A MULTI-LAYER DIMENSION REDUCTION ALGORITHM FOR TEXT MINING OF NEWS IN FOREX

ARMAN KHADJEH NASSIRTOUSSI

THESIS SUBMITTED IN FULFILMENT

OF THE REQUIREMENTS

FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

FACULTY OF COMPUTER SCIENCE AND INFORMATION TECHNOLOGY

UNIVERSITY OF MALAYA

KUALA LUMPUR

AUGUST 2015

UNIVERSITY MALAYA
ORIGINAL LITERARY WORK DECLARATION

Name of Candidate: **ARMAN KHADJEH NASSIRTOUSSI**

I.C/Passport No: **H95621284 (Old: U16352939)**

Registration/Matric No: **WHA090031**

Name of Degree: **DOCTOR OF PHILOSOPHY**

Title of Project Paper/Research Report/Dissertation/Thesis ("this Work"): **A Multi-Layer Dimension Reduction Algorithm for Text Mining of News in FOREX**

Field of Study: **TEXT MINING**

I do solemnly and sincerely declare that:

(1) I am the sole author/writer of this Work;
(2) This Work is original;
(3) Any use of any work in which copyright exists was done by way of fair dealing and for permitted purposes and any excerpt or extract from, or reference to or reproduction of any copyright work has been disclosed expressly and sufficiently and the title of the Work and its authorship have been acknowledged in this Work;
(4) I do not have any actual knowledge nor do I ought reasonably to know that the making of this work constitutes an infringement of any copyright work;
(5) I hereby assign all and every rights in the copyright to this Work to the University of Malaya ("UM"), who henceforth shall be owner of the copyright in this Work and that any reproduction or use in any form or by any means whatsoever is prohibited without the written consent of UM having been first had and obtained;
(6) I am fully aware that if in the course of making this Work I have infringed any copyright whether intentionally or otherwise, I may be subject to legal action or any other action as may be determined by UM.

Candidate's Signature Date

Subscribed and solemnly declared before,
Witness's Signature Date

Name:
Designation:

ABSTRACT

Information Explosion has caused the demand for customized text-mining in every imaginable area to sky-rocket. Text mining is needed in many areas, a few of which are: search engine development, spam-filtering and text-summarization. Every context requires its own customized text mining algorithms in order to achieve best results. The specific context of this research is *market prediction* for the foreign exchange market. The objective is to utilize news-headlines to predict market-movements 1 to 3 hours after news release.

The literature on recent research efforts in behavioral economics confirms that investors' aggregate behavioral reactions to information released in the news can drive prices up or down. This theoretical basis constitutes the economic foundation of this investigation.

After economic comprehension of the problem at hand; available systems in the literature which operate in a comparable context are reviewed. The major finding of this review is that context-specific text mining algorithms are lacking. The main underlying text-mining challenge that seems to deserve immediate attention is the sparse and high dimensional nature of the feature-space.

Therefore, this work produces a multi-layer dimension reduction algorithm to respond to this need.

The algorithm tackles a different root cause of the problem at each layer. The first layer is termed the *Semantic Abstraction Layer* and addresses the problem of co-reference in text mining that is contributing to sparsity. Co-reference occurs when two or more words in a text corpus refer to the same concept. This work produces a custom approach by the name of Heuristic-Hypernyms Modeling which creates a way to recognize words

with the same parent-word to be regarded as one entity. As a result, prediction accuracy increases significantly at this layer which is attributed to appropriate noise-reduction from the feature-space.

The second layer is termed *Sentiment Integration Layer*, which integrates sentiment analysis capability into the algorithm by proposing a sentiment weight by the name of SumScore that reflects investors' sentiment. This layer reduces the dimensions by eliminating those that are of zero value in terms of sentiment and thereby improves prediction accuracy.

The third layer encompasses a dynamic model creation algorithm, termed Synchronous Targeted Feature Reduction (STFR). It is suitable for the challenge at hand whereby the mining of a stream of text is concerned. It updates the models with the most recent information available and, more importantly, it ensures that the dimensions are reduced to a number that is many times smaller.

The algorithm and each of its layers are extensively evaluated using real market data and news content across multiple years and have proven to be solid and superior to any other comparable solution. On top of a well-rounded multifaceted algorithm, this work contributes a much needed research framework for this context with a test-bed of data that must make future research endeavors more convenient. The produced algorithm is scalable and its modular design allows improvement in each of its layers in future research.

ABSTRAK

Permintaan terhadap perlombongan data terutama di dalam bidang teks terus meningkat saban hari berikutan terdapat banyaknya informasi digital di ruang maya. Perlombongan teks diperlukan di dalam beberapa bidang komputer seperti enjin pencarian maklumat, tapisan data dan ringkasan maklumat. Algoritma perlombongan teks harus di sesuaikan dengan konteks dan aplikasi yang hendak dibina supaya ianya dapat mencapai hasil yang terbaik. Di dalam kajian ini, konteks yang dipilih adalah ramalan pasaran untuk pasaran pertukaran asing. Objektif kajian tertumpu kepada ramalan pergerakan pasaran di antara 1 hingga 3 jam selepas sesuatu berita di terbitkan.

Kajian literasi di dalam bidang tingkah laku ekonomi membuktikan agregasi reaksi tingkah laku para pelabur terhadap sesuatu berita dapat menjana atau menurunkan sesuatu harga pasaran. Dengan itu, kajian yang dijalankan akan bersandarkan kepada teori tersebut.

Di dalam kajian ini, beberapa bidang literasi telah dikaji. Di antaranya adalah kajian terhadap beberapa sistem sedia ada yang berada dalam konteks yang sama. Hasil kajian literasi menunjukkan terdapat banyak kekurangan di dalam perlombogan teks berdasarkan konteks yang spesifik pada sistem dan cara sedia ada. Caraban utama yang harus di ambil perhatian adalah terhadap banyaknya dimensi yang terdapat di dalam ruang ciri. Dengan itu, kajian ini juga akan menghasilkan algoritma lapisan dimensi untuk mengatasi masalah yang telah disebutkan.

Algoritma yang dihasilkan akan menyelesaikan beberapa masalah utama dimensi pelbagai pada setiap lapisan.

Lapisan pertama dikenali sebagai Lapisan Semantik Abstrak. Lapisan ini dapat menyelesaikan masalah referensi bersama yang boleh menyebabkan kekurangan atau kekosongan data. Referensi bersama terjadi apabila dua atau lebih ayat di dalam sesuatu

koleksi tulisan merujuk kepada konsep yang sama. Model Heuristik-Hypernyms akan digunakan untuk mengenal pasti ayat yang mempunyai ayat atasan yang sama sebagai satu entiti. Keputusan kajian mendapati ketepatan jangkaan meningkat pada lapisan pertama dan secara tidak langsung dapat mengurangkan gangguan pada ruang ciri.

Lapisan kedua dikenali sebagai Lapisan Integrasi Sentimen dimana pada lapisan ini, analisis keupayaan sentimen akan diintegrasikan dengan algoritma yang telah dihasilkan dengan mencadangkan satu pemberat bagi merefleksi sentimen pelabur. Pemberat ini dinamakan 'SumScore'. Pada lapisan ini, dimensi ciri dapat dikurangkan dengan membuang dimensi yang mempunyai hasil sifar terhadap sentimen sekaligus dapat meningkatkan ketepatan.

Lapisan ketiga merangkumi algoritma penciptaan model dinamik yang juga dikenali sebagai Pengurangan Ciri Secara Terperinci dan Selari (STFR). Algoritma ini sangat sesuai digunakan untuk mendepani cabaran untuk melombong teks secara terusan. Algoritma ini akan terus mengemas kini model dengan informasi terkini yang sedang tersedia. Algorima ini juga memastikan dimensi ciri dapat dikurangkan pada jumlah yang paling sedikit.

Algoritma yang diutarakan beserta model lapisan dimensi telah diuji menggunakan data sebenar pasaran dan pelbagai berita selama beberapa tahun. Hasil keputusan kajian menunjukkan algortima yang diutarakan telah terbukti berjaya mengatasi cara dan algoritma yang tersedia. Hasil kajian juga telah memudahkan kajian pada masa hadapan dengan tersedianya satu rangka kerja beserta data sebagai kayu pengukur. Algoritma yang telah diutarakan pada kajian ini amat mudah untuk diskala serta ditambah baik pada setiap lapisan dimensi.

ACKNOWLEDGMENT

I would like to thank my first supervisor, associate professor Dr. Teh Ying Wah, for his dedicated attention and unreserved support of this work. His guidance and insightful advice has enabled this work to be what it is today. His patience and encouragement has made the completion of this work possible. I am grateful to have worked with him and wish him continued success.

I must also thank my second supervisor, Dr. Saeed Aghabozorgi, whose detailed review of this work and thoughtful feedback has been tremendously beneficial to its quality. His availability to this work has been an asset and has facilitated progress on multiple challenges on this journey. I feel lucky to have worked with him.

Furthermore, I appreciate having had Professor Dr. David Chek Ling Ngo as initiator and advisor of this work. His considerate and informed suggestions have been guiding this work all the way. His attention to detail and motivation have propelled this work to new fronts.

Moreover, I would like to thank Dr. Khairil Imran Ghauth for both sparking the idea of doing a PhD as well as kindly helping at the end with translation of its abstract to Malay.

Next, I would like to thank my parents whose motivation and support have been there for me during this journey and my entire life.

Last but not least, I am indebted to the patience of my wife, Monika. Her livelihood and strength provide balance in my life and have made spending ample time on this work a breeze.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS AND ACRONYMS

AAPL	Apple Inc.
AI	Artificial Intelligence
AMZN	Amazon.com Inc.
ANN	Artificial Neural Networks
AzTeK	Arizona Text Extractor
BHP.AX	BHP Billiton Ltd.
BI	Business Intelligence
BNS	Bi-Normal Separation
BOW	Bag of Words
BSD	Berkeley Software Distribution
CF	Category Frequency
CHI	Chi-square Statistics
CNG	Common N-Gram
CRF	Conditional Random Fields
CRM	Customer Relation Management
DAX	Deutscher Aktien Index
DF	Document Frequency
DJIA	Dow Jones Industrial Average
EMH	Efficient-Market Hypothesis
FOREX	Foreign Exchange Market
GOOG	Google Inc.
HMM-LDA	Hidden Markov Model LDA
IG	Information Gain
k-NN	k-Nearest Neighbors
LDA	Latent Dirichlet Allocation
LIWC	Linguistic Inquiry, Word Count
MSFT	Microsoft Inc.
MSH	Morgan Stanley High-Tech Index
MUC-7	Message Understanding Conference-7
NASDAQ	National Association of Securities Dealers Automated Quotations
NB	Naïve Bayes
NegScore	Negativity Score
NER	Named-entity recognition
NLP	Natural Language Processing
NLTK	Natural Language Toolkit
NYSE	New York Stock Exchange
ObjScore	Objectivity Score
OLS	Ordinary Least Square
PoS	Parts of Speech
PosScore	Positivity Score
Probit	Probability unit

RSS	Really Simple Syndication
Sensex	Stock Exchange Sensitive Index
SOFNN	Self-Organizing Fuzzy Neural Network
STFR	Synchronous Targeted Feature Reduction
STLP	Synchronous Targeted Label Prediction
SVM	Support vector Machine
SVR	Support Vector Regression
TF	Term Frequency
TF-CDF	Term Frequency-Category Discrimination
TF-IDF	Term Frequency-Inverse Document Frequency

1 Introduction

1.1 Overview

In this chapter these sections are following: 1.2. Background; 1.3. Problem Statement; 1.4. Objectives; 1.5. Research Questions; 1.6. Motivation; 1.7. Significance of Study; 1.8. Scope; 1.9. Other Chapters.

1.2 Background

The welfare of modern societies of today depends heavily on their market-economies (Joseph, 1991). At the heart of any market-economy lie financial markets (Gregory & Stuart, 2004). Understanding market movements primarily facilitates one with the ability to predict future movements. Ability to predict in a financial market in a market economy is equal to being able to generate wealth by avoiding financial losses and making financial gains (Bacchetta, Mertens, & van Wincoop, 2009). However, the nature of markets is as such that they are extremely difficult to predict (Potì & Siddique, 2013). There are many types of financial markets, namely: 1- Capital markets (Stock and Bond), 2- Commodity markets, 3- Money markets, 4- Derivative markets, 5- Future markets, 6- Insurance markets and 7- Foreign exchange markets.

This work observes the Foreign Exchange Market which facilitates the trading of currencies. Common abbreviations for this market are: FOREX or FX. FX markets are evolving rapidly in response to new electronic trading technologies: Transparency has risen, trading costs have tumbled, and transaction speed has accelerated as new players have entered the market and existing players have modified their behaviour. These changes have profound effects on exchange rate dynamics (King, Osler, & Rime, 2013).

1.2.1 Definition of terms in this thesis title

Before going deeper into the related background topics, a set of 5 definitions are presented in the below to provide adequate initial clarification on the terms used in the title of this thesis.

Definition 1.1: Text Mining of News

Text mining is a subfield of data mining where the focus is on identifying patterns in and learning from textual contents. What differentiates text mining from data mining is primarily investigating a set of techniques through which unstructured text is processed and transformed into a machine usable format that is structured. This format is usually in form of a feature matrix with text chunks as rows and defining features which are usually words as columns.

There are different sources of text available that can be subject to text mining for example email contents, book contents, social media contents and news contents. Each of these textual sources, however, has special characteristics in nature that are different from the next, for example, the words that are used, the type of spelling that is used or the topics that are covered.

Furthermore, the goal of text mining also differs from context to context, for example, in case of a book what is expected from text-mining may be summarization; but in case of emails what is expected from text-mining may be spam filtering or text classification.

Therefore, it becomes imperative to conduct separate research on each of these textual sources and with regards to the specific context that they are subject to. In this work, the textual source that is subject to research is ‘news’ and the goal of text-mining is binary classification of news text in relation to an upward or downward movement of a price-index in a financial market.

Definition 1.2: FOREX Market Prediction

The prediction subject to research is a short-term prediction for movements of a market that occur within a day. This is what is termed as intraday prediction. The specific predictive time span that this research focuses on is the period of 1 to 3 hours after news release. This period is selected based on reviewing other works and its suitability to the gathered dataset subject to this work. Furthermore, the specific market that this work conducts experiments in is the FOREX market i.e. the Foreign Exchange Market. FOREX market is one of the most significant financial markets due to its size and liquidity. The index that is tracked in this market is a currency pair. A currency pair is the relative price of a currency compared to another currency (the base-currency). The currency pair subject to this research is Euro/USD which is the price of 1 Euro in USD. It is one of the most significant currency pairs due to the significance of the world regions that own these currencies and their use in global financial transactions.

Definition 1.3: A Multi-layer Dimension Reduction Algorithm

In this work text mining of news in the context of market prediction is improved in a number of ways to tackle a number of shortcomings in the state of the art. One of the main issues that is addressed is the problem of high-dimensionality, whereby there are too many features in the feature matrix that is to be analyzed. A Multi-layer Dimension Reduction Algorithm is an algorithm proposed in this work that has multiple layers and in each layer the number of features i.e. dimensions are reduced by a proposed model. This is explained in detail in the chapter on solution design.

Definition 1.4: Semantics

Another shortcoming that is observed in available works is a lack of usage of semantic analysis. This is specially significant due to a problem termed as Co-reference, whereby many words are referring to the same concept or entity; this problem is also termed as

'semantic redundancy' in this work. This work proposes a model of semantic abstraction to address this challenge.

Definition 1.5: Sentiment

One more shortcoming that is observed in available works is what is termed in this thesis as 'sentiment ignorance'; which means that the emotional sentiment of the textual content is ignored in many of the available works. This work proposes a model to involve a sentiment weight in the proposed algorithm and thereby take sentiment into consideration.

Next some of the important background topics are introduced.

1.2.2 Financial Markets' Predictability

One of the prominent works in the literature that structures the work around market predictability is the work of Fama (1965). Fama's work in the past decades has been influential enough to earn him a Nobel Prize in economics in 2013. The core of his work is formulation of Efficient-Market Hypothesis (EMH) which asserts that when financial markets are "informationally efficient" one cannot consistently achieve returns in excess of average market returns on a risk-adjusted basis, given the information available at the time the investment is made. But what is more important is the recognition of the fact that markets are rarely completely efficient and there is room for predictability depending on the context in which a market finds itself. According to Fama (1970) market efficiency can be strong, semi-strong or weak. And markets are most predictable when the efficiency is weak.

Gregory and Stuart (2004) in their book- Comparing Economic Systems in the Twenty-First Century- do exactly that and demonstrate how the nature of markets are evolving and study the transition from planned economies to market economies and how new markets are emerging in the world. On the other hand, technology is playing a

significant role in information dissemination, management of financial markets, transaction cost and generally market microstructure (Lehalle & Laruelle, 2013). These phenomena attribute markets with highly dynamic natures that need to be constantly revisited (H. Yu, Nartea, Gan, & Yao, 2013). And as a result of that market predictability and determination of efficiency levels continues to be the subject of many studies in different contexts (Bacchetta et al., 2009; Bisoi & Dash, 2014; Büyükşahin & Robe, 2014; Eleftherios Soulas, 2013; Mizrach & Otsubo, 2014; Potì & Siddique, 2013; K.-L. Wang, Fawson, Chen, & Wu, 2014; H. Yu et al., 2013) .

1.2.3 Prediction Avenues

There are two main avenues to predict price-movements in financial markets. One avenue is technical analysis with its premise on the assumption that historic market movements are bound to repeat themselves; that there are visual patterns in a market graph that can be detected. Despite its popularity the effectiveness of this approach is questionable. H. Yu et al. (2013) study the predictive ability of technical trading rules and show that under many circumstances such rules have limited predictive power. Fang, Jacobsen, and Qin (2014) also confirm that they find no evidence that several well-known technical trading strategies predict stock markets over their study period. Kuang, Schröder, and Wang (2014) study technical analysis specifically in the foreign exchange markets and go on further to say that profitability of technical analysis is illusionary.

The other market prediction avenue is called fundamental analysis. Fundamental analysis considers any information from the outside world with regards to the economy or financial situation of a company or an asset in question to estimate its value and thereby enables prediction by determining if an asset is currently undervalued or overvalued and expect the market to eventually adjust and resolve the inefficiency through absorbing and reflecting the new information. Fundamental analysis is a very

promising prediction avenue. Bekiros (2014) reconfirms that fundamentals are important determinants of FX rates. W. Yin and Li (2014) report a close link between macroeconomic fundamentals and the exchange rate dynamics. Fundamental analysis can be done in a bottom-up-approach by starting from the information about a company like its balance-sheet or management-quality; and go upwards towards a global perspective. Dorantes Dosamantes (2013) demonstrate the relevance of using accounting fundamentals to understand firm value. Fundamental analysis can also be carried out top-down; starting by observing macro-economic data like: Gross Domestic Product (GDP), inflation rates, unemployment rates, national income, housing, manufacturing, etc and then down to microeconomics of an industry or a region like information about supply and demand in a specific market etc. Q. Li and Chand (2013) show that important market fundamentals such as the levels of income, construction costs, impending marriages, user cost and land prices are the primary determinants of house prices.

However, it is not an easy task to carry out fundamental analysis because it requires access to a lot of information and analytical capability and, moreover, it all depends on the analyst's perspective and perception of the world. Kaltwasser (2010) argues that the individual's perception of fundamental variables creates uncertainty in the FOREX market. Bacchetta and van Wincoop (2013) show that relationship between exchange rates and fundamentals is driven not by the structural parameters themselves, but rather by expectations of these parameters. De Martino, O'Doherty, Ray, Bossaerts, and Camerer (2013) suggest that incorporating inferences about the intentions of others when making value judgments in a complex financial market could lead to the formation of market bubbles. ter Ellen, Verschoor, and Zwinkels (2013) make an effort identify expectation formation rules for institutional investors in the foreign exchange market. Muehlfeld, Weitzel, and van Witteloostuijn (2013) analyze investors' trading

behaviour and demonstrate that their style of coping with fundamental shocks in asset value depends on individual differences in the sensitivity and type of their behavioural systems. All of the above indicate that the sentiment of the investors towards the information they receive plays a significant role in the price-movements in financial markets.

1.2.4 Role of News in Market Prediction

Knowing the above discussed 3 facts, which are to recap: 1- markets can be predicted under certain circumstances, 2-fundamental information like financial, geopolitical or macroeconomic information can be of great assistance in helping with prediction and 3- a big part of an investor's behavioural reaction is based on psychological sentiment and not entirely rational. The next thing to look at is obviously news, because news is a channel where all of the above information that is needed for prediction is released in. So simply put, a fundamental analyst reads the news and in reaction to it predicts the market and makes investments accordingly and then profits from it if his predictions are right. The aggregate reaction of all investors causes the market to move in a certain direction. In short, investors react to the news and hence the market moves. As explained in the previous section, a big part of this reaction in addition to their rational fundamental analysis is dependent on the investors' psychological behaviour and sentiment.

One recent study shows the impact of US news, UK news and Dutch news on three Dutch banks during the financial crisis of 2007-2009 (Kleinnijenhuis, 2013). The media do not report market status only, but they actively create an impact on market dynamics based on the news they release (Robertson, Geva, & Wolff, 2006; Wisniewski & Lambe, 2013). Chatrath, Miao, Ramchander, and Villupuram (2014) investigate the impact of macro news on currency jumps and indicate that 9–15% of currency jumps can be directly linked to U.S. macroeconomic announcements. Nizer and Nievola

(2012) demonstrate that it is possible to find out whether certain news may cause a considerable impact on prices of a stock.

Therefore, analysis of news is a legitimate basis for market prediction.

1.2.5 Role of Text Mining in Market Prediction

Now that it is established in the above sections that news content is a legitimate basis for market prediction, it is time to consider how news content can be processed. Textual format of news content is one highly available form of news that can be processed by computers. Text mining is a branch of data mining that investigates the process of deriving information from text. This is done by identifying patterns and trends within a text corpus through the help of statistical pattern recognition or machine learning algorithms. Text mining process is composed of three significant steps: one, the act of structuring the text as textual information is usually unstructured; two, deriving patterns and three, evaluating the interpretations. Prominent text mining tasks include text categorization, text clustering, concept/entity extraction, production of granular taxonomies, sentiment analysis, document summarization, and entity relation modeling.

As rationalized in the previous sections, this work is premised on the sentimental reaction of investors after reading the news. This context brings at least two specific text mining tasks to the forefront. One is *sentiment analysis* to satisfy the need to identify the sentiment that is captured in the news content which is driving investors' decisions. Sentiment analysis in the literature generally deals with detecting the emotional sentiment preserved in text through specialized semantic analysis for a variety of other purposes for instance to gauge the quality of market reception for a new product and the overall customer feedback or to estimate the popularity of a product or brand among people (Ghiassi, Skinner, & Zimbra, 2013; Mostafa, 2013).

Two is *text classification* as the ultimate objective is to predict if a price is going to move up or down in the market after the investors' reactions to textual news content have been absorbed by the market. 'Up' and 'Down' compose two classes into which news text can be assigned.

Different variations of textual classification with the above line of thought is being increasingly attempted by a number of researchers specially in the stock market (Dai, Yang, & Li, 2011; Hagenau, Liebmann, & Neumann, 2013; C.-J. Huang, Liao, Yang, Chang, & Luo, 2010; Kleinnijenhuis, 2013; Nizer & Nievola, 2012; Schumaker, Zhang, Huang, & Chen, 2012; Shou-Hsiung, 2010; L.-C. Yu, Wu, Chang, & Chu, 2013).

Very few happen to experiment with unstructured news text in the FOREX market (Jin et al., 2013; Peramunetilleke & Wong, 2002). There are more attempts in FOREX market regarding the use of structured news, for example, scheduled macroeconomic announcements (Chatrath et al., 2014; Égert & Kočenda, 2013; Evans & Lyons, 2008; Hutchison & Sushko, 2013).

Moreover, sentiment analysis of the investors' perception of and reaction to new information in the news content has been ignored in most of the above works, while many research works in behavioral economics point to significance of some form of sentiment repeatedly and study it specifically in the FOREX market (Bacchetta & van Wincoop, 2013; Kaltwasser, 2010; Muehlfeld et al., 2013).

Furthermore, in order to approach text classification like any other text mining task, the first step is structuring the text. However, structuring the text is a challenging research topic of its own with different avenues to approach (Chen, Huang, Tian, & Qu, 2009; Tsai, Eberle, & Chu, 2013; Uysal & Gunal, 2014). Surprisingly the previous works of research in relation to market prediction and FOREX prediction specifically seem to have stopped at the most basic form of structuring of text (Jin et al., 2013; Schumaker et

al., 2012). As only the surface of text structuring has been scratched (Kleinnijenhuis, 2013), this work investigates this aspect as well further and goes deeper into text structuring by inclusion of some level of semantic analysis.

Therefore, the resulted text mining approach introduced in this work is a composite of text classification through semantic and sentiment analysis which are reflected in its proposed feature selection and weighting aspects respectively.

Additionally, text mining is further improved in this work by devising and integrating a dimensionality reduction technique in the above approach. Dimensionality reduction itself is an established research area in text mining and information retrieval (Berka & Vajteršic, 2013; Shi, He, Liu, Zhang, & Song, 2011).

The proposed approach and each of the above aspects are discussed later in individual sections accordingly in chapter 4.

1.3 Problem Statement

It is a great challenge of our time to predict financial markets (Fang et al., 2014). Behavioral economics suggests that it may be possible to utilize fundamental information that is found in the news content in order to make predictions (Chatrath et al., 2014; Égert & Kočenda, 2013; Hutchison & Sushko, 2013; Jin et al., 2013; Kleinnijenhuis, 2013). The form of news content that is highly malleable by machines and is widely available is textual (Hagenau et al., 2013). However, the state of text-mining in this specific context is immature (Kleinnijenhuis, 2013). Partly because this is an emerging field as such great amount of timely textual content has just in the recent decade become available via the emergence and domination of the Internet; And partly because most researchers have had to deal with either the highly complex nature of the financial markets (Urquhart & Hudson, 2013; W. Yin & Li, 2014) or the challenges of text-mining in general (Balahur, Mihalcea, & Montoyo, 2014; Cambria, Schuller,

Yunqing, & Havasi, 2013; Haddi, Liu, & Shi, 2013; Uysal & Gunal, 2014). The prerequisite to enter the research arena of this context is at least to observe two disciplines: 1- computer science and 2- economics. This further marginalizes this research topic as a challenge that is interdisciplinary. For all the above reasons, namely the emerging and the interdisciplinary nature of this topic the few available techniques are far from mature (Hagenau et al., 2013; Kleinnijenhuis, 2013; L.-C. Yu et al., 2013). Taking a step in maturation of text-mining techniques in this context is the problem that occupies this research effort.

The problem statement in short is:

"It is not clear if the problem of intraday prediction accuracy in the foreign exchange market can be addressed through enhancement of text mining of available news-headlines released prior to it."

Additional supporting observations in the literature:

Lack of enhancement in text mining with regards to this problem is clear in the literature from a number of aspects that demand further investigation:

A) Lack of investigation on dimensionality reduction

Dimensionality reduction (DR) is an important research aspect when text mining is concerned (Berka & Vajteršic, 2013; Shi et al., 2011). In this research it is shown that in this context DR can help increase prediction accuracy significantly. However, past research has stopped short of enhancing this aspect of text mining in this context (Hagenau et al., 2013; Jin et al., 2013; Schumaker et al., 2012). When words produce features, it clearly leads to very large numbers of features which can easily lead to the curse of dimensionality (Pestov, 2013) that decreases the accuracy of most machine learning algorithms significantly. The way to mitigate this would be designing an

effective dimensionality-reduction mechanism (Gönen, 2014; Gracia, Gonzalez, Robles, & Menasalvas, 2014; Kyoungok Kim & Lee, 2014; Lu & Yuan, 2014). This work makes an effort to address this issue.

B) Lack of investigation on sentiment analysis

On the one hand, research has shown that the perception and attitude in other words sentiment of the investor regarding fundamental information plays a significant role in price movements in markets (Bacchetta & van Wincoop, 2013; Muehlfeld et al., 2013; ter Ellen et al., 2013; W. Yin & Li, 2014). On the other hand, sentiment analysis is a thriving field on its own with increasing number of applications (Ara et al., 2013; Balahur et al., 2014; Cambria et al., 2013; Di Caro & Grella, 2013; Young & Soroka, 2012; Zhou, Chen, & Wang, 2013).

However, Kleinnijenhuis (2013) indicates that the current approaches for automated sentiment analysis are still pretty much based on elementary techniques like word lists counts, named entity recognition and analyze combinations rather than on smart combinations of syntactic parsing, knowledge representation, logic and semantic web technologies when it comes to market prediction through text mining. Sentiment analysis can be better investigated and integrated in this context (Schumaker et al., 2012; L.-C. Yu et al., 2013). This work addresses this point-of-view to some extent as well.

C) Lack of investigation on semantic analysis

Linguistically, semantic analysis of text is the process of relating the text to concepts and meaning. In machine learning, it refers to the task of building structures that approximate concepts from a large set of documents. On the other hand, the first step in text mining is the structuring of text. Hence, structuring of text for the purpose of text mining or text classification can be enhanced by semantic analysis (Kwanho Kim et al.,

2014; C. H. Li, Yang, & Park, 2012; Nasir, Varlamis, Karim, & Tsatsaronis, 2013; Tang, Yan, & Tian, 2013; Yang, Li, Ding, & Li, 2013). But this has not been explored adequately in the literature in this context (Hagenau et al., 2013; Jin et al., 2013; Schumaker & Chen, 2009; Schumaker et al., 2012; Y. Yu, Duan, & Cao, 2013). This work addresses this aspect too.

In addition to the weaknesses regarding the state of text mining in this context, certain further lacks in the literature in support of the above problem are apparent that are also important to note:

D) Lack of investigation on FOREX

There exists a lack of work related to intraday prediction of currencies through text mining of unstructured news (Jin et al., 2013; Peramunetilleke & Wong, 2002), however, the FOREX is the largest asset class in the world with a $5.3 trillion dollar turnover per day in April 2013 according to the Bank for International Settlements and demands a lot more attention.

E) Lack of investigation on non-topic specific news:

In the past most of the literature regarding the relationship between news and markets has looked at the news released about a specific company (Hagenau et al., 2013; Schumaker et al., 2012). Unfiltered financial news has been rarely investigated.

F) Lack of investigation on text-mining of head-lines

News headlines are rarely used in this context as input as opposed to the text of the article-body (Peramunetilleke & Wong, 2002), however, they may be argued to be more straight-to-the-point and, hence, of less noise caused by verbose body text (C.-J. Huang et al., 2010).

1.4 Objectives

The main objectives of this work can be summarized as below:

- To devise a methodology to test the existence of a predictive relationship between the content of financial news and a foreign-exchange-market currency-pair
- To identify decisive text-mining elements that can improve the accuracy of such market prediction through news mining.
- To devise improvement-techniques for those text-mining elements

1.5 Research Questions

The main questions that this work strives to answer are:

- Is there a predictive relationship between financial news-headlines and a foreign-exchange-market currency-pair?
- What factors can cause the prediction-accuracy to be improved?
- Can an approach be developed to enhance those factors?

1.6 Motivation

The motivation for this research is at least two-fold: One, from a computer-science perspective and, two, from an economics perspective. Firstly, computer-science: in computer science artificial intelligence (AI) is the crown that brings all different aspects of the field together to advance human beings. One of the most effective and much needed areas of AI is in the natural language processing (NLP). The focus of this work on news-mining provides a framework to utilize AI to solve an instance of a widespread problem which is extraction of meaning out of textual manifestation of a natural language. Any meaningful improvement at this front in an age of information-overload directly facilitates progress in turning of currently readily available masses of data into

new information and knowledge that can improve the quality of critical day-to-day decision making.

From an economics perspective, the significance of behavioral-economics is becoming more and more apparent in the world of today. Behavioral-economics is the study of the effects of social, cognitive, and emotional factors on the economic decisions. This research provides an opportunity to put some of the principles of this new field to test in a very specific use-case, namely, the reaction of market-participants to news and its impact on the market.

The result of the above is the ultimate playground for the computational modeling of behavioral-economics at a macro scale which can lead into many new insights into nature of markets, human-behavior and news-communication.

1.7 Significance of Study

The significance of this study can be summarized on two fronts: 1- Real world impact and 2- Pure academic significance.

Firstly, the real world impact: Most modern economies of the world today are market based. Even a historically communist country like China that is the second biggest economy in the world is trying to develop a market-centric approach to its economy so that it can participate in and integrate into the global economy and benefit from it. In market-economies even currencies exchange rates are determined by markets. And market rates are determined by decisions made by the people. People make their decisions based on the information available to them and information is made available to people in news. Hence, in order to comprehend market dynamics around the globe it becomes paramount to comprehend the impact of news on it. Such comprehension not only facilitates increasing financial gains and loss-avoidance in business. But it also equips the economies of the future by means of better regulation to avoid dire crises

which occur once financial bubbles burst. Financial bubbles come to existence due to drastic market inefficiencies and a system like the one proposed in this work can mitigate such bubbles by improving market-efficiency; through improving the decisions that are made by market participants.

Secondly, the pure academic significance: This part has a number of aspects to it. One, it is exploring some new territories. Some researchers have studied the relationship between the stock market and the news but very few have done so regarding the foreign-exchange-market (FOREX). This research contributes to filling that gap. FOREX is very significant for many reasons like its being the biggest asset class in the world and its widely global and decentralized nature. Other researchers who have explored the relationship between stock market and news have done so in the frame of specific company news and hence this works contributes to filling a gap of using general financial news that is not filtered for a specific company or topic. Two, it is integrating and inter-disciplinary view on the problem that ties behavioral-economics, sentiment analysis and machine learning together. This inter-disciplinary awareness facilitates a well-rounded problem-solving framework. Three, it identifies the nature of factors that can improve prediction accuracy in this context which sets a clear path for future research. Four, the resulted methodology can be implemented in other markets and sources of textual data. Five, the dataset that is accumulated for system design and experiments in this work can be used in future works.

1.8 Scope of research

To ensure that this research maintains its focus and delivers on clearly set objectives the below scope is defined and stipulated:

1- This research is focused on a textual binary classification problem that is in a supervised-learning setup and unsupervised algorithms and clustering approaches are out of scope.
2- The textual source that this research investigates is the ‘news text ‘and other kinds of textual content are out of scope. Furthermore, it utilizes news-headlines to be specific rather than the body of news articles.
3- The type of prediction that is subject to this work is short-term. Specifically in a 1 to 3 hour time-frame after news release and investigation on other timeframe variations are out of scope.
4- The type of financial market that is chosen to be investigated in this research is the Foreign Exchange Market (FOREX) and other financial markets are out of scope.
5- The index that is tracked and investigated in the FOREX market in this research is the price of the Euro/USD currency pair and other currency pairs are out of scope.

1.9 Chapters’ Organization

The rest of the chapters of this thesis are organized as follows:

Chapter 2: Literature Review; Chapter 3: Research Methodology; Chapter 4: Detailed Design and Implementation; Chapter 5: Experimental Results & Discussion; Chapter 6: Conclusion.

2 Background and Literature Review

2.1 Introduction

Text mining is an up and coming field in much need of breakthroughs. The reason for text mining demanding increasing amounts of research is simply dominance of the Internet. The web has enabled real-time and social communication channels, through which tones of data is being transferred about any kind of news at every moment. It has also enabled a cloud repository that is growing exponentially. It is said that the amount of information being recorded in the web in every few minutes is now more than the total amount of information ever recorded in human history before the advent of the Internet. This is a staggering amount. However, the growing problem that is being faced at the same exponential rate is making sense of this world of data. And that is the research problem for data mining as a whole. Furthermore, the easiest form of documentation for humans remains to be unstructured text. Therefore, unstructured textual data is composing the vast majority of the data available. Dealing with text with an initial format that is unstructured poses even further challenges than the data mining of purely quantitative data. Hence, the field text mining on its own is recognized to focus on the specialized problems of dealing with text. But text mining itself is still too wide an area of research because there are so many differentiating factors involved that the researchers are forced to devote their attention to in order to solve very context specific problems. Although they try to transfer the learning from one problem to another but functional solutions are hardly fully transferrable from one context to the next, and such transfer efforts constitute full-fledged research initiatives. There are many factors that make contexts so different from each other, for example, the language involved. Languages have different semantic and syntactic rules and text mining algorithms made to deal with one are not instantly transferrable to another. Another example is the level of formality in speech, the content in the news is different from

email contents and that is different from the content on social media. One can go further to say that even the contents on different types of social media are different from one another. People use different conventions to write on Twitter, Facebook, etc. Some further examples of differentiating factors for text mining contexts are: Frequency – whether the content, that is to be taken into consideration by the algorithm, is released every millisecond or every month. Order – whether it is important to the algorithm if one content comes before or after another. Size – the average number of words to be considered in each piece. Metadata Availability – whether data about data is available in terms of time of release, publisher, etc. There are many more differentiating factors to think of, but these suffice for the argument that each of these factors has an impact on the design and effectiveness of the text algorithm that is developed and makes the performance of the algorithm very context-sensitive.

Therefore, it becomes vital to conduct research on specialized contexts in text mining. In the case of this work the specialized context for text mining is the mining of real-time news-headlines that are in English in order to predict intraday directional movements of a currency pair in the foreign exchange market. This context sets the above variables for the differentiating factors.

Besides the context that matters in research of text mining algorithms, there are a number of functions that an algorithm can focus on. For example, information extraction, text summarization, supervised or unsupervised learning methods, dimensionality reduction, probabilistic techniques, etc. A text mining algorithm like the one subject to this research has specific functions chosen from the above to focus on in order to increase performance and accuracy in the context that it is to be applied in.

Last but not least, it is of paramount importance to understand the specialized context as much as possible. In this work, the context is heavily related to economics. It is crucial

to establish at least from a theoretical perspective that predicting the market from text of news is actually relevant and acceptable from an economics point of view. Otherwise, an algorithm is built for a non-problem or a wrong problem. Furthermore, the algorithm may not address exactly the best functions that are desirable if the appropriate economic comprehension is lacking.

In this chapter, first a deeper dive into the text mining literature is made to identify relevant contexts and functions to this work. Then the theoretical legitimacy of the problem from an economics standpoint is reviewed and established based on the literature. Then a comparative survey is conducted of most of the significant text mining algorithms and efforts for comparable contexts in the past research in order to learn from them and identify their weaknesses. And in the final section a summary of this chapter is presented that reconfirms the problem statement and its treatment approach.

2.2 Text Mining Definition

Text mining or text data mining or text analytics refers to the process of deriving high-quality information from text. High-quality information is typically derived through drawing patterns and trends through means such as statistical pattern recognition. Text mining can be regarded as going beyond information access, which is the goal of information retrieval, to further help users analyze and digest information and facilitate decision making (Aggarwal & Zhai, 2012).

2.3 Text Mining Domains

Text mining is researched in numerous domains some of which are: Market Prediction, Business Intelligence (BI), Bioinformatics, Biomedical Literature Analysis, Internet Marketing, Online Advertisement Placement, Advertisement Campaign Monitoring, Sentiment Analysis, Opinion Mining, National Security/Intelligence, Social Media

Monitoring, Customer Relation Management (CRM), Trend Prediction, and Scientific Discovery.

Each of the above domains has multiple sub-domains each constituting a research area. In terms of market prediction, for example, there are different kinds of markets that can be taken into consideration. Stock markets, commodity markets, and the foreign exchange markets are just some of them. And inside each of these markets prediction scope of milliseconds to months can be considered. This work is investigating short-term prediction of the bidirectional movement of a currency pair in the foreign exchange market within the next 1 to 3 hours of news release.

2.4 Text Mining Objective: Predictive Binary Classification

Text mining can be utilized in different domains to realize different objectives. It may be used to cluster similar documents in order to achieve topic-based categorization. It could be used to extract key parts of the text in order to achieve text summarization. In the context of this work, the ultimate objective of the text mining system is to make a binary classification based on a group of news headlines in order to predict if the market (the price point of a currency pair) is going to go UP or DOWN. Therefore the news-headlines that cause the market to go up are supposed to be grouped in one class which could be labeled as UP or Positive-Movement and the rest in another class which could be termed as DOWN or Negative-Movement. As there are two classes involved it is termed a binary classification, which is a form of multi-class classification. Since it is supposed to be classifying the news headlines as they appear and before the impact on the market is clear, it is called a predictive classification. Hence, the main objective of this algorithm is a market predictive binary classification of news headlines,

2.5 Documents collection

In some cases, a data collection process may be needed. For example, in case of a web application, a software tool like a web crawler may be utilized to collect the documents. In another example, a logging process attached to an input data-stream may be used. For instance, an e-mail audit application may log all inbound and outbound messages at a mail server over a span of time (Weiss, Indurkhya, & Zhang, 2010).

In the case of news articles, a Really Simple Syndication (RSS) feed of the news website can be used for documents collection.

2.6 Text Mining Process Overview

There are generally at least 3 segments or sub-processes involved in any process of text mining that require paying attention to. They are:

1- Preprocessing or Data Structuring

2- Machine Learning or Statistical Pattern Recognition

3- Evaluation and Interpretation

Each of the above aspects of work are discussed in detail in the following sections. In the below figure these 3 main sub-processes of text mining are illustrated.

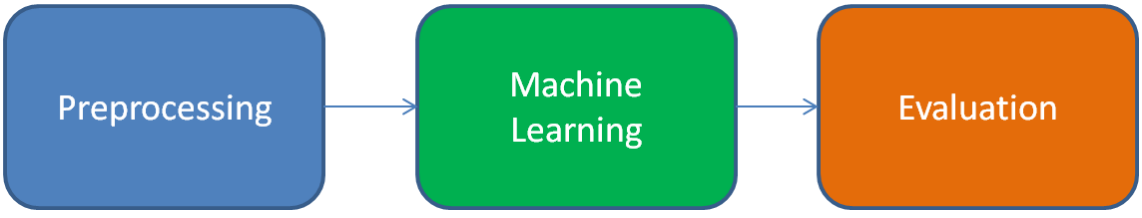

Figure 2.1 Text Mining Sub-Processes

2.7 Preprocessing

In data mining pre-processing is the step whereby data is prepared to be used as input in machine learning algorithms. In other words, the product of data pre-processing is the training set. It is a very important step as if it is not carried out properly it can easily result in a situation that can be termed as ‘garbage in, garbage’ which indicates nonsensical input to algorithms (garbage in) will most certainly lead to nonsensical outputs (garbage out).

The process includes activities like cleaning, normalization, transformation, feature extraction and selection, etc.

In text mining specifically pre-processing equals transformation of textual data into numeric data that is machine-readable. This is the step where unstructured format of text is turned into a structured format that can then be treated by machine learning algorithms somewhat similar to other data mining scenarios. Therefore, this is a significant step whose specialization for dealing with textual data differentiates text mining from data mining. It involves a number of key steps and concepts that are listed below:

2.7.1 Documents Standardization and Cleansing

Once the documents are collected, it is important to convert them to a standard format and cleanse them from any unnecessary words or characters.

2.7.2 Stop-words Removal

Stop-words are words that are common in most documents and do not correspond to any particular subject matter. In linguistics these words are sometimes called function words or closed-class words. They include pronouns (you, he, it) connectives (and, because, however), prepositions (to, of, before), and auxiliaries (have, been, can, should). Stop words may also include generic nouns (amount, part, nothing) and verbs (do, have).

The elimination of such words from the documents is referred to as stop-words removal. It occurs via using a pre-defined list of words.

2.7.3 Tokenization

The first step in dealing with text is to separate the stream of characters into tokens. This is essential to be able to proceed with analysis. Without defining and specifying the tokens, it is virtually not possible to extract higher-level information from the document. Dividing a stream of characters into tokens is simple for a human being who is aware of the language structure. However, a computer program, would find this task more challenging. The reason is that some characters act at times as token-delimiters and at other times they do not; this depends on the context they are used in. The characters space, tab, and newline are always assumed as delimiters and are not counted as tokens. They are normally collectively called 'white-space'. The characters ( ) < > ! ? " are always delimiters but can also be tokens. The characters . , : - ' may or may not be delimiters, based on their context (Weiss et al., 2010).

2.7.4 Document Representation (Feature Space Construction)

A corpus of text i.e. a collection of documents is represented as a two-dimensional matrix where each row describes a document and each column corresponds to a feature. The simplest form is to consider each word as a feature. Each entry in this matrix may take different forms; it may be a binary value of 1 if the feature exists or 0 if it doesn't. It may also have other forms of value based on some calculation of significance or weight. The resulted matrix is usually sparse and highly dimensional.

A sparse matrix is a matrix populated primarily with zeros.

A highly dimensional matrix is a matrix with a high number of columns.

2.7.4.1 Bag of words

Bag of Words (BOW) is the simplest and most common form for representation of the documents whereby each word in the document is considered as a feature.

Bag of words may be improved under some circumstances by introducing some abstraction via different techniques like stemming or named entity recognition as described below.

Understanding possible avenues for abstraction is central to this work as it devises an abstraction method for part of the proposed feature reduction algorithm.

2.7.4.2 Inflectional Stemming

In some languages like English, words appear in text in more than one form. For example, the nouns “bag” and “bags” are two forms of the same word. It is usually a good idea to omit this kind of variation before any further processing. This can be achieved by normalizing both words to the single form “bag”. This process is called “inflectional stemming”, if the normalization is limited to choosing one form to represent multiple grammatical variants such as singular/plural or present/past. From a linguistic perspective, however, this process is termed “morphological analysis” (Weiss et al., 2010).

An inflectional-stemming algorithm is partly rule-based and partly dictionary-based. Such a stemming algorithm is deemed to make some mistakes as it functions only on tokens, and has no more grammatical information such as part-of-speech. This is known as the problem of ambiguity. For instance, it is not clear whether the word “scared” is an adjective as in the phrase “she was scared” or is it the past-tense of the verb “scare” as in “you scared me”. (Weiss et al., 2010)

2.7.4.3 Stemming to a Root

More thorough stemming than inflectional stemming can be beneficial for some types of text-processing applications. The goal of such stemmers is to obtain a root form with no inflectional or derivational prefixes and suffixes. For instance, the term “deformation” can be reduced to the stem “form.” The end goal of such stemming is to make distributional statistics more reliable by decreasing the number of types in a text collection to a great extent. In addition to that, words with the same core meaning are grouped together, so that a concept such as “class” has only one stem, in spite of the fact that in the text there may be “classified”, “classification”, etc. (Weiss et al., 2010)

2.7.4.4 Named Entity Recognition

Named-entity recognition (NER) is to classify elements in text into pre-defined categories such as the names of persons, organizations, locations, expressions of times, quantities, monetary values, percentages, etc.

2.7.5 Feature Representation and Weighting

So far it has been discussed how a text corpus is represented in terms of a matrix with documents as rows and features as columns. However the actual value that may be put in each cell of the matrix for each of the columns dedicated to a document may vary. Below some of the fundamental variations are discussed.

2.7.5.1 Binary

The binary form is the most common and convenient form where if a feature exists in a document a ‘1’ is put in the matrix cell accordingly and if it does not exist a ‘0’.

In this form no feature is preferred in any sense to another. However, it makes sense as some features may be more important than others; hence, it makes sense to come up with weighting schemes that facilitate this.

2.7.5.2 Term Frequency (TF)

The simplest form to create a weighting scheme would be to just use the frequency of the appearances of a term i.e. feature in a document as the according value in the matrix. This way the terms that are more frequent receive a higher weight and the assumption is that the higher the frequency the more the significance of a term.

2.7.5.3 Term Frequency-Inverse Document Frequency (TF-IDF)

The assumption of the Term Frequency may not always be true as there may be terms that appear in many documents frequently but as they are in too many documents they cannot possible play the role of a differentiating feature for any one document in any sense. Therefore a more advance weighting scheme is devised by the name of Term Frequency-Inverse Document Frequency (TF-IDF). The TF-IDF value increases proportionally to the number of times a word appears in the document, but is offset by the frequency of the word in the corpus, which helps to control for the fact that some words are generally more common than others.

The **inverse document frequency** is a measure of whether the term is common or rare across all documents. It is obtained by dividing the total number of documents by the number of documents containing the term, and then taking the logarithm of that quotient.

$$idf(t, D) = log \frac{N}{|\{d \in D : t \in d\}|}$$

(2.1)

with

- N: *total number of documents in the corpus*
- $|\{d \in D : t \in d\}|$ *: number of documents where the term* t *appears*

If the term is not in the corpus, this will lead to a division-by-zero. It is therefore common to adjust the denominator to:

$$1 + |\{d \in D : t \in d\}|$$

(2.2)

Mathematically the base of the log function does not matter and constitutes a constant multiplicative factor towards the overall result.

Then TF-IDF is calculated as:

$$tfidf(t, d, D) = tf(t, d) \times idf(t, D)$$

(2.3)

A high weight in TF-IDF is reached by a high term frequency (in the given document) and a low document frequency of the term in the whole collection of documents; the weights hence tend to filter out common terms. Since the ratio inside the IDF's log function is always greater than or equal to 1, the value of IDF (and TF-IDF) is greater than or equal to 0. As a term appears in more documents, the ratio inside the logarithm approaches 1, bringing the IDF and TF-IDF closer to 0.

2.7.6 Labeling

In order to use a feature matrix for prediction an extra column needs to be added to it. This column contains a label. It is a one or zero indicating that the correct answer is either true or false. A label can be a topic to classify the document. Science or history are examples of topics. Any answer that can be perceived as true or false is acceptable. It could be a topic or class, or it could be an article that appeared before a rise in a stock price. Any definition for a class is acceptable so long as the answers are labeled correctly relative to the concept (Weiss et al., 2010).

2.7.7 Feature Selection

Feature selection is the study of algorithms for reducing dimensionality of data to improve machine learning performance. For a dataset with N features and M dimensions (or features, attributes), feature selection aims to reduce M to N where N < M. An Assumption of feature selection is that we have defined an original feature space that can be used to represent the data, and our goal is to reduce its dimensionality by selecting a subset of original features. The original feature space of the data is then mapped onto a new feature space (Brank, Mladenić, & Grobelnik, 2010).

The role of feature selection in machine learning is:

1. To reduce the dimensionality of the feature space

2. To speed up a learning algorithm

3. To improve the predictive accuracy of a classification algorithm

4. To improve the comprehensibility of learning results

2.7.8 Dimensionality Reduction

Every data object in a computer is represented and stored as a set of features, for instance, size, color, price, and so on. The term ‘dimension’ can be used instead of the term ‘feature’, because an object with n features can also be represented as a multidimensional point in an n-dimensional space. Hence, dimensionality reduction refers to the process of mapping an n-dimensional point, into a lower k-dimensional space. This operation reduces the size for representing and storing an object or a dataset generally; therefore, dimensionality reduction can be seen as a method for data compression. Additionally, this process promotes data visualization, particularly when objects are mapped onto two or three dimensions. Finally, in the context of classification, dimensionality reduction can be useful for the following: (a) making

classification schemes that are super-linear with respect to dimensionality, (b) reducing the variance of classifiers that need it in higher dimensionalities, and (c) removing the noise that may be present, thereby improving classification accuracy (Vlachos, 2010).

2.7.9 Dimensionality Reduction via Semantic Abstraction

The bag-of-words representation is commonly used in text analysis, because it can be analyzed very efficiently and retains a great deal of useful information. However, it is also troublesome because the same concept can be referred to using many different terms (Co-reference) or the same term can have many different meanings (ambiguity). Dimensionality reduction can group together terms that have the same semantics, to identify and disambiguate terms with multiple meanings and to provide a lower-dimensional representation of documents that reflects concepts instead of raw terms (Aggarwal & Zhai, 2012).

It is common to represent a document as a bag of words (BOW), noting the number of occurrences of each term but dismissing the order. This type of representation is good for computational efficiency while maintaining the content of a document. The resulted vector representation can go through dimensionality reduction which is actually a technique from applied mathematics. It is used to identify a lower-dimensional representation of a set of vectors that preserves important properties. BOW vectors typically have a very high dimensionality, because each dimension corresponds to one term from the language or the dictionary. However, a lower-dimensional semantic space is needed for the task of analyzing the concepts present in documents, where each dimension corresponds to one concept or one topic. An appropriate application of dimensionality reduction can find this semantic space and its relationship to the BOW representation. The new representation in semantic space reveals the topical structure of the corpus more clearly than the original representation (Aggarwal & Zhai, 2012). This is referred to as Semantic Abstraction in this text.

2.7.10 Sentiment Analysis

Sentiment Analysis deals with detecting the general sentiment that is available in online resources and social media to understand how people feel about a topic. For example it can be used to determine consumer sentiment towards a brand (Ghiassi et al., 2013; Mostafa, 2013). F. Moraes et al. (2013) analyze the polarity of micro-reviews in Foursquare, which is one of the currently most popular location-based social networks. Such sentiment or polarity classification provides useful tools for opinion summarization, which can help business owners as well as potential new customers to quickly obtain a predominant view of the opinions posted by users at a specific venue.

There is a body of research that is focused on sentiment analysis or the so called "opinion mining" (Balahur, Steinberger, Goot, Pouliquen, & Kabadjov, 2009; Cambria et al., 2013; Hsinchun & Zimbra, 2010). It is mainly based on identifying positive and negative words and processing text with the purpose of classifying its emotional stance as positive or negative. An example of such sentiment analysis effort is the work of Maks and Vossen (2012) which presents a lexicon model for deep sentiment analysis and opinion mining. A similar concept can be used in other area of research like emotion detection in suicide notes as done by Desmet and Hoste (2013).

Sentiment of short online textual snippets like tweets is actively studied as well (Ghiassi et al., 2013; Ikeda, Hattori, Ono, Asoh, & Higashino, 2013; Kontopoulos, Berberidis, Dergiades, & Bassiliades, 2013). Tweets are also studied in relation to prediction of stock markets (J. M. Bollen, Huina; Zeng, Xiao-Jun, 2010).

Sentiment analysis of news can be a good source to tap into for market prediction as news expresses the point of view and sentiment of opinion leaders, forms the public opinion to an extent and triggers public reactions. The impact of news on stock markets of different companies and regions has been the target of emerging extensive research

(Hagenau et al., 2013; C.-J. Huang et al., 2010; Nizer & Nievola, 2012; Schumaker et al., 2012) .

Schumaker et al. (2012) have tried to evaluate the sentiment in financial news articles in relation to the stock market in his research but has not been completely successful. One more successful recent example of this would be (L.-C. Yu et al., 2013), where a contextual entropy model was proposed to expand a set of seed words by discovering similar emotion words and their corresponding intensities from online stock market news articles. This was accomplished by calculating the similarity between the seed words and candidate words from their contextual distributions using an entropy measure. Once the seed words have been expanded, both the seed words and expanded words are used to classify the sentiment of the news articles. Their experimental results show that the use of the expanded emotion words improved classification performance, which was further improved by incorporating their corresponding intensities which caused accuracy results to range from 52% to 91.5% by varying the difference of intensity levels from positive and negative classes from (-0.5 to 0.5) to >9.5 respectively. It is also interesting to note that emotional analysis of text does not have to be mere based on positivity-negativity and it can be done on other dimensions or on multi-dimensions (Ortigosa-Hernández et al., 2012). A recent piece of research by Loia and Senatore (2014) introduces a framework for extracting the emotions and the sentiments expressed in the textual data. The sentiments are expressed by a positive or negative polarity. The emotions are based on the Minsky's conception of emotions that consists of four affective dimensions (Pleasantness, Attention, Sensitivity and Aptitude). Each dimension has six levels of activation, called sentic levels. Each level represents an emotional state of mind and can be more or less intense, depending on the position on the corresponding dimension. Another interesting development in sentiment analysis is an effort to go from sentiment of chunks of text to specific features or aspects that are

related to a concept or product; Kontopoulos et al. (2013) propose a more efficient sentiment analysis of Twitter posts, whereby posts are not simply characterized by a sentiment score but instead receive a sentiment grade for each distinct notion in them which is enabled by the help of an ontology. W. Li and Xu (2014) take on another angle by looking for features that are ‘‘meaningful’’ to emotions instead of simply choosing words with high co-occurrence degree and thereby create a text-based emotion classification using emotion cause extraction.

Surprisingly, there are too few research efforts on impact of textual data in news content on the foreign exchange market for example Peramunetilleke and Wong (2002) is one of the rare examples that studies impact of news-headlines on FOREX. Evans and Lyons (2008) term a phenomenon as “news puzzle”, and argue that directional effects are harder to detect in exchange rates since they are likely to be swamped by other factors. A recent work of Chatrath et al. (2014) studies the currency jumps, cojumps, however, it is based on the role of macro-news which entails structured scheduled macro-economic news announcements that reveal specific indexes like the unemployment rate of a country or its inflation rate etc.

2.8 Machine Learning

2.8.1 Classification Problem

Classification is a well-understood problem. A sample is collected. The data are organized in a structured format. Every example is measured in the same way. The answer is expressed in terms of true or false, a binary decision. In mathematical terms, a solution is a function that maps examples to labels, $f : w \rightarrow L$, where w is a vector of attributes and L is a label. In our case, the attributes are words or tokens (Weiss et al., 2010).

The labels can be a goal that is potentially related to the words. Most prior research has been done on indexing, where the label is a broad topic, such as categorizing a document as a financial story. But the label could be anything from a political event to the direction of a currency-pair in FOREX (Weiss et al., 2010).

2.8.2 Applicable Machine Leaning Algorithms

In general there are four types or categories of machine learning algorithms that can be applied in a textual classification problem. These are:

1- k-Nearest Neighbors (k-NN)

2- Decision Rules

3- Probabilistic methods

4- Weighted-scoring methods. (Weiss et al., 2010)

2.8.2.1 k-Nearest Neighbors (k-NN)

A common algorithm to approach the problem of information retrieval in the context of text-mining is called k-Nearest Neighbors. A complete document will have many words, and it is unlikely that it will completely match a stored document. Instead of an exact match, one can try to find the closest matches to the stored documents. This can be done, for example, by pulling out the ten best matches, looking at their labels, and picking the label that occurs most frequently. This simple process is the basic algorithm for k-Nearest Neighbors. It is one of the most widespread approaches to prediction relating to text (Weiss et al., 2010).

The formal name for the method is k-Nearest Neighbors, but finding k is not so straightforward and should usually be estimated by experimental procedures (Weiss et al., 2010).

Most nearest-neighbor applications compare two examples by measuring the distance between them. The below equation is a general distance measure used to compare two examples. In this case, it is simply the square of the difference between each attribute. Absolute values may also be used. The larger the distance, the weaker the connection between the examples (Weiss et al., 2010).

$$Distance(x, y) = (x_1 - y_1)^2 + \cdots + (x_m - y_m)^2$$

(2.4)

"Similarity" is a more intuitive measure that is used for text. The most basic measure of similarity is to count the number of words that two documents have in common.

There are several ways of representing the measurements in the feature matrix; binary, term-frequency, and TF-IDF are some of them. In practice, using the TF-IDF variation will usually give the best predictive results for similarity methods. The actual computation of distance in this case is called the cosine-similarity and has been widely used for information retrieval (Weiss et al., 2010).

The nearest-neighbor method requires no special effort to learn from the data and provides no special value in finding generalized patterns in the data. It is just a retrieval program and under the best of circumstances will require more computation time to apply than most other methods. The main advantage is the virtually zero training effort, which means that one can just collect the documents and store them (Weiss et al., 2010).

2.8.2.2 Decision Rules & Trees

Decision rules and trees is another set of algorithms that can be used for text-mining problems. A rule can be a phrase of one or more words that must occur together to match a document. Once a new, unlabeled document is presented, its label is assigned depending on whether any of the rules are satisfied (Weiss et al., 2010).

In many categorization applications, the main objective is to get the label right. For decision rule categorizers, however, the objective is wider. Since the rules consist of words, and words have meaning, the rules themselves can be insightful. They can expand our knowledge and suggest reasons for reaching a conclusion. Beyond just attempting to assign a label. For example, the rules may suggest a pattern of words found in news articles prior to the rise of the rate of a currency-pair in FOREX. The disadvantage of rules is that they can be less predictive if the underlying concept is complex. However, even in such situations, they can lead to hints into the nature of the key predictive words and phrases (Weiss et al., 2010).

Although decision rules can be particularly satisfying solutions for text mining, the procedures for finding them are more complicated than other methods. The expectation is that a relatively small number of words and phrases will provide a good solution. Yet, the search for these words and phrases that distinguish one class from the other can be time-consuming and complex (Weiss et al., 2010).

Decision trees are special decision rules that are organized into a tree structure. A decision tree divides the document space into non-overlapping regions at its leaves, and predictions are made at each leaf (Weiss et al., 2010).

2.8.2.3 Probabilistic Methods (Naïve Bayes)

The most obvious method of classification is direct lookup of the probabilities of words in a document. Let C be the class label we are interested in and x be a feature vector that denotes the presence or absence of words from a dictionary. Mathematically, the objective is to estimate Pr(C|x), the probability of a class, given the presence or absence of words from a dictionary. For singly labeled document collections, we can choose the category C that has the largest probability score Pr(C|x). For multiply labeled document collections, if our interest is to maximize the accuracy, then C is selected whenever

Pr(C|x) is greater than 0.5. Another way to look at the multiply labeled case is that for each label we divide the document collection into two classes: one class with label C and the other class with a label that is not C. Therefore, we have a binary classification problem for each label value C. This is reasonable because the multiple labels assigned to documents are usually independent of each other, and hence it is possible to view each label assignment as a separate classification problem with two classes (labeled and not labeled). It thus suffices to consider the binary class problem (Weiss et al., 2010).

However, we know that, even for this problem, a complete computation of probability is impossible. Even a 100-word dictionary has many possible combinations. However, a simplified approach to probability estimation, called Bayes with independence or naive Bayes, has often been attempted. The mathematics is straightforward and the computation is efficient, which leads to wide application of this approach, especially in applications where a quick implementation takes priority over accuracy (Weiss et al., 2010).

$$\Pr(C|x) = \Pr(x|C) \times \frac{\Pr(C)}{\Pr(x)}$$

(2.5)

Bayes' rule is given in the above, where *C* is the class of interest and *x* is a vector of ones and zeros corresponding to the presence or absence of dictionary words for a specific document. When there are two classes, C_1 and C_2, *Pr(x)* is readily computed as below (Weiss et al., 2010).

$$\Pr(x) = \Pr(x|C_1)\Pr(C_1) + \Pr(x|C_2)\Pr(C_2)$$

(2.6)

The key to using these equations is to compute *Pr(x|C)*. If we assume that the words are independent then instead of looking up the probability of the complete vector of *x*, we can look up the probability of the presence or absence of each word, $Pr(x_j|C)$, and multiply them all together. We use x_j to denote the j-th component of *x*. The equation below states this mathematically (Weiss et al., 2010).

$$\Pr(x|C) = \prod_j \Pr(x_j|C)$$

$$\Pr(x) = \sum_C \Pr(C) \prod_j \Pr(x_j|C)$$

(2.7)

The conditional probabilities in (2.7) are readily estimated if one uses the simple binary presence or absence of a word as a feature value that would give only two possible values for each feature (Weiss et al., 2010).

The probability estimates are easy to obtain from the feature matrix. *PrI* is determined from the frequency of ones in the last column divided by n, the number of examples, *freqI/n*. Each x_j is either a 1 or a 0 (presence or absence of the word w_j). The quantity $Pr(x_j = 1|C)$ is computed from the frequency of ones for x_j , where only the examples labeled *C* are considered, $freq(x_j = 1, label = C)/freqI$. The probability of w_j not occurring in *C*, $Pr(x_j = 0|C)$, is $1-Pr(x_j = 1|C)$ (Weiss et al., 2010).

The performance on text benchmark applications for naive Bayes is usually weaker than for the other methods. However, it needs almost no memory, and requires little computation. It usually works best with a relatively small dictionary representing the key words needed to make a decision for that class (Weiss et al., 2010).

2.8.2.4 Support Vector Machine (Linear Scoring Methods)

In order to achieve good prediction performance, it is often necessary to create a feature vector of very high dimensions. Although many of the features are not useful, it can be difficult for a human to tell what feature is useful and what feature is not. Therefore, the prediction algorithm should have the ability to take a large set of features and then select only useful features from the full set. A very useful method to achieve this is by using linear scoring (Weiss et al., 2010).

The naive Bayes method described above can be regarded as a special case of the linear scoring method. However, the performance can be significantly improved using more sophisticated training methods to obtain the weight vector $w = [w_j]$ and bias b (Weiss et al., 2010).

Mathematically, this method is a linear scoring function. The general form is below, where D is the document and w_j is the weight for the j-th word in the dictionary, b is a constant, and x_j is a one or zero, depending on the j-th word's presence or absence in the document (Weiss et al., 2010).

$$Score(D) = \sum_j w_j x_j + b = w.x + b$$

(2.8)

Linear scoring methods are classical approaches to solving a prediction problem. The weaknesses of this method are well-known. Geometrically, the method can be described as producing a line or hyperplane. Although a line cannot fit complex surfaces, and a curvy shape might be needed, it is often possible to create appropriate non-linear features so that a curve in the original space lies in a hyperplane in the enlarged space with the additional nonlinear features. In this way, nonlinearity can be explicitly captured by constructing sophisticated nonlinear features. An advantage of this

approach is that the modeling aspect becomes conceptually very simple since we can focus on creating useful features and let the learning algorithm determine how to assign a weight to each feature we create. Another advantage is that the linear scoring method can efficiently handle sparse data. This is important for text-mining applications since although feature vectors can have high dimensionality, they are usually very sparse (Weiss et al., 2010).

Word in Model	***Weight***
dividend	0.8741
earnings	0.4988
eight	-0.0866
extraordinary	-0.0267
months	-0.1801
payouts	0.6141
rose	-0.0253
split	0.9050
york	-0.1908
…	…

Words in example doc	***Score***
dividend, payout, rose	1.4629

Figure 2.2 Example for Weighted Scoring

It is known from various benchmarks that the linear scoring approach does surprisingly well on text classification (Weiss et al., 2010). The simple naive Bayes methods have severe problems with redundant attributes, which in text corresponds to words that behave like synonyms. Classical methods were developed to handle a small number of attributes, certainly not the tens of thousands of words in a global dictionary. The newer linear methods are oblivious to these limitations. A major advance in linear methods for text has been their ability to work with huge dictionaries and find weights for every word in a complete dictionary. If there are ten synonyms, it can weigh each one. This capability to work with so many words and weigh all of them both positively and negatively seems to capture the subtleties of language, where some words are precise and strong predictors and others are vague and weak predictors (Weiss et al., 2010).

The key problem with these weighted-scoring methods is that of learning the weights, the second column in Figure 2.2. The words are those in the dictionary, and the weights for them will be learned from a collection of documents. The label is assigned by applying the mentioned general distance formula (2.8). However, the method is a mathematical process, an application of numerical analysis (Weiss et al., 2010).

The representation of the words can be binary, but the TF-IDF transformation usually yields better results (Weiss et al., 2010).

The two-class prediction problem is considered to be one that determines a label $y \in \{-1, 1\}$ from an associated vector *x* of input variables. Given a continuous model *p(x)*, the following prediction rule is considered:

$$\textit{predict } y = 1 \textit{ if } p(x) \geq 0, \textit{ and}$$

$$\textit{predict } y = -1 \textit{ otherwise.}$$

(2.9)

The classification error is:

$$I(p(x), y) = \begin{cases} 1 \; if \; p(x)y \leq 0, \\ 0 \; if \; p(x)y > 0. \end{cases}$$

(2.10)

A useful method for solving this problem is by linear predictors. These consist of linear combinations of the input variables $p(x) = w \cdot x + b$, where *w* is often referred to as weight and *b* as bias. *(w, b)*is called the weight vector and the term *bias* is used for statistical bias (Weiss et al., 2010).

Let (x^i, y^i) be the i-th row of the spreadsheet, where x^i is the vector representation of the i-th training data, and y^i represent the label, which takes the value *1* if the document

belongs to category *C* and value *−1* otherwise. A very natural way to compute a linear classifier is by finding a weight $(\hat{w}, \hat{b})$ that minimizes the average classification error in the training set (Weiss et al., 2010):

$$(\hat{w}, \hat{b}) = \arg min_{w,b} \frac{1}{n} \sum_{i=1}^{n} I(w.x^i + b, y^i)$$

(2.11)

Unfortunately, this formulation leads to a non-convex optimization problem which may have many local minima. Finding the global optimal solution is generally very hard (to be mathematically precise, it is NP-hard). It is thus desirable to replace the non-convex classification error loss *I (p,y)* with a convex formulation that is computationally more desirable (Weiss et al., 2010).

In order to motivate a convex formulation, we shall first consider the situation where the problem is linearly separable. In this case, we want to find a linear separator *(w, b)* such that $w.x^i + b < 0$ when $y^i < 0$ and $w.x^i + b > 0$ when $y^i > 0$. That is, we want to find *w* such that $(w.x^i + b)y^i > 0$ for all *i*. Note that if a linear separator *w* exists, then there are more than one linear separators, since a small perturbation of *w* still separate the data. It is thus preferable to find an "optimal" linear separator. An idea, popularized by Vapnik, is to find the most stable linear separator, so that any small perturbation of x^i does not change the classification rule. The stability of a linear separator's prediction on the i-th point (x^i, y^i) can be measured by its margin (Weiss et al., 2010):

$$\gamma^i = \frac{(w.x^i + b)y^i}{||w||}$$

Where

$$||w||^2 = w.w = \sum_j w_j^2$$

(2.12)

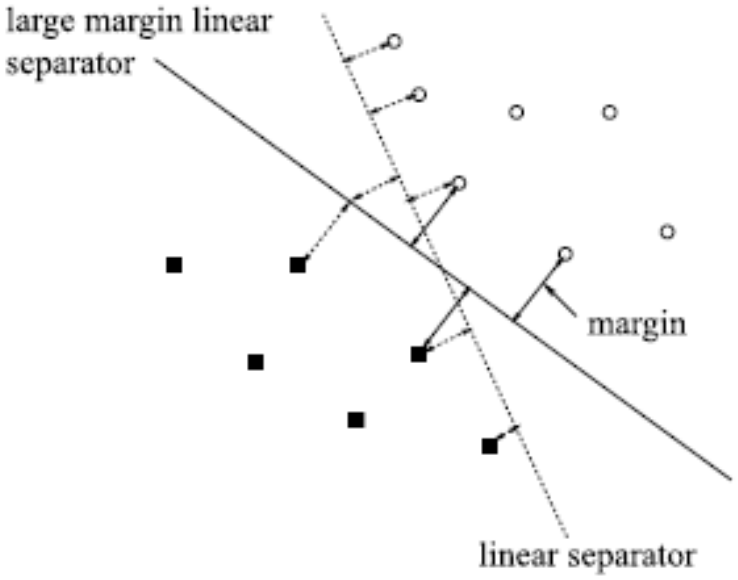

Figure 2.3 Linear Separator with Largest Margin

If the margin $\gamma^i$is large, then the prediction on this data point is stable. In order to change the sign of the prediction, we have to move

$$x^i \rightarrow x^i + \Delta x^i$$

Such that

$$\left(w.\left(x^i + \Delta x^i\right) + b\right)y^i < 0$$

That is

$$\frac{-w.\Delta x^i y^i}{||w||} > \gamma^i$$

(2.13)

The larger γ^i is, the larger modification Δx^i is needed to switch the sign (Weiss et al., 2010).

The margin idea is illustrated in Figure 2.3. We want to find a linear separator such that the smallest γ^i is as large as possible. Mathematically, the optimization problem for finding the linear separator with the largest margin is (Weiss et al., 2010):

$$\frac{min_{i=1,\dots,n}(w.x^i + b)y^i}{||w||}$$

(2.14)

A more popular method, which is equivalent to the above formulation, is to minimize ||*w*|| under the constraint *mini(w · xi + b)yi ≥ 1*. That is, the optimal hyperplane is the solution to

$$[\widehat{w}, \hat{b}] = \arg min_{w,b}||w||^2$$

Subject to

$$\left(w.x^i + b\right)y^i \geq 1\ (i = 1, \dots, n)$$

(2.15)

If the data is not linearly separable, then the idea of margin maximization cannot be directly applied (Weiss et al., 2010). Instead, one considers the so-called ***soft-margin formulation*** as follows:

$$[\widehat{w}, \hat{b}] = \arg min_{w,b}[\left|\left|w\right|\right|^2 + C\sum_{i=1}^{n}\varepsilon^i]$$

Subject to

$$y^i\left(w.x^i + b\right) \geq 1 - \varepsilon^i \quad \varepsilon^i \geq 0(i = 1, \dots, n)$$

(2.16)

The parameter ε^i is a variable that allows the margin constraint for each point *i* to be violated (when $\varepsilon^i \geq 0$). However, a linear penalty is included into the optimization when margin constraint is violated. The larger *C* is, the heavier the penalty of such violation. In particular, if $C \rightarrow \infty$, we obtain back the original separable margin maximization formulation (Weiss et al., 2010).

By eliminating ε^i from the formula (2.16), and letting $\lambda = 1/(nC)$, we obtain the following equivalent formulation (Weiss et al., 2010):

$$[\hat{w}, \hat{b}] = \arg min_{w,b}[\frac{1}{n}\sum_{i=1}^{n} g\left(\left(w.x^i + b\right)y^i\right) + \lambda||w||^2]$$

Where

$$g(z) = \begin{cases} 1 - z & if\ z \leq 1, \\ 0 & if\ z > 0. \end{cases}$$

(2.17)

This method, often referred to as *Support Vector Machine (SVM)*, can be regarded as a modification of the classification error minimization formulation, where we replace classification error minimization by minimizing the following upper bound (Weiss et al., 2010):

$$min_{w,b}\frac{1}{n}\sum_{i=1}^{n} g(w.x^i + b, y^i)$$

(2.18)

2.9 Evaluation

The standard statistical model assumes that a sample is randomly drawn from some general population. The new examples are unlabeled, and their labels will be assigned. To evaluate the performance of a solution, we train on one sample and test on another

sample. Typically, our data might be randomly divided into two parts: one for training and one for testing (Weiss et al., 2010).

Once the data are split into training and testing samples, learning takes place exclusively on the training set. Performance can be estimated in terms of several measures. The standard measure for classification is the error rate below (2.19) and its standard error is given there too (2.20). The error rate is binomially distributed and is approximately normal. Two standard errors are often used to approximate 95% confidence bounds (Weiss et al., 2010).

$$Error\ rate\ (erate) = \frac{number\ of\ errors}{number\ of\ documents}$$

(2.19)

$$Standard\ Error\ (SE) = \sqrt{\frac{erate * (1 - erate)}{number\ of\ documents}}$$

(2.20)

Although error-rates and associated standard errors are useful for estimating the performance of predictors in general, for most text applications, such as text categorization, a more detailed analysis of the errors is desirable. For information retrieval applications, there is usually a large number of negative data. A classifier can achieve a very high accuracy (i.e., a very low error rate) by simply saying that all data are negative. It is thus useful to measure the classification performance by ignoring correctly predicted negative data and then examining the sorts of errors made by the classifier (Weiss et al., 2010). Three ratios have achieved particular prominence: precision, recall, and F-measure. Their definitions are given below:

$$precision = \frac{number\ of\ correct\ positive\ predictions}{number\ of\ positive\ predictions}$$

(2.21)

$$recall = \frac{number\ of\ correct\ positive\ predictions}{number\ of\ positive\ class\ documents}$$

(2.22)

$$F - measure = \frac{2}{\frac{1}{precision} + \frac{1}{recall}}$$

(2.23)

Precision, recall, and their combination in the F-measure are all more interesting measures of the quality of binary decisions on documents.

The following example illustrates these measures of performance. Let's assume that there is a database of labeled documents. Let's focus on a particular label, such as sports. Now consider a classifier that labels documents as sports or not, and let's use it to retrieve all the documents that it labels. We can assess the performance of the classifier from the set of retrieved documents by computing the three measures as follows (Weiss et al., 2010):

- The percentage of all sports documents that are retrieved is the recall.
- The percentage of documents that it correctly labels as sports is the precision.
- F-measure is defined as the harmonic mean of precision and recall. It is often used to measure the performance of a system when a single number is preferred.

Because document collections are typically large, high precision is often more valued. For high precision, the computer's positive decisions are usually correct, but it may fail to catch all positives (this is measured by recall). Thus, if a program identifies spam e-

mail with high precision and low recall, it may often leave spam in your Inbox (low recall), but when it puts a spam document in the trash, it is usually correct (high precision) (Weiss et al., 2010).

Since precision and recall measure different kinds of errors, if the overall error rate remains the same, increasing the precision (reducing one kind of error) lowers the recall (increases the other kind of error). This leads to a ***precision–recall tradeoff***. Most classifiers have a way of making this tradeoff by a simple variation of a constant. For the classifiers discussed in this chapter, the process is as follows (Weiss et al., 2010):

- For k nearest-neighbor methods, the threshold can be varied from a simple majority to some other value. For example, if five nearest neighbors being used classify a document within a topic, instead of requiring that three of the five nearest neighbors belong to the topic, one may change this threshold to a different value. A value less than 3 would boost recall, whereas values greater than 3 would boost precision.
- For decision rules, the cost of different kinds of errors can be altered. For example, if false negative errors are made twice as costly as false positive errors, then recall would be boosted.
- For probabilistic scoring (Naïve Bayes), the threshold for a class can be altered from 0.5 to some other value. Lower thresholds would boost recall, while higher values would boost precision.
- For linear models (SVM), the threshold can be changed from zero to a different value. Lower values would help recall, and higher values would boost precision.

It is important to pay attention to this tradeoff by varying the constant and try to find the best solutions (Weiss et al., 2010).

2.10 Theoretical Economic Legitimacy

The problem that this research investigates is the feasibility of adequately accurate short-term predictions in a financial market via text mining of written information available as news.

In approaching the above problem, it is crucial to investigate and establish the feasibility and legitimacy of such predictability in financial markets from a theoretical standpoint in economics. This is accomplished by reviewing the literature on economics from the below three perspectives:

1- Conventional Economic Theory

2- Behavioral Economic Theory

3- Market Prediction Avenues: Fundamental vs. Technical Analysis

It is discussed in the following how each of the above perspectives contributes to this research.

2.10.1 Conventional Economic Theory

In conventional economics, Efficient-Market Hypothesis or EMH (Fama, 1965) asserts that when financial markets are "informationally efficient", one cannot consistently achieve returns in excess of average market returns on a risk-adjusted basis, given the information available at the time the investment is made. EMH is further refined later to include 3 levels of efficiency as strong, semi-strong and weak (Fama, 1970). This indicates that there are many markets where predictability is plausible and viable and such markets are termed as "weakly efficient". It also indicates that 'market efficiency' i.e. 'unpredictability' is correlated with 'information availability' and a market is only "strongly efficient" when all information is completely available, which realistically is rarely the case. Hence, Fama concedes that his theory is stronger in certain markets

where the information is openly, widely and instantly available to all participants and it gets weaker where such assumption cannot be held concretely in a market.

When markets are weakly efficient then it must be possible to predict their behaviour or at least determine criteria with predictive impact on them. Although, the nature of markets is as such that once such information is available, they absorb it and adjust themselves and therefore become efficient against the initial predictive criteria and thereby making them obsolete. Information absorption by markets and reaching new equilibriums constantly occur in markets and some researchers have delved into modelling its dynamics and parameters under special circumstances (García & Urošević, 2013). Nonetheless, the existence of speculative economic bubbles indicates that the market participants operate on irrational and emotional assumptions and do not pay enough attention to the actual underlying value. The research of (Potì & Siddique, 2013) indicates existence of predictability in the Foreign Exchange Market (FOREX) too, which is the focus of the experiments in this work. Furthermore, it has been demonstrated that markets in the emerging economies like Malaysia, Thailand, Indonesia, and the Philippines tend to be significantly less efficient compared to the developed economies and hence predictive measures like technical trading rules seem to have more power (H. Yu et al., 2013). Additionally, the short-term variants of the technical trading rules have better predictive ability than long-term variants plus that markets become more informationally efficient over time (H. Yu et al., 2013). Hence, there is a need to continuously revisit the level of efficiency of economically dynamic and rapidly growing emerging markets (H. Yu et al., 2013).

In summary, literature speaks to the fact that markets emerge and mature. What renders any prediction ever possible in a financial market is called ‘information-inefficiency’. It means that markets absorb information as it is made available and converge to new levels accordingly. Hence, the speed and quality of information dissemination in a

geographical location has to do with market convergence time and only if the convergence time is not instant, the market is considered inefficient. Predictability is only feasible once inefficiency occurs. Therefore, as markets and economies emerge and mature the level and nature of predictability needs to be revisited from time to time and from market to market.

2.10.2 Behavioral Economic Theory

Cognitive and behavioural economists look at price as a purely perceived value rather than a derivative of the production cost. The media do not report market status only, but they actively create an impact on market dynamics based on the news they release (Robertson et al., 2006; Wisniewski & Lambe, 2013).

People's interpretation of the same information varies greatly. Market participants have cognitive biases such as overconfidence, overreaction, representative bias, information bias, and various other predictable human errors in reasoning and information processing (Friesen & Weller, 2006).

Behavioral economics literature indicates the existence of a relationship between price-movements in a market and the fundamental information released in the news. Research confirms that investors' aggregate behavioral reactions to such information can drive prices up or down. Behavioural finance and investor sentiment theory have firmly established that investors' behaviour can be shaped by whether they feel optimistic (bullish) or pessimistic (bearish) about future market values (J. Bollen & Huina, 2011). However, it is yet to be determined how investors react to what information and in what direction it may drive a market. However, multiple studies indicate that a predictive link exists. This theoretical basis constitutes the foundation of this investigation.

2.10.3 Market Prediction Avenues: Fundamental vs. Technical Analysis

Those who attempt to predict a market are segmented into two camps. Those who believe that historic market movements are bound to repeat themselves are known as technical analysts. They simply believe there are visual patterns in a market graph that an experienced eye can detect. Based on this belief many of the graph movements are named which forms the basis of 'technical analysis'. At a higher level technical analysts try to detect such subtle mathematical models by the use of computation power and pattern recognition techniques. Although, technical analysis techniques are most wide spread among many of the market brokers and participants, to a scientific mind with a holistic point of view, technical analysis alone cannot seem very attractive; specially because most technical analysts do not back any of their observations up with anything more than stating that patterns exist. They do very little if at all to find out the reason behind existence of patterns. Some of the common techniques in technical analysis are the moving average rules, relative strength rules, filter rules and the trading range breakout rules. In a recent study the effectiveness and limitations of these rules were put to test again and it was demonstrated that in many cases and contexts these rules are not of much predictive power (H. Yu et al., 2013). Nevertheless, there are and continue to be many sophisticated financial prediction modelling efforts based on various types or combinations of machine learning algorithms like neural networks (Anastasakis & Mort, 2009; Ghazali, Hussain, & Liatsis, 2011; Sermpinis, Laws, Karathanasopoulos, & Dunis, 2012; Vanstone & Finnie, 2010), fuzzy logic (Bahrepour, Akbarzadeh-T., Yaghoobi, & Naghibi-S., 2011), Support Vector regression (S.-C. Huang, Chuang, Wu, & Lai, 2010; Premanode & Toumazou, 2013), rule-based genetic network programming (Mabu, Hirasawa, Obayashi, & Kuremoto, 2013).

However, there is a second school of thought known as 'fundamental analysis', which seems to be more promising. In fundamental analysis analysts look at fundamental data

that is available to them from different sources and make assumptions based on that. We can come up with at least 5 main sources of fundamental data: 1- the financial data of a company like data in its balance sheet or financial data about a currency in the FOREX market, 2- Financial data about a market like its index, 3- Financial data about the government activities and banks, 4- Political circumstances, 5- Geographical and meteorological circumstances like natural or unnatural disasters.

Most market participants try to keep an eye on both technical and fundamental data. However, determining underlying fundamental value of any asset may be challenging and with a lot of uncertainty (Kaltwasser, 2010). Therefore, the automation of fundamental analysis is rather rare. Fasanghari and Montazer (2010) design a fuzzy expert system for stock exchange portfolio recommendation that takes some of the company fundamentals through numeric metrics as input.

Moreover, fundamental data is usually of an unstructured nature and it remains to be a challenge to make the best use of it efficiently through computing. The research challenge here is to deal with this unstructured data. One recent approach that is emerging in order to facilitate such topical unstructured data and extract structured data from it is the development of specialized search engines like this financial news semantic search engine (Lupiani-Ruiz et al., 2011). However, it remains to be a challenge to extract meaning in a reliable fashion from text and a search engine like the above is limited to extracting the available numeric data in the relevant texts. One recent study shows the impact of US news, UK news and Dutch news on three Dutch banks during the financial crisis of 2007-2009 (Kleinnijenhuis, 2013). This specific study goes on to explore market panics from a financial journalism perspective and communication theories specifically in an age of algorithmic and frequency trading.

2.11 Market-Predictive Text-Mining Works

Despite the existence of multiple systems in this area of research we have not found any dedicated and comprehensive comparative analysis and review of the available systems. Nikfarjam, Emadzadeh, and Muthaiyah (2010) have made an effort in form of a conference paper that provides a rough summary under the title of "Text mining approaches for stock market prediction". Hagenau et al. (2013) have also presented a basic comparative table in their literature review that has been referred to in some parts of this work. In this section we are filling this gap by reviewing the major systems that have been developed around the past decade.

2.11.1 Generic Overview

All of these systems have some of the components depicted in Figure 2.4. At one end text is fed as input to the system and at the other end some market predictive values are generated as output.

In the next sections a closer look is taken at each of the depicted components, their roles and theoretical foundation.

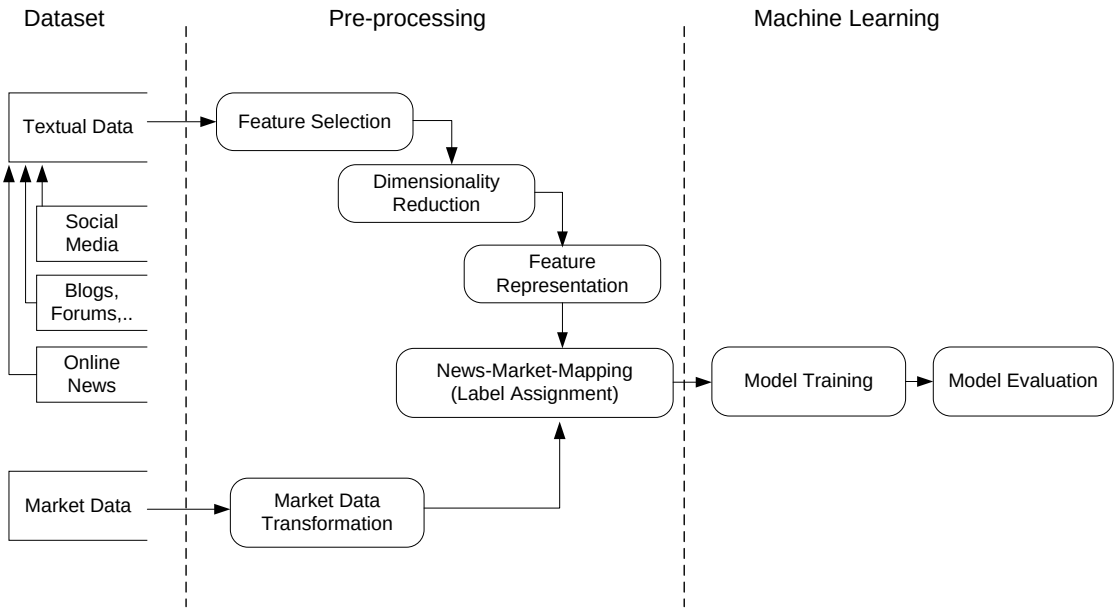

Figure 2.4 System Components Diagram

2.11.2 Input Dataset

All systems are taking at least two sources of data as input, namely, the textual data from the online resources and the market data.

2.11.3 Textual Data

The textual input can have several sources and content types accordingly as shown in table 2.1. The majority of the sources used are major news websites like The Wall Street Journal (Werner & Myrray Z., 2004), Financial Times (Wuthrich et al., 1998), Reuters (Pui Cheong Fung, Xu Yu, & Wai, 2003), Dow Jones, Bloomberg (Chatrath et al., 2014; Jin et al., 2013), Forbes (Rachlin, Last, Alberg, & Kandel, 2007) as well as Yahoo! Finance (Schumaker et al., 2012). The type of the news is either general news or special financial news. The majority of the systems are using financial news as it is deemed to have less noise compared with general news. What is being extracted here is the news text or the news headline. News headlines are occasionally used and are argued to be more straight-to-the-point and, hence, of less noise caused by verbose text (C.-J. Huang et al., 2010; Peramunetilleke & Wong, 2002). Fewer researchers have looked at less formal sources of textual information e.g. Das and Chen (2007) have looked at the text on online message boards in their work. And more recently Y. Yu et al. (2013) have looked at the textual content from the social media like twitter and blog posts. Some researchers focus solely on twitter and utilize it for market prediction and public mood analysis more efficiently (J. Bollen & Huina, 2011; Vu, Chang, Ha, & Collier, 2012). A third class of textual source for the systems has been the company annual reports, press releases and corporate disclosures. We observe that a difference about this class of information about companies is the nature of their timing, whereby regular reports and disclosures have prescheduled times. Pre-set timing is suspected to have an impact on prediction capability or nature which may have been caused by anticipatory reactions among market participants (Chatrath et al., 2014; C.-J. Huang et

al., 2010). Therefore, we have included a column for this information in the table 2.1. It is also important to remember that some textual reports can have structured or semi-structured formats like macroeconomic announcement that come from governments or central banks on the unemployment rates or the Gross Domestic Product (GDP). Chatrath et al. (2014) have used such structured data to predict jumps in the foreign exchange market (FOREX). Furthermore, with regards to information release frequency, one interesting observation that has been made in the recent research is that increase of the news release frequency can cause a decrease in informed-trading and hence the degree of information asymmetry is lower for firms with more frequent news releases (Sankaraguruswamy, Shen, & Yamada, 2013). This raises an interesting point whereby the uninformed traders increase and thereby play a significant role in the market with too frequent news releases. Although it may feel counter-intuitive as one may expect more informed trading to occur under such circumstances.

Table 2.1 Comparison of the textual input for different systems

Reference	Text Type	Text Source	No. of items	Pre-scheduled	Un-structured
Wuthrich et al. (1998)	General news	The Wall Street Journal, Financial Times, Reuters, Dow Jones , Bloomberg	Not given	No	Yes
Peramunetilleke and Wong (2002)	Financial news	HFDF93 via www.olsen.ch	40 headlines per hour	No	Yes
Pui Cheong Fung et al. (2003)	Company news	Reuters Market 3000 Extra	600,000	No	Yes
Werner and Myrray Z. (2004)	Message postings	Yahoo! Finance, Raging Bull, Wall Street Journal	1.5 million messages	No	Yes
Mittermayer (2004)	Financial news	Not mentioned	6,602	No	Yes
Das and Chen (2007)	Message postings	Message boards	145,110 messages	No	Yes
Soni, van Eck, and Kaymak (2007)	Financial news	FT Intelligence (Financial Times online service)	3493	No	Yes
Zhai, Hsu, and Halgamuge (2007)	Market-sector news	Australian Financial Review	148 direct company news and 68 indirect ones	No	Yes
Rachlin et al. (2007)	Financial news	Forbes.com, today.reuters.com	Not mentioned	No	Yes

Paul C. Tetlock, Saar-Tsechansky, and Macskassy (2008)	Financial news	Wall Street Journal, Dow Jones News Service from Factiva news database.	350,000 stories	No	Yes
Mahajan, Dey, and Haque (2008)	Financial news	Not mentioned	700 news articles	No	Yes
Butler and Kešelj (2009)	Annual reports	Company websites	Not mentioned	Yes	Yes
Schumaker and Chen (2009)	Financial news	Yahoo Finance	2800	No	Yes
F. Li (2010)	Corporate filings	Management's Discussion and Analysis section of 10-K and 10-Q filings from SEC Edgar Web site	13 million forward-looking-statements in 140,000 10-Q and K filings	Yes (annual report)	Yes
C.-J. Huang et al. (2010)	Financial news	Leading electronic newspapers in Taiwan	12,830 headlines	No	Yes
Groth and Muntermann (2011)	Adhoc announcements	Corporate disclosures	423 disclosures	No	Yes
Schumaker et al. (2012)	Financial news	Yahoo! Finance	2802	No	Yes
Lugmayr and Gossen (2012)	Broker newsletters	Brokers	Not available	No	Yes
Y. Yu et al. (2013)	Daily conventional and social media	Blogs, forums, news and micro blogs (e.g., Twitter)	52,746 messages	No	Yes
Hagenau et al. (2013)	Corporate announcements & financial news	DGAP, EuroAdhoc	10870 & 3478 respectively	No	Yes
Jin et al. (2013)	General news	Bloomberg	361,782	No	Yes
Chatrath et al. (2014)	Macroeconomic news	Bloomberg	Not mentioned	Yes	No
J. Bollen and Huina (2011)	Tweets	Twitter	9,853,498	No	Yes
Vu et al. (2012)	Tweets	Twitter	5,001,460	No	Yes

2.11.4 Market Data

The other source of input data for the systems comes from the numeric values in financial markets in form of price-points or indexes. This data is used mostly for purpose of training the machine learning algorithms and occasionally it is used for prediction purposes whereby it is fed into the machine learning algorithm as an independent variable for a feature, this topic will be discussed in a later section. In table 2.2, crucial details of such market data are provided. Firstly, there is a differentiation

made between the stock markets and the foreign exchange market (FOREX). Past research has been mostly focused on stock market prediction, either in form of a stock market index like the Dow Jones Industrial Average (J. Bollen & Huina, 2011; Werner & Myrray Z., 2004; Wuthrich et al., 1998), the US NASDAQ Index (Rachlin et al., 2007), Morgan Stanley High-Tech Index (MSH) (Das & Chen, 2007), the Indian Sensex Index (Mahajan et al., 2008), S&P 500 (Schumaker & Chen, 2009) or the stock price of a specific company like BHP Billiton Ltd. (BHP.AX) (Zhai et al., 2007) or like Apple, Google, Microsoft and Amazon in Vu et al. (2012) or a group of companies (Hagenau et al., 2013). The FOREX market has been only occasionally addressed in about ten percent of the reviewed works; like in the work of Peramunetilleke and Wong (2002) and more recently in the works of Chatrath et al. (2014) and Jin et al. (2013). It is worth considering that the efficiency levels of the FOREX markets around the world vary (G.-J. Wang, Xie, & Han, 2012), and hence, it should be possible to find less efficient currency pairs that are prone to predictability.

Moreover, almost all the forecast types on any of the market measures above are categorical with discrete values like Up, Down and Steady. There are very few pieces of research that have explored an approach based on linear regression (Jin et al., 2013; Schumaker et al., 2012; Paul C. Tetlock et al., 2008).

Furthermore, the predictive timeframe for each work is compared. The timeframe from the point of news release to a market impact observation can vary from seconds to days, weeks or months. The second or millisecond impact prediction is the element that fuels an entire industry by the name of micro-trading in which special equipment and vicinity to the news source as well as the market computer systems is critical; a work on trading trends has well explained such quantitative trading efforts (Chordia, Roll, & Subrahmanyam, 2011). Another name for the same concept is High Frequency Trading which is explored in detail by Chordia, Goyal, Lehmann, and Saar (2013). Another

similar term is Low-Latency trading which is amplified in the work of Hasbrouck and Saar (2013). However, past research has indicated sixty minutes to be looked at as a reasonable market convergence time to efficiency (Chordia, Roll, & Subrahmanyam, 2005). Market convergence refers to the period during which a market reacts to the information that is made available and becomes efficient by reflecting it fully. Information availability and distribution channels are critical here and the research on the market efficiency convergence time is ongoing (Reboredo, Rivera-Castro, Miranda, & García-Rubio, 2013). Most of the works compared in table 2.2 have a daily timeframes followed by intraday timeframes which are in the range of 5 minutes (Chatrath et al., 2014), 15 minutes (Werner & Myrray Z., 2004), 20 minutes (Schumaker & Chen, 2009) to 1, 2, or 3 hours (Peramunetilleke & Wong, 2002). Experiment periods are also contrasted with shortest being only 5 days in case of the work of Peramunetilleke and Wong (2002) and up to multiple years with the longest at 24 years from 1980 to 2004 by P. C. Tetlock (2007) followed by 14 years from 1997 to 2011 in the work of Hagenau et al. (2013) , 13 years from 1994 to 2007 in the work of F. Li (2010) with the latter looking at an annual timeframe and the formers at daily timeframes. The remaining majority of the works take on an experiment period with a length of multiple months as detailed in table 2.2.

Table 2.2 The input market data, experiment timeframe, length and forecast type

Reference	Market	Market Index	Time-frame	Period	Forecast type
Wuthrich et al. (1998)	Stocks	Dow Jones Industrial Average, the Nlkkei 225, the Financial Times 100, the Hang Seng, and the Singapore Straits	Daily	6 Dec 1997 to 6 Mar 1998	Categorical: Up, Steady, Down
Peramunetilleke and Wong (2002)	FOREX	Exchange rate(USD-DEM, USD-JYP)	Intraday(1, 2, or 3 hours)	22 to 27 Sept 1993	Categorical: Up, Steady, Down
Pui Cheong Fung et al. (2003)	Stocks	33 stocks from the Hang Seng	Daily (No delay & varying lags)	1 Oct 2002 to 30 Apr 2003	Categorical: Rise, Drop

Werner and Myrray Z. (2004)	Stocks	Dow Jones Industrial Average, and the Dow Jones Internet	Intraday(15-min, 1 hour and 1 day)	Year 2000	Categorical: Buy, Sell, and Hold messages
Mittermayer (2004)	Stocks	Stock prices	Daily	1 Jan to 31 Dec 2002	Categorical: good news, bad news , no movers
Das and Chen (2007)	Stocks	24 tech-sectors in the Morgan Stanley High-Tech	Daily	July and August 2001	Aggregate sentiment index
Soni et al. (2007)	Stocks	11 oil and gas companies	Daily	1 Jan 1995 to 15 May 2006	Categorical: Positive, Negative
Zhai et al. (2007)	Stocks	BHP Billiton Ltd. From Australian Stock Exchange	Daily	1 Mar 2005 to 31 May 2006	Categorical: Up, Down
Rachlin et al. (2007)	Stocks	5 stocks from US NASDAQ	Days	7 Feb to 7 May 2006	Categorical: Up, Slight-Up, Expected, Slight-Down, Down
Paul C. Tetlock et al. (2008)	Stocks	Individual S&P 500 firms and their future cash flows	Daily	1980 to 2004	Regression of a measure called neg.
Mahajan et al. (2008)	Stocks	Sensex	Daily	5 Aug to 8 Apr	Categorical
Butler and Kešelj (2009)	Stocks	1-Year market drift	Yearly	2003 to 2008	Categorical: Over- or under-perform S&P 500 index over the coming year
Schumaker and Chen (2009)	Stocks	S&P 500 stocks	Intraday (20 min)	26 Oct to 28 Nov 2005	Categorical: Discrete numeric
F. Li (2010)	Stocks	(1) Index (2) Quarterly earnings and cash flows (3) Stock returns	Annually (Quarterly with 3 dummy quarters)	1994 to 2007	Categorical based on tone: Positive, Negative, Neutral, Uncertain
C.-J. Huang et al. (2010)	Stocks	Taiwan Stock Exchange Financial Price	Daily	Jun to Nov 2005	Just significance degree assignment
Groth and Muntermann (2011)	Stocks	Abnormal risk exposure ARISKτ (Thomson Reuters DataScope Tick History)	Intraday (volatility during the τ=15 and 30 min)	1 Aug 2003 to 31 Jul 2005	Categorical: Positive, Negative
Schumaker et al. (2012)	Stocks	S&P 500	Intraday (20min)	26 Oct to 28 Nov 2005	Regression
Lugmayr and Gossen (2012)	Stocks	DAX 30 Performance Index	Intraday (3 or 4 times per day)	Not available	Categorical: Sentiment [-1, 1]; Trend (Bear, Bull, Neutral); Trend strength (in %)

Y. Yu et al. (2013)	Stocks	Abnormal returns and cumulative abnormal returns of 824 firms	Daily	1 Jul to 30 Sept 2011	Categorical: Positive or Negative
Hagenau et al. (2013)	Stocks	Company specific	Daily	1997 to 2011	Categorical: Positive or Negative
Jin et al. (2013)	FOREX	Exchange rate	Daily	1 Jan to 31 Dec 2012	Regression
Chatrath et al. (2014)	FOREX	Exchange rate	Intraday(5min)	Jan 2005 to Dec 2010	Categorical: Positive or Negative jumps
J. Bollen and Huina (2011)	Stock	DJIA	Daily	28 Feb to 19 Dec 2008	Regression Analysis
Vu et al. (2012)	Stock	Stock prices (at NASDAQ for AAPL, GOOG, MSFT, AMZN)	Daily	1 Apr 2011 to 31 May 2011, online test 8 Sep to 26 Sep 2012	Categorical: up and down

2.11.5 Pre-processing

Once the input data is available, it must be prepared so that it can be fed into a machine learning algorithm. This for the textual data means to transform the unstructured text into a representative format that is structured and can be processed by the machine. In data mining in general and text mining specifically the pre-processing phase is of significant impact on the overall outcomes (Uysal & Gunal, 2014). There are at least three sub-processes or aspects of pre-processing which we have contrasted in the reviewed works, namely: Feature-Selection, Dimensionality-Reduction, and Feature-Representation.

2.11.5.1 Feature-Selection

The decision on features through which a piece of text is to be represented is crucial because from an incorrect representational input nothing more than a meaningless output can be expected. In table 2.3 the type of feature selection from text for each of the works is listed.

In most of the literature the most basic techniques have been used when dealing with text-mining-based market-prediction-problems as alluded to by Kleinnijenhuis (2013) as well consistent with our findings. The most common technique is the so called “bag-of-words” which is essentially breaking the text up into its words and considering each of the as a feature. As presented in table 2.3, around three quarters of the works are relying on this basic feature-selection technique in which the order and co-occurrence of words are completely ignored. Schumaker et al. (2012) and Schumaker and Chen (2009) have explored two further techniques namely noun-phrases and named entities. In the former, first the words with a noun part-of-speech are identified with the help of a lexicon and then using syntactic rules on the surrounding parts of speech, noun-phrases are detected. In the latter, a category system is added in which the nouns or the noun phrases are organized. They used the so called MUC-7 framework of entity classification, where categories include date, location, money, organization, percentage, person and time. However, they did not have any success in improving their noun-phrase results via this further categorization in their named-entities technique. On the other hand, Vu et al. (2012) successfully improve results by building a Named Entity Recognition (NER) system to identify whether a Tweet contains named entities related to their targeted companies based on a linear Conditional Random Fields (CRF) model. Another less frequently used but interesting technique is the so called Latent Dirichlet Allocation (LDA) technique used by Jin et al. (2013) as well as Mahajan et al. (2008) in table 2.3 to categorize words into concepts and use the representative concepts as the selected features. Some of the other works are using a technique called n-grams (Butler & Kešelj, 2009; Hagenau et al., 2013). An n-gram is a contiguous sequence of n items which are usually words from a given sequence of text. However, word sequence and syntactic structures could essentially be more advanced. An example for such setup could be the use of Syntactic N-grams (Sidorov, Velasquez, Stamatatos, Gelbukh, &

Chanona-Hernández, 2013). However, inclusion of such features may give rise to language-dependency problems which need to be dealt with (Kwanho Kim et al., 2014). Combination techniques of lexical and semantic features for short text classification may also be enhanced (Yang et al., 2013).

Feature selection is a standard step in the pre-processing phase of data mining and there are many other approaches that can be considered for textual feature selection, genetic algorithms being one of them as described in detail in the work of Tsai et al. (2013). In another work Ant Colony Optimization has been successfully used for textual feature-selection (Aghdam, Ghasem-Aghaee, & Basiri, 2009). Feature selection for text classification can also be done based on a filter-based probabilistic approach (Uysal & Gunal, 2012). Feng, Guo, Jing, and Hao (2012) propose a generative probabilistic model, describing categories by distributions, handling the feature selection problem by introducing a binary exclusion/inclusion latent vector, which is updated via an efficient Metropolis search. Delicate pre-processing has been shown to have significant impact in similar text mining problems (Haddi et al., 2013; Uysal & Gunal, 2014).

2.11.5.2 Dimensionality-Reduction

Having a limited number of features is extremely important as the increase in the number of features which can easily happen in feature-selection in text can make the classification or clustering problem extremely hard to solve by decreasing the efficiency of most of the learning algorithms, this situation is widely known as the curse of dimensionality (Pestov, 2013). In table 2.3 under Dimensionality-Reduction the approach taken by each of the works is pointed out. Zhai et al. (2007) does this by choosing the top 30 concepts with the highest weights as the features instead of all available concepts. Mittermayer (2004) does this by filtering for the top 1000 terms out of all terms. The most common approach however is setting a minimum occurrence limit and reducing the terms by selecting the ones reaching a number of occurrences

(Butler & Kešelj, 2009; Schumaker & Chen, 2009). Next common approach is using a predefined dictionary of some sort to replace them with a category name or value. Some of these dictionaries are specially put together by a market expert like the one used by Wuthrich et al. (1998) or Peramunetilleke and Wong (2002) or they are more specific to a specific field like psychology in the case of Harvard-IV-4 which has been used in the work of Paul C. Tetlock et al. (2008). And other times they are rather general use dictionaries like the WordNet Thesaurus used by Zhai et al. (2007). At times a dictionary or thesaurus is created dynamically based on the text corpus using a term extraction tool (Soni et al., 2007). Another set of activities that usually constitute the minimum of dimensionality-reduction are: features stemming, conversion to lower case letters, punctuation removal and removal of numbers, web page addresses and stop-words. These steps are almost always taken and in some works are the only steps taken as in the work by Pui Cheong Fung et al. (2003).

Feature-reduction can be enhanced in a number of ways but the current research has yet not delved into it much as observed in the reviewed works. Berka and Vajteršic (2013) introduce a detailed method for dimensionality reduction for text, based on parallel rare term vector replacement.

2.11.5.3 Feature-Representation

After the minimum number of features is determined, each feature needs to be represented by a numeric value so that it can be processed by machine learning algorithms. Hence, the title “Feature-Representation” in table 2.3 is used for the column whereby the type of numeric value that is associated with each feature is compared for all the reviewed works. This assigned numeric value acts like a score or a weight. There are at least 5 types for it that are very popular, namely, Information Gain (IG), Chi-square Statistics (CHI), Document Frequency (DF), Accuracy Balanced (Acc2) and Term Frequency-Inverse Document Frequency (TF-IDF). A comparison of these five

metrics along with proposals for some new metrics can be found in the work of Taşcı and Güngör (2013).

The most basic technique is a Boolean or a binary representation whereby two values like 0 and 1 represent the absence or presence of a feature e.g. a word in the case of a bag-of-words technique as in these works (Mahajan et al., 2008; Schumaker et al., 2012; Wuthrich et al., 1998). The next most common technique is the Term Frequency–Inverse Document Frequency or TF-IDF (Groth & Muntermann, 2011; Hagenau et al., 2013; Peramunetilleke & Wong, 2002; Pui Cheong Fung et al., 2003). The TF-IDF value increases proportionally to the number of times a word appears in the document, but is offset by the frequency of the word in the corpus, to balance out the general popularity of some words. There are also other similar measures that are occasionally used like the Term Frequency-Category Discrimination (TF-CDF) metric that is derived from Category Frequency (CF) and proves to be more effective than TF-IDF in this work (Peramunetilleke & Wong, 2002).

Generally in text-mining enhanced feature-reduction (dimensionality-reduction) and feature weighting (feature-representation) can have significant impact on the eventual text-classification efficiency (Shi et al., 2011).

Table 2.3 Pre-processing: Feature-Selection, Feature-Reduction, Feature-Representation

Reference	Feature Selection	Dimensionality Reduction	Feature Representation
Wuthrich et al. (1998)	Bag-of-words	Pre-defined dictionaries (word-sequences by an expert)	Binary
Peramunetilleke and Wong (2002)	Bag-of-words	Set of keyword records	Boolean, TF-IDF, TF-CDF
Pui Cheong Fung et al. (2003)	Bag-of-words	Stemming, conversion to lower-case, removal of punctuation, numbers, web page addresses and stop-words	TF-IDF
Werner and Myrray Z. (2004)	Bag-of-words	Minimum information criterion (top 1,000 words)	Binary
Mittermayer (2004)	Bag-of-words	Selecting 1000 terms	TF-IDF
Das and Chen (2007)	Bag-of-words, Triplets	Pre-defined dictionaries	Different discrete values

			for each classifier
Soni et al. (2007)	Visualization	Thesaurus made using term extraction tool of N.J. van Eck	Visual coordinates
Zhai et al. (2007)	Bag-of-words	WordNet Thesaurus (stop-word removal, POS tagging, higher level concepts via WordNet). Top 30 concepts.	Binary, TF-IDF
Rachlin et al. (2007)	Bag-of-words, commonly used financial values	Most influential keywords list (Automatic extraction)	TF, Boolean, Extractor software output
Paul C. Tetlock et al. (2008)	Bag-of-words for negative words	Pre-defined dictionary. Harvard-IV-4 psychosocial dictionary.	Frequency divided by total number of words
Mahajan et al. (2008)	Latent Dirichlet Allocation (LDA)	Extraction of twenty-five topics	Binary
Butler and Kešelj (2009)	Character N-Grams, three readability scores, last year's performance	Minimum occurrence per document.	Frequency of the n-gram in one profile
Schumaker and Chen (2009)	Bag of words, noun phrases, named entities	Minimum occurrence per document	Binary
F. Li (2010)	Bag-of-words, tone and content	Pre-defined dictionaries	Binary, Dictionary value
C.-J. Huang et al. (2010)	Simultaneous terms, ordered pairs	Synonyms replacement	Weighted based on the rise/fall ratio of index
Groth and Muntermann (2011)	Bag-of-words	Feature scoring methods using both Information Gain and Chi-Squared metrics	TF-IDF
Schumaker et al. (2012)	OpinionFinder overall tone and polarity	Minimum occurrence per document	Binary
Lugmayr and Gossen (2012)	Bag-of-words	Stemming	Sentiment Value
Y. Yu et al. (2013)	Bag-of-words	Not mentioned	Binary
Hagenau et al. (2013)	Bag-of-words, Noun Phrases, Word-combinations, N-grams	Frequency for news, Chi^2-approach and bi-normal separation (BNS) for exogenous-feedback-based feature selection, Dictionary.	TF-IDF
Jin et al. (2013)	Latent Dirichlet Allocation (LDA)	Topic extraction, top topic identification by manually aligning news articles with currency fluctuations,	Each article's topic distribution
Chatrath et al. (2014)	Structured Data	Structured Data	Structured Data
J. Bollen and Huina (2011)	By OpinionFinder	By OpinionFinder	By OpinionFinder
Vu et al. (2012)	Daily aggregate number of positives or negatives on Twitter Sentiment Tool (TST) and an emoticon lexicon. Daily mean of	Pre-defined company related keywords, Named Entity Recognition based on linear Conditional Random Fields(CRF)	Real number for Daily Neg_Pos and Bullish_Bearish

	Pointwise Mutual Information (PMI) for pre-defined bullish-bearish anchor words		

2.11.6 Machine Learning

After the pre-processing is completed and text is transformed into a number of features with a numeric representation, machine learning algorithms can be engaged. In the following a brief summary of these algorithms is presented as well as a comparison on some of the other detail technicalities in the designs of the reviewed systems.

2.11.6.1 Machine Learning Algorithms

In this section, it is attempted to provide a summary of the machine learning algorithms used in the reviewed works. It is noted that comparing such algorithms is not easy and full of pitfalls (Salzberg, 1997). Therefore, the main objective is to report what is used; so that it helps understand what lacks and may be possible for future research. Almost all the machine learning algorithms that have been used are classification algorithms as listed in Table 2.4. Basically the systems are using the input data to learn to classify an output usually in terms of the movement of the market in classes such as UP, DOWN and STEADY. However, there is also a group of works which use regression analysis for making predictions and not classification.

Table 2.4, categorizes the reviewed works based on their used algorithm into 6 classes:

A) Support Vector Machine (SVM)

B) Regression Algorithms

C) Naïve Bayes

D) Decision Rules or Trees

E) Combinatory Algorithms

F) Multi-algorithm experiments

A) Support Vector Machine: This section in table 2.4 contains the class of algorithms that the vast majority of the reviewed works are using (Mittermayer, 2004; Pui Cheong Fung et al., 2003; Schumaker & Chen, 2009; Soni et al., 2007; Werner & Myrray Z., 2004). SVM is a non-probabilistic binary linear classifier used for supervised learning. The main idea of SVMs is finding a hyperplane that separates two classes with a maximum margin. The training problem in SVMs can be represented as a quadratic programming optimization problem. A common implementation of SVM that is used in many works (Mittermayer, 2004; Pui Cheong Fung et al., 2003) is SVM Light (T. Joachims, 1999). SVM Light is an implementation of an SVM learner which addresses the problem of large tasks (T. Joachims, 1999). The optimization algorithms used in SVM Light are described in T. Joachims (2002). Another commonly used implementation of SVM is LIBSVM (Chang & Lin, 2011). LIBSVM implements an SMO-type (Sequential Minimal Optimization) algorithm proposed in a paper by Fan, Chen, and Lin (2005). Soni et al. (2007) is using this implementation for prediction of stock price movements. Aside from the implementation, the input to the SVM is rather unique in the work of Soni et al. (2007). They take the coordinates of the news items in a visualized document-map as features. A document-map is a low-dimensional space in which each news item is positioned on the weighted average of the coordinates of the concepts that occur in the news item. SVMs can be extended to nonlinear classifiers by applying kernel mapping (kernel trick). As a result of applying the kernel mapping, the original classification problem is transformed into a higher dimensional space. SVMs which represent linear classifiers in this high-

dimensional space may correspond to nonlinear classifiers in the original feature space (Burges, 1998). The kernel function used may influence the performance of the classifier. Zhai et al. (2007) are using SVM with a Gaussian RBF kernel and a polynomial kernel.

B) Regression algorithms: They take on different forms in the research efforts as listed in table 2.4. One approach is *Support Vector Regression (SVR)* which is a regression based variation of SVM (Drucker, Burges, Kaufman, Smola, & Vapnik, 1997). It is utilized by Hagenau et al. (2013) and Schumaker et al. (2012).

Hagenau et al. (2013) use SVM primarily but they also assess the ability to predict the discrete value of the stock return using SVR. They predict returns and calculate the R2 (squared correlation coefficient) between predicted and actually observed return. The optimization behind the SVR is very similar to the SVM, but instead of a binary measure (i.e., positive or negative), it is trained on actually observed returns. While a binary measure can only be 'true' or 'false', this measure gives more weight to greater deviations between actual and predicted returns than to smaller ones. As profits or losses are higher with greater deviations, this measure better captures actual trading returns to be realized (Hagenau et al., 2013).

Schumaker et al. (2012) choose to implement the SVR Sequential Minimal Optimization (Platt, 1999) function through Weka (Witten & Frank, 2005). This function allows discrete numeric prediction instead of classification. They select a linear kernel and ten-fold cross-validation.

Sometimes linear regression models are directly used (Chatrath et al., 2014; Jin et al., 2013; Paul C. Tetlock et al., 2008). Paul C. Tetlock et al. (2008) use OLS

(Ordinary Least Square) method for estimating the unknown parameters in the linear regression model. They use two different dependent variables (raw and abnormal next-day returns) regressed on different negative words measures. Their main result is that negative words in firm-specific news stories robustly predict slightly lower returns on the following trading day.

Chatrath et al. (2014) use a stepwise multivariate regression in a Probit (Probability unit) model. The purpose of the model is to estimate the probability that an observation with particular characteristics will fall into a specific category. In this case it is to ascertain the probability of news releases that result in jumps. Jin et al. (2013) apply topic clustering methods and use customized sentiment dictionaries to uncover sentiment trends by analyzing relevant sentences. A linear regression model estimates the weight for each topic and makes currency forecasts.

C) Naïve Bayes: It is the next algorithm used in a group of works in table 2.4. It is probably the oldest classification algorithm (Lewis, 1998). But it is still very popular and is used among many of the works (Groth & Muntermann, 2011; F. Li, 2010; Werner & Myrray Z., 2004; Wuthrich et al., 1998). It is based on the Bayes Theorem and it is called naïve because it is based on the naïve assumption of complete independence between text features. It differentiates itself from approaches such as k-Nearest Neighbors (k-NN), Artificial Neural Networks (ANN), or Support Vector Machine (SVM) in that it builds upon probabilities (of a feature belonging to a certain category) whereas the other mentioned approaches interpret the document feature-matrix spatially.

Y. Yu et al. (2013) apply the Naïve Bayes (NB) algorithm to conduct sentiment analysis to examine the effect of multiple sources of social media along with the

effect of conventional media and to investigate their relative importance and their interrelatedness. F. Li (2010) uses Naïve Bayes to examine the information content of the forward-looking statements in the Management Discussion and Analysis section of company filings. He uses the Niave Bayes module in the Perl programming language to conduct the computation.

D) Decision rules and trees: It is the next group of algorithms used in the literature as indicated in table 2.4. A few of the researchers have made an effort to create rule-based classification systems (C.-J. Huang et al., 2010; Peramunetilleke & Wong, 2002; Vu et al., 2012).

Peramunetilleke and Wong (2002) use a set of keywords provided by a financial domain expert . The classifier expressing the correlation between the keywords and one of the outcomes is a rule set. Each of the three rule sets (DOLLAR_UP, DOLLAR_STEADY, DOLLAR_DOWN) yields a probability saying how likely the respective event will occur in relation to available keywords.

C.-J. Huang et al. (2010) observe that the combination of two or more keywords in a financial news headline might play a crucial role on the next trading day. They thus applied weighted association rules algorithm to detect the important compound terms in the news headlines.

Rachlin et al. (2007) use a Decision Tree Induction algorithm, which doesn't assume attribute independence. The algorithm is C4.5 developed by Quinlan (Quinlan, 1993). This algorithm yields a set of trend predicting rules. They further show the effect of the combination between numerical and textual data. Vu et al. (2012) also use C4.5 decision tree for the text binary classification problem for predicting the daily up and down changes in stock prices.

For decision rule categorizers, the rules are composed of words, and words have meaning, the rules themselves can be insightful. More than just attempting to assign a label, a set of decision rules may summarize how to make decisions. For example, the rules may suggest a pattern of words found in newswires prior to the rise of a stock price. The downside of rules is that they can be less predictive if the underlying concept is complex (Weiss et al., 2010). Although decision rules can be particularly satisfying solutions for text mining, the procedures for finding them are more complicated than other methods (Weiss et al., 2010).

Decision trees are special decision rules that are organized into a tree structure. A decision tree divides the document space into non-overlapping regions at its leaves, and predictions are made at each leaf (Weiss et al., 2010).

E) Combinatory algorithms: In table 2.4, it is referring to a class of algorithms which are composed of a number of machine learning algorithms stacked or grouped together. Das and Chen (2007) have combined multiple classification algorithms together by a voting system to extract investor sentiment. The algorithms are namely, Naive Classifier, Vector Distance Classifier, Discriminant-Based Classifier, Adjective-Adverb Phrase Classifier, Bayesian Classifier. Accuracy levels turn out to be similar to widely used Bayes classifiers, but false positives are lower and sentiment accuracy higher.

Mahajan et al. (2008) identify and characterize major events that impact the market using a Latent Dirichlet Allocation (LDA) based topic extraction mechanism. Then a stacked classifier is used which is a trainable classifier that combines the predictions of multiple classifiers via a generalized voting procedure. The voting step is a separate classification problem. They use a decision tree based on information gain for handling numerical attributes in

conjunction with an SVM with sigmoid kernel to design the stacked classifier. The average accuracy of the classification system is 60%.

Butler and Kešelj (2009) propose 2 methods and then combine them to achieve best performance. The first method is based on character n-gram profiles, which are generated for each company annual report, and then analyzed based on the Common N-Gram (CNG) classification. The second method combines readability scores with performance inputs and then supplies them to a support vector machine (SVM) for classification. The combined version is setup to only make decisions when the models agreed.

J. Bollen and Huina (2011) deploy a self-organizing fuzzy neural network (SOFNN) model to test the hypothesis that including public mood measurements can improve the accuracy of Dow Jones Industrial Average (DJIA) prediction models. A fuzzy neural network is a learning machine that finds the parameters of a fuzzy system (i.e., fuzzy sets, fuzzy rules) by exploiting approximation techniques from neural networks. Therefore it is classified as a combinatory algorithm in table 2.4.

F) Multi-algorithm experiments: It is another class of works in table 2.4, whereby the same experiments are conducted using a number of different algorithms.

Wuthrich et al. (1998) is one of the earliest works of research in this area. They do not stack multiple algorithms together to form a bigger algorithm. However, they conduct their experiments using multiple algorithms and compare the results.

Werner and Myrray Z. (2004) also carry out all tests using two algorithms, Naïve Bayes and SVM. Furthermore, Groth and Muntermann (2011) employ Naïve Bayes, k-Nearest Neighbors (k-NN), Artificial Neural Networks (ANN),

and SVM in order to detect patterns in the textual data that could explain increased risk exposure in stock markets.

In conclusion, SVM has been extensively and successfully used as a textual classification and sentiment learning approach while some other approaches like Artificial Neural Networks (ANN), K-nearest neighbors (k-NN) have rarely been considered in the text mining literature for market prediction. This is also confirmed by Moraes, Valiati, and Gavião Neto (2013). Their research presents an empirical comparison between SVM and ANN regarding document-level sentiment analysis. Their results show that ANN can produce superior or at least comparable results to SVM's. Such results can provide grounds for looking into usage of other algorithms than the currently mostly used SVM. C. H. Li et al. (2012) demonstrate high performance of k-NN for text categorization as well as Jiang, Pang, Wu, and Kuang (2012). Tan, Wang, and Wu (2011) claim that for document categorization, centroid classifier performs slightly better than SVM classifier and beats it in running time too. Gradojevic and Gençay (2013) present a rare piece of research that uses fuzzy logic to improve technical trading signals successfully on the EUR-USD exchange rates but fuzzy logic is rarely used in the reviewed works for marker prediction based on text mining. Only J. Bollen and Huina (2011) use it in combination with neural networks to devise a self-organizing fuzzy neural network (SOFNN) successfully. However, Loia and Senatore (2014) show that it can be very useful for emotion modelling in general. Exploration of such under-researched algorithms in the context of market-prediction may lead to new insights that may be of interest for future researchers.

In order to better comprehend the reviewed systems in which the machine learning algorithms have been used, an additional number of system properties have been reviewed in table 2.4 which are explained in the following sections.

2.11.7 Training vs. testing volume and sampling

In this column, in Table 2.4, two aspects are summarized if the information were available. Firstly, the volume of the examples which were used for training versus the volume used for testing; around 70 or 80 percent for training vs. 30 or 20 percent for testing seems to be the norm. Secondly, if the sampling for training and testing was of a special kind; what is specially of interest here is to know if a linear sampling have been followed as in essence the samples are on a time-series. Some of the works have clearly mentioned the sampling type like Stratified (Groth & Muntermann, 2011) whereas most of the others surprisingly have not mentioned anything at all.

2.11.8 Sliding Window

The overall objective of the reviewed systems is to predict the market movement in a future time-window (prediction-window) based on the learning gained in a past time-window (training-window) where the machine learning algorithms are trained and patterns are recognized.

For example, a system may learn patterns based on the available data for multiple days (training-window) in order to predict the market movement on a new day (prediction-window). The length and location of the training-window on the timeline may have two possible formats: *fixed* or *sliding*. If the training window is fixed, the system learns based on the data available from point '*A'* to point '*B'* on the timeline and those 2 points are fixed; for instance, from date '*A'* to date *'B'*. In such a scenario the resulted learning from the training window is applied to the prediction-window regardless of where on the timeline the prediction-window is located. It may be right after the training-window or it may be farther into the future with a distance from the training-window. It is obvious that if there is a big distance between the training-window and the prediction-window the learning captured in the machine learning algorithm may not be up-to-date

and therefore accurate enough because the information available in the gap is not used for training.

Hence, a second format is introduced to solve the above problem whereby the entire training-window or one side of it (the side at the end) is capable of dynamically sliding up to the point where the prediction-window starts. In other words, if the training window starts at point 'A' and ends at point 'B' and the prediction-window starts at point 'C' and ends at point 'D'. In the sliding-window format, the system always ensures that point 'B' is always right before and adjacent to point 'C'. This approach is simply referred to in this work as "sliding window". The reviewed works which do possess a *sliding window* as a property of their system design are identified and marked in table 2.4 under a column with the same name.

Although, it intuitively seems necessary to implement a sliding-window, there are very few of the reviewed works which actually have (Butler & Kešelj, 2009; Jin et al., 2013; Peramunetilleke & Wong, 2002; Paul C. Tetlock et al., 2008; Wuthrich et al., 1998). This seems to be an aspect that can receive more attention in the future systems.

2.11.9 Semantics and Syntax

In Natural Language Processing (NLP) two aspects of language are attentively researched: Semantics and Syntax. Simply put: *semantics* deals with the meaning of words and *syntax* deals with their order and relative positioning or grouping. In this section: Firstly, a closer look is cast at the significance of each of them and some of the recent related works of research. Secondly, it is reported if and how they have been observed in the reviewed text-mining works for market-prediction.

Tackling semantics is an important issue and research efforts are occupied with it in a number of fronts. It is important to develop specialized ontologies for specific contexts like finance; Lupiani-Ruiz et al. (2011) present a financial news semantic search engine

based on Semantic Web technologies. The search engine is accompanied by an ontology population tool that assists in keeping the financial ontology up-to-date. Furthermore, semantics can be included in design of feature-weighting schemes; Luo, Chen, and Xiong (2011) propose a novel term weighting scheme by exploiting the semantics of categories and indexing terms. Specifically, the semantics of categories are represented by senses of terms appearing in the category labels as well as the interpretation of them by WordNet (Miller, 1995). Also, in their work the weight of a term is correlated to its semantic similarity with a category. WordNet (Miller, 1995) provides a semantic network that links the senses of words to each other. The main types of relations among WordNet synsets are the super-subordinate relations that are hyperonymy and hyponymy. Other relations are the meronymy and the holonymy (Loia & Senatore, 2014). It is critical to facilitate semantic relations of terms for getting a satisfactory result in the text categorization; C. H. Li et al. (2012) show a high performance of text categorization in which semantic relations of terms drawing upon two kinds of thesauri, a corpus-based thesaurus (CBT) and WordNet (WN), were sought. When a combination of CBT and WN was used, they obtained the highest level of performance in the text categorization.

Syntax is also very important and proper observation and utilization of it along with semantics (or sometimes instead of it) can improve textual classification accuracy; Kwanho Kim et al. (2014) propose a novel kernel, called language independent semantic (LIS) kernel, which is able to effectively compute the similarity between short-text documents without using grammatical tags and lexical databases. From the experiment results on English and Korean datasets, it is shown that the LIS kernel has better performance than several existing kernels. This is essentially a syntax-based pattern extraction method. It is interesting to note that there are several approaches to such syntax-based pattern-recognition methods: In the *word occurrence method*, a

pattern is considered as a word that appears in a document (Thorsten Joachims, 1998). In the *word sequence method*, it consists of a set of consecutive words that appear in a document (Lodhi, Saunders, Shawe-Taylor, Cristianini, & Watkins, 2002). In the *parse-tree method* which is based on syntactic structure, a pattern is extracted from the tree of a document by considering not only the word occurrence but also the word sequence in the document (Collins & Duffy, 2001).

Duric and Song (2012) propose a set of new feature-selection schemes that use a Content and Syntax model to automatically learn a set of features in a review document by separating the entities that are being reviewed from the subjective expressions that describe those entities in terms of polarities. The results obtained from using these features in a maximum entropy classifier are competitive with the state-of-the-art machine learning approaches (Duric & Song, 2012). Topic models such as Latent Dirichlet Allocation (LDA) are generative models that allow documents to be explained by unobserved (latent) topics. The Hidden Markov Model LDA (HMM-LDA) (Griffiths, Steyvers, Blei, & Tenenbaum, 2005) is a topic model that simultaneously models topics and syntactic structures in a collection of documents. The idea behind the model is that a typical word can play different roles. It can either be part of the content and serve in a semantic (topical) purpose or it can be used as part of the grammatical (syntactic) structure. It can also be used in both contexts (Duric & Song, 2012). HMM-LDA models this behaviour by inducing syntactic classes for each word based on how they appear together in a sentence using a Hidden Markov Model. Each word gets assigned to a syntactic class, but one class is reserved for the semantic words. Words in this class behave as they would in a regular LDA topic model, participating in different topics and having certain probabilities of appearing in a document (Duric & Song, 2012).

In table 2.4, there is one column dedicated to each of these aspects, namely: Semantics and Syntax. About half of the systems are utilizing some semantic aspect into their text mining approach which is usually done by using a dictionary or thesaurus and categorizing the words based on their meaning but none is advanced in the directions pointed out above. Moreover, very few works have made an effort to include Syntax i.e. order and role of words. These basic and somewhat indirect approaches are noun-phrases (Schumaker & Chen, 2009), word-combinations and n-grams (Hagenau et al., 2013) and simultaneous appearance of words (C.-J. Huang et al., 2010) and the "triplets" which consist of an adjective or adverb and the two words immediately following or preceding them (Das & Chen, 2007). Some works like Vu et al. (2012) include Part of Speech (POS) tagging as a form of attention to syntax. Loia and Senatore (2014) achieve phrase-level sentiment analysis by taking into account four syntactic categories, namely: nouns, verbs, adverbs and adjectives. The need of deep syntactic analysis for the phrase-level sentiment-analysis has been investigated by Kanayama and Nasukawa (2008).

2.11.10 Combining news and technical data or signals

It is possible to pass technical data or signals along with the text features into the classification algorithm as additional independent variables. Examples for technical data could be a price or index level at a given time. Technical signals are the outputs of technical algorithms or rules like the moving average, relative strength rules, filter rules and the trading range breakout rules. Few of the researchers have taken advantage of these additional inputs as indicated in table 2.4 (Butler & Kešelj, 2009; Hagenau et al., 2013; Rachlin et al., 2007; Schumaker & Chen, 2009; Schumaker et al., 2012; Zhai et al., 2007).

2.11.11 Used software

Lastly it is interesting to observe what some of the common third-party applications are that are used for the implementation of the systems. In this column of table 2.4 the reader can see the names of the software pieces which were used by the reviewed works as a part of their pre-processing or machine learning. They are mostly dictionaries (F. Li, 2010; Paul C. Tetlock et al., 2008), classification algorithm implementations (Butler & Kešelj, 2009; Werner & Myrray Z., 2004), concept extraction and word combination packages (Das & Chen, 2007; Rachlin et al., 2007) or sentiment value providers (Schumaker et al., 2012).

Table 2.4 Classification algorithms and other machine learning aspects

Reference	Algorithm Type	Algorithm Details	Training vs. testing volume and sampling	Sliding Window	Sema-ntics	Syntax	News & tech. data	Software
Pui Cheong Fung et al. (2003)	***SVM***	SVM-Light	First 6 consecutive months vs. the last month	No	No	No	No	Not mentioned
Mittermayer (2004)		SVM-Light	200 vs. 6,002 examples	No	No	No	No	NewsCATS
Soni et al. (2007)		SVM with standard linear kernel	80% vs. 20%	No	Yes	No	No	LibSVM package
Zhai et al. (2007)		SVM with Gaussian RBF kernel and polynomial kernel	First 12 months vs. the remaining two months	No	Yes	No	Yes	Not mentioned
Schumaker and Chen (2009)		SVM	Not mentioned	No	Yes	Yes	Yes	Arizona Text Extractor (AzTeK) & AZFin Text.
Lugmayr and Gossen (2012)		SVM	Not mentioned	No	Yes	No	Yes	SentiWordNet
Hagenau et al. (2013)	***Regression Algorithms***	SVM with a linear kernel, SVR	Not mentioned	No	Yes	Yes	Yes	Not mentioned
Schumaker et al. (2012)		SVR	Not mentioned	No	Yes	No	Yes	OpinionFinder
Jin et al. (2013)		Linear regression model	Previous day vs. a given day (2 weeks for regression)	Yes	Yes	No	No	Forex-foreteller, Loughran-McDonald financial dic., AFINN dic.
Chatrath et al. (2014)		Stepwise Multivariate Regression Model	Not applicable	No	No	No	No	Not mentioned

Paul C. Tetlock et al. (2008)		OLS Regression	30 and 3 trading days prior to an earnings announcement	Yes	Yes	No	No	Harvard-IV-4 psychosocial dictionary
Y. Yu et al. (2013)	***Naïve Bayes***	Naïve Bayes	Not mentioned	No	Yes	No	No	Open-source Natural Language Toolkit (NLTK)
F. Li (2010)		Naïve Bayes & dictionary-based	30,000 randomly vs. itself and the rest.	No	No	No	No	Diction, General Inquirer, the Linguistic Inquiry, Word Count (LIWC).
Peramunetilleke and Wong (2002)	***Decision Rules or Trees***	Rule classifier	22 Sept 12:00 to 27 Sept 9:00 vs. 9:00 to 10:00 on 27 Sept	Yes	Yes	No	No	Not mentioned
C.-J. Huang et al. (2010)		Weighted association rules	2005 Jun to 2005 Oct vs. 2005 Nov	No	Yes	Yes	No	Not mentioned
Rachlin et al. (2007)		C4.5 Decision Tree	Not mentioned.	No	No	No	Yes	Extractor Software package
Vu et al. (2012)		C4.5 Decision Tree	Trained by previous day features	Yes	Yes	Yes	No	CRF++ toolkit, Firehose, TST, CMU POS Tagger, AltaVista
Das and Chen (2007)	***Combinatory Algorithms***	Combination of different classifiers	1,000 vs. the rest	No	Yes	Yes	No	General Inquirer
Mahajan et al. (2008)		Stacked classifier	August 05 – Dec 07 vs. Jan 08 – Apr 08	No	Yes	No	No	Not mentioned
Butler and Kešelj (2009)		CNG distance measure & SVM & combined	year x vs. years x – 1 and x – 2. & all vector representations vs. particular testing year	Yes	No	No	Yes	Perl n-gram module Text::Ngrams developed by Keselj . LIBSVM
J. Bollen and Huina (2011)		self-organizing fuzzy neural network (SOFNN)	28 Feb to 28 Nov vs. 1 to 19 Dec 2008	No	N/A	N/A	No	GPOMS, Opinion-Finder
Wuthrich et al. (1998)	***Multi-algorithm experiments***	k-NN, ANNs, naïve Bayes, rule-based	Last 100 training days to forecast 1 day	Yes	Yes	No	No	Not mentioned
Werner and Myrray Z. (2004)		Naïve Bayes, SVM	1,000 messages vs. the rest	No	No	No	No	Rainbow package
Groth and Muntermann (2011)		Naïve Bayes, k-NN, ANN, SVM	Stratified cross validations	No	No	No	No	Not mentioned

2.11.12 Findings of the reviewed works

In Table 2.5, the evaluation mechanisms have been looked at as well as the new findings for each piece of research.

Most of the works are presenting a confusion matrix or parts thereof to present their results. And calculate accuracy, recall or precision and sometimes the F-measure, with accuracy being the most common. The accuracy in majority of the cases is reported in the range of 50 to 70 percent, while arguing for better than chance results which is estimated at 50 percent (Butler & Kešelj, 2009; F. Li, 2010; Mahajan et al., 2008; Schumaker & Chen, 2009; Schumaker et al., 2012; Zhai et al., 2007). It is a common evaluation approach and results above 55% have been considered report-worthy in other parts of the literature as well (Garcke, Gerstner, & Griebel, 2013). However, what makes most of the results questionable is that the majority of them surprisingly have not examined or reported if their experiment data is imbalanced or not. As this is important in data-mining (Duman, Ekinci, & Tanrıverdi, 2012; Thammasiri, Delen, Meesad, & Kasap, 2014) an additional column has been placed in table 2.5 to check for this. Among the reviewed works only Soni et al. (2007), Mittermayer (2004) and Peramunetilleke and Wong (2002) have paid some attention to this topic in their works. It is also crucial to note if an imbalanced dataset with imbalanced classes is encountered specially with a high dimensionality in the feature-space, devising a suitable feature-selection that can appropriately deal with both the imbalanced-data and the high dimensionality becomes critical. Feature-selection for high-dimensional imbalanced data is amplified in detail in the work of L. Yin, Ge, Xiao, Wang, and Quan (2013). Liu, Loh, and Sun (2009) tackle the problem of imbalanced textual data using a simple probability based term weighting scheme to better distinguish documents in minor categories. Smales (2012) examine the relationship between order imbalance and

macroeconomic news in the context of Australian interest rate futures market and identify nine major macroeconomic announcements with impact on order imbalance.

Another popular evaluation approach in addition to the above for half of the reviewed works is the assembly of a trading strategy or engine (Groth & Muntermann, 2011; Hagenau et al., 2013; C.-J. Huang et al., 2010; Mittermayer, 2004; Pui Cheong Fung et al., 2003; Rachlin et al., 2007; Schumaker & Chen, 2009; Schumaker et al., 2012; Paul C. Tetlock et al., 2008; Zhai et al., 2007). Through which a trading period is simulated and profits are measured to evaluate the viability of the system.

In general researchers are using evaluation mechanisms and experimental data that widely vary and this makes an objective comparison in terms of concrete levels of effectiveness unreachable.

Table 2.5 Findings of the reviewed works, existence of a trading strategy and balanced-data

Reference	Findings	Trading Strategy	Balanced data
Wuthrich et al. (1998)	Ftse 42%, Nky 47%, Dow 40% Hsi 53% and Sti 40%.	Yes	Not mentioned
Peramunetilleke and Wong (2002)	Better than chance	No	Yes
Pui Cheong Fung et al. (2003)	The cumulative profit of monitoring multiple time series is nearly double to that of monitoring single time series.	Yes	Not mentioned
Werner and Myrray Z. (2004)	Evidence that the stock messages help predict market volatility, but not stock returns.	No	No
Mittermayer (2004)	Average profit 11% compared to average profit by random trader 0%	Yes	Yes
Das and Chen (2007)	Regression has low explanatory power	No	Not mentioned
Soni et al. (2007)	Hit rate of classifier: 56.2% compared to 47.5% for naïve classifier and 49.1% SVM bag-of-words	No	Yes (roughly)
Zhai et al. (2007)	Price 58.8%, Direct news 62.5%, Indirect news 50.0%, Combined news 64.7%, Price & News 70.1% Profit: for Price & News 5.1% in 2 moths and for Price and News alone around half of it each	Yes	Not mentioned
Rachlin et al. (2007)	Cannot improve the predictive accuracy of the numeric analysis. Accuracy 82.4% for join textual and numeric analysis and 80.6% for textual analysis, and 83.3% numeric alone.	Yes	Not mentioned

Paul C. Tetlock et al. (2008)	1) the fraction of negative words in firm-specific news stories forecasts low firm earnings; 2) firms' stock prices briefly under-react to the information embedded in negative words; and 3) the earnings and return predictability from negative words is largest for the stories that focus on fundamentals.	Yes	Not relevant
Mahajan et al. (2008)	Accuracy 60%	No	Not mentioned
Butler and Kešelj (2009)	First method: 55% and 59% for character-grams and word-grams accuracy respectively, still superior to the benchmark portfolio. Second method: overall accuracy and over-performance precision was 62.81% and 67.80% respectively.	No	Not mentioned
Schumaker and Chen (2009)	Directional Accuracy 57.1% , Return 2.06% , Closeness 0.04261	Yes	Not mentioned
F. Li (2010)	Accuracy for tone 67% and content 63% with naïve Bayes and less than 50% with dictionary-based.	No	No
C.-J. Huang et al. (2010)	Prediction accuracy and the recall rate up to 85.2689% and 75.3782% in average, respectively.	Yes	Not mentioned
Groth and Muntermann (2011)	Accuracy (slightly) above the 75% guessing equivalent benchmark.	Yes	No
Schumaker et al. (2012)	Objective articles were performing poorly in Directional Accuracy versus Baseline. Neutral articles had poorer Trading Returns versus Baseline. Subjective articles performed better with 59.0% Directional Accuracy and a 3.30% Trading Return. Polarity performed poorly versus Baseline.	Yes	Not mentioned.
Lugmayr and Gossen (2012)	In progress	No	Not mentioned
Y. Yu et al. (2013)	Polarity with 79% accuracy and 0.86 F-measure on the test set. Only total number of social media counts has a significant positive relationship with risk, but not with return. It is shown that the interaction term has a marginally negative relationship with return, but a highly negative significant relationship with risk.	No	Not mentioned
Hagenau et al. (2013)	Feedback-based feature selection combined with 2-word combinations achieved accuracies of up to 76%	Yes	No
Jin et al. (2013)	Precision around 0.28 on average.	No	Not mentioned
Chatrath et al. (2014)	(a) jumps are a good proxy for news arrival in currency markets; (b) there is a systematic reaction of currency prices to economic surprises; and (c) prices respond quickly within 5-minutes of the news release	No	Not mentioned
J. Bollen and Huina (2011)	The mood 'Calm' had the highest Granger causality relation with the DJIA for time lags ranging from two to six days (p-values < 0.05). The other four GPOMS mood dimensions and OpinionFinder didn't have a significant correlation with stock market changes.	No	Not mentioned
Vu et al. (2012)	Combination of Previous days's price movement, Bullish/bearish and Pos_Neg features create a superior model in all 4 companies with accuracies of: 82.93%, 80.49%, 75.61% and 75.00% and for the online test as: 76.92%, 76.92%, 69.23% and 84.62%	NO	Not mentioned

2.12 Gaps Identification

Market prediction mechanisms based on online text mining are just emerging to be investigated rigorously utilizing the radical peak of computational processing power and network speed in the recent times. We foresee this trend to continue. This research helps put into perspective the role of human reactions to events in the making of markets and can lead to a better understanding of market efficiencies and convergence via information absorption. The below concrete suggestions in terms of further research avenues are noteworthy based on this review:

A. Dealing with the imbalanced market data specifically with the purpose of this kind of prediction.

B. Investigation of more advanced Semantic and Syntactic approaches specialized for this context.

C. Creation of hybrid models through deep integration of technical signals and investigation of their improvement potential via these text mining approaches.

D. Establishing a correlation between success of these approaches and market efficiency measures calculated through other mechanisms.

E. Integration of other machine learning algorithms specially clustering algorithms in such prediction systems as sources of prior information.

F. Pursuit of deeper investigation in relevant behavioural-economics principles and their utilization in this context.

G. Deeper investigation in utilization of relevant sentiment and emotional analysis in this context.

2.13 Chapter Summary & Problem Restatement

In this chapter the related literature is reviewed from multiple perspectives. The 3 major perspectives that are discussed are: text mining fundamentals, theoretical economics and text mining works that are in a market-predictive context.

In the 1st major aspect, the related fundamental topics of text mining are explored in order to identify the elements that are applicable in this research. The text mining process is broken down into 3 stages namely: Pre-processing, Machine Learning Algorithms and Evaluation methods. The first part of the discussion on the pre-processing phase revolves mainly around the techniques that are used in order to transform textual content that is unstructured into a structured format that is usable by machine. It is discussed how every record of textual news contents can be changed into a feature vector. In this regards topics like stop-word removal and tokenization are presented. The next part of the emphasis in this phase is how the feature vector can become more effective. For that a specific topic by the name of dimensionality reduction is introduced and explored.

Next machine learning algorithms that can be applied to this context are explored and compared. First, the nature of the problem subject to this research is described and classified as a binary classification problem. Then specific applicable algorithms are introduced and reviewed. Some of the discussed algorithms are: k-nearest neighbors, decision trees, naïve Bayes and Support Vector Machines.

Furthermore, it is explored how the results of the machine learning algorithms are evaluated in this context; what different evaluation measures there are and what the logic behind using each one is.

In the 2nd major aspect, the legitimacy of this research in an economics context is observed through two research avenues of theoretical economics, namely, conventional economics and behavioral economics.

In the 3rd major aspect, available systems in the literature which operate in a comparable context are reviewed and their models and designs are compared on multiple levels. As a result of this comparative analysis in the literature, a number of potential areas of improvements or gaps are identified. This is what forms the grounds for a problem formulation for this research. There are two levels of gap identification.

1- In general the text mining in the available works regarding market prediction is not yet as sophisticated as it can be. Partly because market predictive text mining is an emerging field and partly because some of the researchers are focusing more on some other discipline that is involved in this research; for example, they come from the economics department or the operations research department. Furthermore, many of the researchers that are working on text mining itself are using standard databases like movie reviews and are not yet applying and experimenting their finding in other contexts. Therefore in the available works the text-mining component is not customized and advanced as much as it may.
2- The second major finding from reviewing the literature from different aspects is on how the text mining component in the field of market prediction based on text mining can be enhanced and customized. The major problem that needs to be tackled is 'high dimensionality' of the feature matrix. There are too many features available once each feature is a word from the news text. Literature on text mining identifies this as a problem. In addition to high dimensionality, two further areas are identified that are underexplored in the current works in market predictive text mining of news but are supposed to be of significant impact as indicated in the literature on text mining. Those two problematic areas are

termed as Co-reference and Sentiment Ignorance. The former deals with the existence of semantic redundancy in a feature matrix that is word-based. The latter addresses the significance of sentiment analysis in natural language processing that is, however, being ignored in previous works in the context of market prediction through text mining of news.

In short, point 1 above is indicating a lack of needed enhancement in text mining in the specific context of market-predictive text-mining of news. This point is the result of a comparative analysis between state of text mining in other contexts versus the state of its application in this context. Point 2 is indicating how exactly that enhancement may be made and what specific problems should be addressed. This point is also the result of the former comparative analysis as well as an additional comparative analysis of the previous works in the context of market predictive text mining of news. With the above 2 observations in mind the problem that is tackled in this work is restated below:

"It is not clear if the problem of intraday prediction accuracy in the foreign exchange market can be addressed through enhancement of text mining of available news-headlines released prior to it."

3 Research Methodology

3.1 Introduction

The quality of the interpretation of the sentiment in the online buzz in the social media and the online news can determine the predictability of financial markets and cause huge gains or losses. That is why a number of researchers have turned their full attention to the different aspects of this problem lately. However, there is no well-rounded theoretical and technical framework for approaching the problem to the best of our knowledge. We believe the existing lack of such clarity on the topic is due to its interdisciplinary nature that involves at its core an array of topics from artificial

intelligence to behavioural-economics topics. We dive deeper into the interdisciplinary nature and contribute to the formation of a clear frame of discussion, propose models and demonstrate improvement in prediction accuracy.

In the next section of this chapter titled “Approaches to Research” we detail the research methodology. In its first part we dive deeper into the logic behind the conducted interdisciplinary literature review. The results of this review and analysis provide grounds for the 2nd and 3rd part in the Research Approach, namely, the Research Problem Formulation and Objectives Definition. Both of which are explained in detail accordingly in their respective sections. Once the gap that is targeted in this research is properly formulated in form of research objectives, Proposed Models are presented in the 4th part of the Research Approach. Based on which a detailed system design is presented in the 5th part, followed by Analysis of Methods in part 6, and Evaluation Methods in part 7. This chapter is concluded at the end with a summary section.

3.2 Approaches to Research

The flow of the research approaches and the main topics in each stage are depicted in Figure 3.1.

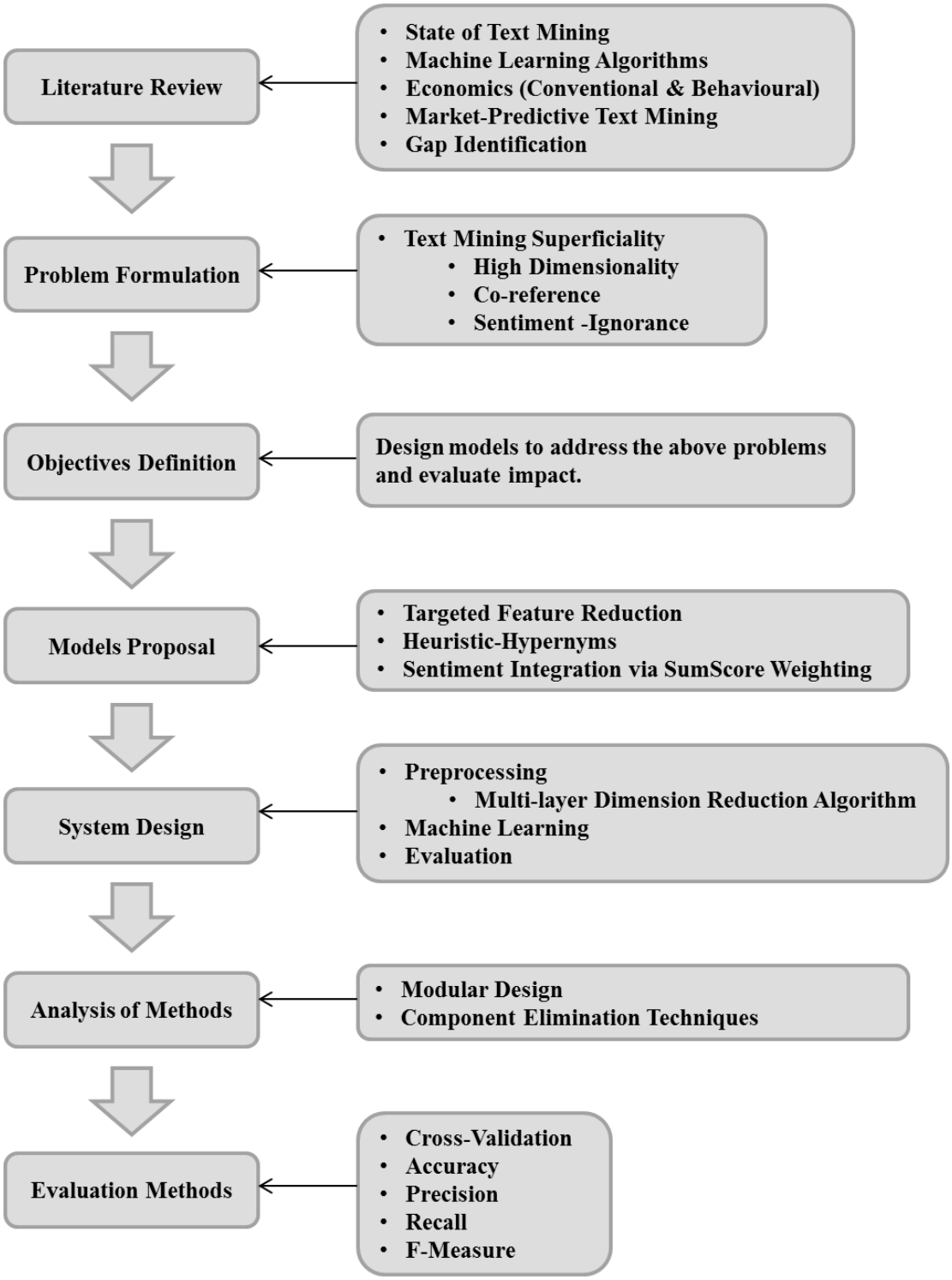

Figure 3.1 Research Methodology Framework

The approach in each of the phases is discussed in more detail in what follows.

3.2.1 Review of Related Works

The literature review contains 5 significant aspects that are required in this research to reach the necessary comprehension and devise an improvement model:

A. Text Mining State (Text Pre-processing Techniques)

B. Machine Learning Algorithms (Statistical Pattern Recognition)

C. Theoretical Economics (Conventional & Behavioural)

D. Market-Predictive Text-Mining

E. Gap Identification

There are 2 core themes in this research, namely, Text-Mining and Economics; and each of these themes has 2 sub-themes. The former, breaks down into Text Pre-processing aspect of text mining (item A) and Machine Learning Algorithms (item B) and the latter theme breaks down into Conventional Economics and Behavioural Economics (item C). Of course, as this is a computer science thesis the dominant attention is given to the former theme, however, it is explained why it is crucial to comprehend the latter theme economics in order to produce a solid and defensible research foundation. Once the above two themes are sufficiently explored another important section of the literature review is dived into, namely, a comparative analysis of the existing work that deals with the usage of text-mining for market-prediction (item D). Such a comprehensive comparative analysis is unprecedented and its results are published in form of a paper as a part of this thesis (Arman Khadjeh Nassirtoussi, Aghabozorgi, Ying Wah, & Ngo, 2014) . This comparative analysis helps identify the existing gaps (item E) in the current works and produces a foundation for the Proposed Models detailed in the next section.

In summary, it is vital to observe what are the state of the art techniques that can be used to process unstructured text in a way that it becomes machine readable and usable. This is termed as Text-Pre-processing in this work (item A). Then it is necessary to explore how machines can manipulate the output of the previous section and learn from it by recognizing patterns. This is termed Machine Learning Algorithms or Statistical

Pattern Recognition (item B). But before moving any further it is crucial to realize that the application area of these techniques in the case of this research is different from many others because it must make sense from an economic perspective. In that this research contains an additional complexity factor compared to other text mining classification problems, say, a spam-filtering algorithm, whereby, in the case of a spam-filter it is clear what text is supposed to be considered spam with absolute certainty. However, in this case financial markets are concerned which are inherently uncertain. Hence, it needs to be checked whether from at least a theoretical economic perspective the assumption that news has any impact on a financial market is legitimate. This is done in this work from both a conventional and behavioural economics perspective (item C). Once the theoretical economic legitimacy is established, an array of current works is delved into in which some comparable market-predictive activity is constituted based on text-mining. The results of this part of the review form a comparative analysis (item D) which in turn clarifies the weaknesses and lacks in the current systems in form of a gap-identification (item E). This then provides grounds for the Problem Formulation, Research Objectives and Proposed Models in the rest of this work as discussed in the following sections.

3.2.2 Problem Formulation

After a comprehensive literature review of both what is possible with text mining as well as the state of the art text mining that is used in current market-predictive systems, it becomes clear that in the mentioned context the text mining techniques can become customized and thereby enhanced. Therefore, this research identifies the problem as a lack of sophistication and customization for the context text mining is applied to and formulates the problem as a need to make improvement in text mining techniques in this context. Hence, the general formulation of the problem for this research is as below:

"It is not clear if the problem of intraday prediction accuracy in the foreign exchange market can be addressed through enhancement of text mining of available news-headlines released prior to it."

To be absolutely specific where the lack of enhancement seems to lie exactly, the below areas are identified:

A. Lack of investigation on dimensionality reduction

B. Lack of investigation on sentiment analysis

C. Lack of investigation on semantic analysis

D. Lack of investigation on FOREX in particular

E. Lack of investigation on non-topic specific news

F. Lack of investigation on text-mining of head-lines

More specification details on the problem that this research addresses and its requirements is briefed in the below:

The first aspect of the problem definition is a focus on a specific market-type. In general, there are multiple types of financial markets, namely: 1- Capital markets (Stock and Bond), 2- Commodity markets, 3- Money markets, 4- Derivative markets, 5- Future markets, 6- Insurance markets and 7- Foreign exchange markets. As their names imply, different assets are traded in each market; therefore they demonstrate different behaviors and separate research is conducted on each of them. As it is pointed out more specifically in the literature review section of this work, most of the works in the literature concerning some kind of usage of text-mining for a predictive purpose in a financial market is mostly attending to the stock-markets and specific company stocks based on textual content about those companies. Hence, this work enters a less explored

financial market namely the Foreign Exchange Market (FOREX) which facilitates the trading of currencies.

Furthermore, this work aims to take into use uncategorized breaking news rather than categorized news based on topic or company, etc. As pointed out in the literature the usual explored path in the past works is to isolate company-specific news, for example, and make predictions for the stock of a company based on that. However, the news channel that is used for this experiment is for financial breaking news. A focus on financial breaking news rather than a source of news that has all kinds of news pieces released is assumed to provide logical relevance and avoid noise. This is inspired by what traders in financial markets actually read. But no further categorization of news is utilized.

Moreover, in terms of the length of text, subject of this research is short-texts of news-headlines. The requirement of using news article-headlines rather than news article-bodies creates a text-mining focus on short texts rather than long texts for the proposed system. Naturally, when short pieces of texts are concerned there is less repetition of words in the same document and there are also fewer irrelevant words. Therefore, in such a context the level of significance of a word in a news piece cannot be determined by its repetition within it; however, at the same time there is less noise in the space as headlines are usually concise.

In terms of prediction time-line in the financial markets, both short and long-term predictions are subjects of research. In this work, however, the short-term prediction is explored as sudden impacts of news on the market are of interest and with the passage of time the number of factors producing noise on the initial impact increases. The short-term prediction that targets market-moves within the same day is termed as intra-day market prediction. To be specific, what is predicted is the directional movement (Up or

Down) of the market (price of a currency pair e.g. EUR/USD) 1 hour after the end of a 2-hour interval which includes the news-headlines released within it. This upwards or downwards movement at the 1-hour point after the interval is determined in relation to the point at which the market was 1-hour before the interval. This latter margin before the news-release interval ensures that the news release is indeed after the first point in time as different news sources may release breaking news with a slight time difference. The details of this structure are fully elaborated in a later section titled: *System Description*. However, at this point it suffices to mention that the system has a prediction time-line of 1 hour after a 2-hour news-interval which means the impact that is monitored and taken into consideration by the system can be 1 to 3 hours away from the exact release time of a new-headline.

In terms of required accuracy for practical use of such prediction system, because a binary decision between Up and Down is concerned, any results above 50% prediction accuracy is of interest and significance from a statistical perspective. Almost all of the previous works also, as listed in the next literature review section, compare their results with the odds of chance of 50% in such a context. Practical traders agree, too, that a system that can be accurate more than half of the time can be of value to them on a day to day basis. However, accuracy results of recent comparable efforts in the same research space are also provided in this work for a better evaluation of the results achieved here.

3.2.3 Definition of Research Objectives

Based on the above problem formulation the objectives of this research are:

1) To devise a methodology to test the existence of a predictive relationship between the content of financial news and a foreign-exchange-market currency-pair

2) To identify decisive text-mining improvement elements that can increase the accuracy of such market prediction through news mining
3) To devise improvement-techniques for those text-mining elements

3.2.4 Proposed Models

To address the ***1st objective*** of the research a mode[l] is designed that in its first phase takes news-headlines and historic market data at one end as input and produces a labeled news database in which each record contains a group of news-headline-words that are released in the same time-interval and the predictive market-movement label that is associated with that interval.

The next phase of the model allows the released words in each time interval to be processed by an algorithm in order to train a machine learning algorithm.

In this section it is identified that there are at least 3 problematic factors that are hindering the accuracy in a system like this and other systems in the literature. Finding solutions to these 3 problematic aspects can improve the prediction capability of the machine learning algorithm. This, thereby, addresses the ***2nd objective*** of this research which [t]argets identification of problematic aspects that may be improved. The 3 identified problems are:

A) High Dimensionality

B) Co-reference

C) Sentiment Ignorance

The *High Dimensionality problem* (A) is the most significant problem. It is caused by having a high number of words in the vocabulary that used in the news. As each word constitutes a feature, there are too many features available in the feature-matrix and this

causes the machine learning algorithm to encounter a decreased performance. This problem is widely known in the literature as "the curse of dimensionality".

The next problem is *Co-reference* (B) which is basically the fact that many words have the same meaning and are semantically referring to the same thing; however, they are looked at as separate features. Having many features to refer to the same thing is unnecessary but the problem is that it is not clear how to identify which words are referring to the same thing and group them together. Therefore, Co-reference is a problem and causes semantic inefficiency. If Co-reference is addressed, in addition to having more semantic accuracy the system will have fewer features as well. Therefore, dealing with the problem of Co-reference (B) additionally helps with the problem of High Dimensionality (A), too.

The next problem is termed as *Sentiment Ignorance* (C). Such text-based predictive systems are expected to predict human reaction in markets based on human language. Therefore, it is clear that the notion of language sentiment is vital and must play a role in the system. However, most systems do not have a model to integrate the sentiment associated with words into their feature matrix and therefore are ignorant of the sentiment associated with the language.

The ***3rd objective*** of this research is ad[dr]essed by devising and proposing models that are targeted at the above problems.

The High Dimensionality problem (A) is addressed in two ways: Firstly, an algorithm is designed by the name of "Targeted Feature Reduction" that reduces the number of features to a bare minimum. Secondly, the two models that are proposed to address Co-reference and Sentiment Ignorance are also helping with feature reduction. The fact that in the proposed system, high dimensionality is considered as a crucial problem to tackle and therefore is addressed at multiple layers causes the overall model that is produced in

this research to be called “A Multi-layer Dimension-Reduction Algorithm”. How this is done is explained further in the design section.

The Co-reference problem (B) is addressed by introducing a Semantic Abstraction model with the use of hypernyms of the news-headline words. A hypernym of a word is like a super-word or a parent-word and replacing the child-words with their parent words contributes to solving the Co-reference problem and reduces the number of features as well. The proposed model here is termed “Heuristic-Hypernyms”.

The Sentiment Ignorance problem (C) is addressed by introducing a Sentiment Integration model. It integrates sentiment analysis capability into the algorithm by proposing a sentiment weight by the name of “SumScore” that reflects investors’ sentiment. Thereby the system is not ignorant of the sentiment anymore. Additionally, this model reduces the dimensions by eliminating those that are of zero value in terms of sentiment and thereby contributes to solving the high dimensionality problem as well.

3.2.5 System Design

The system design revolves around realization of the above models. To recap, there are 3 specific problems for which a model is proposed to tackle:

1- “Targeted Feature Reduction” to address the *High Dimensionality* problem

2- “Heuristic-Hypernyms” to address the *Co-reference* problem.

3- “Sentiment Integration by SumScore” to address the *Sentiment Ignorance* problem.

Furthermore, all of the above models are subsections of an overall model that is named:

“Multi-layer Dimension Reduction Algorithm”

The reason that dimension reduction is emphasized on is that it is a crucial problem and not only model 1 in the above is dedicated to it but also models 2 and 3 are also

contributing to solving this matter as a secondary goal besides their primary goals which are countering Co-reference and Sentiment-Ignorance respectively.

Figure 3.2 shows the actual flow in the design of the Multi-layer Dimension Reduction Algorithm that has Semantics in layer 1 and Sentiment in layer 2.

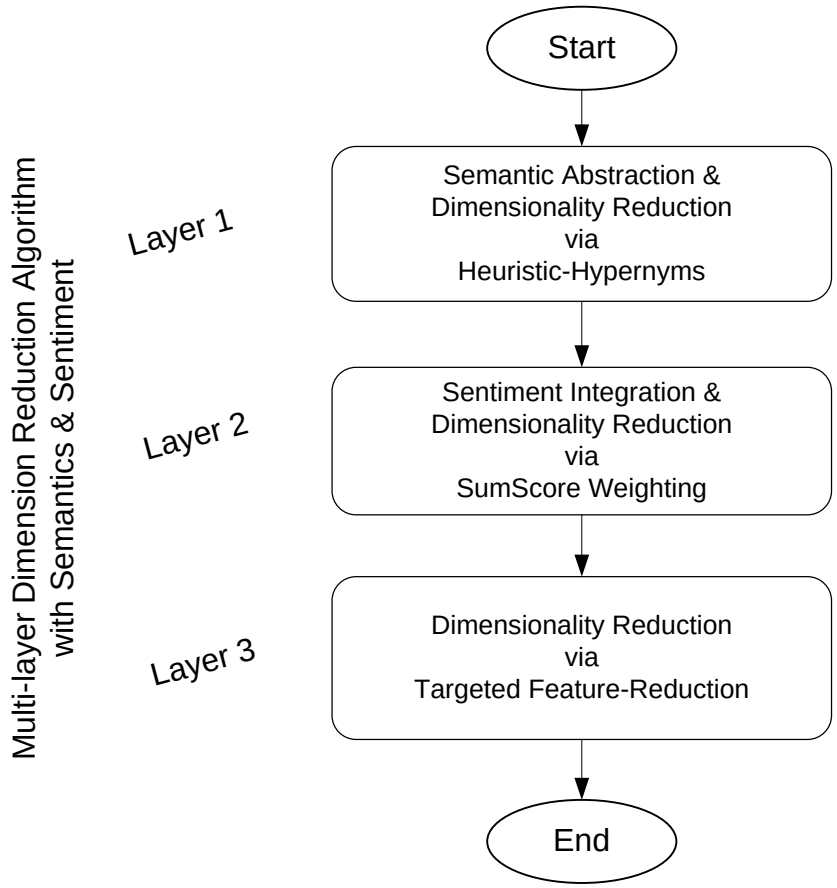

Figure 3.2 Multi-layer Dimension Reduction Algorithm

The entire market predictive system has 3 major components as depicted in Figure 3.3.

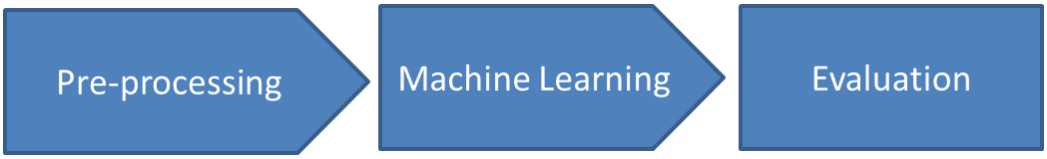

Figure 3.3 Market Predictive System Components

The "Multi-layer Dimension Reduction Algorithm with Semantics and Sentiment" sits in the first phase of the overall system namely the "Pre-processing" phase.

The Pre-processing phase is responsible for turning the textual content into a feature matrix that is usable by the machine learning algorithm that comes in the next phase.

Different machine learning algorithms are experimented with namely, k-Nearest Neighbors, Naïve Bayes, Support Vector Machines (SVM) among others. It is concluded that SVM produces the best performance. This is explored and explained more in the chapter titled *Experimental Results and Analysis* in Section 5.9 titled *Machine-Learning-Algorithm Variation* accordingly.

And in the last phase the output of the machine learning phase is evaluated by a number of measures as discussed later in the section on evaluation methods (3.2.7).

A detailed implementation design of the system is presented in Figure 3.4. It can be observed that in the Pre-processing phase of the overall system, before the Multi-layer Dimension Reduction Algorithm there are two more phases, namely for Data Retrieval and Data Preparation. The output of the Data Preparation phase is a labeled news database in which each record contains a group of news-headline-words that are released in the same time-interval and the predictive market-movement label that is associated with that interval. (This is previously referred to under the section for Proposed Models for the 1st research objective as a part of the first phase of the model.)

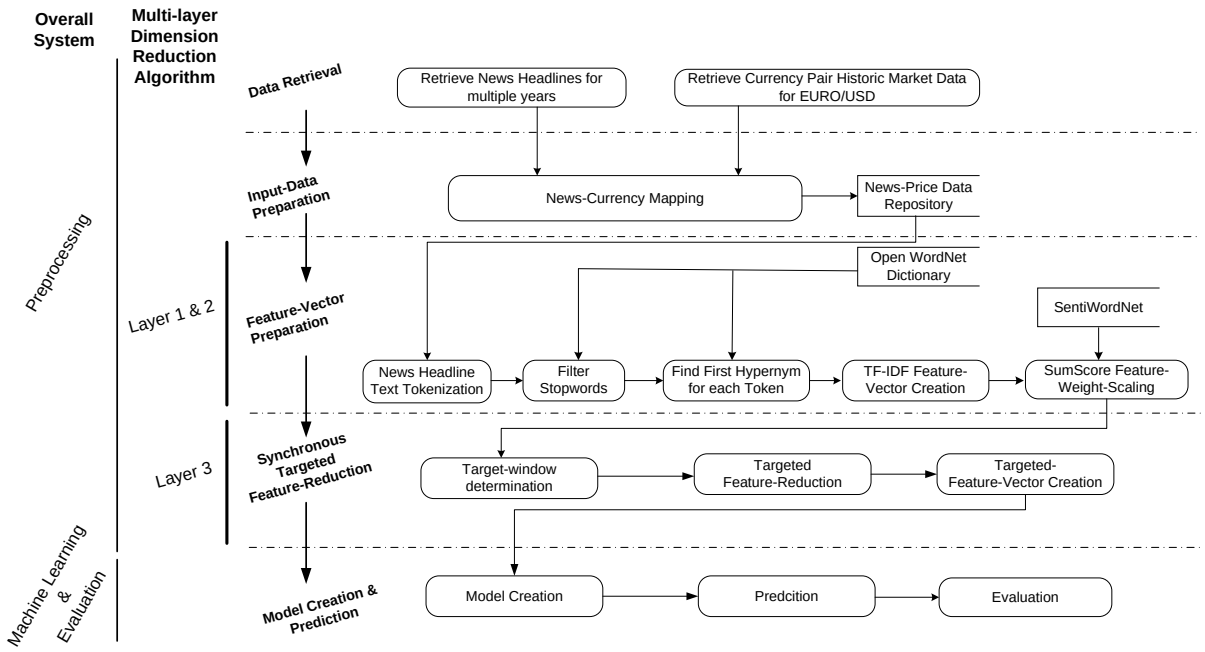

Figure 3.4 Detail System Design

3.2.6 Analysis of Methods

In order to analyze the proposed models, a prototype of the system is implemented with RapidMiner. The prototype takes as input the labeled news database in form of an excel sheet that is retrieved and prepared in the first phase of Pre-processing. Then it uses the *Multi-layer Dimension Reduction Algorithm with Semantics and Sentiment* to create a feature vector that is minimal for a record that is targeted from prediction. From there a feature matrix is created that is used in the next part for training of the machine learning algorithm. Once a prediction is made for a record the prototype compares the predicted label with the actual label and records the result for evaluation purposes.

The modular design of the system allows analysis to be conducted for each of the sub-models of the algorithm.

Here is a detail account of how the analysis is conducted for each of the sub-models:

A) Heuristic-Hypernyms Model for Semantic Abstraction

In order to analyze the impact of this model the Heuristic-Hypernym Model is eliminated from the system. This is done by elimination of the Semantic Abstraction layer which occurs by looking up the hypernym of a given news word in an ontology by the name of WordNet. However, it is still needed to change each word to a standard form that can be looked up in WordNet and thereafter in SentiWordNet which is the Sentiment oriented ontology. Therefore, in this section of the analysis each word is replaced by its WordNet stem that can later receive a sentiment weight and not by its hypernym. In this way the effectiveness of the hypernym-based abstraction is assessed versus the scenario where every word is present (although in the stemmed form).

B) Sentiment Integration Model via SumScore Weighting

In order to analyze the impact of this model, the SumScore weight, which is the sentiment score that is devised in this research, is eliminated from the weighting equation. The overall weighting equation looks like this: TF-IDF*SumScore. Once the SumScore is eliminated from it the system is run by a TF-IDF weighting alone in place. This determines the positive impact of the existence of the SumScore, produced in this work, in the weighting equation.

C) Targeted Feature Reduction Model

In order to analyze the role of this model, it is eliminated from the system for comparison purposes in the following manner.

The Targeted Feature Reduction model basically reduces the features to only those features that are used in a certain record which is targeted for prediction and then builds the feature matrix from the entire database for those only. To determine the

effectiveness of this model, the matrix can be simply created by all the available features and not only the reduced number. In this way it becomes obvious how the prediction accuracy decreases as the ability of the machine learning algorithm plunges as the number of features spike.

3.2.7 Evaluation Methods

Every time an experiment is run using a model, it is evaluated using cross-validation. Cross-validation is a model validation technique to assess how it performs on an independent dataset that is separate from the training dataset. It helps to estimate how accurately a predictive model will perform in practice. In a prediction problem, a model is usually trained using a dataset that is referred to as a training dataset, then it is tested against a dataset of unknown data which is called a testing dataset.

The generic cross validation model has been adapted for this research in the following manner. The algorithm that is devised in this research has a time series of news-headline-groups labeled with market movements at its disposal for training and testing purposes. The generic cross validation would just split the records available and assign a part for testing and another for training. However, as the proposed algorithm is dealing with a time-series, it has a dynamic design with regards to the dataset. In order to predict a record, it takes all records that are available before that record for training. In this sense it is slightly different in that the training dataset has a different number of records available in it depending on which record in the testing dataset is being tested. Hence, the implementation is unique but at the end, every time a model is tested it is tested on a record that has not been used in training and in that the cross validation is applied. In this research, the prototype of the algorithm is designed and implemented in such a way that different values can be assigned for the size of the testing dataset. It then automatically conducts the required numbered of experiments and calculates a number of measures to evaluate the model. These measures are Accuracy, Precision,

Recall and F-Measure. These measures are standard and widely used in the assessment of the research regarding binary classification. They are described in the following.

To recap, the algorithm in the prototype is performing a binary classification on text-groups, deciding whether each group is driving the market UP i.e. having a Positive (P) impact on the market or DOWN i.e. having a Negative (N) impact on the market. Hence, there are 2 classes, namely, P and N; to which a text-group can be assigned.

If a testing sample size is 12, the below results are produced which can be formed in a table that describes the calculation of the first 3 evaluation measures.

Table 3.1 Evaluation Measures Calculation

Test Sample Size = 12

Accuracy = 10/12 = 83.33%	True N (Total 9)	True P (Total 3)	Class **Precision**
Predicted N (Total 9)	8	1	8/9 = 88.89%
Predicted P (Total 3)	1	2	2/3 = 66.67%
Class **Recall**	8/9 = 88.89%	2/3 = 66.67%	Total Correct 10

As one can see in table 3.1, the sample size is 12, from which the algorithm classifies 9 records in class N and 3 records in class P. However, from the 9 records in class N, 8 truly belong to that class and 1 is an error. Furthermore, from the 3 records in class P, 2 truly belong to that class and 1 is an error. Therefore, in total 10 records have been assigned the correct class and 2 have been assigned an erroneous class.

A) Accuracy

The accuracy is the number of correct predictions out of all predictions. In this case, as explained above, it would be 10 out of 12 i.e. 83.33%.

However, it is not enough to observe the overall accuracy alone, because it is crucial to be aware how each class is performing. For that, 2 further measures are introduced in the rest, namely, Precision and Recall.

B) Precision

There is a precision calculated for each of the classes. Precision is the percentage of the relevant records that are assigned to a class. For example, out of the 12 test records, 9 have been assigned in class N. However, only 8 of them truly belong there. Therefore, the precision for class N is 8 out 9 or 88.89%. In the same manner, 3 records in total have been assigned to class P, but only 2 of them truly belong there. Therefore, the precision for class P is calculated as 2 out of 3 or 66.67%.

C) Recall

Recall is different from Precision in that it is not looking at the predicted classes per se. It rather takes into account the total number of records that truly belong to a class, and then observes how many of them have actually been assigned to that class in the classification.

For example, there is a total of 9 records in our test sample which in reality belong to class N, however, the classification algorithm only manages to determine that 8 of them belongs to class N. Therefore, the Recall for class N is 8 out of 9 or 88.89%. In the same fashion, among the testing dataset, there are 3 records that belong to class P in reality. However, the algorithm manages to predict 2 of them correctly. Therefore, the Recall for class P is 2 out of 3 or 66.67%.

D) F-Measure

Another measure that is widely used in the literature is the so called F-Measure. It is one measure that somewhat covers Precision and Recall and in certain circumstances can evaluate one as more important than the other by assigning more weight to one. In short, the F-measure is a weighted average of Precision and Recall, its best value is at 1 and its

worst value is at 0. To be exact, F-measure is the harmonic mean of precision and recall as below:

$$F = \frac{1}{\alpha\frac{1}{P} + (1-\alpha)\frac{1}{R}}$$

(3.1)

Where P is the Precision and R is the Recall.

The usual and common form of the F-measure is the so called balanced F_1 measure. In which α is ½ whereby the weight assigned to Precision and Recall are equal. This produces the below form:

$$F_1 = 2 . \frac{P . R}{P + R}$$

(3.2)

Therefore, in the case of the example provided in the above table, the F_1 measure for class N is calculated as below:

$$F_1 = 2 \times \frac{0.8889 \times 0.8889}{0.8889 + 0.8889} = 0.8889$$

In this case there result has the same value because Precision and Recall in this experiment happen to have the same value but it does not always need to be the case.

In the same manner for class P, Precision and Recall happen to have the same value being 66.67% which will render F_1 equally.

3.3 Chapter Summary

In this chapter a high level description of the research methodology is presented. A framework of research is introduced that is organized into these phases: Literature Review, Problem Formulation, Objectives Definition, Proposed Models, System Design, Analysis of Methods and Evaluation.

In the Literature Review, the work is structured into Text Mining, Machine Learning, Economics and Market-Predictive Text-Mining Works and the motivation behind this organization and its contribution is laid out.

Then the problem is clearly formulated based on the existing gaps in the current systems and research objectives are defined accordingly to address them and create improvements.

In terms of the proposed models, an overall model of the system is presented as well its sub-models. Each of the proposed sub-models is focused on a specific aspect of the problem that is amplified in an according section. The main 3 of such problems are titled as: High-Dimensionality, Co-reference and Sentiment-Ignorance.

Based on the model descriptions a high level system design is devised and presented that is explored in detail in the next chapter. It is illustrated at this stage where in the overall system the Multi-layer Dimension Reduction Algorithm sits, what other system components there are and how they interact with each other.

In the section on analysis of methods, the approach that is utilized to analyze each of the proposed models is explained. Essentially it is described how exactly each of the models can be eliminated from the system so that their impact and effectiveness on the accuracy of the overall system can be determined.

In order to determine the effectiveness of the classification performed by the proposed models a number of evaluation measurement are introduced in the section on evaluation methods. These measurements are widely used in the literature. The motivation behind using each one of them is also alluded to.

Next chapter delves into a detailed system design and further amplifies the proposed models.

4 System Design

4.1 Introduction

In this chapter the proposed system design is explained in detail. It starts with an overview of proposed models in Section 4.2 that is followed by every major component of the system described in dedicated sections as follows: 4.3-Data Retrieval, 4.4-Input-Data Preparation (News-Currency Mapping), 4.5-Text Tokenization and Stop-word removal, 4.6-Semantic Abstraction via Heuristic-Hypernym Modeling, 4.7 Sentiment Integration, 4.8-Synchronous Targeted Feature-Reduction, 4.9-Model Creation and Prediction, 4.10-Machine Learning, 4.11-Evaluation Phase, and finally 4.12-Chapter Summary.

4.2 Overview of Proposed Models

The job of the system that is proposed in this work is to take time-stamped news headlines as input, group them together based on a preset interval and predict if the market is headed upwards or downwards in the next interval.

This prediction is realized by making a decision on assigning a group of news headlines in an interval to an ‘Up’ or ‘Down’ class. In other words, the job of the machine-learning component of the system is a binary classification of the textual input.

In this work, the relevant text mining process is considered to have 3 major phases, namely:

1- Pre-processing

2- Machine Learning

3- Evaluation

These are depicted in the first column on the left in Figure 4.1.

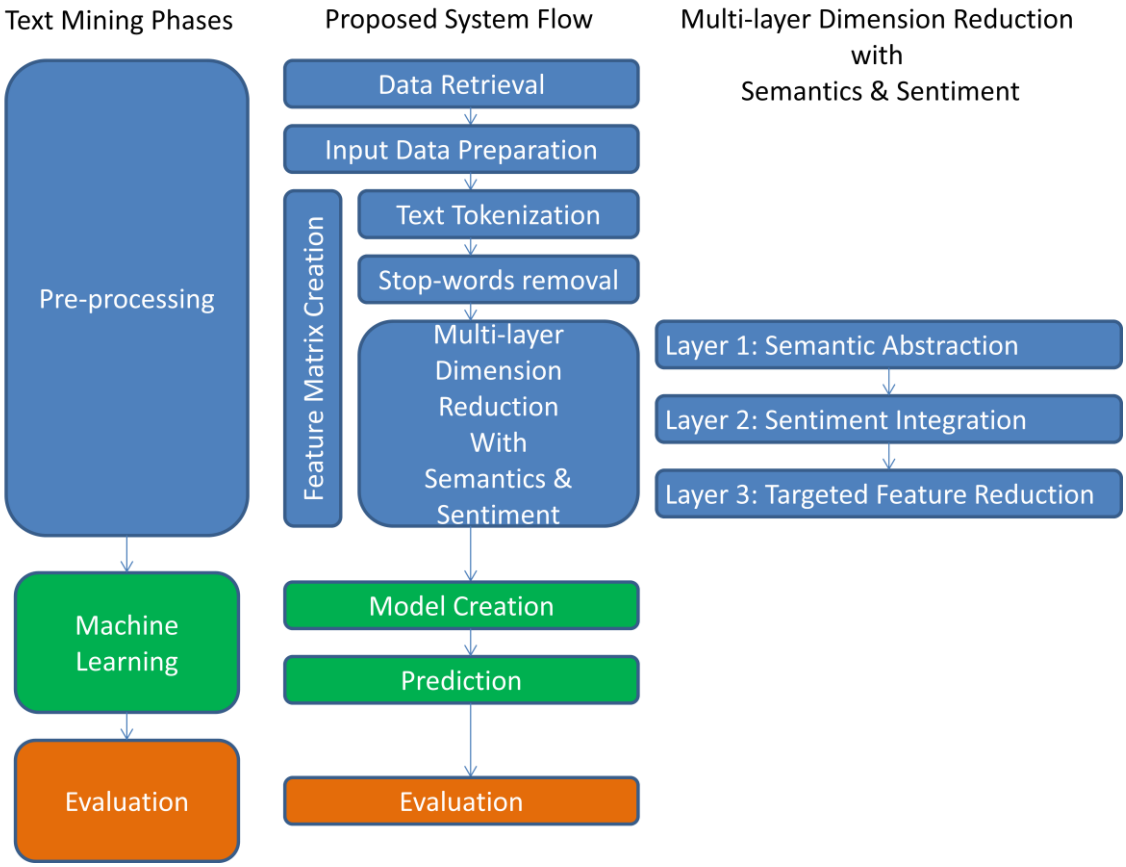

Figure 4.1 High Level and Low Level System Flow and the Multi-layer Algorithm

The system that is proposed in this work maintains the above 3 phases. A more detail view of the main components that are proposed for each of the above phases in the proposed system is to be seen in the second column in Figure 4.1.

As it can be seen most of the proposed components belong to the Pre-processing phase. Pre-processing in this thesis is considered to be the component that defines text-mining

and differentiates it from data-mining. It covers everything that requires to be done (processed) before a machine readable input is ready. When such input is ready, the rest of the work is very similar to data-mining. Therefore, as the focus of this work is text-mining, its main contribution is made in this phase of the proposed system.

The column on the right in Figure 4.1 provides deeper insights into the core contribution of this work which is an algorithm termed as the 'Multi-layer Dimension Reduction with Semantics and Sentiment' or 'Multi-layer Dimension Reduction Algorithm'. In this work, it is also sometimes referred to as just the 'Multi-layer algorithm' in short. The algorithm is placed in a phase by the name of Feature Matrix Creation. This is the phase where a feature-vector is decided on and based on it a feature-matrix is created that is ready to be fed into the machine learning algorithm that comes next for the purpose of its training (model creation) and prediction execution.

The Multi-layer algorithm has 3 main layers, namely:

1- Semantic Abstraction

2- Sentiment Integration

3- Targeted Feature Reduction

In this thesis, the last layer is sometimes referred to by its full name which is 'Synchronous Targeted Feature Reduction'. Each of these layers and their responsibilities and significance is explained in detail in an according section later in this chapter.

The Multi-layer algorithm allows the system to reach a high accuracy by tackling some of the most fundamental text-mining challenges around in a manner that is enhanced and customized for the context of the objectives of this work. In short it fulfills 3 main objectives:

1- Reduction of dimensions (features) at every layer in an incremental fashion.

2- Integration of semantics in a manner that reduces semantic redundancy (co-reference) which is usage of multiple words for the same concept or entity.

3- Integration of language sentiment in a way that the amount of emotional-charge or sentiment-load of a word is taken into consideration in weighting a feature. Language is more than words and letters; and the emotion or sentiment that each word carries matters a lot. But many text-mining systems are ignorant of this fact or are not enhanced enough to accommodate for it.

Figure 4.2 provides a detailed design of the flow of activities in each layer of the proposed system.

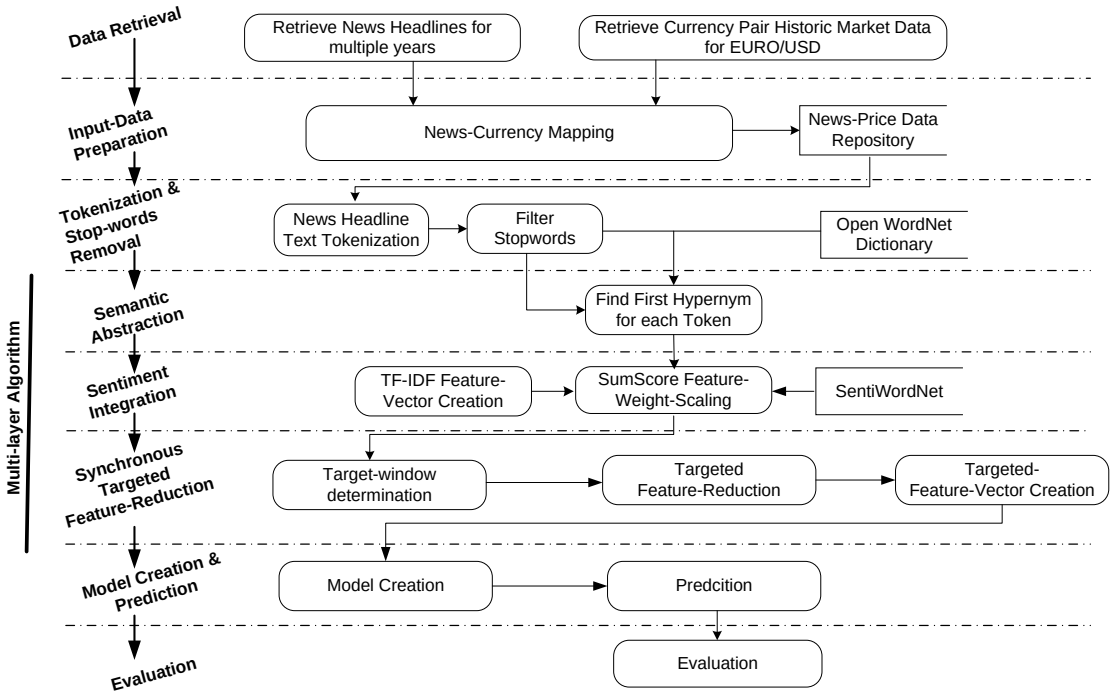

Figure 4.2 Detail System Flow

The rest of the content in this chapter is organized according to the detail system flow that is presented in Figure 4.2. In this way the logical order of the activities is observed.

4.3 Data Retrieval

In the Data Retrieval phase two main datasets are retrieved. One is the news-headlines dataset for multiple years which can be retrieved from a financial news website like MarketWatch.com or others with a Really Simple Syndication (RSS) function. In order to retrieve headlines for multiple years, Google cache is accessed via the Google RSS reader API. Table 4.1 lists a number of retrieved news headlines as examples. Note that the news date and time of the publication is also retrieved.

Table 4.1 Example of Retrieved News Headlines

News Headline	News Date & Time (GMT)
'Strong' demand outside North America propels Caterpillar to 13% quarterly profit growth	18/4/2008 11:37:05
Dow futures stage relief rally after Citigroup results	18/4/2008 11:01:27
Citigroup swings to quarterly loss of $5 billion, revenues fall 48%	18/4/2008 10:57:23

The other dataset is the foreign exchange market (FOREX) historic data for the desired currency-pair which in our case is Euro/USD. There are many sources to obtain this data. In this case it is retrieved by the help of the FXCMMicro Desktop client. Table 4.2 lists some examples of such data. The retrieved data here is for 2-hour time intervals. At each entry 8 pieces of data are present, namely the Open, High, Low and Close for both the Ask and the Bid price. Note that the (Close, Bid) of one entry is equal to the (Open, Bid) of the next entry and so forth. Also note that Open marks the beginning of the interval and Close marks the end of an interval. Last but not least, note that 18/04/2008 13:00:00 for example is referring to the interval from 13:00 to 15:00 i.e. an interval is marked by its starting time.

Table 4.2 Example of Retrieved Currency-Pair Data

Date & Time	Open, Ask	High, Ask	Low, Ask	Close, Ask	Open, Bid	High, Bid	Low, Bid	Close, Bid
18/04/2008 15:00:00	1.58041	1.58159	1.57980	1.58103	1.58016	1.58134	1.57955	1.58082
18/04/2008 13:00:00	1.57571	1.58140	1.57422	1.58041	1.57549	1.58115	1.57397	1.58016
18/04/2008 11:00:00	1.57329	1.57581	1.57221	1.57571	1.57301	1.57556	1.57196	1.57549
18/04/2008 09:00:00	1.57404	1.57664	1.57141	1.57329	1.57382	1.57639	1.57116	1.57301
18/04/2008 07:00:00	1.58441	1.58557	1.57129	1.57404	1.58412	1.58534	1.57104	1.57382

4.4 Input-Data Preparation (News-Currency Mapping)

In this phase of the system, the input-data for the next phase is prepared by mapping the news-headlines as well as the currency data onto a time series as depicted in Figure 4.3.

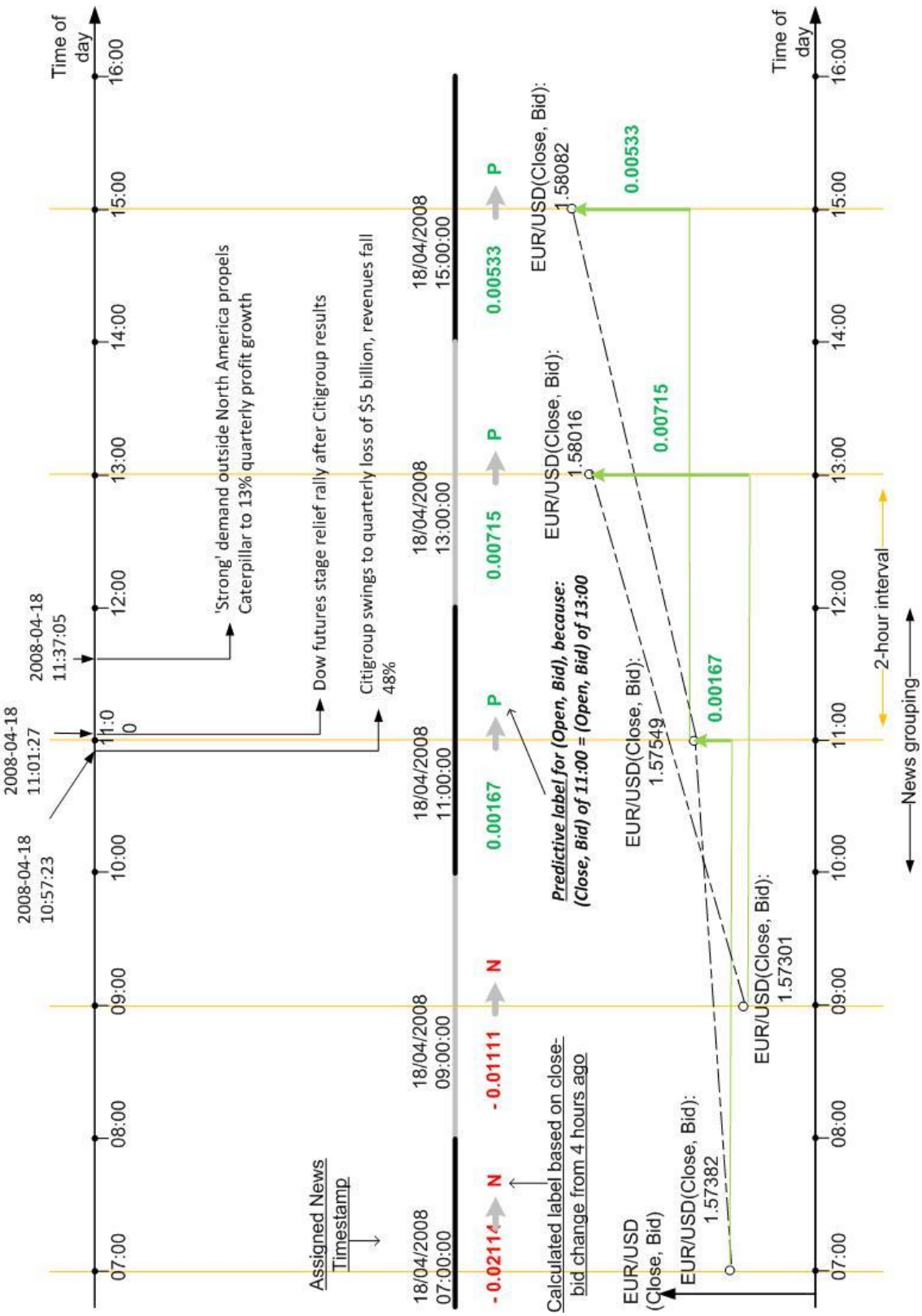

Figure 4.3 News-Currency Mapping

Figure 4.3 is actually composed of 3 separate sections:

The first section that is at the top of the figure is simply a time-line that pin-points the release-time of each of our example news pieces.

The second section that is placed in the middle is a line that indicates how news-headlines are grouped together. All news-headlines that are released between 10:00 and 12:00 for example are grouped together as shown in Table 4.3. This means the news-headlines are grouped based on their published time; so that the system can observe if there are eventually words in an interval that can cause a reaction in the market afterwards. Then a date-time-stamp is assigned to this news-group which in this case is "18/04/2008 11:00:00". This date-time-stamp is for grouping of the news and hence is called the news-grouping date-time-stamp. Note, the time component "11:00:00" in the stamp is chosen because it is in the middle of 10:00 and 12:00 in our example. The news-grouping date-time-stamp is different from the currency-data date-time-stamp that is explained in the next part.

The third section that is at the bottom of the figure has 2 axes. The X axis is the time-line again. Note that on this time-line 2-hour intervals are indicated differently. The 2-hour interval is not e.g. 10:00 to 12:00 anymore but 11:00 to 13:00. For this interval a new time-stamp exists that looks the same as "18/04/2008 11:00:00" for example but in this new context it is referring to the 11:00 to 13:00 time-interval. This is called the currency-data date-time-stamp and is different from the above news-grouping date-time-stamp in that it is 1 hour in the future i.e. the example interval does not end at 12:00 but at 13:00. These intervals are marked in Figure 4.3 by vertical lines across the figure.

In this way the system has 2 separate sets of time-stamps: one for the news-grouping and one for the currency-data. They look the same in terms of their composition but are lagged by 1 hour. The reason for this setup is to use the news-group to predict the

currency-data that is 1 hour in the future. It is assumed that a time-lag is required for market impact realization of the news, having taken into consideration suggestions made by the past research like the work of Reboredo et al. (2013). Past research has indicated sixty minutes to be looked at as a reasonable market convergence time to efficiency (Chordia et al., 2013; Chordia et al., 2005). Market convergence refers to the period during which a market reacts to the information that is made available and becomes efficient by reflecting it fully. Furthermore, this setup provides a 1-hour margin before the news-release time as well, which is just to ensure that the news is really exactly between two points in time as different news sources may have somewhat different release-times and margins around news release-time increase the certainty of news-release occurrence in the desired time-window. What exactly is predicted is explained in the following.

Next, the Y axis is the price-point for Euro/USD (Close, Bid) at that point in time. Note, that this price-point is available at the turning point of each currency-data date-time-stamped interval. For example, the Euro/USD (Close, Bid) price-point for "18/04/2008 13:00:00" is 1.58016, the Euro/USD (Close, Bid) price-point for "18/04/2008 11:00:00" is 1.57549 and the Euro/USD (Close, Bid) price-point for "18/04/2008 09:00:00" is 1.57301 as shown in Figure 4.3.

The next important piece of information that is illustrated in this part of Figure 4.3 is the calculation of a label for each news-group. The ultimate objective of this system is to predict the direction of the price-movement based on a given set of news-headlines. In other words, each group of headlines must be associated with a label that indicates the upward or downward movement of the price according to those headlines i.e. that news-group. As mentioned before, the piece of price data that the system is working with is Euro/USD (Close, Bid). But Euro/USD (Close, Bid) alone is just an indication of a

price-point and not a price direction. Hence, a new value is created in the system to indicate the change in Euro/USD (Close, Bid) as below:

$$C_i = Cb_i - Cb_{i-2} \qquad (4.1)$$

$$Label: \begin{cases} IF\ (C_i > 0) : P \\ IF\ (C_i <= 0) : N \end{cases} \qquad (4.2)$$

Where C_i is the change in Euro/USD (Close, Bid) of the interval-turning-point *i*, Cb_i is the (Close, Bid) at that point and the Cb_{i-2} is the (Close, Bid) at 2 intervals ago or 4 hours ago (formula 4.1). The 4-hour interval is composed of 2 times 2-hour market-intervals; and the 2-hour news-interval is aligned at its center so that there is a 1-hour margin at each end (Figure 4.3). As mentioned before, the margins help ensure the news release-time is between the two points in time at the ends of the 4 hour-interval. Furthermore, the 1-hour margin at the right hand-side constitutes the predictive timeline which is 1-hour at minimum.

If C_i is greater than 0 then label "*P"* for Positive is chosen and otherwise label "*N"* for Negative is chosen (formula 4.2).

For example, as mentioned at point 13:00 on the time-line the Euro/USD (Close, Bid) is 1.58016 and it is 1.57301 at point 09:00 which is 4 hours before it. The change here would be $C_{13:00} = 0.00715$ which is greater than 0 which is construed as label "P".

Note that point 13:00 on the time-line is at the end of the interval between 11:00 and 13:00. This interval as explained below is currency-data date-time-stamped as "18/04/2008 11:00:00". At the same time the same date-time-stamp value is used for a news-group among the news-grouping date-time-stamps. However, this news-group is composed of the news-headlines released between 10:00 and 12:00 as explained before.

This association of a group of headlines in a news-group with a label via the above date-time-stamping system is called the News-Currency Mapping, which is the ultimate

illustration objective in Figure 4.3. It is at the core of the labeling mechanism in this system. The above described process flow of News-Currency Mapping is also summarized in Figure 4.4.

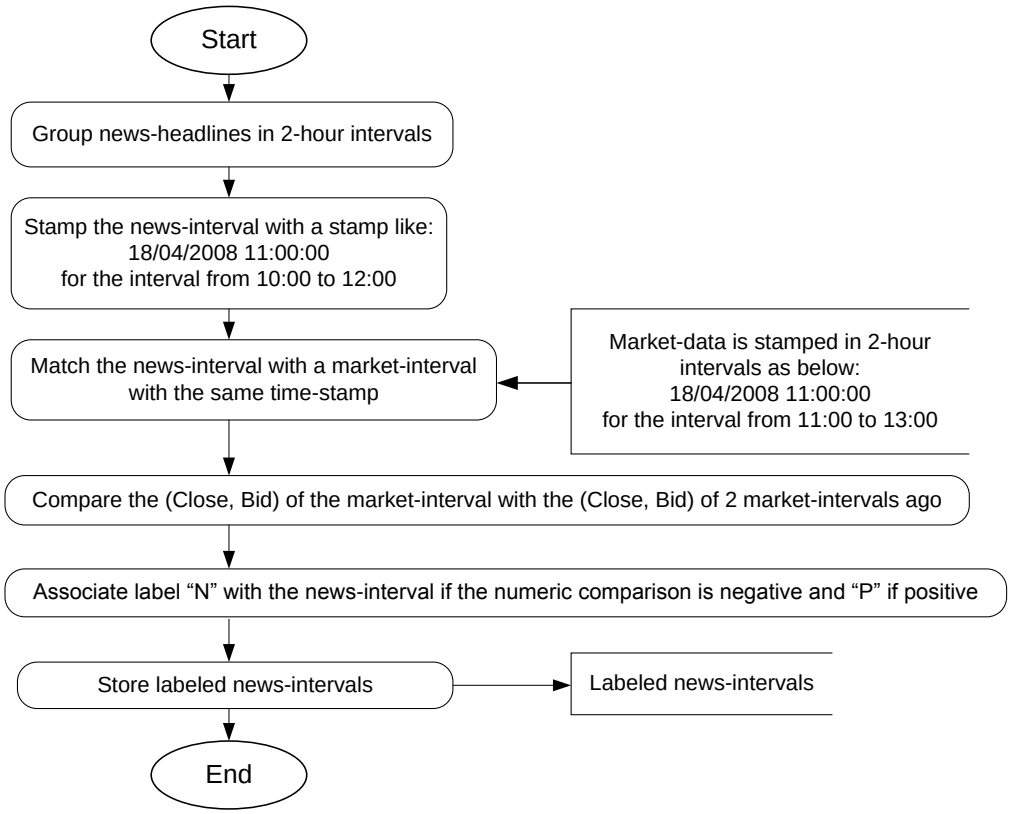

Figure 4.4 News-Currency Mapping Process Flow

There is one last point to note while thinking about this mechanism: In order to refer to the point 13:00 on the time-line in terms of the currency-data in the above, the interval before it i.e. 11:00 to 13:00 is selected. The reason is that a (Close, Bid) is referring to the Bid-value at the end of an interval. However, as described before the (Close, Bid) at 11:00 to 13:00 is equal to the (Open, Bid) at 13:00 to 15:00. Hence, in other words, it is exchangeable in this work to state that the system is predicting the (Close, Bid) of date-time-stamp “18/04/2008 11:00:00” or the (Open, Bid) of “18/04/2008 13:00:00”. Furthermore, this prediction is made by inputting the news-headlines released from 10:00 to 12:00 into the learned-model, the creation of which is described in a separate section.

At the end of this phase, each news-group is associated with a calculated label per above description and stored in a repository. Table 4.3, demonstrates one entry of that repository which is based on the example headlines in Figure 4.3.

Table 4.3 Example of News-Grouping

News-Grouping Date-Time-Stamp	**All news-headlines in the same news-group**	**Label**
18/04/2008 11:00:00	'Strong' demand outside North America propels Caterpillar to 13% quarterly profit growth Dow futures stage relief rally after Citigroup results Citigroup swings to quarterly loss of $5 billion, revenues fall 48%	P

From this point on, the activities are focused on creation of a feature-matrix. First, a feature-vector is prepared for each record in the news-grouping repository. That means in the case of the record in Table 4.3 for example, the text of the news-headlines, which is shown in the second column in the table, is boiled down into a number of features. This process is depicted in Figure 4.4, followed by a step-by-step demonstration based on the above example record in each of the following sections.

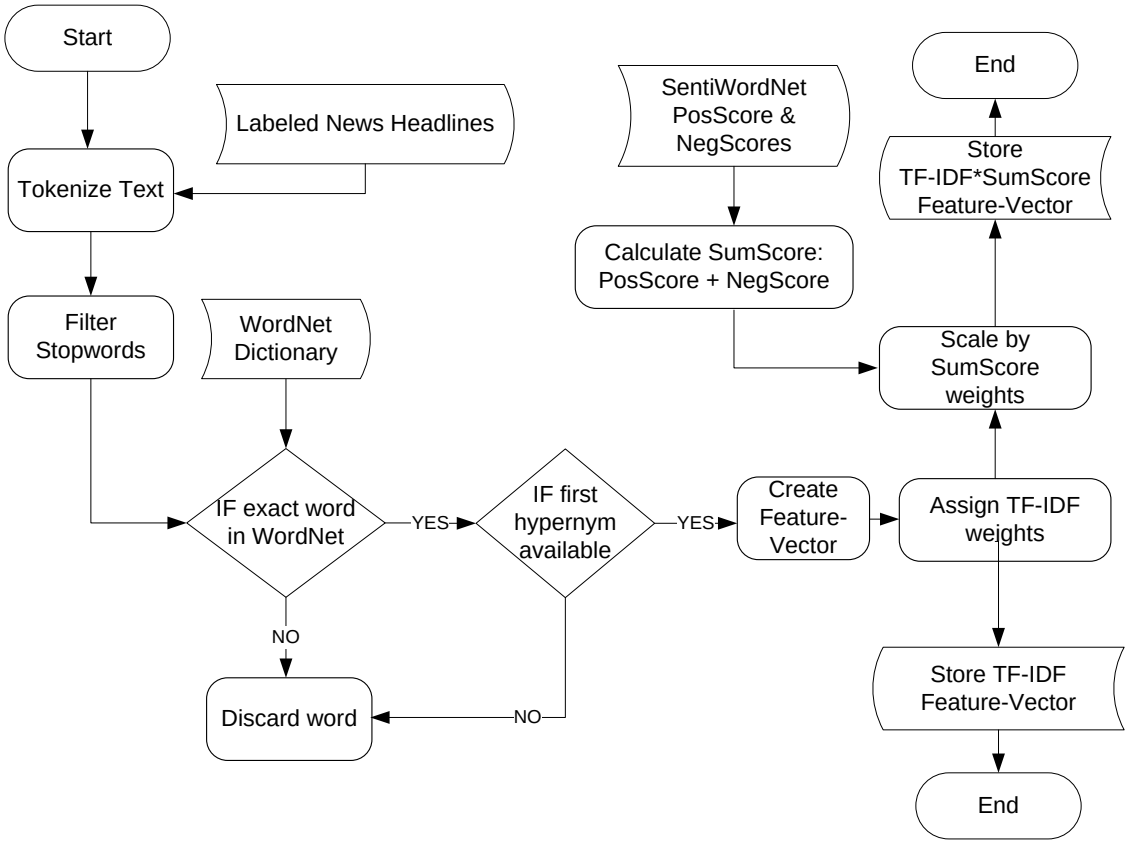

Figure 4.5 Complete Feature-Vector Preparation Flow

4.5 Text Tokenization and Stop-word removal

As seen in Figure 4.5, at first the text of the grouped news-headlines is tokenized and the stop-words are removed. That means for the example record the following transformation takes place.

Original text:

'Strong' demand outside North America propels Caterpillar to 13% quarterly profit growth Dow futures stage relief rally after Citigroup results Citigroup swings to quarterly loss of $5 billion, revenues fall 48%

After tokenization and stop-word removal:

Strong* demand * North America propels Caterpillar * * quarterly profit growth Dow futures stage relief rally * Citigroup results * swings * * loss * * billion * revenues fall

A '*' is placed in the above section wherever a word or a punctuation mark is removed to indicate the place of removal. Note that at the end of this transformation there are no punctuation marks left as well as no repeated words, no numbers like '13%', '$5', '48%' and no stop-words which include words like 'outside', 'to', 'after', 'of' in the above example.

4.6 Semantic Abstraction via Heuristic-Hypernym Modeling

4.6.1 Semantic Abstraction

Semantic Abstraction is a concept that is defined in this work with the following logical analysis.

The most common method in the past research for the preprocessing phase has been the so called 'Bag of Words'. In this method the news text is represented as a group of words. Each of the words is regarded as a feature. This method can be improved by creating a layer of abstraction (Schumaker & Chen, 2009).

Definition 4.1: Abstraction

Abstraction means having every word associated with a word of a higher order or generality i.e. a word that acts as a super-category for all subordinate words.

Abstraction can be perceived to have two main advantages:

1- It simplifies the feature space by reducing the number of words that are used as features.

2- It may help make similar conclusions from similar words by referring to them in the same way i.e. by referring to their category name.

For example, if we devise an abstraction system where synonymous verbs like “plunge” and “plummet” are categorized under “decrease”, then it is plausible to enjoy both of the above advantages. However, we argue that an abstraction system must suit the prediction philosophy of a prediction system. For example, if the abstraction system categorizes both of the words “green” and “red” under “color”, then the system must be designed in a way that does not require the two words to emit different signals to it. In other words, the two words often have symbolically opposite meanings in, say, “green light” and “red light” and if in a prediction system they are supposed to be emitting opposing signals, putting them in the same abstract category of “color” completely defeats the purpose. It is noteworthy that in the available abstraction systems, this aspect is usually not adequately analyzed and addressed. Nevertheless, past systems have reported positive impact although at times minimal (Schumaker & Chen, 2009). In general in the past research an abstraction system is devised and improved as follows.

4.6.2 Semantic Abstraction Methods

4.6.2.1 Named Entities

There are a number of approaches to improve the Bag of Words technique. One approach can be termed as ‘Noun Phrases’, whereby a lexicon is used to identify the part of speech of the words and sort out the nouns. Then a set of grammar rules allow the noun phrases around those nouns to be identified and extracted. For example, in the sentence ‘EUR/USD has started the new trading week quietly’, the word ‘week’ is a noun and the phrase ‘the new trading week’ is a noun phrase. In this method each identified noun phrase becomes a feature of the text which will receive a weight later whereas in Bag of Words (BoW) each word plays such a role, bare in mind nouns form

only a small portion of the words available in text and in BoW all those other words are also considered features . This introduction to Noun Phrases was necessary to pave the path for introduction of the next method. It should be noted that this next method is considered an existing form of semantic abstraction from the point of view of this research. The method is called “Named Entities”. It is created on top of the previously explained approach of Noun Phrases. Named Entities is considered an abstraction method because it categorizes the resulted nouns and noun phrases in the previous approach into certain predefined entity titles. The common framework for choosing the titles in this method is MUC-7. It is the last in the series of Message Understanding Conference Evaluations, funded mainly by Defense Advanced Research Projects Agency (DARPA). MUC-7 introduces a framework for named entity categorization. In short, the different systems participating in the conference are tasked with categorizing strings from news text about airplane crashes, and rocket or missile launches into predefined categories, namely, dates, location, money, organization, percentage, person and time. (Robert P. Schumaker & Chen, 2009) uses the MUC-7 categories and conducts a financial news article classification experiment for stock prediction and compares it with a plain Bag of Words approach as well as a Noun Phrase approach. It is determined as a result that semantic abstraction in form of Named Entities in the above manner does not produce significant improvement. However, the introduction of a new approach termed “Proper Nouns” does.

4.6.2.2 Proper Nouns

Proper Nouns is an abstraction method that is functionally devised between Noun Phrases and Named Entities. Proper Nouns are all the category titles that are available in Named Entities plus those nouns themselves which do not fall under one of the predefined Named Entities. This is a strategy to increase the number of the categories. In this way, terms such as NYSE, standing for New York Stock Exchange, in the text,

which cannot be automatically categorized under a named entity but may be important, will appear in Proper Noun representation as a new named entity. And therefore the number of the named entities increases to a lot more than just those defined by MUC-7. Schumaker and Chen (2009) report that in comparison to Named Entities, Proper Nouns performed better in terms of directional accuracy by a 1.2 percentage point. They conclude that the direction that they have taken, i.e. further abstraction, is correct. However, it requires refinement. They suggest that future research should evaluate increasing the number of entity categories. They attribute their results to Proper Nouns adequately using the article terms in a manner that was freer of the noise producing Noun Phrases and free of the constraining categories used by Named Entities.

Therefore, this work too, assumes that abstraction can have an impact on improving prediction accuracy. However, this work proposes a new method of abstraction and calls it "***Hueristic-Hypernyms Modeling***" or "***Heuristic-Hypernyms***" in short.

4.6.2.3 Hueristic-Hypernyms Modeling

Heuristic-Hypernyms Modeling is a novel semantic abstraction technique that is devised in this work to extract features. In this method words are still being assigned to categories similar to Named Entities and Proper Nouns methods, explained in the above two sections; however, a completely different approach is taken on what categories there are and what words are supposed to be kept and assigned to them.

This work argues that the above described Proper Nouns method can be challenged on several fronts.

First, it produces a heterogeneous set of named entities for category titles, whereby, a group of them are coming from MUC-7, like time, location etc. and a group of them are randomly generated based on the parsed news text like NYSE (New York Stock

Exchange) as mentioned in a previous example. An abstraction system that produces a homogenous set of category names can be expected to be cleaner of noise.

Second, in this method under NYSE, only NYSE is tagged. In other words, the named entity is not abstracted in terms of meaning and cannot accommodate other words with a very close or similar meaning. Hence, for such group of words no abstraction is applied at all.

Third, Proper Nouns is ignoring other Parts of Speech (PoS) that are intuitively important and meaningful like verbs.

This work proposes a hypernym-based approach to address the above 3 challenges which is named "Heuristic-Hypernyms Modeling".

There are two fundamental aspects to the proposed "Heuristic-Hypernyms Modeling" approach as its name implies:

1- Use of hypernyms

2- Heuristic selection of significant hypernyms

Each of these aspects is delved into in detail in a separate section in this chapter. These two sections are following.

4.6.3 Use of Hypernyms

Definition 4.2: Hypernym & Hyponym

In linguistics, a hyponym is a word or phrase whose semantic field is included within that of another word, its hypernym (Stede, 2000). In simpler terms, a hyponym shares a type-of relationship with its hypernym. For example, pigeon, crow, eagle and seagull are all hyponyms of bird (their hypernym); which, in turn, is a hyponym of animal. Computer science often terms this relationship an "is-a" relationship. For example, the

phrase “Blue is-a colour” can be used to describe the hyponymic relationship between blue and colour.

Hypernyms have proven to be effective in increasing classification accuracy in other areas before in the literature (Jeong & Myaeng, 2013) . How hypernyms are selected in this technique follows.

One place to look up the hypernym of a word is WordNet.

Definition 4.3: WordNet

WordNet is a lexical database for the English language. It groups English words into sets of synonyms called synsets, provides short, general definitions, and records the various semantic relations between these synonym sets (Miller, 1995). The purpose is twofold: to produce a combination of dictionary and thesaurus that is more intuitively usable, and to support automatic text analysis and artificial intelligence applications. The database and software tools have been released under a BSD style license and can be downloaded and used freely. The database can also be browsed online.

In the proposed system in this work, as illustrated in the algorithm in Figure 4.5, after removal of stop-words, each remaining token is looked up in WordNet Dictionary (Miller, 1995). If the exact word exists in the dictionary and it has a hypernym it is replaced with it. If there is more than 1 hypernym then the first listed hypernym is chosen. If the exact word is not available in WordNet dictionary or there are no hypernyms available, then the word is simply discarded from the potential feature list. In this way this approach for selection of words is contributing to tackling the problem of high dimensionality (Gracia et al., 2014) as well as to an additional objective, namely, semantic-abstraction.

Table 4.4 lists the looked up hypernyms for the example at hand.

Table 4.4 Example for a Heuristic-Hypernym Model

Exact words after stop-word removal	Hypernyms
America	N/A
Caterpillar	larva
Citigroup	N/A
Dow	N/A
North	cardinal compass point
Strong	N/A
billion	large integer
demand	request
fall	season
futures	N/A
growth	organic process
loss	transferred property
profit	income
propels	N/A
quarterly	series
rally	gathering
relief	comfort
results	N/A
revenues	N/A
stage	time period
swings	N/A

Sometimes the hypernyms chosen, do not make any sense in terms of meaning for example in Table 4.4, 'larva' is the hypernym that is determined by the system for the word 'Caterpillar', however, in this context the word 'Caterpillar' is merely a company name and has nothing to do with the biological hypernym 'larva' in terms of meaning. It is noted that this may logically introduce some noise or at least an unreliable association, however; on the whole such a case of absolute irrelevance, as seen in the above example set, lies in the minority. Moreover, in this system the contextual meaning is less significant than a statistical place-holder i.e. the word 'larva' may very well constitute a somewhat unique representation of the word 'Caterpillar' in the context of financial news and thereby still allow the system to determine statistical significance, if any at all, towards the ultimate goal of the system that is of a statistical-pattern-recognition nature.

Another example for such “statistical place-holder” assumption comes into play in the case of a polysemous word like “fall” which could mean “drop” or be a “season”. And the system as shown in Table 4.4 chooses the hypernym “season”. If we assume that a word like “fall” in context of the financial markets predominantly means “drop”; and the current system replaces the word with “season” which is the hypernym for the other meaning of “fall”; then “season” can play the role of a statistical place-holder. In other words, the impact of the word “fall” in the system is assumed to be captured by the word “season” which is literally not the correct hypernym meaning-wise but because every instance of “fall” is replaced with it, it can be assumed to play the same statistical role. However, it should be conceded to, that an improvement on this aspect of the system may lead to even better results. This problem is referred to in the literature as “disambiguation” and is a challenging research topic (Hodson & Zhang, 2014; Kulkarni, Agarwal, Shah, Rathod, & Ramakrishnan, 2014; Lipczak, Koushkestani, & Milios, 2014; C. Wu, Lu, & Zhou, 2014). It is really not easy to choose the correct meaning for a word based on its context and therefore the proposed solution despite its imperfection is reasonable; as it is in essence providing a way around “disambiguation” with the above described “place-holder” assumption. However, this aspect is yet to be enhanced in the future.

Additionally as shown in Table 4.4 some words are not available in WordNet dictionary like ‘Citigroup’, ‘Dow’ and some words do not have any hypernyms like ‘America’ or ‘Strong’ and hence are marked as Not Available(N/A) in the table. Furthermore, words like ‘futures’, ‘propels’, ‘results’, ’revenues’, ’swings’ that end with an ‘s’ are also discarded if the form with the ‘s’ does not possess an entry in the dictionary. The same goes for words ending in ‘ed’, ‘ing’, etc. This is the heuristic aspect of this technique. The logic behind this heuristic mechanism, whereby only exact dictionary entries are kept, is described in the next section.

4.6.4 Heuristic selection

Heuristic selection of hypernyms is the mechanism that is proposed whereby only the hypernyms of those words are selected which possess an exact WordNet entry.

Some words like ‘futures’, ‘propels’, ‘results’, ’revenues’, ’swings’ are not available in their exact form as a WordNet entry but are available as their stemmed forms which is the form without the ‘s’ at the end in the above examples. However, these words are being discarded in this system based on the assumption that words that end in an ‘s’ are either plural nouns or third person singular verbs. In the literature, it is indicated that in general parts of speech of noun and verb carry less sentimental or subjective value than adjectives and adverbs (Esuli & Sebastiani, 2006). This makes intuitive sense as sentimental content is mostly captured and expressed in more emotionally descriptive words namely adjectives and adverbs.

We have tested the system after elimination of all words of the kind ‘noun’ or ‘verb’ or both, however, our tests show best results when only words like the above that are ending in an ‘s’ or other non-exact matches like words ending in an ‘ed’ or ‘ing’ are eliminated.

This makes logical sense, as blanket removal of words based on a part-of-speech may result in removal of some words unnecessarily. Words can have more than one part of speech for example the third form of a verb can be referred to as an adjective as well. For instance, the words ‘weakened’ or ‘diminished’ are adjectives but once stemmed they go to their verb form of ‘weaken’ and ‘diminish’ and can be eliminated if all verbs are removed as a part of speech category. However, they both can remain in the proposed mechanism as both ‘weakened’ and ‘diminished’ have their own entries in WordNet.

Furthermore, in case of a blanket elimination of nouns, words like ‘growth’ and ‘loss’ are also discarded which are potentially valuable for the system. Hence, logically and experimentally blanket-removals are not useful.

Experiments (Section 5.6) show that the choice of a word by the system based on the existence of an exact match in WordNet reveals best results. In order to test these assumptions, the feature-vector is once more created but this time the words are stemmed first and then their hypernyms are looked up in WordNet. Note that in the proposed system the words are not stemmed and the exact words are passed to be looked up in WordNet.

This experiment (Section 5.6) proves the above assumptions and demonstrates that the existence of an exact entry for a word in WordNet is indicative of some value. Note that words that have similar endings but are not in possession of an entry in WordNet, are discarded and this heuristic seems to strike a meaningful balance in the logic of feature selection and reduction.

Furthermore, Heuristic-Hypernyms Modelling addresses the 3 challenges mentioned in Section 4.6.2.3 that are faced by other methods as follows: Firstly, it is able to provide a homogenous set for category titles as they are all hypernyms of words. Secondly, words like NYSE are simply dropped if they do not have a hypernym in WordNet, thereby no words can be found among the category titles that is not abstracted, as opposed to the previous methods. Thirdly, other Parts of Speech (PoS) are not ignored as they are now processed too and can now have category titles using their hypernyms.

Once the choice of hypernyms per record is completed, an initial set of features for the feature-vector is actually determined. Next, the features are weighted by a combination of two different metrics as explained in the next two sections in order to integrate sentiment content.

4.7 Sentiment Integration

Once features are determined in the feature space, it is crucial to realize that they have different levels of impact. In other words, each word in a piece of text certainly has a different degree of significance in the context of the classification decision that is targeted to be made. In the view of this research there are at least 2 aspects that are vital to be taken note of:

1- Sentiment Load

2- Frequency in document and frequency in corpus

Each of the above aspects and how they are implemented in the proposed model is detailed in the following two sections.

4.7.1 Sentiment Integration via SentiWordNet SumScore Weighting

Each word (or in this case hypernym) that is considered a feature in the designed feature space carries a sentiment value. Many researchers consider the sentiment value on a negative to positive spectrum. One of the most prominent works of research in this area that provides a valuable sentiment dictionary is SentiWordNet.

The TF-IDF weighted features are scaled by another weighting value next, namely, the SentiWordNet SumScore that is a new score defined as a part of this work.

Definition 4.4: SentiWordNet

SentiWordNet is a dictionary of sentiment values (Baccianella & Sebastiani, 2010) that contains a Positivity Score (PosScore) and/or a Negativity Score (NegScore) between 0 and 1 as well as an Objectivity Score (ObjScore) for each WordNet entry i.e. synset.

Most researchers prefer to differentiate between a negative sentiment and a positive sentiment, for example, if they are separating negative movie reviews from positive ones.

However, in this research a different perspective is taken as the context that is investigated is significantly different from movie-reviews in the following manner.

In the context of classification of review-text it is clear for the researcher what text is considered positive or negative and that is directly related to the direction of the sentiment captured in each word or feature. However, in the context of mining of news article headlines, the negativity or the positivity of words does not matter per se as it is not known what exactly the subject of discussion in the text is and how exactly that subject may have an impact on market movements. For instance, a positively worded article about the value of the Euro can have the opposite impact on the Euro/USD currency pair value compared to a positively worded article about the value of USD. Bear in mind that both articles are positively worded but, as one can see in this example, the identification of this mere fact is not useful in this context.

Therefore, the solution that this work proposes to the above dilemma is to consider the sentiment without attention to its direction. It is assumed that there is a connection between how much sentiment exists in text (regardless of its positivity or negativity) and the impact it has on the market. In other words, if there is excitement or disappointment in the market there is an impact. And the algorithm should not care which feeling is there as the excitement of some may be the disappointment of others. Therefore, the work is not targeting to identify which feeling exists but rather to identify whether the presence of feelings, emotions and the total amount thereof can be used as an indicator for market movements. This work demonstrates with its experiments that the proposed model is helpful and that it is a viable solution for this context.

Hence, what is needed to be measured is the emotional or sentimental load or charge of each feature. However, there is not a ready-made sentiment score for this purpose. That

is why this work proposes a new sentiment value by the name of SumScore. But first, let's see what sentiment scores exist in SentiWordNet.

There are 3 sentiment scores available in SentiWordNet:

1- Negativity Score or NegScore (between 0 to 1)
2- Positivity Score or PosScore (between 0 to 1)
3- Objectivity Score of ObjScore

The NegScore and PosScore are independent from each other and are calculated automatically (Baccianella & Sebastiani, 2010). They are directional values and due to the reasoning presented in the above, are not suitable for the intention of this work on their own.

The only value that is available that is not directional is the ObjScore, however, it is targeting exactly the opposite of what is required in this context. ObjScore measures the lack of emotions or sentiment, whereas what is needed is a score to measure total presence of sentiment.

Objectivity Score (ObjScore) is defined as follows:

$$O_i = 1 - (P_i + N_i) \qquad (4.3)$$

Where O_i is the ObjScore of word i which is the result of subtracting the sum of P_i or PosScore of word i and N_i or NegScore of word i from 1.

What is proposed in this work is a novel measure by the name of SumScore with the below definition:

$$S_i = P_i + N_i \qquad (4.4)$$

Where S_i is the SumScore of word i which is the result of summation of P_i or PosScore of word i and N_i or NegScore of word i. In other words SumScore of a word can be calculated from its ObjScore as below:

$$S_i = 1 - O_i \qquad (4.5)$$

Where O_i is the ObjScore of word i. This is important to realize that SumScore is inclusive of both positive (PosScore) and negative (NegScore) indications of emotion. It is measuring total existence of sentiment. Whereas ObjScore is intended to indicate objectivity or lack of any sentiment or emotional charge, but as it turns out it is more effective to use a measure that is indicative of existence of sentiment i.e. SumScore rather than a measure that is indicative of its absence i.e. ObjScore. And rightly so as absence or non-existence of a phenomenon is hardly measurable from a logical stand point. What is measured is existence. In the same way that in measurement of temperature what is measured is the existence of heat and not lack thereof i.e. coldness. Coldness is merely the absence of heat; hence it does not have its own existence and therefore cannot be measured.

As explained before, it is important to note that measuring one direction of emotions alone i.e. positive or negative does not make much sense as a market-participant may feel positive or negative about any given market direction. However, it does make sense to anticipate market activity when the total amount of emotions regardless of direction i.e. the value of SumScore tends to change.

In an experiment in Section 5.7, it is demonstrated that the proposed SumScore has significant positive impact on the accuracy of the prediction results. It outperforms ObjScore as well as NegScore and PosScore per se. SumScore definition and usage design under the title of “Sentiment Integration” is one of the contributions of this work.

However, as mentioned before the role of frequency of features shall not be ignored either. How the frequency is considered in characterizing of each feature in addition to the calculation of SumScore is described in the next section.

4.7.2 Frequency Integration via TF-IDF Weighting

As explained in the beginning of Section 4.7, the second item that must be considered while determining the level of impact of a feature has to do with its frequency of appearance. A word, and therefore the feature representing it, can have at least two types of frequency of appearance:

1- Frequency within a document
2- Frequency within a corpus

Frequency within a document is basically the number of times a term or a feature appears within a document.

Definition 4.5: Document

A document is basically the chunk of text that is referred to as one unit. A document possesses one record in the database. It can be on piece of text like one article or one page. It can also be a number of them bundled together, for instance, a number of pages of a text or an entire book or a number of articles grouped together. In the case of this work the document is a number of headlines that appear in a certain 2-hour time interval.

Frequency within a document is commonly termed as ‘Term-Frequency’ or ‘TF’ in short. If a feature appears in a document for 5 times the then TF = 5.

Frequency within a corpus can be defined in two ways. One can look at the total number of appearance of a term (feature) in the entire corpus. Or one can look at the number of documents within a corpus that appear to contain the desired term (feature).

Definition 4.6: Corpus

The term corpus refers to the entire text repository that is accessible to the system and is subject of text mining. Hence, in the case of this research, it is all the news headlines which are available to the system. In other words, the corpus contains all the documents as defined in definition 4.5.

The latter type of frequency in the corpus whereby the number of documents containing the desired term is counted is called ‘Document Frequency’ or ‘DF’ in short. DF is the form of frequency in corpus that is commonly used and is also used in this research in the following manner.

TF somewhat determines how significant a term (feature) is to a document. However, if the same term (feature) is frequent in too many documents in the corpus then it does not have much differentiating value. That is when DF comes into play; because DF is the number of the documents which contain that term. Therefore, if DF is high although TF is high the term is not significant. To represent this mathematically DF is inversed. Because, in that form the higher the DF the lower the Inverse DF. Hence, Inverse DF is the value that is supposed to be placed in the equation. Inverse DF or ‘Inverse Document Frequency’ is termed in short as ‘IDF’.

Therefore, in order to create a value that takes frequency into account in a reasonable way TF times IDF simply written as ‘TF-IDF’ is used.

Definition 4.7: TF-IDF

Term Frequency–Inverse Document Frequency (TF-IDF) is a standard numerical statistic that reflects how important a word is to a document in a collection or corpus. It is often used as a weighting factor in information retrieval and text mining. The TF-IDF value increases proportionally to the number of times a word appears in the document, but is offset by the frequency of the word in the corpus, which helps to control for the fact that some words are generally more common than others.

Although TF-IDF is commonly used in text mining research, it is vital for this work to form a comprehensive understanding of it as explained above in order to correctly place it as a component of the bigger weighting formula which is proposed in this work.

4.7.3 Proposed Weighting Model (i.e. TF-IDF*SumScore)

So far, it has been explained how 2 weights can be calculated for each feature in the feature-vector of each record in the feature-space, namely:

1- SumScore (Proposed in this work)

2- TF-IDF (Commonly used in text mining research)

The overall formula that this work is proposing for an accurate weighting of a feature that considers both sentiment and frequency in the correct proportions is:

S_o = TF-IDF * SumScore (4.6)

S_o refers to the Overall Sentiment Score of a feature. It basically helps magnify or scale the SumScore, which is the pure sentiment score, based on frequency characteristics of the corpus at hand.

In other words, SumScore enables the system to integrate sentiment value based on what is achieved in the SentiWordNet project and its multiplication by TF-IDF ensures that it is compatible with the terms or features that are existing in the documents at hand as well as the spread of their appearance at the level of the corpus that the system is trying to analyze. In the experiment Section 5.7, it is demonstrated how exactly the proposed model of weighting based on sentiment is positively contributing to the overall classification accuracy improvement of the proposed system.

Furthermore, it is crucial to realize that SumScore contributes to tackling the problem of high dimensionality as well. This occurs in the following manner.

Not all terms (features) have a value for their SumScore. There are many words which have neither a positive nor a negative sentiment associated with them in the SentiWordNet dictionary. Such words have zero for the value of their SumScore as a result. The existence of zero in the weighting calculation that produces S_o as described above, leads to a zero value for the weight of that feature. A feature with a weight of zero is basically canceled out as it has no impact. Thereby in such cases the sentiment weight is helping to reduce the number of features (dimensions) and in this way it is contributing to tackling the problem of high dimensionality in addition to its main role of weighting.

At this stage, as seen in the flowchart in Figure 4.5, a TF-IDF*SumScore weighted feature-vector is ready and stored for each record. The TF-IDF weighted feature-vector is also stored separately for an accessory adjustment purpose that is explained in Section 4.9.1.

4.8 Synchronous Targeted Feature-Reduction

So far a comprehensive feature space is defined in form of a feature-matrix. First, each feature is extracted based on the Heuristic-Hypernym of each word. Then each feature receives a weight that integrates their sentiment load in an adjusted manner for frequency. In addition to their main responsibilities, both of the above layers help to reduce the number of dimensions in the feature-space in their own right. The former layer does this by eliminating words which have no Heuristic-Hypernyms and therefore can produce no features and thereby reduces the number of features. And the latter layer by elimination of features based on multiplication by a zero value of sentiment weight for features with no sentiment value or a SumScore of zero.

However, even after the above two layers of feature-reduction, there are still too many features available in the feature-space and this has a serious negative impact on the

classification accuracy. Nevertheless, past works in literature are using such big feature-spaces as input to their machine learning algorithms. This work suggests a new approach that follows.

This work proposes a further layer of feature-reduction via a new algorithm that is termed in this work as 'Synchronous Targeted Feature-Reduction' or sometimes simply referred to as 'Targeted Feature-Reduction'. Devising and utilizing this algorithm-layer is another main contribution of this work and is explained in detail in the below.

In short, the common approach for training and model creation in the literature (Hagenau et al., 2013; Schumaker et al., 2012; Y. Yu et al., 2013) is some variation of the below steps:

Step 1: Take a feature-matrix as input.

Step 2: Usually conduct no further feature-reduction or sometimes reduce the features to a top random number, say, 100 or 200 features after sorting by a predefined weight.

Step 3: Build a model based on part of the records called the training-set.

Step 4: Use the above built model to predict other records.

There are at least two main problems with the above approach:

Problem 1: If there is no effective feature-reduction, there are too many features available in the feature-vector which leads to the curse of high dimensionality (Pestov, 2013).

Problem 2: If there is a feature-reduction method, for example choosing the top features according to a criterion, then the reduction is somewhat random as it is in no special way optimized for the record(s) to be predicted.

"Synchronous Targeted Feature-Reduction" solves the above two issues effectively and increases the results significantly; as later shown in the experimentation chapter of this thesis in Section 5.8. It proposes the below flow of steps:

Step 1: Take a feature-matrix as input.

Step 2: Take a single record whose label is to be predicted.

Step 3: Reduce all the features in the feature-space to only those with a value in this record and create a new feature-space thereof. In other words:

SELECT the columns of the initial feature-matrix

WHERE the value of the features in the targeted-record (or the record whose value is to be predicted) is non-zero, and

CREATE a new feature-matrix thereof.

The word Synchronous in the name is referring to this synchronous feature-matrix table creation; which is happening as the system is being run on the records to be predicted.

Step 4: Build a model based on all the other records available for training.

Step 5: Run the single record chosen in step 2 through the created model in step 4 in order to predict its label.

In the above, steps 1 to 3 summarize the proposed feature-reduction method based on the targeted-record for prediction which is termed Synchronous Targeted Feature-Reduction (STFR). STFR basically reduces all the features to those only that are available in the record that is targeted for prediction.

4.9 Model Creation and Prediction

In steps 4 and 5 a model is created and used to predict the label for the targeted record accordingly. These two steps are essentially a part of the next stage that is discussed under Section 4.10, titled 'Machine Learning'. It is important to note that STFR presides on absolute optimization of feature-reduction by reducing the features to the minimum that is needed for the one prediction task at hand and creates a new model per prediction. In other words it creates new models synchronously and just in time as the prediction needs to happen. In the experiments section execution-time has been observed and the net-advantage of the proposed method is far greater than the extra few seconds needed for model-creation per prediction. The total of all steps from 1 to 5 summarize the entire label prediction activity that includes STFR; and can be termed as Synchronous Targeted Label Prediction or STLP. Table 4.5 details STLP further as a method.

Table 4.5 Pseudo-code for Synchronous Targeted Label Prediction (STLP)

Method: Synchronous Targeted Label Prediction (STLP)
Input: TF-IDF*SumScore Feature Matrix (M)
Intermediary Output: Reduced Matrix Based on K (R)
Overall Output: Predicted Label (LABEL)

```
*INITIALIZATION*
M_{i,j} = TF-IDF*SumScore Feature Matrix upto K
K = Index of last record of M_{i,j} i.e. Prediction Target
*RUN Synchronous Targeted Feature Reduction (STFR)*
T = M_{K, J}  (Targeted/K^th record)
R_{i,j} = M_{i,j}
FOR T_i FROM i=1 TO i=J
  IF Ti == 0
    DROP COLUMN Column_i FROM R_{K,J}
  END-IF
END-FOR
*GENERATE  Synchronous Model & RUN it*
TRAINING = SELECT * FROM R WHERE ID != K
TARGET = SELECT * FROM R WHERE ID == K
MODEL = GENERATE_MODEL(TRAINING)
LABEL= RUN_MODEL(TARGET)
RETURN LABEL
```

This approach reduces the features effectively to a number that is many times smaller as it takes into account features from a single record only. It produces very good results in the context of this work and makes a lot of logical sense in terms of efficiency and accuracy in feature-reduction. This is experimentally proven in this work and the results are provided in Section 5.8 of this thesis accordingly.

4.9.1 Adjustment Algorithm for Occasional Empty Vectors

As the primary input feature-vector in this work is the TF-IDF*SumScore feature-vector, there are some rare instances whereby a selected record to be predicted is of no feature with a non-zero TF-IDF*SumScore value which is caused by the SumScore being zero for too many features in that record. Hence, in such a situation no features can be determined via the TF-IDF*SumScore values as there are none. To address this predicament, if such a record is encountered the system will dismiss the TF-IDF*SumScore feature-vector and use the TF-IDF feature vector as input for feature-reduction and model creation for that occasional record instead. This is illustrated in the flow chart in Figure 4.6 and this decision is made at the decision point where this question is asked: *"Are there any non-zero attributes?"*

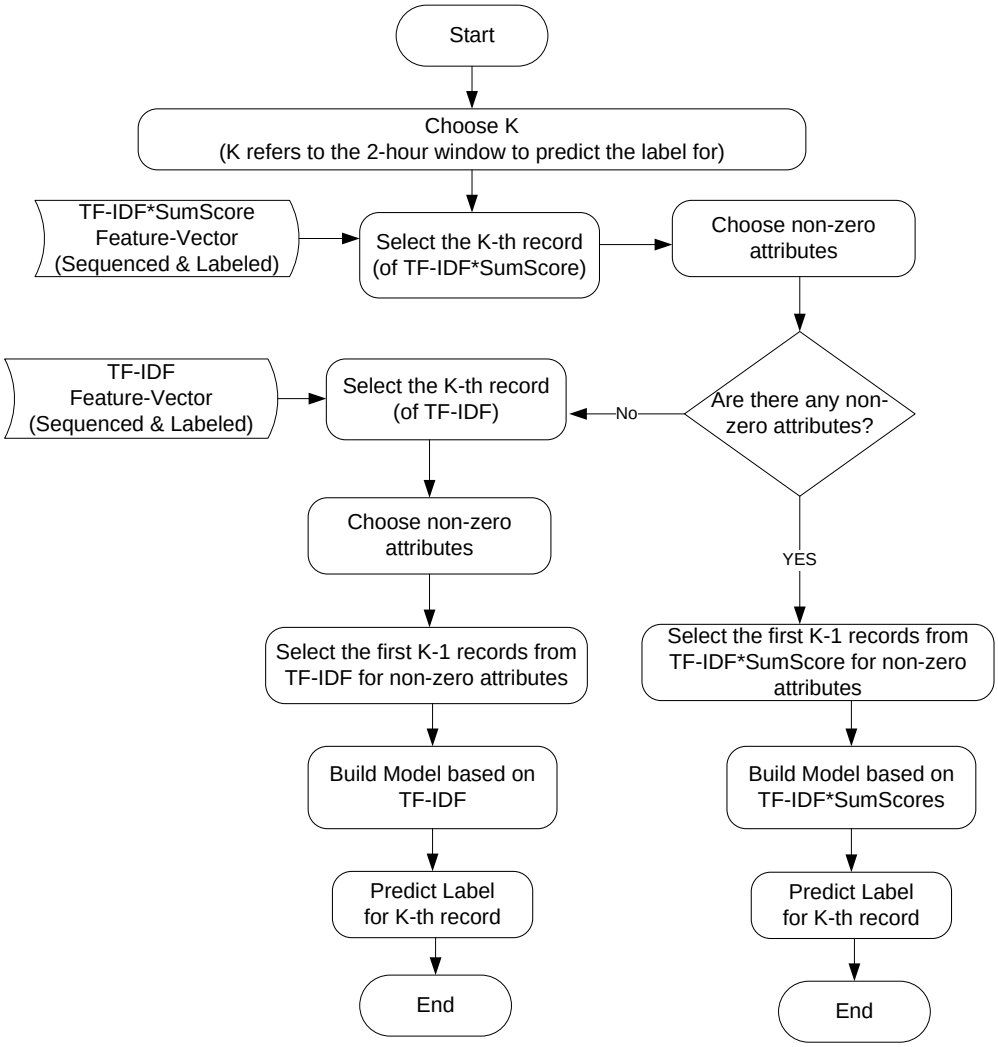

Figure 4.6 Flow of Synchronous Targeted Feature-Reduction, Model-Creation and Prediction

Note that the input feature-vector that is used in this system is chronologically ordered and hence, the K^{th} record which is the record to be predicted is always the last record and the K-1 records before it are the records used for training.

4.10 Machine Learning

Pre-processing is considered in this work to be the stage that prepares a feature-matrix that is ready to be handed over to the next stage which is called 'Machine Learning'; whereby a machine learning algorithm is used to create (learn) a model based on a training dataset (training feature-matrix) and make predictions using the model for new records (testing data-set).

However, as seen in the previous section, the proposed model for Synchronous Targeted Feature Reduction is designed in a way that facilitates an intertwined synchronous collaboration between these two segments of the system, namely, STFR and Machine Learning. The combination of which is termed as STLP in the previous section.

In essence based on the record that is targeted for prediction, a feature-matrix is created which entails only relevant features to that record. Then a model is made to conduct that specific prediction. The next record that is up for prediction has another number of features and accordingly a feature-matrix is created based on those. The new feature-matrix is then used for training a model that enables the prediction for this new record; and so on and so forth.

It is worth noting that from a system design perspective, all that is required by the system to be performed to produce one specific feature-matrix for one specific record is a selection of columns from the overall feature-matrix, that contains all available features in the corpus and is already built and weighted in the first two layers of the Multi-layer Feature Reduction algorithm as described previously.

4.10.1 Choice of Machine Learning Algorithm

In order to choose the right machine learning algorithm for the proposed system, 2 main perspectives are considered:

1- The suggestions made in the literature regarding the capabilities of given machine learning algorithms in different contexts.
2- Results of the experiments conducted on multiple machine learning algorithms in this work, namely Naïve Bayes, K-Nearest Neighbors, Support Vector Machines. The results are to be found in Section 5.9 of the chapter on experiments.

As a result a standard Support Vector Machines (SVM) algorithm (Cortes & Vapnik, 1995) is chosen to be the suggested machine learning algorithm in this work for model creation and label prediction. However, due to the modular design and implementation of the system, replacing the current machine-learning algorithm with another one is easily feasible and is not intrusive to the flow and design of the system.

4.10.2 Brief overview of SVM

Support Vector Machine is the most common machine-learning algorithm in the literature for similar classification problems (Fung, Yu, & Lam, 2002; Mittermayer, 2004; Soni et al., 2007). SVM is a supervised learning model for classification. It generally outperforms other models like neural networks in financial time series prediction (K.-j. Kim, 2003; Tay & Cao, 2001). This is attributed to its ability for structural risk minimization, where multiple local minima can be avoided (Premanode & Toumazou, 2013). Moreover, the computational complexity of SVM does not depend on the dimensionality of the input space (Olson & Delen, 2008). It is a binary classifier, which means the input is classified in one of two categories as output. With an SVM model the representation of each news-group is mapped as a point in a feature-space where the examples of one category are separated by a line or a curve with as much margin as possible from the other category.

4.11 Evaluation Phase

Once the predictions are made the results are analyzed and interpreted in an evaluation phase. In this phase it is determined how accurately predictions are made. There are a number of evaluation measures that are calculated in this phase, namely:

1- Accuracy

2- Precision

3- Recall

4- F-Measure

What these measures are and how they are calculated is described in Section 2.9 in the chapter on literature review.

In short, note that in addition to Accuracy, which is the primary evaluation measure, Precision and Recall are also reported for each of the classes (N and P). They provide more insights into the relevance of results and are defined as below:

Precision is the fraction of predicted instances that are true for a class *C*. High precision means that an algorithm returns substantially more true results than false for a class *C*.

$$Precison\ (C) = \frac{True(C) \cap Predicted(C)}{Predicted(C)} \quad (4.7)$$

Recall is the fraction of true instances of a class *C* that are predicted. High recall means that an algorithm returns most of the true results for a class *C*.

$$Recall\ (C) = \frac{True(C) \cap Predicted\ (C)}{True(C)} \quad (4.8)$$

The results for evaluation measures vary from experiment to experiment based on a number of variables and the size of the sample that is being observed. Each experiment and the relevant evaluation results are presented in the experiments chapter (Chapter 5) in a separate section.

As described in detail in a chapter 5, the overall performance of the proposed system peaks at 83.33% in terms of accuracy, which is significant, compared to other systems in the literature. In comparison with the accuracy of chance, which is assumed to be at 50% for a binary classification problem, the performance of the proposed system is obviously noteworthy.

Furthermore, the experiments are designed in a way that facilitates interpretation of the amount of contribution made to prediction-results based on each of the layers of the proposed Multi-layer algorithm.

4.12 Chapter Summary

In this chapter the proposed text mining solution is presented. The objective of the solution is to take news-headlines as input and make a binary prediction for movement of a currency pair (Euro/USD) in the Foreign Exchange Market, which is the financial market for currencies. The news-headlines are collected within 2-hour time-intervals and the prediction is made for a point in time one hour after the end of the collection interval.

In essence the core challenge at hand is a binary classification problem of textual content. However, it differentiates itself from other textual classification problems, say, a movie-review classification into good or bad reviews, or spam filtering of email into good emails and spam-emails in that it is not a known fact what news-headlines drive the market up and which ones drive it down.

Therefore, the system needs to learn from the data at hand during training and recognize patterns that are not clearly known. However, once the performance of the system is evaluated, it becomes clear if the system is contributing to increasing the classification accuracy from a 50% probability of chance in a binary decision.

Therefore a prediction system that can achieve this objective is not only automating what could have been performed by humans at a slower rate, similar to the case of spam-filtering, but achieves a classification that would not have otherwise been possible by humans. Because, humans can simply not recognize patterns when too many factors are involved and the context is too complex.

In terms of design the high level flow that is pursued in this work for text mining is presented in Section 4.2 as:

1- Pre-processing
2- Machine Learning
3- Evaluation

Pre-processing in this work encompasses every activity that is carried out before the existence of a feature-matrix that can be handed over to a machine-learning algorithm. In the view of this research, this is the segment of work that differentiates text-mining from data-mining in essence as the statistical pattern recognition algorithms are relatively similar. In Pre-processing 3 major scopes of work are placed:

1- Data Retrieval
2- Input-Data Preparation
3- Feature-Matrix Creation

Data Retrieval is explained in Section 4.3. It can also be considered as a completely separate activity, however, for the sake of clarity of the model at a higher level it has been bundled in Pre-processing as it is after all among the initial activities that are carried out. The section details how the textual data and the market data that is used by the system is retrieved and from which sources.

Input-Data Preparation (Section 4.4) contains all the activities that are required to be carried out on the textual data that is retrieved so that it is organized and has all the required elements. The output of this section is a clear set of text documents, which are associated with a release date and time and are mapped accordingly onto the retrieved market data.

Feature-Matrix Creation includes all the activities that happen on the above prepared data source so that eventually a machine-learning algorithm can be engaged in the next major stage. In essence there are 3 activities that take place here:

1- Text Tokenization
2- Stop-word removal
3- Multi-layer Dimension Reduction with Semantics and Sentiment

The first 2 activities are described in Section 4.5 and are relatively standard. The core of the discussion in Feature-Matrix Creation is with regards to part number 3 above which constitute a major contribution of this research in terms of algorithms. Part 3 above refers to a Multi-layer algorithm that is devised in this work to address a number of challenges. The 3 main layers of the algorithm are explained in detail and are:

1- Semantic Abstraction via Heuristic-Hypernyms (Section 4.6)
2- Sentiment Integration via SumScore Sentiment Weighting (Section 4.7)
3- Targeted Feature Reduction (Section 4.8)

Each of the above layers deals with distinct problems via a proposed algorithmic solution in this work.

The result of all of the above is an appropriate feature-matrix that can be utilized by a machine-learning algorithm for successful textual classification. The machine-learning algorithm is explored in Section 4.9. It is followed by an overview of the last stage, which deals with evaluation in Section 4.10. In the evaluation section in this chapter the high level design of the evaluation is briefly discussed. This is picked up and enhanced, by delving into details and presenting the exact results of all the experiments in this work, in chapter 5 of this thesis that is devoted to experiments and results.

5 Experimental Results and Analysis

5.1 Introduction

The system proposed in this work classifies textual content of bundles of news-headlines that are released in 2 hour intervals, into two classes of Positive or Negative based on the impact that occurs on the price of a currency pair namely Euro/USD in 1 hour after the end of the interval.

This work proposes an end-to-end solution that takes news-headlines as input at one end and makes predictions as output at the other end. This act simply puts to test the viability of market prediction based on text mining of news. As it produces positive results, it strongly indicates that a relationship between textual news and market movements exists.

Beyond the overall conclusion of existence of a relationship between news-headlines and movements of a currency pair in the Foreign Exchange Market, this work identifies the significance of customized text-mining enhancements.

This work at the core of its proposed system proposes a multi-layer algorithm that tackles specific text-mining challenges.

The multi-layer algorithm is primarily tasked with dimension-reduction as high-dimensionality is a key challenge for machine learning in text-mining, specifically when the text corpus is as diverse as the news-headlines released over multiple years. The two further challenges that are identified and addressed in their layers accordingly are semantic abstraction and sentiment integration.

In order to determine the contribution of each layer, an experiment is designed to evaluate the performance of the system with and without the existence of that layer and

compare the two. Each experiment verifies that the layer subject to it is significant to the objectives of the system and the achieved accuracy.

In the rest of this chapter: First, a comprehensive description of the datasets used for these experiments is presented. Next, the experimental design is amplified followed by a detailed description and results section for each experiment. Then, the final results of this work are reiterated. At the end, a chapter summary is provided to conclude the chapter.

5.2 Datasets (&execution time)

It is important to note that the presented datasets are gathered and produced for this work and is one of its additional contributions. The reason for producing a dataset to use is that there is simply none found that could fulfill the requirements of what is intended to be achieved in the specialized context of this research and is accessible for use by the researcher. This is a challenge that is overcome by the gathered and consolidated dataset.

In essence there are two sources of historic data that are required for the proposed system each of which have their own dataset, plus a consolidation of the two. Here is a brief list of the datasets:

1- News-headlines over a number of years

2- Currency pair (Euro/USD) price movement over the same period of years

3- Consolidation of the above two datasets into one

The first two sources alone are not enough for the purpose of this work until they are consolidated. Therefore, a 3rd dataset is produced which is basically the consolidation of the first two and it is the dataset that is directly used in the experiments.

Each of the above 3 is discussed in detail in a separate sub-section that follows.

5.2.1 News-Headlines Dataset

The News-Headline dataset has 3 pieces of information in it for each record: The news-headline, the date and the time of release. The latter two, form a date and time column. Table 6.1 contains 3 records from this dataset as an example. The news-headlines dataset contains 1307 records in total.

Table 5.1 News-Headlines Dataset Example

News Headline	News Date & Time (GMT)
'Strong' demand outside North America propels Caterpillar to 13% quarterly profit growth	18/4/2008 11:37:05
Dow futures stage relief rally after Citigroup results	18/4/2008 11:01:27
Citigroup swings to quarterly loss of $5 billion, revenues fall 48%	18/4/2008 10:57:23

The source of the data is historic news-headlines from MarketWatch.com, which is a popular and significant online source of financial news with a special focus on interesting topics regarding financial markets.

There are multiple categories available on the website. The chosen category for this dataset is the breaking news. The reason behind using the breaking news is to obtain the news that is not too frequent and of enough significance to be considered a piece of impactful information. This is a measure put in place to avoid unnecessary noise by receiving too many pieces of news of a random nature. Furthermore, this logically is what a human trader looks at as well.

Moreover, only the news headlines are collected. Every news article has a headline and a body. The headline is a good summary of the article with very few words. It is safe to assume most of the impactful key words do appear in the headline. Most of research in the literature looks at entire bodies of text. However, usually their objective is different from the objective of this work. Analyzing the body is a more appealing option when

one makes an effort to identify differences between two pieces of text in terms of their sentiment or subtopics. For example, it can be considered while working with product reviews, etc. In the case of this work, each piece of breaking news is highly likely to address one specific event that potential has an impact on the market and that event is revealed in the topic so that the readers know what the article is targeting to address. A mention of the event or things and places associated with it, is all what this system requires. Of course, the reason behind the choice of headlines is also to avoid noise. If entire article bodies are considered there are just too many redundant words in the feature-space which do not contribute to the objective of the system and can heavily affect the results negatively and reduce the accuracy. Moreover, the more words involved, the more time-consuming the processing becomes and the slower things get carried out, besides the increase in need of processing power which can easily surge.

The span of time for which the news-headlines are collected is over multiple years; to be exact, from 02/04/2008 at 09:12:26 GMT to 18/9/2012 at 11:33:27 GMT. The limitation on the covered time spam is imposed by the availability of historic cache that the retrieval mechanism accessed to prepare this experimental dataset. The retrieval mechanism, namely, accesses Google cache of the RSS releases of MarketWatch.com to accumulate the dataset and the depth of drilling possible in historic data was the above date at the time the effort was made. However, in a real world system the data is being constantly collected and accumulated and the system maintains its own database of news-headlines to which it can also be added from other possible sources. But for the sake of the experiments in this work the above time spam for which the headlines were successfully collected is sufficient. As a matter of fact a similar dataset is not found elsewhere and this collection enabled this research to be carried forward and is one additional contribution of this work.

5.2.2 Currency-Pair Prices Dataset

The price dataset is retrieved for 2-hour intervals from an online broker by the name of FXCM via their desktop client application called FXCMMicro. The 2 hour length is chosen because in the literature 1 to 3 hours is mentioned as a good time period to observe impact of news in markets. Furthermore, if the chosen length for the time-interval is too short the chances of having any breaking news within it small.

The same time span as for the news dataset is considered here for the currency-pair dataset collection. Therefore, the dataset covers from the interval for 02/04/2008 09:00:00 GMT to the interval for 18/09/2012 11:00:00 GMT.

When an interval is referred to by 02/04/2008 09:00:00 GMT, for example, it is the interval that covers the span from 09:00:00 to 11:00:00 on that date and so on. A number of records of this dataset are presented in Table 5.2 as examples.

Table 5.2 Examples of Currency-Pair Dataset Records

Date & Time	Open, Ask	High, Ask	Low, Ask	Close, Ask	Open, Bid	High, Bid	Low, Bid	Close, Bid
18/04/2008 15:00:00	1.58041	1.58159	1.57980	1.58103	1.58016	1.58134	1.57955	1.58082
18/04/2008 13:00:00	1.57571	1.58140	1.57422	1.58041	1.57549	1.58115	1.57397	1.58016
18/04/2008 11:00:00	1.57329	1.57581	1.57221	1.57571	1.57301	1.57556	1.57196	1.57549
18/04/2008 09:00:00	1.57404	1.57664	1.57141	1.57329	1.57382	1.57639	1.57116	1.57301
18/04/2008 07:00:00	1.58441	1.58557	1.57129	1.57404	1.58412	1.58534	1.57104	1.57382

As one can see in Table 5.2, every record has a unique time and date value in an according column. This value is referred to as date-time stamp in this work and identifies a unique interval.

The data is collected for the Euro/USD currency pair.

Definition 5.1: Currency-Pair

A currency-pair is the quotation of the relative value of a currency-unit against another, in the foreign exchange market. The currency that is used as the reference is called the quote or counter currency; and the currency that is quoted in relation is termed the base or transaction currency.

Usually when an index or price point is observed in a financial market for a defined interval, 8 values are produced. The same is the case here as Euro/USD is watched for 2-hour intervals.

There are 2 main categories of prices:

1- *Ask price*
2- *Bid price*

'Ask' price is the price the sellers are asking for and the '*Bid'* price is the price that the buyers are offering.

Any interval has 4 noteworthy aspects to it:

1- *Open*
2- *Close*
3- *High*
4- *Low*

'Open' and *'Close'* indicate the point at the beginning and end of the interval respectively.

'High' and *'Low'* indicate the point in the interval where the price reaches a maximum and minimum respectively.

And both the *Ask* price and the *Bid* price have naturally all the above 4 values. Therefore, the total number of tracked values for each interval is 8, namely: (Open,

Ask), (High, Ask), (Low, Ask), (Close, Ask), (Open, Bid), (High, Bid), (Low, Bid), (Close, Bid).

However, in the case of this work, one value is sufficient to be taken into consideration to indicate the price of a currency-pair at a point in time.

This work chooses the (Close, Bid) value for a technical reason i.e. a more straightforward system design. The technical reason is basically that the *Close* refers to the end of an interval and this helps the algorithm to consider it as a future value that comes into existence after news occurrence. Such value reflects news-impact. How this is exactly implemented is detailed in Figure 4.3 in Input-Data Preparation section (4.4) in System Design chapter. The implementation includes an additional 1-hour margin around the window in which news is released. Furthermore, the reason that *High* and *Low* values are not considered is that they are not referring to the price value at any specific point in time; they rather just indicate the maximum and minimum value wherever they occur during the interval.

5.2.3 Consolidated Dataset

The above two datasets are combined into one dataset which is called the consolidated dataset. It is the resulted dataset that is used by the algorithms in the experiments.

The consolidated dataset has 3 columns. First, the date-time-stamp for the interval that is considered for news grouping i.e. grouping of news headlines, second, is the combination of the text of all the news headlines released in that specific interval, and third, a label that is either P (Positive) or N (Negative) depending on the observed change in the Close Bid. How the value of label is exactly calculated is presented in Figure 4.3 in Input-Data Preparation section (4.4) in System Design chapter. An example record of the consolidated dataset is presented in Table 5.3.

Table 5.3 Example Record from Consolidated Dataset

News-Grouping Date-Time-Stamp	All news-headlines in the same news-group	Label
18/04/2008 11:00:00	'Strong' demand outside North America propels Caterpillar to 13% quarterly profit growth Dow futures stage relief rally after Citigroup results Citigroup swings to quarterly loss of $5 billion, revenues fall 48%	P

There are a total of 6906 records in the consolidated dataset. Bear in mind, that the total number of news-headlines available in the news dataset is 13017. Once they are grouped together based on their release-times being in the same 2-hour intervals, the consolidated dataset is formed. In other words, each record in the consolidated dataset is associated with a group of news-headlines released in the same interval and therefore bundled together. In the following, it is explained how a myriad of experiments is designed and conducted using this dataset.

5.3 Experimental Design

The experimentations on the proposed system in this work are presented in the following manner.

Firstly, the entire system performance is demonstrated in the following with all components and layers in place. The details of the execution are provided and a discussion is presented on the significance of results. The overall result in terms of accuracy is rated at 83.33% for the standard system experiment that is used as the benchmark and repeated for other scenarios below.

Secondly, all the 3 core layers of the algorithm are eliminated from the algorithm. The results are revisited and the changes are observed and discussed. The benchmark-result in this case falls to a 50% which is equal to the probability of chance for a binary

classification. This demonstrates the effective impact of the proposed algorithm as a whole in the system.

Thirdly, the impact of each layer is observed by removal of only that layer from the flow of the algorithm in a described manner and the changes in results are studied. In every case of a layer removal, there is a drop of accuracy in the results for the benchmark experiment, which proves the relevance, and positive impact of each layer accordingly.

Fourthly, all the above experiments are conducted on a set sample size. This is the way a benchmark experiment is defined and used to compare outcomes of different experiments. However, at another level there are also a number of experiments run with the system in its entirety but with a varying sample size in order to observe the changes in results that may occur by this variation. These results are reported. A variation is observed, however, all experiments have significant positive results and on average the results are also positively noteworthy. This aspect of experimentation is also reported in detail in Section 5.10 in the following.

Now that the major aspects of the experimental design and the flow of experiments are explained, a more detailed look is presented on each of the above matters in what follows.

5.4 System in Entirety

5.4.1 Experiment Description

Before describing the experiments, here is a brief account about the used datasets for training and testing. The total data-set spans from 2008 to 2011 and has 6906 records. The records are chronologically in order. The proposed system makes a prediction per record at a time. Hence, in order to make a prediction for the last record all records

before it are used for training and for the second-last all the ones before that and so forth. Each record is associated with a 2-hour time interval so that if the system is left to run for 24 hours 12 records are predicted. 24 hours is the time-span covered by 12 records and not the running time needed to make the relevant predictions. The total execution time for 12 predictions on a PC with total of 4GB RAM and 3.10GHz Intel CPU is about 2min or 10sec per prediction. As a 24 hour-coverage seems reasonable to test a system that is predicting the next hour, the below tests are conducted on the last 12 records of the dataset.

5.4.2 Experiment Results

When the proposed techniques are available in the system and the experiment is run an accuracy of 83.33% is achieved on the above described testing sample.

5.4.3 Discussion

Just to point out the accuracy range in other works in the literature for the sake of curious comparison: The accuracy in majority of the cases is reported in the range of 50 to 70 percent, while commonly arguing for better than chance results which is estimated at 50 percent (Butler & Kešelj, 2009; F. Li, 2010; Mahajan et al., 2008; Schumaker & Chen, 2009; Schumaker et al., 2012; Zhai et al., 2007). In other words, the reported systems in the literature require different inputs and therefore are not compared with each other in the same experimental settings but most of them report their results because they have achieved better than chance results. Furthermore, generally results above 55% have been considered categorically report-worthy in similar contexts in the literature (Garcke et al., 2013).

In the following each of the techniques is removed from the system individually and its impact on accuracy is observed. In all cases a significant improvement in accuracy is

noticed when the technique is present in the system. And this is our main evaluation approach for the results presented in this work.

5.5 Complete Removal of Multi-layer Algorithm

5.5.1 Experiment Description

The core of the proposed system contains the Multi-layer Algorithm that entails 3 layers, namely: Semantic Abstraction, Sentiment Integration and Dimension Reduction.

In this experiment the entire Multi-layer Algorithm is eliminated from the system and its impact is studied. In other words, the benchmark experiment as described above is conducted again and the accuracy level of the outcomes is observed. The results are detailed in the following section.

5.5.2 Experiment Results

If the entire Multi-layer Algorithm is removed from the system, the benchmark experiment results in accuracy of 50.00%; which coincidently is what is presumed to be the accuracy of classification by chance when 2 classes are available.

5.5.3 Discussion

This above results re-emphasizes the significance of the proposed layers in the Multi-layer Algorithm; specially because once the algorithm is removed the accuracy level goes to a level equivalent to the probability of chance for a binary classification.

5.6 Abstraction-Layer Removal

5.6.1 Experiment Description

The Abstraction-Layer in the Multi-layer Algorithm is where the Heuristic-Hypernyms are selected. In order to evaluate the effectiveness of this layer, the Heuristic component is eliminated in the following manner.

The Heuristic component as described in Section 4.6.4 comes into effect in selection of features by observing if a given token possesses an entry in WordNet. Thereby token that do not possess an exact match are eliminated. And the remaining tokens are the ones that are referred to in this work as heuristically selected ones. Their Hypernyms compose the feature set from this point on. In order to put the effectiveness of this heuristic selection method to test, this experiment resorts to stemming in the following fashion.

If each token is stemmed first then the chances of existence of the stem as an entry in WordNet are much higher. In this form, almost all the tokens possess an entry in WordNet and therefore are not eliminated from the feature list. This is the context in which the elimination of the Heuristic-method is assumed. This is what this experiment is targeting to evaluate.

However, the Hypernym-based element of the abstraction layer is still in play. That element cannot be simply removed because that is how the tokens are associated with WordNet entries. In order to eliminate that component the stems could be used as direct anchor points to the ontology (WordNet). However, in that case there are simply too many features available and the experiment becomes infeasible.

In this experiment the impact of the elimination of the Heuristic component of the Abstraction Layer is observed and the results are presented in the next section.

5.6.2 Experiment Results

The results are shown in Table 5.4 and are significantly lower when the heuristic is not used.

Table 5.4 Prediction Results using Hypernyms of Stems instead of Hypernyms of Exact Words

	Precision (N)	Precision (P)	Recall (N)	Recall (P)	Accuracy
Hypernyms (TF-IDF*SumScore)	88.89%	66.67%	88.89%	66.67%	**83.33%**
Hypernyms of Stems (TF-IDF*SumScore)	60.00%	14.29%	33.33%	33.33%	**33.33%**

5.6.3 Discussion

In this experiment, the number of produced hypernyms based on stems is 2318 which is higher than the number of produced hypernyms based on exact words that is 2149. Once scaled by TF-IDF*SumScore the numbers of produced features for stems and exact words become 495 and 435 respectively as many of them are eliminated by a SumScore of 0. This indicates that a difference of 60 features is causing the accuracy to go from 83.33% to 33.33%.

A drop in accuracy from 83.33% to 33.33% indicates that when the heuristic approach is not utilized for elimination of words, more words are replaced with their hypernyms; these new words introduce noise and therefore the results turn out to be poor. This indicates that the additional hypernyms are less valuable for the predictive purpose. This is an interesting discovery that gives value to the availability of exact matches in WordNet and the proposed heuristic approach that is designed to take advantage of it.

As seen in Table 5.4, in addition to the accuracy, both precision and recall of the two classes (N and P) are also significantly higher when the proposed Heuristic approach is in place.

To be exact, once the Heuristic approach is in place, Precision and Recall for class N are both 88.89% and for class P are both 66.67%; while they drop to 60% and 33.33% for class N and 14.29% and 33.33% for class P when the heuristic approach is not used.

This indicates that when each of the classes (N and P) are looked at individually, they both still perform better in the case where the heuristic approach is utilized. Moreover, they perform better at two levels: Firstly, in terms of the number of the correctly predicted cases from a class out of all the *available* cases in that classes (Recall). Secondly, in terms of the number of the correctly predicted cases from a class out of all of the *predicted* cases from that class (Precision). This shows that the drop in accuracy is not caused by a drop in one class only and both of the classes experience an improvement via the proposed heuristic approach.

This test proves the above assumptions and demonstrates that the existence of an exact entry for a word in WordNet is indicative of some value. Note that words that have similar endings but are not in possession of an entry in WordNet, are discarded and this heuristic seems to strike a meaningful balance in the logic of feature selection and reduction.

5.7 Sentiment-Layer Removal

5.7.1 Experiment Description

In order to determine the effectiveness of the usage of TF-IDF*SumScore weighting to scale the feature-vector in comparison with other options a number of tests are conducted on the above dataset.

Other options instead of the proposed SumScore are namely: NegScore, PosScore and ObjScore as well as the scenario whereby none of the above sentiment scores are employed.

NegScore, PosScore and ObjScore are the Negativity Score, Positivity Score and Objectivity Score available in SentiWordNet for each entry as described in section 4.7.1.

In the Sentiment-Layer of the Multi-layer algorithm, what occurs is a multiplication or in other words scaling of the TF-IDF weight by the SumScore that is the sentiment score proposed in this work.

In order to contrast the impact of SumScore compared to the other scores, 3 experiments are conducted whereby the above scaling or multiplication happens with ObjScore, PosScore and NegScore respectively instead of the SumScore and the results are shown and discussed in the next section. One more experiment is conducted in addition to the above 3, whereby TF-IDF is simply used on its own and is not scaled (multiplied) by any sentiment weight at all. The results of this experiment are also presented in the next section.

Moreover, to satisfy curiosity about the significance of TF-IDF in TF-IDF*SumScore a new set of experiments is conducted which are similar to the above described, however, TF-IDF in them is simply replaced with a binary value of 0 or 1. The results of these tests are also presented in the next section.

5.7.2 Experiment Results

Table 5.5 illustrates the effectiveness of SumScores against other weighting possibilities for hypernyms as well as no weighting at all. TF-IDF*SumScore is clearly leading at 83.33% which is significantly high, followed by TF-IDF*NegScore at 75% and then 58.33% for both TF-IDF*PosScore and TF-IDF*ObjScore as well as the TF-IDF alone.

Table 5.5 Sentiment Weight (SumScore) Evaluation with TF-IDF-Base

	Precision (N)	Precision (P)	Recall (N)	Recall (P)	Accuracy
TF-IDF	75.00%	25.00%	66.67%	33.33%	**58.33%**
TF-IDF*SumScore	88.89%	66.67%	88.89%	66.67%	**83.33%**
TF-IDF*ObjScore	75.00%	25.00%	66.67%	33.33%	**58.33%**
TF-IDF*PosScore	83.33%	33.33%	55.56%	66.67%	**58.33%**
TF-IDF*NegScore	87.50%	50.00%	77.78%	66.67%	**75.00%**

To see if TF-IDF itself is of any impact on the results, the above tests are run again but this time on a binary feature-vector of the hypernyms instead of a TF-IDF weighted one and the results are listed in Table 5.6. In this context, as well, SumScore and NegScore do best but this time equally. The main objective of this set of tests is to determine if the inclusion of TF-IDF in the weighting is bringing any value to the table and the clear answer is yes as indicated by the results that are significantly lower compared to their TF-IDF counterparts in Table 5.5.

Table 5.6 Sentiment Weight (SumScore) Evaluation with Binary-Base

	Precision (N)	Precision (P)	Recall (N)	Recall (P)	Accuracy
Binary	80.00%	28.57%	44.44%	66.67%	**50.00%**
Binary*SumScore	77.78%	33.33%	77.78%	33.33%	**66.67%**
Binary*ObjScore	66.67%	16.67%	44.44%	33.33%	**41.67%**
Binary*PosScore	71.43%	20.00%	55.56%	33.33%	**50.00%**
Binary*NegScore	85.71%	40.00%	66.67%	66.67%	**66.67%**

5.7.3 Discussion

Low accuracy results in Table 5.5 for PosScore and ObjScore indicate the lack of any positive impact by them when compared to their complete absence in the case of sole TF-IDF presence. The higher accuracy results for NegScore indicates its potential. It is indicated in the literature that negative words in stories about fundamentals predict earnings and returns more effectively than negative words in other stories (Paul C. Tetlock et al., 2008) and the effectiveness of negative words is observed in this experiment as well. However, the determination of Objectivity by ObjScore seems to be of very little value in this context. This approves of our assumptions made in section 4.7.1 on the desired nature of sentiment integration via a sentiment weight.

5.8 Feature-Reduction-Layer Removal

5.8.1 Experiment Description

In order to test the effectiveness of the proposed Synchronous Targeted Feature-Reduction technique, it is eliminated from the system and an experiment is conducted. In this experiment all available features are used to create the model and not only the ones relevant to the record targeted for prediction.

5.8.2 Experiment Results

This test results in a significantly lower accuracy as shown in Table 5.7.

Table 5.7 Feature Reduction Layer Evaluation

	Precision (N)	Precision (P)	Recall (N)	Recall (P)	Accuracy
TF-IDF*SumScores	75.00%	25.00%	33.33%	66.67%	**41.67%**

5.8.3 Discussion

The average number of available features to consider for a target prediction is 8.7 when the feature-reduction component is in place; whereas the total number of features available without the feature reduction layer is 435. The experiment demonstrates that the excessive availability of features leads to a low accuracy of 41.67%. The proposed layer, however, successfully manages to reduce the number of features that are to be considered by the machine learning algorithm for the prediction task and do it in a way that the most relevant features for a specific prediction at hand are taken into consideration. Thereby, the proposed method increases the prediction accuracy at this experiment from 41.67% to 83.33% and the results prove the significance of its existence.

5.9 Machine-Learning-Algorithm Variation

5.9.1 Experiment Description

The same experiment is also conducted on the system with 3 different machine leaning algorithms, namely: SVM, k-NN and Naïve Bayes and variations of C and K for SVM and k-NN respectively. The used values of 0 to 10 for C and 1 to 6 for K are simply chosen because they are common values for the constants which can help develop a feeling for the performance of the algorithms and the tradeoffs those values may be causing. However, as tweaking the algorithms to perfection is out of scope of this work, further values are not tested. Additionally, k-NN is tested with and without weighted votes based on distance.

5.9.2 Experiment Results

The conclusion is that SVM with a C = 0 maintains the best position in terms of results at 83.33% for accuracy in all experiments. A detailed graphical breakdown of the results follows.

In Fig. 5, different accuracy levels for 11 different C values in the SVM algorithm are illustrated. C=0 produces the highest result at 83.33%.

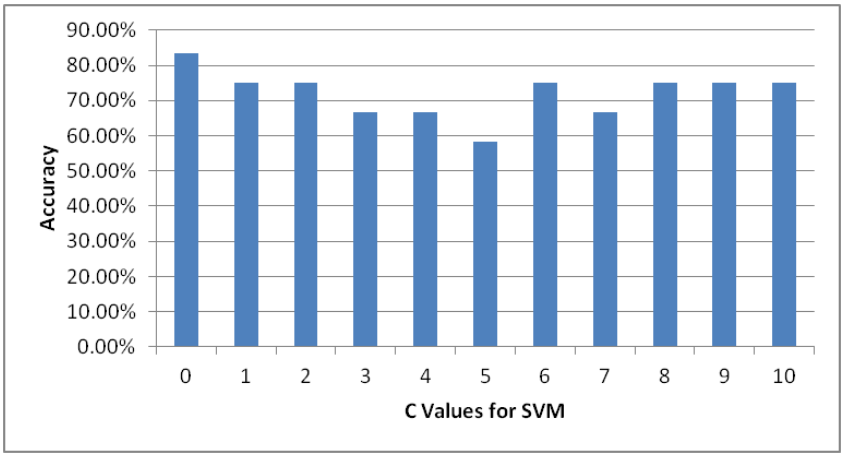

Figure 5.1 Accuracy Levels for Different C Values in SVM

Fig. 6 compares accuracy levels for 6 different K-values in k-NN for 2 different setups: First, with normal or non-weighted votes and second with weighted votes. In general, weighted votes in this context do not seem to be of assistance. Furthermore, k-NN results with their peak at 58.33% are significantly lower than SVM results peaking at 83.33%.

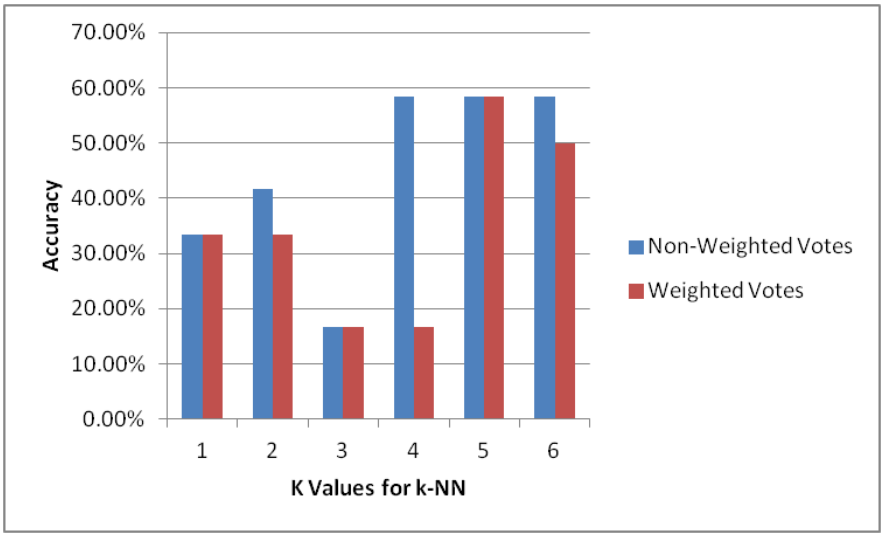

Figure 5.2 Accuracy Levels for Different K values in k-NN for Weighted (right columns) and Non-Weighted Votes (left columns)

In addition to SVM and k-NN another popular machine learning algorithm is experimented with, namely, Naïve Bayes. But its result is only as good as those of k-NN and is at 58.33%. The overall comparison of the 3 major algorithms is depicted in Fig. 7 with SVM at 83.33% and k-NN and Naïve Bayes at 58.33% at most on the used testing sample set.

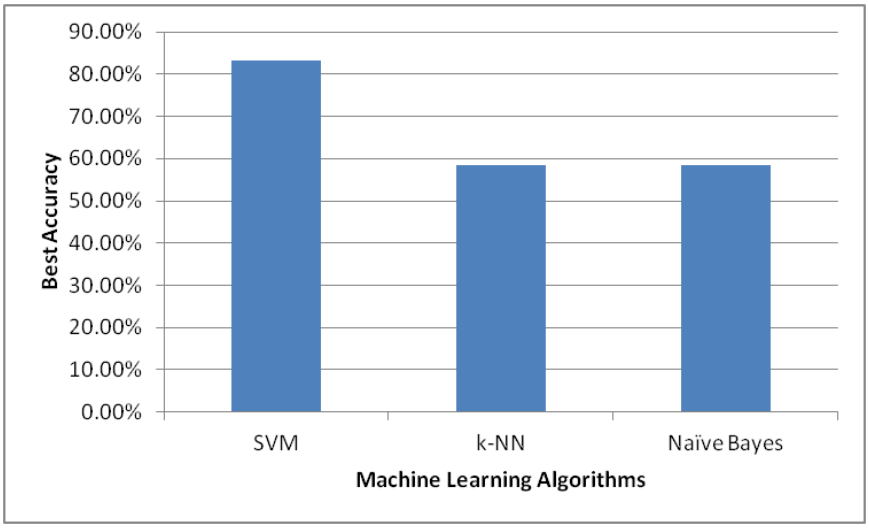

Figure 5.3. Accuracy Levels for Different Machine Learning Algorithms

5.9.3 Discussion

SVM in general is very popular in text classification problems and proves to be distinctly better than the other alternatives in the context of this experiment too. This may be contributed to the ability of SVM in calculating a weight for each feature at hand and assigning it to them in a way that availability of different features results to different weights. Furthermore, SVM is also good at handling sparse data; and the data that is dealt with here, in this context, is sparse. Although, the proposed multi-layer feature reduction algorithm reduces the number of features and, therefore, high dimensionality is not present at a horizontal level in terms of features in the feature matrix but vertically the data is considerably sparse and the ability of SVM to deal with sparse data proves to be helpful at this level too.

5.10 Sample-Size Variation

5.10.1 Experiment Description

In all of the above experiments the test sample size is 12, which is less than 1% of the total number of records that are available in the dataset. Therefore, the entire system is

experimented with multiple times with varying test sample-sizes. This is presented in this section.

5.10.2 Experiment Results

Table 5.8 summarizes the results of the conducted experiments with regards to sample-size variation. For each experiment, it provides the standard evaluation measures, namely, precision and recall for each class and the overall accuracy. The only factor that is changing from one experiment to the next is the number of data instances that are used to conduct the test. The training data-set is the remaining portion.

Table 5.8 Results for different sizes for the testing dataset

Testing dataset size	Precision (N)	Precision (P)	Recall (N)	Recall (P)	Accuracy
6	100.00%	0.00%	83.33%	0.00%	**83.33%**
12	88.89%	66.67%	88.89%	66.67%	**83.33%**
18	61.54%	80.00%	88.89%	44.44%	**66.67%**
24	52.94%	85.71%	90.00%	42.86%	**62.50%**
30	47.62%	77.78%	83.33%	38.89%	**56.67%**
36	45.83%	66.67%	73.33%	38.10%	**52.78%**
42	46.15%	75.00%	75.00%	46.15%	**57.14%**
48	54.84%	70.59%	77.27%	46.15%	**60.42%**
54	50.00%	65.00%	70.83%	43.33%	**55.56%**

5.10.3 Discussion

The experiment is conducted 9 times and the testing data-set size is increased from 6 to 54 as multiples of 6. As more test instances are considered, fewer instances are left for training and the overall accuracy decreases; in other words, the smaller the size of the testing dataset the bigger the size of the training dataset and therefore the more accurate the results as illustrated in Fig. 5.4.

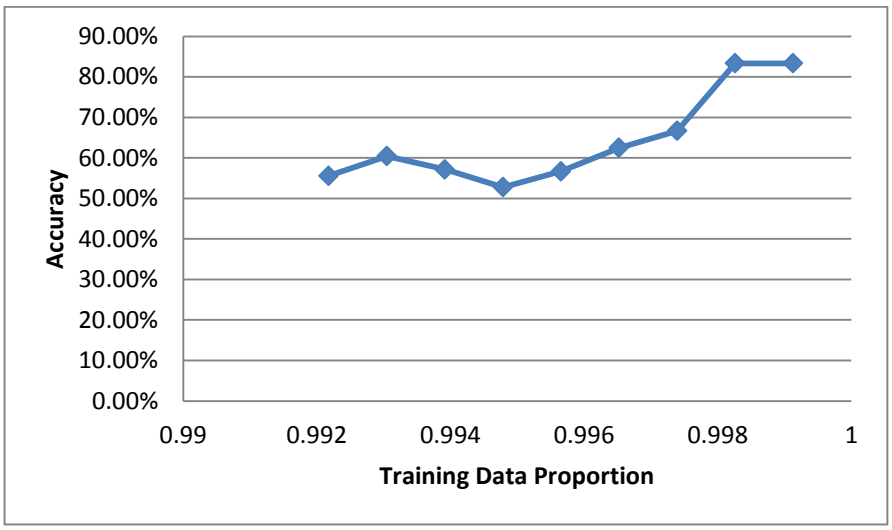

Figure 5.4. Resulted accuracy by different training data-set sizes

5.11 Chapter Summary

In this chapter the proposed system with the multi-layer algorithm is put to test from multiple perspectives. At first, the specific data-set that the experiments are conducted on, is described in detail. Then, the design of the experimentation strategy is amplified. In short, the proposed system with the multi-layer algorithm at its core is evaluated through removing each component i.e. layer of the algorithm and observing its impact on the produced prediction accuracy.

Hence, in order to produce a benchmark the system is run on a benchmark test-dataset in its entire form with all its components present initially. Subsequently, the entire multi-layer algorithm is assumed not to exist and an experiment is conducted to gauge prediction accuracy in its complete absence. Following this, separate experiments are conducted to identify the impact of each component i.e. layer of the algorithm individually. The 3 layers of the system that are eliminated at each experiment are: Abstraction-Layer, Sentiment-Layer and Feature-Reduction-Layer, respectively. In terms of results, each of the above experiments demonstrates a clear lack of accuracy when the layer subjected to the experiment is absent. This proves every aspect of the

proposed multi-layer algorithm is relevant and positively impactful in producing a high level of accuracy. Furthermore, the system as a whole demonstrates significant improvement in prediction accuracy as well, compared to a scenario where it is not implemented at all.

Moreover, the proposed algorithm is experimented with via multiple machine-learning algorithms so that the most suitable for the context is identified. The result turns out to be SVM and thereby it is proposed as the core of the machine learning component. However, machine learning algorithms can be further fine-tuned but that activity is not within the scope and objectives of this work.

Finally, multiple experiments are conducted to observe the impact that the size of the testing-sample may have. As a result, it is viewed that the number of instances that are dedicated to the testing-data proportion, and accordingly, the number of instances dedicated to the training-data portion in the dataset appears to have an impact on accuracy results. The results appear to be better as the training dataset grows. However, in all experiments, results turn out to be distinctly promising for a bi-directional prediction.

6 Conclusion

6.1 Introduction

On a board level, this study addresses a topic that is rapidly becoming very significant in today's age of information, namely: context-specific text-mining. The amount of textual data available on any conceivable topic and in numerous formats compels the field of text-mining to start specializing in different functional contexts. This study is an early attempt to mold and thereby advance text-mining in a specific context.

The specific context tackled in this work is also highly significant as it delves into the nature of financial markets, which have become inseparable elements of economic well-being of today's societies. The specific market-type that is focused on is the Foreign Exchange Market (FOREX) and in it, the particular index that is used for experimentation is Euro/USD, which is the conversion rate of Euro based on US dollar. The textual source of data that applies to this context and is utilized in this work is the financial breaking news from a prominent news website without any further topic specific filtration.

The high level objective of this study is to investigate the existence of a predictive short-term relationship between the news pieces released the bi-directional movements of the market in terms of the specified index. From a technical perspective the focus remains on news-headlines and not news-bodies in this investigation. This summarizes the context that this study focuses on, and its overall objective.

The above question is relatively open, and in order to devise a strategy to address it, it becomes crucial to comprehend the past efforts made that are comparable with the target of this work and also the theoretical economics and more specifically behavioral economics that is required to form a foundation for the above mentioned hypothesis. This is performed comprehensively in the literature review of this work and is hoped to be of aid to future researchers on this path.

As a result of a comparative analysis performed in the literature review, it becomes clear that in order to get the best out of text-mining for this context, it must be improved on a number of fronts beyond what has been previously attempted in a similar context. The specific topics that are concluded and tackled as a result of this work are:

1. Dimensionality-Reduction
2. Sentiment-Integration

3. Semantic-Integration

A multi-layer algorithm is devised that addresses dimensionality reduction at all levels and sentiment-integration and semantic-integration at dedicated layers.

The algorithm is implemented in a prototype system and is experimented with extensively and from various angles. The experimental results demonstrate positive existence of the investigated predictive relationship. Furthermore, the above identified aspects for enhancement in text-mining methods prove to be of meaningful impact on outcome results. Thereby, the study manages to successfully formulate a predictive-framework with a modular design, in form of layers of algorithms, which tackle correctly identified context-specific text-mining challenges.

In the rest of this chapter you can find concluding accounts on: Summary of Results and Findings, Achievements of the Objectives, Contributions, Limitations of the Current Study; and Recommendations and Future Directions.

6.2 Summary of Results and Findings

This research entails multiple results and findings:

A) As a result of this work multiple disciplines are brought together to address the research question of short-term predictability of FOREX based on news-headlines. The main disciplines integrated in this work are: text-mining (including sentiment analysis and semantic analysis), statistical pattern recognition or machine learning and economics (specifically behavioral economics).

B) To the best knowledge of the researcher, there is no previous comprehensive comparative analysis of the available works in the realm of market-predictive text-mining. A first attempt in this direction is produced as a result of this work.

C) This work identifies in its findings, specific aspects of text-mining that can be enhanced in order to achieve improved results in the context-specific text-mining algorithms, namely: 1- Dimension-Reduction, 2- Sentiment-Integration and 3- Semantic-Integration.

D) To address each of the above aspects a method is found and proposed as summarized below:

1. Dimension-reduction is addressed by all layers of the multi-layer algorithm. And in one dedicated layer via a novel technique termed Synchronous Targeted Feature-Reduction.
2. Sentiment-integration is addressed via a novel sentiment-weighting score termed SumScore, in the sentiment-integration layer.
3. Semantic-Integration is addressed via a novel feature-selection technique termed Heuristic-Hypernyms.

E) A system prototype is devised and implemented as a result of this work. The devised prototype is modular and scalable.

F) Experiments are designed and conducted to determine the effectiveness of each of the methods proposed, as summarized below:

1. To determine the effectiveness of the entire system, an experiment is conducted to make short-term predictions based on news headlines. In this experiment the proposed system is in place in its entirety. The produced results indicate an accuracy of 83.33%. Firstly, the results are very promising when compared with the odds of chance for a bidirectional decision which are at 50%. Secondly, the training and the testing dataset as well as the resulted accuracy rate of this experiment are used as benchmark-setup and -accuracy details for the rest of the experiments as detailed in the following.

2. In order to further confirm the effectiveness of the proposed system as a whole, an experiment is devised and conducted on the same training and testing data but without the proposed multi-layer algorithm in place. Rather, the experiment uses a mechanism that is common in past research, which entails running the machine learning algorithm directly on the text content transformed into a matrix with words as tokens i.e. features. The produced results here reconfirm the effectiveness of the proposed system as in its absence, the accuracy results on the same dataset fall dramatically down to 50.00%; which coincidently is what is presumed to be the accuracy of classification by chance when 2 classes are available.
3. The proposed multi-layer algorithm that composes the core of the proposed system has a layer for semantic integration. This layer utilizes WordNet hypernyms to produce an abstraction of the textual content in news-headlines via a heuristic method that is proposed. In order to determine the effectiveness of the proposed heuristic mechanism, an experiment is designed and carried out whereby the mechanism is eliminated from the system and the predictions on the benchmark data are executed again. The results show that the accuracy is much higher with the proposed mechanism than without it. The benchmark experiment produces and accuracy 33.33% without the proposed mechanism and an accuracy of 83.33% with it. This comparison proves the effectiveness of utilization of the heuristic approach for the hypernym-based abstraction. The heuristic approach simply assumes that the existence of an exact entry in WordNet for a word supersedes stemming all words to a root that has an entry. This perspective in usage of WordNet is new and evidently effective in the experimental context.

4. Another layer of the multi-layer algorithm is designated to sentiment integration. This work proposes a method to capture sentiment-load of words in the textual content. Many of the previous works have simply ignored sentiment or have not been enhanced enough to contain a sentiment-integration method in a market-predictive text-mining system. One significant aspect of the proposed sentiment-integration is that it introduces a sentiment weight by the name of SumScore that is new. The sentiment scores that are available in SentiWordNet address positivity, negativity and objectivity of a word. However, this work assumes that what is most important in the given classification context is a score that entails the sentiment-load of a given word regardless of its positive or negative direction and therefore proposes the SumScore that is calculated by summing up the PosScore and the NegScore. Experiments show that the usage of the proposed score in the sentiment-integration mechanism produces much better results compared to all the other alternatives, namely: PosScore, NegScore and ObjScore.
5. The proposed system primarily focuses on dimensionality reduction. Every layer of the algorithm plays a role in this perspective i.e. the semantic-integration and the sentiment-integration layers are also fulfilling a dimensionality reduction responsibility in the way they are designed. However, one layer that is termed Synchronous Targeted Feature Reduction layer is dedicated fully to dimensionality reduction. In contrast to available text-mining methods, whereby a set training dataset is used to train a model that is then used to classify each of many instances in a testing dataset, this work proposes that a model that is most customized for a prediction works best for that prediction. Therefore, this work proposes the Synchronous Targeted Feature Reduction mechanism that enables reducing the available features in the training dataset to the minimum number of

features available in an instance targeted for prediction i.e. classification. This is done in a synchronous manner in the multi-layer algorithm. As the algorithm is being executed and while predictions are being made, right before every prediction a custom model is generated and utilized. In order to measure the effectiveness of the proposed approach, an experiment is devised and conducted. In the experiment the proposed approach is contrasted against a scenario whereby the targeted feature reduction is eliminated from the algorithm in that the model is trained using all the available features and then run to predict a certain instance. Results show that when the proposed approach is in place the results are much better in terms. The absence of the proposed approach produces an accuracy rate of 41.67% for the benchmark experiment while when the proposed targeted feature reduction mechanism is in place the accuracy is at 88.33% for the benchmark experiment. This proves the significance of the proposed assumption of custom models for individual prediction instances that is a novel approach.

6. The next finding of this work is with regards to the most suitable machine learning algorithm for the market-predictive text-mining system. An array of prominent machine learning algorithms are placed inside the prototype and experimented with, namely, Support Vector Machine (SVM), K-nearest neighbors and Naïve Bayes among others. The findings indicate that SVM leads by far in terms of the produced accuracy in this context at 83.33%. Whereas, the other two algorithms produce results just below 60% for the benchmark experiment. Since fine-tuning of algorithms is not within the scope of this work, this initial suitability comparison is deemed to be sufficient for the context of this work and is regarded as a peripheral finding.

7. Lastly, in terms of experiments conducted in this work the benchmark experiment dataset is revisited. The set size of the testing dataset is increased multiple times and thereby the set size of the training portion from the available data is shrunk. The benchmark experiment is run again with the exact same format of the system but for a variety of sample sizes for test and training dataset portions. The results positively indicate that for all of the experimented sample sizes utilized in 9 different experiments the results produced by the proposed system constantly remains distinctly above the 50% odds of chance. This ensures that the produced results are not dependent on a set of testing instances. However, of course different experiments bear a degree of variation in accuracy which is deemed normal as prediction results are not absolute. Interestingly, the more the testing sample-size increases or in other words the more the training sample-size decreases, the accuracy seems to decrease accordingly. This indicates that an increase in training sample-size, leads to an increase in accuracy in this context.

The above summarizes the findings of this work as well as brief descriptions of the conducted experiments and their produced results and the learning thereof. In the next section, it is detailed how the objectives of this work are achieved.

6.3 Achievement of the Objectives

The set objectives for this research are listed below with a description of how each one is met in the work at hand.

Objective 1: *To devise a methodology to test the existence of a predictive relationship between the content of financial news and a foreign-exchange-market currency-pair.*

To address this objective a complete end-to-end system is designed and implemented in form of a prototype. The system receives news-headlines from a prominent financial

news-website as input and possesses and array of activities in multiple layers which result in a binary-classification decision at the end as output. The binary-classification is set to determine market directional movement in the next 1 to 3 hours after news release.

Since the proposed system is able to successfully determine market-direction in the specified context a head of time, with distinctly better than chance accuracy; the proposed methodology determines positively that there exists a predictive-relationship in the specified context and is able to utilize it to make successful predictions.

Thereby, the first objective of this research is fully met. Furthermore, the created framework can be used to lay grounds for more advanced experimentation by future researchers on the continuation of this path.

Objective 2: *To identify decisive text-mining elements that can improve the accuracy of such market prediction through news mining.*

The proposed system is modular and provides the capability to experiment with different elements that are involved in such a text-mining environment. This combined with a thorough literature review that contains a detailed comparative analysis of the available systems, has made it possible to identify specific elements in the market-predictive text-mining context that are lacking enhancement in the available systems and can be decisive in improvement of prediction accuracy results.

The identified text-mining elements for this objective are:

1- Dimension Reduction i.e. techniques to tackle the problem of high-dimensionality in the text-mining of news.

2- Sentiment Integration i.e. techniques that help with taking note of the relevant sentiment available in the natural language that is presented in textual format.

3- Semantic-Integration i.e. techniques that make use of semantics or the inter-related meaning of words available in a language.

In this work, the above elements and aspects are identified as factors that can be decisive for performance of a system in the given context. Thereby, the second objective of this research which is to identify such factors is fulfilled. As the next objective of this work, possible effective improvements on such factors are determined. This is described in the following section.

Objective 3: *To devise improvement-techniques for the identified text-mining elements*

After identifying the possible improvement elements for the market-predictive text-mining context of this work, an array of algorithms are proposed to address each aspect. Each algorithm is then tested on a benchmark experiment to determine the level of its impact. The entire system is proposed in the form of a multi-layer algorithm, with a designated layer to each identified aspect of the problem at hand. The entire system altogether addresses the most prominent problematic element which is the problem of high dimensionality as well.

Here is a brief recap of the devised techniques for this objective:

1- For the dimension-reduction aspect, a layer is designated in the multi-layer algorithm that contains the proposed technique in the form of an algorithm by the name of Synchronous Targeted Feature Reduction. The proposed technique brings the number of features that are used for model creation down significantly and only to those that are absolutely necessary on the contrary to previous systems which consider many features in their model creation that are indeed irrelevant. This work experimentally demonstrates that the proposed technique proves to be meaningfully impactful.

In addition to the above described designated layer, the multi-layer algorithm takes advantage of the other two available layers to reduce dimensionality in the below manner:

A) The sentiment-integration layer automatically eliminates words that are of zero sentiment value. This occurs via and association of zero for the proposed sentiment-weight. The primary function of the sentiment-weight is pointed out in the next section.

B) The semantic-integration layer, not only integrates semantics but also it performs semantic-abstraction. In other words, it provides semantic integration through semantic abstraction. Semantic-abstraction in essence is the replacement of words in the textual content which have the same parent-word via a hypernym-relation with the hypernym itself. In this way, features are reduced as the words with the same hypernym are regarded as one entity or feature and not many separate ones.

2- For the sentiment-integration aspect, an algorithm is devised that is placed in a dedicated layer in the multi-layer algorithm. The proposed sentiment-integration technique has a novel sentiment-weight at its core that is termed SumScore. SumScore's novelty lies in its emphasis on the sentiment-load of a given word and not the nature or direction of the sentiment in terms of positivity or negativity. Experiments prove the capability and significance of the proposed technique in this layer towards improvement in prediction accuracy results.

3- For the semantic-integration aspect, an algorithm is also devised and placed in a dedicated layer in the multi-layer algorithm. The algorithm performs semantic-integration via semantic-abstraction. The novel technique proposed here utilizes WordNet hypernym-relationships among words and is titled: "Heuristic-Hypernyms". As apparent in the name in addition to utilization of hypernyms, the technique relies on a heuristic component for selection of most effective features. The novel logic behind

the heuristic is that the existence of a direct entry in WordNet for a word can be assumed to give that word a usage significance that can be heuristically utilized to sort out words that have the most semantic impact. The conducted experiments in this work provide evidence for this assumption and the proposed technique as a whole.

In addition to addressing the research objectives one by one in the above, in the rest of this section, each of the research questions are also reiterated followed by the answer provided to them as a result of this work.

Research Questions and summary of concluded answers to them are as below:

1- Is there a predictive relationship between financial news-headlines and a foreign-exchange-market currency-pair?

Yes, the proposed system in this work indicates that there exists a predictive relationship between financial news-headlines and a foreign-exchange-market currency-pair. The proposed system provides and environment and a setup whereby the impact of groups of news-headlines can be determined as positive or negative on a currency-pair. This is carried out with an accuracy that is significantly above odds of chance and reaches 83.33% in one experiment.

2- What factors can cause the prediction-accuracy to be improved?

The factors that can cause the prediction accuracy to be improved are identified as below:

A) Dimension-Reduction, which tackles high dimensionality of the feature space

B) Sentiment-Integration, which takes note of the emotional content or sentiment of words in a natural language; and thereby differentiates more emotionally impactful words from the rest.

C) Semantic-integration, which utilizes semantic networks among words in order to identify related ones and regard them as one. Thereby include meaning of word into the equation.

3- Can an approach be developed to enhance those factors?

Yes, for each f the above identified factors a novel technique is devised and implemented in form of an algorithm that successfully enhances the specific aspect. The summary of the developed approaches are as below:

A) A novel algorithm is devised for dimension-reduction is termed Synchronous Targeted Feature Reduction and is located in a dedicated layer of the proposed multi-layer algorithm. Furthermore, the multi-layer algorithm as a whole is targeted at tackling high dimensionality and enables dimension reduction in the other two layers as well.

B) A novel sentiment-weight is developed to enable sentiment-integration in an innovative way. The new weight is termed SumScore.

C) A novel approach is developed for semantic-integration. It is titled Heuristic-Hypernyms.

Each of the above techniques that are developed in this work is tested through various experiments which prove them to be relevant and indeed assist with improvement of prediction results.

6.4 Contributions

This study offers multiple contributions on different levels as summarized below:

1- A comprehensive comparative analysis and gap identification of currently available market-predictive text-mining systems that identifies a number of areas that deserve

enhancements customized for this specialized context, primarily: dimension-reduction, semantics-integration and sentiment-integration.

2- A prototype system with a modular design for market-prediction through text-mining of news that successfully predicts directional market-movements in 1 to 3 hours after news release. This system is equipped with modules in form of layers for each of the above aspects which are targeted for customized enhancement.

3- A novel multi-layer algorithm that sits at the core of this proposed system and fulfills three main responsibilities: dimension-reduction, semantics-integration and sentiment integration. The three aspects are designed and combined together in a form that they support each other and reinforce dimension-reduction at multiple points in the system-flow.

4- A novel semantics-integration algorithm that enables semantic-abstraction through utilization of WordNet hypernyms combined with a novel heuristic method. The proposed new algorithm is titled: Heuristic Hypernyms. It integrates semantics through utilization of the semantic network available in WordNet. It uses hypernym-hyponym relations among words to combine words with the same hypernym and categorize them as the same entity or feature and thereby abstract the available features into fewer ones. Hence, as the end-result of this process, not only semantics are utilized but also is the dimensionality of features reduced.

5- A novel sentiment-integration algorithm that enables sentiment of words to be taken into consideration through a newly proposed sentiment-weight that is titled "SumScore". In the proposed weighting mechanism, the combination of SumScore with TF-IDF leads to the proper consideration of sentiment in proportion to frequency of occurrence at document and corpus level for the textual content. The core novelty of SumScore lies in proposal of a sentiment-score that is not based on direction of

emotions i.e. positive or negative emotions. It is rather based on the total emotional content; which is proposed to be the summation of both positive and negative emotions.

6- A novel dimension-reduction algorithm by the name of Synchronous Targeted Feature Reduction. It provides a synchronous model-creation methodology for the machine-learning algorithm so that only the minimum features related to a textual-instance are targeted for training the algorithm. In this way all the features that are irrelevant to a specific prediction are gotten ridden of and thereby dimensionality is reduced in the most relevant way.

7- An end-to-end scalable framework that is reproducible and provides solid grounds and directions for future researchers to delve into this emerging area of work and continue on the path that this work has started.

8- An experimental dataset without precedence that is the result of collection and organization of consolidated records of news-headlines mapped onto Euro/USD currency movements based on real data from multiple years.

6.5 Limitations of the Current Study

In order to point out the limitations of this study a number of issues can be considered. These issues are mostly not addresses in this work because they can be considered as full-fledged research topics and could not be included within the scope of this work. A summary of some of the main limiting issues can be found below:

1- *Trading engine*: in a prediction system like the one proposed in this work an additional component may be devised which can be termed a “trading engine”. It would be responsible for acting on prediction results produced by the system and enable real-time trading in the market based on them. It needs to take into account transaction costs. It also needs to possess one or more trading strategies. It should be able to perform

trades and calculate profits and losses over different time periods and with different strategies. Such a trading system would be a logical extension to this work and would enable concrete determination of the prediction-value based on gained profits or incurred losses. This work is limited by the absence of such engine in its concrete financial value determination.

2- *Dataset*: The dataset used for experimentation and its content is crucial to this research. Due to prior absence of such dataset, one has been created for this research that entails multiple years of news and market prices; however, a more comprehensive dataset with consolidation of more news sources, over even a much longer period of time could enable research of additional aspects like historic time-periods and incidents. It would also simply provide more input to the algorithms which is something that the researchers find as necessary for furthering this work.

3- *Deeper integration of semantics and sentiment*: Integration of both semantics and sentiment into text-mining are ongoing research topics and can be continuously further deepened. An extensive deepening on each of the above topics can be considered as independent full-fledged research topics and are forced to be left out of the scope of this research but would be invaluable in its continuation.

4- *Experimentation in other contexts*: The findings of this work can be further examined via experimentation in other contexts and on other datasets in a variety of ways from different textual data-sources to market-types; or even in predictive text-mining contexts other than market related ones like consumer sentiment related predictions etc. However, again, such experimentations would be work-intensive enough to constitute and trigger their own research-projects in separate time-frames from this work.

6.6 Implications

The main implications of this work are summarized in the below:

1- This work is among the first exploration efforts of the predictive relationship of news and the FOREX market. The promising results of this work indicate that such relationship exists and can be exploited in a predictive system like the one proposed. Therefore, the first implication of this work is that it acts as a successful feasibility test.

2- This work is an example of context-specific enhancement and specialization of text-mining methods through which promising results are achieved. This has the suggestive implication that text-mining research should be conducted in a more context-specific manner than it currently is.

3- This work emphasizes on Semantic Abstraction and Integration, Sentiment Integration and Dimensionality Reduction and produces promising results. An implication of this for future research is to also consider this strategy as one for improvement of text-mining methods in other contexts.

4- At a practical level, investment institutions and traders can benefit from the proposed market-prediction system. It can help make better financial decisions in the Foreign Exchange Market which lead to financial returns on investments and avoidance of severe losses.

5- Financial markets are challenging to comprehend and lack of insights into them can lead to financial crises like the recent one in 2008 with negative impact on a wide range of people. A market-predictive text-mining solution may help bring about more confidence on comprehension of market-movements and its relation to human mass-psychology through text-mining of the available textual resources on the Internet.

6.7 Recommendations and Future Directions

Market prediction mechanisms based on online text mining are just emerging to be investigated rigorously utilizing the radical peak of computational processing power and network speed in the recent times. We foresee this trend to continue. This research helps put into perspective the role of human reactions to events in the making of markets and can lead to a better understanding of market efficiencies and convergence via information absorption. In summary, this work recommends the below as areas or aspects in need of future research and advancement:

A) Semantics: Advancements of techniques in semantics are crucial to the text-classification problem at hand as text-mining researchers have already shown. However, such advancements have not yet entered into the field of market-predictive text-mining and this work is an early effort to do so. Development of specialized ontologies by creating new ones or customization of current dictionaries like WordNet requires more attention. Many of the current works of research are still too much focused on word occurrence methods and they rarely even use WordNet. Moreover, semantic relations can be researched with different objectives, from defining weighting schemes for feature-representation to semantic compression or abstraction for feature-reduction.

B) Syntax: Syntactic analysis techniques has received probably even less attention than semantic ones in current systems. More advanced syntax-based techniques like usage of parse-trees for pattern recognition in text can improve the quality of text-mining significantly. This aspect requires the attention of future researchers too, starting with attempting to transfer some of the learning in other areas of text-mining like reviews classification into market-predictive text-mining.

C) Sentiment: Sentiment and emotion analysis has gained significance and prominence in the field of text-mining due to the interest of governments and multi-national

companies to maintain a finger on the pulse of the public mood to win elections in the case of the former or just surprise their customers by the amount of insight about their preferences for the latter. Interestingly, market-prediction is very closely related to the mood of public or market-participants as established by behavioural-economics. However, in case of the analysis of sentiment with regards to a product the anticipation of what a piece of text entails is far more straightforward than in the case of market-prediction. There are no secrets as to whether a product-review entails positive or negative emotions about it. However, even the best traders and investors can never be completely sure what market-reaction to expect as a result of a piece of news-text. Therefore, there is a lot of room for market-predictive sentiment investigation for future research.

D) Text-mining component, textual-source or application-market specialization: This work has learnt that the current works of research on market-predictive text-mining are rather holistic with one-off end-to-end systems. However, in the future, the text-mining process should be broken down into its critical components like feature-selection, feature-representation and feature-reduction and each of those needs to be specifically researched for the specialized context of market-prediction. Furthermore, market-predictive text-mining can also become even more specialized by focusing on a specific source of text e.g. a specific social media outlet or news-source or text-role like news-headlines vs. news-body etc. Moreover, there is a need for specialized research on each type of financial markets (stocks, bond, commodity, money, futures, derivatives, insurance, forex) or on each geographical location.

E) Machine Learning Algorithms: SVM and Naïve Bayes are heavily favoured by researchers, probably due to their straightforwardness, while many other machine learning algorithms or techniques like Artificial Neural Networks (ANN), k-Nearest Neighbors (k-NN), fuzzy-logic, etc. show seriously promising potentials for textual-

classification and sentiment-analysis elsewhere in the literature but have not yet been experimented with in the context of market-predictive text-mining or are significantly under-researched at this stage.

F) Integration of technical signals: Despite their practical popularity with market-traders, technical signals which are the outputs of technical algorithms or rules like the moving average, relative strength rules, filter rules and the trading range breakout rules, are almost always left out of the market-predictive research that is based on text-mining. It may be because of the fact that the researchers who are for prediction based on text-mining are most probably for fundamental-analysis approaches and therefore opposed to the technical-analysis approaches. However, it is logically conceivable that hybrid-models based on the sum of the best of the two worlds (technical and fundamental) must produce even better results and this should be considered more vigorously in future research.

G) Relation with behavioural-economics research: As pointed out in this work, there exists a substantial interdisciplinary nature to this field of research specially between economics and computer-science. Deepening economics comprehension is crucial for future researchers. Currently, economics and specially behavioural economics theories are referred to in the literature just superficially and only to establish that public-mood has an impact on markets. However, a deeper study of behavioural-economics should reveal principles and learning whose parallel implementation in text-mining by computer-scientists may lead to true breakthroughs. This direction, although somewhat immature or vague at this stage, is highly encouraged by this work.

H) Availability and quality of experimental datasets: one of the major challenges observed is the unavailability of highly standardized datasets that contain mappings of text onto markets for certain periods of times that researchers can use for assimilation of

their experimentation and evaluation efforts. In the available work, most researchers have attempted to accumulate their own datasets. This has naturally resulted in fragmented dataset-formats and contents and a lack of adequate observation for critical characteristics in datasets. Future researchers are encouraged to standardize and release datasets for experimentation in market-predictive text-mining. Currently, the predominant standard datasets circling around in text-mining works are of movie-reviews which are not appropriate for this work. Q. Wu and Tan (2011) present an intriguing piece of research on transferring sentiment domain knowledge from one domain to another by building a framework between them that acts as a bridge between the source domain and the target domain. This may inspire some new thoughts in this area too.

I) Evaluation methods: much like the experimental datasets, evaluation methods as a whole are highly subjective. Most researchers are still comparing their results with the probabilities of chance and not so much with each others' works. Researchers are generally using evaluation mechanisms that widely vary; which make an objective comparative performance-evaluation virtually impossible. Therefore, future researchers could focus on such standardization initiatives as main objectives of their research in this field as market-predictive text-mining is here to stay.

7 References

Aggarwal, C. C., & Zhai, C. (2012). *Mining Text Data* (C. C. Aggarwal & C. Zhai Eds.).

Aghdam, M. H., Ghasem-Aghaee, N., & Basiri, M. E. (2009). Text feature selection using ant colony optimization. *Expert Systems with Applications, 36*(3, Part 2), 6843-6853. doi: http://dx.doi.org/10.1016/j.eswa.2008.08.022

Anastasakis, L., & Mort, N. (2009). Exchange rate forecasting using a combined parametric and nonparametric self-organising modelling approach. *Expert Syst. Appl., 36*(10), 12001-12011. doi: 10.1016/j.eswa.2009.03.057

Ara, M., #250, jo, Gon, P., #231, alves, & Benevenuto, F. (2013). *Measuring sentiments in online social networks*. Paper presented at the Proceedings of the 19th Brazilian symposium on Multimedia and the web, Salvador, Brazil.

Bacchetta, P., Mertens, E., & van Wincoop, E. (2009). Predictability in financial markets: What do survey expectations tell us? *Journal of International Money and Finance, 28*(3), 406-426. doi: http://dx.doi.org/10.1016/j.jimonfin.2008.09.001

Bacchetta, P., & van Wincoop, E. (2013). On the unstable relationship between exchange rates and macroeconomic fundamentals. *Journal of International Economics, 91*(1), 18-26. doi: http://dx.doi.org/10.1016/j.jinteco.2013.06.001

Baccianella, A. E. S., & Sebastiani, F. (2010). *SentiWordNet 3.0: An Enhanced Lexical Resource for Sentiment Analysis and Opinion Mining*. Paper presented at the Proceedings of the Seventh conference on International Language Resources and Evaluation LREC'10, Valletta, Malta. http://www.lrec-conf.org/proceedings/lrec2010/pdf/769_Paper.pdf

Bahrepour, M., Akbarzadeh-T., M.-R., Yaghoobi, M., & Naghibi-S., M.-B. (2011). An adaptive ordered fuzzy time series with application to FOREX. *Expert Syst. Appl., 38*(1), 475-485. doi: 10.1016/j.eswa.2010.06.087

Balahur, A., Mihalcea, R., & Montoyo, A. (2014). Computational approaches to subjectivity and sentiment analysis: Present and envisaged methods and applications. *Computer Speech & Language, 28*(1), 1-6. doi: http://dx.doi.org/10.1016/j.csl.2013.09.003

Balahur, A., Steinberger, R., Goot, E. v. d., Pouliquen, B., & Kabadjov, M. (2009). *Opinion Mining on Newspaper Quotations*. Paper presented at the Proceedings of the 2009 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology - Volume 03.

Bekiros, S. D. (2014). Exchange rates and fundamentals: Co-movement, long-run relationships and short-run dynamics. *Journal of Banking & Finance, 39*(0), 117-134. doi: http://dx.doi.org/10.1016/j.jbankfin.2013.11.007

Berka, T., & Vajteršic, M. (2013). Parallel rare term vector replacement: Fast and effective dimensionality reduction for text. *Journal of Parallel and Distributed Computing, 73*(3), 341-351. doi: http://dx.doi.org/10.1016/j.jpdc.2012.08.008

Bisoi, R., & Dash, P. K. (2014). A hybrid evolutionary dynamic neural network for stock market trend analysis and prediction using unscented Kalman filter. *Applied Soft Computing, 19*(0), 41-56. doi: http://dx.doi.org/10.1016/j.asoc.2014.01.039

Bollen, J., & Huina, M. (2011). Twitter Mood as a Stock Market Predictor. *Computer, 44*(10), 91-94. doi: 10.1109/MC.2011.323

Bollen, J. M., Huina; Zeng, Xiao-Jun. (2010). Twitter mood predicts the stock market. *Journal of Computational Science, 2*(1), 1-8. doi: 10.1016/j.jocs.2010.12.007

Brank, J., Mladenić, D., & Grobelnik, M. (2010). Feature Construction in Text Mining. In C. Sammut & G. Webb (Eds.), *Encyclopedia of Machine Learning* (pp. 397-401): Springer US.

Büyükşahin, B., & Robe, M. A. (2014). Speculators, commodities and cross-market linkages. *Journal of International Money and Finance, 42*(0), 38-70. doi: http://dx.doi.org/10.1016/j.jimonfin.2013.08.004

Burges, C. C. (1998). A Tutorial on Support Vector Machines for Pattern Recognition. *Data Mining and Knowledge Discovery, 2*(2), 121-167. doi: 10.1023/A:1009715923555

Butler, M., & Kešelj, V. (2009). Financial Forecasting Using Character N-Gram Analysis and Readability Scores of Annual Reports. In Y. Gao & N. Japkowicz (Eds.), *Advances in Artificial Intelligence* (Vol. 5549, pp. 39-51): Springer Berlin Heidelberg.

Cambria, E., Schuller, B., Yunqing, X., & Havasi, C. (2013). New Avenues in Opinion Mining and Sentiment Analysis. *Intelligent Systems, IEEE, 28*(2), 15-21. doi: 10.1109/MIS.2013.30

Chang, C.-C., & Lin, C.-J. (2011). LIBSVM: A library for support vector machines. *ACM Trans. Intell. Syst. Technol., 2*(3), 1-27. doi: 10.1145/1961189.1961199

Chatrath, A., Miao, H., Ramchander, S., & Villupuram, S. (2014). Currency jumps, cojumps and the role of macro news. *Journal of International Money and Finance, 40*(0), 42-62. doi: 10.1016/j.jimonfin.2013.08.018

Chen, J., Huang, H., Tian, S., & Qu, Y. (2009). Feature selection for text classification with Naïve Bayes. *Expert Systems with Applications, 36*(3, Part 1), 5432-5435. doi: http://dx.doi.org/10.1016/j.eswa.2008.06.054

Chordia, T., Goyal, A., Lehmann, B. N., & Saar, G. (2013). High-frequency trading. *Journal of Financial Markets*(0). doi: http://dx.doi.org/10.1016/j.finmar.2013.06.004

Chordia, T., Roll, R., & Subrahmanyam, A. (2005). Evidence on the speed of convergence to market efficiency. *Journal of Financial Economics, 76*(2), 271-292. doi: http://dx.doi.org/10.1016/j.jfineco.2004.06.004
Chordia, T., Roll, R., & Subrahmanyam, A. (2011). Recent trends in trading activity and market quality. *Journal of Financial Economics, 101*(2), 243-263. doi: http://dx.doi.org/10.1016/j.jfineco.2011.03.008
Collins, M., & Duffy, N. (2001, 2001). *Convolution Kernels for Natural Language*.
Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine Learning, 20*(3), 273-297. doi: 10.1007/BF00994018
Dai, S., Yang, X., & Li, N. (2011). Predicting Stock Price Fluctuations Using Online News. *Energy Procedia, 13*(0), 7591-7597. doi: 10.1016/j.egypro.2011.12.493
Das, S. R., & Chen, M. Y. (2007). Yahoo! for Amazon: Sentiment Extraction from Small Talk on the Web. *Manage. Sci., 53*(9), 1375-1388. doi: 10.1287/mnsc.1070.0704
De Martino, B., O'Doherty, John P., Ray, D., Bossaerts, P., & Camerer, C. (2013). In the Mind of the Market: Theory of Mind Biases Value Computation during Financial Bubbles. *Neuron, 79*(6), 1222-1231. doi: http://dx.doi.org/10.1016/j.neuron.2013.07.003
Desmet, B., & Hoste, V. (2013). Emotion detection in suicide notes. *Expert Systems with Applications, 40*(16), 6351-6358. doi: http://dx.doi.org/10.1016/j.eswa.2013.05.050
Di Caro, L., & Grella, M. (2013). Sentiment analysis via dependency parsing. *Computer Standards & Interfaces, 35*(5), 442-453. doi: http://dx.doi.org/10.1016/j.csi.2012.10.005
Dorantes Dosamantes, C. A. (2013). The Relevance of Using Accounting Fundamentals in the Mexican Stock Market. *Journal of Economics Finance and Administrative Science, 18, Supplement*(0), 2-10. doi: http://dx.doi.org/10.1016/S2077-1886(13)70024-6
Drucker, H., Burges, C. J. C., Kaufman, L., Smola, A., & Vapnik, V. (1997, 1997). *Support Vector Regression Machines.* Paper presented at the ADVANCES IN NEURAL INFORMATION PROCESSING SYSTEMS.
Duman, E., Ekinci, Y., & Tanrıverdi, A. (2012). Comparing alternative classifiers for database marketing: The case of imbalanced datasets. *Expert Systems with Applications, 39*(1), 48-53. doi: http://dx.doi.org/10.1016/j.eswa.2011.06.048
Duric, A., & Song, F. (2012). Feature selection for sentiment analysis based on content and syntax models. *Decision Support Systems, 53*(4), 704-711. doi: http://dx.doi.org/10.1016/j.dss.2012.05.023
Égert, B., & Kočenda, E. (2013). The impact of macro news and central bank communication on emerging European, forex markets. *Economic Systems*(0). doi: http://dx.doi.org/10.1016/j.ecosys.2013.01.004
Eleftherios Soulas, D. S. (2013). *Online Machine Learning Algorithms For Currency Exchange Prediction.* NYU, New York.
Esuli, A., & Sebastiani, F. (2006). *SENTIWORDNET: A publicly available lexical resource for opinion mining*. Paper presented at the In Proceedings of the 5th Conference on Language Resources and Evaluation LREC'06. SENTIWORDNET: A publicly available lexical resource for opinion mining
Evans, M. D. D., & Lyons, R. K. (2008). How is macro news transmitted to exchange rates? *Journal of Financial Economics, 88*(1), 26-50. doi: http://dx.doi.org/10.1016/j.jfineco.2007.06.001
Fama, E. F. (1965). Random Walks in Stock Market Prices. *Financial Analysts Journal, 21*(5), 55-59. doi: 10.2469/faj.v21.n5.55
Fama, E. F. (1970). Efficient Capital Markets: A Review of Theory and Empirical Work. *The Journal of Finance, 25*(2), 383-417. doi: 10.2307/2325486
Fan, R.-E., Chen, P.-H., & Lin, C.-J. (2005). Working Set Selection Using Second Order Information for Training Support Vector Machines. *J. Mach. Learn. Res., 6*, 1889-1918.
Fang, J., Jacobsen, B., & Qin, Y. (2014). Predictability of the simple technical trading rules: An out-of-sample test. *Review of Financial Economics, 23*(1), 30-45. doi: http://dx.doi.org/10.1016/j.rfe.2013.05.004
Fasanghari, M., & Montazer, G. A. (2010). Design and implementation of fuzzy expert system for Tehran Stock Exchange portfolio recommendation. *Expert Systems with Applications, 37*(9), 6138-6147. doi: 10.1016/j.eswa.2010.02.114

Feng, G., Guo, J., Jing, B.-Y., & Hao, L. (2012). A Bayesian feature selection paradigm for text classification. *Information Processing & Management, 48*(2), 283-302. doi: http://dx.doi.org/10.1016/j.ipm.2011.08.002

Friesen, G., & Weller, P. A. (2006). Quantifying cognitive biases in analyst earnings forecasts. *Journal of Financial Markets, 9*(4), 333-365. doi: http://dx.doi.org/10.1016/j.finmar.2006.07.001

Fung, G., Yu, J., & Lam, W. (2002). News Sensitive Stock Trend Prediction. In M.-S. Chen, P. Yu & B. Liu (Eds.), *Advances in Knowledge Discovery and Data Mining* (Vol. 2336, pp. 481-493): Springer Berlin / Heidelberg.

García, D., & Urošević, B. (2013). Noise and aggregation of information in large markets. *Journal of Financial Markets, 16*(3), 526-549. doi: http://dx.doi.org/10.1016/j.finmar.2012.07.003

Garcke, J., Gerstner, T., & Griebel, M. (2013). Intraday Foreign Exchange Rate Forecasting Using Sparse Grids. In J. Garcke & M. Griebel (Eds.), *Sparse Grids and Applications* (Vol. 88, pp. 81-105): Springer Berlin Heidelberg.

Ghazali, R., Hussain, A. J., & Liatsis, P. (2011). Dynamic Ridge Polynomial Neural Network: Forecasting the univariate non-stationary and stationary trading signals. *Expert Syst. Appl., 38*(4), 3765-3776. doi: 10.1016/j.eswa.2010.09.037

Ghiassi, M., Skinner, J., & Zimbra, D. (2013). Twitter brand sentiment analysis: A hybrid system using n-gram analysis and dynamic artificial neural network. *Expert Systems with Applications, 40*(16), 6266-6282. doi: http://dx.doi.org/10.1016/j.eswa.2013.05.057

Gönen, M. (2014). Coupled dimensionality reduction and classification for supervised and semi-supervised multilabel learning. *Pattern Recognition Letters, 38*(0), 132-141. doi: http://dx.doi.org/10.1016/j.patrec.2013.11.021

Gracia, A., Gonzalez, S., Robles, V., & Menasalvas, E. (2014). A methodology to compare Dimensionality Reduction algorithms in terms of loss of quality. *Information Sciences*(0). doi: http://dx.doi.org/10.1016/j.ins.2014.02.068

Gradojevic, N., & Gençay, R. (2013). Fuzzy logic, trading uncertainty and technical trading. *Journal of Banking & Finance, 37*(2), 578-586. doi: http://dx.doi.org/10.1016/j.jbankfin.2012.09.012

Gregory, P. R., & Stuart, R. C. (2004). How Markets Work *Comparing Economic Systems in the Twenty-First Century* (Seventh Edition ed., pp. 96-99): George Hoffman.

Griffiths, T. L., Steyvers, M., Blei, D. M., & Tenenbaum, J. B. (2005, 2005). *Integrating topics and syntax*.

Groth, S. S., & Muntermann, J. (2011). An intraday market risk management approach based on textual analysis. *Decision Support Systems, 50*(4), 680-691. doi: http://dx.doi.org/10.1016/j.dss.2010.08.019

Haddi, E., Liu, X., & Shi, Y. (2013). The Role of Text Pre-processing in Sentiment Analysis. *Procedia Computer Science, 17*(0), 26-32. doi: http://dx.doi.org/10.1016/j.procs.2013.05.005

Hagenau, M., Liebmann, M., & Neumann, D. (2013). Automated news reading: Stock price prediction based on financial news using context-capturing features. *Decision Support Systems, 55*(3), 685-697. doi: http://dx.doi.org/10.1016/j.dss.2013.02.006

Hasbrouck, J., & Saar, G. (2013). Low-latency trading. *Journal of Financial Markets*(0). doi: http://dx.doi.org/10.1016/j.finmar.2013.05.003

Hodson, J. A., & Zhang, J. Y. (2014). *Entity extraction and disambiguation in finance*. Paper presented at the Proceedings of the first international workshop on Entity recognition & disambiguation, Gold Coast, Queensland, Australia.

Hsinchun, C., & Zimbra, D. (2010). AI and Opinion Mining. *Intelligent Systems, IEEE, 25*(3), 74-80.

Huang, C.-J., Liao, J.-J., Yang, D.-X., Chang, T.-Y., & Luo, Y.-C. (2010). Realization of a news dissemination agent based on weighted association rules and text mining techniques. *Expert Syst. Appl., 37*(9), 6409-6413. doi: 10.1016/j.eswa.2010.02.078

Huang, S.-C., Chuang, P.-J., Wu, C.-F., & Lai, H.-J. (2010). Chaos-based support vector regressions for exchange rate forecasting. *Expert Syst. Appl., 37*(12), 8590-8598. doi: 10.1016/j.eswa.2010.06.001

Hutchison, M., & Sushko, V. (2013). Impact of macro-economic surprises on carry trade activity. *Journal of Banking & Finance, 37*(4), 1133-1147. doi: http://dx.doi.org/10.1016/j.jbankfin.2012.10.022

Ikeda, K., Hattori, G., Ono, C., Asoh, H., & Higashino, T. (2013). Twitter user profiling based on text and community mining for market analysis. *Knowledge-Based Systems, 51*(0), 35-47. doi: http://dx.doi.org/10.1016/j.knosys.2013.06.020

Jeong, Y., & Myaeng, S.-H. (2013). Using WordNet Hypernyms and Dependency Features for Phrasal-Level Event Recognition and Type Classification. In P. Serdyukov, P. Braslavski, S. Kuznetsov, J. Kamps, S. Rüger, E. Agichtein, I. Segalovich & E. Yilmaz (Eds.), *Advances in Information Retrieval* (Vol. 7814, pp. 267-278): Springer Berlin Heidelberg.

Jiang, S., Pang, G., Wu, M., & Kuang, L. (2012). An improved K-nearest-neighbor algorithm for text categorization. *Expert Systems with Applications, 39*(1), 1503-1509. doi: http://dx.doi.org/10.1016/j.eswa.2011.08.040

Jin, F., Self, N., Saraf, P., Butler, P., Wang, W., & Ramakrishnan, N. (2013). *Forex-foreteller: currency trend modeling using news articles*. Paper presented at the Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining, Chicago, Illinois, USA. http://doi.acm.org/10.1145/2487575.2487710

Joachims, T. (1998). Text categorization with Support Vector Machines: Learning with many relevant features. In C. Nédellec & C. Rouveirol (Eds.), *Machine Learning: ECML-98* (Vol. 1398, pp. 137-142): Springer Berlin Heidelberg.

Joachims, T. (1999). Making large-Scale {SVM} Learning Practical. In B. Schölkopf, C. Burges & A. Smola (Eds.), (pp. 169-184). Cambridge, MA: MIT Press.

Joachims, T. (2002). *Learning to Classify Text Using Support Vector Machines -- Methods, Theory, and Algorithms*: Kluwer/Springer.

Joseph, E. S. (1991). The Invisible Hand and Modern Welfare Economics: National Bureau of Economic Research, Inc.

Kaltwasser, P. R. (2010). Uncertainty about fundamentals and herding behavior in the FOREX market. *Physica A: Statistical Mechanics and its Applications, 389*(6), 1215-1222. doi: 10.1016/j.physa.2009.11.012

Kanayama, H., & Nasukawa, T. (2008). *Textual demand analysis: detection of users' wants and needs from opinions*. Paper presented at the Proceedings of the 22nd International Conference on Computational Linguistics - Volume 1, Manchester, United Kingdom.

Khadjeh Nassirtoussi, A., Aghabozorgi, S., Ying Wah, T., & Ngo, D. C. L. (2014). Text mining for market prediction: A systematic review. *Expert Systems with Applications, 41*(16), 7653-7670. doi: http://dx.doi.org/10.1016/j.eswa.2014.06.009

Khadjeh Nassirtoussi, A., Ying Wah, T., & Ngo Chek Ling, D. (2011). A novel FOREX prediction methodology based on fundamental data. *African Journal of Business Management, 5*(20), 8322-8330. doi: 10.5897/AJBM11.798

Kim, K.-j. (2003). Financial time series forecasting using support vector machines. *Neurocomputing, 55*(1–2), 307-319. doi: http://dx.doi.org/10.1016/S0925-2312(03)00372-2

Kim, K., Chung, B.-s., Choi, Y., Lee, S., Jung, J.-Y., & Park, J. (2014). Language independent semantic kernels for short-text classification. *Expert Systems with Applications, 41*(2), 735-743. doi: http://dx.doi.org/10.1016/j.eswa.2013.07.097

Kim, K., & Lee, J. (2014). Sentiment visualization and classification via semi-supervised nonlinear dimensionality reduction. *Pattern Recognition, 47*(2), 758-768. doi: http://dx.doi.org/10.1016/j.patcog.2013.07.022

King, M. R., Osler, C. L., & Rime, D. (2013). The market microstructure approach to foreign exchange: Looking back and looking forward. *Journal of International Money and Finance*(0). doi: http://dx.doi.org/10.1016/j.jimonfin.2013.05.004

Kleinnijenhuis, J., Schultz, F., Oegema, D. & Atteveldt, W.H. van. (2013). Financial News and Market Panics in the age of high frequency trading algorithms. *Journalism, 14*.

Kontopoulos, E., Berberidis, C., Dergiades, T., & Bassiliades, N. (2013). Ontology-based sentiment analysis of twitter posts. *Expert Systems with Applications, 40*(10), 4065-4074. doi: http://dx.doi.org/10.1016/j.eswa.2013.01.001

Kuang, P., Schröder, M., & Wang, Q. (2014). Illusory profitability of technical analysis in emerging foreign exchange markets. *International Journal of Forecasting, 30*(2), 192-205. doi: http://dx.doi.org/10.1016/j.ijforecast.2013.07.015

Kulkarni, A., Agarwal, K., Shah, P., Rathod, S. R., & Ramakrishnan, G. (2014). *System for collective entity disambiguation*. Paper presented at the Proceedings of the first international workshop on Entity recognition & disambiguation, Gold Coast, Queensland, Australia.

Lehalle, C.-A., & Laruelle, S. (2013). *Market Microstructure in Practice* (1 ed.): World Scientific Publishing.

Lewis, D. (1998). Naive (Bayes) at forty: The independence assumption in information retrieval. In C. Nédellec & C. Rouveirol (Eds.), *Machine Learning: ECML-98* (Vol. 1398, pp. 4-15): Springer Berlin Heidelberg.

Li, C. H., Yang, J. C., & Park, S. C. (2012). Text categorization algorithms using semantic approaches, corpus-based thesaurus and WordNet. *Expert Systems with Applications, 39*(1), 765-772. doi: http://dx.doi.org/10.1016/j.eswa.2011.07.070

Li, F. (2010). The Information Content of Forward-Looking Statements in Corporate Filings—A Naïve Bayesian Machine Learning Approach. *Journal of Accounting Research, 48*(5), 1049-1102. doi: 10.1111/j.1475-679X.2010.00382.x

Li, Q., & Chand, S. (2013). House prices and market fundamentals in urban China. *Habitat International, 40*(0), 148-153. doi: http://dx.doi.org/10.1016/j.habitatint.2013.04.002

Li, W., & Xu, H. (2014). Text-based emotion classification using emotion cause extraction. *Expert Systems with Applications, 41*(4, Part 2), 1742-1749. doi: http://dx.doi.org/10.1016/j.eswa.2013.08.073

Lipczak, M., Koushkestani, A., & Milios, E. (2014). *Tulip: lightweight entity recognition and disambiguation using wikipedia-based topic centroids*. Paper presented at the Proceedings of the first international workshop on Entity recognition & disambiguation, Gold Coast, Queensland, Australia.

Liu, Y., Loh, H. T., & Sun, A. (2009). Imbalanced text classification: A term weighting approach. *Expert Systems with Applications, 36*(1), 690-701. doi: http://dx.doi.org/10.1016/j.eswa.2007.10.042

Lodhi, H., Saunders, C., Shawe-Taylor, J., Cristianini, N., & Watkins, C. (2002). Text classification using string kernels. *J. Mach. Learn. Res., 2*, 419-444. doi: 10.1162/153244302760200687

Loia, V., & Senatore, S. (2014). A fuzzy-oriented sentic analysis to capture the human emotion in Web-based content. *Knowledge-Based Systems, 58*(0), 75-85. doi: http://dx.doi.org/10.1016/j.knosys.2013.09.024

Lu, X., & Yuan, Y. (2014). Hybrid structure for robust dimensionality reduction. *Neurocomputing, 124*(0), 131-138. doi: http://dx.doi.org/10.1016/j.neucom.2013.07.019

Lugmayr, A., & Gossen, G. (2012). *Evaluation of Methods and Techniques for Language Based Sentiment Analysis for DAX 30 Stock Exchange – A First Concept of a "LUGO" Sentiment Indicator*. Paper presented at the SAME 2012 – 5th International Workshop on Semantic Ambient Media Experience. http://www.tut.fi/emmi/WWW/ameamain/same2012

Luo, Q., Chen, E., & Xiong, H. (2011). A semantic term weighting scheme for text categorization. *Expert Systems with Applications, 38*(10), 12708-12716. doi: http://dx.doi.org/10.1016/j.eswa.2011.04.058

Lupiani-Ruiz, E., García-Manotas, I., Valencia-García, R., García-Sánchez, F., Castellanos-Nieves, D., Fernández-Breis, J. T., & Camón-Herrero, J. B. (2011). Financial news semantic search engine. *Expert Systems with Applications, 38*(12), 15565-15572. doi: http://dx.doi.org/10.1016/j.eswa.2011.06.003

Mabu, S., Hirasawa, K., Obayashi, M., & Kuremoto, T. (2013). Enhanced decision making mechanism of rule-based genetic network programming for creating stock trading signals. *Expert Systems with Applications, 40*(16), 6311-6320. doi: http://dx.doi.org/10.1016/j.eswa.2013.05.037

Mahajan, A., Dey, L., & Haque, S. M. (2008, 9-12 Dec. 2008). *Mining Financial News for Major Events and Their Impacts on the Market.* Paper presented at the Web Intelligence

and Intelligent Agent Technology, 2008. WI-IAT '08. IEEE/WIC/ACM International Conference on.
Maks, I., & Vossen, P. (2012). A lexicon model for deep sentiment analysis and opinion mining applications. *Decision Support Systems, 53*(4), 680-688. doi: http://dx.doi.org/10.1016/j.dss.2012.05.025
Miller, G. A. (1995). WordNet: a lexical database for English. *Commun. ACM, 38*(11), 39-41. doi: 10.1145/219717.219748
Mittermayer, M. A. (2004, 5-8 Jan. 2004). *Forecasting Intraday stock price trends with text mining techniques.* Paper presented at the System Sciences, 2004. Proceedings of the 37th Annual Hawaii International Conference on.
Mizrach, B., & Otsubo, Y. (2014). The market microstructure of the European climate exchange. *Journal of Banking & Finance, 39*(0), 107-116. doi: http://dx.doi.org/10.1016/j.jbankfin.2013.11.001
Moraes, F., Vasconcelos, M., Prado, P., Almeida, J., Gon, M., #231, & alves. (2013). *Polarity analysis of micro reviews in foursquare*. Paper presented at the Proceedings of the 19th Brazilian symposium on Multimedia and the web, Salvador, Brazil.
Moraes, R., Valiati, J. F., & Gavião Neto, W. P. (2013). Document-level sentiment classification: An empirical comparison between SVM and ANN. *Expert Systems with Applications, 40*(2), 621-633. doi: http://dx.doi.org/10.1016/j.eswa.2012.07.059
Mostafa, M. M. (2013). More than words: Social networks' text mining for consumer brand sentiments. *Expert Systems with Applications, 40*(10), 4241-4251. doi: http://dx.doi.org/10.1016/j.eswa.2013.01.019
Muehlfeld, K., Weitzel, U., & van Witteloostuijn, A. (2013). Fight or freeze? Individual differences in investors' motivational systems and trading in experimental asset markets. *Journal of Economic Psychology, 34*(0), 195-209. doi: http://dx.doi.org/10.1016/j.joep.2012.09.014
Nasir, J. A., Varlamis, I., Karim, A., & Tsatsaronis, G. (2013). Semantic smoothing for text clustering. *Knowledge-Based Systems, 54*(0), 216-229. doi: http://dx.doi.org/10.1016/j.knosys.2013.09.012
Nikfarjam, A., Emadzadeh, E., & Muthaiyah, S. (2010, 26-28 Feb. 2010). *Text mining approaches for stock market prediction.* Paper presented at the Computer and Automation Engineering (ICCAE), 2010 The 2nd International Conference on.
Nizer, P. S. M., & Nievola, J. C. (2012). Predicting published news effect in the Brazilian stock market. *Expert Systems with Applications, 39*(12), 10674-10680. doi: http://dx.doi.org/10.1016/j.eswa.2012.02.162
Olson, D., & Delen, D. (2008). Support Vector Machines *Advanced Data Mining Techniques* (pp. 111-123): Springer Berlin Heidelberg.
Ortigosa-Hernández, J., Rodríguez, J. D., Alzate, L., Lucania, M., Inza, I., & Lozano, J. A. (2012). Approaching Sentiment Analysis by using semi-supervised learning of multi-dimensional classifiers. *Neurocomputing, 92*(0), 98-115. doi: http://dx.doi.org/10.1016/j.neucom.2012.01.030
Peramunetilleke, D., & Wong, R. K. (2002). Currency exchange rate forecasting from news headlines. *Aust. Comput. Sci. Commun., 24*(2), 131-139. doi: 10.1145/563932.563921
Pestov, V. (2013). Is the -NN classifier in high dimensions affected by the curse of dimensionality? *Computers & Mathematics with Applications, 65*(10), 1427-1437. doi: http://dx.doi.org/10.1016/j.camwa.2012.09.011
Platt, J. C. (1999). Fast training of support vector machines using sequential minimal optimization *Advances in kernel methods* (pp. 185-208): MIT Press.
Potì, V., & Siddique, A. (2013). What drives currency predictability? *Journal of International Money and Finance, 36*(0), 86-106. doi: http://dx.doi.org/10.1016/j.jimonfin.2013.03.004
Premanode, B., & Toumazou, C. (2013). Improving prediction of exchange rates using Differential EMD. *Expert Systems with Applications, 40*(1), 377-384. doi: http://dx.doi.org/10.1016/j.eswa.2012.07.048
Pui Cheong Fung, G., Xu Yu, J., & Wai, L. (2003, 20-23 March 2003). *Stock prediction: Integrating text mining approach using real-time news.* Paper presented at the

Computational Intelligence for Financial Engineering, 2003. Proceedings. 2003 IEEE International Conference on.

Quinlan, J. R. (1993). *C4.5: programs for machine learning*: Morgan Kaufmann Publishers Inc.

Rachlin, G., Last, M., Alberg, D., & Kandel, A. (2007, March 1 2007-April 5 2007). *ADMIRAL: A Data Mining Based Financial Trading System.* Paper presented at the Computational Intelligence and Data Mining, 2007. CIDM 2007. IEEE Symposium on.

Reboredo, J. C., Rivera-Castro, M. A., Miranda, J. G. V., & García-Rubio, R. (2013). How fast do stock prices adjust to market efficiency? Evidence from a detrended fluctuation analysis. *Physica A: Statistical Mechanics and its Applications, 392*(7), 1631-1637. doi: http://dx.doi.org/10.1016/j.physa.2012.11.038

Robertson, C., Geva, S., & Wolff, R. (2006). *What types of events provide the strongest evidence that the stock market is affected by company specific news?* Paper presented at the Proceedings of the fifth Australasian conference on Data mining and analystics - Volume 61, Sydney, Australia.

Salzberg, S. (1997). On Comparing Classifiers: Pitfalls to Avoid and a Recommended Approach. *Data Mining and Knowledge Discovery, 1*(3), 317-328. doi: 10.1023/A:1009752403260

Sankaraguruswamy, S., Shen, J., & Yamada, T. (2013). The relationship between the frequency of news release and the information asymmetry: The role of uninformed trading. *Journal of Banking & Finance, 37*(11), 4134-4143. doi: http://dx.doi.org/10.1016/j.jbankfin.2013.07.026

Schumaker, R. P., & Chen, H. (2009). Textual analysis of stock market prediction using breaking financial news: The AZFin text system. *ACM Trans. Inf. Syst., 27*(2), 1-19. doi: 10.1145/1462198.1462204

Schumaker, R. P., Zhang, Y., Huang, C.-N., & Chen, H. (2012). Evaluating sentiment in financial news articles. *Decision Support Systems*(0). doi: 10.1016/j.dss.2012.03.001

Sermpinis, G., Laws, J., Karathanasopoulos, A., & Dunis, C. L. (2012). Forecasting and trading the EUR/USD exchange rate with Gene Expression and Psi Sigma Neural Networks. *Expert Systems with Applications, 39*(10), 8865-8877. doi: http://dx.doi.org/10.1016/j.eswa.2012.02.022

Shi, K., He, J., Liu, H.-t., Zhang, N.-t., & Song, W.-t. (2011). Efficient text classification method based on improved term reduction and term weighting. *The Journal of China Universities of Posts and Telecommunications, 18, Supplement 1*(0), 131-135. doi: http://dx.doi.org/10.1016/S1005-8885(10)60196-3

Shou-Hsiung, C. (2010, 11-14 July 2010). *Forecasting the change of intraday stock price by using text mining news of stock.* Paper presented at the Machine Learning and Cybernetics (ICMLC), 2010 International Conference on.

Sidorov, G., Velasquez, F., Stamatatos, E., Gelbukh, A., & Chanona-Hernández, L. (2013). Syntactic N-grams as machine learning features for natural language processing. *Expert Systems with Applications*(0). doi: http://dx.doi.org/10.1016/j.eswa.2013.08.015

Smales, L. A. (2012). Order imbalance, market returns and macroeconomic news: Evidence from the Australian interest rate futures market. *Research in International Business and Finance, 26*(3), 410-427. doi: http://dx.doi.org/10.1016/j.ribaf.2012.04.001

Soni, A., van Eck, N. J., & Kaymak, U. (2007, 1-5 April 2007). *Prediction of Stock Price Movements Based on Concept Map Information.* Paper presented at the Computational Intelligence in Multicriteria Decision Making, IEEE Symposium on.

Stede, M. (2000). *The hyperonym problem revisited: conceptual and lexical hierarchies in language generation*. Paper presented at the Proceedings of the first international conference on Natural language generation - Volume 14, Mitzpe Ramon, Israel.

Tan, S., Wang, Y., & Wu, G. (2011). Adapting centroid classifier for document categorization. *Expert Systems with Applications, 38*(8), 10264-10273. doi: http://dx.doi.org/10.1016/j.eswa.2011.02.114

Tang, H.-j., Yan, D.-f., & Tian, Y. (2013). Semantic dictionary based method for short text classification. *The Journal of China Universities of Posts and Telecommunications, 20, Supplement 1*(0), 15-19. doi: http://dx.doi.org/10.1016/S1005-8885(13)60256-3

Taşcı, Ş., & Güngör, T. (2013). Comparison of text feature selection policies and using an adaptive framework. *Expert Systems with Applications, 40*(12), 4871-4886. doi: http://dx.doi.org/10.1016/j.eswa.2013.02.019

Tay, F. E. H., & Cao, L. (2001). Application of support vector machines in financial time series forecasting. *Omega, 29*(4), 309-317. doi: http://dx.doi.org/10.1016/S0305-0483(01)00026-3

ter Ellen, S., Verschoor, W. F. C., & Zwinkels, R. C. J. (2013). Dynamic expectation formation in the foreign exchange market. *Journal of International Money and Finance, 37*(0), 75-97. doi: http://dx.doi.org/10.1016/j.jimonfin.2013.06.001

Tetlock, P. C. (2007). Giving content to investor sentiment: The role of media in the stock market. *The Journal of Finance, 62*(3), 1139-1168.

Tetlock, P. C., Saar-Tsechansky, M., & Macskassy, S. (2008). More Than Words: Quantifying Language to Measure Firms' Fundamentals. *The Journal of Finance, 63*(3), 1437-1467. doi: 10.1111/j.1540-6261.2008.01362.x

Thammasiri, D., Delen, D., Meesad, P., & Kasap, N. (2014). A critical assessment of imbalanced class distribution problem: The case of predicting freshmen student attrition. *Expert Systems with Applications, 41*(2), 321-330. doi: http://dx.doi.org/10.1016/j.eswa.2013.07.046

Tsai, C.-F., Eberle, W., & Chu, C.-Y. (2013). Genetic algorithms in feature and instance selection. *Knowledge-Based Systems, 39*(0), 240-247. doi: http://dx.doi.org/10.1016/j.knosys.2012.11.005

Urquhart, A., & Hudson, R. (2013). Efficient or adaptive markets? Evidence from major stock markets using very long run historic data. *International Review of Financial Analysis, 28*(0), 130-142. doi: http://dx.doi.org/10.1016/j.irfa.2013.03.005

Uysal, A. K., & Gunal, S. (2012). A novel probabilistic feature selection method for text classification. *Knowledge-Based Systems, 36*(0), 226-235. doi: http://dx.doi.org/10.1016/j.knosys.2012.06.005

Uysal, A. K., & Gunal, S. (2014). The impact of preprocessing on text classification. *Information Processing & Management, 50*(1), 104-112. doi: http://dx.doi.org/10.1016/j.ipm.2013.08.006

Vanstone, B., & Finnie, G. (2010). Enhancing stockmarket trading performance with ANNs. *Expert Systems with Applications, 37*(9), 6602-6610. doi: 10.1016/j.eswa.2010.02.124

Vlachos, M. (2010). Dimensionality Reduction on Text via Feature Selection. In C. Sammut & G. Webb (Eds.), *Encyclopedia of Machine Learning* (pp. 279-279): Springer US.

Vu, T. T., Chang, S., Ha, Q. T., & Collier, N. (2012). An Experiment in Integrating Sentiment Features for Tech Stock Prediction in Twitter. *Proceedings of the Workshop on Information Extraction and Entity Analytics on Social Media Data*. Mumbai, India: The COLING 2012 Organizing Committee.

Wang, G.-J., Xie, C., & Han, F. (2012). Multi-Scale Approximate Entropy Analysis of Foreign Exchange Markets Efficiency. *Systems Engineering Procedia, 3*(0), 201-208. doi: http://dx.doi.org/10.1016/j.sepro.2011.10.030

Wang, K.-L., Fawson, C., Chen, M.-L., & Wu, A.-C. (2014). Characterizing Information Flows Among Spot, Deliverable Forward and Non-Deliverable Forward Exchange Rate Markets: A Cross-Country Comparison. *Pacific-Basin Finance Journal*(0). doi: http://dx.doi.org/10.1016/j.pacfin.2014.01.002

Weiss, S. M., Indurkhya, N., & Zhang, T. (2010). *Fundamentals of Predictive Text Mining*.

Werner, A., & Myrray Z., F. (2004). Is All That Talk Just Noise ? The Information Content of Internet Stock Message Boards. *Journal of Finance*, 1259--1294.

Wisniewski, T. P., & Lambe, B. (2013). The role of media in the credit crunch: The case of the banking sector. *Journal of Economic Behavior & Organization, 85*(0), 163-175. doi: http://dx.doi.org/10.1016/j.jebo.2011.10.012

Witten, I. H., & Frank, E. (2005). *Data Mining: Practical Machine Learning Tools and Techniques, Second Edition (Morgan Kaufmann Series in Data Management Systems)*: Morgan Kaufmann Publishers Inc.

Wu, C., Lu, W., & Zhou, P. (2014). *An optimization framework for entity recognition and disambiguation*. Paper presented at the Proceedings of the first international workshop on Entity recognition & disambiguation, Gold Coast, Queensland, Australia.

Wu, Q., & Tan, S. (2011). A two-stage framework for cross-domain sentiment classification. *Expert Systems with Applications, 38*(11), 14269-14275. doi: http://dx.doi.org/10.1016/j.eswa.2011.04.240

Wuthrich, B., Cho, V., Leung, S., Permunetilleke, D., Sankaran, K., & Zhang, J. (1998, 11-14 Oct 1998). *Daily stock market forecast from textual web data.* Paper presented at the Systems, Man, and Cybernetics, 1998. 1998 IEEE International Conference on.

Yang, L., Li, C., Ding, Q., & Li, L. (2013). Combining Lexical and Semantic Features for Short Text Classification. *Procedia Computer Science, 22*(0), 78-86. doi: http://dx.doi.org/10.1016/j.procs.2013.09.083

Yin, L., Ge, Y., Xiao, K., Wang, X., & Quan, X. (2013). Feature selection for high-dimensional imbalanced data. *Neurocomputing, 105*(0), 3-11. doi: http://dx.doi.org/10.1016/j.neucom.2012.04.039

Yin, W., & Li, J. (2014). Macroeconomic fundamentals and the exchange rate dynamics: A no-arbitrage macro-finance approach. *Journal of International Money and Finance, 41*(0), 46-64. doi: http://dx.doi.org/10.1016/j.jimonfin.2013.10.004

Young, L., & Soroka, S. (2012). Affective News: The Automated Coding of Sentiment in Political Texts. *Political Communication, 29*, 205-231.

Yu, H., Nartea, G. V., Gan, C., & Yao, L. J. (2013). Predictive ability and profitability of simple technical trading rules: Recent evidence from Southeast Asian stock markets. *International Review of Economics & Finance, 25*(0), 356-371. doi: http://dx.doi.org/10.1016/j.iref.2012.07.016

Yu, L.-C., Wu, J.-L., Chang, P.-C., & Chu, H.-S. (2013). Using a contextual entropy model to expand emotion words and their intensity for the sentiment classification of stock market news. *Knowledge-Based Systems*(0). doi: http://dx.doi.org/10.1016/j.knosys.2013.01.001

Yu, Y., Duan, W., & Cao, Q. (2013). The impact of social and conventional media on firm equity value: A sentiment analysis approach. *Decision Support Systems*(0). doi: http://dx.doi.org/10.1016/j.dss.2012.12.028

Zhai, Y., Hsu, A., & Halgamuge, S. K. (2007). *Combining News and Technical Indicators in Daily Stock Price Trends Prediction*. Paper presented at the Proceedings of the 4th international symposium on Neural Networks: Advances in Neural Networks, Part III, Nanjing, China.

Zhou, S., Chen, Q., & Wang, X. (2013). Active deep learning method for semi-supervised sentiment classification. *Neurocomputing, 120*(0), 536-546. doi: http://dx.doi.org/10.1016/j.neucom.2013.04.017

8 Appendices

8.1 Appendix A: Additional Experiment on Numeric Fundamental Data

An additional experiment is conducted by the researcher in the course of this work whose results are published in Arman Khadjeh Nassirtoussi, Ying Wah, and Ngo Chek Ling (2011).

The experiment takes on fundamental data that is manifested in numeric form. It differentiates itself with the core of this research that is presented in this thesis in that the focus is on numeric and not textual data. However, what is common between this experiment and the rest of this work is an attention to and utilization of fundamental data. In essence the researcher conducts this experiment to explore the relationship between FOREX and fundamental data in numeric format before approaching the relationship between FOREX and fundamental data in textual format. The reason is that what this work is putting to test is twofold: firstly, the relation between fundamental data as a whole and FOREX and secondly, the textual manifestation of this data.

A brief overview of this experiment and its results is presented in the following in this appendix.

FOREX & NATIONAL ECONOMIC DATA

This work at first hypothesizes about existence of possible relationships between movements of a currency pair in FOREX as an example for a fluctuating price point in the market and the national economic data for the relevant countries as possible fundamental data that are external to the FOREX market charts. The objective is to devise a mechanism that can identify plausible relationships between specific economic data and the price moves of the currency pair with a precision. Hence, the objective is two fold: Firstly to propose a mechanism to put existence of such relation to test and

secondly, to put a number of different types of economic data to test and identify the ones with a relationship that can be observed.

The strongest currencies in the market are USD, Euro and Pound Sterling. Hence, possible data sources for national economic data for the U.S., Europe and the U.K. are identified. The most convenient and useful sources for the purpose of this research are determined based on the comprehensiveness and reliability of their data, their presented data format and their historic data availability as well as their frequency of data release. The successful sources based on the above criteria are: 1- Bureau of Economic Analysis - U.S. Department of Commerce (http://www.bea.gov), 2- Bank of England (http://www.bankofengland.co.uk)

BEA is an agency of the Department of Commerce. Along with the Census Bureau and STAT-USA, BEA is part of the Department's Economics and Statistics Administration. BEA produces economic accounts statistics that enable government and business decision-makers, researchers, and the American public to follow and understand the performance of the Nation's economy. A more comprehensive introduction to BAE can be found at its mission statement website page at http://www.bea.gov/about/mission.htm.

On the other hand, the Bank of England is the central bank of the United Kingdom which is the center of the UK's financial system; the Bank contributes to promoting and maintaining monetary and financial stability. Both of the above institutions provide the exact kind of data that is required as input for the experiments in this research.

Since the above two sources present financial data about the U.S. and England respectively, intuitively the currency pair of USD/GPD is chosen as the currency pair whose price moves are to be monitored in the experiments. The fundamental data that can be derived about the U.S. economics from the former data source and about the UK

economics from the latter one is to be investigated for relationships with the pair's value. This is the relationship that is hypothesized about its existence and is to be identified and put to test. If the test succeeds, the currency pair moves can be predicted with a known precision based on the economic data.

Next is to identify some economic data among the many sets of the available data in the above sources. The primary criteria taken into consideration for choice of the data sets are as follows: Firstly, there needs to be an intuitive relationship between the fundamental data set and the currency pair USD/GBP. All data sets are related to the U.S. or the UK economics and are supposed to have an impact on the currency pair or be impacted by it but some seem to be better choices at least intuitively and those are given priority in this experiment. Secondly, the data set is to be available on a monthly basis as opposed to many fundamental data sets that are available only on a yearly basis, so that relatively shorter terms can be explored which are more attractive for prediction purposes. Thirdly, the monthly data should be available for the same period for all data. (Which in this experiment is from February 1996 onwards)

Based on the above criteria 3 main sources for data sets are identified, which are: 1- UK International Reserves (in US Dollar Millions) from Bank of England, 2- U.S. National Retail and Food Services Sales from Bureau of Economic Analysis, 3- U.S. International Trade in Goods and Services (Total Import and Export) from Bureau of Economic Analysis. These data sets were available in the needed frequency for the period set for this experiment. Moreover, they were intuitively very relevant to USD/GPD moves. This is expanded a bit more in the following.

UK International Reserves is any kind of reserve funds that can be passed between the central banks of the UK and other countries as an acceptable form of payment. The reserves can either be gold or a specific currency, such as the dollar in this case. Buying

and selling official international reserves by central banks throughout the world may influence exchange rates. The quantity of foreign exchange reserves can change as a central bank implements monetary policy. Hence, there should be a solid relationship between the released data on international reserves and the fluctuations of the USD/GBP. There are multiple indices with regards to the internal reserves, each pertaining to a specific aspect of impact, the following were chosen for the purpose of this research based on data availability and plausibility of the relationship based on the above:

1-Monthly amounts outstanding of Central Government foreign currency total reserves (in US dollar millions) not seasonally adjusted.

2- Monthly amounts outstanding of Central Government IMF reserve tranche position total in special drawing rights (in US dollar millions) not seasonally adjusted.

3- Monthly amounts outstanding of Central Government Gold swapped or on loan total (in US dollar millions) not seasonally adjusted.

4- Monthly amounts outstanding of Central Government all foreign currency forwards and swaps (incl sterling leg) total (in US dollar millions) not seasonally adjusted.

5- Monthly amounts outstanding of Bank of England Banking Department all foreign currency total bills issued (in US dollar millions) not seasonally adjusted.

6- Quarterly amounts outstanding of Bank of England Banking Department total US dollar assets (in US dollar millions) not seasonally adjusted. The above were

used as part of the input for the experiments in this work as indicators on international reserves.

Next identified data set is U.S. National Retail and Food Services Sales from Bureau of Economic Analysis. Retail sales occur when businesses sell goods or services to households. How much is spent on retail and food services by consumers is tied closely with purchasing power and economic growth. It is plausible to assume the strength of the U.S. currency can have a relationship with the National Retail and Food Services Sales.

Next proposed data set for experiment is U.S. International Trade in Goods and Services (Total Import and Export) from Bureau of Economic Analysis. In this category two indicators are used, firstly, the balance of import and export in goods and services. The total export of goods and services minus the total import of the goods and services on a monthly basis, composes the balance of import and export. Second is the total of monthly export of goods and services in the U.S.

The above are the 3 categories of fundamental data found in statistical resources. Intuitively relationships are plausible between any of them and the currency pair moves or between a combination of them and the pair's moves. This is put to test to identify if such relationship exists, which potentially can be used for forecasting. Nevertheless, the restrictions which were imposed in selection of these datasets should not be dismissed. Prediction of market behaviour on a monthly basis requires availability of the kind of fundamental data that is required in this approach that is numeric data that is released in periodic reports of official financial organizations. Such data at the required frequency is not easily available as many of the official reports are of longer intervals. Furthermore, the data had to be available for the same period of time for all the sources that is from February 1996 onwards. Hence, such use of numeric fundamental data

extracted from periodic reports is limited but it is novel and this study demonstrates its effectiveness.

EXPERIMENT DESIGN AND METHODOLOGY

6 experiments are designed based on the above specified 3 sources of data. All input is monthly values starting from February 1996 to March 2010. That is a total of 170 values for each of the criteria that are to train the Neural Networks. So firstly, datasets are chosen that start at least from February 1996. Secondly, they are available on a monthly basis. The extra data trail before and after these dates are omitted. Each of the criteria has 3 entries for each month:

1- The date

2- The name of the factor

3- The value

The name of the factor or one of the criteria that is used for training the Neural Networks in GoldenGem is to be called a ticker from now on, because the program in its default mode uses other stock market tickers to train the Neural Networks for the prediction of a particular ticker. And in these experiments we are using fundamental data posed in the above particular structure and the name of the data factor that is used for training would be the so called ticker in this context.

The input values for a particular ticker would look like below:

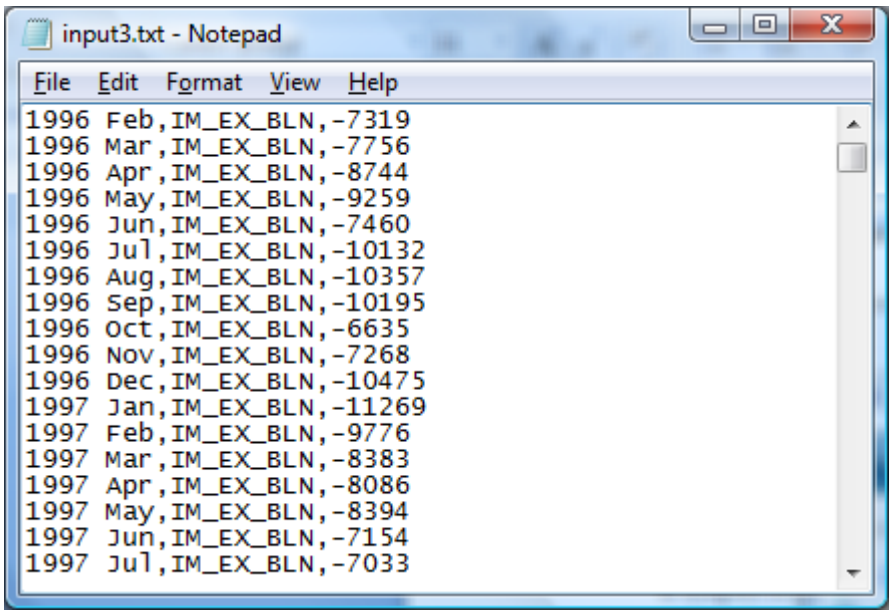

Figure 8.1 Import export balance as a monthly ticker value fed into the neural networks in a text file

For each experiment an input text file is created that has the date, ticker name and the value as above for all the months in the mentioned time period and for all the tickers, meaning, if we have a combination of six tickers to train the neural networks, all of them are put in the same file one after the other and also the available values for USD/GBP for the same period are placed in the same file in the same format.

Later on in GoldenGem under ‘related group of tickers’ field which is in the bottom right corner of the program console as you can see below, all tickers are to be mentioned, separated by comma. Then when the text file is loaded using the file menu, the tickers can be seen in the drop down menu on the right hand side, too. There one can choose which of the tickers is to be predicted and the rest are used as training data and prediction input.

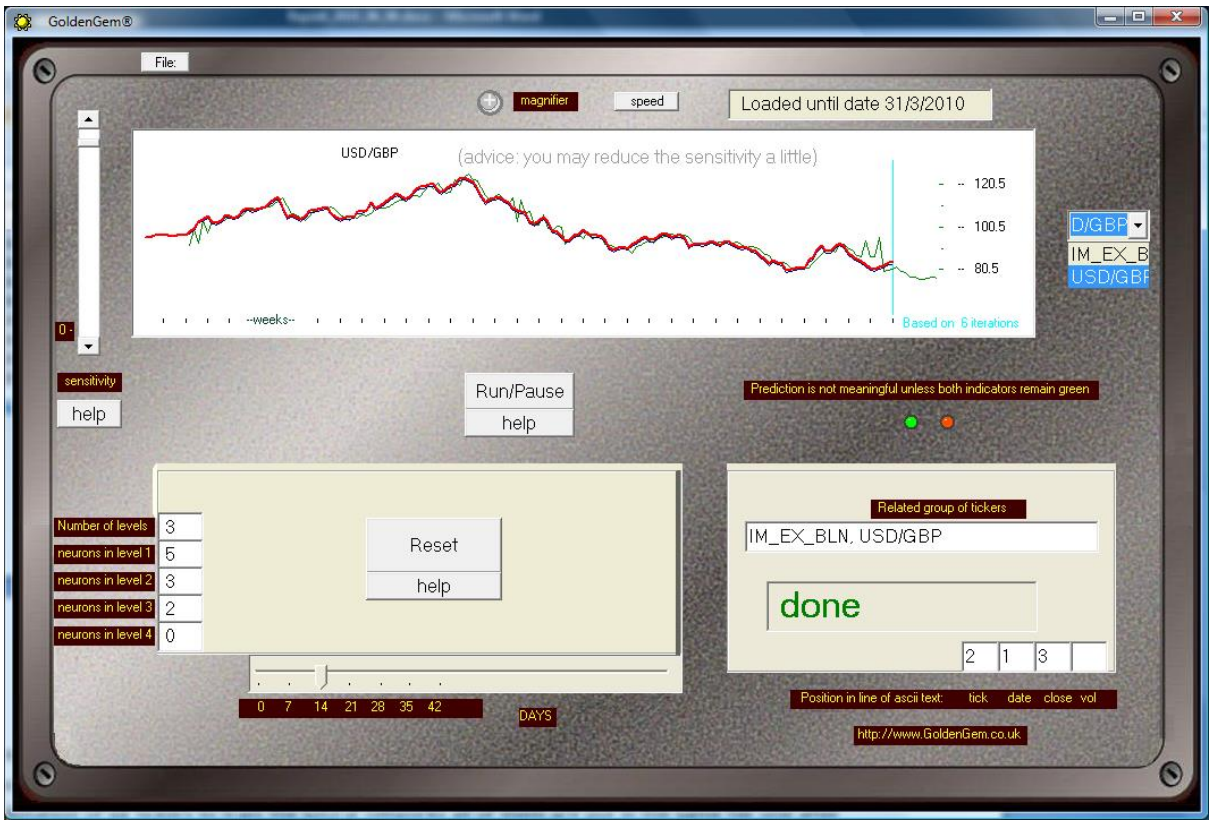

Figure 8.2 GoldenGem's output for the relationship between the import export monthly value and USD/GBP

On the slide at the bottom left number of days to be predicted are set, in our context because the data is presented in a monthly format that would be the number of months for which the particular ticker is to be predicted. In this setting the prediction is for the next 14 months.

The slide on the top left is for adjusting the sensitivity level to the real ticker value during the learning process. At the end of the learning process the two indicator lights need to be green while the sensitivity is set to zero. This means that the green graph which is the prediction is created by the learned neural networks and is blind to the actual values of the ticker but it matches the actual value (blue/red graph) in an acceptable proximity.

METHOD OF CONDUCT

After having set the ticker names and having loaded the input file, the sensitivity is adjusted to the highest. The ticker that is to be predicted is chosen and the iterations start. There are two indicator lights as long as the left one is green the level of sensitivity can be reduced little by little and eventually it can be set to zero. Then if the left light is still green after a few iterations the right light may go green two. As soon as the two light are green, it is accepted that the data sets can predict the particular ticker (USD/GBP). Otherwise the group of tickers is not able to predict the particular ticker.

RESULTS AND DISCUSSION

A total of 6 experiments were conducted by providing the different sets of available fundamental data as input to the neural networks. This accumulated history data that is fed to the networks is used for training it. If relationships exist between the input and the currency pair moves, after limited number of iterations the networks reaches a "learned" state. This indicates that based on the input alone the neural networks is capable of predicting the currency pair's price.

As shown in the below table in half of the experiments the neural networks reached a "learned" state. The neural networks did not manage to identify any relationship between different aspects of monthly UK international reserves as combined input, nor was any relationship found for the monthly balance of import and export in goods and services and the total monthly export of goods and services in the U.S.

However, the Monthly U.S. retail and food sales proved to have a relationship with the currency pair's moves which was detected by the neural networks. This indicates the sensitivity of domestic US markets to international currency markets. The money spent on retail and food services by consumers is tied closely with purchasing power and economic growth. The identified relationship by the neural networks proves that there is

a clear relationship between the strength of the US currency and the national retail and food services sales, most probably because when US economy and people's purchasing power is on the rise more retail and food service purchases are made and the currency value behaves accordingly. Furthermore, interestingly experiments 5 and 6 did manage to bring the neural networks to a "learned" state. These two experiments are special because the inputs for both of them are combined input elements which have been used in other experiments and have not lead to a learned state. Both this work finds that the combination of those input sets and re-feeding them into the neural networks proves to be able to train the neural networks.

This proves that relationship between fundamental data and currency pair moves is of course very complicated, however, if different facets are put together and fundamental data is combined from different sources, neural networks can detect predictability. As in experiment 5, in which international reserves monthly data for the UK, combined with the monthly balance of import and export in goods and services in the U.S. surprisingly manages to bring the neural networks to a "learned" state. Furthermore, in experiment 6, the input for experiments 3 and 4 are combined, that is, the monthly balance of import and export in goods and services in the U.S. and the total of monthly export of goods and services in the U.S. and again a positive result is gained which indicates predictability after combination of data.

Table 8.1 The experiment results on achieving "learned state" by the neural networks for different inputs

Experiment	Input	Learned State for NN
1	Different Aspects of Monthly UK International Reserves (in US Dollar Millions)	NO
2	Monthly U.S. Retail and Food Services Sales	YES
3	The Monthly Balance of Import and Export in Goods and Services in the U.S.	NO
4	Total of Monthly Export of Goods and Services in the U.S.	NO
5	The Combination of the Input of Experiment 1 and 3	YES
6	The Combination of the Input of Experiments 3 and 4	YES

CONCLUSION

In this work an effort is made to explore the possibilities of using fundamental data to predict currency price moves in the foreign exchange market. Such prediction is very much in demand; however, technical analysis is the approach that is widely looked at in research in this area. This work introduces an approach that can be undertaken for integration of fundamental analysis in automated prediction. The proposed approach in this work that resides on utilization of neural networks proves to be successful through the conducted experiments. The experiment results indicate solid plausibility in determining currency moves through the proposed methodology and with the use of the identified input.

In addition to identification of some fundamental data that can be used for such prediction and proposing a methodology, this work also manages to demonstrate though the conducted experiments that while a set of fundamental data might not be indicative of price moves on its own, it might very well contribute to determination of such indication when combined with other sets of such data. This clearly demonstrates the multitude of aspects of information that are involved, and points to the direction of combining different possible fundamental data inputs in order to get the best results.

The feasibility of the act of taking such multiple aspects as input and producing an indicative output having taken all of that into consideration becomes only possible with the help of neural networks. A successful example of such use of neural networks is demonstrated in this work.

The experiments conducted on the sets of fundamental data showed that there is a plausible chance for prediction of the currency pair USD/GDP based on such data. Although at times (experiments 1, 3 and 4) individual fundamental factors as input prove to be ineffective predictors independently but in those cases a combination could be formed of such data sets which has strong prediction capability. This prediction capability can be observed and learned by Neural Networks.

Therefore, this work produces an initial outlook on a new methodology for prediction of market moves based on fundamental data and also identifies a few data sources which prove to be effective for prediction through the proposed methodology.

8.2 Appendix B: Prototype Flow Screenshots

The two phases discussed in the previous appendix produce the below screenshots in the visual section of RapidMiner.

Preparation:

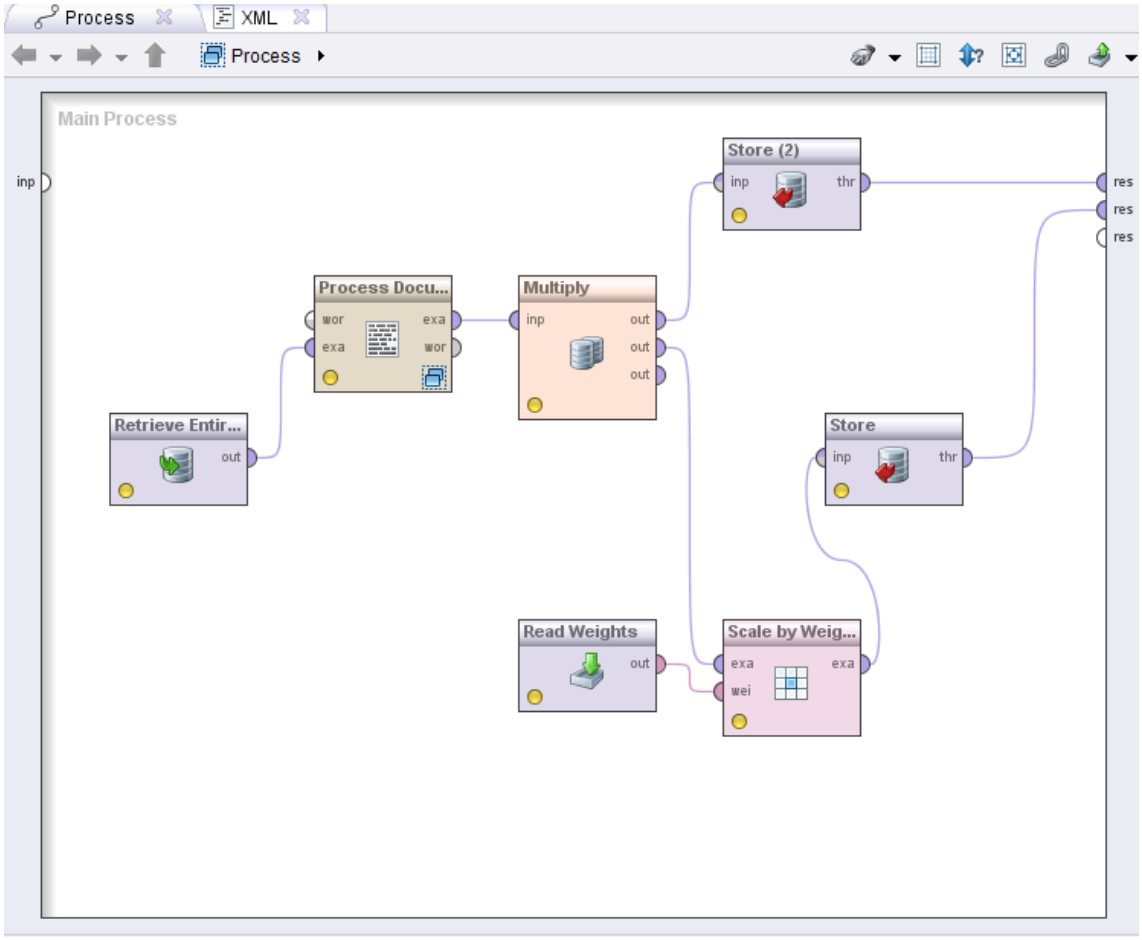

Figure 8.3 High-Level Feature Matrix Preparation

Process:

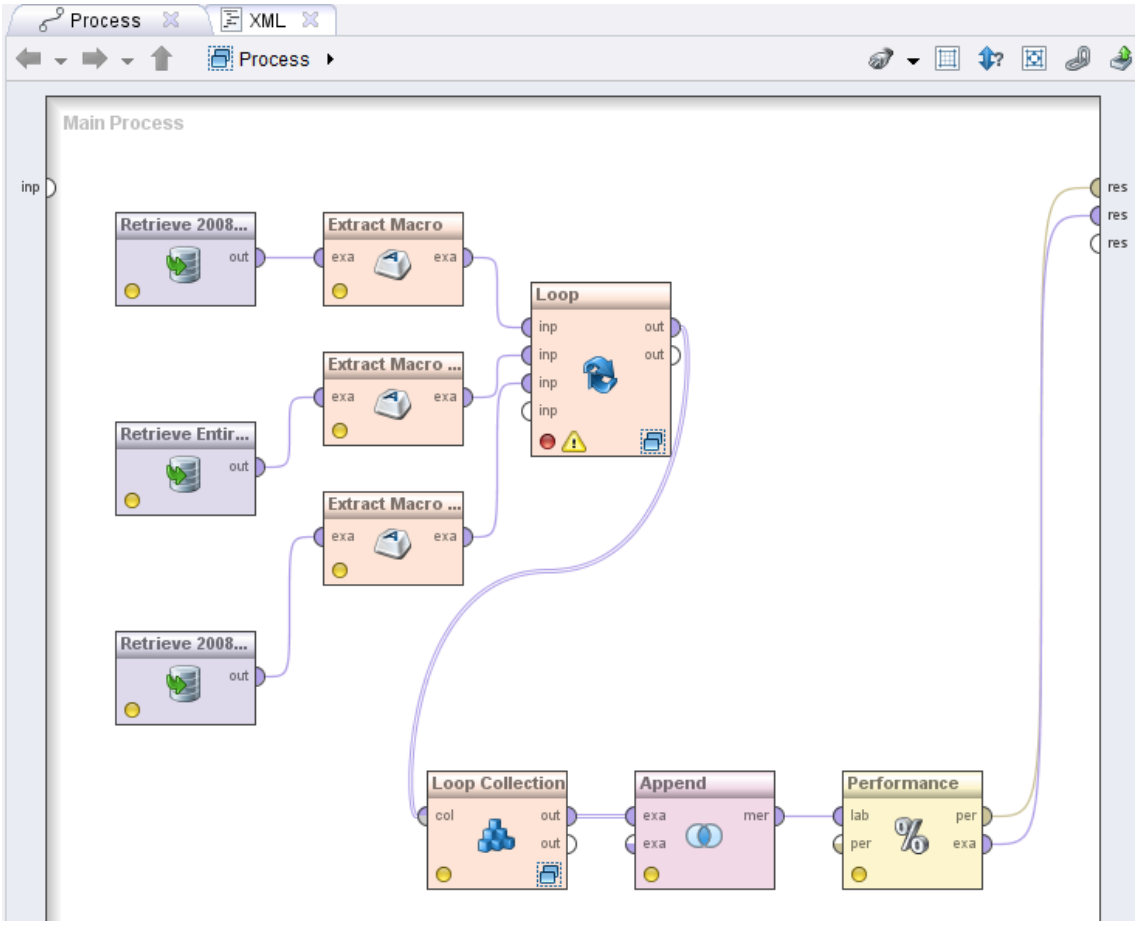

Figure 8.4 High-Level Feature Matrix Processing

8.3 Appendix C: Journal Publications

In the course of this work 3 journal papers have been published as below:

Khadjeh Nassirtoussi, A., Aghabozorgi, S., Ying Wah, T., & Ngo, D. C. L. (2015). Text mining of news-headlines for FOREX market prediction: A Multi-layer Dimension Reduction Algorithm with semantics and sentiment. *Expert Systems with Applications, 42*(1), 306-324. doi: http://dx.doi.org/10.1016/j.eswa.2014.08.004

Khadjeh Nassirtoussi, A., Aghabozorgi, S., Ying Wah, T., & Ngo, D. C. L. (2014). Text mining for market prediction: A systematic review. *Expert Systems with Applications, 41*(16), 7653-7670. doi: http://dx.doi.org/10.1016/j.eswa.2014.06.009

Khadjeh Nassirtoussi, A., Ying Wah, T., & Ngo Chek Ling, D. (2011). A novel FOREX prediction methodology based on fundamental data. *African Journal of Business Management, 5*(20), 8322-8330. doi: 10.5897/AJBM11.798

Table 8.2 Ranking of the ISI Journal Expert Systems with Applications

Category Name	Total Journals in Category	Journal Rank in Category	Quartile in Category
COMPUTER SCIENCE, ARTIFICIAL INTELLIGENCE	121	30	Q1
ENGINEERING, ELECTRICAL & ELECTRONIC	247	63	Q2
OPERATIONS RESEARCH & MANAGEMENT SCIENCE	79	11	Q1

8.4 Appendix D: Code for Prototype Reproduction

The below provided XML code can be used to reproduce the prototype system in RapidMiner. The code is segmented into two portions: firstly, the preparation phase whereby the feature matrix is prepared in the required format for the algorithms and

secondly, the process phase, whereby the classification is conducted and prediction results are produced.

Code for Preparation Phase:

<?xml version="1.0" encoding="UTF-8" standalone="no"?>

<process version="5.3.013">

<context>

<input/>

<output/>

<macros/>

</context>

<operator activated="true" class="process" compatibility="5.3.013" expanded="true" name="Process">

<process expanded="true">

<operator activated="true" class="read_weights" compatibility="5.3.013" expanded="true" height="60" name="Read Weights" width="90" x="313" y="345">

<parameter key="attribute_weights_file" value="D:\Dropbox\Arman\SentiWordNet_3.0.0\home\swn\www\admin\dump\Weight Prep\NewScores\SumScore\SumScoreComplete.xml"/>

</operator>

<operator activated="true" class="retrieve" compatibility="5.3.013" expanded="true" height="60" name="Retrieve EntireNewNoTimeInorder2008_11 (2)" width="90" x="45" y="210">

<parameter key="repository_entry" value="EntireNewNoTimeInorder2008_11"/>

</operator>

<operator activated="true" class="text:process_document_from_data" compatibility="5.3.002" expanded="true" height="76" name="Process Documents from Data (3)" width="90" x="179" y="120">

<list key="specify_weights"/>

<process expanded="true">

<operator activated="true" class="text:tokenize" compatibility="5.3.002" expanded="true" height="60" name="Tokenize (3)" width="90" x="45" y="30"/>

<operator activated="true" class="text:filter_stopwords_english" compatibility="5.3.002" expanded="true" height="60" name="Filter Stopwords (3)" width="90" x="179" y="30"/>

<operator activated="true" class="wordnet:open_wordnet_dictionary" compatibility="5.2.000" expanded="true" height="60" name="Open WordNet Dictionary (3)" width="90" x="179" y="300">

<parameter key="directory" value="D:\Dropbox\Arman\WNdb-3.0\dict"/>

</operator>

<operator activated="false" class="wordnet:stem_wordnet" compatibility="5.2.000" expanded="true" height="76" name="Stem (WordNet)" width="90" x="179" y="165"/>

<operator activated="true" class="wordnet:find_hypernym_wordnet" compatibility="5.2.000" expanded="true" height="76" name="Find Hypernyms (3)" width="90" x="380" y="165">

<parameter key="use_prefix" value="false"/>

<parameter key="multiple_meanings_per_word_policy" value="Take only first meaning"/>

<parameter key="take_ID_instead_of_words" value="true"/>

</operator>

<connect from_port="document" to_op="Tokenize (3)" to_port="document"/>

<connect from_op="Tokenize (3)" from_port="document" to_op="Filter Stopwords (3)" to_port="document"/>

<connect from_op="Filter Stopwords (3)" from_port="document" to_op="Find Hypernyms (3)" to_port="document"/>

<connect from_op="Open WordNet Dictionary (3)" from_port="dictionary" to_op="Find Hypernyms (3)" to_port="dictionary"/>

<connect from_op="Find Hypernyms (3)" from_port="document" to_port="document 1"/>

<portSpacing port="source_document" spacing="0"/>

<portSpacing port="sink_document 1" spacing="0"/>

<portSpacing port="sink_document 2" spacing="0"/>

</process>

</operator>

<operator activated="true" class="multiply" compatibility="5.3.013" expanded="true" height="94" name="Multiply" width="90" x="313" y="120"/>

<operator activated="true" class="store" compatibility="5.3.013" expanded="true" height="60" name="Store (2)" width="90" x="447" y="30">

<parameter key="repository_entry" value="2008_11_WV_Complete_TFIDF"/>

</operator>

<operator activated="true" class="scale_by_weights" compatibility="5.3.013" expanded="true" height="76" name="Scale by Weights" width="90" x="447" y="345"/>

<operator activated="true" class="store" compatibility="5.3.013" expanded="true" height="60" name="Store" width="90" x="514" y="210">

<parameter key="repository_entry" value="2008_11_WV_Complete_SumScore_TFIDF"/>

</operator>

<connect from_op="Read Weights" from_port="output" to_op="Scale by Weights" to_port="weights"/>

<connect from_op="Retrieve EntireNewNoTimeInorder2008_11 (2)" from_port="output" to_op="Process Documents from Data (3)" to_port="example set"/>

<connect from_op="Process Documents from Data (3)" from_port="example set" to_op="Multiply" to_port="input"/>

<connect from_op="Multiply" from_port="output 1" to_op="Store (2)" to_port="input"/>

<connect from_op="Multiply" from_port="output 2" to_op="Scale by Weights" to_port="example set"/>

<connect from_op="Store (2)" from_port="through" to_port="result 1"/>

<connect from_op="Scale by Weights" from_port="example set" to_op="Store" to_port="input"/>

<connect from_op="Store" from_port="through" to_port="result 2"/>

<portSpacing port="source_input 1" spacing="0"/>

<portSpacing port="sink_result 1" spacing="0"/>

<portSpacing port="sink_result 2" spacing="0"/>

<portSpacing port="sink_result 3" spacing="0"/>

</process>

</operator>

</process>

Code for Process Phase:

<?xml version="1.0" encoding="UTF-8" standalone="no"?>

<process version="5.3.013">

<context>

<input/>

<output/>

<macros/>

</context>

<operator activated="true" class="process" compatibility="5.3.013" expanded="true" name="Process">

<process expanded="true">

<operator activated="true" class="retrieve" compatibility="5.3.013" expanded="true" height="60" name="Retrieve 2008_11_WV_Complete_SumScore_TFIDF (2)" width="90" x="45" y="75">

<parameter key="repository_entry" value="2008_11_WV_Complete_SumScore_TFIDF"/>

</operator>

<operator activated="true" class="extract_macro" compatibility="5.3.013" expanded="true" height="60" name="Extract Macro" width="90" x="179" y="75">

<parameter key="macro" value="%{last}"/>

<parameter key="attribute_name" value="A"/>

<list key="additional_macros"/>

</operator>

<operator activated="true" class="retrieve" compatibility="5.3.013" expanded="true" height="60" name="Retrieve EntireNewNoTimeInorder2008_11 (2)" width="90" x="45" y="210">

<parameter key="repository_entry" value="EntireNewNoTimeInorder2008_11"/>

</operator>

<operator activated="true" class="extract_macro" compatibility="5.3.013" expanded="true" height="60" name="Extract Macro (2)" width="90" x="179" y="165">

<parameter key="macro" value="%{last}"/>

<parameter key="attribute_name" value="A"/>

<list key="additional_macros"/>

</operator>

<operator activated="true" class="retrieve" compatibility="5.3.013" expanded="true" height="60" name="Retrieve 2008_11_WV_Complete_TFIDF (2)" width="90" x="45" y="345">

<parameter key="repository_entry" value="2008_11_WV_Complete_TFIDF"/>

</operator>

<operator activated="true" class="extract_macro" compatibility="5.3.013" expanded="true" height="60" name="Extract Macro (3)" width="90" x="179" y="255">

<parameter key="macro" value="%{last}"/>

<parameter key="attribute_name" value="A"/>

<list key="additional_macros"/>

</operator>

<operator activated="true" class="loop" compatibility="5.3.013" expanded="true" height="112" name="Loop" width="90" x="313" y="120">

<parameter key="set_iteration_macro" value="true"/>

<parameter key="macro_name" value="iteration_no"/>

<parameter key="iterations" value="12"/>

<process expanded="true">

<operator activated="true" class="generate_macro" compatibility="5.3.013" expanded="true" height="112" name="Generate Macro" width="90" x="45" y="30">

<list key="function_descriptions">

<parameter key="iteration_last" value="%{last}-%{iteration_no}+1"/>

</list>

</operator>

<operator activated="true" class="generate_macro" compatibility="5.3.013" expanded="true" height="112" name="Generate Macro (2)" width="90" x="180" y="30">

<list key="function_descriptions">

<parameter key="before_iteration_last" value="%{iteration_last}-1"/>

</list>

</operator>

<operator activated="true" class="filter_example_range" compatibility="5.3.013" expanded="true" height="76" name="Filter Example Range (SumScore_TFIDF_WV)" width="90" x="380" y="30">

<parameter key="first_example" value="1"/>

<parameter key="last_example" value="%{iteration_last}"/>

</operator>

<operator activated="true" class="filter_example_range" compatibility="5.3.013" expanded="true" height="76" name="Filter Example Range (EntireHeadlines)" width="90" x="45" y="165">

<parameter key="first_example" value="%{iteration_last}"/>

<parameter key="last_example" value="%{iteration_last}"/>

</operator>

<operator activated="false" class="read_weights" compatibility="5.3.013" expanded="true" height="60" name="Read Weights" width="90" x="112" y="300">

<parameter key="attribute_weights_file" value="D:\Dropbox\Arman\SentiWordNet_3.0.0\home\swn\www\admin\dump\Weight Prep\NewScores\SumScore\SumScoreComplete.xml"/>

</operator>

<operator activated="true" class="extract_log_value" compatibility="5.3.013" expanded="true" height="60" name="Extract Log Value(Headline Text)" width="90" x="45" y="255">

<parameter key="attribute_name" value="A"/>

<parameter key="example_index" value="1"/>

</operator>

<operator activated="true" class="text:process_document_from_data" compatibility="5.3.002" expanded="true" height="76" name="Process Documents from Data (Last Example)" width="90" x="179" y="165">

<parameter key="vector_creation" value="Binary Term Occurrences"/>

<list key="specify_weights"/>

<process expanded="true">

<operator activated="true" class="text:tokenize" compatibility="5.3.002" expanded="true" height="60" name="Tokenize (2)" width="90" x="45" y="30"/>

<operator activated="true" class="text:filter_stopwords_english" compatibility="5.3.002" expanded="true" height="60" name="Filter Stopwords (2)" width="90" x="246" y="30"/>

<operator activated="true" class="wordnet:open_wordnet_dictionary" compatibility="5.2.000" expanded="true" height="60" name="Open WordNet Dictionary (2)" width="90" x="45" y="345">

<parameter key="directory" value="D:\Dropbox\Arman\WNdb-3.0\dict"/>

</operator>

<operator activated="false" class="wordnet:stem_wordnet" compatibility="5.2.000" expanded="true" height="76" name="Stem (WordNet)" width="90" x="112" y="165"/>

<operator activated="true" class="wordnet:find_hypernym_wordnet" compatibility="5.2.000" expanded="true" height="76" name="Find Hypernyms (2)" width="90" x="380" y="165">

<parameter key="use_prefix" value="false"/>

<parameter key="multiple_meanings_per_word_policy" value="Take only first meaning"/>

<parameter key="take_ID_instead_of_words" value="true"/>

</operator>

<connect from_port="document" to_op="Tokenize (2)" to_port="document"/>

<connect from_op="Tokenize (2)" from_port="document" to_op="Filter Stopwords (2)" to_port="document"/>

<connect from_op="Filter Stopwords (2)" from_port="document" to_op="Find Hypernyms (2)" to_port="document"/>

<connect from_op="Open WordNet Dictionary (2)" from_port="dictionary" to_op="Find Hypernyms (2)" to_port="dictionary"/>

<connect from_op="Find Hypernyms (2)" from_port="document" to_port="document 1"/>

<portSpacing port="source_document" spacing="0"/>

<portSpacing port="sink_document 1" spacing="0"/>

<portSpacing port="sink_document 2" spacing="0"/>

</process>

</operator>

<operator activated="false" class="scale_by_weights" compatibility="5.3.013" expanded="true" height="76" name="Scale by Weights" width="90" x="246" y="300"/>

<operator activated="true" class="multiply" compatibility="5.3.013" expanded="true" height="94" name="Multiply (2)" width="90" x="313" y="165"/>

<operator activated="true" class="union" compatibility="5.3.013" expanded="true" height="76" name="Union" width="90" x="447" y="165"/>

<operator activated="true" class="select_attributes" compatibility="5.3.013" expanded="true" height="76" name="Select Attributes (based on last if available)" width="90" x="581" y="210">

<parameter key="attribute_filter_type" value="no_missing_values"/>

</operator>

<operator activated="true" class="branch" compatibility="5.3.013" expanded="true" height="112" name="Branch" width="90" x="447" y="345">

<parameter key="condition_type" value="min_attributes"/>

<parameter key="condition_value" value="1"/>

<process expanded="true">

<operator activated="true" class="filter_example_range" compatibility="5.3.013" expanded="true" height="76" name="Filter Example Range (2)" width="90" x="45" y="120">

<parameter key="first_example" value="1"/>

<parameter key="last_example" value="%{iteration_last}"/>

</operator>

<connect from_port="condition" to_op="Filter Example Range (2)" to_port="example set input"/>

<connect from_op="Filter Example Range (2)" from_port="example set output" to_port="input 1"/>

<portSpacing port="source_condition" spacing="0"/>

<portSpacing port="source_input 1" spacing="0"/>

<portSpacing port="source_input 2" spacing="0"/>

<portSpacing port="source_input 3" spacing="0"/>

<portSpacing port="sink_input 1" spacing="0"/>

<portSpacing port="sink_input 2" spacing="0"/>

</process>

<process expanded="true">

<operator activated="true" class="filter_example_range" compatibility="5.3.013" expanded="true" height="76" name="Filter Example Range (All before last from TF-IDF)" width="90" x="112" y="30">

<parameter key="first_example" value="1"/>

<parameter key="last_example" value="%{iteration_last}"/>

</operator>

<operator activated="true" class="union" compatibility="5.3.013" expanded="true" height="76" name="Union (2)" width="90" x="179" y="120"/>

<operator activated="true" class="select_attributes" compatibility="5.3.013" expanded="true" height="76" name="Select Attributes (2)" width="90" x="313" y="75">

<parameter key="attribute_filter_type" value="no_missing_values"/>

</operator>

<operator activated="true" class="filter_example_range" compatibility="5.3.013" expanded="true" height="76" name="Filter Example Range (3)" width="90" x="313" y="210">

<parameter key="first_example" value="1"/>

<parameter key="last_example" value="%{iteration_last}"/>

</operator>

<operator activated="true" class="log" compatibility="5.3.013" expanded="true" height="76" name="Log (else of branch/tf-idf usage)" width="90" x="380" y="345">

<list key="log">

<parameter key="LogNo" value="operator.Filter Example Range (All before last from TF-IDF).value.applycount"/>

<parameter key="Headline Row No" value="operator.Filter Example Range (EntireHeadlines).parameter.last_example"/>

<parameter key="Headline Text" value="operator.Extract Log Value(Headline Text).value.data_value"/>

</list>

</operator>

<connect from_port="input 1" to_op="Filter Example Range (All before last from TF-IDF)" to_port="example set input"/>

<connect from_port="input 2" to_op="Union (2)" to_port="example set 2"/>

<connect from_op="Filter Example Range (All before last from TF-IDF)" from_port="example set output" to_op="Union (2)" to_port="example set 1"/>

<connect from_op="Union (2)" from_port="union" to_op="Select Attributes (2)" to_port="example set input"/>

<connect from_op="Select Attributes (2)" from_port="example set output" to_op="Filter Example Range (3)" to_port="example set input"/>

<connect from_op="Filter Example Range (3)" from_port="example set output" to_op="Log (else of branch/tf-idf usage)" to_port="through 1"/>

<connect from_op="Log (else of branch/tf-idf usage)" from_port="through 1" to_port="input 1"/>

<portSpacing port="source_condition" spacing="0"/>

<portSpacing port="source_input 1" spacing="0"/>

<portSpacing port="source_input 2" spacing="0"/>

<portSpacing port="source_input 3" spacing="0"/>

<portSpacing port="sink_input 1" spacing="0"/>

<portSpacing port="sink_input 2" spacing="0"/>

</process>

</operator>

<operator activated="true" class="materialize_data" compatibility="5.3.013" expanded="true" height="76" name="Materialize Data" width="90" x="45" y="390"/>

<operator activated="true" class="multiply" compatibility="5.3.013" expanded="true" height="94" name="Multiply" width="90" x="45" y="525"/>

<operator activated="true" class="filter_example_range" compatibility="5.3.013" expanded="true" height="76" name="Filter Example Range (Last Example)" width="90" x="179" y="570">

<parameter key="first_example" value="%{iteration_last}"/>

<parameter key="last_example" value="%{iteration_last}"/>

</operator>

<operator activated="true" class="filter_example_range" compatibility="5.3.013" expanded="true" height="76" name="Filter Example Range (All before last)" width="90" x="179" y="435">

<parameter key="first_example" value="1"/>

<parameter key="last_example" value="%{before_iteration_last}"/>

</operator>

<operator activated="true" class="support_vector_machine" compatibility="5.3.013" expanded="true" height="112" name="SVM" width="90" x="313" y="435"/>

<operator activated="true" class="apply_model" compatibility="5.3.013" expanded="true" height="76" name="Apply Model" width="90" x="447" y="570">

<list key="application_parameters"/>

</operator>

<operator activated="true" class="free_memory" compatibility="5.3.013" expanded="true" height="76" name="Free Memory" width="90" x="581" y="525"/>

<connect from_port="input 1" to_op="Generate Macro" to_port="through 1"/>

<connect from_port="input 2" to_op="Generate Macro" to_port="through 2"/>

<connect from_port="input 3" to_op="Generate Macro" to_port="through 3"/>

<connect from_op="Generate Macro" from_port="through 1" to_op="Generate Macro (2)" to_port="through 1"/>

<connect from_op="Generate Macro" from_port="through 2" to_op="Generate Macro (2)" to_port="through 2"/>

<connect from_op="Generate Macro" from_port="through 3" to_op="Generate Macro (2)" to_port="through 3"/>

<connect from_op="Generate Macro (2)" from_port="through 1" to_op="Filter Example Range (SumScore_TFIDF_WV)" to_port="example set input"/>

<connect from_op="Generate Macro (2)" from_port="through 2" to_op="Filter Example Range (EntireHeadlines)" to_port="example set input"/>

<connect from_op="Generate Macro (2)" from_port="through 3" to_op="Branch" to_port="input 1"/>

<connect from_op="Filter Example Range (SumScore_TFIDF_WV)" from_port="example set output" to_op="Union" to_port="example set 1"/>

<connect from_op="Filter Example Range (EntireHeadlines)" from_port="example set output" to_op="Extract Log Value(Headline Text)" to_port="example set"/>

<connect from_op="Read Weights" from_port="output" to_op="Scale by Weights" to_port="weights"/>

<connect from_op="Extract Log Value(Headline Text)" from_port="example set" to_op="Process Documents from Data (Last Example)" to_port="example set"/>

<connect from_op="Process Documents from Data (Last Example)" from_port="example set" to_op="Multiply (2)" to_port="input"/>

<connect from_op="Multiply (2)" from_port="output 1" to_op="Union" to_port="example set 2"/>

<connect from_op="Multiply (2)" from_port="output 2" to_op="Branch" to_port="input 2"/>

<connect from_op="Union" from_port="union" to_op="Select Attributes (based on last if available)" to_port="example set input"/>

<connect from_op="Select Attributes (based on last if available)" from_port="example set output" to_op="Branch" to_port="condition"/>

<connect from_op="Branch" from_port="input 1" to_op="Materialize Data" to_port="example set input"/>

<connect from_op="Materialize Data" from_port="example set output" to_op="Multiply" to_port="input"/>

<connect from_op="Multiply" from_port="output 1" to_op="Filter Example Range (All before last)" to_port="example set input"/>

<connect from_op="Multiply" from_port="output 2" to_op="Filter Example Range (Last Example)" to_port="example set input"/>

<connect from_op="Filter Example Range (Last Example)" from_port="example set output" to_op="Apply Model" to_port="unlabelled data"/>

<connect from_op="Filter Example Range (All before last)" from_port="example set output" to_op="SVM" to_port="training set"/>

<connect from_op="SVM" from_port="model" to_op="Apply Model" to_port="model"/>

<connect from_op="Apply Model" from_port="labelled data" to_op="Free Memory" to_port="through 1"/>

<connect from_op="Free Memory" from_port="through 1" to_port="output 1"/>

<portSpacing port="source_input 1" spacing="0"/>

<portSpacing port="source_input 2" spacing="0"/>

<portSpacing port="source_input 3" spacing="0"/>

<portSpacing port="source_input 4" spacing="0"/>

<portSpacing port="sink_output 1" spacing="0"/>

<portSpacing port="sink_output 2" spacing="0"/>

</process>

</operator>

<operator activated="true" class="loop_collection" compatibility="5.3.013" expanded="true" height="76" name="Loop Collection" width="90" x="246" y="435">

<process expanded="true">

<operator activated="true" class="select_attributes" compatibility="5.3.013" expanded="true" height="76" name="Select Attributes (3)" width="90" x="232" y="30">

<parameter key="attribute_filter_type" value="subset"/>

<parameter key="attributes" value="|B|confidence(N)|confidence(P)|prediction(B)"/>

</operator>

<connect from_port="single" to_op="Select Attributes (3)" to_port="example set input"/>

<connect from_op="Select Attributes (3)" from_port="example set output" to_port="output 1"/>

<portSpacing port="source_single" spacing="0"/>

<portSpacing port="sink_output 1" spacing="0"/>

<portSpacing port="sink_output 2" spacing="0"/>

</process>

</operator>

<operator activated="true" class="append" compatibility="5.3.013" expanded="true" height="76" name="Append" width="90" x="380" y="435"/>

<operator activated="true" class="performance_classification" compatibility="5.3.013" expanded="true" height="76" name="Performance" width="90" x="514" y="435">

<list key="class_weights"/>

```
    </operator>

    <connect from_op="Retrieve 2008_11_WV_Complete_SumScore_TFIDF (2)" from_port="output" to_op="Extract Macro" to_port="example set"/>

    <connect from_op="Extract Macro" from_port="example set" to_op="Loop" to_port="input 1"/>

    <connect from_op="Retrieve EntireNewNoTimeInorder2008_11 (2)" from_port="output" to_op="Extract Macro (2)" to_port="example set"/>

    <connect from_op="Extract Macro (2)" from_port="example set" to_op="Loop" to_port="input 2"/>

    <connect from_op="Retrieve 2008_11_WV_Complete_TFIDF (2)" from_port="output" to_op="Extract Macro (3)" to_port="example set"/>

    <connect from_op="Extract Macro (3)" from_port="example set" to_op="Loop" to_port="input 3"/>

    <connect from_op="Loop" from_port="output 1" to_op="Loop Collection" to_port="collection"/>

    <connect from_op="Loop Collection" from_port="output 1" to_op="Append" to_port="example set 1"/>

    <connect from_op="Append" from_port="merged set" to_op="Performance" to_port="labelled data"/>

    <connect from_op="Performance" from_port="performance" to_port="result 1"/>

    <connect from_op="Performance" from_port="example set" to_port="result 2"/>

    <portSpacing port="source_input 1" spacing="0"/>

    <portSpacing port="sink_result 1" spacing="0"/>

    <portSpacing port="sink_result 2" spacing="0"/>

    <portSpacing port="sink_result 3" spacing="0"/>

  </process>

 </operator>

</process>
```