{P(x|y)P(y)}{P(x)}. \tag{3.6}$$

In the same way as in the logistic regression, the predictions are made by taking the value of y that maximizes $P(y|x)$. In contrast to the logistic regression in which the posterior probability is estimated directly, the naive Bayes classifier focuses on the probability $P(x|y)$. This probability is referred to as the "likelihood" and represents the probability of generating the observed data x given the value of the

class y. Due to this, the naive Bayes classifier is considered a **generative classifier**. Conversely, the logistic regression is a **discriminant classifier**.

From Equation 3.6 we can see that the denominator $P(x)$ is constant for any value of y, and hence it is not necessary to calculate it in order to make a prediction. Therefore, we can use the following approximation:

$$P(y|x) \sim P(x|y)P(y). \tag{3.7}$$

The value $P(y)$ is referred as the prior probability, and can be estimated directly from the data. However, the likelihood function $P(x|y)$ depends on the joint distribution of x given y, and since x is a multivariate random variable, $P(x|y)$ is expensive to estimate.

According to the chain rule, the joint distribution of $P(x|y)$ can be expressed as follows:

$$P(x_1, \ldots, x_n|y) = P(x_1|y)P(x_2|x_1, y) \cdots P(x_n|x_{n-1} \cdots x_2, x_1, y).$$

In order to avoid the expensive estimation of $P(x|y)$, the naive Bayes classifier takes the "strong assumption" that all pairs of features x_i and x_j are independent to each other given the evidence of y. In this manner, we have that $P(x_\mathrm{i}|x_j, y) = P(x_\mathrm{i}|y)$ for any pair $\mathrm{i}, j \in [1, n]$. Thus, the likelihood function can be represented according of the following expression:

$$P(x|y) = P(x_1|y)P(x_2|y) \cdots P(x_n|y) = \prod_{\mathrm{i}=1}^{n} P(x_\mathrm{i}|y). \tag{3.8}$$

In this way, the probabilities $P(x_\mathrm{i}|y)$ can be estimated directly from the data. As categorical features are estimated according to a multinomial distribution, numerical features are estimated from a Gaussian one.

3.1.4 Support Vector Machine Classifier

The Support Vector Machine (SVM) is a discriminant binary classifier aimed at finding the optimal hyperplane $(\omega^T \cdot x + b)$ that separates the two possible values of the target variable $y \in \{+1, -1\}$ according to the feature space represented by x. The optimal hyperplane is the one that maximizes the margin between positive and negative observations in the training dataset formed by N observations. The task of learning a SVM from a dataset is formalized as the following optimization problem:

$$\begin{aligned} \min_{w,b} \quad & \frac{1}{2}||w||^2 + C\sum_{\mathrm{i}=1}^{N}\xi_{\mathrm{i}} \\ \text{subject to} \quad & y_{\mathrm{i}}\left(w^T\mathbf{x}_{\mathrm{i}} + b\right) \geq 1 - \xi_{\mathrm{i}}\ \forall \mathrm{i} \in \{1,\ldots,N\} \\ & \xi_{\mathrm{i}} \geq 0\ \forall \mathrm{i} \in \{1,\ldots,N\}. \end{aligned} \tag{3.9}$$

The objective function of the problem focuses both on obtaining the maximum margin hyperplane and on minimizing the errors $\sum_{\mathrm{i}}^{N}\xi_{\mathrm{i}}$. The parameter C is referred to as the "soft margin regularization parameter" and controls the sensitivity of the SVM to possible outliers.

It is also possible to make SVMs find non-linear patterns efficiently using the kernel trick. A function $\phi(x)$ that maps the feature space x into a high-dimensional space is used. This high-dimensional space is a Hilbert space, where the dot product $\phi(x)\cdot\phi(x')$ is known as the kernel function $K(x,x')$. In this manner, the hyperplane is calculated in the high-dimensional space $(\omega^T\cdot\phi(x)+b)$. The dual formulation of the SVM allows replacing every dot product by a kernel function as is shown in the following expression:

$$\begin{aligned} \max_{\alpha} \quad & \sum_{\mathrm{i}=1}^{N}\alpha_{\mathrm{i}} - \frac{1}{2}\sum_{\mathrm{i},j=1}^{N}\alpha_{\mathrm{i}}\alpha_j y_{\mathrm{i}} y_j \cdot K(x_{\mathrm{i}}, x_j) \\ \text{subject to} \quad & \alpha_{\mathrm{i}} \geq 0, \forall \mathrm{i} \in \{1,\ldots,N\} \\ & \sum_{\mathrm{i}=1}^{N}\alpha_{\mathrm{i}} y_{\mathrm{i}} = 0. \end{aligned} \tag{3.10}$$

Where the parameters $\alpha_{\mathrm{i}}, \mathrm{i} \in \{1,\ldots,N\}$ correspond to the Lagrange multipliers of the constrained optimization problem. Once the parameters α were determined, it is possible to classify a new observation x_j according to the following expression:

$$\operatorname{sign}\left(\sum_{\mathrm{i}=1}^{N}\alpha_{\mathrm{i}} y_{\mathrm{i}} \cdot K(x_{\mathrm{i}}, x_j) + b\right). \tag{3.11}$$

3.1.5 Feature Selection

Feature selection is the task of identifying the best subset of variables within the training dataset for the learning purpose. In several supervised learning algorithms, factors such as the presence of features which are strongly related to each other, or that do not provide relevant information to predict the target variable, can affect

the performance of the learned model. The feature selection methods are commonly divided into the following three groups:

- Filter approaches, which rank the features according to a certain criteria, e.g. information gain, mutual information.
- Embedded approaches which occur as part of the classification method such as the decision trees.
- Wrapper approaches which use a classification algorithm as a black box to search and evaluate the desired feature subset such as the greedy forward selection method.

3.1.6 Evaluation Criteria

When we train a classifier over a dataset, and the same dataset is used for training and evaluating the performance of the classifier, the resulting model could be overfitted. Normally, in order to evaluate a classifier, the dataset is split into training and testing datasets. Afterwards, the classifier is trained over the training set and then used to classify the values of the testing set. Finally, the predicted outputs are compared with their corresponding real values from the testing dataset. This approach is known as the "hold-out" technique. Using this approach for a binary classification problem, four possible outputs can be obtained, as is shown in the confusion matrix in Table 3.1.6. These outputs are explained below.

Correctly classified positive observations or True Positives (TP), correctly classified negative observations or True Negative (TN), negative observations wrongly classified as positive (FP), and positive observations wrongly classified as negative or False Negative (FN).

	$y = +1$	$y = -1$
$c(x) = +1$	TP	FP
$c(x) = -1$	FN	TN

Table 3.1: Classification Confusion Matrix.

Using the different outputs described above, the following evaluation criteria can be used:

- Precision, the fraction of correctly classified positive observations over all the observations classified as positive:

$$\text{Precision} = \frac{TP}{TP + FP}. \tag{3.12}$$

- Recall, the fraction of correctly classified positive observations over all the positive observations:

$$\text{Recall} = \frac{TP}{TP + FN}. \tag{3.13}$$

- F-measure, the harmonic mean between the precision and recall:

$$\text{F-measure} = (1 + \beta^2)\frac{2 \cdot \text{Precision} \cdot \text{Recall}}{(\beta^2 \cdot \text{Precision}) + \text{Recall}}. \tag{3.14}$$

- Accuracy, the overall percentage of correctly classified observations:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}. \tag{3.15}$$

A drawback of the "hold-out" approach is that all the examples within the testing set are not used for training purposes. In several experiments, the labeled observations are expensive to obtain, and hence it is expected that all of them should be included in the training task. The k-fold cross-validation approach tackles this problem and is described next. Firstly, the dataset is randomly partitioned into k subsamples of the same size. Then, for each subsample k a classifier is trained over the remainder subsamples and evaluated over the retained subsample. Finally, the evaluation measures are averaged for all the subsamples ensuring that all observations were used for both training and evaluation purposes.

3.2 Time Series Analysis

A time series $Y = Y_1, Y_2, \ldots, Y_n$ is an ordered set of n real-values variables. Time series are commonly modeled as discrete time stochastic processes, which are sequences of random variables. A deeper discussion of the methods referenced in this chapter can be checked in [CC09]. The main idea is to learn a stochastic process from the observed data. Let Y_t be a time series stochastic process, the following properties can be calculated from it:

- The mean function μ_t is the expected value of the process at time t.

$$\mu_t = E(Y_t). \tag{3.16}$$

- The autocovariance function $\gamma_{t,s}$ is defined as:

$$\gamma_{t,s} = \text{Cov}(Y_t, Y_{t+k}) = E[(Y_t - \mu_t)(Y_{t+k} - \mu_{t+k})]. \tag{3.17}$$

- The autocorrelation function (ACF) $\rho_{t,t+k}$ is defined as:

$$\rho_{t,s} = \text{Corr}(Y_t, Y_{t+k}) = \frac{\text{Cov}(Y_t, Y_{t+k})}{\sqrt{\text{Var}(Y_t), \text{Var}(Y_{t+k})}}. \tag{3.18}$$

- The partial autocorrelation function (PACF) $\pi_{t,t+k}$ corresponds to the correlation between Y_t and Y_{t+k} after removing the linear dependence of all the intermediate variables $Y_{t+1}, \ldots, Y_{t+k-1}$.

- A random process is strictly stationary if the join distribution of Y_t does not change when shifted in time. A more relaxed stationarity condition that is normally studied is the weakly stationarity. Weakly stationary conditions are met in a time series when the mean function is constant through time $\mu_t = \mu$, and the covariance function is finite and identical for any pair of periods with the same distance between them, $\gamma_{t,t+k} = \gamma_{s,s+k}\ \forall t, s$.

3.3 Forecasting and Evaluation Measures

The forecasting problem in time series data is defined as follows:

- Given the sequence of data $X_1, X_2, \ldots, X_n$
- Find the values $X_{n+1}, X_{n+2}, \ldots, X_{n+m}$

In order to evaluate the forecasting accuracy of a model we compare the real value of sequence X with the estimated value $\hat{X}$ using the following measures:

1. Mean Absolute Error (MAE):

$$\text{MAE} = \frac{1}{m}\sum_{t=n+1}^{n+m}|X_t - \hat{X}_t|. \tag{3.19}$$

2. Mean Absolute Percentage Error (MAPE):

$$\text{MAPE} = \frac{1}{m}\sum_{t=n+1}^{n+m}\frac{|X_t - \hat{X}_t|}{X(t)}. \tag{3.20}$$

3. Mean Squared Error (MSE):

$$\text{MSE} = \frac{1}{m}\sum_{t=n+1}^{n+m}(X_t - \hat{X}_t)^2. \tag{3.21}$$

4. Root Mean Squared Error (RMSE):

$$\text{RMSE} = \sqrt{\frac{1}{m}\sum_{t=n+1}^{n+m}(X_t - \hat{X}_t)^2}. \tag{3.22}$$

5. Normalized Mean Squared Error (NMSE):

$$\text{NMSE} = \frac{\sum_{t=n+1}^{n+m}(X_t - \hat{X}_t)^2}{\sum_{t=n+1}^{n+m}(X_t - \overline{X})^2}. \tag{3.23}$$

3.3.1 ARMA/ARIMA Models

In order to establish if an opinion time-series should be discarded as a basis for a predictive model, it is recommended to perform at least a minimum amount of tests. Indeed, without these methodological tests, favorable results of predictive models do not provide enough evidence to support their forecasting power.

In this work, we follow the Box-Jenkins methodology [BJ94] based on ARMA/ARIMA models for modelling the expected mean of the process. An ARMA(p, q) process is defined by the following expression:

$$X_t = \sum_{\mathrm{i}=1}^{p} \alpha_{\mathrm{i}} X_{t-\mathrm{i}} + \sum_{j=0}^{q} \beta_j \varepsilon_{t-j}. \tag{3.24}$$

The first summation refers to the autoregressive part of the model (AR(p)), the second one refers to the moving average part (MA(q)) and ε_t is a series of uncorrelated random variables with mean zero and variance σ^2. An ARIMA(p, d, q) model is a process whose d-th difference is an ARMA process.

As the Box-Jenkins methodology suggests, first of all, a model specification step must be conducted. Here, the stationarity of the time-series must be checked using methods such as the Augmented Dickey-Fuller unit root test.

Due to the fact that ARMA models are defined for stationary time series, in the case of having a non-stationary time series, it should be differenced until the stationarity conditions are satisfied, with d being the number of times the time-series is differenced. Moreover, the order of the autoregressive and moving average parameters (p, q) of the ARIMA(p, d, q) model can be identified from the shape of the autocorrelation and partial autocorrelation plots. Parameters $\alpha_{\mathrm{i}}, \beta_j$ from Equation 3.24 can be estimated by one of several methods such as: the methods of moments, least squares or maximum likelihood.

It is recommended to fit a grid of plausible models varying the values of p and q. Then the best model can be chosen according to the Akaike's information criterion (AIC) and the Bayesian information criterion (BIC), among others. Once the correct parameters are estimated, a diagnostic check or residual analysis should be performed.

The previous procedure generates an ARMA or ARIMA predictive model, but it will not be reliable if the one-step-ahead conditional variance does not always show the same value as the noise variance. Time-series which present volatility do not meet this criteria, therefore ARIMA models cannot generate reliable predictive models.

3.3.2 Volatility and GARCH models

It is important to state, that while ARIMA models are used to model the expected mean of the time-series, GARCH models are focused on modeling the past conditional variance. As stated in [CC09], the past conditional variance or volatility of a time-series given past observations, measures the uncertainty in the deviation of the time-series from its conditional mean.

Volatility effects have been studied in price theory for many years. Mandelbrot [Man63] observed that large price changes were followed by large price fluctuations and small price changes were followed by small price fluctuations. The patterns of changing from quiet to volatile periods is named as *volatility clustering*. Time-sensitive volatility analysis allows the identification of hectic periods (large fluctuations) and calm periods (small fluctuations). The most suitable models that deal with volatility are the Generalized Autoregressive Conditional Heteroscedasticity (GARCH) models [Eng82, Bol86] which are discussed below.

Let $\sigma^2_{t|t-1}$ be the expected conditional variance or volatility of a zero-mean time-series r_t at period t, the GARCH(q, p) process that models $\sigma^2_{t|t-1}$ is defined as follows:

$$\sigma^2_{t|t-1} = \alpha_0 + \sum_{\mathrm{i}=1}^{q} \alpha_{\mathrm{i}} \varepsilon^2_{t-1} + \sum_{j=1}^{p} \beta_j \sigma^2_{t-j|t-j-1}. \tag{3.25}$$

Here, the first term α_0 together with the first summation refer to the autoregressive conditional heteroscedasticity (ARCH) part of the model, whereas the second summation reflects how past values of $\sigma^2_{t|t-1}$ are feedback to the present value. It is clear from Equation 3.25 that ARCH models are included in GARCH models as a special case. For the remainder of this thesis we will use the term "GARCH" to refer to both ARCH and GARCH models. Considering that GARCH models require a zero-mean time-series, a common approach used in financial time-series is to model the continuously compounded return r_t of a positive time-series X_t (e.g., stock prices), where the return values are expressed as: $r_t = \log(\frac{X_t}{X_{t-1}})$. It is also possible to work with the residuals of an ARMA/ARIMA model fitted from a non-zero-mean time-series.

We rely on the McLeod-Li test for detecting conditional heteroscedasticity in the time-series. This test is equivalent to the Ljung-Box statistics applied to the squared returns or residuals, and basically detects if the squared data are autocorrelated. In GARCH models the squared returns are unbiased estimators of the unobserved conditional variance. In the case of rejecting the null hypothesis, this implies the presence of a past conditional variance. In this scenario, opinion series are volatile, being very difficult to forecast the opinion trends in the long term using ARMA or ARIMA models. Moreover, if the volatile time-series is correctly specified, a GARCH model could be fitted using maximum likelihood estimators and hence future volatility values could be forecasted.

We hypothesize that volatility can be a very relevant aspect in opinion time-series. Intuitively, during hectic periods, people tend to be more sensitive to information and hence opinion trends register larger fluctuations. Therefore, large opinion changes are followed by large opinion fluctuations. In this situation, ARMA/ARIMA predictive models are not reliable.

Chapter 4

Static Analysis of Twitter Opinions

In this chapter we study Twitter opinions in a static fashion. The main goal of the study is to enhance Twitter sentiment classification by combining existing sentiment analysis methods and lexical resources. We refer to this analysis as static, because we handle each tweet as a single entity. On the other hand, in the dynamic analysis carried out in Chapter 6, tweets are treated in an aggregated way.

As discussed in Chapter 2, several methods and resources have been developed aiming to extract sentiment information from text sources. Supervised and unsupervised approaches have been explored to fulfill the polarity evaluation task. In the case of the unsupervised approaches, a number of lexicon resources with words labeled with positive and negative scores have been released [Nie11, BL, ES06], among others. Another related task is the detection of subjectivity, which is the specific task of separating factual and opinionated text. This problem has also been addressed by using supervised approaches [WWH05]. Opinion intensities (strengths) have also been measured. From a strength scored method, SentiStrength [TBP12] can estimate positive and negative strength scores at sentence level. Finally, emotion estimation has also been addressed by developing lexicons. The Plutchik wheel of emotions (Figure. 4.1[1]) was proposed in [Plu01]. The wheel is composed of four pairs of opposite emotion states: **joy-trust**, **sadness-anger**, **surprise-fear**, and **anticipation-disgust**. Mohammad et.al [MT12] labeled a number of words according to Plutchik emotional categories, developing the NRC word-emotion association lexicon.

According to the previous paragraph, we see that sentiment analysis tools can focus on different scopes of the opinions. Although these scopes are very difficult to categorize explicitly, we propose the following categories.

1. **Polarity**: These methods and resources aim to extract polarity information from the passage. Polarity-oriented methods normally return a categorical variable whose possible values are positive, negative and neutral. On the other

[1]source: `http://en.wikipedia.org/wiki/File:Plutchik-wheel.svg`

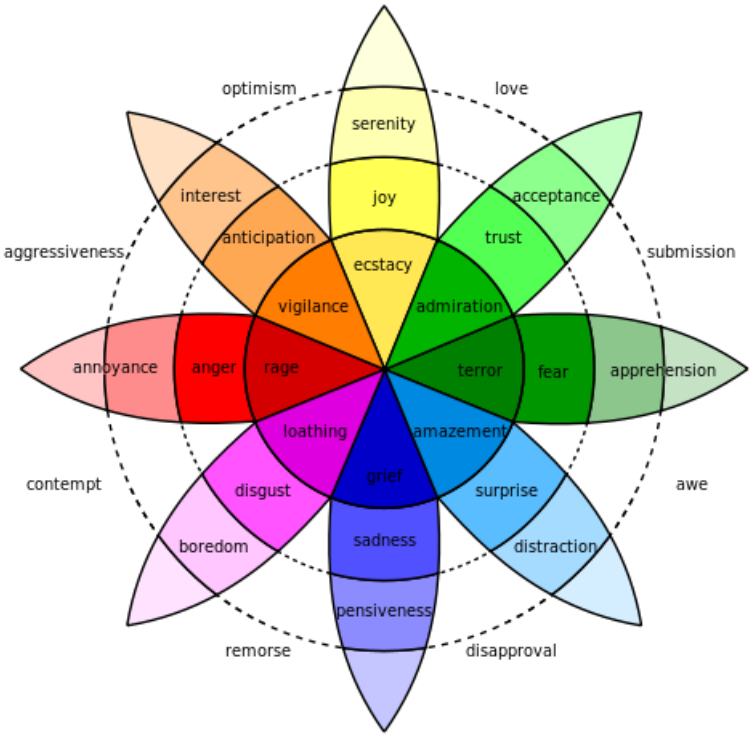


Figure 4.1: Plutchik Wheel of emotions.

hand, polarity-oriented lexical resources are composed of lists of positive and negative words.

2. **Emotion**: Methods and resources focused on extracting emotion or mood states from a text passage. An emotion-oriented method should classify the message to an emotional category such as sadness, joy, surprise, etc. An emotion-oriented lexical resource should provide a list of words or expressions marked according to different emotion states.
3. **Strength**: These methods and resources provide intensity levels according to a certain sentiment dimension which can have a polarity or an emotional scope. Strength-oriented methods return different numerical scores indicating the intensity or the strength of an opinion dimension expressed in the passage, for instance, numerical scores indicating the level of positivity, negativity or another emotional dimension. Strength-oriented lexical resources provide lists of opinion words together with intensity scores regarding an opinion dimension.

The proposed Twitter sentiment analysis approach combines methods and resources from the different scopes using classification techniques. To validate our approach we evaluate our methodology on two existing datasets. Our results show that the composition of these features achieves significant improvements over single approaches. This fact indicates that strength, emotion and polarity-based resources are complementary and address different dimensions of the same problem. Therefore, a tandem approach should be more appropriate. The approach is detailed in

the following section.

This chapter is organized as follows. In Section 4.1 we describe the classification approach for Twitter sentiment classification together with the features considered to represent the tweets. The main experiments are presented in Section 4.2. Finally, we conclude in Section 4.3 with a brief discussion.

4.1 Classification Approach

In this section we describe the proposed approach for automatic Twitter sentiment classification. We consider two classification tasks: subjective and polarity classification. In the former, tweets are classified as subjective or objective, and in the latter as positive or negative. Our approach relies on supervised learning algorithms, and hence a dataset of manually annotated tweets is required for training and evaluation purposes. A vector of sentiment features is calculated to characterize each tweet in the dataset. In contrast to the common text classification approach, in which the words contained within the passage are used as features (e.g., unigrams, n-grams), our features are based on existing lexical resources and sentiment analysis methods. These resources and methods, summarize the main efforts discussed in the state-of-the-art to address sentiment analysis, and cover three different dimensions of the problem: polarity, strength, and emotions. Moreover, the resources are publicly available, facilitating repeatability of our experiments.

Once the feature vectors of all the tweets from the dataset have been extracted, they are used together with the annotated sentiment labels as input for supervised learning algorithms. Finally, the resulting learned function can be used to automatically infer the sentiment label regarding an unseen tweet.

4.1.1 Features

We provide some details about the feature extraction procedure. From each lexical resource we calculate a number of features according to the number of matches between the words from the tweet and the words from the lexicon. When the lexical resource provides strength values associated to the words, the features are calculated through a weighted sum. Finally, for each sentiment analysis method, its outcome is included as a dimension in the feature vector. The features are summarized in Table 4.1, and are described along with their respective methods and resources in the following sections.

Scope	Feature	Source	Description
Polarity	SSPOL	SentiStrength	method label (negative, neutral, positive)
	S140	Sentiment140	method label (negative, neutral, positive)
	OPW	OpinionFinder	number of positive words that match OpinionFinder
	ONW		number of negative words that match OpinionFinder
Strength	SSP	SentiStrength	method score for the positive category
	SSN		method score for the negative category
	SWP	SentiWordNet	sum of the scores for the positive words that match the lexicon
	SWN		sum of the scores for the negative words that match the lexicon
	APO	AFINN	sum of the scores for the positive words that match the lexicon
	ANE		sum of the scores for the negative words that match the lexicon
Emotion	NJO	NRC	number of words that match the joy word list
	NTR		... matches the trust word list
	NSA		... matches the sadness word list
	NANG		... matches the anger word list
	NSU		... matches the surprise word list
	NFE		... matches the fear word list
	NANT		... matches the anticipation word list
	NDIS		... matches the disgust word list

Table 4.1: Features can be grouped into three classes with a scope of Polarity, Strength, and Emotion, respectively.

OpinionFinder Lexicon

The **OpinionFinder Lexicon** (**OPF**) is a polarity-oriented lexical resource created by Wilson et al. [WWH05]. It is an extension of the Multi-Perspective Question-Answering dataset (MPQA), that includes phrases and subjective sentences. A group of human annotators tagged each sentence according to their polarity. Then, a pruning phase was conducted over the dataset to eliminate tags with low agreement. Thus, a list of sentences and single words was consolidated with their polarity tags. In this study we consider single words (unigrams), that correspond to a list of $8,221$ English words.

We extract from each tweet two features related to the OpinionFinder lexicon, **OpinionFinder Positive Words** (OPW) and **OpinionFinder Negative Words** (ONW), that are the number of positive and negative words of the tweet that match the OpinionFinder lexicon, respectively.

AFINN Lexicon

This lexicon is based on the **Affective Norms for English Words** lexicon (ANEW) proposed by Bradley and Lang [BL]. ANEW provides emotional ratings for a large number of English words. These ratings are calculated according to the psychological reaction of a person to a specific word, being "valence" the most useful value for sentiment analysis. "Valence" ranges in the scale "pleasant-unpleasant". ANEW was released before the rise of microblogging and, hence, many slang words commonly

used in social media were not included. Considering that there is empirical evidence about significant differences between microblogging words and the language used in other domains [BYR11] a new version of ANEW was required. Inspired in ANEW, Nielsen [Nie11] created the **AFINN** lexicon, which is more focused on the language used in microblogging platforms. The word list includes slang and obscene words as also acronyms and web jargon. Positive words are scored from 1 to 5 and negative words from -1 to -5, hence the reason why this lexicon is useful for strength estimation. The lexicon includes $2,477$ English words. We extract from each tweet two features related to the AFINN lexicon, **AFINN Positivity** (APO) and **AFINN Negativity** (ANE), that are the sum of the ratings of positive and negative words of the tweet that match the AFINN lexicon, respectively.

SentiWordNet Lexicon

SentiWordNet 3.0 (**SWN3**) is a lexical resource for sentiment classification introduced by Baccianella et al. [BES10], that it is an improvement of the original SentiWordNet proposed by Esuli and Sebastiani [ES06]. SentiWordNet is an extension of **WordNet**, the well-known English lexical database where words are clustered into groups of synonyms known as **synsets** [MBF$^+$90]. In SentiWordNet each synset is automatically annotated in the range $[0, 1]$ according to positivity, negativity and neutrality. These scores are calculated using semi-supervised algorithms. The resource is available for download[2].

In order to extract strength scores from SentiWordNet, we use the word's scores to compute a real value from -1 (extremely negative) to 1 (extremely positive), where neutral words receive a zero score. We extract from each tweet two features related to the SentiWordnet lexicon, **SentiWordnet Positiveness** (SWP) and **SentiWordnet Negativeness** (SWN), that are the sum of the scores of positive and negative words of the tweet that match the SentiWordnet lexicon, respectively.

SentiStrength Method

SentiStrength is a lexicon-based sentiment evaluator that is focused on short social web texts written in English [TBP12]. SentiStrength considers linguistic aspects of the passage such as a negating word list and an emoticon list with polarities. The implementation of the method can be freely used for academic purposes and is available for download[3]. For each passage to be evaluated, the method returns a positive score, from 1 (not positive) to 5 (extremely positive), a negative score from -1 (not negative) to -5 (extremely negative), and a neutral label taking the values: -1 (negative), 0 (neutral), and 1 (positive).

We extract from each tweet three features related to the SentiStrength method,

[2] `http://sentiwordnet.isti.cnr.it/`
[3] `http://sentistrength.wlv.ac.uk/`

SentiStrength Negativity (SSN) and **SentiStrength Positivity** (SSP), that correspond to the strength scores for the negative and positive classes, respectively, and **SentiStrength Polarity** (SSPOL), that is a polarity-oriented feature corresponding to the neutral label.

Sentiment140 Method

Sentiment140[4] is a Web application that classifies tweets according to their polarity. The evaluation is performed using the distant supervision approach proposed by Go et al. [GBH09], previously discussed in the related work section. The approach relies on supervised learning algorithms and due to the difficulty of obtaining a large-scale training dataset for this purpose, the problem is tackled using positive and negative emoticons and noisy labels [GBH09, PP10]. The approach assumes that the orientation of the emoticon defines the orientation of the entire passage. The method provides an API[5] that allows classification of tweets to three polarity classes: positive, negative, and neutral.

We extract from each tweet one feature related to the Sentiment140 output, **Sentiment140** class (S140), that corresponds to the output returned by the method.

NRC Lexicon

NRC is a lexicon that includes a large set of human-provided words with their emotional tags. By conducting a tagging process in the crowdsourcing Amazon Mechanical Turk platform, Mohammad and Turney [MT12] created a word lexicon that contains more than 14,000 distinct English words annotated according to the Plutchik's wheel of emotions. Eight emotions were considered during the creation of the lexicon, joy-trust, sadness-anger, surprise-fear, and anticipation-disgust, which makes up four opposing pairs. The word list is available upon request[6].

We extract from each tweet eight features related to the NRC lexicon, **NRC Joy** (NJO), **NRC Trust** (NTR), **NRC Sadness** (NSA), **NRC Anger** (NANG), **NRC Surprise** (NSU), **NRC Fear** (NFE), **NRC Anticipation** (NANT), and **NRC Disgust** (NDIS), that are the number of words of the tweet that match each category.

[4] `http://www.sentiment140.com/`

[5] `http://help.sentiment140.com/api`

[6] mailto: saif.mohammad@nrc-cnrc.gc.ca

4.2 Experiments

In this section, we conduct several experiments to validate our approach. Firstly, we compare the different lexical resources showing the manner in which they complement each other. Secondly, we describe the datasets used for training and testing purposes. Then, we study the utility of the different features for each classification task. Finally, the classification results are presented.

4.2.1 Lexical Resource Interaction

In this section we study the interaction of words between the different lexical resources: SWN3, NRC, OpinionFinder, and AFINN. The number of words that overlap between each pair of resources is shown in Table 4.2. From the table we can see that SWN3 is much larger than the other resources. Nevertheless, the resource includes many neutral words provided by WordNet that lack of useful information for sentiment analysis purposes.

	SWN3	NRC	AFINN	OPFIND
SWN3	147,306	×	×	×
NRC	13,634	14,182	×	×
AFINN	1,783	1,207	2,476	×
OPFIND	6,199	3,596	1,245	6,884
Total Words	149,114			

Table 4.2: Intersection of words between different Lexical Resources.

	SWN3	NRC	AFINN	OPFIND
SWN3	34,257	×	×	×
NRC	3,870	4,031	×	×
AFINN	1,341	865	2,017	×
OPFIND	4,256	2,207	1,025	4,869
Total Words	35,618			

Table 4.3: Intersection of non-neutral word.

Table 4.3 shows the overlap of words after discarding the neutral words from SentiWordNet, the neutral and mixed words from OpinionFinder and the words without emotion tags from NRC. We can see that although the size of SWN3 was strongly reduced, it still has many more words than the others. The interaction of all the non-neutral words, is better represented in the Venn diagram shown in Figure 4.2. From the diagram we can see that SWN3 covers the majority of the words within the lexical resources. However, if we discard SWN3 we keep with three different sets of words: NRC having words related to emotions, OpinionFinder whose words are related to polarity, and AFINN whose words are also related to polarity with additional strength information. These resources, in addition to having

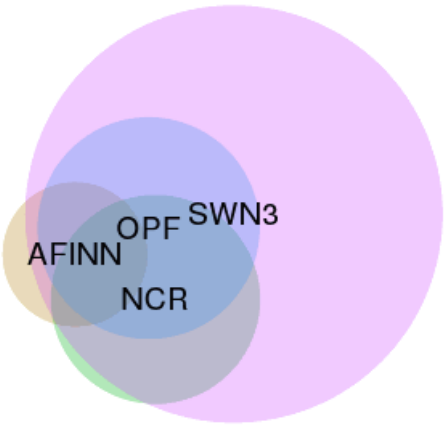


Figure 4.2: Non-neutral words intersection Venn diagram.

word	SWN3	AFINN	OPFIND	NRC
abuse	-0.51	-3	negative	ang, disg, fear, sadn
adore	0.38	3	positive	ant, joy, trust
cheer	0.13	2	positive	ant, joy, surp, trust
shame	-0.52	-2	negative	digs, fear, sadn
stunned	-0.31	-2	positive	fear, surpr
sympathy	-0.13	2	negative	sadn
trust	0.23	1	positive	trust
ugly	-0.63	-3	negative	disg
wonderful	0.75	4	positive	joy, surp, trust

Table 4.4: Sentiment Values of Words included in all the Resources.

different sentiment scopes, cover many different words from each other. It is also revealed from the figure that the AFINN lexicon, despite being smaller, contains some words that are not included in SWN3, nor in the others. We inspected these words included only in AFINN and we found many Internet acronyms and slang words such as “lmao”, “lol”, “rofl”, among other expressions.

We compare the sentiment values assigned by each lexical resource to a sample of words that appear in the intersection of all lexicons in Table 4.4. We can observe a tendency of the different resources to support each other, e.g., words that received negative strength values from SWN3 and AFINN normally receive a negative tag from OpinionFinder and are associated as well with negative NRC emotion states. A similar pattern is observed for positive words. However, we can also see controversial examples such as words “stunned” and “sympathy” which receive contrary sentiment values from polarity and strength-oriented resources. These words may be used to express either positive and negative opinions, depending on the context. Considering that it is very difficult to associate them to a single polarity class, we think that emotion tags explain in a better manner the diversity of sentiment states triggered by these kind of words.

These insights indicate that the resources considered in this work complement

each other, providing different sentiment information.

4.2.2 Datasets

We consider two collections of tweets for our experiments: *Stanford Twitter Sentiment* (STS) [7] which was used by Go et al. [GBH09] in their experiments, and *Sanders*[8]. Each tweet includes a **positive**, **negative** or **neutral** tag. Table 4.5 summarizes both datasets.

Negative and positive tweets were considered as subjective. Neutral tweets were considered as objective. Subjective/objective tags favor the evaluation of subjectivity detection. For polarity detection tasks, positive and negative tweets were considered, discarding neutral tweets.

Both datasets were balanced. Class imbalance was tackled by sampling 139 subjective tweets in STS from the 359 positive and negative tagged tweets, achieving a balance with the 139 neutral tweets. In the case of Sanders, the neutral collection was sampled recovering 1,196 tweets from the 2,429 neutral tweets achieving a balance with the 1,196 positive and negative tagged tweets. A similar process was conducted for class imbalance in the case of polarity recovering 354 and 1,120 tweets from STS and Sanders respectively. Table 4.6 summarizes the balanced datasets.

	STS	Sanders
#negative	177	636
#neutral	139	2,429
#positive	182	560
#total	498	3,625

Table 4.5: Datasets Statistics.

Subjectivity	STS	Sanders
#neutral	139	1,196
#subjective	139	1,196
#total	278	2,392
Polarity	**Sent140**	**Sanders**
#negative	177	560
#positive	177	560
#total	354	1,120

Table 4.6: Balanced Datasets.

4.2.3 Feature Analysis

For each tweet of the two datasets we calculated the features summarized in Table 4.1. In a first analysis we explored how well each feature splits each dataset

[7] `http://cs.stanford.edu/people/alecmgo/trainingandtestdata.zip`
[8] `http://www.sananalytics.com/lab/twitter-sentiment/`

regarding polarity and subjectivity detection tasks. We do this by calculating the information gain criterion of each feature in each category. The information gain criterion measures the reduction of the entropy within each class after performing the best split induced by the feature. Table 4.7 shows the information gain values obtained.

Scope	Feature	Subjectivity		Polarity	
		STS	Sanders	STS	Sanders
Polarity	SSPOL	**0.179**	0.089	**0.283**	**0.192**
	S140	**0.103**	0.063	**0.283**	**0.198**
	OPW	0.088	0.024	0.079	0.026
	ONW	0.097	0.024	**0.135**	0.075
Strength	SSP	0.071	0.037	**0.200**	**0.125**
	SSN	0.090	0.044	**0.204**	**0.118**
	SWN	0.090	0.023	**0.147**	0.089
	SWP	**0.104**	0.030	0.083	0.015
	APO	0.088	0.024	0.079	0.026
	ANE	**0.134**	0.048	**0.200**	**0.143**
Emotion	NJO	0.000	0.000	0.055	0.065
	NTR	0.000	0.000	0.000	0.000
	NSA	0.000	0.017	0.000	0.056
	NANG	0.000	0.016	0.046	0.055
	NSU	0.000	0.000	0.000	0.017
	NFE	0.000	0.008	0.039	0.024
	NANT	0.000	0.000	0.000	0.000
	NDIS	0.000	0.014	0.056	0.030

Table 4.7: Feature information gain for each sentiment analysis task. Bold fonts indicate the best splits.

As Table 4.7 shows, the best polarity splits are achieved by using the outcomes of the methods (see SSPOL, S140, SSP, and SSN). SentiWordNet, OpinionFinder and AFINN-based features are useful for negative polarity detection. These features are also useful for subjectivity detection. In addition, we can observe that the best splits are achieved in the STS. The Sanders dataset is hard to split. By analyzing the scope, we can observe that polarity-based features are the most informative. This fact is intuitive because the target variables belong to the same scope. Finally, although emotion features provide almost no information for subjectivity, some of them like joy, sadness and disgust are able to provide some information for the polarity classification task.

We also explored feature-subsets extracted by the correlation feature selection algorithm (CFS) [Hal99]. This algorithm is a best-first feature selection method that considers different types of correlation as selection criteria. Selected features for each classification task on the two datasets are displayed in Table 4.8.

From the table we can see that the two features that come from polarity-oriented methods (S140 and SSPOL), are selected in all the cases. We can also observe that the algorithm tends to include more features for polarity than for subjectivity classification. Regarding the emotion-oriented features, the only feature that is selected by the CFS algorithm is the NJO feature. Moreover, the feature is only

	Neu.STS	Neu.San	Pol.STS	Pol.San
ANE	✓	✓	✓	✓
APO	✓		✓	✓
ONW	✓		✓	✓
OPW	✓			
NJO				✓
S140	✓	✓	✓	✓
SSN			✓	✓
SSP			✓	
SSPOL	✓	✓	✓	✓
SWN	✓		✓	✓
SWP	✓	✓		

Table 4.8: Selected Features by CFS algorithm.

selected for the polarity task on the Sanders dataset. These results agree with the information gain values discussed above, and support the evidence that most of the features are more informative for polarity than for subjectivity classification.

4.2.4 Classification Results

We evaluate a number of learning algorithms on the STS and Sanders datasets, for both subjectivity and polarity detection. We conducted a 10-fold cross-validation evaluation. As learning algorithms we considered CART, J48, Naive Bayes, Logistic regression, and RBF SVMs. The experiments were performed using the R 2.15.2 environment for statistical computing with the following packages: **rpart**[9] for CART, **rWeka**[10] for J48 and Logistic regression, and **e1071**[11] for Naive Bayes and SVMs.

The performance of many machine learning techniques are highly dependent on the calibration of parameters. Different parameters such as the min-split criterion for trees, γ and C for radial SVMs, among others were tuned using a grid-search procedure with 10-fold cross validation.

An example of the tuning process for the radial SVM is shown in Figure 4.3. The x-axis and y-axis of the chart represent the **gamma** and **cost** parameter respectively. The color of the region corresponds to the classification error obtained using the corresponding parameter values. From the figure we can see that the classification performance varies considerably for different parameter values. Therefore, it is important to remark that the tuning process of machine learning parameters is crucial for obtaining accurate classifiers.

A relevant issue regarding our feature-set is its heterogeneity. Most of the features

[9] `http://cran.r-project.org/web/packages/rpart/`
[10] `http://cran.r-project.org/web/packages/RWeka`
[11] `http://cran.r-project.org/web/packages/e1071/`

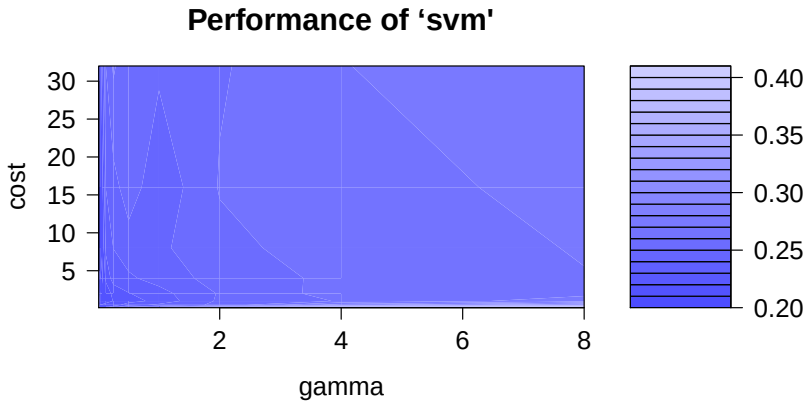


Figure 4.3: RBF SVM parameters performance for Polarity classification on Sanders dataset.

are numerical but two of them are categorical (S140 and SSPOL). A number of supervised learning algorithms are not capable of handling mixed-type features and hence some transformations must be applied before the learning task. For CART and J48 the numerical features are "discretized" as part of the learning process. Naive Bayes handles numerical features by assuming Gaussian distributions. For the SVM and Logistic regression algorithms, we transformed categorical features into dummy variables by mapping the c possible categories to binary values using 1-of-c encoding. Afterwards, these binary variables are handled as numerical features by these learning algorithms.

The performance of our classifiers in both classification tasks is compared with baselines created from isolated methods or resources. In the subjectivity task we considered the features Sent140 and SSPOL as **Baseline.1** and **Baseline.2**, respectively. For both methods, the positive and negative outputs are interpreted as subjective. We chose these features because they are the only ones which explicitly distinguish between subjective and neutral tweets.

Nevertheless, these methods could not be used as baselines for the polarity task, because it is not clear how to handle their neutral outcomes in this context. Therefore, we created two categorical variables whose outcomes are restricted to **positive** and **negative** values. The **Baseline.1** is calculated from strength features SSP and SSN as follows: if the sum of **SSP** and **SSN** is positive the baseline takes a **positive** value, otherwise it takes a **negative** value. Then, the second baseline (**Baseline.2**), is calculated in the same manner as the first one using the features **APO** and **ANE**. Considering that for SentiStrength and AFINN, positivity and negativity are assessed independently. What we are doing in our baselines is combining these dimensions into categorical variables that are constrained to distinguish between positive and negative tweets.

In addition to the feature-subset obtained by the best-first CFS algorithm, we also explored feature-subsets constrained to the scope. Thus, we evaluate five groups

of features –all, best-first, polarity, strength, and emotion– and for each group, five learning algorithms –CART, J48, naive Bayes, logistic regression, and SVMs.

We consider as performance measures **accuracy**, **precision**, **recall** and F_1. We believe that the costs of misclassifying each type of observation for each classification task are equally important. Thus, considering that our datasets are balanced, we will pay more attention to the measures accuracy and F_1 than to precision and recall measures. This is because accuracy and F_1 measures are affected by both false positive and false negative results.

Table 4.9 shows the results for the subjectivity classification task. We can observe that **Baseline.2** outperforms **Baseline.1** in both datasets. This is because Sentiment140 is not focused on subjectivity classification.

There are significant performance differences between both datasets. We hypothesize that STS's tweets have good properties for classification because they show clear differences between neutral and non neutral tweets. On the other hand, in the Sanders dataset, we found tweets marked as neutral that contain mixed positive and negative opinions. Two examples of this kind of tweets are presented below.

- *Hey @Apple, pretty much all your products are amazing. You blow minds every time you launch a new gizmo. That said, your hold music is crap.*
- *#windows sucks... I want #imac so bad!!! why is it so damn expensive :(@apple please give me free imac and I will love you :D*

Both tweets are about the company **Apple**. The first tweet shows a positive opinion about Apple's products and at the same time shows a negative opinion about Apple's hold music. This example contains contrary opinions about two different aspects of the entity Apple. The second example is even more complicated because it expresses opinions on two different entities: **Windows** and **Apple**. The tweet compares two products and shows a clear preference for Apple's product **iMac**. Additionally, the message indicates that the product **iMac** is too expensive, something that could be interpreted as a negative opinion about the product. By inspection, those kinds of tweets are not included in STS. Due to this fact, we believe that in addition to being larger, Sanders captures the sentiment diversity of tweets in a better way than the STS corpus. Nevertheless, considering that tweets with mixed positive and negative indicators are subjective, we believe that labeling them as **neutral** may increase the level of noise in the data.

Regarding learning algorithms, SVM tends to outperform other methods in accuracy and F_1, and most of the best results are achieved using the best feature selection algorithm. As was expected, the emotion feature subset achieves poor classification results for this task.

Polarity performance results are shown in Table 4.10. In this case, both baselines are strongly competitive. However, the SentiStrength-based baseline achieved a

Dataset		STS				Sanders			
Features	Methods	accuracy	precision	recall	F_1	accuracy	precision	recall	F_1
Baseline.1	Sent140	0.655	0.812	0.403	0.538	0.615	0.686	0.424	0.524
Baseline.2	SSPOL	0.734	0.712	0.784	0.747	0.659	0.632	0.760	0.690
All	CART	0.694	0.696	0.691	0.693	0.686	0.688	0.683	0.685
	J48	0.716	0.742	0.662	0.700	0.694	0.703	0.673	0.688
	Naive Bayes	0.737	0.784	0.655	0.714	0.649	0.718	0.491	0.583
	Logistic	0.755	0.775	0.719	0.746	0.678	0.679	0.675	0.677
	SVM	0.763	0.766	0.755	0.761	**0.701**	0.696	0.713	0.705
Best.First	CART	0.730	0.735	0.719	0.727	0.677	0.639	**0.816**	**0.717**
	J48	0.701	0.730	0.640	0.682	0.673	0.639	0.796	0.709
	Naive Bayes	0.759	0.821	0.662	0.733	0.651	**0.727**	0.483	0.581
	Logistic	0.748	0.756	0.734	0.745	0.683	0.676	0.704	0.690
	SVM	0.773	0.757	**0.806**	**0.780**	0.680	0.663	0.732	0.696
Polarity	CART	0.734	0.712	0.784	0.747	0.677	0.639	**0.816**	**0.717**
	J48	0.676	0.684	0.655	0.669	0.673	0.639	0.797	0.709
	Naive Bayes	0.748	0.772	0.705	0.737	0.671	0.688	0.625	0.655
	Logistic	0.748	0.767	0.712	0.739	0.676	0.656	0.742	0.696
	SVM	0.759	0.765	0.748	0.756	0.674	0.637	0.810	0.713
Strength	CART	0.719	0.729	0.698	0.713	0.661	0.653	0.686	0.669
	J48	0.701	0.697	0.712	0.705	0.646	0.628	0.716	0.669
	Naive Bayes	0.766	**0.830**	0.669	0.741	0.636	0.711	0.460	0.558
	Logistic	0.763	0.797	0.705	0.748	0.662	0.688	0.593	0.637
	SVM	**0.777**	0.824	0.705	0.760	0.694	0.683	0.725	0.703
Emotion	CART	0.579	0.634	0.374	0.471	0.586	0.638	0.398	0.490
	J48	0.590	0.647	0.396	0.491	0.575	0.628	0.370	0.465
	Naive Bayes	0.579	0.628	0.388	0.480	0.573	0.647	0.320	0.428
	Logistic	0.583	0.624	0.417	0.500	0.585	0.635	0.402	0.492
	SVM	0.597	0.622	0.496	0.552	0.594	0.627	0.462	0.532

Table 4.9: 10-fold Cross-Validation Subjectivity Classification Performances.

better performance than the reamining one. This result agrees with the results reported by Nielsen [Nie11] where it was shown that the AFINN lexicon was not able to outperform SentiStrength. We can observe also that the detection of polarity is a more difficult task in Sanders than in STS, as was also observed for the subjectivity detection task.

The best tree obtained for polarity classification by the CART algorithm using all the features on the Sanders dataset is shown in Figure 4.4. From the figure we see that top level nodes of the tree correspond to features related to SentiStrength, Sentiment140 and AFINN. These results agree with the information gain values obtained and explains in some manner why these methods are competitive as baselines. The tree also indicates that negative words from the different lexical resources are more useful than the positive ones.

In a similar way as in the subjectivity task, SVM achieves the best results in accuracy and F_1. This fact suggests that there are non-linearities between the features that are successfully tackled by using the RBF kernel. The performance tends also in both datasets to be better for the polarity task than for the subjectivity problem. This is because most of the lexical resources and methods are more focused on the

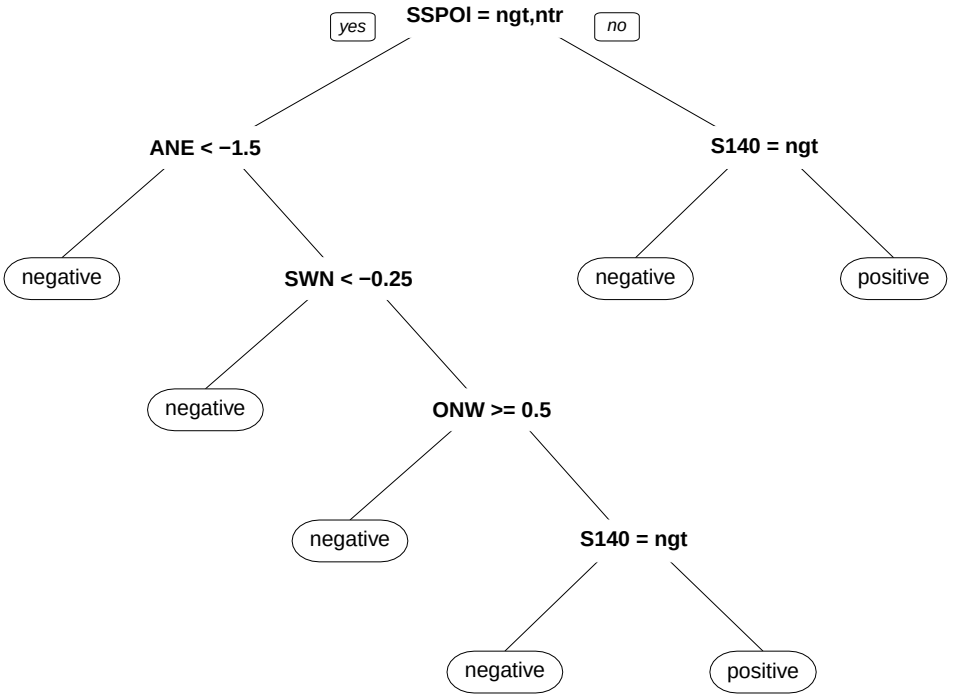


Figure 4.4: Best Tree trained with CART for polarity classification on the Sanders dataset.

detection of polarity rather than detecting subjectivity.

As discussed before, emotion-oriented features tend to have low information gain values and also present a poor classification performance. Therefore, it makes sense to think that emotion-oriented features are not useful for sentiment classification. However, if we consider the performance obtained by RBF SVMs on the Sanders dataset for both classification tasks, we can see that the best accuracies are obtained when all types of features are included. That means that emotion-oriented features are useful for sentiment classification when they are combined with polarity and strength-oriented features in a non-linear fashion.

4.3 Discussions

We present a novel approach for sentiment classification on microblogging messages or short texts based on the combination of several existing lexical resources and sentiment analysis methods. Our experimental validation shows that our classifiers achieve very significant improvements over any single method, outperforming state-of-the-art methods by more than 5% accuracy and F_1 points.

Regarding our research hypothesis formulated in Section 1.2, we can see that the

Dataset		STS				Sanders			
Features	Methods	accuracy	precision	recall	F_1	accuracy	precision	recall	F_1
Baseline.1	SentiStrength	0.777	0.766	0.797	0.781	0.733	0.735	0.729	0.732
Baseline.2	AFINN	0.771	0.804	0.718	0.758	0.713	0.747	0.643	0.691
All	CART	0.788	0.790	0.785	0.788	0.780	0.759	0.821	0.789
	J48	0.788	0.768	0.825	0.796	0.775	0.769	0.786	0.777
	Naive Bayes	0.794	0.757	0.864	0.807	0.774	0.729	0.873	0.794
	Logistic	0.805	0.784	0.842	0.812	**0.801**	0.782	0.834	0.807
	SVM	0.808	**0.808**	0.808	0.808	**0.801**	0.775	0.848	**0.810**
Best.First	CART	0.791	0.775	0.819	0.797	0.789	**0.790**	0.788	0.789
	J48	0.802	0.789	0.825	0.807	0.781	0.778	0.788	0.783
	Naive Bayes	0.811	0.775	**0.876**	0.822	0.788	0.750	0.863	0.802
	Logistic	0.814	0.803	0.831	0.817	0.778	0.765	0.802	0.783
	SVM	**0.816**	0.795	0.853	**0.823**	0.792	0.760	0.854	0.804
Polarity	CART	0.802	0.796	0.814	0.804	0.779	0.736	0.870	0.797
	J48	0.791	0.764	0.842	0.801	0.775	0.728	0.877	0.796
	Naive Bayes	0.805	0.787	0.836	0.811	0.756	0.736	0.800	0.766
	Logistic	0.799	0.779	0.836	0.807	0.786	0.771	0.813	0.791
	SVM	0.799	0.770	0.853	0.810	0.776	0.728	0.882	0.797
Strength	CART	0.780	0.783	0.774	0.778	0.705	0.686	0.757	0.720
	J48	0.777	0.772	0.785	0.779	0.746	0.732	0.775	0.753
	Naive Bayes	0.780	0.746	0.847	0.794	0.762	0.711	0.880	0.787
	Logistic	0.797	0.800	0.791	0.795	0.752	0.747	0.761	0.754
	SVM	0.799	0.805	0.791	0.798	0.779	0.747	0.845	0.793
Emotion	CART	0.684	0.637	0.853	0.729	0.658	0.630	0.766	0.691
	J48	0.681	0.629	0.881	0.734	0.650	0.620	0.777	0.689
	Naive Bayes	0.641	0.599	0.853	0.704	0.654	0.604	**0.891**	0.720
	Logistic	0.661	0.623	0.814	0.706	0.671	0.637	0.795	0.707
	SVM	0.624	0.598	0.757	0.668	0.656	0.624	0.784	0.695

Table 4.10: 10-fold Cross-Validation Polarity Classification Performances.

classification results obtained in this section provide evidence to support the first subordinate research hypothesis about the static properties of social media opinions. The best learned functions obtained for each classification task outperformed the results achieved by the baselines created from isolated methods. In this manner, our results validate the hypothesis that the combinations of different sentiment analysis methods and resources enhances the overall sentiment classification.

The classification results varied significantly from one dataset to another. The manual sentiment classification of tweets is a subjective task that can be biased by the evaluator's perceptions. This fact should serve as a warning call against bold conclusions from inadequate evidence in sentiment classification. It is very important to check beforehand whether the labels in the training dataset correspond to the desired values, and if the training examples are able to capture the sentiment diversity of the target domain.

Chapter 5

Building Opinion Time Series from Twitter

To study the dynamics of Twitter opinions using a computational approach, a number of tasks must be followed that involves extracting opinions from Twitter and transforming them into time series data. In this chapter, the process by which opinion time series are created from Twitter data is presented. The complete process is presented in Figure 5.1 and all the steps are described in the following sections. First of all, the main elements of the process are described. Then, three different tasks required to create the opinion series are presented: topic tracking, sentiment evaluation of tweets, and time series creation.

5.1 Elements of the Process

The main elements that participate in the process in which the opinion time series are built from Twitter are presented as follows:

- **Trackable Topic**: A trackable topic TT_{i} corresponds to an entity which is mentioned multiple times by Twitter users. Each trackable topic is composed of a set of key words.
- **Topic query**: A topic query q_{TT} is a subset of the key words of a certain trackable topic TT. These queries are submitted to the Twitter API in order to retrieve *posts* or *tweets* related to the topic.
- **Post**: A post p is a message submitted by a user in a social media platform. In particular, posts in Twitter are limited to 140 characters and receive the name of "tweets". Considering that this work will be focused on Twitter, for the rest of this work, the terms "post" and "tweet" will be used interchangeably. A post is formed by a sequence of words $W_p = \{w_0, \ldots, w_n\}$. We will assume that all posts retrieved from the Twitter API using a topic query q_{TT} will be related to the topic TT from which the query was created. As posts in Twitter are

timestamped, each post p will have a time period t associated with it.

- **Sentiment Variable**: Through opinion mining methods it is possible to extract *sentiment variables* s_p from a post p. These variables can be numerical or categorical and are calculated from sentiment analysis methods and resources. The sentiment variables and can have three different scopes: polarity, strength, and emotion.
- **Time Period**: The time is represented as a sequence of continuous periods $t_0, \ldots, t_n$, where each t_i corresponds to a time slice. It is important to remark that the time slices are spaced at uniform time intervals. The periods can be spaced using several granularities, i.e., days, months, and years. In this work, we will use daily periods by default.
- **Public Opinion Variable**: A public opinion variable X_t is a measure which reflects a dimension of the public opinion in a certain period t regarding a specific trackable topic TT. These variables are normally calculated by aggregating *sentiment variables* from posts corresponding to the same topic and time period. These variables are detailed in Section 5.4.
- **Opinion Time Series**: An opinion time series is defined as a sequence of continuous values of a certain public opinion variable $X_1, \ldots, X_t, \ldots, X_n$ calculated from the same trackable topic.

5.2 Topic Tracking Tool

The **Topic Tracking Tool** (TTT) is an application responsible for retrieving messages related to a list of topics from Twitter. As mentioned in Section 1.1, Twitter provides a **Search API** for developers which allows the retrieval of tweets related to a query. According to the Search API documentation[1] the API does not allow the retrieval of tweets older than about a week. Moreover, in order to assess the opinion dynamics regarding a certain topic, we require a collection of tweets covering a significant period of time. Indeed, according to Box and Jenkins [BJ94], a minimum of about 50 observations is required in order to fit an adequate ARIMA model. Therefore, the tracking tool must periodically retrieve tweets related to all the topics of the list during a period longer than 50 days. Considering such restrictions, the TTT must satisfy the following requirements:

- It must allow the representation of the topics to be tracked.
- It must provide a mechanism for the generation of queries from the topics.
- It must periodically submit queries through the Twitter API.
- The retrieved tweets must be stored with their respective timestamps.

The tool is developed in the `Java` programming language using the `twitter4j`[2] library to access to the Twitter search API. The most relevant properties of the tool

[1] `https://dev.twitter.com/docs/using-search`
[2] `http://twitter4j.org/en/index.html`

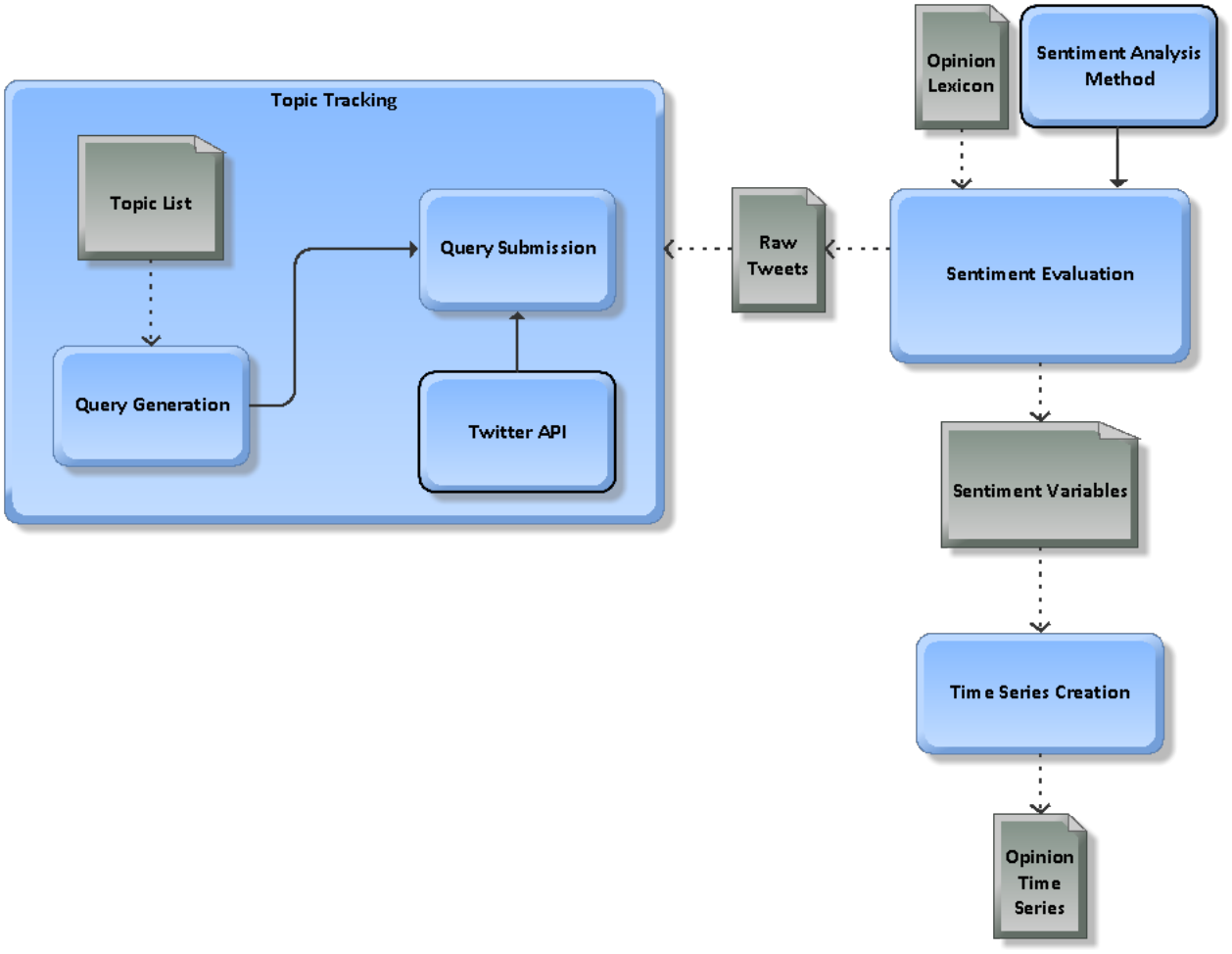


Figure 5.1: Opinion Time Series Building Process.

are presented in the following subsections.

5.2.1 Topic Specification

The topics are represented by different expressions which are formed by one or more words, and are defined in a configuration file of the tracking tool. This file has one line per topic with the following structure:

$$[\text{Topic Name}] : [\text{language}] : [\text{primExp}_1, \ldots, \text{primExp}_n] : [\text{secExp}_1, \ldots, \text{secExp}_m]$$

where:

- Topic Name: is the name of the topic. The tool creates a folder with the name where the retrieved tweets are stored.
- Language: the language of the topic. The tool will set the language as a parameter to the API in order to retrieve tweets written in that language.

- primExp: the list of primary expressions that define the topic. A primary expression must be able to define the topic by itself. Normally, one expression should be enough to define the topic. However, when different expressions are used to represent the same entity, it is recommended to include all of them in the list. For instance, the topic Obama could be represented by the primary expressions *Barack Obama* and *US president*.
- secExp: an optional list of secondary expressions to complement the primary list. These expressions should be used when we want to track specific events related to an entity. A possible secondary expression for the topic Obama could be *election* if we want to retrieve tweets talking about Obama as a candidate.

Some examples of trackable topics are presented as follows:

- `obama:en:obama, barack obama:white house,election`
- `libya:en:libya,#libya:gadaffi,prisioner,rebel`
- `iphone:en:iphone`

We can see that the third example has only one word *iphone* as primary expression and no secondary expressions. In this work we mainly defined the topics in that manner.

5.2.2 Query Submission and Storing

The tracking tool sends queries (q_{TT}) for all the topics defined in the list in a periodically manner. Queries are generated by the random extraction of expressions. Each query is formed by one expression from the primary list and other one from the secondary list. The selected expressions from both lists are sampled uniformly at random. Additionally, retweets which are re-posts of other tweets are discarded. For example, for the topic *Obama* showed in the previous example, the following queries would be generated with equal probability:

- `obama election -RT lang:en`
- `obama white house -RT lang:en`
- `barack obama election -RT lang:en`
- `barack obama white house -RT lang:en`

Then, for the topic *iphone*, the tool would generate always the same query:

- `iphone -RT lang:en`

The tweets are stored according to the following procedure. First of all, for each topic a folder is created where the corresponding tweets are stored. Then, for each tracking day a new file is created in which the tweets are stored in CSV format. We store from the tweets the content, the tweet id, the user id, the timestamp, and the

query. Considering the rate limits of the Twitter Search API[3], the tool sleeps one minute after each query.

We have to remark that in this work, tweets are retrieved using the Twitter search API instead of the Streaming API. We identify two major concerns in relation to the search API. First of all, the API returns only 10% of the hits found for a given query. Secondly, the number of results per query returned by the API is restricted to a maximum of $1,000$ tweets. The first concern implies that the tweets retrieved through the API should be treated as a sample, rather than a full population of the tweets talking about the topic in the specific time period. Then, the second concern indicates that the number of tweets retrieved for popular topics may be truncated to the maximum value. Therefore, our sample of tweets is likely to be a biased sample in which popular topics are underrepresented. Nevertheless, we believe that our sampling approach allows to distinguish between popular and non-popular topics.

Unfortunately, the concerns described above make it very difficult to build a sample of tweets for a group of specific topics in which the topics are represented according to their relative popularity. A possible strategy to address this is discussed below.

First of all, in each time period a sample from the complete stream should be obtained through the Streaming API. Then, all the tweets that do not mention the selected topics should be discarded. The relative popularity of each topic is estimated according to the fraction of tweets that mention the words associated with them. In parallel to this, the topics should be tracked using the Search API in the same way as the topic tracking tool does. The tracked tweets should be sampled according to the estimated relative popularity of their corresponding topics using stratified sampling techniques. Thus, the stratified sample would be a representative sample of the topics in Twitter. However, considering that the Streaming API does not ensure that all of our selected topics will be covered in the provided sample, we believe that the problem of creating a representative sample for a set of given topics using both the Twitter Search and Streaming API is still open.

5.3 Sentiment Evaluation

The sentiment evaluation tool takes all the raw tweets tracked with TTT and calculates sentiment variables for them. The sentiment variables considered in this work are the same as the features calculated in the static analysis in Section 4.1.1. In this way, the sentiment evaluation task uses the lexical resources: OpinionFinder, SentiWordnet, AFINN, and the methods SentiStrength and Sentiment140. These values together with the tweets are stored in new files following the same structure as the raw tweets.

[3] `https://dev.twitter.com/docs/rate-limiting`

5.4 Time Series Building

The latest part of the process consists of aggregating all the sentiment variables for the different topics by days and calculating the **public opinion variables**. The public opinion variables are detailed in Table 5.1. Finally, for each topic we create a multidimensional time series in which each public opinion variable is a different dimension. These series are stored as CSV files, and can be analyzed according to the methodology described in Chapter 6.

Variable Name	Description	Formula
Activity Level (AL)	number of tweets	$\sum p_{\mathrm{i}}$
Average OpinionFinder Positive Words (AOPW)	average value of OPW	$\sum \mathrm{OPW}(p_{\mathrm{i}})/\mathrm{AL}$
Average OpinionFinder Negative Words (AONW)	average value of ONW	$\sum \mathrm{ONW}(p_{\mathrm{i}})/\mathrm{AL}$
Average AFINN Positivity (AAPO)	average value of APO	$\sum \mathrm{APO}(p_{\mathrm{i}})/\mathrm{AL}$
Average AFINN Negativity (AANE)	average value of ANE	$\sum \mathrm{ANE}(p_{\mathrm{i}})/\mathrm{AL}$
Average SWN3 Positiveness (ASWP)	average value of SWP	$\sum \mathrm{SWP}(p_{\mathrm{i}})/\mathrm{AL}$
Average SWN3 Negativeness (ASWN)	average value of SWN	$\sum \mathrm{SWN}(p_{\mathrm{i}})/\mathrm{AL}$
Average SentiStrength Positivity (ASSP)	average value of SSP	$\sum \mathrm{SSP}(p_{\mathrm{i}})/\mathrm{AL}$
Average SentiStrength Negativity (ASSN)	average value of SSNN	$\sum \mathrm{SSN}(p_{\mathrm{i}})/\mathrm{AL}$
Sentiment140 Positiveness (S140POS)	fraction of tweets where S140="pos"	$count_p(\mathrm{S140}(p_{\mathrm{i}}) = pos)/\mathrm{AL}$
Sentiment140 Neutrality (S140NEU)	fraction of tweets where S140="neu"	$count_p(\mathrm{S140}(p_{\mathrm{i}}) = neu)/\mathrm{AL}$
Sentiment140 Negativeness (S140NEG)	fraction of tweets where S140="neg"	$count_p(\mathrm{S140}(p_{\mathrm{i}}) = neu)/\mathrm{AL}$

Table 5.1: Public Opinion Variables.

Chapter 6

Opinion Dynamics in Twitter

The emergence of the social web has allowed researchers to analyze how people feel and react about different things. We define "opinion dynamics" as how these feelings evolve over time. In the previous chapters we have presented a methodology composed of different steps to transform Twitter data into opinion time series. In this chapter, motivated by the idea of understanding the nature of these series, we present an in-depth analysis of their statistical properties. The chapter is organized in two parts.

The first part is a case study in which we explore opinion time series related to a specific event: the U.S. 2008 elections. The opinion time series are created using a single lexical resource. The study is focused on modeling both the expected conditional mean and the expected conditional variance of the time series using ARMA/ARIMA and GARCH models respectively. We discuss how the presence of a conditional variance or volatility in the series limits the long-term forecasting performance of ARMA models.

In the second part of the chapter we expand on the first study by comparing the temporal properties of opinion time series created from different topics using two different sentiment analysis methods. The series are created using the tools introduced in Chapter 5. Four types of topics are considered: politicians, countries, high-tech companies, and long-standing topics. Furthermore, all time series cover the same period of time. The properties of the series considered in the analysis are related to the conditional mean, the forecasting performances and the volatility. Additionally, the relationship between the different topics and their properties is studied using clustering techniques.

6.1 Case Study: U.S. 2008 Elections

In this section, we conduct an experimental exploration of Twitter opinion time series related to the 2008 U.S. Presidential elections. We analyze these time series finding that they present an important volatility factor, which would impede producing accurate long-term predictions.

In this study we considered the public opinion variables **AOPW** and **AONW**, both calculated using the **OpinionFinder** lexicon. These variables are computed by averaging the number of positive and negative words that matched the opinion lexicon within the content of the tweets. The variables are aggregated by periods of one day. Through the remainder of this work we will also refer to variables **AOPW** and **AONW** as **positiveness** and **negativeness**, respectively. The commonly used measure of **polarity** can also be expressed as the difference between positiveness and negativeness.

It is important to note that these variables model temporal occurrence of positive and negative words as two separated processes, as it was also done in [JZSC09]. We believe that this representation facilitates a better observation of the temporal properties of the event and, at least in the datasets used for this study, negative and positive sentiment are mutually uncorrelated as we will show in the next section. Furthermore, as both **AOPW** and **AONW** measures take always positive values they can be transformed into log-return values which are more appropriate for GARCH models and, hence, for assessing the volatility.

6.1.1 Dataset Description

The dataset consists of tweets associated with the U.S. elections of 2008. A thorough description of the data gathering process is detailed in [GA11]. This collection contains 250,000 tweets, published by 20,000 Twitter users from June 1, 2008 to November 11, 2008. All of the tweets are related either to the Democrat candidates **Barack Obama** and **Joe Biden**, or to the Republican ones **John McCain** and **Sarah Pallin**. The Twitter Search API was used using one query per candidacy.

Opinion time series created from this dataset are shown in Figure 6.1. From top to bottom, the first plot shows polarity time series, the second one shows the activity level together with relevant dates related to the event, and the last ones show **AOPW** and **AONW** time series for each candidate. We will denote Obama and McCain opinion time series by $(O.+)$ and $(M.+)$ for **AOPW** (positiveness) and by $(O.-)$ and $(M.-)$ for **AONW** (negativeness) respectively.

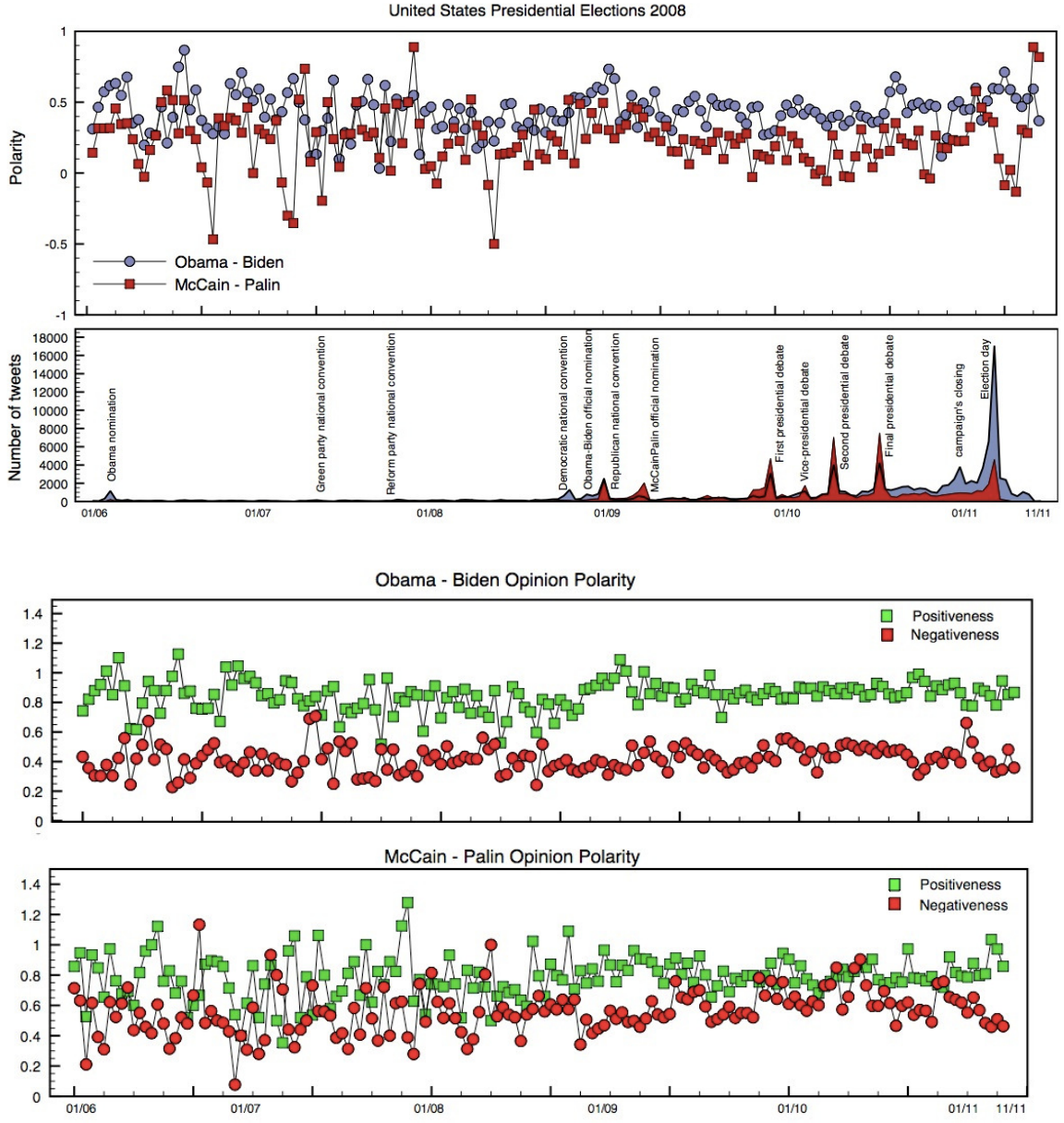
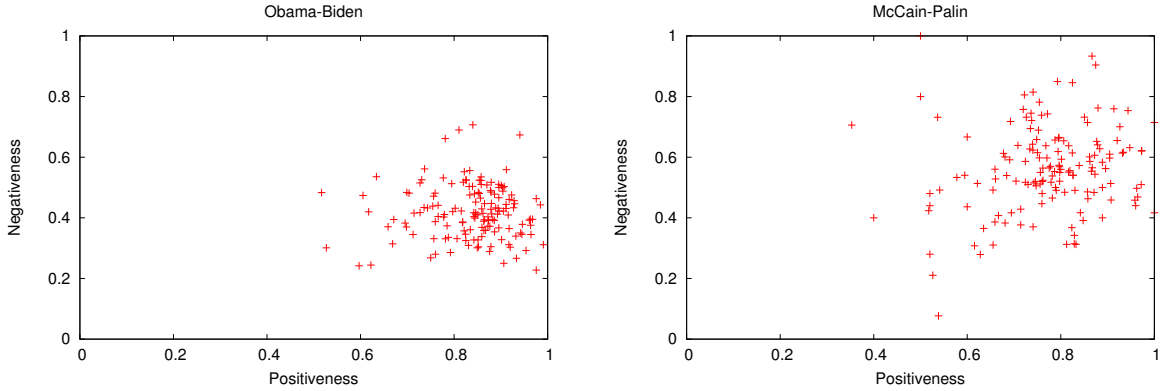


Figure 6.1: Opinion time series for the U.S. Presidential Election of 2008.

6.1.2 Analysis of the Conditional Mean

Following Box and Jenkins methodology presented in Chapter 3, we have analyzed the conditional mean of the time series described in the previous section. We first performed an exploratory analysis of the time series.

Scatter plots between positiveness and negativeness opinion time series are shown in Figure 6.2, (a) Obama-Biden, and (b)McCain-Palin. Pearson correlation between positiveness and negativeness are -0.0698, and 0.0682 for Obama and McCain respectively.



(a) (b)

Figure 6.2: Scattering analysis for the polarity time series of the U.S. Election 2008.

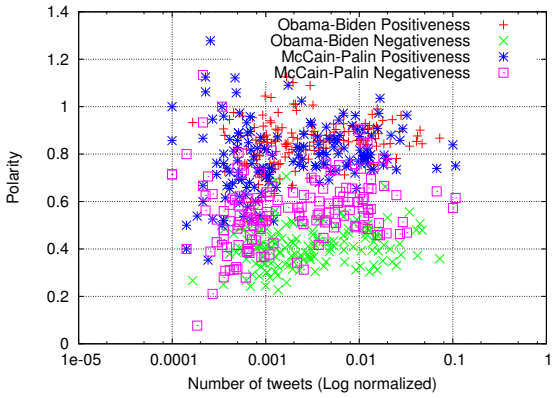


Figure 6.3: Polarity and Activity Scatter Plot.

Cross-correlation coefficients between different series pairs are: $(O.+, M.+) = 0.21$, $(O.-, M.-) = -0.14$, $(O.+, M.-) = 0.17$, and $(O.-, M.+) = 0.01$. Figure 6.3 shows a scatter plot between Twitter activity and opinion polarity, using log axes. Pearson correlation coefficients for $O.+$, $O.-$, $M.+$, and $M.-$ are 0.13, 0.08, 0.08, and 0.11, respectively. Furthermore, we performed Pearson correlation tests between all pairs mentioned above with a significance level of $\alpha = 0.05$. With the exception of pair $(O.+, M.+)$, all p-values obtained are greater than 0.05. These results validate the idea of modeling positiveness and negativeness as separate time series and show us that sentiment measures have no linear relationship with the level of activity in the period.

To check the stationarity of the time series we conduct the Augmented Dickey-Fuller test whose results are shown in Table 6.1. Obtained Augmented Dickey-Fuller (ADF) statistics and p-values allow us to reject the null hypothesis for every opinion time series. Stationarity implies that each time series can be studied without having to differenciate or apply any other transformation to it. Thus, we can apply the Box-Jenkins methodology by fitting stationary models to each time series.

Seasonal patterns are also studied in order to find cyclical periods for the **AOPW**

Time series	ADF test	p-value
O.+	-7.117	< 0.01
O.-	-9.567	< 0.01
M.+	-10.715	< 0.01
M.-	-6.016	< 0.01

Table 6.1: Augmented Dickey-Fuller statistics for trend non-stationarity testing.

and **AONW** in the data. A possible approach is to estimate multiplicative seasonality factors (e.g. day of the week) for each season. As was suggested in [MdR06], we estimated weekly seasonal coefficients for each U.S. elections time series. For each day of the week, we calculate the ratio of actual values divided by predicted values, according to linear trend regression models applied to the series. These values should be close to 1 in the absence of seasonal patterns. As can be seen from Table 6.2, there are coefficients that are no equal to one when we consider a period of one week. Correlograms for each time series shown in Figure 6.4 present similar results when we analyze the 7-th lag. This suggests that seasonal patterns are conditioned to the day of the week.

Day	O.+	O.-	M.+	M.-
Sunday	1.018	0.944	1.055	1.156
Monday	0.961	1.019	0.995	0.942
Tuesday	0.971	0.986	0.975	0.930
Wednesday	1.026	1.063	0.963	0.944
Thursday	1.002	1.020	1.013	0.932
Friday	0.990	0.992	0.987	1.044
Saturday	1.030	0.975	1.007	1.046

Table 6.2: Trend seasonality factors.

Opinion time series can also be smoothed in order to derive a more consistent signal as in [OBRS10]. Some possible smoothing approaches are moving average, moving median, and exponential smoothing, among others. We evaluate the use of moving averages of seven days according to the weekly seasonal patterns described above. It is important to consider that smoothed opinion time series can cause the opinion variable to respond more slowly to recent changes [OBRS10]. Thus, we fit ARMA/ARIMA models to each time series, considering also its smoothed versions.

Model selection was performed by fitting high order models to each time series. In the case of the U.S. elections we consider an additional multiplicative seasonal ARMA model to incorporate seasonality. The use of high order models allows us the observation of the coefficient values for over-parameterized models, identifying coefficients with significant error standard measures. We avoid the use of models with poor fitting properties by conducting model checking with a confidence level of 95%. Model check tests were rejected under this threshold, suggesting to us the presence of ARIMA components. Then we fit ARIMA models to this subset of time series, conducting similar model selection and model evaluation steps. We separate

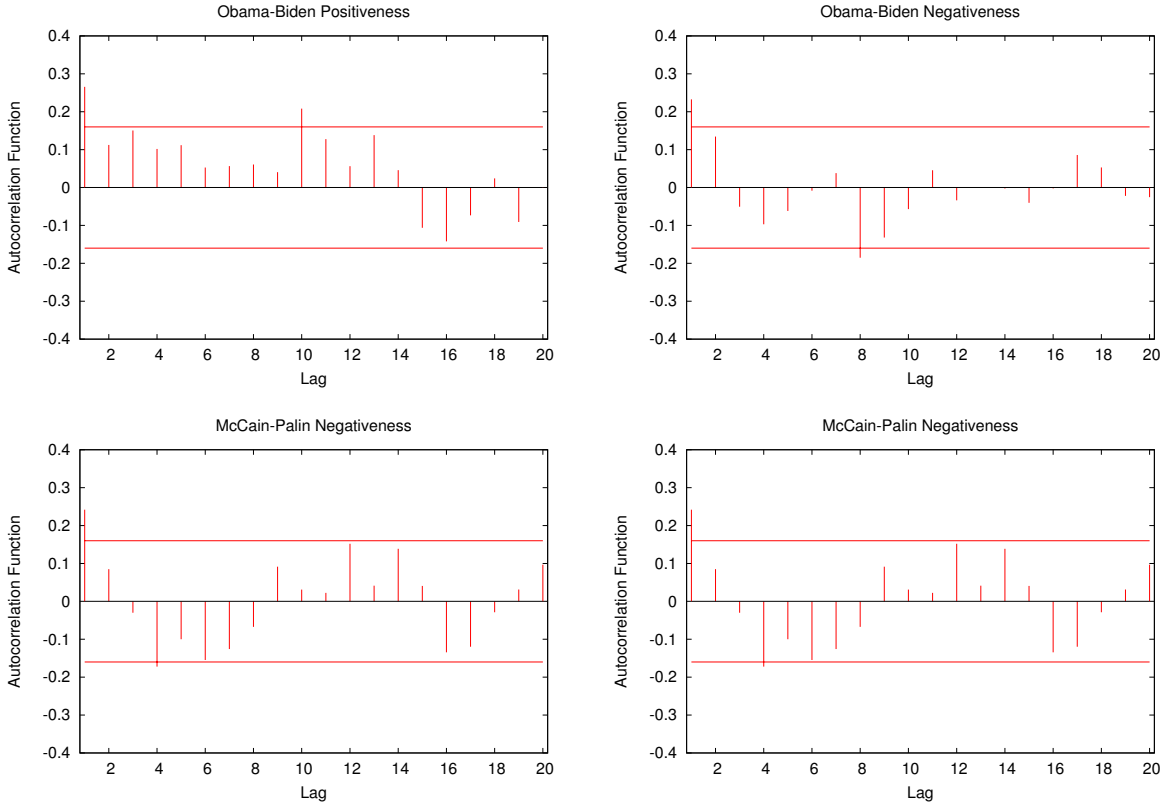


Figure 6.4: Autocorrelation plots of Opinion time series of the U.S. Election 2008.

each time series into two parts, one for model fit/test and a second part for time series forecasting. Model fitting/testing was conducted over the first three months of the U.S. elections.

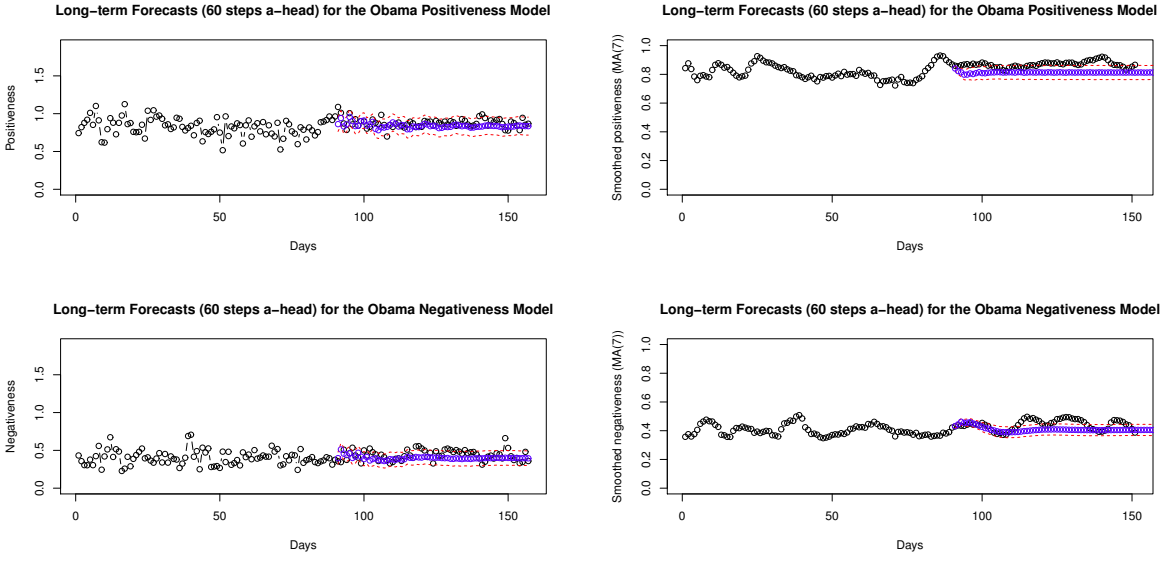


Figure 6.5: Obama Long term forecasts.

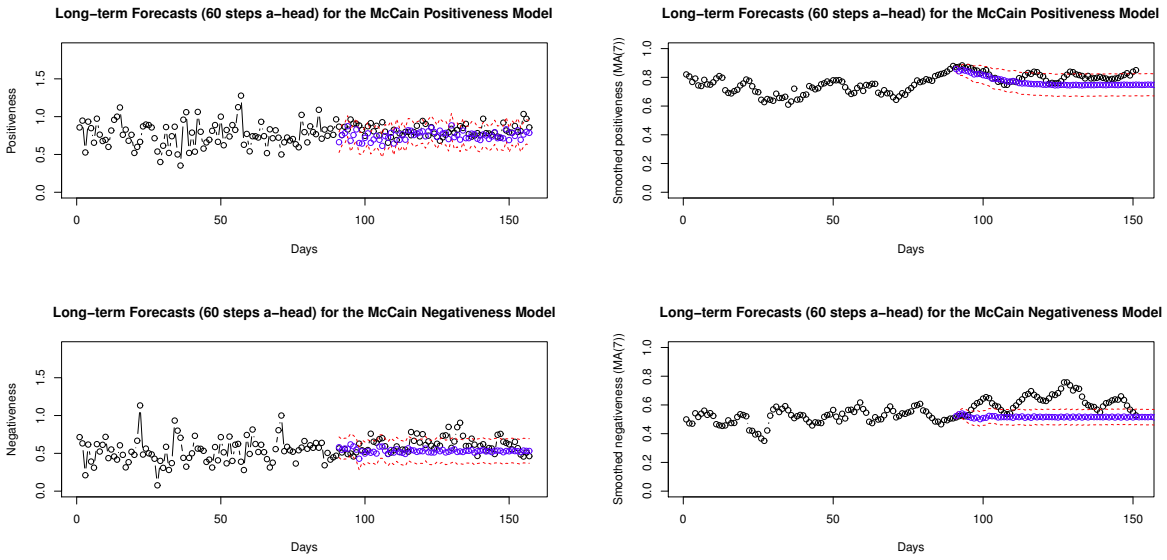


Figure 6.6: McCain Long term forecasts.

Forecasting results obtained from each time series are shown in the first two columns of Figure 6.5. From top to bottom, we show Obama Positiveness, and Negativeness, and McCain Positiveness, and Negativeness forecasts, respectively. The first column shows the results achieved for the original time series and the second one shows results for smoothed versions of each time series, using moving averages of seven days. Actual values are shown with black points and predicted values with blue points. Error margins for predicted values are depicted with segmented lines.

Forecasting results are far from being accurate in the original time series. Forecasts can at least model the mean of future outcomes, but not the variance. For the smoothed versions of the time series, forecasting results are improved in some cases. As will be seen in the following analysis, the difficulty of predicting future outcomes in these time series is due to the presence of volatility.

6.1.3 Volatility Analysis

In order to assess the volatility we used the financial risk management convention of converting the original time series to log return values using the following expression:

$$r_t = \log(\frac{X_t}{X_{t-1}})$$

The return time series are referred to as $R_O.+$ and $R_O.-$ for the **AOPW** and **AONW** of Obama and likewise as $R_M.+$ and $R_M.-$ for McCain respectively. The results of the volatility analysis for the transformed time series is summarized in Table 6.3. Below we describe the tests and measures which were considered in the analysis.

GARCH models are commonly used to represent time series with a zero conditional mean and with a variable variance. Typical time series that meet these conditions are stock price returns and the residuals of ARMA models fitted to volatile time series.

We first checked if the desirable conditions for GARCH modeling were met in the transformed series. The zero-mean condition was tested through a zero-mean t-test, where in all cases we failed to reject the null hypothesis ($\mu = 0$). Considering that volatile time series capture a non-Gaussian fat-tailed distribution [CC09], and that fat-tailed distributions have positive kurtosis, we evaluated the excess of kurtosis of the returns. As shown in the table, these values were positive in all cases.

Normally, a volatile time series presents a higher-order serial dependence structure [CC09]. We studied this property by calculating the autocorrelation values of the squared time series. In Figure 6.7 we can see that all the series have significant autocorrelation coefficients, indicating the presence of a dependence structure for the squared returns. Afterwards, we checked the presence of volatility in the return series through the Box-Ljung statistics applied to the squared returns. This test, when applied to squared returns or residuals, is also known as McLeod-Li test. The test checks whether the first autocorrelations of the squared values are jointly different from zero. As shown in Figure 6.8 and Table 6.3, the average p-values for the lags considered were less than 0.05 in all the time series. These results support the hypothesis that the squared returns present significant autocorrelation coefficients.

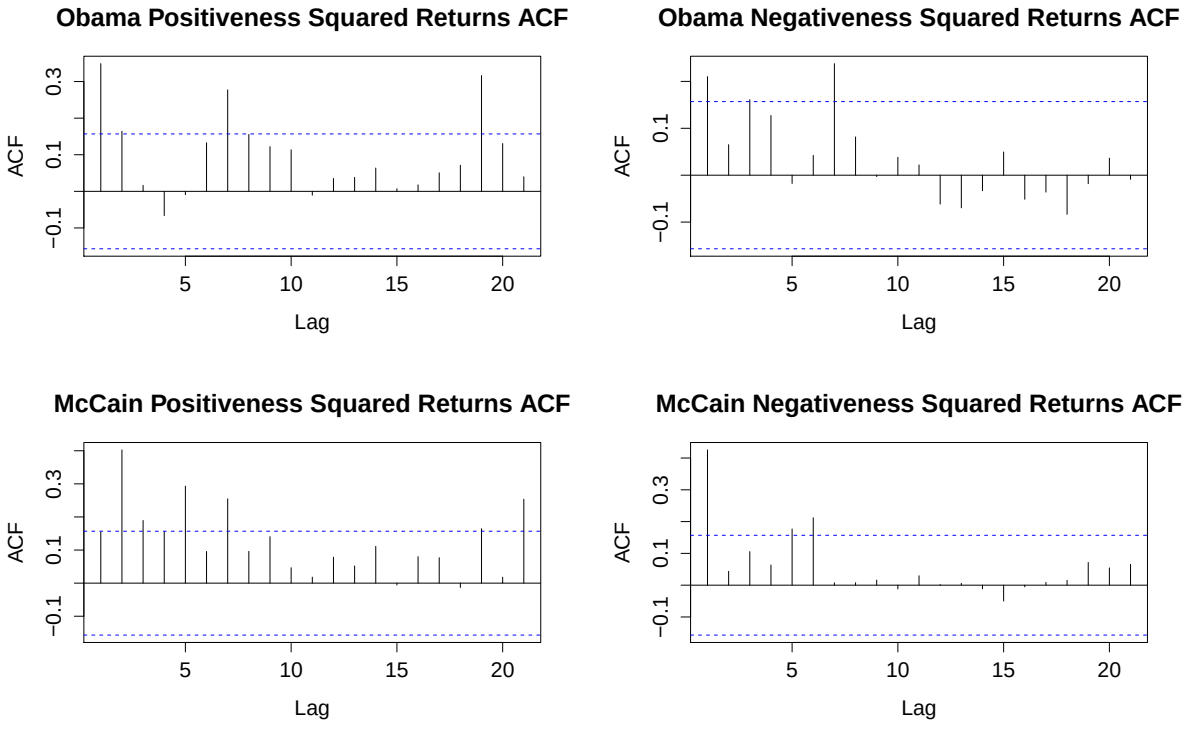


Figure 6.7: Autocorrelation of the squared returns.

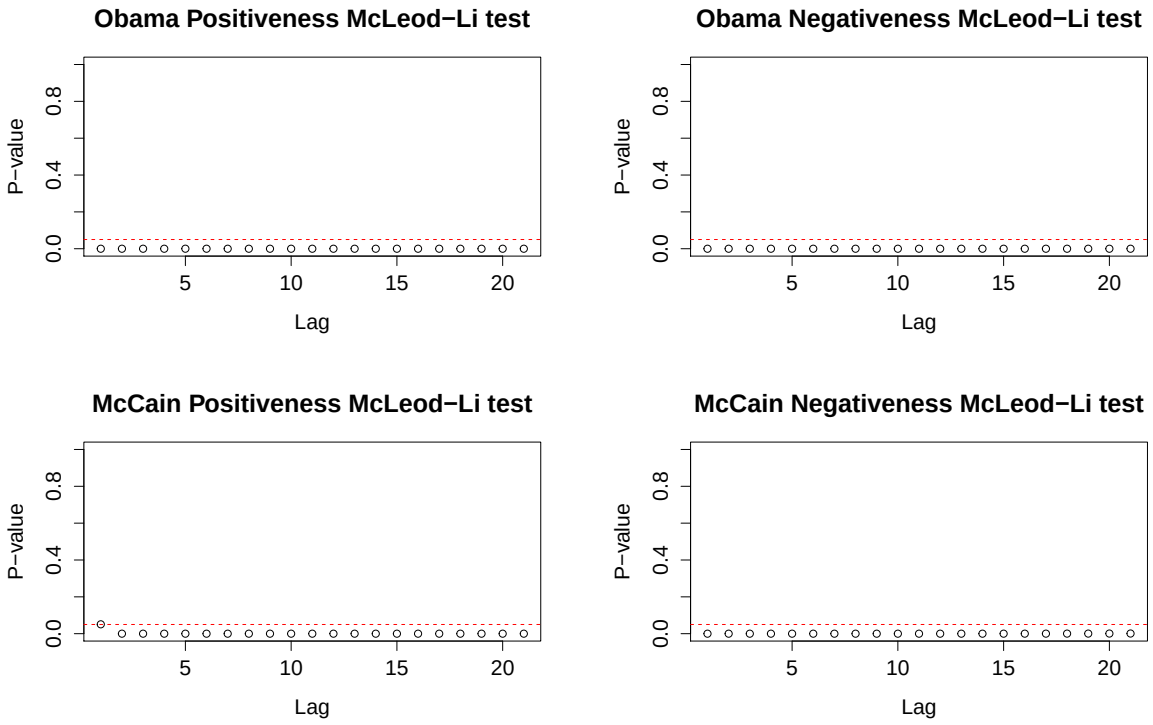


Figure 6.8: McLeod-Li test results for the different time series.

All these results indicate the presence of volatility in the time series. Therefore, we have evidence that our transformed time series are appropriate for GARCH modeling. Moreover, it is important to remark that these conditions were not satisfied in the original series.

We fitted a grid of possible GARCH models to the series varying the orders of q and p from 1 to 3, where in all cases the model that better fitted the data was a GARCH(1,1) model. The quality of the fitted models was assessed by considering the significance of the α_{i} and β_j coefficients through zero-mean t-tests. As it shown in Table 6.3 the p-values obtained from the t-tests applied to the coefficients of the GARCH(1,1) models were all close to zero. Thus, we have statistical evidence that GARCH(1,1) models are appropriate for modeling our transformed time series. Finally the fitted models were used to estimate the conditional variance of the transformed time series.

	$R_O.+$	$R_O.-$	$R_M.+$	$R_M.-$
Kurtosis	2.09	0.978	0.346	1.99
Zero-mean t-test p-value	0.936	0.955	0.999	0.925
McLeod-Li avg. p-value	0.000	0.023	0.002	0.000
α_1 t-test p-value	0.000	0.015	0.004	0.001
β_1 t-test p-value	0.000	0.000	0.000	0.000
Mean Volatility	0.028	0.073	0.058	0.119

Table 6.3: Volatility Analysis of log return time series of the U.S. Election 2008.

Figure 6.9 shows, from top to bottom, the fitted conditional variances of $R_O.+$, $R_O.-$, $R_M.+$ and $R_M.-$ time series, respectively. From the figure we can clearly observe that all volatility time series exhibit calm and volatile periods. An interesting insight derived from these plots is that the volatility or conditional variance of the series tends in all cases to decrease while approaching the election day. This can be interpreted in the following way: at the beginning of the election period people could have been more open to new information and hence, there was more uncertainty about the voting preferences. However, as the election day grew closer, the preferences of the voters became clearer and hence the change in the opinion pattern was reduced.

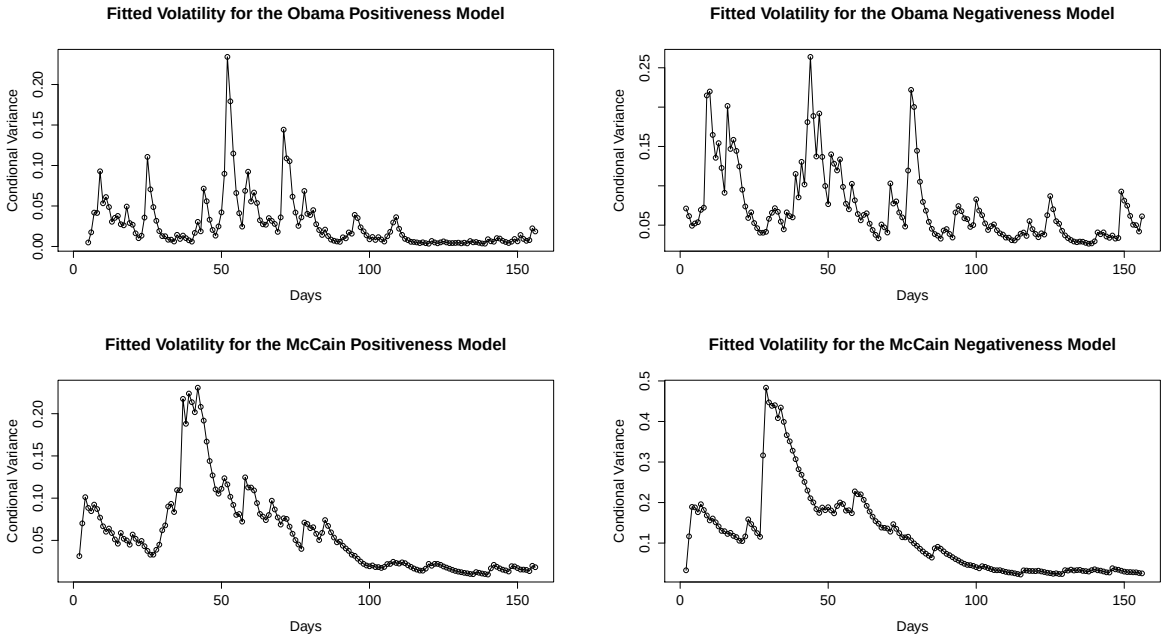


Figure 6.9: Fitted conditional variances for the different time series.

6.1.4 Discussions of the Case Study

The experimental results presented in the previous section show us how opinion time series created from Twitter data regarding the 2008 U.S. elections tend to be volatile. Hence, predictions of future outcomes in the long-term following the Box-Jenkins methodology are limited. A possible approach to reduce the level of uncertainty is to smooth the time series. These transformed time series are probably more appropriate for prediction, yet their outcomes will respond slowly to changes in the opinion pattern. The presence of volatility in our return transformed time series suggests that the past conditional variance can be modeled by estimating the parameters of the GARCH model. Thus, calm and volatile periods could be identified and predicted in the future. This means, that although the conditional mean cannot be predicted in the long term using ARMA/ARIMA models, the volatility could be properly modeled and forecasted using GARCH models.

6.2 Analyzing Opinion Type Series from Different Topics

In this section, we complement the previous study by analyzing Twitter opinion time series related to a variety of topics. We tracked 11 topics from Twitter in the English language using the Topic Tracking Tool described in Section 5.2 over a period of 102 days from 12th April to 22nd July 2012.

The topics were chosen from different types of entities which are frequently discussed in Twitter: politicians, countries, companies and long-standing topics. For the case of politicians, the studied period coincided with the electoral campaign of the U.S. 2012 presidential elections. We retrieved tweets related to both Republican and Democrat candidates *Mitt Romney* and *Barack Obama*. Additionally, we tracked tweets on the U.K prime minister *David Cameron* and the U.S. Democrat politician *Hillary Clinton*. Regarding the tracked countries, we selected countries which face either an internal or external conflict situation: *Iran*, *Israel* and *North Korea*. We also tracked tweets related to two of the most influential high-tech companies in the world: *Google* and *Facebook*. The tracked period coincided with the debut of Facebook in the NASDAQ stock market on 18 May 2012. Finally, we selected two long-standing topics that are constantly discussed by the public: *Abortion* and *Global Warning*.

The selected topics along with the number of tweets, the average number of tweets per day and the standard deviation are shown in Table 6.4. We can see that topics Facebook, Google, Obama, and Romney received a greater amount of tweets in comparison to the others. This indicates, that topics related to newsworthy events such as an election or the debut of a company in the stock market are commented on more frequently in Twitter.

Type	Topic	Number of Tweets	Mean	Standard Deviation
Politician	David Cameron	92,385	905.7	805.9
	Hillary Clinton	29,738	291.5	169.5
	Barack Obama	965,925	9,469.9	903.8
	Mitt Romney	774,931	7,597.4	1,230.7
Country	Iran	262,566	2,574.2	533
	Israel	582,360	5,709.4	1,041.2
	North Korea	73,570	721.3	871.3
Company	Facebook	1,047,619	10,270.8	1,080.2
	Google	976,129	9,569.9	971.9
Long-standing	Abortion	356,103	3,491.2	703.1
	Global Warming	146,058	1,431.9	429.0

Table 6.4: Number of Tweets for each Topic.

The behavior of the Activity Level (number of tweets) time series is compared for the different topics in Figure 6.10. Broadly speaking, the figure is consistent with

the data reported in Table 6.4. We can observe a strong cyclical pattern in most popular topics. This likely occurs because the number of queries performed by the tracking tool for each topic could vary from one day to another.

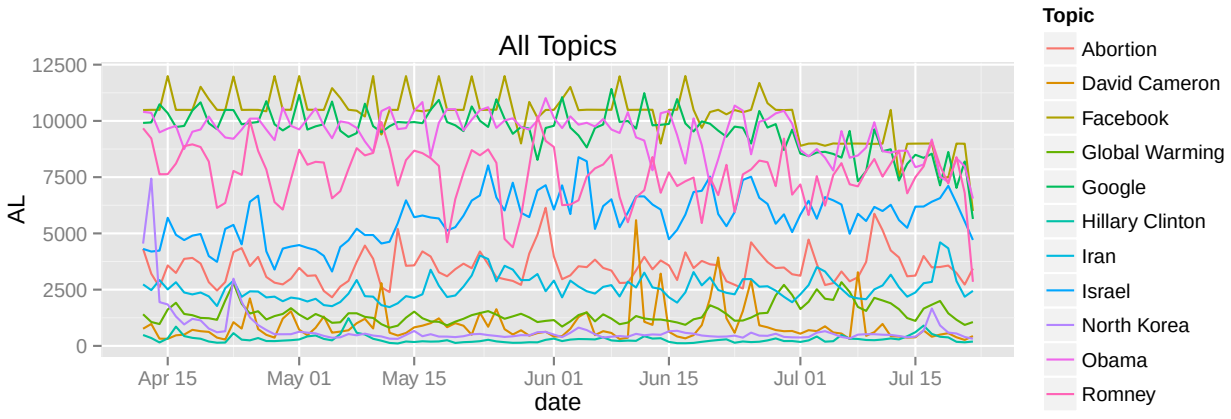


Figure 6.10: Activity Level of different Opinion Time Series.

6.2.1 Exploring Opinion Time Series

Following the process described in Chapter 5 we created a number of opinion time series considering different public opinion variables from the tweets presented in the previous section. We have to keep in mind that public opinion variables can be created using different sentiment analysis methods and lexical resources. These methods and resources along with the sentiment variables that can be extracted from them were described in the static analysis in Chapter 4. In that chapter, we also analyzed the usefulness of these sentiment variables in discriminating real Twitter opinions using the information gain criterion. This analysis leads to conclude that methods *SentiStrength* and *Sentiment140* produce more informative variables than the others. Moreover, we believe that opinion time series created from informative sentiment variables may represent real public opinion in a better manner. Thus, we focus our analysis on the time series created from both SentiStrength and Sentiment140 methods. The other opinion time series calculated from the rest of the sentiment analysis resources are presented in the Appendix.

The series ASSP and ASSN, are calculated by averaging the positive and negative outputs from SentiStrength method, and the series S140POS and S140NEG are calculated by counting the fraction of positive and negative tweets according to Sentiment140.

The time series ASSP, ASSN, S140POS, and S140NEG regarding companies Facebook and Google, are presented in Figure 6.11. For both companies the positive time series ASSP and S140POS present a similar behavior. Google has a significant peak in the second half of May in both positive series. Facebook presents a very significant increase at the end of the tracking period. For the negative series the gap

between both companies becomes wider. As in ASSN, Facebook presents as more negative than Google, the fraction of negative tweets according to Sentiment140 is greater for Facebook than for Google. These results suggest that the methods SentiStrength and Sentiment140 capture an equivalent temporal sentiment pattern for the companies opinion time series.

Regarding the countries, as is shown in Figure 6.12, Israel is the most positive and less negative time series according to both methods. With respect to the remaining countries, Iran and North Korea, despite of being less positive and more negative than Israel, there is no consensus between the methods about which country is more positive or less negative than the other. If we consider that our time series are created from English tweets, it seems reasonable that countries such as Iran and North Korea, which are in conflict with the U.S., present both a more negative and less positive behavior than Israel.

The opinion time series calculated for different politicians are presented in Figure 6.13. We have to recall from the previous analysis that topics Obama and Romney, which correspond to the U.S. 2012 candidates, are much more popular than the other politicians, David Cameron and Hillary Clinton. As the figure indicates, the candidate Barack Obama tends to be more positive than his opponent Mitt Romney according to ASSP and S140POS time series. In contrast to this, negative opinions are much more tied for both candidates. Regarding the other politicians Clinton and Cameron, the opinion time series present strong fluctuations. We believe that this is because of the number of tweets associated with these politicians is much smaller than for the U.S. candidates. There is a basic statistical principle that says that large samples are more stable than small samples and tend to fluctuate less from the long-term average. Extreme views about both U.S. candidates are likely mitigated by the predominance of neutral tweets or moderate opinions.

Finally, opinion time series for the two long-standing topics Abortion and Global Warming are shown in Figure 6.14. There is no clear sentiment pattern between the four opinion time series. The only outstanding observation is that the topic Abortion is much more negative than Global Warming with respect to SentiStrength. However, this pattern is not observed in the S140NEG time series.

Figure 6.11: Company Opinion Time Series.

Figure 6.12: Country Opinion Time Series.

Figure 6.13: Politician Opinion Time Series.

Figure 6.14: Long-stand Opinion Time Series.

6.2.2 Studying the Conditional Mean

In the following analysis we study some statistical properties related to the conditional mean of the opinion time series. The first properties that we calculate are the mean and the variance of the data. Afterwards, we study the significance of the first k autocorrelation coefficients using the Box–Pierce or Ljung–Box portmanteau Q statistic. This test is used to determine whether a time series has a linear dependence structure. Under the null hypothesis H_0, the test states that the data is independently distributed, and consequently the first k autocorrelation coefficients of the time series are zero. On the other hand, the alternative hypothesis H_a states that the data is not independently distributed, implying that the time series has significant autocorrelation coefficients at the first k lags. The advantage of this test is that it allows to study multiple autocorrelation coefficients simultaneously, in contrast to other approaches in which coefficients are tested one by one. Normally, this test is applied to the residuals of a fitted ARMA or ARIMA models. In our case, we apply the test to the original data in order to check whether an opinion time series has a temporal dependence structure. We consider the firsts seven lags to assess the weekly temporal structure.

Finally, we analyze the presence of a linear trend in the time series through a linear regression. We create a vector of times t composed of a sequence of numbers $1, 2, \ldots, n$ with the same length as the corresponding time series. We model the mean μ of time series according to the following linear model:

$$\mu_t = \beta_0 + \beta_1 t. \tag{6.1}$$

We fit a linear regression to the data, estimating parameters β_0 and β_1 from the model. The parameter β_1 represents the slope of the fitted line and it is used to quantify the trend. Afterwards, we test if the slope is significantly different from zero using a t-test.

The mean, the variance, the resulting p-value of the Box-Pierce test, and the p-value of the trend t-test for opinion time series: ASSP, ASSN, S140POS and S140NEG are shown in Tables 6.5, 6.6, 6.7, and 6.8 respectively. The values mean and variance are consistent with the time series plots presented before. An interesting insight is that topics with the higher variances for ASSP David Cameron and Hillary Clinton, also present the higher variance for ASSN. Thus, an ordinal correspondence between the variances of both positive and negative opinion is observed for SentiStrength opinion time series.

Regarding the p-values from the Box–Pierce test, we have to remember that the p-value measures the evidence against the null hypothesis. The closer to zero this value is, the stronger the evidence we have to reject the null hypothesis that the first k autocorrelations are zero. We can see from the tables, that most of these values are less than the significance level of 0.05, especially for negative time series

ASSN and S140NEG. This means that several opinion time series present a temporal dependence structure. However, for company topics Facebook and Google we failed to reject the null hypothesis in almost all the opinion series. This result suggests us that the daily observations in the opinion time series for companies Google and Facebook are somewhat independent. This could occur because these topics, besides being popular, are too general. General topics may likely be composed of different sub-topics or aspects. For instance, a tweet about the topic Google could talk about Google's stock price, Google's search engine, or the company itself. If all the opinions about these aspects of Google are merged in the times series it is hard to observe a significant temporal structure in the data.

In the same way as the p-values from Box–Pierce test, the closer to zero the p-values of the trend t-test are, the more likely it is that the slope coefficient from linear trend is different from zero. The resulting p-values from this test are shown in the fourth column of Tables 6.5, 6.6, 6.7, and 6.8. No clear results are obtained from this test. Although several topics present a linear trend for a certain public opinion variable, this pattern is not necessarily supported by the other time series.

Topic	Mean	Var	Box.test	Trend.pvalue
Abortion	1.537	0.001	0.006	0.227
David Cameron	1.579	0.007	0.018	0.028
Facebook	1.643	0.001	0.995	0.327
Global Warming	1.512	0.001	0.000	0.000
Google	1.629	0.001	0.142	0.268
Hillary Clinton	1.541	0.005	0.201	0.006
Iran	1.470	0.001	0.138	0.155
Israel	1.665	0.003	0.000	0.187
North Korea	1.542	0.003	0.000	0.000
Obama	1.607	0.001	0.000	0.547
Romney	1.560	0.000	0.042	0.385

Table 6.5: ASSP Conditional Mean Properties.

Topic	Mean	Var	Box.test	Trend.pvalue
Abortion	-1.705	0.004	0.000	0.044
David Cameron	-1.569	0.010	0.000	0.287
Facebook	-1.456	0.001	0.000	0.000
Global Warming	-1.480	0.001	0.294	0.914
Google	-1.310	0.000	0.034	0.017
Hillary Clinton	-1.418	0.013	0.010	0.001
Iran	-1.611	0.005	0.000	0.056
Israel	-1.435	0.004	0.000	0.123
North Korea	-1.504	0.010	0.000	0.000
Obama	-1.560	0.001	0.000	0.050
Romney	-1.576	0.002	0.000	0.448

Table 6.6: ASSN Conditional Mean Properties.

Topic	Mean	Var	Box.test	Trend.pvalue
Abortion	0.089	0.000	0.000	0.001
David Cameron	0.143	0.001	0.139	0.823
Facebook	0.179	0.000	0.296	0.053
Global Warming	0.100	0.000	0.000	0.087
Google	0.188	0.000	0.528	0.008
Hillary Clinton	0.102	0.001	0.002	0.005
Iran	0.117	0.000	0.600	0.296
Israel	0.204	0.001	0.000	0.004
North Korea	0.094	0.001	0.000	0.055
Obama	0.124	0.000	0.001	0.001
Romney	0.111	0.000	0.000	0.094

Table 6.7: S140POS Conditional Mean Properties.

Topic	Mean	Var	Box.test	Trend.pvalue
Abortion	0.162	0.000	0.000	0.148
David Cameron	0.145	0.001	0.003	0.004
Facebook	0.151	0.000	0.000	0.000
Global Warming	0.166	0.000	0.000	0.016
Google	0.103	0.000	0.282	0.251
Hillary Clinton	0.086	0.001	0.065	0.165
Iran	0.114	0.000	0.000	0.001
Israel	0.102	0.000	0.002	0.020
North Korea	0.123	0.000	0.001	0.009
Obama	0.139	0.002	1.000	0.028
Romney	0.127	0.000	0.000	0.149

Table 6.8: S140NEG Conditional Mean Properties.

The previous analysis showed the difficulty in finding interesting patterns regarding the properties of the conditional mean from different opinion time series. With the aim of tackling this problem, we propose a multidimensional representation of each time series using properties related to the conditional mean as numerical dimensions. This representation allows the comparison of different opinion time series using the euclidean distance measure. The dimensions included in the representation are the following:

- **Mean**: is the sample mean of the time series.
- **Variance**: is the sample variance of the time series.
- **Skewness**: is the sample skewness which is used as a measure of asymmetry of the distribution.
- **Kurtosis**: is the sample kurtosis and in the same way as the skewness is used to describe the shape of the distribution. The kurtosis measures the "fatness" of tails of the distribution. High kurtosis values correspond to heavy-tailed distributions. Conversely, low kurtosis values suggest thin-tail distributions.

- **Partial autocorrelation coefficients (PACF)**: we consider the firsts seven partial autocorrelation coefficients. We used these values instead of the simple autocorrelations, because PACF values are more independent of one other.
- **Linear Trend Slope**: this value is the estimated β_1 coefficient from the linear trend regression.

To avoid having dimensions with large values dominating the distance values between different time series, we scaled each dimension to have a zero mean and unit variance using the following transformation:

$$x' = \frac{x - \overline{x}}{\sigma},$$

where $\overline{x}$ is the sample mean and σ is the sample standard deviation. Using the proposed multidimensional representation of the time series, we conducted a cluster analysis to investigate whether groups of topics with similar temporal opinion patterns are formed. Specifically, we used an agglomerative hierarchical clustering technique with average linkage. The algorithm starts assigning each data point to one single cluster. Then, iteratively closest clusters are merged into single new clusters. In order to compare a pair of clusters, the average linkage approach considers the average pairwise distance of all pairs of elements within the clusters.

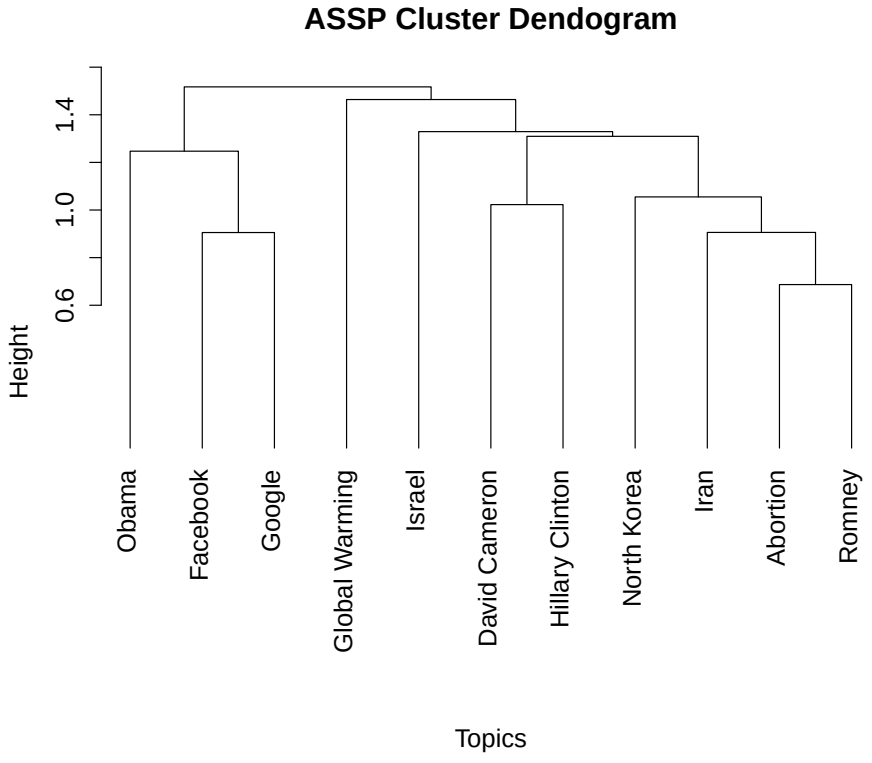


Figure 6.15: ASSP Dendogram.

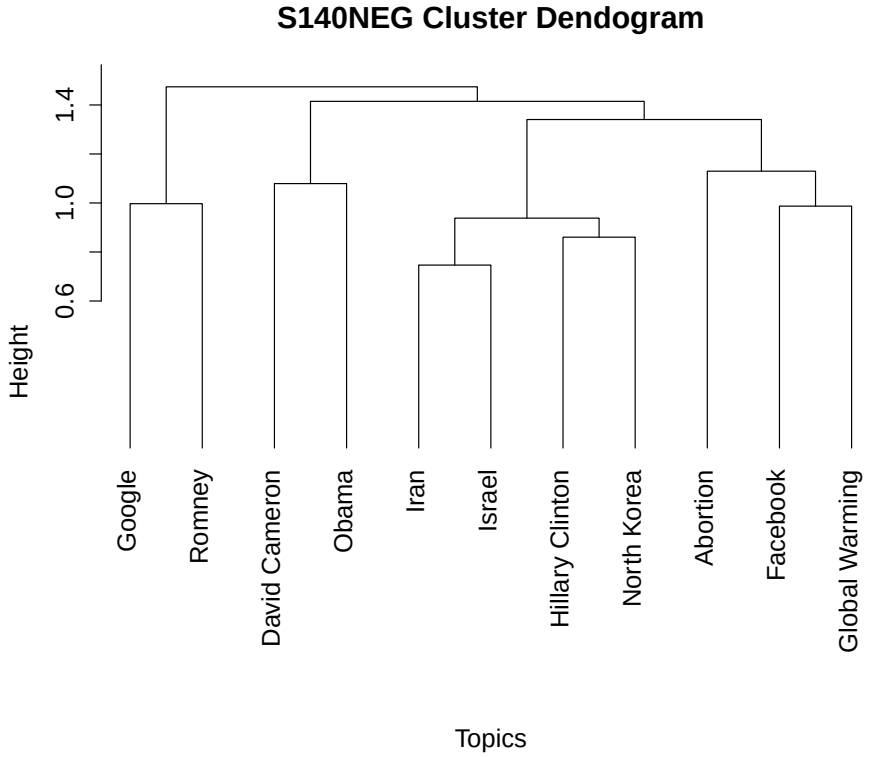


Figure 6.16: S140NEG Dendogram.

The resulted clusters for opinion time series ASSP and S140NEG are displayed as dendrograms in Figures 6.15, and 6.16, respectively. As shown in the ASSP dendrogram, companies *Facebook* and *Google* are grouped together. In a similar manner, politicians *David Cameron* and *Hillary Clinton* belong to the same cluster. Another interesting insight, is that countries *Iran* and *North Korea* are relatively close to each other.

In the second dendrogram (Fig. 6.16) calculated from the S140NEG time series, we can see that topics are grouped in a completely different manner. We also found some patterns, such as that the U.S. President *Barack Obama* and British Prime Minister *David Cameron* are very close to each other. Likewise, the countries *Iran* and *Irael* belong to the same pattern, and both long-standing topics *Abortion* and *Global Warming* are close to each other. Despite the cases discussed above in which topics from the same category appeared very close to each other, there are many other cases in which topics with no apparent relationship clustered together. Therefore, we do not have enough evidence to state that the opinions on topics related to similar events share common temporal properties.

6.2.3 Model Fitting and Forecasting

In this section we investigate the predictability of the opinion time series from past data using ARMA/ARIMA models. For each time series, we fitted an ARMA/ARIMA model using the first 92 observations. Afterwards, each model was employed to forecast the remaining 10 observations using multi-step-ahead prediction. We used the stepwise selection procedure proposed by Hyndman and Khandakar in [HK08] in order to choose the optimal ARIMA models. This method is implemented in the `R` pacakge `forecast`[1] under the name of `auto.arima`. The algorithm conducts a search over possible ARIMA models and returns the best model according to a certain criteria. In this work, we used the Akaike information criterion. We evaluated the forecasting performance of the fitted ARIMA models using the measures **MAE**, **MAPE**, **MSE**, and **RMSE** presented in Section 3.3.

The obtained ARIMA models returned by the `auto.arima` algorithm together with their forecast results are shown in Tables 6.9, 6.10, 6.11, and 6.12 for opinion time series ASSP, ASSN, S140POS, and S140NEG respectively.

Topic	Model	MAE	MAPE	MSE	RMSE
Abortion	ARIMA(0,1,1)	0.016	1.035	0.000	0.022
David Cameron	ARIMA(1,0,0)	0.056	3.551	0.005	0.074
Facebook	ARIMA(0,1,1)	0.023	1.396	0.004	0.066
Global Warming	ARIMA(2,1,3)	0.024	1.577	0.001	0.029
Google	ARIMA(0,0,1)	0.011	0.697	0.000	0.014
Hillary Clinton	ARIMA(0,0,2)	0.062	4.020	0.005	0.071
Iran	ARIMA(0,1,2)	0.021	1.448	0.001	0.027
Israel	ARIMA(1,0,0)	0.027	1.597	0.001	0.034
North Korea	ARIMA(0,1,1)	0.046	2.950	0.004	0.060
Obama	ARIMA(1,0,0)	0.023	1.443	0.001	0.027
Romney	ARIMA(2,1,2)	0.018	1.138	0.000	0.021

Table 6.9: ASSP Forecasting Results.

As stated in Chapter 3, the order of ARIMA models depends on three values: p, d, and q, where p refers to the order of the autoregressive part, d refers to the number of times the time series has to be differenced to obtain a stationary series, and q refers to the order of the moving average part. Furthermore, the higher the orders of the autoregressive or moving average part of the model are, more parameters must be estimated in order to fit the corresponding model.

Our results show that most of the fitted ARIMA models have few parameters. Indeed, the maximum orders for parameters p, d, and q were 3, 1, and 3 respectively. This implies that the time series are somehow independent from very old observations. Regarding the values of parameters d, they are greater than zero for the majority of the time series related to long-standing topic. This means that our long-standing topics tend to produce non-stationarity opinion time series.

[1] `http://cran.r-project.org/web/packages/forecast/`

Topic	Model	MAE	MAPE	MSE	RMSE
Abortion	ARIMA(1,1,1)	0.049	2.897	0.004	0.064
David Cameron	ARIMA(0,1,2)	0.044	2.797	0.003	0.055
Facebook	ARIMA(0,1,1)	0.014	0.956	0.001	0.023
Global Warming	ARIMA(1,0,1)	0.024	1.650	0.001	0.035
Google	ARIMA(1,0,0)	0.015	1.177	0.000	0.019
Hillary Clinton	ARIMA(0,0,1)	0.114	8.173	0.032	0.178
Iran	ARIMA(1,0,0)	0.080	4.999	0.013	0.116
Israel	ARIMA(2,1,2)	0.051	3.555	0.007	0.082
North Korea	ARIMA(0,1,2)	0.052	3.544	0.004	0.059
Obama	ARIMA(1,1,1)	0.029	1.853	0.002	0.041
Romney	ARIMA(1,1,2)	0.057	3.702	0.005	0.070

Table 6.10: ASSN Forecasting Results.

Topic	Model	MAE	MAPE	MSE	RMSE
Abortion	ARIMA(0,1,1)	0.008	10.367	0.000	0.012
David Cameron	ARIMA(1,0,0)	0.022	15.423	0.001	0.027
Facebook	ARIMA(0,1,1)	0.016	9.714	0.002	0.040
Global Warming	ARIMA(0,1,2)	0.012	12.065	0.000	0.016
Google	ARIMA(1,0,0)	0.007	3.604	0.000	0.008
Hillary Clinton	ARIMA(1,0,1)	0.017	17.253	0.000	0.021
Iran	ARIMA(1,0,0)	0.008	6.528	0.000	0.010
Israel	ARIMA(0,0,1)	0.012	5.780	0.000	0.016
North Korea	ARIMA(0,1,1)	0.014	14.844	0.000	0.018
Obama	ARIMA(1,0,0)	0.012	9.523	0.000	0.017
Romney	ARIMA(0,1,2)	0.012	10.476	0.000	0.014

Table 6.11: S140POS Forecasting Results.

Topic	Model	MAE	MAPE	MSE	RMSE
Abortion	ARIMA(1,1,1)	0.011	6.924	0.000	0.013
David Cameron	ARIMA(0,0,1)	0.027	16.512	0.001	0.034
Facebook	ARIMA(0,1,1)	0.008	4.907	0.000	0.010
Global Warming	ARIMA(2,1,1)	0.022	13.441	0.001	0.025
Google	ARIMA(0,0,0)	0.005	4.498	0.000	0.006
Hillary Clinton	ARIMA(0,0,1)	0.016	18.233	0.000	0.021
Iran	ARIMA(1,0,0)	0.010	9.291	0.000	0.013
Israel	ARIMA(1,0,0)	0.008	7.808	0.000	0.012
North Korea	ARIMA(3,0,0)	0.020	15.759	0.000	0.022
Obama	ARIMA(0,0,0)	0.021	15.333	0.004	0.062
Romney	ARIMA(1,0,3)	0.009	7.638	0.000	0.011

Table 6.12: S140NEG Forecasting Results.

In relation to the forecast performances, we can see that the topics related to politicians David Cameron and Hillary Clinton have the poorest results. Conversely,

Google is the topic in which the corresponding ARIMA models achieved the best forecasting performance. We can see from the previous analysis of the conditional mean, that the variance of the different opinion time series has a close relation to the performance obtained for the corresponding ARIMA model. While for topics with high forecasting errors, like Hillary Clinton and David Cameron, the series have high variability, for topics with low errors, like Google, where the variance is low. These results support the intuition that time series with high variability are harder to predict.

6.2.4 Volatility Analysis

The last part of the analysis consists of the study of the conditional variance or volatility of the opinion time series. The aim of the study is to determine whether the opinion time series created from different topics exhibit a time-dependent variance, and whether GARCH models are appropiate for the observed data. We conducted a very similar analysis to the one carried out above in Section 6.1.3 for the U.S. 2008 elections. In the same way as in the case study, all the time series were transformed to log return values. Before discussing our results, we want to remark the properties that our transformed log returns time series should meet in order to state that they correspond to a volatile GARCH stochastic process:

- The data has a zero mean and hence we should fail to reject the null hypothesis that the mean is zero. In this manner, the p-value of a zero-mean t-test should be close to 1. Otherwise, it would be better to study the residuals of a fitted ARIMA model.
- The data presents a fat-tailed distribution and hence a positive kurtosis. That is, extreme values are observed more often than in a normal distribution.
- The conditional variance has a temporal dependence structure and hence the squared log returns should have significant autocorrelation coefficients. In this way, we expect low p-values from the McLeod-Li test.
- If we fit a GARCH(1,1) model to the log returns series, at least one of the parameters α_1 or β_1 should be significant. For instance, if the data was generated from an ARCH(1) model which is equivalent to a GARCH(1,0), while the α_1 coefficient from the fitted GARCH(1,1) model should be significant, β_1 should not. Likewise, if the data was generated by a higher-order GARCH(p, q) $(p, q > 1)$ process, the coefficients α_1 or β_1 should be both significant. In this way, we should reject the null hypothesis of a zero-mean test for coefficients α_1 or β_1, and hence at least one of both p-values should be close to zero.

In order to study the properties described above, we conducted a one-sample t-test with a zero mean under the null hypothesis. The resulting p-value of this test is refered to as **ret.ttets**. We evaluated the excess of kurtosis (**ret.kurt**) for the transformed series. Afterwards, we applied the McLeod-Li test to the returns considering up to the first ten lags, averaging the resulting p-values (**mean.mc.test**).

Finally, we fitted a GARCH(1,1) model to each transformed time series using `fGarch` R package[2], and tested the significance of the coefficients α_0, α_1, and β_1 through zero-mean t-tests.

The results obtained for our volatility analysis experiments are presented in Tables 6.13, 6.14, 6.15, and 6.16 for opinion time series ASSP, ASSN, S140POS, and S140NEG respectively.

Topic	ret.ttest	ret.kurt	mean.mc.test	α_0.pval	α_1.pval	β_1.pval
Abortion	0.992	7.014	0.002	0.315	0.055	1.000
David Cameron	0.980	5.756	0.609	0.576	0.453	0.922
Facebook	0.437	36.342	0.999	NA	NA	NA
Global Warming	0.770	0.253	0.097	0.394	0.283	0.027
Google	0.963	9.986	0.126	0.000	0.000	0.966
Hillary Clinton	0.953	-0.566	0.432	0.514	0.401	0.949
Iran	0.994	-0.464	0.124	0.660	0.742	0.980
Israel	0.939	1.700	0.000	0.164	0.049	0.004
North Korea	0.970	-0.142	0.109	0.408	0.319	0.280
Obama	0.990	1.401	0.516	0.324	0.442	0.668
Romney	0.818	0.130	0.537	0.349	0.401	0.114

Table 6.13: ASSP Volatility Results.

Topic	ret.ttest	ret.kurt	mean.mc.test	α_0.pval	α_1.pval	β_1.pval
Abortion	0.964	0.242	0.000	0.100	0.056	0.002
David Cameron	0.917	0.023	0.012	0.501	0.284	0.402
Facebook	0.782	0.414	0.358	0.430	0.191	1.000
Global Warming	0.918	0.618	0.108	0.495	0.249	1.000
Google	0.950	3.048	0.047	0.034	0.010	1.000
Hillary Clinton	0.945	1.003	0.172	NA	NA	NA
Iran	0.956	5.508	0.400	0.076	0.000	0.044
Israel	0.999	2.741	0.724	NA	NA	NA
North Korea	0.847	3.066	0.528	0.174	0.385	1.000
Obama	0.940	0.350	0.943	0.808	0.734	0.859
Romney	0.919	-0.312	0.470	0.707	0.595	0.874

Table 6.14: ASSN Volatility Results.

The first observation from our results is that the different criteria to study the volatility of the series are related to each other. For instance, all the series in which a negative kurtosis was observed, presented high p-values for the McLeod-Li test, and also non-significant α_1 and β_1 coefficients. Thus, a negative kurtosis is a strong indicator that a time series is not volatile. On the other hand, we can see that series with at least one of the two GARCH(1,1) coefficients α_1 and β_1 being significant (low p-values) present both a low value for **mean.mc.test**, and a positive kurtosis.

[2] `http://cran.r-project.org/web/packages/fGarch/index.html`

Topic	ret.ttest	ret.kurt	mean.mc.test	α_0.pval	α_1.pval	β_1.pval
Abortion	0.981	4.725	0.009	0.000	0.105	0.790
David Cameron	0.959	2.370	0.227	0.536	0.436	1.000
Facebook	0.511	20.056	0.992	0.018	0.000	1.000
Global Warming	0.990	6.418	0.484	0.238	0.008	0.128
Google	0.827	1.649	0.104	0.015	0.034	0.307
Hillary Clinton	0.945	1.345	0.025	0.099	0.066	1.000
Iran	0.897	0.149	0.120	0.287	0.177	1.000
Israel	0.959	1.128	0.026	0.058	0.031	0.000
North Korea	0.810	1.076	0.561	0.334	0.322	0.359
Obama	0.972	3.615	0.010	0.001	0.019	0.732
Romney	0.980	1.058	0.723	0.219	0.113	0.995

Table 6.15: S140POS Volatility Results.

Topic	ret.ttest	ret.kurt	mean.mc.test	α_0.pval	α_1.pval	β_1.pval
Abortion	0.965	-0.588	0.107	0.372	0.261	1.000
David Cameron	0.968	0.934	0.211	0.461	0.315	1.000
Facebook	0.854	0.689	0.028	0.005	0.151	0.585
Global Warming	0.740	-0.435	0.372	0.543	0.619	1.000
Google	0.943	7.534	0.066	0.001	0.001	0.169
Hillary Clinton	0.947	0.612	0.680	0.469	0.233	1.000
Iran	0.887	0.181	0.005	0.486	0.431	0.790
Israel	0.955	0.539	0.421	0.152	0.236	0.000
North Korea	0.819	0.726	0.756	0.267	0.224	0.357
Obama	0.999	18.125	0.002	0.092	0.018	1.000
Romney	0.733	-0.337	0.199	0.175	0.143	0.423

Table 6.16: S140NEG Volatility Results.

In this way, the significance of these coefficients are also strong indicators to support the volatility of the time series.

Regarding the mean of the log returns, as it is shown in the **ret.ttest** values, we failed to reject the null hypothesis in all the opinion time series. That means, that the log returns tend to have a zero mean. Nevertheless, for the topic Facebook, in both positive series ASSP and S140POS we observed more evidence against the null hypothesis than in the remaining topics. This could be due to the significant increase observed in ASSP and S140POS time series for Facebook at the end of the period (Figure 6.11).

In contrast to the the U.S. 2008 elections analysis, in which all the transformed opinion time series satisfied the desired properties of a GARCH model, in this case only a number of the series met the conditions of a GARCH or ARCH stochastic process. Furthermore, there were cases where it was not possible to fit the GARCH coefficients to the data and the `garchFit` function returned missing values (NA) for them. This occurred in the topic Facebook for ASSP time series, and in the topics

Hillary Clinton, and North Korea for ASSN time series. The opinion time series which fit relatively well to a volatile time series were the following:

- ASSP: Abortion, Global, Warming, Google, and Israel.
- ASSN: Abortion, Google, Iran.
- S140POS: Facebook, Global Warming, Google, Israel, Obama.
- S140NEG: Google, Israel, Obama.

Topics such as Google and Israel turned out to be volatile in almost all the opinion time series. Regarding the U.S. 2012 elections, only the candidate Obama satisfied the conditions of a GARCH model for the time series S140POS and S140NEG.

Our experimental results showed that several opinion time series created from Twitter data present an important volatility factor which can be properly modeled by GARCH stochastic processes. However, although we found series which met the GARCH conditions, they were found in all the different types of topics: companies, countries, politicians, and long-standing topics. Likewise, series that do not meet the GARCH conditions were also found within all the types of topics. In this manner, our results do not allow us to state that certain types of topics are more volatile than others.

6.2.5 Discussions of the Analysis

In this section we have analyzed Twitter opinion time series for a number of topics. Although the series were calculated from different sentiment analysis methods and resources, the analysis was focused on methods Sentiment140 and SentiStrength. The aspects of the series which were analyzed included properties such as stationarity, trends, predictability, and volatility. An important difference with the former study about the 2008 U.S. elections is that in this case, both the number of topics and the volume of tweets considered were larger.

The first issue we realized from our approach was the difficulty of obtaining representative samples of tweets for the different topics using the Twitter API. Despite this issue, our series exhibited in general an important temporal dependence structure showing significant autocorrelations. This fact supported our approach of creating time series from Twitter sentiment data, and also gave evidence that Twitter opinion series are far from being random noise.

Most of the temporal properties of the opinion time series resulted to be very sensitive to the opinion dimension considered. For instance, it was possible to observe an opinion time series of a topic showing clear temporal patterns, without necessarily showing the same characteristics when a different dimension or sentiment analysis approach was considered.

Regarding the volatility results for the U.S. 2012 elections, we observed that the

series for candidates Obama and Romney fitted less well to GARCH models than the series Obama and McCain from the former elections. This could have occurred due to an important increase in the level of noise in Twitter data. As Twitter popularity has grown exponentially since its launch in 2006, the level of spam, and fake accounts, among other malicius activities has also increased dramatically in recent years [3]. This situation could have possibly affected the quality of the public opinion information reflected in the data.

All the experiments described in this Section were conducted with the aim of supporting our second subordinate research hypothesis proposed in Section 1.2. This hypothesis proposes that the evolution of social media opinions can be determined by time series models. In this context, we can say that although it is not possible to produce accurate forecasts of social media opinions using time series models, there are several properties such as seasonality, stationarity, and volatility which can be properly determined by means of time series models.

[3] `https://dev.twitter.com/docs/security/best-practices`

Chapter 7

Conclusions

This work focuses on the study and analysis of both the static and dynamic properties of Twitter opinions. As for the static part we proposed a method for the sentiment classification of tweets, whereas in the dynamic part we studied the temporal properties of opinion time series.

In the former analysis, we trained supervised classifiers for both subjectivity and polarity classification tasks. Different sentiment analysis resources and methods were used to extract sentiment features focused on three different scopes of the opinions: polarity, strength, and emotions. Our classification results showed that our tandem approach outperformed the classification performance of any isolated method, especially when features from the different scopes were combined in a non-linear fashion using RBF SVMs.

Considering that the proposed feature representation does not depend directly on the vocabulary size of the collection, it provides a considerable dimensionality reduction in comparison to word-based representations such as unigrams or n-grams. Likewise, our approach also avoids the sparsity problem presented by word-based feature representations for Twitter sentiment classification discussed in [SHA12]. Due to this, our methodology allows the efficient use of learning algorithms which do not work properly with high-dimensional data such as decision trees.

Another important remark is that opinions are multidimensional objects. In this way, when we classify tweets into polarity classes, we are essentially projecting these multiple dimensions into one single categorical dimension. Furthermore, it is not clear how to project tweets having mixed positive and negative expressions to a single polarity class. Therefore, we have to be aware that the sentiment classification of tweets may lead to the loss of valuable sentiment information.

We believe that the classification performance could be enhanced if other twitter-focused features are included, such as the presence of links or hashtags within the content. Moreover, our approach could be expanded by including other sentiment resources and methods which were not considered at this time. For instance we

could create semantic-level features from concept-based resources such as *SenticNet*. Additionally, it would be valuable to evaluate our approach on the *SemEval* task datasets in order to compare our results with other systems that participated in the task.

Regarding the second part of this thesis, we proposed a new methodology to assess how opinions evolve over time in Twitter. We showed, through experimental results, that volatility is a key aspect of opinion time series and hence, that it is very difficult to perform long term forecasts from these series.

As it is shown in Section 2.3, a significant amount of work has been done regarding prediction based on opinion mining in social media. According to this and to our experimental results for the different opinion time series, we state the following question: Is social media an accurate proxy to study how public opinion evolves over time? If the answer is yes, how could it be checked beforehand to prove that forecasting is feasible and thus, that positive results are not a product of chance?

Certainly, we cannot say a priori if the event we want to analyze will be predictable or not. Indeed, our experimental results did not give enough evidence to state that certain topics are more predictable than others. The data itself and their statistical properties can only answer that question. The quality of the series could probably be improved substantially by reducing the level of noise in the data. Messages that are not relevant to the topics or that do not express an opinion, should be filtered out. We also believe that as more accurate NLP methods for assessing the sentiment in social media messages are developed, opinion time series created with those methods will reflect in a better manner the opinion dynamics of the population.

Besides this, the social value of our aggregated sentiment indicators calculated from twitter data is still unclear. We find it necessary to develop a mechanism to assess how well the opinion time series reflect the real evolution of the public opinion. This could be done by fitting the sentiment indicators of topics tracked in Twitter to long-standing indicators drawn from traditional polls and surveys.

The main contribution of this analysis is a methodology that can be used as a framework for analyzing opinion time series. This methodology considers a number of statistical tests and allows to identify if the opinion time series are indeed predictable, or if the past conditional variance of the log returns of the process could be modeled due to the presence of volatility. Finally, we believe that forecasting the volatility of opinion time series could be used as a measurement of risk in public opinion as it is used in the finance field.

Finally, in relation to the research hypothesis stated in Section 1.2, we believe that the main findings of this thesis provide evidence to support it. Thus, we are confident to say that several public opinion properties can be determined by data-driven models applied to social media data. At the one hand, the static analysis showed that the combination of different sentiment dimensions in a supervised learning schema allows the accurate classification of the sentiment polarity of tweets. On

the other hand, the dynamic analysis showed that time series models are capable to capture in several occasions relevant properties related to the evolution of public opinion such as as seasonality and volatility.

7.1 Future Work

In this section, we propose a methodology that could be developed as future work to tackle some issues regarding Twitter sentiment analysis in a stream data model.

A relevant characteristic of Twitter is that tweets arrive at high speed following a data stream model. This means that in order to extract knowledge from this data in an online fashion, special algorithms are required that are able to work under time and space restrictions [BF10]. These kind of algorithms are called **stream data mining** methods and must be able to process data streams in a single pass, or a small number of passes, using as little memory as possible.

As was discussed within this thesis, opinions in Twitter can be expressed about different domains such as politics, products, movie reviews, and sports, among others. More specifically, opinions are expressed about particular topics, entities or subjects of a certain domain. For example, "Barack Obama" is a specific entity of the domain "politics".

Recalling from Chapter 2, the words and expressions that define the sentiment orientation of a text passage are referred to in the literature as *opinion words*. While non-supervised sentiment analysis methods rely on lexicons made of opinion words to estimate the polarity of a passage, supervised methods use opinion words as features for machine learning classifiers. Opinion words present two major issues for the sentiment classification of tweets in a stream data model.

First, many opinion words are domain-dependent [Rea05]. That means that words or expressions that are considered as positive or negative for a certain domain will not necessarily have the same relevance or orientation in a different context. Thus, a sentiment classifier that was trained on data of a particular domain, may not necessarily have the same classification performance for other topics or domains.

Secondly, as discussed in [BF10] and [BHPG11], on several occasions, the opinion words associated with a topic can change over time. For instance, new words or expressions could appear and change the polarity pattern of the topic. Hence, a sentiment classifier whose features include opinion words could be affected by this change and its accuracy would decrease over time.

In addition to the issues presented above, we believe that the major limitation of the approach proposed in this thesis is that topics to be analyzed have to be defined a priori. It would be much more useful to identify the topics and domains in an unsupervised manner from the data stream.

We propose to tackle the issues described above developing an adaptive domain-focused sentiment analysis framework. The framework will allow the sentiment classification of tweets from different domains and also will have the capability of adapting itself to changing sentiment patterns. The idea here is to design and develop a methodology to track and extract continuously sentiment information regarding different domains from Twitter streams considering that the sentiment pattern associated to each domain evolves over time.

In order to develop the framework, we propose a methodology composed by the following steps to develop the framework. is proposed.

Firstly, the retrieval of the tweets will be performed from the Twitter streaming API rather than from the Twitter Search API.

Secondly, to identify all relevant domains from the stream we will rely on non-supervised methods like text clustering algorithms [WFH11] and topic models such as the Latent Dirichlet Allocation (LDA) [BNJ03]. It is important to remark, that most topic-detection methods are not suitable for stream data models. In this work we will work with online topic detection or clustering methods. The problem of data stream clustering has been addressed by Guha et al. [GMM+03] among other researchers. An online adaption of the LDA topic model method was proposed in [ABD08]. Twitter Hashtags could be used to identify topics or as ground truth for evaluation purposes.

Once the domains of the stream were identified in the previous step, the idea is to have a sentiment classifier for each domain detected. The hypothesis is that the opinion words that induce the polarity of a tweet are particular to the domain. Thus, in order to obtain a better performance in the overall sentiment classification, it would be better to count with several classifiers. A static classification approach such as the proposed in this thesis would not work properly if the sentiment pattern of the domain is non-stationary. Concept drift refers to the phenomenon in which the statistical properties of the target variable change over time. This situation is also referred to as "concept drift". When data arrives in a stream model and concept drift is expected, learning needs to be adaptive [ZBHP11] if we expect to make predictions in real time.

Incremental classifiers such as the multinomial naive Bayes classifier, the Hoeffding Tree and stochastic gradient descent are more suited to evolving contexts. Possible approaches discussed in [BF10] to evaluate data streams in real time are the **Holdout** and the **Interleaved Test-Then-Train or Prequential** evaluation methods. Considering that opinion streams tend to be unbalanced [BF10], we will rely on the Kappa statistic based on a sliding window as evaluation criterion. Several stream data mining algorithms are implemented in the **Massive Online Analysis** (MOA)[1] framework.

Considering that supervised learning methods rely on annotated corpora for train-

[1] `http://moa.cs.waikato.ac.nz/`

ing, we could use **Amazon Mechanical Turk** (AMT)[2], which is a crowdsourcing mechanism for low cost labeling of massive datasets. Twitter conventions such as hashtags and emoticons could also be used as semi-supervised information.

Finally, change detection methods such as the **Adaptive Sliding Window** (ADWIN) [BF10] to detect when the sentiment pattern of a certain domain has changed should also be studied. We could use this approach in order to update the classifiers only when a change was found and hence avoid the continuous update of the sentiment classifiers when new data arrives.

[2] `https://www.mturk.com`

Bibliography

[ABD08] Loulwah AlSumait, Daniel Barbará, and Carlotta Domeniconi. On-line lda: Adaptive topic models for mining text streams with applications to topic detection and tracking. In *Proceedings of the 2008 Eighth IEEE International Conference on Data Mining*, pages 3–12, Washington, DC, USA, 2008. IEEE Computer Society.

[ABDF10] Cuneyt Gurcan Akcora, Murat Ali Bayir, Murat Demirbas, and Hakan Ferhatosmanoglu. Identifying breakpoints in public opinion. In *Proceedings of the First Workshop on Social Media Analytics*, SOMA '10, pages 62–66, New York, NY, USA, 2010. ACM.

[AH10] Sitaram Asur and Bernardo A. Huberman. Predicting the Future with Social Media. In *Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence*, March 2010.

[All02] J. Allan. Topic detection and tracking: Event-based information organization. *Kluwer Academic Publishers*, 2002.

[BES10] Stefano Baccianella, Andrea Esuli, and Fabrizio Sebastiani. Sentiwordnet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta, may 2010. European Language Resources Association (ELRA).

[BF10] Albert Bifet and Eibe Frank. Sentiment knowledge discovery in Twitter streaming data. In *Proceedings of the 13th international conference on Discovery science*, DS'10, pages 1–15, Berlin, Heidelberg, 2010. Springer-Verlag.

[BHPG11] Albert Bifet, Geoffrey Holmes, Bernhard Pfahringer, and Ricard Gavaldà. Detecting sentiment change in Twitter streaming data. *Journal of Machine Learning Research - Proceedings Track*, 17:5–11, 2011.

[BJ94] George Edward Pelham Box and Gwilym M. Jenkins. *Time Series Analysis: Forecasting and Control*. Prentice Hall PTR, Upper Saddle River, NJ, USA, 3rd edition, 1994.

[BL] Margaret M. Bradley and Peter J. Lang. Affective Norms for English Words (ANEW) Instruction Manual and Affective Ratings. *Technical Report C-1, The Center for Research in Psychophysiology.*

[BMZ11] Johan Bollen, Huina Mao, and Xiao-Jun Zeng. Twitter mood predicts the stock market. *J. Comput. Science*, 2(1):1–8, 2011.

[BNJ03] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent Dirichlet allocation. *J. Mach. Learn. Res.*, 3:993–1022, March 2003.

[Bol86] Tim Bollerslev. Generalized autoregressive conditional heteroskedasticity. *Journal of Econometrics*, 31(3):307–327, April 1986.

[BYR11] Ricardo A. Baeza-Yates and Luz Rello. How bad do you spell?: The lexical quality of social media. In *The Future of the Social Web*, 2011.

[BYRN99] Ricardo Baeza-Yates and Berthier Ribeiro-Neto. *Modern Information Retrieval*. Vol. 463. New York: ACM press, 1999.

[BYT+04] Steven Bethard, Hong Yu, Ashley Thornton, Vasileios Hatzivassiloglou, and Dan Jurafsky. Automatic extraction of opinion propositions and their holders. In James G. Shanahan, Janyce Wiebe, and Yan Qu, editors, *Proceedings of the AAAI Spring Symposium on Exploring Attitude and Affect in Text: Theories and Applications*, Stanford, US, 2004.

[CC09] Jonathan D. Cryer and Kung-Sik Chan. *Time Series Analysis: With Applications in R (Springer Texts in Statistics)*. Springer, 2nd edition, June 2009.

[CNP06] Giuseppe Carenini, Raymond T. Ng, and Adam Pauls. Multi-document summarization of evaluative text. In *Proceedings of the 11st Conference of the European Chapter of the Association for Computational Linguistics*, EACL '06.

[CSHH10] Erik Cambria, Robert Speer, Catherine Havasi, and Amir Hussain. Senticnet: A publicly available semantic resource for opinion mining. *Artificial Intelligence*, pages 14–18, 2010.

[CSSd09] Paula Carvalho, Luís Sarmento, Mário J. Silva, and Eugénio de Oliveira. Clues for detecting irony in user-generated contents: oh...!! it's "so easy" ;-). In *Proceeding of the 1st international CIKM workshop on Topic-sentiment analysis for mass opinion*, TSA '09, pages 53–56, New York, NY, USA, 2009. ACM.

[DCSJS08] Munmun De Choudhury, Hari Sundaram, Ajita John, and Dorée Duncan Seligmann. Can blog communication dynamics be correlated with stock market activity? In *Hypertext*, pages 55–60. ACM, 2008.

[DD10] Peter Dodds and Christopher Danforth. Measuring the happiness of large-scale written expression: Songs, blogs, and presidents. *Journal of Happiness Studies*, 11(4):441–456, August 2010.

[DKEB11] Dipankar Das, Anup Kumar Kolya, Asif Ekbal, and Sivaji Bandyopadhyay. Temporal analysis of sentiment events: a visual realization and tracking. In *Proceedings of the 12th international conference on Computational linguistics and intelligent text processing - Volume Part I*, CICLing'11, pages 417–428, Berlin, Heidelberg, 2011. Springer-Verlag.

[Eng82] Robert F. Engle. Autoregressive conditional heteroscedasticity with estimates of the variance of United Kingdom inflation. *Econometrica*, 50(4):pp. 987–1007, 1982.

[ES06] Andrea Esuli and Fabrizio Sebastiani. Sentiwordnet: A publicly available lexical resource for opinion mining. In *Proceedings of the 5th Conference on Language Resources and Evaluation*, LREC'06, pages 417–422, 2006.

[Fel98] C. Fellbaum. *WordNet: An Electronical Lexical Database*. The MIT Press, Cambridge, MA, 1998.

[GA11] Daniel Gayo-Avello. Don't turn social media into another 'literary digest' poll. *Commun. ACM*, 54(10):121–128, 2011.

[GBH09] Alec Go, Richa Bhayani, and Lei Huang. Twitter sentiment classification using distant supervision. CS224N Project Report, Stanford, 1-12, 2009.

[GCHP11] Marco Grassi, Erik Cambria, Amir Hussain, and Francesco Piazza. Sentic web: A new paradigm for managing social media affective information. *Cognitive Computation*, 3(3):480–489, 2011.

[GMM+03] Sudipto Guha, Adam Meyerson, Nina Mishra, Rajeev Motwani, and Liadan O'Callaghan. Clustering data streams: Theory and practice. *IEEE Trans. on Knowl. and Data Eng.*, 15(3):515–528, March 2003.

[Hal99] M. Hall. *Correlation-based Feature Selection for Machine Learning*. PhD thesis, University of Waikato, 1999.

[HK08] Rob J. Hyndman and Yeasmin Khandakar. Automatic time series forecasting: The forecast package for r. *Journal of Statistical Software*, 27(i03), 2008.

[HK10] Daniel Hopkins and Gary King. A method of automated nonparametric content analysis for social science. *American Journal of Political Science*, 54(1):229–247, 01/2010 2010.

[HL04] Minqing Hu and Bing Liu. Mining and summarizing customer reviews.

In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '04, pages 168–177, New York, NY, USA, 2004. ACM.

[HM97] Vasileios Hatzivassiloglou and Kathleen R. McKeown. Predicting the semantic orientation of adjectives. In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics and Eighth Conference of the European Chapter of the Association for Computational Linguistics*, ACL '98, pages 174–181, Stroudsburg, PA, USA, 1997. Association for Computational Linguistics.

[HW00] Vasileios Hatzivassiloglou and Janyce Wiebe. Effects of adjective orientation and gradability on sentence subjectivity. In *Proceedings of the 18th conference on Computational linguistics*, COLING'2000, pages 299–305. Association for Computational Linguistics.

[JJS12] Andreas Jungherr, Pascal Jürgens, and Harald Schoen. 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". *Soc. Sci. Comput. Rev.*, 30(2):229–234, May 2012.

[JSFT07] Akshay Java, Xiaodan Song, Tim Finin, and Belle Tseng. Why We Twitter: Understanding Microblogging Usage and Communities. *Proceedings of the Joint 9th WEBKDD and 1st SNA-KDD Workshop 2007*, August 2007.

[JYZ+11] Long Jiang, Mo Yu, Ming Zhou, Xiaohua Liu, and Tiejun Zhao. Target-dependent Twitter sentiment classification. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, volume 1, pages 151–160, 2011.

[JZSC09] Bernard J. Jansen, Mimi Zhang, Kate Sobel, and Abdur Chowdury. Twitter power: Tweets as electronic word of mouth. *Journal of the American Society for Information Science and Technology*, 60(11):2169–2188, 2009.

[KH04] Soo-Min Kim and Eduard Hovy. Determining the sentiment of opinions. In *Proceedings of the 20th international conference on Computational Linguistics*, COLING '04, Stroudsburg, PA, USA, 2004. Association for Computational Linguistics.

[KWM11] E. Kouloumpis, T. Wilson, and J. Moore. Twitter Sentiment Analysis: The Good the Bad and the OMG! In *Fifth International AAAI Conference on Weblogs and Social Media*, 2011.

[Lee11] Kalev Leetaru. Culturomics 2.0: Forecasting large-scale human behavior

using global news media tone in time and space. *First Monday*, 15(9), September 2011.

[LHAY07] Yang Liu, Xiangji Huang, Aijun An, and Xiaohui Yu. Arsa: a sentiment-aware model for predicting sales performance using blogs. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 607–614. ACM, 2007.

[Liu09] Bing Liu. *Web Data Mining: Exploring Hyperlinks, Contents, and Usage Data (Data-Centric Systems and Applications)*. Springer, 1st ed. 2007. corr. 2nd printing edition, January 2009.

[Liu10] Bing Liu. Sentiment analysis and subjectivity. In Nitin Indurkhya and Fred J. Damerau, editors, *Handbook of Natural Language Processing, Second Edition*. CRC Press, Taylor and Francis Group, Boca Raton, FL, 2010. ISBN 978-1420085921.

[LLG12] Kun-Lin Liu, Wu-Jun Li, and Minyi Guo. Emoticon smoothed language models for Twitter sentiment analysis. In *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence*, AAAI' 12.

[LP11] A Logunov and V Panchenko. Characteristics and predictability of Twitter sentiment series. In *19th International Congress on Modeling and Simulation–Sustaining Our Future: Understanding and Living with Uncertainty, MODSIM2011, Perth, WA*, 2011.

[Man63] Benoit Mandelbrot. The Variation of Certain Speculative Prices. *The Journal of Business*, 36(4):394–419, 1963.

[MBF+90] George A. Miller, Richard Beckwith, Christiane Fellbaum, Derek Gross, and Katherine Miller. Wordnet: An on-line lexical database. *International Journal of Lexicography*, 3:235–244, 1990.

[MBSJ09] Mike Mintz, Steven Bills, Rion Snow, and Dan Jurafsky. Distant supervision for relation extraction without labeled data. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2 - Volume 2*, ACL '09, pages 1003–1011, Stroudsburg, PA, USA, 2009. Association for Computational Linguistics.

[Md06] Gilad Mishne and Maarten de Rijke. Moodviews: Tools for blog mood analysis. In *AAAI Symposium on Computational Approaches to Analysing Weblogs (AAAI-CAAW)*, pages 153–154, 2006.

[MdR06] G.A. Mishne and M. de Rijke. Capturing global mood levels using blog posts. In *AAAI 2006 Spring Symposium on Computational Approaches to Analysing Weblogs*. AAAI Press, 2006.

[MG06] Gilad Mishne and Natalie Glance. Predicting movie sales from blogger sentiment. In *AAAI Symposium on Computational Approaches to Analysing Weblogs (AAAI-CAAW)*, pages 155–158, 2006.

[MRS08] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. *Introduction to Information Retrieval*. Cambridge University Press, New York, NY, USA, 2008.

[MT12] Saif M. Mohammad and Peter D. Turney. Crowdsourcing a word–emotion association lexicon. *Computational Intelligence*, 2012. Wiley Blackwell Publishing Ltd.

[Nie11] Finn Å. Nielsen. A new ANEW: Evaluation of a word list for sentiment analysis in microblogs. In *Proceedings of the ESWC2011 Workshop on 'Making Sense of Microposts': Big things come in small packages*, pages 93–98, May 2011.

[OBRS10] Brendan O'Connor, Ramnath Balasubramanyan, Bryan R. Routledge, and Noah A. Smith. From Tweets to Polls: Linking Text Sentiment to Public Opinion Time Series. In *Proceedings of the International AAAI Conference on Weblogs and Social Media*, pages 122–129, 2010.

[Ols12] Daniel J Olsher. Full spectrum opinion mining: Integrating domain, syntactic and lexical knowledge. In *Data Mining Workshops (ICDMW), 2012 IEEE 12th International Conference on*, pages 693–700. IEEE, 2012.

[Par01] W. Gerrod Parrot. *Emotions in social psychology*. Essential readings. Psychology Pr., 2001.

[PL05] Bo Pang and Lillian Lee. Seeing stars: exploiting class relationships for sentiment categorization with respect to rating scales. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, ACL '05, pages 115–124, Stroudsburg, PA, USA, 2005. Association for Computational Linguistics.

[PL08] Bo Pang and Lillian Lee. Opinion mining and sentiment analysis. *Found. Trends Inf. Retr.*, 2:1–135, January 2008.

[Plu01] Robert Plutchik. The nature of emotions human emotions have deep evolutionary roots, a fact that may explain their complexity and provide tools for clinical practice. *American Scientist*, 89(4):344–350, 2001.

[PLV02] Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. Thumbs up? Sentiment classification using machine learning techniques. In *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 79–86, 2002.

[POL10] Saša Petrović, Miles Osborne, and Victor Lavrenko. The edinburgh Twitter corpus. In *Proceedings of the NAACL HLT 2010 Workshop on Computational Linguistics in a World of Social Media*, WSA '10, pages 25–26, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.

[PP10] A. Pak and P. Paroubek. Twitter as a corpus for sentiment analysis and opinion mining. *Proceedings of the Seventh conference on International Language Resources and Evaluation*, LREC'10, Valletta, Malta, European Language Resources Association ELRA.

[Rea05] Jonathon Read. Using emoticons to reduce dependency in machine learning techniques for sentiment classification. In *Proceedings of the ACL Student Research Workshop*, ACLstudent '05, pages 43–48, Stroudsburg, PA, USA, 2005. Association for Computational Linguistics.

[SHA12] H. Saif, Y. He, and H. Alani. Alleviating data sparsity for Twitter sentiment analysis. In *Proceedings of the 2nd Workshop on Making Sense of Microposts*, pages 2–9, 2012.

[SP96] Howard Schuman and Stanley Presser. *Questions and answers in attitude surveys: Experiments on question form, wording, and context*. Sage Publications, Inc., 1996.

[TBP12] Mike Thelwall, Kevan Buckley, and Georgios Paltoglou. Sentiment strength detection for the social web. In *Journal of the American Society for Information Science and Technology*, 63(1):163–173, 2012.

[TSSW10] Andranik Tumasjan, Timm O. Sprenger, Philipp G. Sandner, and Isabell M. Welpe. Predicting elections with Twitter: What 140 characters reveal about political sentiment. In *Proceedings of the Fourth International AAAI Conference on Weblogs and Social Media*, pages 178–185. The AAAI Press, 2010.

[Tur02] Peter D. Turney. Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, pages 417–424, Stroudsburg, PA, USA, 2002. Association for Computational Linguistics.

[WBO99] Janyce Wiebe, Rebecca F. Bruce, and Thomas P. O'Hara. Development and use of a gold-standard data set for subjectivity classifications. In *Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics*, pages 246–253, 1999. Association for Computational Linguistics.

[WFH11] Ian H. Witten, Eibe Frank, and Mark A. Hall. *Data Mining: Practical*

Machine Learning Tools and Techniques. Morgan Kaufmann, Burlington, MA, 3 edition, 2011.

[Wik11a] Wikipedia. Opinion poll — Wikipedia, the free encyclopedia, 2011. [Online; accessed 22-July-2011].

[Wik11b] Wikipedia. Public opinion — Wikipedia, the free encyclopedia, 2011. [Online; accessed 22-July-2011].

[WKN+13] Theresa Wilson, Zornitsa Kozareva, Preslav Nakov, Alan Ritter, Sara Rosenthal, and Veselin Stoyonov. Semeval-2013 task 2: Sentiment analysis in Twitter. In *Proceedings of the 7th International Workshop on Semantic Evaluation*. Association for Computation Linguistics, 2013.

[WR05] Janyce Wiebe and Ellen Riloff. Creating Subjective and Objective Sentence Classifiers from Unannotated Texts. In *Proceeding of CICLing-05, International Conference on Intelligent Text Processing and Computational Linguistics*. Volume 3406 of *Lecture Notes in Computer Science*, pages 475–486, Mexico City, MX, 2005. Springer-Verlag.

[WWH05] Theresa Wilson, Janyce Wiebe, and Paul Hoffmann. Recognizing Contextual Polarity in Phrase-Level Sentiment Analysis. In *Proceedings of Human Language Technologies Conference/Conference on Empirical Methods in Natural Language Processing (HLT/EMNLP 2005)*, pages 347–354, Vancouver, CA, 2005.

[YH03] Hong Yu and Vasileios Hatzivassiloglou. Towards answering opinion questions: separating facts from opinions and identifying the polarity of opinion sentences. In *Proceedings of the 2003 conference on Empirical methods in natural language processing*, EMNLP '03, pages 129–136, Stroudsburg, PA, USA, 2003. Association for Computational Linguistics.

[YK12] S. Yu and S. Kak. A Survey of Prediction Using Social Media. *ArXiv e-prints*, March 2012.

[ZBHP11] Indre Zliobaite, Albert Bifet, Geoff Holmes, and Bernhard Pfahringer. Moa concept drift active learning strategies for streaming data. *Journal of Machine Learning Research - Proceedings Track*, 17:48–55, 2011.

[ZGD+11] Lei Zhang, Riddhiman Ghosh, Mohamed Dekhil, Meichun Hsu, and Bing Liu. Combining lexicon-based and learning-based methods for Twitter sentiment analysis. Technical report, Hewlett-Packard Development Company, L.P., 2011.

[ZJ11] Xin Zhao and Jing Jiang. An empirical comparison of topics in Twitter and traditional media. Technical report, Singapore Management University School of Information Systems Technical Paper Series, January

2011.

[ZL11] Lei Zhang and Bing Liu. Identifying noun product features that imply opinions. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers - Volume 2* , pages 575–580, 2011.

[ZNSS11] Cäcilia Zirn, Mathias Niepert, Heiner Stuckenschmidt, and Michael Strube. Fine-grained sentiment analysis with structural features. In *Proceedings of the 5th International Joint Conference on Natural Language Processing*, pages 336–344, 2011.

Appendix A

Visualization of other Opinion Time Series

In the following pages we present the plots of the remaining opinion time series.

Figure A.1: AAPO Opnion Time Series.

Figure A.2: AANE Opniion Time Series.

Figure A.3: S140NEU Opinion Time Series.

Figure A.4: ASWP Opinion Time Series.

Figure A.5: ASWN Opinion Time Series.

Figure A.6: AOPW Opinion Time Series.

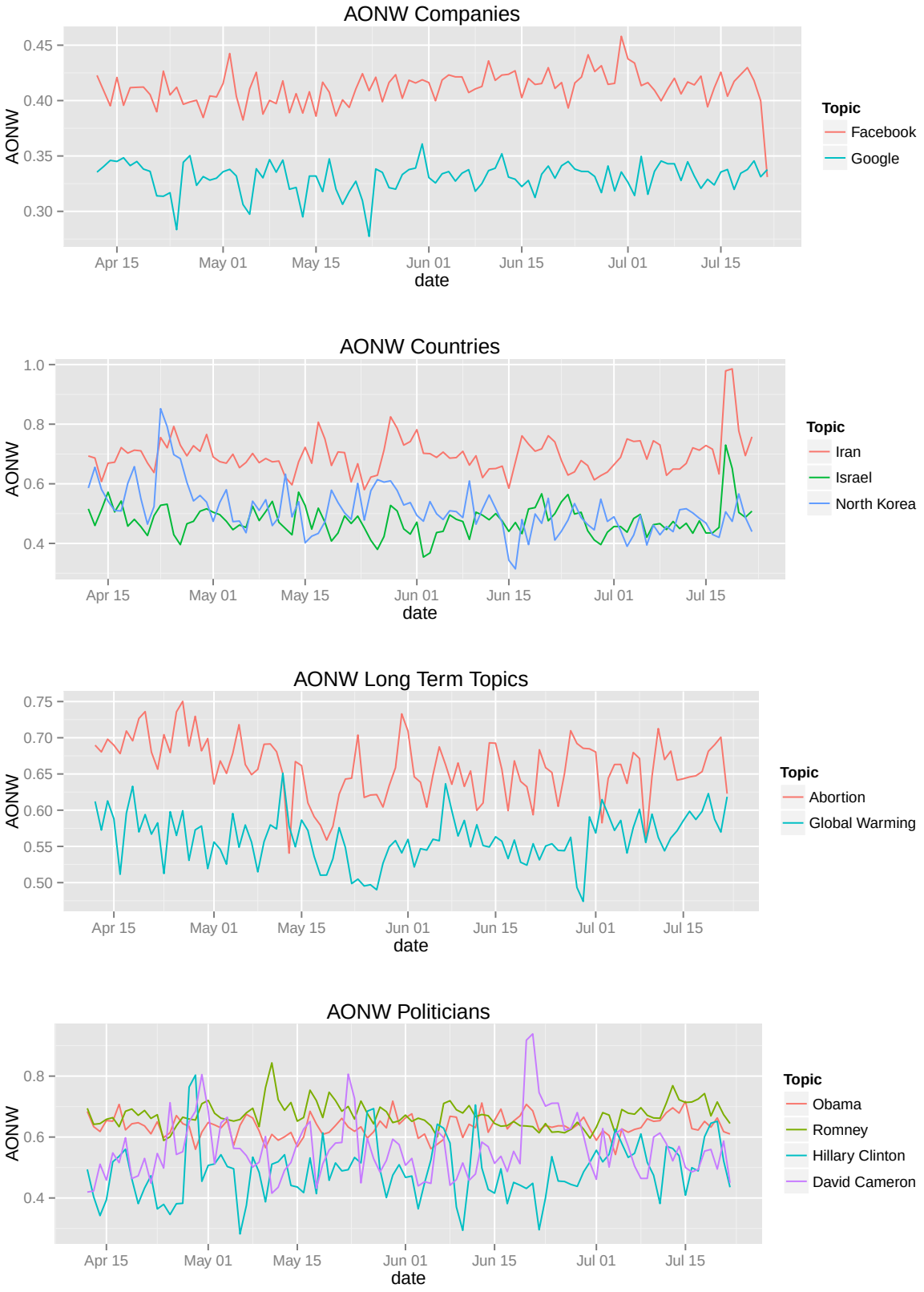


Figure A.7: AOPW Opinion Time Series.