Charles University

Faculty of Social Sciences
Institute of Economic Studies

MASTER'S THESIS

News Feed Classifications to Improve Volatility Predictions

Author: **Bc. Ksenia Pogodina**

Supervisor: **PhDr. Boril Šopov, MSc., LL.M.**

Academic Year: **2017/2018**

Declaration of Authorship

The author hereby declares that he compiled this thesis independently, using only the listed resources and literature, and the thesis has not been used to obtain a different or the same degree.

The author grants to Charles University permission to reproduce and to distribute copies of this thesis document in whole or in part.

Prague, January 2, 2018

Signature

Acknowledgments

I would like to express my gratitude to my supervisor PhDr. Boril Šopov, MSc., LL.M. for his guidance, valuable advices and suggestions. I am also grateful to my family and friends for their continuous support and encouragement.

Bibliography Reference

Pogodina, Ksenia. *News Feed Classifications to Improve Volatility Predictions* Prague, 2018. 64 p. Master Thesis (Mgr.) Charles University, Faculty of Social Sciences, Institute of Economic Studies. Supervisor: PhDr. Boril Šopov, MSc., LL.M.

Extent of the Thesis

107,029 characters (with spaces)

Abstract

This thesis analyzes various text classification techniques in order to assess whether the knowledge of published news articles about selected companies can improve its' stock return volatility modelling and forecasting. We examine the content of the textual news releases and derive the news sentiment (polarity and strength) employing three different approaches: supervised machine learning Naive Bayes algorithm, lexicon-based as a representative of linguistic approach and hybrid Naive Bayes. In hybrid Naive Bayes we consider only the words contained in the specific lexicon rather than whole set of words from the article. For the lexicon-based approach we used independently two lexicons one with binary another with multiclass labels. The training set for the Naive Bayes was labeled by the author. When comparing the classifiers from the machine learning approach we can conclude that all of them performed similarly with a slight advantage of the hybrid Naive Bayes combined with multiclass lexicon. The resulting quantitative data in form of sentiment scores will be then incorporated into GARCH volatility modelling. The findings suggest that information contained in news feeds does bring an additional explanatory power to traditional GARCH model and is able to improve it's forecast. On the contrary, we could not provide enough evidence for favouring specific sentiment-derivation method. While the model employing hybrid Naive Bayes approach provided a bitter in-sample fit, the preferred model in the out-of-sample evaluation was the one employing multiclass lexicon. We also showed an asymmetric news effect, where both positive and negative news increase volatility with a latter having a more pronounced effect.

JEL Classification	C22, C52, C58, G14, G17, G41
Keywords	volatility, text, classifier, lexicon, sentiment, news

Author's e-mail	`ksenia1105@gmail.com`
Supervisor's e-mail	`boril.sopov@gmail.com`

Abstrakt

Tato práce analyzuje různé metody klasifikace textu za účelem zjištění, zda-li publikované novinové články o konkrétních společnostech umožňují lepší simulaci a predikci volatility akcií dané společnosti. V práci zkoumáme obsah textu publikovaných novinových článků a z toho vycházející sentiment (směr a síla) za použití tří různých přístupů: supervised machine learning Naive Bayes algoritmus, lexicon-based jako zástupce lingvistického přístupu a hybridní Naive Bayes. V rámci hybridního Naive Bayes jsou uvažována pouze slova obsažená v daném lexikonu a nikoliv celý obsah článku. Pro lexicon-based přístup používáme nezávisle dva lexikony, jeden s binárním a jeden vícetřídním hodnocením sentimentu. Sentiment v trénovacím setu pro Naive Bayes byl přiřazen autorem. Z porovnání klasifikační metod založených na machine learning dojdeme k závěru, že všechny metody dosahují podobných výsledků z nichž nejlépe vychází hybridní Naive Bayes používající vícetřídní lexikon. Výstupní kvantitativní data ve formě hodnot sentimentu jsou pak dále zahrnuta do modelování volatility pomocí GARCH. Výsledky ukazují, že informace obsažené v novinových článcích přinášejí další vysvětlující prvek do tradičního GARCH modelu a jsou schopné zlepšit odhad. Nicméně, nejsme schopni získat dost podkladů pro určení nejlepší metody kvantifikace sentimentu. Model používající hybridní Naive Bayes přístup přinesl lepší in-sample výsledky, pro out-of-sample bylo však lepší užít vícetřídní lexikon. Také se nám podařilo ukázat asymetrický efekt, kdy pozitivní i negativní zprávy zvyšují volatilitu, nicméně u zpráv negativních je tento efekt silnější.

Klasifikace JEL	C22, C52, C58, G14, G17, G41
Klíčová slova	volatilita, text, klasifikátor, lexikon, sentiment, novinové články

E-mail autora	ksenia1105@gmail.com
E-mail vedoucího práce	boril.sopov@gmail.com

Contents

List of Tables ix

List of Figures x

Acronyms xi

Thesis Proposal xiii

1 Introduction 1

2 Literature Review and Theoretical Framework 4

2.1 Efficient Market Hypothesis vs Behavioral Finance 4

2.2 News Sentiment and Stock Market Volatility 5

2.3 Data Analytics: Text Processing and Classification 8

3 Text-based Sentiment Derivation Techniques 12

3.1 Lexicon-based Approach 13

3.2 Machine Learning Approach: Naive Bayes Algorithm 14

3.2.1 Training the Naive Bayes Classifier 17

3.2.2 Test sample and Cross-validation 18

3.2.3 Classifiers' performance evaluation 20

4 Conditional Heteroscedastic Models for Volatility 25

4.1 ARCH 26

4.2 GARCH 28

4.3 Model Specification with News Sentiment 28

4.4 Out-of-sample model performance 30

5 Data 32

5.1 Financial Time-series Data 32

5.2 News Sentiment Data 37

6 Model Estimation and Results **41**

6.1 Apple Inc. 41

6.1.1 In-sample Model fitting without News Sentiment 41

6.1.2 In-sample Model fitting with News Sentiment 45

6.1.3 Out-of-sample evaluation 47

6.2 Microsoft Corporation 49

6.2.1 In-sample Model fitting without News Sentiment 49

6.2.2 In-sample Model fitting with News Sentiment 50

6.2.3 Out-of-sample evaluation 52

6.3 Amazon 53

6.3.1 In-sample Model fitting without News Sentiment 53

6.3.2 In-sample Model fitting with News Sentiment 54

6.3.3 Out-of-sample evaluation 56

7 Conclusion **58**

Bibliography **65**

A Appendix **I**

List of Tables

3.1 Accuracy and F-scores for all stocks 23
3.2 Accuracy and F-scores with adjusted threshold 23

5.1 Information about selected companies 32
5.2 Summary statistics of stocks log returns 37
5.3 Summary statistics of sentiments for all stocks 40

6.1 Plain GARCH estimation of Apple 43
6.2 Apple: Augmented GARCH with News Sentiments 46
6.3 Apple Inc: RMSE and MAE for out-of-sample 48
6.4 Plain GARCH estimation of Microsoft 50
6.5 Microsoft: Augmented GARCH with News Sentiments 51
6.6 Microsoft: RMSE and MAE for out-of-sample 53
6.7 Plain GARCH estimation of Amazon 54
6.8 Amazon: Augmented GARCH with News Sentiments 55
6.9 Amazon: RMSE and MAE for out-of-sample 56

List of Figures

3.1 Source code: Text preprocessing 13
3.2 Source code: Likelihood Estimation for the train sample 18
3.3 10-fold cross-validation scheme 19
3.4 Confusion matrix 22

5.1 Time series plot of Apple 2014-2017 33
5.2 Log returns of Apple stock 2014-2017 34
5.3 Time series plot of Microsoft 2014-2017 34
5.4 Log returns of Microsoft stock 2014-2017 35
5.5 Time series plot of Amazon.com 2014-2017 36
5.6 Log returns of Amazon stock 2013-2017 36
5.7 Sentiment time-series plots 39

6.1 ACF and PACF of squared residuals for Apple 42
6.2 ACF and PACF of squared residuals after GARCH(1,1) 44

A.1 Source code: Likelihood Estimation for the test sample I
A.2 ACF and PACF of Apple time series 2014-2017 II
A.3 ADF test for Apple time series 2014-2017 II
A.4 ADF test for ldApple time series 2014-2017 III
A.5 KPSS test for ldApple time series 2014-2017 III
A.6 ARCH-LM test of squared residuals of Apple stock III

Acronyms

EMH	Efficient Market Hypothesis
NB	Naive Bayes
AIC	Akaike information criterion
BIC	Bayesian information criterion
LL	Log-likelihood
SP	Positive sentiment
SN	Negative sentiment
MLE	Maximum likelihood estimation
AR	Autoregressive
MA	Moving Average
ARMA	Autoregressive Moving Average
ARIMA	Autoregressive Integrated Moving Average
ARCH	Autoregressive conditional heteroskedasticity
GARCH	Generalized autoregressive conditional heteroskedasticity
ACF	Autocorrelation function
PACF	Partial autocorrelation function
ADF	Augmented Dickey-Fuller test
KPSS	Kwiatkowski-Phillips-Schmidt-Shin test
NLP	Natural language processing
SVM	Support vector machines
KNN	K-nearest neighbors algorithm
ANN	Artificial neural network
NYSE	New York Stock Exchange
NASDAQ	National Association of Securities Dealers Automated Quotations

DJIA	Dow Jones Industrial Average
IT	Information technology
NLTK	Natural Language Toolkit
RMSE	Root-mean-square deviation
MAE	Mean absolute error

Master’s Thesis Proposal

Author	Bc. Ksenia Pogodina
Supervisor	PhDr. Boril Šopov, MSc., LL.M.
Proposed topic	News Feed Classifications to Improve Volatility Predictions

Motivation In the world of finance, stock markets and their tendencies are highly volatile by nature. This fact attracts market analysts and investors in order to capture the volatility and to be able to forecast its further movements. Consequently, they build their buy or sell strategies based on examined market behaviour, where the outcome and profitability are dependent on the accuracy of the forecasts. Volatility analysis plays an important role in portfolio creation, security valuations, risk management and monetary policy making, being an indicator of risk or uncertainty.

In line with the criticism of EMH, Stiglitz & Grossman (1980) argued that the informed market agents are capable to take positions which are ”better” than those of uninformed market agents. Additionally, the market does not reflect at any time all available information. That is, when a new piece of information enters the market, the market state changes. This new information influences the investors decisions and is considered to be an important data source while building a financial forecast. For that reason, the role of various types of news (general, macroeconomic, political, etc.) and its sentiments have been thoroughly studied in order to better understand the securities price formation and stock market returns.

The very crucial part of analysis on how the news sentiment about the company is influencing the volatility of that company’s stock is to derive the sentiment itself. This is very sophisticated and important task, since the accuracy of the sentiment can determine the predictive power of the model and eventually its profitability. The news sentiment derivation is built on text processing and relevant information extraction, turning the qualitative data into quantitative, that is, assigning the mood to the articles represented in numbers (scores). For this purpose there are exist several approaches: machine learning ones (Support Vector Machine, Naive Bayes and so on) and lexicon or pre-defined dictionary based. However, there is no conclusive results on such method selection and opinions of the researchers differ.

Hypotheses

Hypothesis #1: Incorporating news sentiment information into volatility modelling can improve volatility forecasts

Hypothesis #2: The Naive Bayes algorithm for news sentiment derivation can outperform the lexicon-based approaches in terms of accuracy of the volatility forecasts

Hypothesis #3: The positive and negative news have asymmetric impact on the stock volatility, with latter having a more pronounced effect

Methodology The first step of Sentiment&Volatility analysis is gathering time series data of selected stocks (Apple Inc, Microsoft Corporation, Amazon.com) from Yahoo Finance database. Then the returns of selected stocks will be calculated and analysed. Then the daily news articles of selected companies will be collected from Factiva database (under subscription of Charles University). The next important part is news articles classification to negative, neutral and positive. Moreover, we will also account for the strength or magnitude of negativness and positivness. For these purposes the Naive Bayes algorithm and two pre-defined lexicons (binary and multiclass) will be used. Naive Bayes is a probabilistic classifier based on applying Bayes Theorem. Python programming language will be implemented for building classifications. The final step will be to model stock volatility with GARCH family of models and with a help of obtained sentiments, compare the results.

Expected Contribution First of all, the news processing algorithm involves supervised learning technique, where the train dataset should be labeled manually. This procedure will be performed by author and is thus unique. Compared to other studies we have also used various thresholds when assessing the accuracy of the classifiers. Also the combination of the datasets, approaches applied for classifications as well as volatility models differs from the previous studies.

Outline

1. Intoduction
2. Literature Review and Theoretical Framework
3. Text-based Sentiment Derivation Techniques

4. Conditional Heteroscedastic Models for Volatility
5. Data
6. Model Estimation and Results
7. Conclusion

Core bibliography

Stiglitz, J., Grossman, S. On the Impossibility of Informationally Efficient Markets, 1980. The American Economic Review. 70 (3), 393-408.

Laakkonen, H., Lanne, M., 2009. Asymmetric News Effects on Exchange Rate Volatility: Good vs. Bad News in Good vs. Bad Times. Studies in Nonlinear Dynamics. 14 (1), 1-38.

Poon, S., Granger, W., 2003. Forecasting Volatility in Financial Markets: A Review. Journal of Economic Literature 41 (2), 478-539.

Schumaker, R., Chen, H., 2009. Textual analysis of stock market predictions using breaking financial news: the azfin text system. ACM Transactions on Information Systems. 27 (2), 12.

Tetlock, P.C, 2007. Giving Content to Investor Sentiment: The Role of Media in the Stock Market. The Journal of American Finance Association. 62 (3), 1139-1168.

Author

Supervisor

Chapter 1

Introduction

In 1960s the predominant view on financial market functioning was based on the idea of non-predictability of the asset prices and of investors holding a rational expectations. This view is built on the traditional Efficient Market Hypothesis (EMH) developed by Eugene Fama (1970). However, during the following years EMH has been challenged due to its simplified assumptions and the fact that acquiring and analyzing information is costly. Stiglitz & Grossman (1980) argued that the informed market agents are capable to take positions which are "better" than those of uninformed market agents. Prechter & Parker (2007) claimed that stock prices can be, at least to some extent, predictable. The alternative views relax the assumption of rational expectations and at this point a behavioral finance comes into play. Behavioral finance is relatively new field, attempting to fix the flaws of the conventional models, suggesting that emotions and other phychological aspects are able to influence investors decisions.

Therefore, when a new piece of information enters the market, the market state changes. This new information shapes the investors decisions and is considered to be an important data source while building a financial forecast. Hence, it makes sense to analyze how this new information moves the market and to develop techniques allowing to analyze this information at the lowest cost. For that reason, the role of various types of news has been thoroughly studied in order to better understand the securities price formation, stock market returns and their volatility. An example can be stock volatility of suppling firm in case of Challanger spaceshuttle crash. Due to live media coverage this information was immidiately reflected in the supplier's stock and then changed it's long-term behaviour. Volatility analysis is important for portfolio creation, security valuations, risk management and monetary policy making, being an indicator of risk or uncertainty. Based on examined market behaviour the investors will further build their trading strategies.

The digitalized format of the formally printed media brings the possibility of more efficient and less costly analysis and processing of the enormous information flow. It is very important task to correctly convert the qualitative data into quantitative in order to capture all its relevant features and to further apply it as an input variable in financial models. The demand for automated solutions of information processing, specifically textual information, speeds up the development of machine learning and linguistic techniques for text analysis. Nevertheless, there is still no conclusive answer among researchers on the preferred approach for extracting the necessary information, namely, news sentiments.

The first objective of this thesis is to check whether the news, in particular their sentiment, bring an additional explanatory power to the traditional volatility models. Specifically, whether the content knowledge of the news released about particular company will help to model and predict the company's stock volatility. In addition, we will test the hypothesis of asymmetric news effect on volatility. That is, to examine whether the behavioral finance framework does matter for financial market modelling. The second objective is to study a various sentiment derivation and text classification techniques to be able to achieve a more accurate articles evaluation and hence the stock volatility modelling and then predictions.

For deriving the news sentiments from textual news articles we will employ a Naive Bayes classifier as a representative of machine learning approach and lexicon-based sentiment derivation technique as a representative of linguistic approach. We will also combine both approaches in order to compare which one gives a better performance. By combining we mean considering only the words contained in the specific lexicon rather than whole set of words from the article. By doing this we will obtain a hybrid Naive Bayes classifier, who's performance will indicate the adequacy of chosen text preprocessing steps. Naive Bayes is a probabilistic classifier based on Bayes Theorem. The lexicon-based approach is usually based on word frequency in the content of each article that matches the categories in a predefined dictionary. Each category contains a quantitative expression of sentiment of the selected words. Moreover, we will utilize two lexicons: binary and multiclass one, therefore we will not only assess the direction of the news sentiment, but also its strength. The Naive Bayes algorithm is also based on the multiclass labelling. As a classical volatility models the GARCH family of models are applied, also being a benchmark to augmented GARCH models with the exogenous news information.

The thesis is structured as follows: Chapter 2 provides an insight into literature about the role and behaviour of news in finance modelling, specifically stock volatility modelling. Additionally, it provides an overview of text analytic techniques for news articles classification as well as sentiment derivation. Chapter 3 describes the text classification methodology of the applied sentiment derivation techniques: Naive Bayes, lexicon and hybrid approaches. Chapter 4 explains the econometrical volatility models (GARCH) and also provides a functional form of the augmented model employed in the thesis. Chapter 5 shed a light on the features and summary statistics of the data used. Chapter 6 compares and interprets an estimation results of the classical volatility models as well as models augmented with news sentiment. Chapter 7 provides a concluding remarks and suggests possibility for further research and improvements.

Chapter 2

Literature Review and Theoretical Framework

2.1 Efficient Market Hypothesis vs Behavioral Finance

One of the first famous investment theories attempting to explain the market behavior is an Efficient Market Hypothesis (EMH), developed by Eugene Fama (1970). The theory states that the asset prices fully reflect all available information and it is impossible to "beat the market" consistently. There are three forms of the hypothesis: weak, semi-strong and strong. The weak form suggests that the prices of a given asset already reflect all past publically avaliable information. The semi-strong form says that prices of the given asset reflect all publically available information and that prices constantly change to reflect new public information. Additionally, the stong form claims that prices instantly incorporate even private information, that is, information which is not publically known. The Efficient Market Hypothesis is based on the rational expectations to the asset pricing. These models utilize the conventional data like stock market data.

The EMH has been challenged by many researchers for its simplified assumptions (Prechter & Parker 2007). Behavioral finance attempted to resolve the shortcomings of the standard finance models, relaxing the assumption of fully rational behaviour and market efficiency, implying that asset prices can be, at least to some extent, predicted. That is the decision-making process can be influenced by market signals, including emotions, mood and human errors of market participants and can be both conscious and unconscious (Bechara & Damasio 2005). A lot of studies have empirically proved that market sentiment in fact possesses the predictive imformation on the stock market return (Schumaker & Chen 2009). Our work is built on the theory of behavioral finance and market inefficiency, where we aim to extend the existing literature, incorporating the news sentiment into standard finance models.

2.2 News Sentiment and Stock Market Volatility

In the world of finance, stock markets and their tendencies are highly volatile by nature. This fact attracts market analysts and investors in order to capture the volatility and to be able to forecast its further movements. Consequently, they build their buy or sell strategies based on examined market behaviour, where the outcome and profitability are dependent on the accuracy of the forecasts. Volatility analysis plays an important role in portfolio creation, security valuations, risk management and monetary policy making, being an indicator of risk or uncertainty (Poon & Granger 2003).

Therefore, during the recent years, a substantial efforts have been put into developing models, which are able to anticipate the future direction of particular stocks or overall market. However, even after a years of research on volatility predictions, there is still no consensus among analysts on how to actually model the volatility predictions.

In line with the criticism of EMH discussed in the previous section, Stiglitz & Grossman (1980) argued that the informed market agents are capable to take positions which are "better" than those of uninformed market agents. Additionally, the market does not reflect at any time all available information. That is, when a new piece of information enters the market, the market state changes. This new information influences the investors decisions and is considered to be an important data source while building a financial forecast. For that reason, the role of various types of news (general, macroeconomic, political, etc.) has been thoroughly studied in order to better understand the securities price formation and stock market returns. Thus, the information derived from the news articles has become an important part of a modern financial forecasting models.

There is a solid strand of literature, that uses a textual data to find the relation between public information arrival in form of news feeds and stock market volatility. Clark (1973), Andersen (1996), Tauchen & Pitts (1983) claimed, that the variance of stock returns at a given time interval is proportional to the rate of information arrival in the market. Some studies showed, that there is a strong relationship between news releases about a company and its stock prices fluctuations (Ng & Fu 2003). Employing a sentiment scores generated by Raven Pack News Analytics, Ho *et al.* (2013) found that firm-specific news sentiment has a significant influence on the intraday volatility persistence. Additionally, he concluded that the firm-specific news sentiment accounts for a bigger proportion of volatility persistence compared to macroeconomic news sentiment. Furthermore, negative news had a bigger influence on the volatility than positive news. Similarly, Borovkova & Mahakena (2015) attempted to study the effect of news sentiment on returns, price jumps and volatility of natural gas futures. Again, their findings support the hypothesis of significant relationships between news sentiment and the futures prices. They also documented an asymmetric effect of positive and negative news on volatility.

A significant part of literature is focused on studying the asymmetric effect of different types of news on stock return volatility. Veronesi (1999) proposed a theory, where the news sentiment plays a crucial role in explaining the volatility of stock returns and its effect depends on the level of volatility. He suggests, that investors tend to overreact to bad news in good times and underreact to good news in bad times due to asymmetric information about the state of economy prevailing on the market. In other work, Laakkonen & Lanne (2009) came to the conclusion, that bad news increase stock market volatility to a greater extent in good times, than in bad times, while good news influence is insignificant in both cases. Chen & Ghysels (2011) found, that moderately good news decrease the stock market volatility, while extremely good and bad news increase volatility, where the latter has a more substantial impact. Additionally, they stated, that the asymmetric effect is ceased to exist over a longer periods of time.

The works mentioned above, however, have a major drawback. Chen & Ghysels (2011) used a five-minute return as a proxy for news variable. That is, the news are perceived to be good, when the return is positive and perceived to be bad, when the return is negative. Laakkonen & Lanne (2009) employed macroeconomic news announcements to proxy the news arrival: the news are said to be good, if the five-minute return following the macroeconomic announcement is positive, and the news are said to be bad if the five-minute return following the macroeconomic announcement is negative. Here, we get to the situation, when the news sentiment is derived directly from stock market volatility and, hence, those models suffer from endogeneity problem. In this thesis, however, we will derive the news sentiment purely based on the textual content of the articles, which is not dependent on the stock market data.

Ding *et al.* (2014) have also conducted the research, investigating the relationship between the sentiment of the financial news articles and stock price movements. Their sentiment formation algorithm, however, did not bring very accurate results.

Number of works are implementing different approaches to modelling the stock market volatility with introduction of news variable. Lamoureux & Lastrapes (1990) are utilizing the trading volumes in GARCH family of models as a proportional proxy for news feeds flow. They managed to eliminate the persistence of GARCH effects, when including the trading volume as exogenous variable in the conditional variance equation. Similarly, Karpoff (1987) found a link between price movements of Treasury bills, futures and currencies and its trading volumes on the stock markets. He claims, that trading volume is one of the commonly used proxies for the news flow, since lager amount of information about specific stock corresponds to more diverse perceptions of investors about the future price chnages. Kalev *et al.* (2004), Cousin & Launois (2006) also suggested, that daily number of press announcements about a particular stock (news intensity) is the best proxy for news variable, that fits the GARCH modelling.

On the contrary, while analysing the NYSE dairly returns over the four years horizon, Sharma *et al.* (1996) came to the opposite conclusion: the simple GARCH model without trading volume as an exogenous variable outperformed the one with trading volume. Also, the GARCH effect did not diminish with the presence of this exogenous variable in the conditional variance equation. The same result was achieved by Sharma *et al.* (2005), who was studying the same issue based on eight countries: the GARCH effect did not reduce with introduction of the trading volume for all of the selected countries.

Besides the proxies for the information arrival mentioned above, other studies utilized different measures to evaluate the impact of news on volatility, for example: the number of daily newspaper headlines and earnings announcements (Berry & Howe 1994). In the current thesis, we will examine the influence of news articles on the volatility at a company level, not on the market level. The rationale behide it is to capture the full firm-specific effect present in the article, which is neglected on the aggregate level. Additionaly, the news sentiment scores, obtained from text processing, will be applied to improve the volatility forecast of the selected stocks. The procedure of acquiring the news sentiment scores from the articles will be detailed described in the following sections.

It is worth to mention, that different nature of the news is proved to have a different effect on the stock volatility (Andersen 1996). By nature, the news are commonly divided into scheduled (governmental announcements on inflation or unemployment statistics, interest rate decisions, etc.) and unscheduled (e.g general news from the newspapers). This more general news do not necessarily need to be connected with economic activity. There are a fairly large number of works showing, that scheduled news announcements play a secondary role in explaining volatility, especially on the daily level. For example, Andersen & Bollerslev (1998) was analyzing the impact of scheduled and unscheduled news releases on Deutsche Mark exchange rate using the GARCH modelling and provided an evidence of unscheduled news to be of a greater importance in modelling volatility. The research of Rangel (2011) also supports this idea. He concluded, that market reacts differently to different kinds of announcements and "the surprise effect" of the news article improves the predictions. His results are based on the daily S&P data.

2.3 Data Analytics: Text Processing and Classification

The very crucial part of analysis on how the news sentiment about the company is influencing the volatility of that company's stock is to derive the sentiment itself. This is very sophisticated and important task, since the accuracy of the sentiment can determine the predictive power of the model and eventually its profitability. This section will shed a light on widely used text analytic techniques and the stages of text processing and categorization.

The growing number of news articles in electronical form and the desire of researches to extract more information out of news releases contributed to the development of automated solutions for text processing. The interest in this topic of many U.S and european organizations is related to opportunity to better forecast the price movements, volatility and trading volumes on the stock market (Tetlock 2007).

The news analytics is closely related to theory of behavioral finance, opposed to EMH. Nowadays, the publishing volumes of various news agencies are so high, that the market participant himself is unable to properly handle all that volume of new information arrival. For that reason, some pieces of important information, that can influence the investor's decisions, can be missed in a large flow of news. That is, the probability that all traders are equally informed about a particular matter at one point of time is very low. Such situation provides a potential to outperform other market participants via using the automated news analytical solution.

In this context, there rises a question how to handle the unexpected news arrival, which do not have a common structure, numerical values, but contain the relevant information affecting the price of particular stock. The dominant approach to this issue relies on the use of artificial intelligence or machine learning techniques. These techniques are built on the Natural Language Processing (NLP), which became very popular during the last decade.

There exists several machine learning techniques, that are widely used for text processing and extracting the necessary information for modelling the financial predictions:

- Support Vector Machine (SVM)
- Naive Bayes
- Decision Tree
- K- Nearest Neighbors (KNN)
- Artificial neural network (ANN)

Some studies like Finnie *et al.* (2010), Toumazou *et al.* (2013) are exploiting the combination of various machine learning techniques trying to achieve more accurate results.

Among the reviewed papers, the Naive Bayes approach is strongly favoured by analysts due to its performance results (Wuthrich *et al.* 1998). Even if it is difficult to compare the machine learning algorithms (Salzberg 1997) other algorithms, except Support Vector Machine, are significantly behind the Naive Bayes in the context of text classifications and sentiment analysis. Naive Bayes algorithm is one of the oldest existing algorithms, which is built on the "naive" assumption of total independence between the text items. It is distinct from other techniques due to the fact, that it is based on probabilities rather than dimensional (spacial) perception.

There are quite a few empirical examples utilizing the properties of Naive Bayes. Yu *et al.* (2013c) was using Naive Bayes to perform a sentiment analysis to assess the correlations between different sources of social media. Li (2010) employed the Naive Bayes in order to analyze the financial statements of given company. Another important study of Tushar & Saket (2012), that chose a Naive Bayes procedure to derive the negative and positive tweets. They gathered four million tweets over the one year horizon to determine the link between the Twitter sentiment and stock prices volatility of NASDAQ-100, DJIA and stocks of selected IT companies. The outcome of this work shows a strong correlation between the Twitter sentiments and financial market data.

Nevertheless, the reviewed literature also makes use of other methods to obtain the sentiment scores. Christiani & Shave-Taylor (2002) performed the analysis of content of the news feeds to find the relationship between the news and the stock price movements through Suport Vector Machine algorithm. Yang & Pedersen (1997) used the same technique in their research and found it to be superior to other included methods.

Another approach for text classification generation is linguistic approach, which uses the set of predefined dictionaries. The algorithm is usually based on word frequency in the content of each article that matches the categories in a predefined dictionary. Each category contains a quantitative measure of sentiment of the selected words. These dictionaries are developed by market experts and can be either general (WordNet thesaurus) or customized to a specific field of study, for example, psychology (Harvard-IV-4 dictionary) or politics (Loughran

& McDonald Financial Sentiment Dictionary). A big portion of literature is employing this kind of text categorization, for instance, Yulan & Zhou (2010). The more detailed mechanism of lexicon-based text classification will be described in the methodology chapter.

For the linguistic or dictionary based approach the the text preprocessing procedure is not that crucial, since it is only taking the set of words defined in the selected lexicon. However, for the machine learning approach along (without combination with dictionary based approach) the text preprocessing is very important. As stated by Wang & Ho (2016) the basic preprocessing usually involves:

- Stemming (minimizing inflected or derived words to their root)
- Tokenization (classifying parts of a string into input items)
- Stop-word removal (removing the noisy words, e.g prepositions)

Ponmuthuramalingam & Devi (2010) went beyond the basic steps and in his work included also other dimension-reducing steps such as conversion to lower case letters, punctuation, numbers and web page links removal.

Chapter 3

Text-based Sentiment Derivation Techniques

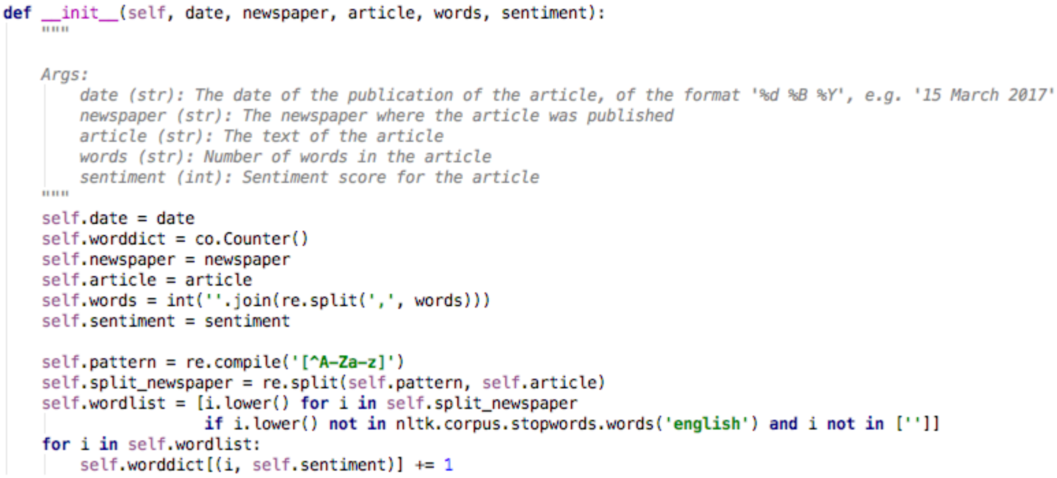
As it was already stated in Chapter 2, there are two main approaches for text-based sentiment generation. The first one is based on machine learning algorithms and the second one is a lexicon-based approach. Machine learning, in turn, is divided into supervised and unsupervised learning techniques. Supervised learning is inferring a function from labeled training data, while unsupervised learning trying to find hidden structure in unlabeled data. In other words, unsupervised methods are not using these labeled training documents and are used less frequently than the supervised approaches. The following sections cover methodology for both supervised machine learning and lexicon-based approach.

This paper examines both approaches separately and then the combination of them (hybrid Naive Bayes approach). By combining two approaches we mean taking only the words contained in the pre-defined dictionary and then applying the machine learning algorithm for this set of words. By doing so, we make sure that the pure Naive Bayes algorithm is not severely lacking any text preprocessing steps, which can influence the score predictions. That is, we assure we have successfully removed all noisy words and our estimates make sense after considering full article for processing. These approaches are executed using the Python programming language, created by Guido van Rossum (1991). The use of Phyton is justified by fairly simple user-friendly interface. Additionally, the PyCharm integrated environment for Python code analysis is used. PyCharm was developed by Czech software development company Jet-

Brains (2010).

Prior applying either of the mentioned approaches one need to process the text of the articles itself in order to correctly extract the relevant features. There are several steps needed to be implemented in order to extract and analyze the words it contains. The number and complexity of steps depends on the purpose and scope of the study. In our analysis, to obtain the reliable data, we proceeded with tokenization, stop words removal, accounting for capital and small letters, removal of hyperlinks and other unnecessary information. For the lexicon approach only the first one is relevant. By tokenization we break the string into words and punctuation signs. For stop words removal procedure we employed the NLTK (Natural Language Toolkit) list of stop words and excluded them from our data sample. These words may add additional noise to our estimation and can lead to the loss of precision and meaningless results (Fernández *et al.* 2014). The stop word list contains 128 English words in total (e.g ”i”, ”me”, ”my”, ”it” etc.). Figure 3.1 below represents the procedure of text processing incorporating stop words removal.

Figure 3.1: Source code: Text preprocessing



```
def __init__(self, date, newspaper, article, words, sentiment):
    """

    Args:
        date (str): The date of the publication of the article, of the format '%d %B %Y', e.g. '15 March 2017'
        newspaper (str): The newspaper where the article was published
        article (str): The text of the article
        words (str): Number of words in the article
        sentiment (int): Sentiment score for the article
    """
    self.date = date
    self.worddict = co.Counter()
    self.newspaper = newspaper
    self.article = article
    self.words = int(''.join(re.split(',', words)))
    self.sentiment = sentiment

    self.pattern = re.compile('[^A-Za-z]')
    self.split_newspaper = re.split(self.pattern, self.article)
    self.wordlist = [i.lower() for i in self.split_newspaper
                     if i.lower() not in nltk.corpus.stopwords.words('english') and i not in ['']]
    for i in self.wordlist:
        self.worddict[(i, self.sentiment)] += 1
```

Source: Author’s computations.

3.1 Lexicon-based Approach

Lexicon-based approach requires the calculation od semantic orientation of a text as an average of the semantic orientations of individual words or phrases it contains. By semantic orientation we understand the polarity and strength of the words and phrases. First, the polarity and strength of each word in a sentence is computed and then the overall sentiment of the sentence is calculated. At the document level the goal is to categorize the full document into positive or negative class (or more classes).

Dictionary-based approach utilizes the sentiment dictionary with opinion words, that are matched with the provided data to define the polarity of the text. Eventually, the semantic orientation of the text is quantitatively expressed as numerical value, based on scoring provided in the dictionary. The ranking of the opinion words as well as their choice vary across dictionaries. Most of the dictionaries provide the binary classifications (e.i classifying the selected opinion words as positive and negative) and therefore utilizing only the polarity feature of the semantic orientation. There are, however, some dictionaries, that apply more ranking levels and thus incorporating also the strength of the opinion feature.

In our work, we will use two dictionaries the binary one and the one with ranking that ranges from -5 (extremely negative) to +5 (extremely positive). The first dictionary "Opinion Lexicon" was developed by professors Minqing Hu and Bing Liu (2004), University of Illinois at Chicago. This opinion lexicon contains 6800 english words divided into positive and negative ones. It is widely used in academic literature (Cambridge University Press) and has a vast media coverage (The New York Times, The Economist, Business Week, etc.). This list of words was compiled over many years and is still being improved constantly. The second dictionary "AFINN-111" comprises 2477 words and phrases of general origin, varying with an integer between minus five and plus five. The scores were manually assigned to words and manually labeled by Finn Arup Nielsen (professor of Informatics and Mathematical Modelling, Technical University of Denmark) in 2009-2011.

3.2 Machine Learning Approach: Naive Bayes Algorithm

The Bayesian categorization represents a supervised learning technique as well as a statistical method for classifications. Naive Bayes classifiers is a family of probabilistic algorithms, that make use of properties of probability theory and

Bayes Theorem to predict the class of a particular sample. They are probabilistic, meaning that they compute the probability of each class for a particular sample, and afterwards return the class with the highest probability. Importantly, those algorithms are based on the following principle: it is assumed, that the value of a particular feature is completely independent of the value of any other feature, given the category variable.

Specifically, the task of classification is to take an input d and a fixed set of output categories C = c_1 , c_2, ..., c_M and return a predicted category $c \in C$, where d stands for document and c for category/class. In the supervised case we have a training set of N documents, where each of them is manually assigned to a class: $(d_1, c_1), ..., (d_N, c_N)$. Our aim is to teach a classifier so that it is able to perform a mapping from a new document d to its correct category $c \in C$. Additionally, the probabilistic classifier (Naive Bayes) will show us the probability of the observation being in the particular category. That is, we can formulate the intuition of the classifier as follows: a text document is viewed as if it were a bag-of-words. Bag-of-words stands for an unorganized set of words, where their relative positions are ignored and only their frequency in the document matters. Naive Bayes is a probabilistic classifier, since for a given document d out of all categories $c \in C$ the classifier outputs the category $\hat{c}$ having the maximum posterior probability given the document:

$$\hat{c} = \underset{c \in C}{argmax} P(c|d), \tag{3.1}$$

where $\hat{c}$ is an estimate of the correct category.

Then, we can convert the above equation into other probabilities with some useful properties, using the Bayes rule:

$$P(x|y) = \frac{P(y|x)P(x)}{P(y)}. \tag{3.2}$$

It allows us to slice any conditional probability $P(x|y)$ into three other probabilities. Combining the above two equations we get:

$$\hat{c} = \underset{c \in C}{argmax} P(c|d) = \underset{c \in C}{argmax} \frac{P(d|c)P(c)}{P(d)}. \tag{3.3}$$

Further, we can cross out the denominator $P(d)$, since we will calculate $\frac{P(y|x)P(x)}{P(y)}$ for each possible category and $P(d)$ is the same for each category. That is, we are always asking about the most probable category for the same document d, which must have the same probability $P(d)$:

$$\hat{c} = \underset{c \in C}{argmax} P(c|d) = \underset{c \in C}{argmax} P(d|c)P(c). \tag{3.4}$$

This formula says, that we calculate the most probable category $\hat{c}$ given some document d by selecting the category, which yields the highest product of two probabilities: the prior probability of the category $P(c)$ and the likelihood of the document $P(d|c)$:

$$\hat{c} = \underset{c \in C}{argmax}\ \underbrace{P(d|c)}_{\text{likelihood}}\ \underbrace{P(c)}_{\text{prior}}. \tag{3.5}$$

Without loss of generalization, we will define a document d as a set of features $f_1, f_2, ..., f_n$:

$$\hat{c} = \underset{c \in C}{argmax}\ \underbrace{P(f_1, f_2, ..., f_n|c)}_{\text{likelihood}}\ \underbrace{P(c)}_{\text{prior}}. \tag{3.6}$$

Obviously, it would be very difficult to calculate this equation if not to apply the assumption of bag-of-words and the Naive Bayes conditional independence assumption, where the probabilities $P(f_i|c)$ are independent given the class c. Thus, we can "naively" multiply them:

$$P(f_1, f_2, ..., f_n|c) = P(f_1|c) \cdot P(f_2|c) \cdot ... \cdot P(f_n|c). \tag{3.7}$$

Hence, the category proposed by Naive Bayes classifier will look as follows:

$$c_{NB} = \underset{c \in C}{argmax} P(c) \prod_{f \in F} P(f|c) \tag{3.8}$$

Naive Bayes computations are conducted in log-space, to evade an underflow and acceleration. Consequently, it is generally expressed as:

$$c_{NB} = \underset{c \in C}{argmax} log P(c) + \sum_{f \in F} log P(f|c). \tag{3.9}$$

Naive Bayes is a linear classifier, since it uses a linear combination of inputs to obtain a classification decision. It is worth to mention, that even despite its simplicity, it can often outperform more sophisticated classification methods.

3.2.1 Training the Naive Bayes Classifier

Commonly, the parameter estimation for Naive Bayes models utilizes the method of maximum likelihood. For the document prior $P(c)$ we would like to find out the percentage of documents in our training set, that belong each category c. Suppose, N_c is the number of documents in our training data with category c and N_{doc} is the total number of documents.
Then:

$$\hat{P}(c) = \frac{N_c}{N_{doc}}. \tag{3.10}$$

First step in obtaining the probability $P(f_i|c)$ is to determine $P(w_i|c)$ (assuming the feature is only the existence of particular word among all words in the document). It can be achieved by taking a fraction of times the word w_i occurs across all words in all documents of topic c. Secondly, we will merge together all documents with class c into one large "class c" text. Consequently, by using the frequency of w_i in this merged document, we will get a maximum likelihood estimate of the probability:

$$\hat{P}(w_i|c) = \frac{count(w_i, c)}{\sum\limits_{w \in V} count(w, c)}, \tag{3.11}$$

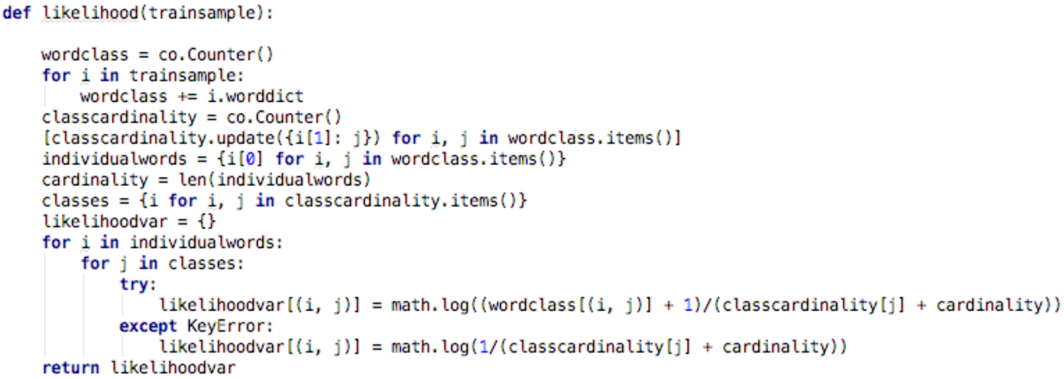
where V stands for vocabulary, containing the union of all kind of words in all categories, not just the words in one category c.

Unfortunately, maximum likelihood estimation has a major drawback: if a certain category and feature value from the test dataset have never appeared in pair in the training dataset, then the probability estimate based on frequency will return zero, and hence will be unable to calculate a prediction. This then eliminates all information in other probabilities during their multiplication. To solve this problem, we can incorporate the smoothing technique, called Laplace smoothing. It modifies all probability estimates in such a way, that no probability is ever an exact zero. In our specific case, we will implement an add-one smoothing, which is simply adding the smoothing parameter (1) to both numerator and denominator. Hence, our final equation will be represented as follows:

$$\hat{P}(w_i|c) = \frac{count(w_i, c) + 1}{\sum\limits_{w \in V} (count(w, c) + 1)} = \frac{count(w_i, c) + 1}{(\sum\limits_{w \in V} count(w, c)) + |V|}. \tag{3.12}$$

Figure 3.2 below shows the extract from the Python code, with the command for training sample returning the dictionary with $(word, class)$ tuple and the likelihood $P(w|c)$ as a value. "Trainsample" argument is a list of train articles objects. This code is using the methodology described above, utilizing the add-one Laplace smoothing.

Figure 3.2: Source code: Likelihood Estimation for the train sample



```
def likelihood(trainsample):

    wordclass = co.Counter()
    for i in trainsample:
        wordclass += i.worddict
    classcardinality = co.Counter()
    [classcardinality.update({i[1]: j}) for i, j in wordclass.items()]
    individualwords = {i[0] for i, j in wordclass.items()}
    cardinality = len(individualwords)
    classes = {i for i, j in classcardinality.items()}
    likelihoodvar = {}
    for i in individualwords:
        for j in classes:
            try:
                likelihoodvar[(i, j)] = math.log((wordclass[(i, j)] + 1)/(classcardinality[j] + cardinality))
            except KeyError:
                likelihoodvar[(i, j)] = math.log(1/(classcardinality[j] + cardinality))
    return likelihoodvar
```

Source: Author's computations.

3.2.2 Test sample and Cross-validation

After training the classifier on the training sample, it is necessary to validate and assess it's performance based on data not included in the initial training sample. For this reason, the testing dataset should be created. There are several ways to split the original dataset for training and testing. One of them is simply using a fixed 70%-80% of data for training and remaining 30%-20% for testing. While such procedure avoids overfitting, it can however result in problem when the test sample will not be large enough to be representative. Additionally, by partitioning the initial set into two fixed sets will dramatically reduce the number of examples for training.

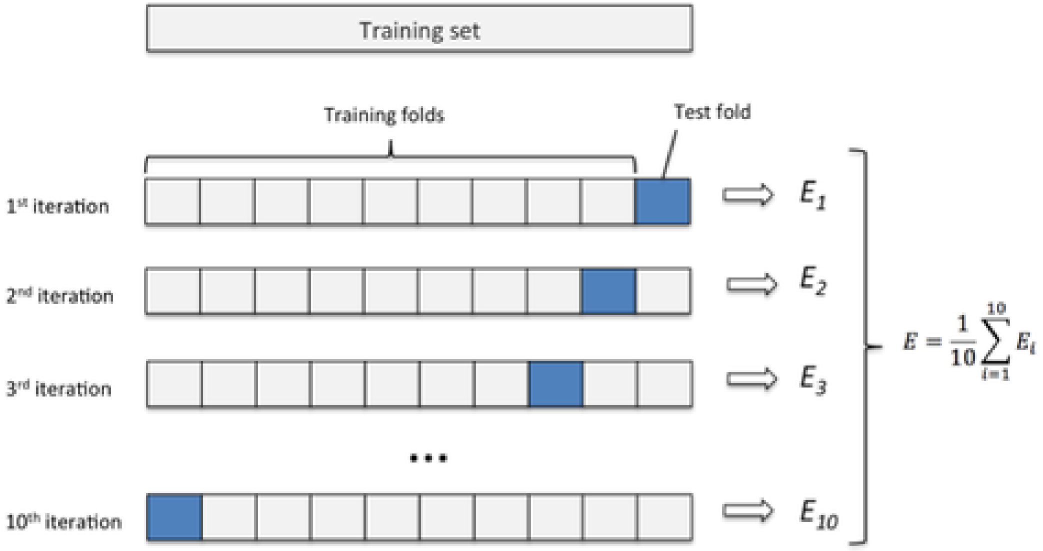
A solution to this will be a method called k-fold cross-validation, where all the data is utilized for both training and testing. The procedure is the following:

- The training set is randomly split into k smaller sets
- The model is trained using $k-1$ folds as training sample and hold out the rest as a test sample

- Validating and computing the error rate on the test sample
- Repeat the procedure with different randomly selected split
- Do the sampling process k times, average these k runs to get an average error rate

We have chosen 10 iterations (10-fold cross-validation), taking into account the size of our training sample being 500 observations. According to the reviewed literature, this approach provides valid and robust results and preferred by many researches (Salzberg 1997). The illustrative example of 10-fold cross validation is shown in Figure 3.3.

Figure 3.3: 10-fold cross-validation scheme



Source: Python Machine Learning, 2nd Edition.

Figure A.1 represents a function, that calculates the likelihood of combinations of different classes for the test sample. It appends the posterior probability attribute to the dataset object, i.e. aggregate $P(c|w)$ and also final decision that is the maximization procedure over different classes. "Dataset" argument is a list of test article objects. "Index" stands for an index of values, which are classified according to this algorithm, i.e. in the case when not all objects in dataset are being classified. "Priorvar" is a dictionary returned from prior function. "Likelihoodvar" is a dictionary returned from likelihood function.

3.2.3 Classifiers' performance evaluation

The goal of measuring performance in machine learning is to determine the usefulness of our learned classifier and learning algorithm. Those measures are focused on ability of a classifier to identify classes correctly. Before introducing those metrics it is important to define a following notions: true positives (tp_i) is the number of correctly recognized observations for class C_i; true negative (tn_i) is the number of correctly recognized observations that do not belong to the class C_i; false positive (fp_i) is the number of observations that were incorrectly assigned to the class C_i; false negative (fn_i) is the number of observations that were not recognized as belonging to the class C_i.

The most widely used measures to evaluate the effectiveness of classification are: accuracy, precision, recall, F-score. Accuracy is considered as a simplest and easiest to derive measure and defined as a fraction of the number of correct predictions over the total number of predictions. Hence, having a multi-class problem, where each document is assigned to one of the five classes ranging from [-2; 2], we will take a sum of correct predictions per each class divided by the total number of predictions to obtain an overall accuracy of the classifier. We considered the range from -2 to +2 to be an optimal one taking into account the size of the training dataset to capture both the magnitude and direction of the sentiment. At the same time there should be enough observations for each label to be able to reasonably train the classifier. The formula for computation of accuracy will look as follows:

$$\text{Accuracy} = \frac{\sum\limits_{i=1}^{l} tp_i}{N}, \qquad \text{where } l \text{ is the number of classes} \tag{3.13}$$

However, if the data is skewed or unbalanced, accuracy may not fully reflect and indicate the performance. For example, when considering binary classification, if 1% of the articles have negative sentiment then an accuracy of 99% can be achieved simply by always classifying an article as positive. Even though our data does not fall into such an extreme case, we do not have it also completely uniformly distributed (will be discussed later in this section). For this reason we will employ additional measures (precision, recall and F-score) to make sure we assessed our classifier in the correct way.

Precision is defined as proportion of instances the model classified correctly to the total number of true positives and true negatives examples. Thus, precision shows the exactness of the classifier with respect to every individual class. Recall is defined as a proportion of instances the model classified correctly to the total number of true positives and false negatives. In other words, recall determines the completeness of the classifier with respect to each class. To apply those metrics for the multi-class classifier we will take the average over classes. Combining together average precision and average recall metrics we will obtain the average F-score (van Rijsbergen, 1975):

$$\text{Average F-score} = \frac{2 * AveragePrecision * AverageRecall}{AveragePrecision + AverageRecall}. \tag{3.14}$$

A higher F-score indicates better joint recall and precision compared to a lower F-score, and thus a better performance of the classifier. F-measure is thus a harmonic mean of precision and recall, where harmonic mean is defined as a reciprocal of the arithmetic mean of reciprocals:

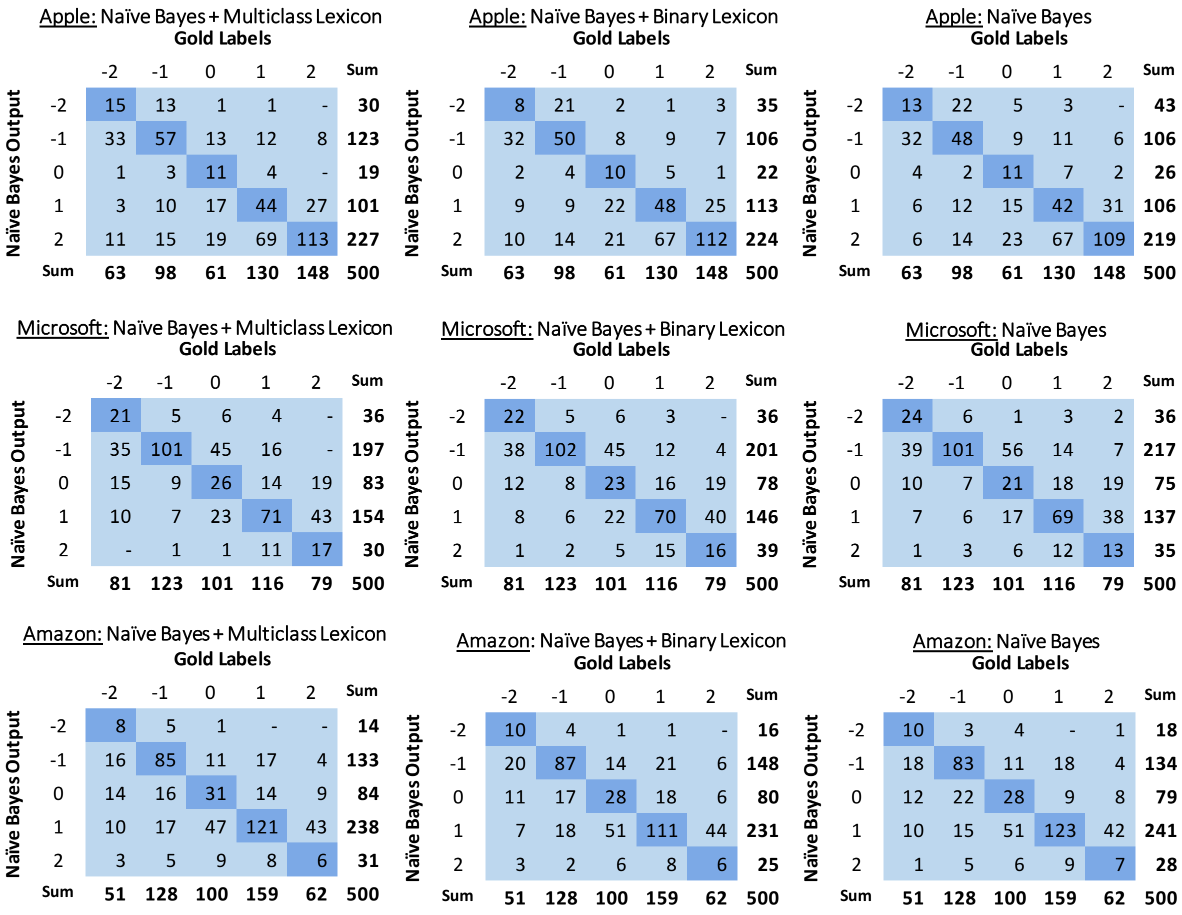
$$\text{Harmonic Mean}(x_1, x_2, ..., x_n) = \frac{n}{\frac{1}{x_1} + \frac{1}{x_2} + ... + \frac{1}{x_n}}. \tag{3.15}$$

$$\text{F-score} = \frac{n}{\frac{1}{x_1} + \frac{1}{x_2} + ... + \frac{1}{x_n}}. \tag{3.16}$$

The values of the metrics for each selected stock (Apple, Microsoft, Amazon) will be obtained from the confusion matrix built based on actual "gold labels" (human-assigned labels) and the predicted sentiments calculated by Naive Bayes Algorithm. Each row of the matrix represents the instances of predicted class while each column shows the instances in the actual class. The confusion matrix for three selected stocks is presented in the Figure 3.4 below. The accuracy and F-score results are shown in the Table 3.1, where from left to right: Naive Bayes approach combined with multiclass lexicon, Naive Bayes approach combined with binary lexicon, pure Naive Bayes approach.

From the confusion matrix you can see that for a given stock each classifier has similar pattern in resulting predictions. For example, for the Apple stock, in each case the very positive class with "2" values and negative class with "-1" values were the best performing ones. Also, the confusion matrix shows the distribution of the sentiment values of the training set. For the Apple stock, the very positive setiment with the assigned values ”2” belongs to the majority class consisting of 148 observations, while neutral setiment with values ”0” represents the minority class with 61 observations. The size of very negative sentiment class with values ”-2” is approximately the same as minority neutral class. For the Microsoft, the distribution is different: the class with negative sentiment ”-1” is the majority class, while very positive class with values ”2” is the minority one. In addition, comparing the distribution within stocks, the very negative class with values ”-2” is the largest for the Microsoft stock. For the Amazon, the extreme cases of ”-2” and ”2” are the least frequent.

Figure 3.4: Confusion matrix



Apple: Naïve Bayes + Multiclass Lexicon
Gold Labels

Naïve Bayes Output	-2	-1	0	1	2	Sum
-2	15	13	1	1	-	30
-1	33	57	13	12	8	123
0	1	3	11	4	-	19
1	3	10	17	44	27	101
2	11	15	19	69	113	227
Sum	63	98	61	130	148	500

Apple: Naïve Bayes + Binary Lexicon
Gold Labels

Naïve Bayes Output	-2	-1	0	1	2	Sum
-2	8	21	2	1	3	35
-1	32	50	8	9	7	106
0	2	4	10	5	1	22
1	9	9	22	48	25	113
2	10	14	21	67	112	224
Sum	63	98	61	130	148	500

Apple: Naïve Bayes
Gold Labels

Naïve Bayes Output	-2	-1	0	1	2	Sum
-2	13	22	5	3	-	43
-1	32	48	9	11	6	106
0	4	2	11	7	2	26
1	6	12	15	42	31	106
2	6	14	23	67	109	219
Sum	63	98	61	130	148	500

Microsoft: Naïve Bayes + Multiclass Lexicon
Gold Labels

Naïve Bayes Output	-2	-1	0	1	2	Sum
-2	21	5	6	4	-	36
-1	35	101	45	16	-	197
0	15	9	26	14	19	83
1	10	7	23	71	43	154
2	-	1	1	11	17	30
Sum	81	123	101	116	79	500

Microsoft: Naïve Bayes + Binary Lexicon
Gold Labels

Naïve Bayes Output	-2	-1	0	1	2	Sum
-2	22	5	6	3	-	36
-1	38	102	45	12	4	201
0	12	8	23	16	19	78
1	8	6	22	70	40	146
2	1	2	5	15	16	39
Sum	81	123	101	116	79	500

Microsoft: Naïve Bayes
Gold Labels

Naïve Bayes Output	-2	-1	0	1	2	Sum
-2	24	6	1	3	2	36
-1	39	101	56	14	7	217
0	10	7	21	18	19	75
1	7	6	17	69	38	137
2	1	3	6	12	13	35
Sum	81	123	101	116	79	500

Amazon: Naïve Bayes + Multiclass Lexicon
Gold Labels

Naïve Bayes Output	-2	-1	0	1	2	Sum
-2	8	5	1	-	-	14
-1	16	85	11	17	4	133
0	14	16	31	14	9	84
1	10	17	47	121	43	238
2	3	5	9	8	6	31
Sum	51	128	100	159	62	500

Amazon: Naïve Bayes + Binary Lexicon
Gold Labels

Naïve Bayes Output	-2	-1	0	1	2	Sum
-2	10	4	1	1	-	16
-1	20	87	14	21	6	148
0	11	17	28	18	6	80
1	7	18	51	111	44	231
2	3	2	6	8	6	25
Sum	51	128	100	159	62	500

Amazon: Naïve Bayes
Gold Labels

Naïve Bayes Output	-2	-1	0	1	2	Sum
-2	10	3	4	-	1	18
-1	18	83	11	18	4	134
0	12	22	28	9	8	79
1	10	15	51	123	42	241
2	1	5	6	9	7	28
Sum	51	128	100	159	62	500

Source: Author’s computations.

As for the accuracy and F-scores, all classifiers across all stocks provided similar results, with accuracy around 50% and F-score around 45%. This is more than two times better than predicting one particular class (20% threshold given 5 classes) and almost two times better than predicting the highest class. It is

Table 3.1: Accuracy and F-scores for all stocks

Apple Inc.	NB+Multiclass Lex.	NB+Binary Lex.	Naive Bayes
Accuracy	48%	46%	45%
F-score	45%	40%	40%
Microsoft			
Accuracy	47%	47%	46%
F-score	46%	44%	44%
Amazon			
Accuracy	50%	48%	50%
F-score	42%	42%	43%

Source: Author's computations.

good to notice that values of accuracy and F-scores do not diverge dramatically, indicating more or less consistent results for each class. Nevertheless, a slight advantage has a hybrid approach with multiclass lexicon. The resulting sentiment values obtained by this approach we will further incorporate to model volatility of particular stock and compare this method to simple lexicon-derivation approach in terms of accuracy of the volatility forecasts.

Table 3.2: Accuracy and F-scores with adjusted threshold

Apple Inc.	NB+Multiclass Lex.	NB+Binary Lex.	Naive Bayes
Accuracy	76%	75%	75%
F-score	65%	62%	62%
Microsoft			
Accuracy	66%	66%	65%
F-score	59%	59%	57%
Amazon			
Accuracy	65%	64%	65%
F-score	59%	58%	59%

Source: Author's computations.

It is important to notice that the most of mistakes the classifier has made when distinguishing between close classes, e.g "-2" versus "-1" and "1" versus "2". Hence, we can say that even though the accuracy per class may not be very high in all the cases, the classifier succeeded when predicting the direction of the sentiment. This information is very crucial for modelling volatility with the help of news sentiment. Analogously, such logic holds for two remaining stocks. To relax the threshold for measuring accuracy, we can recompute the metrics compressing the confusion matrix to three classes, e.i to assess only the direction of the sentiment predictions. That is, we will consider together classes ”-2” and ”-1” and ”2” and ”1”. The recomputed accuracies and F-scores are presented in Table 3.2. You can see that these values are now significantly higher, reaching the accuracy of 76% for the Apple stock.

Chapter 4

Conditional Heteroscedastic Models for Volatility

The studies conducting research on the stock markets mainly focus on the parametric methodology utilizing the GARCH family of models e.g Sharma *et al.* (1996), Sharma *et al.* (2005). In this context, these models are preferred, for example, over typical ordinary least squares, where the variance should be evenly distributed throughout the data. For the financial time series this homoscedasticity assumption doesn't necessary hold, so we need to count with the heteroskedastic type of errors, specifically conditional heteroscedasticity (conditional on past events). GARCH models are trying to fix the least squares deficiencies by modeling variance.

Autoregressive models are the most basic models commonly employed in the time series analysis. Autoregressive Moving-Average models of orders p, q combine AR(p) and MA(q) models. They were introduced by Box *et al.* (1970). AR(p) process describes a linear dependence of given variable on the previously observed data and is represented in the following form:

$$r_t = \alpha_0 + \alpha_1 r_{t-1} + \alpha_2 r_{t-2} + ... + \alpha_p r_{t-p} + \epsilon_t. \tag{4.1}$$

MA(q) is represented as averages of different subsets of the whole dataset over a given time periods:

$$r_t = \beta_0 + \beta_1 \epsilon_{t-1} + \beta_2 \epsilon_{t-2} + ... + \beta_q \epsilon_{t-q} + \epsilon_t, \tag{4.2}$$

where r_t is the observed times series and ϵ_t is residual or white noise with zero mean and constant variance; p and q are non-negative integers.

If we combine the two processes above, it will give us the ARMA(p,q) process:

$$r_t = \alpha + \sum_{i=1}^{p} \alpha_i r_{t-i} + \sum_{i=1}^{q} \beta_i \epsilon_{t-i} + \epsilon_t, \tag{4.3}$$

where r_t is the observed times series and ϵ_t is residual or white noise with zero mean and constant variance; p and q are non-negative integers. ARMA methodology can be employed only for stationary time series. That is, the series r_t are said to be stationary if:

- $E(\mathrm{r}_t)= \mu$ for $t = 1, 2, ...$ (mean is constant in t)
- $Var(r_t) = \sigma^2$ for $t = 1, 2, ...$ (variance is constant in t)
- $Cov(r_t, r_{t+k}) = \chi_k$ for $t = 1, 2, ...$ and $k \neq 0$ (covariance is constant in t)

Hence, both AR(p) and MA(q) processes have to be stationary. In essence, MA(q) meets this condition based on its definition. However, it does not always applies to AR(p) process. ARMA(p,q) will meet the stationarity condition if:

$$|\sum_{i=1}^{p} \alpha_i| < 1. \tag{4.4}$$

We can expand ARMA(p,q) model to get ARIMA (Autoregressive Integrated Moving Average) model by letting AR process to have a unit-root. In other words, ARIMA(p,d,q) would turn out to be unit-root nonstationary. To transform ARIMA into stationary one can use differencing (logarithmic). In most of the cases, it is enough to use ARIMA (p,1,q) model, which means that the differencing transformation was applied only once. Further differencing might be necessary, when the time series have multiple unit-roots, this, however, would lead to a loss of information.

4.1 ARCH

Financial data, however, has its specific features we need to count with. Firstly, it was emprically shown that large price chnages tend to be followed by large price changes and small price changes tend to be followed by small price changes, which further results in volatility clustering. Secondly, volatility is observed to be higher after negative price shocks than after positive price shocks of the same magnitude, this is called leverage effect. Thirdly, the log-returns are not usually normally distributed, but accumulate more extreme values as well as more values around the sample mean.

Autoregressive Conditional Heteroscedastic Model (ARCH) was first suggested by Engle (1982) where the return series can be modelled as follows:

$$r_t = \mu + a_t, \tag{4.5}$$

where μ is a conditional mean of r_t and h_t is a conditional variance of r_t; a_t is a shock or innovation of the return series at time t. We can then rewrite the above equation:

$$a_t = \sqrt{h_t}\epsilon_t, \tag{4.6}$$

where ϵ_t stands for independent and identically distributed random variables (iid) with zero mean and variance of one. In this model ε_t may follow normal, Student's t or generalized distribution. The underlying idea of ARCH is that the returns are serially uncorrelated but dependent. Hence, the common form of ARCH to model the conditional variance h_t, where the dependence is described with the function of lagged values is:

$$h_t = \alpha_0 + \sum_{i=1}^{q} \alpha_i a_{t-i}^2, \tag{4.7}$$

where $\alpha_0 > 0$ and $\alpha_i \geqslant 0$. The above equation says, that the variance of the error term in the certain period is dependent on a squared error term from the previous period.

Despite its usfulness, ARCH models have several major drawbacks. First, it assigns the same weight to both positive and negative innovations (shocks), even though the literature proves it to have an asymmetric effect. Secondly, sometimes large number of squared lagged residuals must be included to specify the model correctly. Thirdly, it was observed that ARCH models tend to overpredict the volatility. In order to adress some of the listed issues, a vast number of extensions of the standard ARCH model have been proposed, namely, it was generalised to GARCH, which will be a topic of the next section.

4.2 GARCH

Generalized ARCH model (GARCH) followed the standard ARCH model, that was proposed by Bollerslev (1986). It is generalized by adding the autoregressive terms of the volatility. GARCH equation models conditional variance as a weighted average of three elements: long-run average variance, estimate of the conditional variance from the past period and the error term from the past period. Bollerslev (1986) in his research observed, that with the increasing number of lags even very simple GARCH models fitted better than ARCH models. The GARCH model is given by the set of two equations, mean and variance:

$$r_t = \mu + \epsilon_t = \mu + \sqrt{h_t} z_t, \tag{4.8}$$

$$h_t = \alpha_0 + \sum_{i=1}^{q} \alpha_i \epsilon_{t-i}^2 + \sum_{j=1}^{p} \beta_j h_{t-j}, \tag{4.9}$$

where $\epsilon_t \sim N(0, h_t)$, z_t is is independent identically distributed standard normal variable. The assumptions that need to be met are: α_0>0; $\alpha_i \geqslant 0$ and $\beta_j \geqslant 0$ for $i = 1...q$ and $j = 1...p$. The α coefficient measures the short-term impact of ϵ_t on conditional variance and β coefficient measures the long-term impact on conditional variance.

GARCH (p,q) is stationary, if the sum of all a_i and all β_i is is strictly smaller than 1. In fact, GARCH model is built upon ARCH model by letting the lagged conditional variances of GARCH to come into equation. We say that a_i is an ARCH parameter and β_i is a GARCH parameter. Therefore, ARCH can be viewed as special case of GARCH specification, i.e. GARCH (1,0).

4.3 Model Specification with News Sentiment

The methodology described in the previous section is focused on standard form of GARCH model, where only the time series data is employed. The more detailed derivations and explanations of equations and its assumptions are avaliable in book Francq & Zakoian (2010).

The goal of this section is to specify a model incorporating additional information that could potentially improve a standard GARCH modelling abilities. We add a sentiment of the news articles about particular stock as an exogenous explanatory variable into variance equation of the GARCH model. That is, we assume that the information derived from the news articles is influential for market agents decision-making, in particular for volatility modelling and forecasting. Moreover, we are analyzing separately the negative and positive orientations of the news articles in line with behavioral finance frameworks.

The GARCH variance rearranged equation will take the following form:

$$h_t = \alpha_0 + \sum_{i=1}^{q} \alpha_i \epsilon_{t-i}^2 + \sum_{j=1}^{p} \beta_j h_{t-j} + \theta_1 SP_{t-1} + \theta_2 SN_{t-1}, \tag{4.10}$$

where $\theta_1 SP_{t-1} - \theta_2 SN_{t-1} = \theta S_{t-1}$ is the daily news sentiments with SP_{t-1} being a vector of positive sentiment values for $S_{t-1} > 0$ and SN_{t-1} being a vector of negative sentiment values for $S_{t-1} < 0$. Moreover, SN_{t-1} is entering the equation in the absolute values to ensure its positivity, that is $SN_{t-1} = |S_{t-1} < 0|$. In other words, by splitting our $k \times 1$ vector of news sentiments to the ones that carry positive mood and the ones that carry negative mood we are able to assess and analyze their effects separately, rather than looking at the overall impact. In other words, each day's return will correspond to one agreggated sentiment value, either negative, positive or neutral. To the neutral sentiment the value of zero is assigned and is equivalent to the no sentiment values, that is for the days when there were no articles published. Hence, we say that news carrying the neutral mood do not have any significant impact on the volatility and is thus neglected.

To make the predictions possible, we will utilize the sentiment values from time t-1. As it was described in the methodology section, in case of multiple news per day the sentiments of each news are averaged together in order to obtain the agreggate daily sentiment. Additionally, the news from the weekends and public holidays are moved to the closest day of the open stock. This augmented GARCH model is applied for three selected stock (Apple, Microsoft, Amazon) and uses the sentiment values derived by three different approaches: with the help of multiclass lexicon, binary lexicon and hybrid Naive Bayes approach.

4.4 Out-of-sample model performance

The predictability of the employed volatility models is assessed not only based on the in-sample fit, but also by performing an out-of-sample fit obtained from a sequence of rolling regressions. The models with the most suitable specifications were constructed by taking into account only the properties of the series in the in-sample subset then the efficiency of the models is evaluated on the out-of-sample data. In this thesis, we will focus solely on the one-step-ahead volatility forecasts. To evaluate the out-of-sample performance of the forecast we will employ two different loss functions: root mean squared error (RMSE) and mean absolute error (MAE). These statistics provide a measure of the distance of the forecasted from the "actual" values.

Both statistics have its advantages as well as disadvantages, where the choice of either depends on particular case. RMSE is the most common metrics, which measures an average magnitude of the error. The errors are squared before they get averaged, implying it gives a relatively high weight to the large errors. MAE measures an average magnitude of the errors, regardless their direction and giving an equal weight to all individual differencies. The described loss functions are constructed as follows:

$$RMSE = \sqrt{\frac{1}{T}\sum_{i=1}^{T}(\widehat{\sigma_t^2} - \sigma_t^2)} \tag{4.11}$$

$$MAE = \frac{1}{T}\sum_{i=1}^{T}|\widehat{\sigma_t^2} - \sigma_t^2| \tag{4.12}$$

where T is number of forecasts made, σ_t^2 is a proxy for actual volatility and $\widehat{\sigma_t^2}$ is one-step-ahead volatility forecast.

In order to calculate those metrics, one need to know the value of the "actual" volatility. Since conditional volatility is unobservable due to its latent nature, a proxy for actual or true volatility needs to be defined. There exist several ways to derive an actual values of volatility. The most common proxy which has been used for a long time in vast amount of literature is simply daily squared returns series (Cumby & Figlewski 1993). However, such estimator is argued to be very noisy and can lead to poor out-of-sample evaluation, even in case of good in-sample fit (Andersen & Bollerslev 1998).

Except using only the closing prices for construction of the volatility proxy, Garman & Klass (1980) offered to ulitize also the information about open, high and low prices, which are commonly freely available. It is defined by the following equation:

$$\sigma^2_{GK,t} = 0.5[ln(\frac{H_t}{L_t})]^2 - [2ln2 - 1][ln(\frac{C_t}{O_t})]^2, \tag{4.13}$$

where H_t is the highest price of an asset at time t; L_t is the lowest price of an asset at time t; C_t is a price of an asset at the market close at time t; O_t is a price of an asset at the market open at time t. Such measure contains more information compared to a simple daily return, which is built solely on the closing price, since it includes the price fluctuations throughout the day. We will compare this measure to the simple squared returns to assess the improvement.

Chapter 5

Data

5.1 Financial Time-series Data

In this thesis we are analysing our hypotheses on the example of data from three biggest information technology companies: Apple Inc., Microsoft Corporation, Amazon.com Inc. This companies were chosen based on their market capitalization, revenues and news coverage. Also, it would be interesting to analyze the companies belonging to similar industries to have more comparable results. Additionally, all companies are US companies, trading at the New York stock exchange. Table 5.1 represents the basic facts about selected companies relevant for stock market trading.

Table 5.1: Information about selected companies

Name	Industry	Ticker	Country	Stock Exchange
Apple Inc.	IT	AAPL	U.S	New York
Microsoft Corporation	IT	MSFT	U.S	New York
Amazon.com	IT	AMZN	U.S	New York

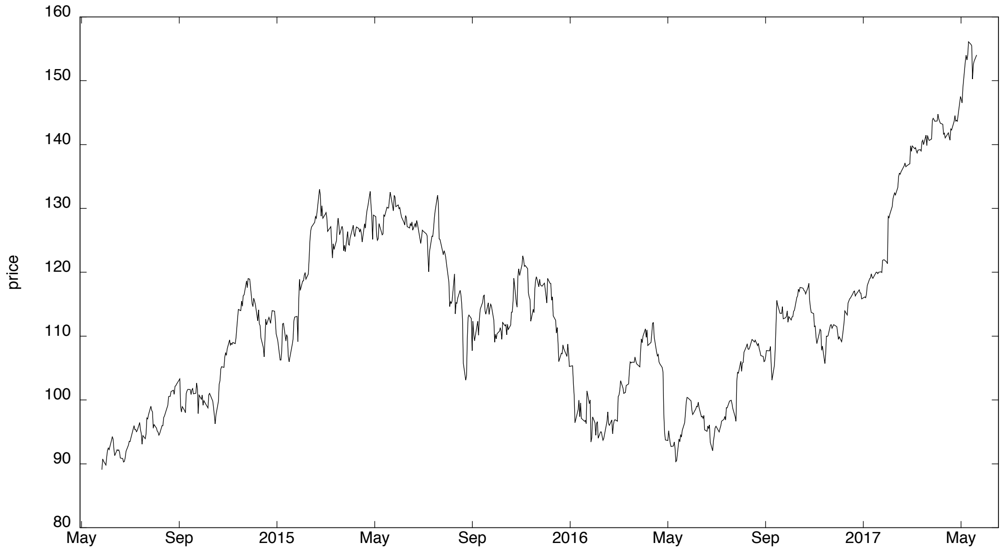
Source: Author's computations.

Their daily historical closing prices are accumulated for the period from May 2014 untill May 2017 from the Yahoo Finance database. The closing prices are adjusted for various corporate actions and distributions like stock splits, dividends and rights offerings. The dataset doesn't contain the values for weekends and public hodildays, since the stock is closed at these days.

We will start our analysis with check whether the selected time-series are stationary, since it is a necessary condition for GARCH modelling. From the

Figure 5.1 we can observe that Apple time series is not stationary in the analyzed period.

Figure 5.1: Time series plot of Apple 2014-2017



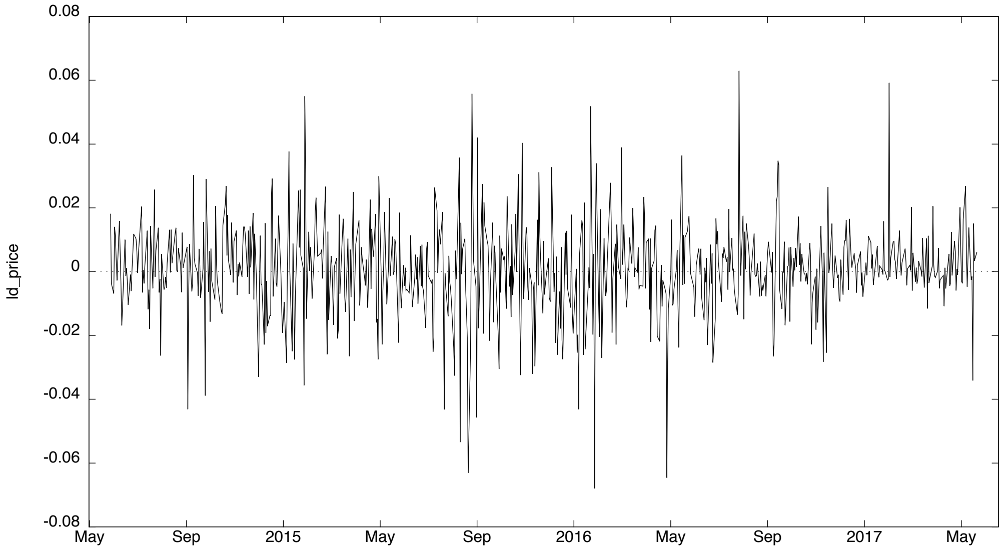
Source: Author's computations.

This observation is also supported by the correlogram (see Appendix, Figure A.2), from which we can see that ACF doesn't decline with the number of observations. We confirmed non-stationarity by formal analysis implementing Augmented Dickey Fuller Test (ADF). According to the results, we can not reject null hypothesis of non-stationarity (existence of the unit-root) with p-value of 0.843, because value of test statistic is bigger than critical values (Appendix A.3).

We stationarize the data by taking first log difference, thus obtaining log returns of the Apple stock. On the Figure 5.2 you can find that log returns are stationary and have nearly zero-mean. Nevertheless, the volatility of the Apple returns is visibly non-constant. In addition we can confirm the stationarity with ADF and KPSS tests (see Appendix, Figure A.4 and A.5). According to the results of KPSS test, we do not reject null hypothesis of stationary with p-value of 0.187. Aditionally, ADF test rejects the null hypothesis of existence unit root.

The same analysis we need to perform for Microsoft time series. From the Figure 5.3 we can observe that Microsoft time series is not stationary in the analyzed period. This observation is also supported by the correlogram, from

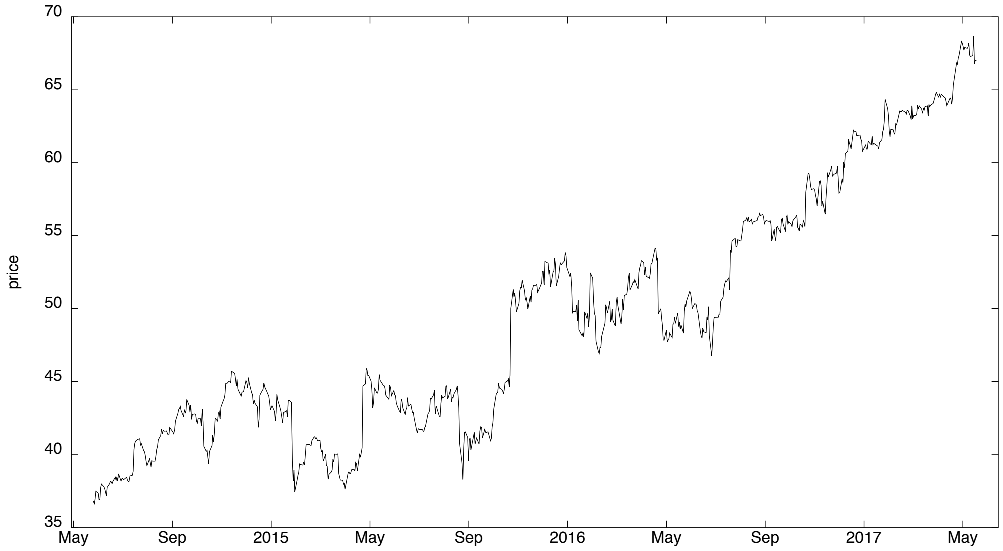
Figure 5.2: Log returns of Apple stock 2014-2017



Source: Author's computations.

which we can see that ACF doesn't decline with the number of observations. We confirmed non-stationarity by formal analysis implementing Augmented Dickey Fuller Test (ADF). According to the results, we can not reject null hypothesis of non-stationarity (existence of the unit-root) with p-value of 0.2033, because value of test statistic is bigger than critical values.

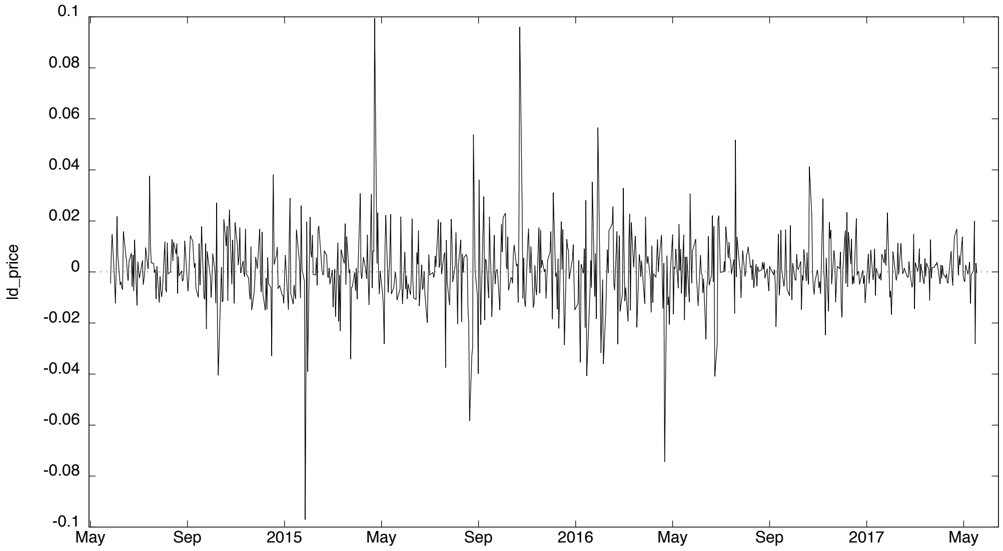
Figure 5.3: Time series plot of Microsoft 2014-2017



Source: Author's computations.

We stationarize the data by taking first log difference, thus obtaining log returns of the Microsoft stock. On the Figure 5.4 you can find that log returns are stationary and have nearly zero-mean. Nevertheless, the volatility of the Microsoft returns is visibly non-constant. In addition we can confirm the stationarity with ADF and KPSS tests. According to the results of KPSS test, we do not reject null hypothesis of stationary with p-value bigger than critical. Also, ADF test rejects the null hypothesis of existence unit root.

Figure 5.4: Log returns of Microsoft stock 2014-2017



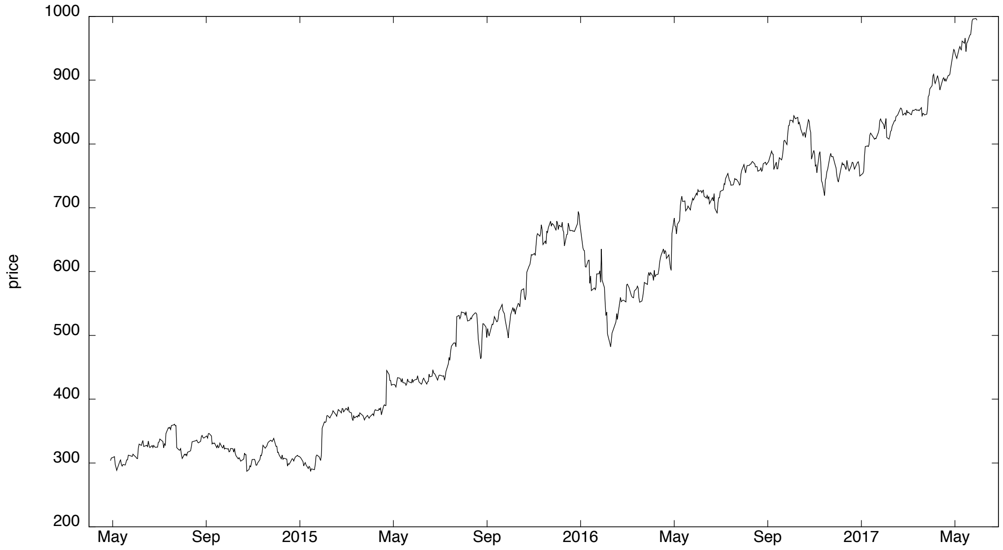
Source: Author's computations.

Analogously, we will proceed with the Amazon.com time-series. From the Figure 5.5 we can observe that Amazon.com time series is not stationary in the analyzed period. This observation is also supported by the correlogram, from which we can see that ACF doesn't decline with the number of observations. We confirmed non-stationarity by formal analysis implementing Augmented Dickey Fuller Test (ADF). According to the results, we can not reject null hypothesis of non-stationarity (existence of the unit-root) with p-value of 0.7714.

We stationarize the data by taking first log difference, thus obtaining log returns of the Amazon stock. On the Figure 5.6 you can find that log returns are stationary and have nearly zero-mean. Nevertheless, the volatility of the Amazon returns is visibly non-constant. In addition we can confirm the stationarity with ADF and KPSS tests. According to the results of KPSS test, we do not reject null hypothesis of stationary with p-value bigger than critical. Also, ADF test rejects the null hypothesis of existence unit root.

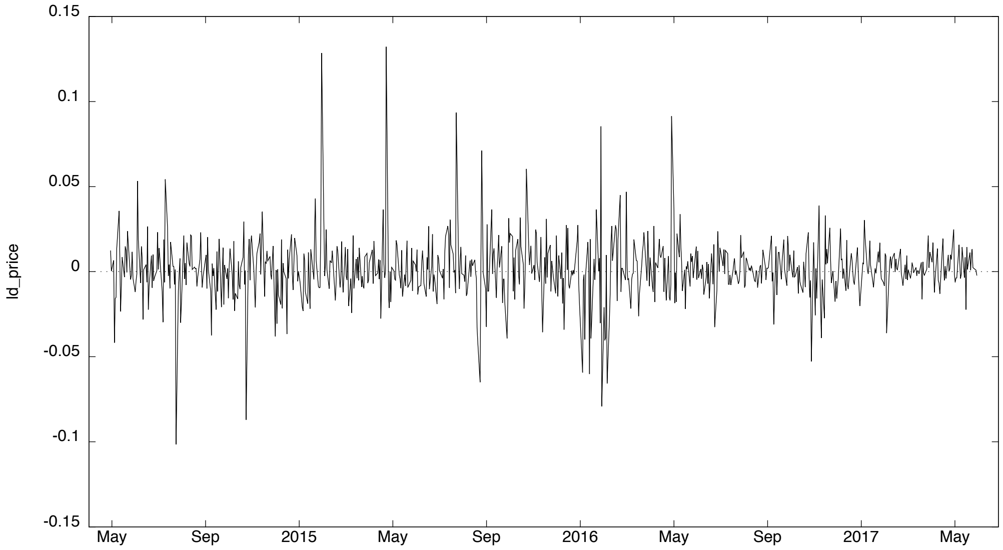
The next table, Table 5.2, describes the summary statistics of selected stock

Figure 5.5: Time series plot of Amazon.com 2014-2017



Source: Author's computations.

Figure 5.6: Log returns of Amazon stock 2013-2017



Source: Author's computations.

returns over the full sample from May 2014 to May 2017. We will further split it into two subsamples May 2014 - Jan 2017 and Feb 2017-May 2017 in order to construct the forecast model.

Table 5.2: Summary statistics of stocks log returns

Stock	Obs.	mean	median	min.	max.	sd.	skewness	kurtosis
Apple	750	0.0007	0.0006	-0.0679	0.0629	0.0145	-0.1859	3.9507
Microsoft	752	0.0008	0.0003	-0.0971	0.0994	0.0143	0.2153	9.835
Amazon	752	0.0015	0.0011	-0.1015	0.1322	0.0184	0.6856	10.326

Source: Author's computations.

According to the Table 5.2 the average return for all the stocks over the analyzed time horizon is positive. Kurtosis, which is the measure of tallness and sharpness of the central peak is exceeding the threshold of three across all stocks (the threshold value of three is a kurtosis of normal distribution). Apple returns are, however, close to normal distribution. Skewness describes the asymmetry of the probability distribution around its mean. The absolute values of skewness for all three stocks is less than one, indicating that our distributions are moderately skewed. The Apple stock returns are skewed left, meaning that the left tail is longer, whereas Amazon and Microsoft returns are skewed to the right.

5.2 News Sentiment Data

All news articles were collected from Factiva database, through the subscription of the Charles University. Factiva is a Dow Jones and Reuters company providing the news feeds and press releases from various sources like The Wall Street Journal, The Financial Times etc. In order to avoid the duplicity of news we will take the news releases only from a single source - The Wall Street Journal. The choice is partially justified by the company selection: Apple Inc, Microsoft Corporation and Amazon.com are the U.S companies with majority of news released by local agencies, written in English. The English language is crucial for text processing procedure.

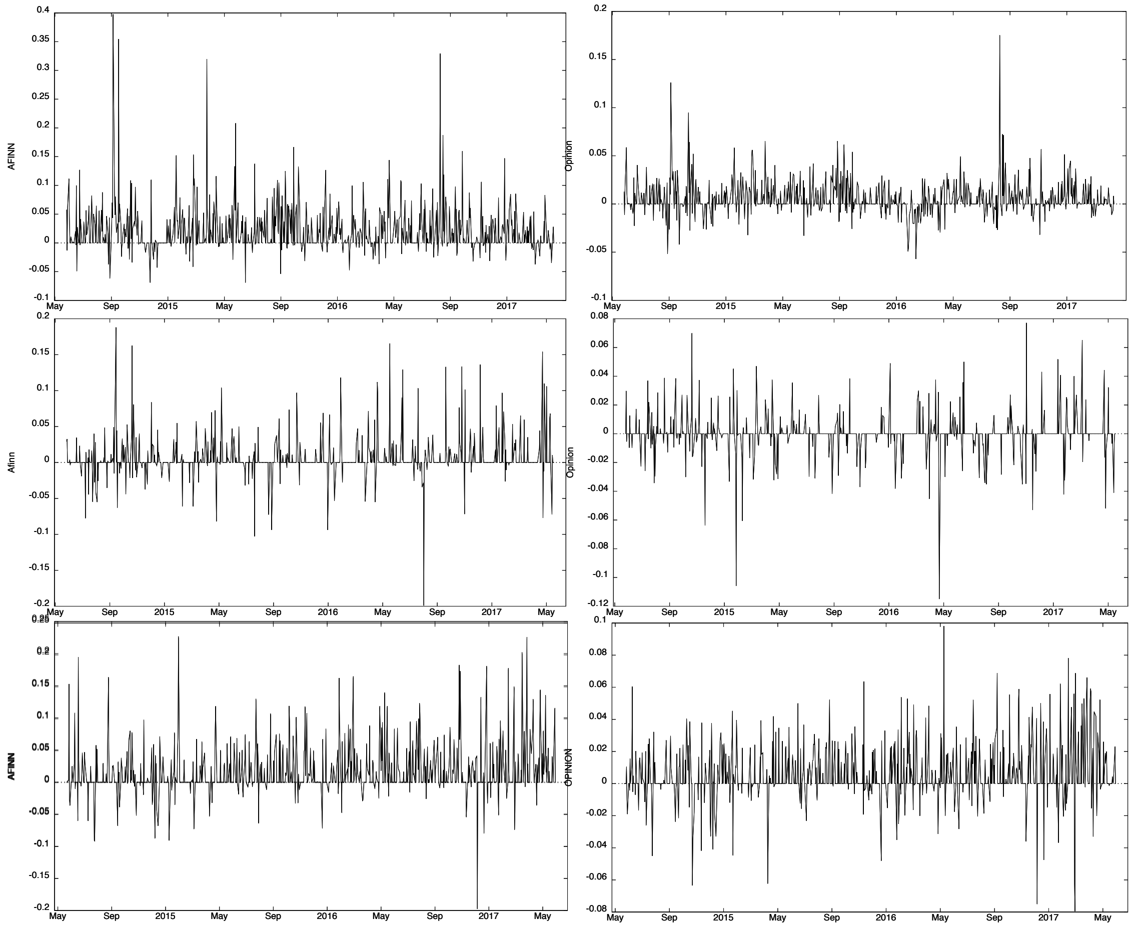
Same as for financial data, the news articles were collected on the daily basis covering the time horizon from May 2014 to May 2017. In contrast to financial stock data, having no values for non-trading days, the news releases are continious regarding the day of the week and holidays. Moreover, the weekend or holiday news may have a huge impact on the stock market, which we can not neglect. Taking this fact into account, we transferred the sentiment scores accumulated for non-trading days to the next nearest trading day. That is, the average news sentiment prevailing over weekend will be applied to Monday next week. The same logic holds for holidays. There could be several articles published per day, in that case we will work with an average sentiment of all articles published that day. The days when there were no articles released the value of zero will be assigned indicating no influence on the market.

Each Factiva aricle has a following format: headline (HD), word count (WC), publication date (PD), source name (SN), language (LA), company code (CO), industry code (IN), leading paragraph (LP), text of the article itself (TD) and others. Again, in order to avoid the duplicity of the information we analyse only the text from the article itself, not taking into account e.g the leading paragraph, which shows the summary of the article.

For obtaining the sentiments corresponding to certain article we employed three approaches: machine learning approach (Naive Bayes algorithm), dictionary-based approach and combination of them. All of those approaches are built on extracting the words from each article (text processing) and deciding on its polarity and strength. Then the inference for the whole article was made, thus deriving a sentiment score of the article. The combined approach means that the Naive Bayes algorithm is taking the vocabulary only from the predefined lexicon and then conduct its usual steps for deriving the sentiment scores. The detailed information about both methods as well as dictionary choices and Phyton scripts extracts are described in Chapter 3. Based on performance evaluation of pure and combined Naive Bayes approaches from the section 3.2.3, we will proceed only with the best performing ones defined for each stock. Therefore, we will further analyze how each approach for deriving the sentiment (multiclass lexicon, binary lexicon, hybrid Naive Bayes) contributes to volatility modelling.

The following graphs (Figure 5.7) represent the sentiment values time series plots for each stock for multiclass and binary lexicons. We did not provide a Naive Bayes sentiment plots since the sentiment values are discrete and the graph is very uninformative. The first row shows the sentiment values for Apple stock, with multiclass lexicon on the left and binary lexicon on the right. The second row reveals the information about Microsoft stock, again with multiclass lexicon being on the left and binary one being on the right. Analougsly, the last row designated for the Amazon stock. From the figure it is visible that density, values as well as distribution of the sentiments is very distinct for each stock. Moreover, the magnitude and sometimes even the direction is different for each lexicon within one stock.

Figure 5.7: Sentiment time-series plots



Source: Author's computations

Note: The first row-Apple Inc; second row-Microsoft Corp.; third row-Amazon.com. From left to right: multiclass dictionary sentiments and binary dictionary sentiments

Notably, an Apple stock news are at large positive, there are very few negative news and their maginude is not that huge. Interestingly, the highest values are reached around September, where the news about the launch of new iPhone were prevailing. As for comparison between dictionaries, the multiclass lexicon sentiments overall carry higher values than binary lexicon sentiments. For the

Microsoft stock the picture is different: amounts of positive and negatives news sentiments are more balanced. In fact, during the analyzed period there were a lot of bad news regarding tax evasion, lawsuit with Samsung, antitrust violation in China as well as several outages of storages and others. We can also see some outliers which do not appear on both plots, meaning that sometimes the dictionaries have predicted different scores. For the Amazon stock, again, we can observe that the amount of predicted positive sentiments by both dictionaries is significantly higher than the negative ones.

All the sentiment series are checked for the stationarity, since it is a necessary condition for GARCH modelling. For this purpose, the ADF and KPSS tests were employed. The summary statistics of the sentiment series obtained from the multiclass and binary lexicons for all three stocks is presented in Table 5.3

Table 5.3: Summary statistics of sentiments for all stocks

Apple Inc.	Obs.	mean	median	min.	max.	sd.	skewness	kurtosis
MultiCl lex.	750	0.022	0.002	-0.069	0.398	0.043	3.247	19.503
Binary lex.	750	0.006	0.000	-0.057	0.175	0.018	2.068	13.894
Microsoft								
MultiCl lex.	752	0.006	0.000	-0.199	0.188	0.029	1.331	12.179
Binary lex.	752	0.0001	0.000	-0.115	0.077	0.014	-0.811	13.336
Amazon								
MultiCl lex.	752	0.017	0.000	-0.198	0.227	0.040	1.439	5.759
Binary lex.	752	0.006	0.000	-0.079	0.098	0.018	0.480	3.964

Source: Author's computations.

The number of observations, provided in Table 5.3 corresponds to the days of available financial data, together with no sentiment days represented as zeros. The total number of raw news gathered, however, is different. There were 1348, 887 and 1368 news articles collected for Apple, Microsoft and Amazon stocks respectively. This is also observable from the sentiment time-series plot where Microsoft total amount of news is significantly lower than for Apple and Amazon.

Chapter 6

Model Estimation and Results

In this chapter the results of estimations using the in-sample and out-of-sample data are presented. First, the model is estimated without exogenous news variable and then we add the news variable to the variance equation. Moreover, the news sentiment variable is split for two vectors: the one containing the negative sentiments and another containing positive sentiments. We can thus separately distinguish their effects. For all the computations we used statistical softwares such as STATA and Gretl. We then comment on the results, comparing the variant with and without news variable and different news variables effects. We will describe the detailed procedure of fitting the mean and variance equation for the Apple stock. For the remaining stocks, the modelling approach is analogous, hence we will only provide estimation results and its' interpretations.

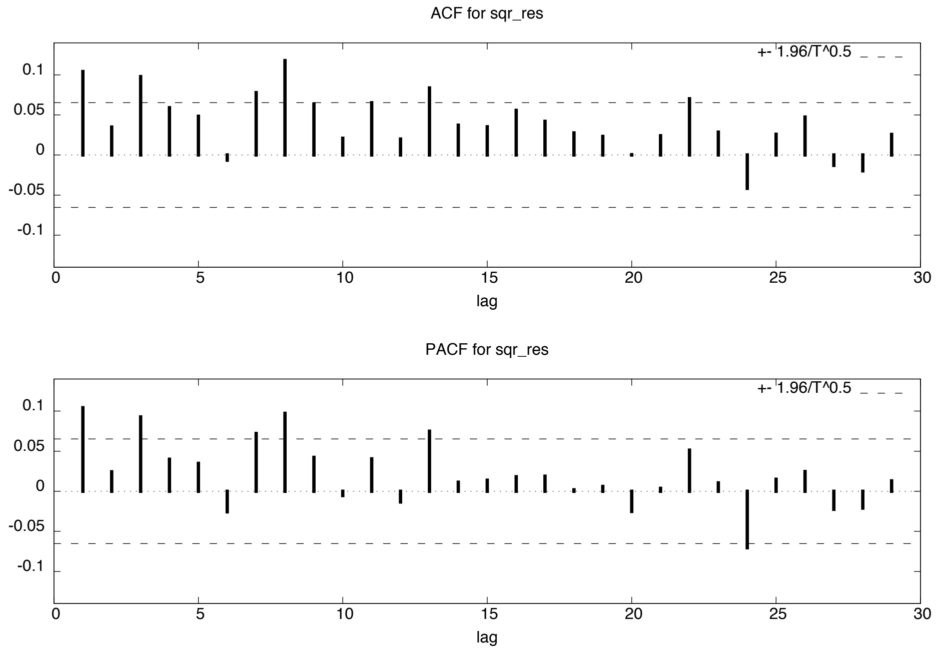
6.1 Apple Inc.

6.1.1 In-sample Model fitting without News Sentiment

After the time series has been stationarized by differencing we can apply the Box-Jenkins (1970) methodology for ARIMA model specification. It suggests Autocorrelation (ACF) and Partial Autocorrelation (PACF) plots of the differenced series to get information about dependencies and identify the numbers of AR and MA terms. In fitting ARIMA model, the idea of parsimony is important in which the model should have as small parameters as possible yet still be capable of explaining the series. The more parameters the greater noise that can be introduced into the model and hence standard deviation. Our ACF and PACF plots of the differenced Apple series reflect no significant lags. Hence, the original return series resemble a random walk model ARIMA(0,1,0). To confirm it is a good model to represent our series, we will perform a residuals diagnostics, where the residuals plot is showing no dependencies in the data. Also, applying Ljung-Box Q test, we do not reject the null hypothesis of no autocorrelation in the residuals with p-value of 0.8989. Such results suggest that the mean equation of our GARCH model will be containing only the constant term.

After having properly specified the mean equation, one need to identify the correct form of the variance equation. Lets have a look at squared residuals ACF and PACF (Figure 6.1). From ACF and PACF and from ARCH-LM test we can see, that there are further dependencies in the data, thus we will model them by allowing for heteroskedasticity (ARCH/GARCH models). The test results the p-value equal to zero for certain first lags. ARCH-LM test is equivalent to the F-statistic for testing the joint hypothesis in a linear regression of squared residuals on its lagged values.

Figure 6.1: ACF and PACF of squared residuals for Apple



Source: Author's computations.

We will fit the ARCH, GARCH until there is no dependencies left in residuals. GARCH (p, q) models are typically selected by Akaike information criterion

(AIC) and Bayesian information criterion (BIC). The difference between the BIC and AIC is that AIC imposes a greater penalty for the number of parameters rather than the BIC. Usually these criteria bring a conclusive results in terms of model preferences, however, in opposite case, Andserson & Burnham (2002) provided a theoretical arguments in favor of the AIC over the BIC. The mean and variance equations are estimated jointly by maximum likelihood (MLE), which is incorporated in used statistical softwares.

After estimating different GARCH models we came to conclusion that GARCH (1,1) is the most appropriate model for the selected time series. The model with the lowest AIC and BIC was selected. This is consistent with the most empirical studies involving the application of GARCH models in financial time series data. Additionally, we have assumed different distributions such as: normal (Gaussian), Student-t and Generalized.

Table 6.1: Plain GARCH estimation of Apple

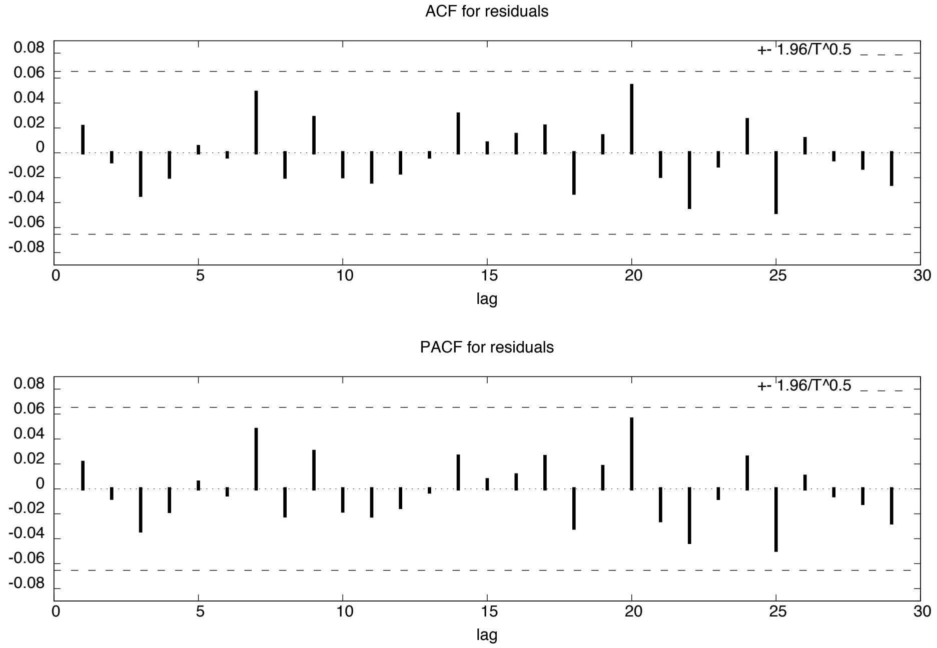
Variance equation			
Variables	Gaussian dist.	Student-t dist.	GED dist.
const	0.0000	0.0000	0.0000
	(0.0000)	(0.0000)	(0.0000)
L1. ARCH	0.1389***	0.2380***	0.1691**
	(0.0318)	(0.0932)	(0.0701)
L1.GARCH	0.8165***	0.6523**	0.7164**
	(0.2014)	(0.2826)	(0.3310)
LL	2115.346	2164.029	2163.528
AIC	-4222.692	-4318.057	-4317.057
BIC	-4204.217	-4294.963	-4293.963

Legend: *$p < 0.1$; **$p < 0.05$; ***$p < 0.01$; Standard errors in parentheses
Source: Author's computations.

Table 6.1 shows the estimation results of GARCH (1,1). The significance on the fitted GARCH terms is very high. ARCH effect captured by L1.ARCH is significant at 1% significance level with assumption of normal and student-t distributions and at 5% significance level with generalized distribution. Such variable represents the response of volatility to previous period shocks in return series. The variable representing the persistence of volatility (L1.GARCH) is also statistically significant at 1% significance level for normal distribution and at 5% significance level for student-t and generalized distributions. It is showing relatively high persistence with values around 0.8, 0.6 and 0.7 respectively, comparing to ARCH terms. As for the magnitude, yesterday's volatility leads to another period of high today's volatility. The distribution assumptions we choose effect the value of the estimates and standard errors, but do not influence the significance of the variables (at least 5% significance level). The sum of ARCH and GARCH coefficients is less than 1, satisfying the stationarity condition. For the normal distribution the sum of ARCH and GARCH terms is nearly one. In terms of explanatory power, AIC and BIC are quite similar for student-t and generalized distributions, while for normal distribution those criteria show worse fit.

From the Figure 6.2 you can see virtually no patterns in ACF or PACF left after GARCH(1,1). Additionally, ARCH-LM test does not reject the null hypothesis of no ARCH effects with p-value of 0.8154, suggesting no significant dependencies. Results of the test is presented in Figure A.6. It suggests that the chosen model fits well our data. However, we would like to note that after estimating GARCH(1,1) the residuals were still not completely normally distributed, even though it has improved a bit. This fact is in line with results in Table 6.1, where the best fit produced the models with student-t and generalized error distributions.

Figure 6.2: ACF and PACF of squared residuals after GARCH(1,1)



Source: Author's computations.

6.1.2 In-sample Model fitting with News Sentiment

This section discusses the estimation results of augmented GARCH models with news sentiments derived based on lexicon and machine learning approaches. In the lexicon derivation approach, the sentiments are obtained purely from the predefined dictionaries, where the selected words carry certain values. In our study, we have employed both multiclass (AFINN) and binary (Opinion Lexicon) one. We then compare the influence of these news sentiments on stock volatility and make a conclusion which one fits best our model and whether it can improve the forecasting abilities. In the machine learning approach the sentiments are calculated based on the hybrid Naive Bayes algorithm (with multiclass lexicon), which was chosen according to its performance in section 3.2.3.

Table 6.2 summarizes these results, appllying different distributional assumptions for robustness check. SP and SN are vectors of positive and negative sentiments respectively added to the GARCH(1,1) variance equation. As in the plain GARCH model estimation, the distributional assumptions change the values of the coefficient estimates, but doesn't influence its significance (at least 10% significance level). For all three model specifications the ARCH and GARCH terms remain significant and, similar to the plain GARCH model, where volatility persistence has much higher impact on volatility rather than the impact of the previous shocks.

Interestingly, the SP variable is statistically significant for all three models regardless the error distribution assumptions, while SN is statistically insignicant. The magnitude of the coefficient estimates of positive sentiment variable is much smaller compared to ARCH and GARCH effects. Moreover, we can observe that the sentiment coefficient estimates from the third (Naive Bayes) model are smaller than the ones from first and second models (lexicon based models), which can be explained by the size of sentiment values itself. The continious sentiments obtained from lexicons range from -1 to 1, while assigned discrete values based on Naive Bayes algorithm are ranging from -2 to 2.

Economically, our results suggest that for the Apple stock the positive news from time t-1 increase the volatility of this stock at time t. On the other hand, the negative news showed to have no impact on the stock volatility. While

Table 6.2: Apple: Augmented GARCH with News Sentiments

Multiclass lexicon	Gaussian dist.	Student-t dist.	GED dist.
const	0.0000	0.0000**	0.0000***
	(0.0000)	(0.0000)	(0.0000)
SP	0.0007***	0.0002*	0.0004**
	(0.0002)	(0.0001)	(0.0001)
SN	0.0000	0.0000	0.0000
	(0.0000)	(0.0000)	(0.0000)
L1. ARCH	0.1597***	0.2112***	0.1714***
	(0.0323)	(0.0756)	(0.0640)
L1.GARCH	0.6525***	0.6539***	0.6721** *
	(0.1385)	(0.2092)	(0.2298)
LL	2120.599	2169.758	2169.223
AIC	-4229.198	-4325.517	-4324.447
BIC	-4203.485	-4292.223	-4292.106
Binary lexicon			
const	0.0000***	0.0000	0.0000*
	(0.0000)	(0.0000)	(0.0000)
SP	0.0002**	0.0002**	0.0002**
	(0.0001)	(0.0001)	(0.0001)
SN	0.0000	0.0000	0.0000
	(0.0000)	(0.0000)	(0.0000)
L1. ARCH	0.0946***	0.2024***	0.1168**
	(0.0278)	(0.0706)	(0.0527)
L1.GARCH	0.7929***	0.7450***	0.6702***
	(0.0267)	(0.1542)	(0.0491)
LL	2130.706	2177.600	2174.256
AIC	-4254.948	-4330.083	-4331.405
BIC	-4236.473	-4306.989	-4308.312
Hybrid Naive Bayes			
const	0.00001***	0.00001***	0.00001***
	(0.00000)	(0.0000)	(0.00000)
SP	0.00002***	0.00001**	0.00001***
	(0.00000)	(0.00000)	(0.00000)
SN	0.00000	0.00000	0.00000
	(0.00000)	(0.00000)	(0.00000)
L1. ARCH	0.1132***	0.1460**	0.1168**
	(0.0205)	(0.0586)	(0.0527)
L1.GARCH	0.7645***	0.7735***	0.6752***
	(0.0173)	(0.0511)	(0.0491)
LL	2135.839	2176.702	2175.339
AIC	-4261.588	-4337.033	-4338.196
BIC	-4243.107	-4313.933	-4315.096

Legend: *$p < 0.1$; **$p < 0.05$; ***$p < 0.01$; Standard errors in parentheses; SP-vector of positive sentiments; SN-vector of negative sentiments.
Source: Author's computations.

the signs of the sentiment variables are as expected, the obtained magnitude values contradict several reviewed studies, where the negative sentiment has higher impact compared to positive one. However, several studies have also distinguished the magnitude of positiveness and negativneess, arguing that very positive and very negative news increase the volatility, while moderately positive and moderately negative do not have that pronounced effect. From the previous sections we know that for the Apple stock there were mainly positive news (with quite significant portion of very positive) prevailing during the analyzed period. This fact is in line with our results.

Looking at the AIC, BIC and LL criteria, we can see that the all augmented models provided a better in-sample fit compared to plain benchmark GARCH(1,1) within same distribution. The only exception is for the model with sentiment obtained from multiclass (AFINN) dictionary, where BIC criteria is higher favouring the plain GARCH model. The difference is however very small, compared to the size of increase in LL and size of decrease in AIC. We thus will not go for the more complex model and will rely on AIC and LL criteria in line with studies favouring AIC over BIC (Yang 2005). Importantly, we are interested in which sentiment derivation techniques was the most suitable for volatility modelling. Based on AIC and BIC we can conclude that the hybrid Naive Bayes approach is the best fitting our data within the same distribution. The second place takes the model with sentiments derived from the binary lexicon and the last one is the model with sentiments derived from the multiclass lexicon. Again, student-t and generalized distributional assumptions provided a better results for both AIC and BIC for each model.

6.1.3 Out-of-sample evaluation

We have got very significant fit for all models in-sample May 2014 - January 2017 in previous subsection. Looking at the residuals the ARCH effect was captured well for all of the specifications. We are however interested whether the model with the best in-sample fit will also produce the best out-of-sample volatility forecast. We thus constructed a one day ahead forecasts for each of the days in the out-of-sample period (February 2017 - May 2017). To assess the out-of sample model performance with will use two loss functions: RMSE and MAE, comparing to Garman-Klass volatility proxy as described in Chapter 4.4. For both loss functions a smaller value is preferred.

Table 6.3 summarizes the results for the the out-of sample estimations. Knowing the results from the in-sample estimations, the results from the out-of-sample are rather surprising and controversial. For the normal distribution, we can say that the only model that performed better than the benchmark plain GARCH according to both loss functions is the one augmented with news sentiments obtained from multiclass lexicon. For the remaining two models the resulting RMSE and MAE do not give a consistent results if the forecasting performance of augmented models is better than the plain one. For the student-t distribution are results are more transparent. According to both RMSE and MAE the models with sentiments from multiclass lexicon and hybrid Naive Bayes approach are visibly better than the simple GARCH(1,1). Moreover, the Naive Bayes model is slightly better than multiclass lexicon model, which is consistent with the in-sample results. For the generalized distribution, again the only model which is undoubtedly better than the plain model is the one with sentiments from multiclass lexicon. This also contradicts the results obtained from the in-sample fit.

Table 6.3: Apple Inc: RMSE and MAE for out-of-sample

Plain GARCH	Gaussian dist.	Student-t dist.	GED dist.
RMSE	0.00020384	0.00022139	0.00020481
MAE	0.00014441	0.00015449	0.00014132
Multiclass lexicon			
RMSE	0.00019233	0.00021562	0.00019832
MAE	0.00013931	0.00015201	0.00013977
Binary lexicon			
RMSE	0.00018684	0.00029830	0.00021154
MAE	0.00015573	0.00022734	0.00016267
Hybrid Naive Bayes			
RMSE	0.00020663	0.00021344	0.00020932
MAE	0.00014541	0.00014904	0.00015161

Source: Author's computations.

Even though the results might seem quite inconsistent, there are several conclusions we can formulate based on our estimations. First of all, it is not necessarily that the model with the best in-sample fit will produce the best out-of-sample forecast. Secondly, the the results for the t-distribution assumption are robust as for in-sample as well as for out-of sample, favouring the augmented GARCH model with sentiments obtained by hybrid Naive Bayes approach. Thirdly, the GARCH model with multiclass lexicon gives robust better results (compared to benchmark) across all distributions when considering both in-sample and out-of-sample. Fourthly, the model which utilizes the binary lexicon for news sentiment derivation, being a second best option for the in-sample, does not give a conclusive answer if it can improve the forecasting abilities of GARCH model.

6.2 Microsoft Corporation

In this section we will perform an analysis based on the data obtained for Microsoft Corporation. The steps are the same as it was described in the previous section, where the Apple company data was researched.

6.2.1 In-sample Model fitting without News Sentiment

Starting with a mean equation, again, it will comprise only the constant term as there were no significant dependencies of first order according to PACF and ACF and ARCH-LM test. Nevertheless, ARCH-LM test suggested autocorrelations of the second order for the squared residuals. We can thus specify the variance equation. Further, the estimation results of a plain best fitting GARCH(1,1) model with no additional external regressor are shown in Table 6.4.

From the Table 6.4 we can observe that ARCH effect, representing shock response, is highly significant for the assumption of normal distribution. However, for t-distribution the variable looses its significance and is only significant at the 10% significance level for the generalized distribution. On the other hand, GARCH terms, representing the persistence of the volatility is highly significant at the 1% significance level for all distributional specifications. The magnitude of those coefficients is high, but never exceeds the bound of 1, which follows GARCH model conditions. The values of LL, AIC and BIC are provided in the bottom part of the table.

Table 6.4: Plain GARCH estimation of Microsoft

Variance equation			
Variables	Gaussian dist.	Student-t dist.	GED dist.
const	0.00004***	0.00001	0.00002
	(0.00001)	(0.00001)	(0.00001)
L1. ARCH	0.2741***	0.1088	0.1406*
	(0.0839)	(0.0695)	(0.0761)
L1.GARCH	0.6031***	0.8617***	0.7676***
	(0.0627)	(0.0977)	(0.1148)
LL	2148.111	2251.309	2243.520
AIC	-4286.223	-4492.618	-4477.040
BIC	-4277.318	-4483.713	-4468.135

Legend: *$p < 0.1$; **$p < 0.05$; ***$p < 0.01$; Standard errors in parentheses
Source: Author's computations.

6.2.2 In-sample Model fitting with News Sentiment

The next step is to add the news sentiment variables into variance equation to capture the exogenous news effect. The results for three sets of sentiments are presented in the Table 6.5. The model behaviour for Microsoft data is distinct from what we have observed for Apple stock.

First of all, in contrast to Apple stock, we found an evidence of negative news sentiment influencing the stock volatility. More specifically, the model employing the binary lexicon for sentiment derivation as well as hybrid Naive Bayes approach provided robust results in terms of significance and direction of negative news coefficients. Moreover, it is interesting to note that the maginitude of the negative sentiment is significantly higher than the one of the positive sentiment. It holds for all cases, except the ones, where the opposite relation is offset by insignificance of the coefficients. This fact indicates that the negative news effect is more pronounced compared to the positive ones.

Secondly, the significance and the magnitude of the coefficients depend on the certain combination of sentiment derivation approach and distributional assumption. There is no generalized common result in favour of specific model specification within same distribution or within same sentiment derivation approach. Again, the coefficient estimates for the model incorporating the Naive

Table 6.5: Microsoft: Augmented GARCH with News Sentiments

Multiclass lexicon	Gaussian dist.	Student-t dist.	GED dist.
const	0.00003	0.00000	0.00000
	(0.00002)	(0.00000)	(0.00000)
SP	0.003926*	0.00228*	0.00285**
	(0.002249)	(0.00136)	(0.001307)
SN	0.001788	0.0004	0.002968*
	(0.001449)	(0.0005)	(0.001255)
L1. ARCH	0.1815***	0.1222	0.0668
	(0.0636)	(0.1090)	(0.0430)
L1.GARCH	0.4602**	0.7382***	0.7770***
	(0.2146)	(0.1706)	(0.0857)
LL	2199.258	2257.679	2255.498
AIC	-4386.519	-4497.359	-4488.997
BIC	-4375.832	-4481.329	-4469.410
Binary lexicon			
const	0.0000	0.0000	0.0000
	(0.0000)	(0.0000)	0.0000
SP	0.00105**	0.00089	0.00087
	(0.0005)	(0.00206)	(0.0018)
SN	0.00176**	0.00165**	0.001533**
	(0.000841)	(0.00078)	(0.000669)
L1. ARCH	0.1034*	0.0724*	0.0970
	(0.0548)	(0.0389)	(0.1388)
L1.GARCH	0.6921***	0.7908***	0.7428***
	(0.1015)	(0.0612)	(0.2479)
LL	2231.721	2262.221	2255.289
AIC	-4443.443	-4506.974	-4492.579
BIC	-4425.637	-4490.945	-4450.987
Hybrid Naive Bayes			
const	0.00005*	0.00001	0.000007
	0.00001	(0.000009)	(0.000006)
SP	0.000115*	0.00005	0.000116*
	(0.00006)	(0.00003)	0.00007
SN	0.00020**	0.00010**	0.00023**
	(0.00008)	(0.00005)	(0.00010)
L1. ARCH	0.2050***	0.1540***	0.0969**
	(0.0585)	(0.0503)	(0.0457)
L1.GARCH	0.4441***	0.7201***	0.7940***
	(0.1359)	(0.1047)	(0.0948)
LL	2229.429	2265.042	2269.764
AIC	-4446.858	-4516.084	-4521.529
BIC	-4419.122	-4483.725	-4479.924

Legend: *$p < 0.1$; **$p < 0.05$; ***$p < 0.01$; Standard errors in parentheses; SP-vector of positive sentiments; SN-vector of negative sentiments.
Source: Author's computations.

Bayes approach are lower by one decimal place, which can be explained by the size of the sentiment values itself.

As for explanatory power of the models, all augmented models show an improvement compared to plain GARCH model according to LL and AIC criterias: the log-likelihood is higher, while AIC is lower. However, the BIC criteria sometimes brings a contradictory results. The same issue we have encountered in case of the Apple stock. For the student-t distribution the model with multiclass lexicon is worse compared to plain model according to BIC and for the model with Naive Bayes approach BIC indicates no improvement. For the generalized distrubtion the model with the use of binary lexicon underperforms the plain GARCH, according to BIC criteria. We could not identify the best model within normal and student-t distributions, however, for the generalized error distribution the best model appeared to be the Naive Bayes one, with lowest AIC and BIC criteria. This outcome is robust for the in-sample instimation, knowing the in-sample results obtained from Apple stock analysis.

Economically it means that the Microsoft stock market is reacting to a larger extend or becoming more volatile, when the related bad news articles are published rather than the articles with a good news. In other words, investors perception of loss is more heterogeneous than perception of gain, investors are differently risk-averse.

6.2.3 Out-of-sample evaluation

Further, to robustly assess the models performance we will conduct an out-of-sample evaluation analogously to the Apple data analysis. The statistics for RMSE and MAE are presented in the Table 6.6.

According to Table 6.6, the only model that brings the consistent results is the model employing the multiclass lexicon: the metrics show an improvement compared to plain GARCH model, except for normal distribution, where RMSE is a little worse, but the difference is insignificant. This model outperformed the remaining two models, similarly to the example of the Apple stock. Other models are visibly underperforming based on either one or both metrics. This pattern is comparable to the one of the Apple stock, when the worst performing model in the in-sample (but still better than plain GARCH) performing the best in the out-of-sample providing a robust results.

Table 6.6: Microsoft: RMSE and MAE for out-of-sample

Plain GARCH	Gaussian dist.	Student-t dist.	GED dist.
RMSE	0.00026268	0.00027849	0.00021923
MAE	0.00015904	0.00018550	0.00014304
Multiclass lexicon			
RMSE	0.00026313	0.00025936	0.00020186
MAE	0.00015135	0.00018237	0.00013082
Binary lexicon			
RMSE	0.00031686	0.00032163	0.00031515
MAE	0.00016180	0.00019875	0.00014775
Hybrid Naive Bayes			
RMSE	0.00030726	0.00028506	0.00025778
MAE	0.00015888	0.00017369	0.00014441

Source: Author's computations.

6.3 Amazon

The last stock of the interest is an Amazon stock. We will follow the same procedure as it was shown before and will further try to generalize our results to derive an overall conclusion.

6.3.1 In-sample Model fitting without News Sentiment

Analogously, lets start with estimating a plain GARCH(1,1) model with no additional external regressor in the variance equation. Again, the mean equation will comprise only the constant term as there were no significant dependencies of first order. Nevertheless, ARCH-LM test suggested autocorrelations of the second order. We can thus specify the variance equation. The estimation results are shown in Table 6.7. ARCH and GARCH terms are significant for each specification, revealing very high volatility persistence. For the t-distribution, the sum of ARCH and GARCH terms are almost one, close to violating the stationarity condition.

Table 6.7: Plain GARCH estimation of Amazon

Variance equation			
Variables	Gaussian dist.	Student-t dist.	GED dist.
const	0.00003***	0.00000	0.00002
	(0.00001)	(0.00000)	(0.00001)
L1. ARCH	0.1352***	0.0400**	0.0888**
	(0.0420)	(0.0184)	(0.0426)
L1.GARCH	0.7852***	0.9162***	0.8277***
	(0.0407)	(0.0238)	(0.0782)
LL	1955.370	2076.565	2061.074
AIC	-3900.741	-4143.131	-4112.148
BIC	-3891.834	-4134.223	-4103.241

Legend: $*p < 0.1$; $**p < 0.05$; $***p < 0.01$; Standard errors in parentheses
Source: Author's computations.

6.3.2 In-sample Model fitting with News Sentiment

The next step is to add the news sentiment variables into variance equation. The results are presented in the Table 6.8. Starting with ARCH and GARCH terms, it is interesting to notice that for the model with the multiclass lexicon and for the model with the binary lexicon the ARCH effects are eliminated (are insignificant) with the inclusion of sentiment variables. This is consistent observation across all studies distributions. Also, overall volatility persistence for the augmented models is reduced, implying that a mean reversion process is quicker. On the other hand, GARCH effects are not eliminated meaning that news sentiment as an explanatory variable does not fully explain the observed heteroskedasticity in stock returns.

In terms of coefficients magnitude, when both SN and SP variables are significnat, the value of SN overall more than two times higher than the value of SP. Such outcome idicates a much higher effect of the negative news compared to the positive ones. As far as a direction of the sentiment is concerned, both SN and SP tend to increase stock volatily. In case of the opposite direction, the coefficient estimate is insignificant and we can ignore it. The significance as well as values of the coefficients depend on the combination of the sentiment derivation technique and error distribution assumption. For two out of three cases, the sentiment variable loses its significance, when considering t-distribution,

Table 6.8: Amazon: Augmented GARCH with News Sentiments

Multiclass lexicon	Gaussian dist.	Student-t dist.	GED dist.
const	0.00015***	0.00000	0.00015***
	(0.00002)	(0.00000)	(0.00005)
SP	0.00434***	-0.00004	0.00232**
	(0.00081)	0.00009	(0.00097)
SN	0.00898***	0.00028	0.00543*
	(0.00307)	(0.00029)	(0.00310)
L1. ARCH	0.2131	0.0334**	0.1617
	(0.1993)	(0.0165)	(0.1224)
L1.GARCH	0.6958**	0.9153***	0.6301**
	(0.1263)	(0.0202)	(0.2761)
LL	2005.762	2101.562	2086.427
AIC	-3999.524	-4149.125	-4158.855
BIC	-3971.725	-4136.691	-4146.366
Binary lexicon			
const	0.00013***	0.00014	0.00013
	(0.00004)	(0.00011)	(0.00009)
SP	0.00054***	-0.00004	-0.00025
	(0.00014)	(0.00045)	(0.00045)
SN	0.00129*	0.00097***	0.00072***
	(0.00078)	(0.000257)	(0.00024)
L1. ARCH	0.0322	0.0795	0.0521
	(0.0264)	(0.0570)	(0.0453)
L1.GARCH	0.6848***	0.6102**	0.6203**
	(0.1185)	(0.2701)	(0.2671)
LL	1996.640	2105.593	2085.704
AIC	-3981.280	-4197.186	-4159.409
BIC	-3953.480	-4164.753	-4131.609
Hybrid Naive Bayes			
const	0.00003***	(0.00003)*	(0.00003)**
	(0.000008)	(0.00002)	(0.00001)
SP	0.000104***	0.00003	0.00005*
	(0.00002)	(0.00002)	(0.00003)
SN	0.000258***	0.000129	0.000181**
	(0.00006)	(0.00008)	(0.00008)
L1. ARCH	0.1687***	0.1026**	0.1207***
	(0.0418)	(0.0493)	(0.0468)
L1.GARCH	0.6316***	0.7717***	0.6975***
	(0.0545)	(0.0975)	(0.0899)
LL	2023.283	2107.733	2096.972
AIC	-4034.567	-4145.466	-4179.944
BIC	-4006.767	-4141.032	-4147.510

Legend: *$p < 0.1$; **$p < 0.05$; ***$p < 0.01$; Standard errors in parentheses; SP-vector of positive sentiments; SN-vector of negative sentiments.
Source: Author's computations.

while ARCH and GARCH terms do remain significant.

When comparing the models in terms of explanatory power to the plain GARCH model, all three models across all distributions indicate an improvement based on the all three criteria: LL, AIC and BIC. In this case, unlike from Apple and Microsoft stock, AIC and BIC results are consistent. When comparing augmented models, again the best performing one is the model utilizing the hybrid Naive Bayes approach for sentiment derivation according to both AIC and BIC. Moreover, again the generalized distribution provided a better fit.

6.3.3 Out-of-sample evaluation

In this subsection we will further proceed with out-of sample evaluation for the remaining part of the dataset. The RMSE and MAE metrics values are presented in Table 6.9.

Table 6.9: Amazon: RMSE and MAE for out-of-sample

Plain GARCH	Gaussian dist.	Student-t dist.	GED dist.
RMSE	0.00039352	0.00039150	0.00036415
MAE	0.00033273	0.00034139	0.00030830
Multiclass lexicon			
RMSE	0.00027308	0.00033551	0.00025439
MAE	0.00021577	0.00027302	0.00019973
Binary lexicon			
RMSE	0.00030087	0.00032129	0.00026304
MAE	0.00025796	0.00027947	0.00021382
Hybrid Naive Bayes			
RMSE	0.00037114	0.00033176	0.00031068
MAE	0.00025616	0.00027360	0.00022646

Source: Author's computations.

According to the Table 6.9, same as for the in-sample sub-sample, all augmented GARCH models provided the lower values of both RMSE and MAE than the plain GARCH model, regardless the distributional assumption. Interestingly, again the multiclass lexicon model overall brought the best out-of-sample results, not being the preferred one in the in-sample. We could not derive the conclusive results for ranking of other models since the RMSE and MAE do not produce robust and stable outcome. It is important to note that the RMSE should always be larger or equal to the MAE. The bigger the difference between them, the bigger the variance in the individual errors in the sample. The lowest spread (difference) was however for the Apple stock, indicating that errors have less variant magnitude.

Chapter 7

Conclusion

Volatility modelling and forecasting has become a very hot topic during the last decades, since it is perceived as a measure of risk or uncertainty. Therefore, a substantial efforts have been put into developing models on volatility predictions, being able to anticipate the future market direction. The knowledge of the recent news can give an additional information for investors and thus effect their decisions influencing the stock market. However, the huge amount of digitalized information makes it costly to acquire, process and analyze any new piece of the information. This gave a rise to the development of automated solutions allowing to analyze the textual news content and derive a relevant features, that can be further incorporated into quantitative financial models.

In this thesis we analysed the role of published news articles about particular company in the stock return volatility modelling. More specifically, we were interested in news sentiment polarity and its magnitude. The ultimate goal of this paper was to compare how various sentiment-derivation techniques contribute to the volatility modelling and forecasting. For this reason, we compared the performance of several text classifiers, which were able to give us the information about articles content. In addition, we were also able to study the asymmetric effect of negative and positive news.

In the first part of the thesis we introduced the Naive Bayes classifier as a representative of the supervised machine learning approach for the news sentiment derivation and lexicon-based approach as a representative of linguistic approach. The Naive Bayes classifier was employed in three different ways. First, we performed the pure Naive Bayes algorithm, where all the words in the article were considered. Then, instead of taking into consideration the full text of the article, we took only the words contained in the pre-defined binary lexicon and then in the multiclass lexicon. This gave as three independent classifiers: pure Naive Bayes classifier, hybrid Naive Bayes with binary lexicon and hybrid Naive Bayes with multiclass lexicon. The supervised learning requires labeling of the training dataset, which was manually executed by the author. We have evaluated their performance based on accuracy and F-measure and found that overall all chosen classifiers performed similarly. Such outcome confirms an adequate choice of the text pre-processing steps, since the results are robust even on the reduced sample length. Nevertheless, the slight advantage across all selected stocks got the hybrid classifier with the multiclass lexicon. The news sentiment scores derived by this classifier we have further incorporated in the volatility modelling.

The second part of the thesis examined the effect of the news sentiment on the volatility modelling and forecasting using GARCH. More specifically, we compared three types of GARCH models augmented with news sentiment: first, with news sentiment derived with the help of multiclass dictionary, then, with the help of binary dictionary and finally with the help of the best performing hybrid Naive Bayes classifier defined in the first part of the thesis. The analysis was conducted on in-sample and the out-of-sample one-day-ahead forecast was built, employing normal, student-t and generalized error distribution assumptions.

The results suggest, that there is no one clearly preferred augmented GARCH model for both in-sample and out-of-sample. Overall, while all augmented GARCH models across all stocks indicated a better fit compared to plain GARCH model on the in-sample, we can not state the same about the out-of-sample. Moreover, the best performing in-sample model is the one utilizing the hybrid Naive Bayes approach for sentiment derivation. In most of the cases, the model employing the multiclass lexicon provided the worst in-sample fit among augmented models. On the other hand, the model utilizing a multiclass lexicon was the best performing for the out-of-sample, while other models did not bring a conclusive results.

We have also studied the asymmetric effect of negative and positive news in respect to stock volatility. Interestingly, for the Apple stock we found no evidence of negative news effecting the stock volatility, whereas positive news have a positive significant impact, meaning increasing the volatility. On the contrary, the two remaining stocks provided an evidence of both positive and negative news impacting the volatility, with the latter having a more pronounced effect. Moreover, the direction of the influence is the same: both positive and negative news do increase volatility. It indicates that the investors perception of loss is more heterogeneous than perception of gain, investors are differently risk-averse. Such outcome is in line with behavioral finance frameworks.

The main contribution of the study is a use of unique combination of classification algorithms, lexicons and selected stocks, which to our knowledge was not employed before. The manual labeling of the trained dataset for the supervised learning purposes can be considered as an original input. For the further research we would suggest to employ and compare also other classification algorithms, which could potentially improve the accuracy of the sentiment strength recognition. Moreover, the extended dataset could be used in order to provide a more robust results.

Bibliography

Andersen, T. (1996): "Return volatility and trading volume: an information flow interpretation of stochastic volatility." *Journal of Finance* **51**: pp. 169–204.

Andersen, T. G. & T. Bollerslev (1998): "Intraday activity patterns, macroeconomic announcements, and longer-run dependencies." *Journal of Finance* **53**: pp. 219–265.

Andserson, D. R. & K. P. Burnham (2002): "Avoiding Pitfalls When Using Information-Theoretic Methods." *Journal of Wildlife Management* **66**: pp. 6–910.

Bechara, A. & A. R. Damasio (2005): "The somatic market hypothesis: A neural theory of economic decision." *Games and Economic Behaviour* **52(2)**: pp. 336–372.

Berry, T. D. & K. M. Howe (1994): "Public Information Arrival." *Journal of Finance* **49**: pp. 1331–1346.

Bollerslev, T. (1986): "Generalized autoregressive conditional heteroscedastic." *Journal of Econometrics* **31**: pp. 307–327.

Borovkova, S. & D. Mahakena (2015): "News, Volatility and Jumps: the Case of Natural Gas Futures." *Quantitative Finance* **15**: pp. 42–1217.

Box, G., G. Jenkins, & G. Reinsel (1970): "Time Series Analysis: Forecasting and Control." *Prentice Hall* .

Chen, H. & E. Ghysels (2011): "News- Good or Bad- and Its Impact on Volatility Predictions over Multiple Horizons." *Review of Financial Studies* **24(1)**: pp. 46–81.

CHRISTIANI, N. & J. SHAVE-TAYLOR (2002): "An Introduction to Support Vector Machines and other kernel-based learning methods." *Cambridge Univ. Press 2002* .

CLARK, P. (1973): "A subordinated stochastic process model with finite variance for speculative prices." *Econometrica* **41**: pp. 135–156.

COUSIN, J.-G. & T. LAUNOIS (2006): "News Intesity and Conditional Volatility on the French Stock Market." *Finance* **27**: pp. 7–60.

CUMBY, R. & S. FIGLEWSKI (1993): "Forecasting Volatility and Correlations with EGARCH Models." *Journal of Derivatives* pp. 51–63.

DING, X., Y. ZHANG, T. LIU, & J. DUAN (2014): "Using structured events to predict stock price movement: An empirical investigation." *2014 Conference on Empirical Methods in Natural Language Processing (EMNLP-14)* pp. 1415–1425.

ENGLE, R. (1982): "Autoregresive conditional heteroscedasticity with estimates of the variance of the United Kingdom inflation." *Econometrica* **50(4)**: pp. 987–1007.

FAMA, E. (1970): "Efficient capital markets: a review of theory and empirical work." *The Journal of Finance* **25(2)**: pp. 383–417.

FERNÁNDEZ, M., H. SAIF, Y. HE, & H. ALANI (2014): "On stopwords, filtering and data sparsity for sentiment analysis of twitter." *LREC* .

FINNIE, G., K. B., & B. VANSTONE (2010): "Financial time series forecasting with machine learning techniques: A survey." *Paper presented at the European Symposium on Artificial Neural Networks: Computational and Machine Learning, Bruges, Belgium* .

GARMAN, M. B. & M. KLASS (1980): "On the Estimation of Security Price Volatilities from Historical Data." *The Journal of Business* **53**: pp. 67–78.

HO, K.-Y., Y. SHI, & Z. ZHANG (2013): "How Does News Sentiment Impact Asset Volatility? Evidence from Long Memory and Regime-Switching Approaches." *The North American Journal of Economics and Finance* **26**: pp. 436–56.

KALEV, P., W.-M. LIU, P. PHAM, & E. JARNECIC (2004): "Public Information Arrival and Volatility of Intraday Stock Returns." *Journal of banking and Finance* **28(6)**: pp. 1447–1467.

KARPOFF, J. M. (1987): "The Relationship Between Price Changes and Trading Volume: A Survey." *The Journal of Financial and Quantitative Analysis* **22(1)**: pp. 109–126.

LAAKKONEN, H. & M. LANNE (2009): "Asymmetric News Effects on Exchange Rate Volatility: Good vs. Bad News in Good vs. Bad Times." *Studies in Nonlinear Dynamics Econometrics* **14(1)**: pp. 1–38.

LAMOUREUX, C. G. & W. D. LASTRAPES (1990): "Heteroskedasticity in Stock Return Data: Volume versus GARCH Effects." *The Journal of Finance* **45(1)**: pp. 221–229.

LI, F. (2010): "The information content of forward-looking statements in corporate filings: a Naive Bayesian machine learning approach." *Journal of Accounting Research* **48**: pp. 1049–1102.

NG, A. & A. FU (2003): "Mining Frequent Episodes for Relating Financial Events and Stock Trends." *7th Pacific-Asia Conference on Advances in Knowledge Discovery and Data Mining, Springer-Verlag* **2637**: pp. 27–29.

PONMUTHURAMALINGAM, P. & T. DEVI (2010): "Effective Dimension Reduction Techniques for Text Documents." *IJCSNS International Journal of Computer Science and Network Security* **10(7)**.

POON, S.-H. & C. W. GRANGER (2003): "Forecating Volatility in Financial Markets: A Review." *Journal of Economic Literature* **41(2)**: pp. 478–539.

PRECHTER, R. & W. D. PARKER (2007): "The Financial/Economic Dichotomy in Social Behavioral Dynamics: The Socionomic Perspective." *The Journal Behavioral of Finance* **8(2)**: pp. 84–108.

RANGEL, J. (2011): "Macroeconomic News, Announcements, and Stock Market Jump Intensity Dynamics." *Journal of Banking and Finance* **35(5)**: pp. 1263–1276.

SALZBERG, S. (1997): "On Comparing Classifiers: Pitfalls to Avoid and a Recommended Approach." *Data Mining and Knowledge Discovery* **1:3**: pp. 317–327.

SCHUMAKER, R. & H. CHEN (2009): "Textual analysis of stock market prediction using breaking financial news: The azfin text system." *ACM Transactions on Information Systems (TOIS)* **27(2)**: p. 12.

SHARMA, J., M. MOUGOU, & R. KAMATH (1996): "Heteroscedasticity in stock market indicator return data: volume versus GARCH effects." *Applied Financial Economics* **6**: pp. 337–342.

SHARMA, J., M. MOUGOU, & R. KAMATH (2005): "Heteroskedasticity in the returns of the main world stock exchange indices: Volume versus GARCH effects." *Journal of International Financial Markets, Institutions and Money* **15(3)**: pp. 271–284.

STIGLITZ, J. & S. GROSSMAN (1980): "On the Impossibility of Informationally Efficient Markets." *The American Economic Review* **70(3)**: pp. 393–408.

TAUCHEN, G. & M. PITTS (1983): "The Price Variability Volume Relationship on Speculative Markets." *Econometrica* **5**: pp. 485–550.

TETLOCK, P. C. (2007): "Giving Content to Investor Sentiment: The Role of Media in the Stock Market." *The Journal of the American Finance Association* **62(3)**: pp. 1139–1168.

TOUMAZOU, C., B. PREMANODE, & J. VONPRASERT (2013): "Prediction of exchange rates using averaging intrinsic mode function and multiclass support vector regression." *Artificial Intelligence Research* **2(2)**.

TUSHAR, R. & S. SAKET (2012): "Analyzing stock market movements using twitter sentiment analysis." *In Proceedings of the 2012 International Conference on Advances in Social Networks Analysis and Mining* pp. 119–123.

VERONESI, P. (1999): "Stock Market Overreaction to Bad News in Good Times: A Rational Expectations Equilibrium Model." *The Review of Financial Studies* **12(5)**: pp. 975–1007.

WANG, W. & K.-Y. HO (2016): "Predicting Stock Price Movements with News Sentiment: An Artificial Neural Network Approach." *Springer International Publishing* p. 628.

WUTHRICH, B., V. CHO, S. LEUNG, D. PERMUNETILLEKE, K. SANKARAN, J. ZHANG, & W. LAM (1998): "Daily Stock Market Forecast from Textual

Web Data." *Proceedings of the IEEE International /Conference on Systems, Man, and Cybernetics, IEEE Computer Society Press* pp. 2720–2725.

Yang, Y. (2005): "Can the Strengths of AIC and BIC Be Shared?" *Biometrika* **92(4)**: pp. 937–950.

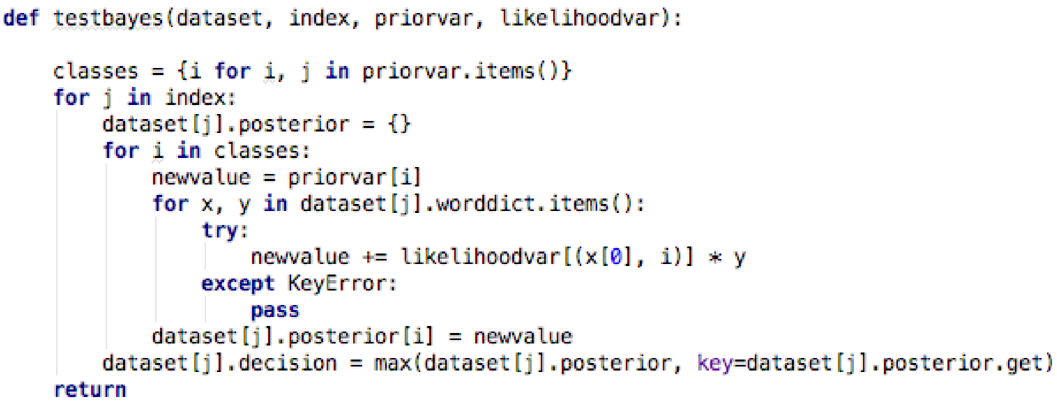
Yang, Y. & J. Pedersen (1997): "A comparative study on feature selection in text categorization." *International Conference on Machine Learning* .

Yulan, H. & D. Zhou (2010): "Self-training from labeled features for sentiment analysis." *ELSEVIER, Information Processing and Management* **47(4)**: pp. 606–616.

Appendix A

Appendix

Figure A.1: Source code: Likelihood Estimation for the test sample

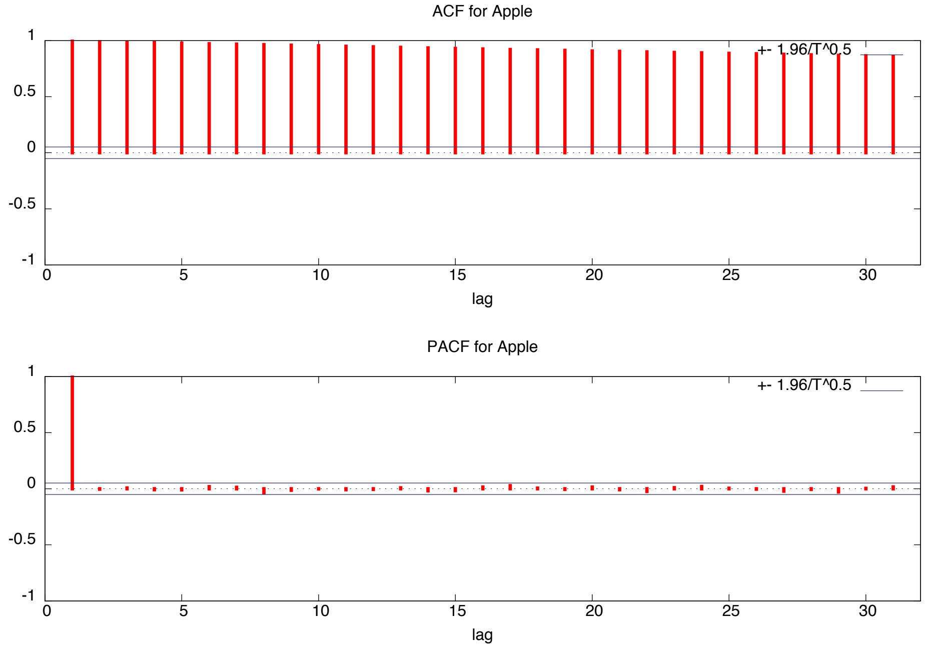


```
def testbayes(dataset, index, priorvar, likelihoodvar):

    classes = {i for i, j in priorvar.items()}
    for j in index:
        dataset[j].posterior = {}
        for i in classes:
            newvalue = priorvar[i]
            for x, y in dataset[j].worddict.items():
                try:
                    newvalue += likelihoodvar[(x[0], i)] * y
                except KeyError:
                    pass
            dataset[j].posterior[i] = newvalue
        dataset[j].decision = max(dataset[j].posterior, key=dataset[j].posterior.get)
    return
```

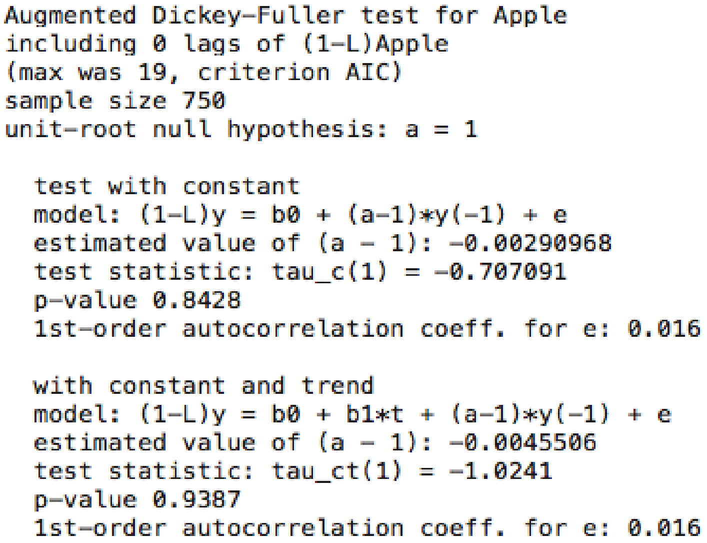
Source: Author's computations.

Figure A.2: ACF and PACF of Apple time series 2014-2017



Source: Author's computations.

Figure A.3: ADF test for Apple time series 2014-2017



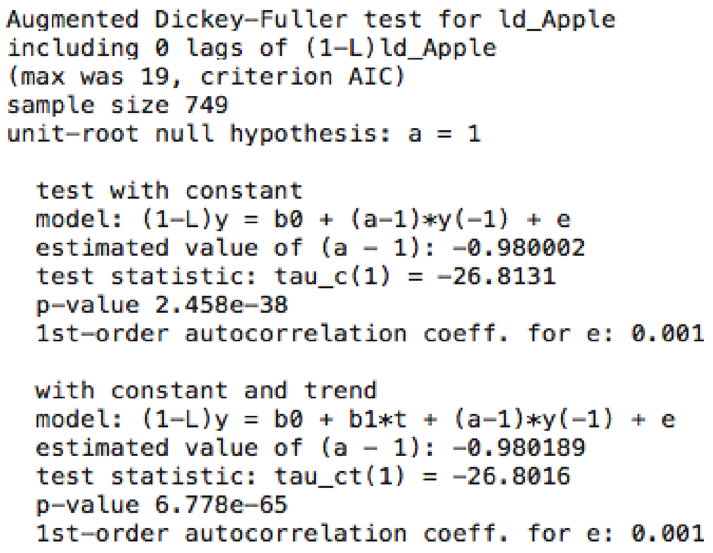
```
Augmented Dickey-Fuller test for Apple
including 0 lags of (1-L)Apple
(max was 19, criterion AIC)
sample size 750
unit-root null hypothesis: a = 1

  test with constant
  model: (1-L)y = b0 + (a-1)*y(-1) + e
  estimated value of (a - 1): -0.00290968
  test statistic: tau_c(1) = -0.707091
  p-value 0.8428
  1st-order autocorrelation coeff. for e: 0.016

  with constant and trend
  model: (1-L)y = b0 + b1*t + (a-1)*y(-1) + e
  estimated value of (a - 1): -0.0045506
  test statistic: tau_ct(1) = -1.0241
  p-value 0.9387
  1st-order autocorrelation coeff. for e: 0.016
```

Source: Author's computations.

Figure A.4: ADF test for ldApple time series 2014-2017



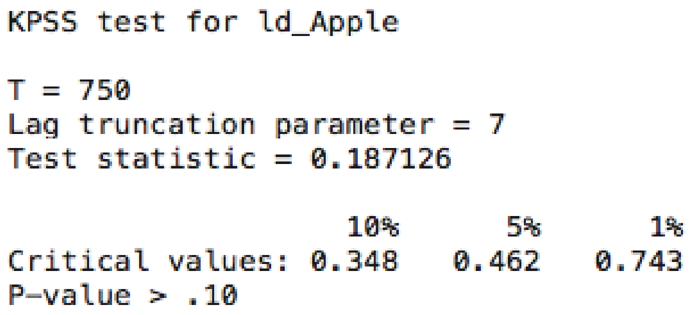
```
Augmented Dickey-Fuller test for ld_Apple
including 0 lags of (1-L)ld_Apple
(max was 19, criterion AIC)
sample size 749
unit-root null hypothesis: a = 1

  test with constant
  model: (1-L)y = b0 + (a-1)*y(-1) + e
  estimated value of (a - 1): -0.980002
  test statistic: tau_c(1) = -26.8131
  p-value 2.458e-38
  1st-order autocorrelation coeff. for e: 0.001

  with constant and trend
  model: (1-L)y = b0 + b1*t + (a-1)*y(-1) + e
  estimated value of (a - 1): -0.980189
  test statistic: tau_ct(1) = -26.8016
  p-value 6.778e-65
  1st-order autocorrelation coeff. for e: 0.001
```

Source: Author's computations.

Figure A.5: KPSS test for ldApple time series 2014-2017



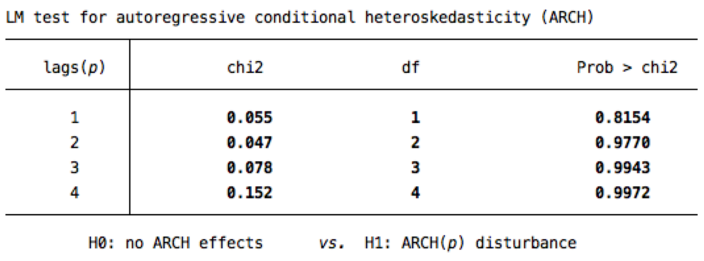
```
KPSS test for ld_Apple

T = 750
Lag truncation parameter = 7
Test statistic = 0.187126

                     10%      5%      1%
Critical values: 0.348   0.462   0.743
P-value > .10
```

Source: Author's computations.

Figure A.6: ARCH-LM test of squared residuals of Apple stock



LM test for autoregressive conditional heteroskedasticity (ARCH)

lags(*p*)	chi2	df	Prob > chi2
1	0.055	1	0.8154
2	0.047	2	0.9770
3	0.078	3	0.9943
4	0.152	4	0.9972

H0: no ARCH effects *vs.* H1: ARCH(*p*) disturbance

Source: Author's computations.