Identifying Expert Investors on Financial Microblog via Artificial Neural Networks

Pierluca Del Buono

A dissertation submitted in partial fulfilment of the requirements of
Dublin Institute of Technology for the degree of
M.Sc. in Computing (Data Analytics)

June 2018

Declaration

I certify that this dissertation which I now submit for examination for the award of MSc in Computing (Data Analytics), is entirely my own work and has not been taken from the work of others save and to the extent that such work has been cited and acknowledged within the text of my work.

This dissertation was prepared according to the regulations for postgraduate study of the Dublin Institute of Technology and has not been submitted in whole or part for an award in any other Institute or University.

The work reported on in this dissertation conforms to the principles and requirements of the Institutes guidelines for ethics in research.

Signed:

Date: 15/06/2018

Abstract

In the recent years, thanks to social media platform, a plethora of information has been available to financial investors, that were traditionally dependant from financial institutions advisors.

Strategies are now shared among web users, performances of stocks are commented in web communities and hints and suggestions are travelling on the internet with a fast pace, in a way that was unthinkable few years before. Several attempts have been made in the recent past, to predict Market movements and trends from activity of Financial Social Networks participants, and to evaluate if contributions from individuals with high level of expertise distinguish themselves from the rest of crowd.

The Present Work is leveraging 6 years of tweets extracted from the financial platform *StockTwits.com*, deep diving in its content, and proposing a predictive Neural Network algorithm of Multi-Layer Perceptron type, based on features derived from text, social network and sentiment analysis.

Users have been classified based on the performance achieved during the training, consistence of their prediction has been verified throughout the time and, finally, a trading strategy has been proposed based on following the top actors. The outcomes highlighted that expert investors are outperforming the wisdom of the crowd, and the trading schema put together generated a return of 38.6%, in 2015, when S&P500 had a slightly negative balance.

Keywords: Stock forecasting, Artificial Neural Networks, Social Network Analysis, Financial Microblogs, StockTwits, Online Social Networks

Acknowledgments

I would like to express my sincere thanks to Dr.Pierpaolo Dondio, my Supervisor, for providing the dataset, for his continuous assistance and for his great feedbacks.
I owe gratitude also to Dr.Luca Longo, for his suggestions on how to handle the whole dissertation process, and also for providing the latex template I used, that definitively made my life easier.
My appreciation goes also to Dr. Niccolò Casnici, for directing my efforts to a proper research question, and to Siobhan McNamara, for providing me insights on her thesis, that was a foundation for mine.
Thanks also to Daniel Voinea, for reviewing my algorithms, and suggesting me better suited alternatives; he proved to be a real subject matter expert in the domain.

Last but not least, I would like to thank Silvia for her support and her constant patience throughout the long journey of my MSc.

Contents

List of Figures

List of Tables

List of Acronyms

ANN	Artificial Neural Network
AWS	Amazon Web Services
CNN	Convolutional Neural Network
CSV	Comma Separated Value
EC2	Elastic Compute
EMH	Efficient Market Hypothesis
EMR	Elastic Map Reduce
JSON	JavaScript Object Notation
MLP	Multi Layer Perceptron
NASDAQ	National Association of Securities Dealers Automated Quotation
S3	Simple Storage Service
S&P500	Standard and Poor 500 stock Index
SNA	Social Network Analysis
SPY	Standard and Poor 500 ticker
STD	Standard Deviation

Chapter 1

Introduction

1.1 Background

Investors for long time had, as only focal point of contact, financial advisors, that were for them the only source to get value-relevant information regarding the stocks they were interested to buy. Throughout the time, anyway, a series of research had found out different weakness in the services offered by those professionals: limited coverage, stale data and, last but not least, bias and conflict of interest. (Bartov, Faurel, & Mohanram, 2017).

New sources of information dedicated to capital market's investors flourished in the past 15 years: online financial forums gradually became essential asset to share and exchange investment ideas or to discover relevant, timely and independent user generated comments, such as trading recommendations evaluated real-time by a large number of users.(Al Nasseri, Tucker, & de Cesare, 2015).

Amidst a great variety of platform, the focus of the present project will be on StockTwits.com microblog. StockTwits is a social network site where a peak of 40 thousands monthly users(December 2015) post and share information related to the financial market and equities. As in other popular blogging services, messages are constrained to 140 characters length and can include links, technical analysis and attachments of any sort. StockTwits contents are strictly related to trading space, making it an ideal source to study financial online communities in different market conditions. (Oliveira, Cortez, & Areal, 2013b).

1.2 Research Project/problem

As soon as financial social networks were reaching a sufficient number of adopter, interest sprung up across research world to pull out from them relevant measures, engineered to give investors a competitive edge in predicting market's directions (H. Mao, Counts, & Bollen, 2011).

H. Mao et al. (2011) also noted that those services introduced disruptive changes from previous investigation paradigms on financial markets, where attention for a particular equity was measured by the volume of trading operations involving it, and surveys monitored investors' mood to shed light on their reactions and expectations caused by economic news.

The increasing volume of data from online social platform, gave to scientists ways to model financial behaviour, and to extract statistically valid knowledge from it. Predictions of financial figures from activity on the most wide-spread micro-blogging platform (in first place *Twitter*) led anyway to mixed results due to the noise generated by the multitude of activities; this was particularly true when targeting measure such as closing prices and traded volume (Ruiz, Hristidis, Castillo, Gionis, & Jaimes, 2012), but, on the other hand, fairly decent results where achieved when volatility indicators were considered. (Liu, Qin, Li, & Wan, 2017).

1.2.1 Research Questions

While Wisdom of the Crowd (Surowiecki, 2004), meant as the aggregation of information contained in the big group of participants, is supporting the decisions of many

investors, more recent findings (Racca, Casarin, Squazzoni, & Dondio, 2016) points to the directions that best returns are obtained by following key members of community that prove to be expert investors.
In the present work will be defined *Expert Investors* those StockTwits users whose activity on the platform can learn a series of high-accurate predictive models, while *Wisdom of the Crowd*, on the other other hand, will be represented by a model learned by data produced by all the users.

On this ground, the following research questions will investigate whether *Expert Investors* are more effective than the *Wisdom of the Crowd* in forecasting the financial market:

1. Do expert investors, individuated by an ANN algorithm based on features derived from *SNA*, Sentiment and Text Analytics, outperform the Wisdom of the Crowd?

2. Is there a degree of correlation between the feature extracted from the *StockTwits* dataset and the markets' direction?

3. Do the expert investors achieve constant results over the time in predicting capital markets' direction?

1.3 Research Objectives

Starting from previous scientific literature achievement in the space of Financial Market predictions, the target of the present thesis is to classify social network users from the information contained in their tweets, processed by SNA, Text and Sentiment Mining techniques.
This differs from many works in the domain, where either those features where solely evaluated or the predictive algorithm was used just to learn Market behaviour. For instance, in Casnici, Dondio, Casarin, and Squazzoni (2015), users where assessed exclusively in terms of their Network relationships; while in the work of McNamara (2016), the results of the above mentioned techniques, was instead used at aggregate

daily level to predict volatility.

In this essay will be rejected, or failed to be rejected, the following null hypotheses.

1. $\mathbf{H_{01}}$: returns obtained from the predictions on 2015 generated by the algorithm trained for each of the top 100 users in the period between 2010 and 2014, are not statistically higher($p < 0.05$) than those coming from the random distribution of the predictions on 2015, generated by the algorithm trained for all users in the period between 2010 and 2014.

2. $\mathbf{H_{02}}$: a weak (> 0.1) degree of correlation is measured between any of Financial Micro-Blog SNA, Text-based or Sentiment specific features, and the return of S&P500 index in 5 days period.

3. $\mathbf{H_{03}}$: the median difference of a Wilcoxon Signed Rank Test for two paired sample, one related to the ranking of investor in period between 2010 and 2014, and the other related to ranking of investors in 2015, is different from zero.

1.4 Research Methodologies

Research Methodologies used in the present thesis will be Secondary by type, since data will be extract from financial sites such as *StockTwits.com* and *Yahoo! Finance*, Quantitative, carrying over measurement on those data, Empirical by form, since based on direct and measurable data, and Deductive by reasoning, moving from data analysis and modelling evidences.

1.5 Scope and Limitations

The dataset encompasses 6 years of tweets, and the present research will focus on predicting the market movement from social network activities. However, as highlighted by G. Wang et al. (2015), the period analysed has been considerably *bullish* and no extended *bearish* periods have been observed. Despite being the target binary variable well balanced, with both classes sufficiently represented, this can introduce bias in the final ANN model, and might introduce condition that can hamper the validity of the trading system proposed, making it not resilient when facing a financial crisis, likely to happen in the future.

Another limitation might come by not taking into considerations the *time-series* nature of the data, as done in similar studies (Ruiz et al., 2012); opinion of tweeters on the financial market, are naturally influenced by what happened previously, and, not taking this into account, might lead to problem of non-stationarity and biased results. However, since the effort has been directed into asserting expertise of end users, across a variety of securities, has been necessary to consider the time directional nature of the data just when dividing the dataset in training and test portions.

Is important also to mention that a great percentage of the data has been discarded, when coming to the actual model implementation. ANN are really sensitive to missing data (Chapman et al., 2000), so only data coming from a user taking part to the relation network has been retained, standing the impossibility to input the missing values. Moreover, when analysing users, only those with at least *380* rows for period of analysis could take part to the experiment, to have enough data to train.
Last but not least, the target label was always calculated evaluating the percentage returns of security mentioned by the user, against the percentage return of S&P500 index. To hunt for *alpha* member of financial community, however, would have had more sense to compare the stock mentioned with an index related to their specific stock category(for instance NASDAQ for AMZN, and so on).

1.6 Document Outline

The remainder of this dissertation is organised as follows:

- **Chapter 2 ("Literature Review")** is focused on making clarity on the past

research in security forecast from Social Network, first of all with the earliest and most generic attempts, and then deep diving in the usage of more recent and complex techniques.
Different efforts based on different features will be summarised, and a specific section will be dedicated to the results of Neural Network in stock market predictions.

- **Chapter 3 ("Design and methodology")** will explore the original dataset, and its preparation to make it a viable input for the research subject of this work. Big part of the chapter will clarify the design of the experiment and the processes executed to extract the features based on *Sentiment Mining*, *Text Analytics* and *Social Network analysis*. Eventually, the correlation testing and the predictive modelling bit will be delineated.

- **Chapter 4 ("Implementation and results")** will detail all the experiments described in the previous chapter. A first section will be dedicated to the tuning of algorithm, in relation to different optimisation techniques and different topologies,
In this section, charts will be provided and metrics related to the quality of the prediction will be presented.

- Critical analysis of results pulled out from the experiments is the main subject for **Chapter 5 ("Evaluation and analysis")**; along with a digest of the learning phases of the algorithm on different section of the dataset, analysis related to the Social Network behaviour are provided. A Trading model will be suggested and discussed as well; Limitations and problems faced during the experimental phases will also be listed here.

- **Chapter 6 ("Conclusion")** will provide a summary of the whole thesis, along with a clarification of the contribution to the existing body of knowledge, and a series of suggestions for further research in the same domain and, ultimately, on the same dataset.

Chapter 2

Review of existing literature

2.1 Introduction

Information spreading in capital markets was studied way before the general availability of internet media; the Efficient Market Hypothesis was introduced with the work by Fama (1970), according to whom, the market prices are function of all the informations available to the public.
For the EMH, that defines an immediate reflection of the widely known fact into prices, investors cannot create a profitable trading strategies from news already available to the crowd of participants of the market. (Fama, 1991)
Also the well-known *Random Walk Theory* (Malkiel & McCue, 1985) gives support to this vision: considering the actions of investors purely driven by rationale behaviour, and complete availability of market information, the conclusion is reached that stock market is unpredictable, and stock picking strategies are outperformed by buy and hold strategies, since any effort of selecting best stocks on the market, is purely driven by chance.

With time, critics came to the EMH model by Malkiel (2003), considering it an abstraction not picturing accurately the reality. Some exceptions have been found, in information in news not transmitted instantly and fully into Capital Markets, and decaying into noise in a short term. Value might then be available to investors, in

the form of textual facts in financial forum, bulletins and newspapers.(Tetlock, Saar-Tsechansky, & Macskassy, 2008)
In this direction, is important to mention the work from Bagnoli, Beneish, and Watts (1999), where unofficial news of potential good results, "whispers" as the authors refer to them in the original text, can actually foresee abnormal earnings. Qualitative information like those above-defined, lays in abundance in the internet world nowadays, while 20 years ago were purely exchanged by words of mouth. Online communities, that are formed and maintained spontaneously on the web, are generating material in real time, disseminating speculations about possible directions of the market, fundamental or technical analysis of the data and mentioning personal preferences.(Felton & Kim, 2002)

With such a massive and continuously growing volume of information and data, tentative of predicting Stock's return from investors' activities trace back to the '90, like in work of Brooks (1998), where volume of Market exchange is put in relation with market's raise; the increased adoption, however, of Social Networks brought to radical changes of paradigm, as it can be see from one of the first work in the space, from Wysocki (1998), where volume of messages on bulletin boards was put in relation with volume of transaction on stocks exchanges and with returns, finding some correlations between agitation before earnings disclosure and returns immediately after.

Throughout the time, three distinct techniques were used to analyse data gathered from financial social media, with the purpose of measuring correlation or influence in the investment market. They were, from the one used least recently onward:

a *Text Analytics*, meant as Web traffic measurement; it was used on different search engine, on news headlines and on social media, and indicators pulled out were tested with financial indicators.

b *Sentiment Mining* has been then widely used, to measure the mood of the public, as an anticipators of future markets movements, in this supporting or bypassing the theory of the "Wisdom of the Crowd", that acknowledges the fact that many can be smarter than few.(Surowiecki, 2004)

c *Social Network Analysis* ultimately has been leveraged to model social media as graphs, in first place to use network characteristics, and then to individuate crucial investors that are influencing the rest of the users, and the market in one specific direction.

Attributes generated by the above mentioned scientific analysis, have been correlated, and have predicted in the past, with different level of precision, peculiar characteristics of the stock markets, as:

a *Trading Volume*, really often related to Volatility and to the Volume of the activities on search engine and on micro-blogs.

b *Volatility* as a measure of risk: many works found out correlation between this metrics and the features generated by techniques listed above.

c *Price Return*, or, alternatively, market direction, expressed as bullish or bearish. This has been the main target variable for many works. but results have been mixed.

In the following sections, relevant literature on each technique will be detailed: it will be described how they changed throughout the time, then their findings, and ultimately their evolution towards detection of the most representative users, with the intention to exploit their forecasts.
Last part of the Chapter will also present an overview of the Machine learning techniques adopted, in terms of predictions.

2.2 Text Analytics

The usage of indicators related to web traffic activity, mined from text, has been the first method used, for its simplicity, and it met continuous popularity, even to present days, when it has been sometimes combined with other methodologies. In the work

of Jones (2006), evidences are present that social media not only reflect the market's status, but they are also influencing its behaviour. The author reviewed market performance and volatility pre and after the introduction of a firm's bulletin boards, the first type of financial social media on the internet, and found shift in volatility and price returns.
As already mentioned, first work to adopt such methodologies was the one from Wysocki (1998); opposite results, however, came by the work of Tumarkin and Whitelaw (2001), where user recommendations of different directions("strong buy,"buy","sell" or "strong sell") were counted on the financial forum *ragingbull.com*, but it wasn't possible to extract any valuable information to predict future market's behaviour.
Same study was carried out by Dewally (2003), who investigated messages on major newsgroups between 1999 and 2001, and it was confirming the EMH: while he found that the majority of the recommendations were positive, outnumbering the negative 7 by 1, he found also that investor were following a naive and unvoluntary momentum strategy, so advising to buy stocks that performed well few days ahead.

With time financial social media evolved, search engines became more widely available, and experimentation tried to correlate bigger amount of financial participants generated comments with financial results, focusing mostly to volume of activities along the time, as in the work of Wolfram (2010), where several NASDAQ stocks closing prices were put in relation with related volume of the tweets on *Twitter* micro-blog platform. The outcome was non-statistically relevant.
Also in the work from Dondio (2012), a big volume of raw messages on *finanzaonline.com*, spanning 8 years and related to the complete set of S&P500, is found to have some statistically significant predictive power of stocks prices, but with utterly small economic impact. This as confirmation of the results already obtained by Antweiler and Frank (2004), who found out that information/noise ratio in trading is extremely small.

A research that had wider domain was the one proposed by H. Mao et al. (2011), who compared financial surveys, the volume of *Google* search, the volume of message on *Twitter* along with one of the first sentiment indicators on stocks tweets, to prognosticate securities daily return. The results found interesting correlations between web traffic metrics on *Google* and on *Twitter*, in terms of volume of messages, and with the logarithm of price returns. The financial surveys lagged behind the internet social media, and thus were not containing any indicator.

Similar outcome were achieved by the work of Y. Mao, Wei, and Wang (2013), where posting volumes on *Twitter*, and their spikes, were used as an index of sudden interest, to foresee S&P500 most quoted stock price changes, and to attribute a cause of those variations. In the same article was proposed a trading strategies based on spike detection that outperformed the hold and buy on S&P500.
In particular, the evidence found from Y. Mao et al. (2013) showed that good results of the trading strategy were achieved just in case of increased activities caused by earning reports, and, for those cases, the Bayesian classifier they developed, was able to return a 15% gain over 55 trading days.

Also notable in the domain, is the study to predict market falls from increased search volumes on search engines by Curme, Preis, Stanley, and Moat (2014), in which increased search on *Google* or *Wikipedia* were found to anticipate market falls. However, it was not possible to find any relation with any topic.
More interesting out-turn is found when text-mining is combined with a more recent technique of Sentiment Analysis on social platforms; an example is found in the work of Oliveira et al. (2013b), where the two approaches have been combined: no evidence have been found that those metrics can act as market movement indicators, but the attributes related to posting volumes, were able to explain good part of Stock volatilities.

2.3 Sentiment Analysis

From 2011 onward, Analysis of Sentiments rose as predominant investigation technique, due to technical improvements, as shown in the notable work of Bollen, Mao, and Zeng (2011). They concluded that there was, 87% of the time, a correlation between positive and negative sentiment and upward/downward trend of Dow Jones Index.

This approach has been performed in many researches, leading anyway to mixed result not so convincing like in Bollen et al. (2011): for instance, Sprenger, Tumasjan, Sandner, and Welpe (2014), using only message pulled down from *Twitter*, measured a positive correlation just between pairwise combinations of abnormal returns, volume of trading on S&P500 and public sentiment conveyed by tweets; this exclusively in case of a buy signal, and in conjunction with the level of agreement.
The authors, while finding dedicated financial services and bulletin boards a precious resource to analyse mood around stocks exchanges, found hard to clean signals from the noise, and assessed a difficulty in studying how information reflected in tweets are incorporated into the markets; however, in the paper is introduced one of the first scheme to exploit signals buried in stocks microblogs.
Bartov et al. (2017) aggregated opinions mined in 1 milion of tweets from *Twitter* microblog, collected in 4 years period, in proximity of earnings disclosure. They found a positive correlation between the sentiment indicators they designed, with the reaction of stocks on the verge of the quarterly earning announcements; a linear regression algorithm was also proposed to predict the "jump" of the stock quotations.

Classification of tweets message, also coming from *StockTwits* platform, is foundation of the work of Bar-Haim, Dinur, Feldman, Fresko, and Goldstein (2011), where a classifier purely dictionary based, is compared to a unsupervised learning model, based on binary features indicating whether some elements are present or not in the tweet structure.

Another example of work on *StockTwits*, using *sentiment analysis*, is the research of J. Watts, Anderson, Asbill, and Mehr (2017) who developed an algorithm capable of learning the sentiment(*Bullish* or *Bearish*) from the tweets on the site already tagged, and using the opinion aggregation technique proposed in Du, Hong, Wang, Wang, and Fan (2017), were able to put together a trading machine, that consisted in the development of a portfolio, risk immune, according to them.

The constant improvement in sentiment classifier is paving the way to better results, with machine learning models close to compete with those purely dictionary based, as shown by Dridi, Atzeni, and Recupero (2018) in their paper, where a training model boosted by semantics achieved an accuracy score of score of 72%.
In the most recent work of Sohangir, Petty, and Wang (2018), different sentiment analysis methods on *StockTwits* data are compared, and lexicon-based methods still outperformed a machine learning approach. Among those, VADER (Gilbert, 2014) had the lower number of neutral messages, and Textblob (Loria et al., 2014) was the the worst one of the lexicon based; however, that Python library based on WordNet (Miller, 1995), was used with success in other Financial micro-blog papers, as in the work of Huang (2016).

2.4 Social Networks in Finance

Since the introductions of Social Networks, researchers debated over the "Wisdom of the Crowd" phenomena. Does it exists? How is it leading and shaping the development of user generated content and information discovery in the modern age? How is possible to capture it and extract valuable information from it? (Adamic, Zhang, Bakshy, & Ackerman, 2008).
Abilities of social communities to gather, maintain and curate knowledge repositories have been subject of different studies, to better understand what motivates contributors, and how relationship among them are modelled and driven. (G. Wang, Gill, Mohanlal, Zheng, & Zhao, 2013)

To better model interaction among all the actors, most recent approaches were based on building a graph on the social network, to better study how users relate to each other, and to weight them according to the trust a member has gained in the community; according to Sprenger et al. (2014), this trust is quantified in some way across the network: for instance in its article, users who are providing advice above the average, are retweeted more often, and have greater number of followers.

In all the studies involving *Social Network Analysis*, the measures extracted with the support of mathematical Graph theory played a role of pivotal importance , where a node(also called a vertex) is the representation of a principal actor of the social network, in most of the case an end user, but not always, and an edge is meant as the action of sharing an information, or a common trait.
In this sense some characteristics of a Financial Social Networks are better suited than other to represent connections: for instance, the study from Cha, Haddadi, Benevenuto, and Gummadi (2010), on *Twitter*, built for the first time in literature a graph from Twitter data, and measured correlation between user's influence and number of citations, in-degree connections and re-tweets. The latest measure was found to have better correlation, while the in-degree connections a user, modelled as a vertex, had, seemed to play no whatsoever role. The study was also deep-diving into topics that were subjects of actors' tweets, and realised that most influential users were focusing on limited number of those.

On same Social Network, Ruiz et al. (2012) expanded the study of Cha et al. (2010), daily measuring not only traffic indicators such as citation retweets, number of tweets, number of followers and citation, but building graphs on user's interaction, and collecting indicators on network daily level. The indicator collected, such as number of degrees, number of edges, PageRank and maximum distances between nodes, showed little or no correlation with market's price, but some with Trading Volumes.
With this findings, however, Ruiz et al. (2012) proposed some trading strategies, and one of those outperformed baseline strategies, that were based on Dow Jones Index,

that was negative for the period of observation, between 1st of March 2010 and 30th of June same year.

A complex and detailed study of how Network metrics evolve during the time, and how different Capital market's conditions shape or are shaped, to some degree, by them, it's present in the work of Casnici et al. (2015); wherein the authors looked into 7 years of messages, around 800 thousands distinct ones, on the forum Finanzaonline.com, targeted to Italian investors, to measure metrics related to the graph structure such as modularity and number of nodes and edges, and proposed an indicator of Network stability, quantified in investors' turnovers and ties refreshing.
Casnici et al. (2015) described also the phenomena of joint-attention efforts put in place by participants, when particular Market's event are catalysing their reaction, pushing their activity to coalesce during the time. The role of market volatility was of paramount importance, modelling more complex relationship between users in period of turbulence, and causing fragmentation.

Social Network Analysis techniques extended outside the mere border of Social Media, having end users as node: in the work of Sankar, Vidyaraj, and Kumar (2015) it's present a critic on the selection criteria of expert investors on Social Media, and a bipartite graph between investment fund and stocks was built, and stock recommendation system was engineered from the trust Mutual Funds management was laying in specific securities. Network metrics such as centrality, eigenvector centrality are built, and a portfolio is built from stock recommendations, capable to outperform Indian Nifty 50 stock index in 2014.
In a similar way Roy and Sarkar (2011) used social network representation to model relation and dependencies between stocks and market indices, building a bipartite graph over stock returns in a period of 120 weeks, and using a Minimum Spanning Tree to reduce complexity. In conclusion, they realised that their model was capturing significantly economic downturn caused by events such as Lehman Brothers' failure, increasing correlations in such time. Clustering measure reported building of evident

aggregation at regional level, and measure of centrality revealed crucial roles for the European indices, making them the most influential.

2.4.1 Expert Investors identification

From the beginning of the studies of users' interaction on social network, it was clear how a small portion of them led influence and redirected community's opinion, that for many participant is sufficient to support decision. Following the actions of expert actors able to inspire trust, for this reason, has became a common pattern in field studies, even before the existence of financial social media. (Dondio, Barrett, Weber, & Seigneur, 2006)

When Online communities became a thing, complex models were proposed on how to model their build-up, with particular focus on trust inception, that represented since then a key factor. (Dondio & Longo, 2011).

Bakshy, Hofman, Mason, and Watts (2011) did a first systematical analysis of the role of trust in a microblog as *Twitter*; not specifically focusing on Financial content, they browsed 1.6 Millions of messages, and they found out a crucial role in influence in their dimension and in some particular element contained in them(such as URL).

Also in the study of Weng, Lim, Jiang, and He (2010), the most influential users are detected, following a Social Network Analysis approach, deepening the role of assortativity in connection, and proposing and algorithm able to replace PageRank in detecting most probable path between users. Homophily, the tendency of similar user to attach to each other in the long term, had a fundamental role in the above mentioned paper.

Following studies in similar domain, in Financial Social Network the attention shifted to the problem of identifying most influential users, and models based on Social Network analysis, sometimes mixed with topic analysis, have been proposed, like in Bar-Haim et al. (2011), where *StockTwits* social network has been studied, and a framework to categorise the users has been proposed, in particular:" a user is an expert if a high

percentage of his or her bullish tweets is followed by a stock rise".
This framework was also used for learning the market, and compared to other methods.

In the paper from Casnici et al. (2015) actors, on an Italian Financial forum are classified as experts according to their:

- Communication activity.
- Presence longevity.
- Communication regularity.
- Pertinence of the content.
- Influence(as centrality degree).

In the same paper are found evidences on how their action can help forecasting daily stock returns, and how they were central hub in moment of high market volatility, while playing a less important role after bad news. In that occasion, the authors say, the informative content of message can drop.
Expert actors, however, communicated more than rest of the crowd in periods following volatility or after news that caused market's shocks.

A more recent work that was notable in classifying the quality of the intervention on a economical social media, was the one from Racca et al. (2016). The Researchers examined more than 10 million of activities in 7 years period on the already mentioned finanzaonline.com, financial forum in Italian language, putting together a categorisation of the site users, dividing them in expert and not-expert investors. In the article is covered the rise and the development of the financial crisis, with the support of indicators related to posting's activity and regularity of the online partecipation.
A key finding in this work was related to the different behaviour expert users had to market's uncertainty shock, giving strength to the importance of following advice of more skilled and leading investors.

2.5 State of the Art

At the date it's evident to researchers that, when dealing with financial micro-blogs, value at stake in classifying influential users, and then, expert investors is high: Studies leveraging combination of techniques arose in the recent years.
For instance, in the paper of G. Wang et al. (2015), sentiment and text analysis on *StockTwits* and on *seekingalpha.com* were the core techniques performed on user granularity, with a SVM machine learning method initially, and a dictionary-based method then. As conclusion, it was found that a subset of expert users had more predictive powers than the rest of the crowd. Those users were selected by the number of their interactions, and a trading schema developed on their tweets, outperformed the markets, meant as the S&P500.

In the work of T. Wang et al. (2017), 4 years of *StockTwits* data, along with 9 years of *seekingalpha.com* data were taken in observation, and top users were select with a mixture of text analytics (based on keywords), Machine-Learning Classifiers and counting the interactions with the rest of the users(not proper SNA). Though a predictor was not proposed, for some investors, deemed as experts, the correlation between all the features analysed and marker returns reached 0.4, whereas really low correlation was found for the mass of users. This was generating prediction of the market directions with 0.75 accuracy.

2.5.1 Learning Market from Social Networks

Once assessed correlations between metrics extracted from Social Network and Volume and Volatility, to a greater extent, or Prices , to a lesser extent, interests switched to predictions.
In the already quoted research from Bar-Haim et al. (2011), *StockTwits* social network has been studied, and two models were proposed to distinguish expert users from the rest: an unsupervised learning model, who performed better, and a SVM classifier; top

level investors in this study were defined by the precision of their forecast on stocks' movement.

Multiple forecasting algorithms were leveraged, but with mixed results also in the work of Oliveira, Cortez, and Areal (2013a), where a multilinear regression was used on StockTwits data. As anticipated, outcome showed no statistically significant results for predicting price of 5 major stocks; however good predicting power was found between trading volumes and posting volumes, and, in second place, between the users' activity and volatility.
More Recently, McNamara (2016) in her thesis has built regression trees based on network and text-based features and on messages' topics, to predict market volatility in different condition. Sticking always to Volatiliy, in the work of Dimpfl and Jank (2016) is present a series of regressive model to predict capital market's volatility out of Google search queries, rather than on social networks.

Recent involvements have seen exploiting the advantages of Artificial Neural Networks because of some interesting characteristics (Di Persio & Honchar, 2016), such as:

- Speed of Classification.
- Ability to deal with discrete, binary and continuous attribute.
- Ability to deal with high number of attributes.
- Tolerance to interdependent attributes.
- Absence of Prerequisites and Constraints.

This advantages are coming with some known weakness as uninterpretable models(no real knowledge transparency) and lack of resilience to missing values and to noise. (Chapman et al., 2000)

2.5.2 Gaps in Research

Usage of Neural Network became more widespread in the recent past, thanks of more efficiency and simplicity to use its libraries, as in the study of (Liu et al., 2017), where a Recurrent Neural Network was used to predict market volatility from finance related tweets sentiments.

A notable usage of Neural Network, in its flavour of Deep Learning, to differentiate users on StockTwits data is the one in the paper of Sohangir and Wang (2018), where expert investors, defined as people that could guess the direction of the market at least 15 times in 2015, are classified training ANN algorithms on 2015 data and testing them on the first semester of 2016.
Two different ANN were proposed: a Convolutional Neural Network based on the word composing the tweets, and a doc2vec ANN algorithm, as architected by Le and Mikolov (2014); results were compared against a baseline of a logistic regression, based on a bag-of-words built on the tweets texts. Outcomes favoured the CNN, the only model to overcome the baseline, that achieved an overall accuracy of 0.92, after being trained for 18000 epochs.

In this new achievements however, *SNA* features have never been inputed to a Neural Network to discern expert investors, and also in past literature, they have been rarely mixed with other features coming from Sentiment and Text Analytics.

Chapter 3

Experiment design and methodology

In this chapter the methods used for the experiment execution will be detailed:

1. Initially is described the process of extracting the data, transforming them in a suitable tabular format, and loading them in a relation database, to ease up the first *Exploratory Data Analysis*, and the sub-sequential transformation to produce a final dataset able to feed the algorithms.

2. In the next section, the core techniques that will be leveraged to enrich the data will be narrated. First of all *Text Analytics* and *Sentiment Mining*, to include domain spefic indicators. Then will be presented the two graphs, based users interactions with each others and on interactions between users and stocks, that will be the core of the *SNA* part.

3. The final dataset will be presented, with regards of the data enrichment phase that will include pre-calculated variables, including the calculated target labels, based on the difference between the selected stocks and the S&P500.

4. A section on the correlation analysis, introduced with the purpose of exploring the different predictive capabilities in the features involved, without dropping

any of them, standing the great flexibility of ANN models.

5. A brief overview of the model definition will be then presented, focusing on why a Dense Multi-Layer Perceptron will serve the present case, and what characteristic have been tuned to the final form, and which have been given as granted.

The workflow of the tasks that were carried out is detailed in the figure 3.1:

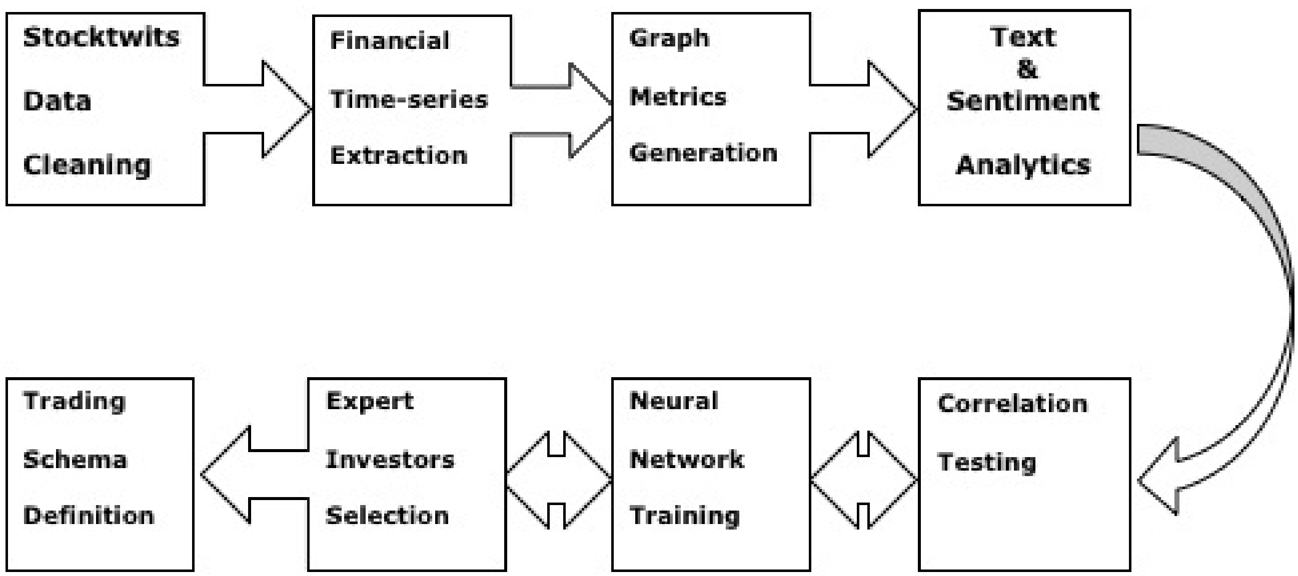


Figure 3.1: Project Workflow

3.1 Source Data

3.1.1 Tweet Data

The dataset was provided by *StockTwits.com*, a financial microblog, and consists of a collection of 6 years of tweets, from 1st of January 2010 to 31st of December 2015, contained in around 37 millions of records in a series of JSON files for a physical dimension of 60 GB. Those records contain tag for the *stock* ticker, the *userName*, the whole body of the message, and information such as link to external URL, pictures or files.

In figure 3.2, the increased adoption of the platform, showing a continuous raise in the number of tweets, unique users and distinct stocks mentioned, throughout the time.

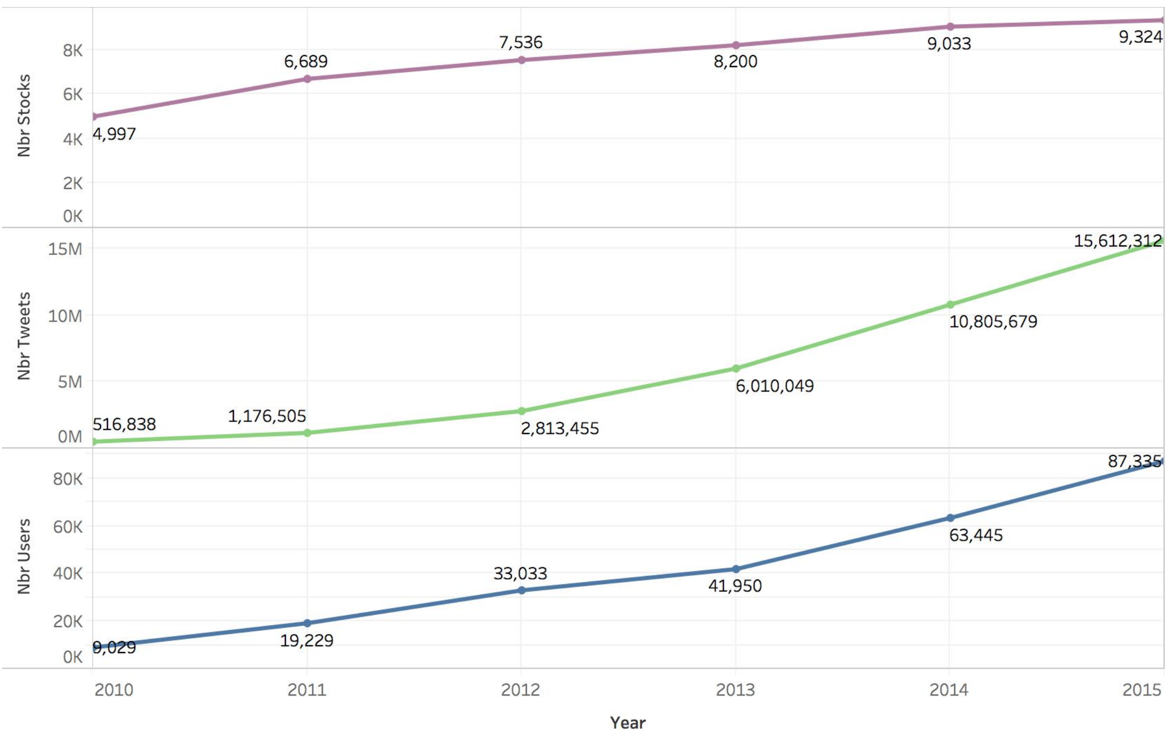


Figure 3.2: StockTwits Adoption

While the number of user was duplicating constantly, year after year, and same can be said about the number of tweets, in the number of stock of interest is observed a less steep growth, meaning that investors prefer to put emphasis on a limited portion of them.

The data was originally available in high nested format, with most of the fields collected into arrays, and the files have been zipped and loaded into a S3 bucket in a dedicated AWS account, from there a Hive staging table was created on top of them, inferring the general structure, and fields were flattened to achieve the tabulated version in table 3.1. Next steps included Extraction, Transformation and Loading of the data to a form that was easier to explore, analyse and enrich.

Table 3.1: StockTwits.com fields descriptor.

Field	**Data Type**	**Description**
Id	Int	Unique id for each tweet
body	Varchar	Tweet body of message
Actorid	Int	Unique id associated with each user
objectType	Varchar	category: person or automated firehose posting
displayName	Varchar	Tweeter name
preferredUsername	Varchar	Tweeter Username
followersCount	Int	Number of followers
followingCount	Int	Number of users he/she is following
statusesCount	Int	Number of tweets so far
summary	Varchar	Self-assigned investement profile
links	Varchar	links related to user's profile
image	Varchar	Link to a chosen profile picture
tradingStrategy	Varchar	Self-tagged description of trading usage
approach	Varchar	Self-assigned category of trading decision mechanisms
experience	Varchar	Self-tagged expertise level
id	Varchar	Id associated with tweet type
objectType	Varchar	Text of link
postedTime	Timestamp	Posting Time
updatedTime	Timestamp	Update Time
summary	Varchar	Actor's profile
link	Varchar	links related to profile
symbold	Varchar	stock's ticker related to profle

Continuation of Table 3.1		
Field	**Data Type**	**Description**
sentiment	Varchar	A sentiment, binary classification
Chart	Varchar	Link to a graph
Video	Varchar	Link to a video

3.1.2 Financial Time Series

For the 6 years timespan of the investigation, financial data have been extracted from Yahoo Finance API via *Python* scripting, and joined with the tweet in which the stock is mentioned using the tweet date. Data is including, for each stock ticker:

- *Date*: Date of quotation, it might not coincide with the tweet date, since people are twittering continuously, while Stock Exchanges are open 252 days a year.
- *Volume*: Number of stocks exchanged.
- *Day Close*: Quoted price at the end of trading day.
- *Day Open*: Quoted price at the beginning of trading day, it might differ from previous day closing price.

Three triples of Volume, Closing price and Opening Price will be associated to each row in the final dataset, with the following criteria:

- *Same Day*: Value on the Same Day of the Tweet. If the Stock Exchange is closed that day, the first day of opening immediately after is selected, no matter how far in time.
- *Day After*: Value on the day after *Same Day*. Same criteria as above applies.
- *5 Days After*: Value in 5 calendar days from *Same Day*. Same criteria as above applies.

It's worth to mention that some Tweets, in the final dataset, were associated to stocks with Quoted Price far ahead in the future: investors were discussing about Facebook, for instance, months before its *Initial Public Offering*.

3.2 Data Preparation

From the Tabulated Version of the data, cleaning transformations took place in an EMR cluster, to populate dedicated Hive Tables as in the schema in figure 3.3.

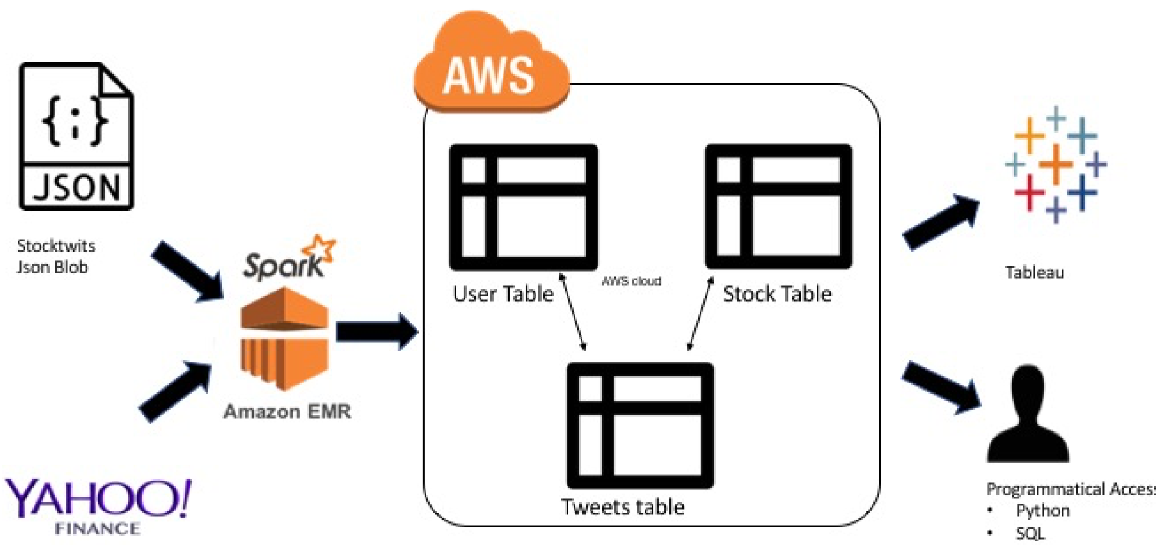


Figure 3.3: StockTwits Data Model

This allowed to source the data really easily, for further analysis via *SQL*, or via *pandas* library in *Python* and even via direct Access with Tableau reporting tool.

3.2.1 Data Model

Data extracted from the two distinct sources, are transformed and loaded in three dedicated tables, listed below, to ease up access and analysis.

- **Tweets**: it's storing the message body, the date of the tweet and the *tweet_id*, it can be joined with user table on *user_id*, and with stock table on *stock_id*, along with the date. The table stores important information as the self tagged *sentiment*, and the *user_id* quoted, retweet or replied to. The last information has been looked up from *user_name*, originally present in that field in the JSON blob.

- **Users**: source of this table are users identified in StockTwits tweets; primary key for the table is the *user_id*, immutable field, on the opposite of *user_name*, that can be changed in any instant, and can belong also to platform distinct from StockTwits. in Tweet table, all occurrence of *user_name* have been looked up to *user_id* at that time. Important information, such as *follower_count*, have been calculated directly from the data. Other, as the self-tagged *tradingStrategy* and *experience*, have been ignored, since not backed by any verification.

- **Stocks**: source of this table are information coming from Yahoo Finance, limited to the stocks mentioned in the StockTwits JSON blob. An Identity value for each stock, *stock_id*, has been generated, to manage potential change of stock tickers, or potential change of stock exchange. The table can be joined on the afore-mentioned Id and on the date, to the *tweets* table; this prevented on purpose to take into consideration stocks that were de-listed, following company's acquisition or failure. Users kept talking of such events, and companies, even years after they happened.

3.3 Social Network Features Computation

Once the data were shaped in a proper tabular way, relations were extracted removing redundant information and engineering additional ones; graphs were then built creating network of the relations leveraging *Networkx* library from *Python*. Features representative of Stability, Activity, Fragmentation, number of nodes and edges will be extracted from the Networks across time; those were helpful also understand how the graphs evolved and behave in different market's contexts, such as flash crashes or disclosures of earnings from the companies. All this features will then go through the predictive algorithm.

Following existent case in literature, two distinct kind of Graphs were modelled and their metrics were calculated daily:

(i) **G01** : an Undirected graph, it has been built monitoring all the possible *interactions between users*, on the base of the work of Ruiz et al. (2012) and Casnici et al. (2015). As already mentioned, interactions can be of three possible types:

   (a) *Quotation:* two nodes, representative of 2 users, are considered connected by an edge, when a user is quoting the others, via the @ notation in the original message body, followed by the *user_name*.

   (b) *Reply:* two nodes, representative of 2 users, are considered connected by an edge, when a user is replying to another, and this is represented by a *user_name* filling the *reply_to* field.

   (c) *Retweet:* two nodes, representative of 2 users, are considered connected by an edge, when a user is quoting the others, via *RT* @ notation in message body, followed by the *user_name*.

(ii) **G02**: a Bipartite Graph, taking into account *Interaction between stocks and users*, considering users and stocks as nodes, and a tweet from a user on a stock, as an edge. This is similar to the experiment executed in the Roy and Sarkar (2011) and in Sankar et al. (2015), where however, were analysed relations between stocks and indices or mutual investment funds.

By technical point of view, SQL Statements ran against the Hive database were extracting the aforementioned relations in forms of edge, and attaching to them labels that were used to indicate the day in which the relation took place, and the security this relation was about(only in the case of graph *G01*). The list of the nodes then was derived from the list of edges, and in this essay there was no labelling on the nodes; neither particular weights were applied to the edges.
Sub-sequentially, for every node, and for each day, were calculated the features listed in the paragraphs below. Only those related to the users and the graph in general

were taken into consideration, and added to the final dataset, those related to stocks in the bipartite graphs *G02* were not taken into account.

3.3.1 Nodes related metrics

In the list below, all the features that have been calculated each day for each node belonging to the two distinct generated graphs *G01* and *G02*.

- **Degree Centrality:** for every single node, it represents the number of edges leading to it, divided by the total number of nodes. In Social Networks theory, the higher it is, the more important is a user within the community. (M. Newman, 2010). For a given node n in a Graph G, it is defined by number of edge incident upon it, divided by total nodes N.

$$C_{\mathrm{D}}(n) = \frac{deg(n)}{N} \tag{3.1}$$

- **Betweenness Centrality:** measure of centrality based on the shortest paths, it's the fraction of the shortest path passing for a node, divided by the total number of shortest paths. A node with high value of this feature, has more control over the graph, since it's passage point of crucial information. (M. Newman, 2010).
  Its value for a node n in graph G can be computed in the following way:

  1. for each pair of node x, y determine the shortest path linking them.
  2. for each pair of node x, y determine the shortest path linking them and passing by n.
  3. divide the item 1 by 2, and do that for all the couple of nodes.

  Being σ_{xy} the first item, and σ_{xy}(n) the second one, betweenness centrality, in a compact way, can be expressed by equation 3.2:

$$C_{\mathrm{B}}(n) = \sum_{x \neq n \neq y} \frac{\sigma_{\mathrm{xy}}(n)}{\sigma_{\mathrm{xy}}} \tag{3.2}$$

- **Closeness Centrality:** indicator calculated as the sum of the shortest paths length between the node and all the other nodes in the networks. The more central a node is, the closer it is to all other nodes. Originally proposed by Bavelas (1950). Given a node n, it can be expressed by equation 3.3, summing the reciprocal of the distance between n and its neighbours.

$$C_{\mathrm{C}}(n) = \frac{1}{\sum_{y} d_{\mathrm{n,y}}} \tag{3.3}$$

- **PageRank:** measure based on the algorithm originally proposed by Page, Brin, Motwani, and Winograd (1998), and named after its main creator. Measuring the importance of a node within a graph, weighting the edge leading to it; the higher is PageRank value, the more important is the node.
  It can be calculated by equation 3.4, where α is an attenuation factor between 0 an 1, and L(j) is equals to $\sum_{i}$ a_{ji}, giving i the number of neighbour nodes.

$$P_{\mathrm{r}}(n_{\mathrm{i}}) = \alpha \sum_{j} a_{\mathrm{ji}} \frac{n_{\mathrm{j}}}{L(j)} + \frac{1-\alpha}{N} \tag{3.4}$$

  where n_i is a node, and N is total number of nodes.

- **Load Centrality:** fraction of all the shortest path that pass for that node. (M. E. Newman, 2001).
  Slightly different from Betweenness centrality, it's defined by a hypothetical flow process with a routing criteria establishing priority.

$$C_{\mathrm{B}}(n) = \sum_{x \neq n \neq y} \frac{\sigma_{\mathrm{xy}}(n)}{\sigma_{\mathrm{xy}}} \epsilon_{\mathrm{xy}} \tag{3.5}$$

  Factor ϵ_{xy} keeps into account this routing criteria.

- **Communicability Centrality:** the sum of closed paths starting and ending at the node, measure also the importance of a user in terms of information exchange within the network. (Estrada & Rodriguez-Velazquez, 2005) it is defined for a node n by equation :

$$C_{\mathrm{A}}(n) = \sum_{i=1}^{n} [e^{A}1]_{\mathrm{i}} \tag{3.6}$$

where A is the adjacency matrix of its eigenvalues.

- **Degree partition:** Compute the partition of the graph nodes which maximises the modularity; for this the node is assigned to a community using the Louvain heuristics. (De Meo, Ferrara, Fiumara, & Provetti, 2011).

3.3.2 Graph related metrics

in the list below all the features that have been calculated each day for the generated graphs, they were describing higher level characteristics and phenomenons involving all the nodes for the graph.

- **Modularity:** indicator that provides insight on how the graph is divided in communities; high modularities means many edges between nodes in the same community, but few edges between nodes in different communities. (M. E. Newman, 2006)
  It's defined as the ratio of the edges that fall within the given community minus the expected ration if edges were randomly distributed. For a graph G with n nodes and m edges, assuming that each node belongs to a community c_{u}, the function $\delta(c_{\mathrm{u}},c_{\mathrm{v}})$ will be equal to 1 if communities c_{u} and c_{v} matches perfectly, and will be equal to 0 whether they have no node in common.
  Given d_{u} and $d_{\mathrm{v}}$the degree of centrality of this two nodes, and A the matrix of adjacency of graph G(equal to 1 if u and v are adjacent, and 0 if not) the Modularity Q for this community partition is:

$$Q = \frac{1}{2m} \sum_{u,v} (A_{\mathrm{u,v}} - \frac{d_{\mathrm{u}} d_{\mathrm{v}}}{2m}) \delta(c_{\mathrm{u}}, c_{\mathrm{v}}) \tag{3.7}$$

- **Average clustering:** measure proposed by D. J. Watts and Strogatz (1998) and equals to the average of local clustering coefficients in all the nodes of the graph; as clustering coefficient is meant an indicators that tells how strongly

nodes tend to knit together within a graph. Evidences are suggesting that nodes tend to coalesce together in Social Networks. (Holland & Leinhardt, 1971).
As specified above, clustering is a metric calculated at node level, and, in this metric, it will be averaged across all nodes in the graph. It's defined for a particular node n with equation 3.8, considering N the total number of nodes in the graph, and d the degree of centrality of n.

$$C(n) = \frac{N}{d_{\mathrm{n}}(d_{\mathrm{n}} - 1)} \tag{3.8}$$

- **Assortativity:** measure of similarity that indicates how likely are nodes within a graph to attach to each others. (M. E. Newman, 2002).
  This can be also defined as the pairwise Pearson correlation coefficient of degree between linked nodes, and varies between -1 and 1.
  The formula for it is expressed by 3.9, given e_{jk} the joint probability distribution of the other degree, q the distribution probability of the remaining degrees, and σ is the standard deviation of the difference of the degrees.

$$A = \frac{\sum_{j,k} jk(e_{\mathrm{jk}} - q_{\mathrm{j}}q_{\mathrm{k}})}{\sigma^2{}_{\mathrm{q}}} \tag{3.9}$$

- **Estrada Index:** sum of subgraph centralities, it's an indicator of the folding degrees of the subgraphs present in the main graph.(de la Peña, Gutman, & Rada, 2007).
  For a graph G, it can be obtained by summing the subgraph centralities for all the nodes n; subgraph centrality is given by equation 3.10, where A is the adjacency Matrix of its eigenvalues.

$$E(n) = \sum_{k=0}^{\infty} \frac{A^k{}_{\mathrm{nn}}}{k!} \tag{3.10}$$

3.4 Text and Sentiment Analytics

Text Analysis has been performed on the message, *body* field in the transformed data; said field is limited to 140 characters, like in most of microblogs platform, and might

contain also other important information, that have been leveraged in other part of the present work, like in the graph generation bit.
Specifically, the text field might contain the item listed in table 3.2:

Content	**Notes**
Stock tag	Reference to particular equities, fundamental for the present work. It can be in the measure of 0,1 or N, the stock tag can also refer to bonds, to indices, to funds, to forex or can tag the text as some kind of analysis
Retweet reference	Expressed by the letters RT followed by a symbol @ and the name of the user, that can be from StockTwits platform of from twitter. It can be in the measure of 0,1 or N
Quotation reference	Expressed by the symbol @ and the name of the user, that can be from StockTwits platform of from twitter. It can be in the measure of 0,1 or N
Sentiment	In the measure of 0 or 1. When present, it might be *Bullish* or *Bearish*
Link	URL to sites or to online Newspaper articles; those are not taken into account in the present work
Images	They might be a simple avatar, or, in some case, Technical or Fundamental analysis explained by graphs; those are not taken into account in the present work

Table 3.2: Possible content of Message Body

In terms of Text Analytics, the following specific features have been looked upon, calculated, and then used in the Final Dataset.

1. Count of the number of tweets.

2. Number of words containing the terms *"Bull"* or *"Bear"*, as in Bollen et al. (2011).

3. Bullish or Bearish Self Tag, that StockTwits platform allows end-user to tie to their tweets, as in McNamara (2016).

4. Citation done by the actors, and number of time the author is cited.

5. Reply done by the actors, and number of time the author is receiving a reply.

6. Retweet done by the actors, and number of time the author retweets.

Finally, for the Sentiment Analysis part, the polarity indicator from *TextBlob Python* library has been used, as in the work of Chee (2017), where led to consistent results when used on Financial Microblogs. The library, built on top of NLTK, takes advantage of a Naive Bayes Classifier, trained on known words and sentence, that assigns a value of sentiment polarity between -1 and 1 to each tweets, and allows also to calculate an objectivity score, that wasn't used in the present essay.

3.5 Final Dataset

In the table 3.3 is described the final dataset that has been feeding the predictive algorithm, obtained enriching the transformed *StockTwits* dataset, with the features mentioned in the previous sections.

Table 3.3: Final Dataset with enriched fields.

Field	**Comments**	**Domain**
Tweet_date	Existing for 365 days a year, while Stock Exchange opens only 252 days	
User	StockTwits ID	
Stock	Stock Ticker	
nbr_tweet	Count of Tweets	Traffic
nbr_retweet_out	Number of Retweets from other user	Text-based
nbr_retweet_in	Number of Retweet of actor's tweets from other users	Text-based
nbr_reply_out	Number of Replies to other users	Text-based

Continuation of Table 3.3		
Field	**Comments**	**Domain**
nbr_reply_in	Number of Replies from other users to him	Text-based
nbr_citation_out	Number of citation of other user	Text-based
nbr_citation_in	Number of citation from other users	Text-based
nbr_positive_polarity_tweet	Naive Bayes Based	Sentiment
nbr_negative_polarity_tweet	Naive Bayes Based	Sentiment
nbr_bullish_in_text_tweet	Count of tweet with Bull word in text	Text-based
nbr_bearish_in_text_tweet	Count of tweet with Bear word in text	Text-based
nbr_bullish_self_tag_tweet	Count of tweet tagged by actor as bullish	Text-based
nbr_bearish_self_tag_tweet	Count of tweet tagged by actor as bearish	Text-based
open_same_day	Stock opening price same day	Financial
close_same_day	Stock closing price same day	Financial
volume_same_day	Stock volume same day	Financial
open_day_after	Stock opening price day after	Financial
close_day_after	Stock closing price day after	Financial
volume_day_after	Stock volume day after	Financial
open_5days_after	Stock opening price 5 days after	Financial
close_5days_after	Stock closing price 5 days after	Financial
volume_5days_after	Stock volume 5 days after	Financial
spy_open_same_day	S&P500 opening price same day	Financial
spy_close_same_day	S&P500 closing price same day	Financial
spy_volume_same_day	S&P500 volume same day	Financial

Continuation of Table 3.3		
Field	**Comments**	**Domain**
spy_open_day_after	S&P500 opening price day after	Financial
spy_close_day_after	S&P500 closing price day after	Financial
spy_volume_day_after	S&P500 volume day after	Financial
spy_open_5days_after	S&P500 opening price 5 days after	Financial
spy_close_5days_after	S&P500 closing price 5 days after	Financial
spy_volume_5days_after	S&P500 volume 5 days after	Financial
retweet_node_degree_centrality	Node metric related to users interactions tweets' network	Network
retweet_node_close_centrality	Node metric related to users interactions tweets' network	Network
retweet_node_betweeness_centrality	Node metric related to users interactions tweets' network	Network
retweet_node_page_rank	Node metric related to users interactions tweets' network	Network
retweet_node_load_centrality	Node metric related to users interactions tweets' network	Network
retweet_node _communication_centrality	Node metric related to users interactions tweets' network	Network
retweet_node_degree_partition	Node metric related to users interactions tweets' network	Network
retweet_modularity	Network metric related to users interactions tweets' network	Network
retweet_avg_clustering	Network metric related to users interactions tweets' network	Network
retweet_assortativity	Network metric related to users interactions tweets' network	Network

Continuation of Table 3.3		
Field	**Comments**	**Domain**
retweet_estrada_index	Network metric related to users interactions tweets' network	Network
bipartite_node_degree_centrality	node metric related to user-stocks network	Network
bipartite_node_close_centrality	node metric related to user-stocks network	Network
bipartite_node _betweeness_centrality	node metric related to user-stocks network	Network
bipartite_node_page_rank	node metric related to user-stocks network	Network
bipartite_node_load_centrality	node metric related to user-stocks network	Network
bipartite_node _communication_centrality	node metric related to user-stocks network	Network
bipartite_node_degree_partition	node metric related to user-stocks network	Network
bipartite_modularity	network metric related to user-stocks network	Network
bipartite_avg_clustering	network metric related to user-stocks network	Network
bipartite_assortativity	network metrics related to user-stocks network	Network
bipartite_estrada_index	network metric related to user-stocks network	Network

Continuation of Table 3.3		
Field	**Comments**	**Domain**
return_over_spy_diff5days	percentage return of the stock subject of the tweet over S&P500 in 5 days, used alternatively at pos_return_5days_on_spy	target variable
pos_return_5days_on_spy	binary value equals to 1 if the stock subject of the tweet will outperform S&P500 in 5 days, or to 0 otherwise. Used alternatively at return_over_spy_diff5days and calculated based on percentages	target variable

3.6 Correlation Testing

Correlation was explored between the generated features and both target variables, to assess whether some features could be removed from dataset or whether they had more predictive power than the other. This measure was fundamental to assess Hypothesis H_{02}; low correlations were expected from literature, as in Oliveira et al. (2013b), in Ruiz et al. (2012) and, more recently, in T. Wang et al. (2017).
Pearson Correlation Coefficient will be adopted and used pairwise between each element of the population; this is defined by the formula 3.11 where X and Y are two elements, cov is the covariance and σ is the standard deviation.

$$\rho_{\mathrm{X,Y}} = \frac{cov(X,Y)}{\sigma_{\mathrm{X}}\sigma_{\mathrm{Y}}} \tag{3.11}$$

3.7 Predictive Modelling

The main purpose of the machine learning algorithms presented in this work was to assess the predictive power of the community of *StockTwits.com*, in order to test the

hypotheses listed in section 1.3; to do so, the final dataset helped learning series of model built with the Keras Framework, developed by Chollet et al. (2015), wrapped into a ScikitLearn pipelines. (Pedregosa et al., 2011)
Distinct Algorithm on distinct portion of the Final Dataset went through training, Specifically.

(i) **ANN01**: Model trained on all the instances in the period between 1st of January 2010 and 31st December 2014.

(ii) **ANN02**: Series of Model, each trained on a single user, in the period between 1st of January 2010 and 31st December 2014.

(iii) **ANN03**: Series of Model, each trained on a single user, in the period between 1st of January 2015 and 31st December 2015.

In the picture 3.4 is synthesised the whole Training Strategy, and the domain of each model or Series of Model, including Time Range and User Repartition.

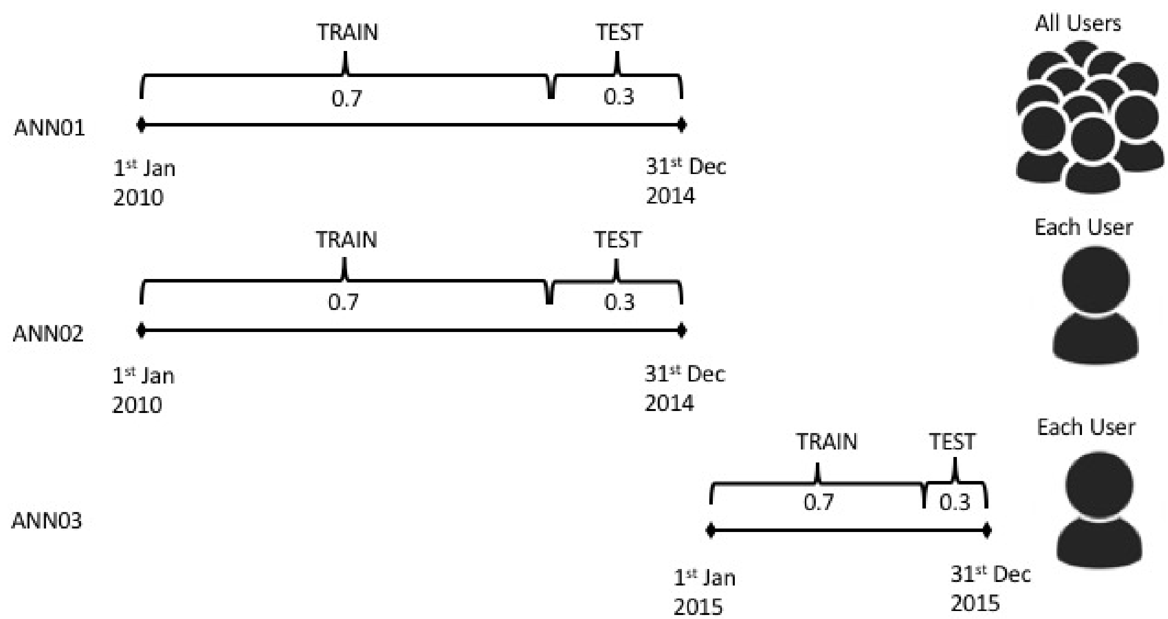


Figure 3.4: Models Training Strategy

Each Model, or series of Models, listed above, was used to predict the market directions and the value of the returns over the market, from specific partition of the Final Dataset, described in section 3.5. In table 3.4, target variable, prediction type and optimisation procedure used are highlighted; it's important to remark that both target variable were calculated from percentage, since they might be expressed in different currency and have scale not comparable.

Algorithm Type	**Optimisation**	**Target Variable**
Binary Classification	Binary Cross-Entropy	return_over_spy_diff5days
Regression	Mean Squared Error	pos_return_5days_on_spy

Table 3.4: Predictive Algorithms

Deep diving on the algorithms construction phase, that will be better describe in the next chapter, the following HyperParameter and Configurations were optimised, to obtain a tuned Multi-Layer Perceptron, using a small portion of the dataset, that will be then discarded.

- Batch size and number of Epochs.
- Optimizer.
- Activation Function and its parameters.
- Number of Hidden Layers.

The best performing ANN topology, coming by the validation phase, went eventually through the following steps, to generate weights for the models, keeping track on each epoch of the best value for Accuracy, Precision, Recall and F1-score, in case of the Binary Classification, and of Median Square Error, in case of Regression.

1. **Iteration** through all the users of the Social Network, with a minimum number of rows, *380*, set to 10 times the number of attributes, as recommended by Raudys, Jain, et al. (1991). This step was undertaken just in models *ANN02* and *ANN03*, while in model *ANN01* all the instance of the dataset were taken into account.

2. **Dropping of all the instance with null features**, since of the impossibility of the MLP Algorithm to handle them, as described in Chapman et al. (2000). Basically, just users that belonged, in a specific day, to the generated graph of users' interaction *G01* were taken into consideration, since the network metrics related to this graph can be null for some user in some day. This was applied to the three series of model, and caused the discarding of many instances.

3. **Dropping all the features not needed**, such as *user_id*, *tweet_date* and all those related to financial market data that are in relation with opening or closing prices. Volumes of exchange have been retained.

4. **Centering and Scaling** all the retained features; this operation, also known as standardisation, consisted in subtracting the mean from each field, and dividing it by its standard deviation.

5. **Neural Network Processing** through 500 Epochs, with a split 70/30 between Training and Test sets, where data were ordered with time, so the most recent data ended up in the Test portion. Different activation functions were used in the final node, according to the prediction type.

The weights obtained for a model that was performing better than the predecessor, in terms of test value, were retained for each epoch; eventually, predictions with the best performing series of weight on specific partition of the dataset were brought up.

3.7.1 Investors Ranking

The results obtained for each user were then compared with those obtained training on the full dataset; to evaluate whether performance of the users were consistent across the time, a Wilcoxon signed rank test was executed between paired sample of predictions of same user groups at different time, to verify null hypothesis H_{03}. The Wilcoxon test was selected as a perfect fit for the case, since it's suited for paired samples, same

group but two different periods, and it's not-parametric, thus not requiring previous normality assumption.(Randles, 1988)
Test is executed under its own null hypothesis H_0 that the distribution of the differences between two paired samples follow a symmetric distribution around 0; and is performed ranking the two sample and calculating the W statistics as in equation 3.12.

$$W = \sum_{i=1}^{N} [sgn(x_{2\mathrm{i}} - x_{1\mathrm{i}}) R_{\mathrm{i}}] \tag{3.12}$$

Where $\mathrm{x_{2i}}$ are $\mathrm{x_{1i}}$ element of the 2 samples, each consistent of the same N elements, and $\mathrm{R_i}$ is the ranking of the Samples. Under null hypotheses, W follow a simple distribution with sum value of 0, and a pre-defined variance dependent of sample size.
To perform a two-sided test, z value can be calculated as $\frac{W}{\sigma}$, where σ is the variance; to reject null hypothesis H_0 of no difference between samples, the absolute z value, calculated with significative statistic precision, must be greater than the critical value, that for 0.05 p value, used in this test, is equal to 1.96.

As further verification, to assess the degree of correlation of the ranking calculated at different periods, a Kendall rank correlation coefficient was extracted. (Kendall, 1955)
This coefficient is measuring ordinal association between quantities, is non-parametric and is defined by equation 3.13

$$\tau = \frac{(n_{\mathrm{c}} - n_{\mathrm{d}})}{\frac{N(N-1)}{2}} \tag{3.13}$$

Pairs are same element taken from different rankings, and are said to be concordant when their position matches, and discordant when it doesn't. N is total number of element, while the number of concordant pair is $\mathrm{n_c}$ and the number of disconcordant pair is $\mathrm{n_d}$.

The predictions obtained by the fitted models, were used eventually to propose a Trading Schema.

Chapter 4

Implementation and results

In this section of the thesis, the results and the implementations of the experiment will be presented in details:

- The Software tools leveraged, and all the practical aspect related to the experimentation will be initially discussed.
- Text Analytics and Sentiment Mining implementation and results will then be introduced.
- Metrics and techniques associated to the generation of the 2 networks built on top of *StockTwits* data will be the core of a specific section.
- Visualisations of correlation between features and target label of the dataset will be presented.
- Tuning of the dataset will be analysed, with focus on the choice of the optimiser, and of the ANN topology, including number and type of layers.
- Finally, Data about the executions of the Predictive Algorithms will be discussed.

4.1 Software Tools

As already mentioned, the Initial dataset was a nested *JSON*, containing around 37 Millions of distinct Tweets.

All the Data Cleaning and Data Preparation phase was executed on an *AWS* account, uploading first the source files on a *S3* bucket, and inferring and transforming their structure via *Apache Spark* on *EMR* jobs. The data has been eventually loaded into *Hive* Tables, as described in section 3.2.1; all the following transformation have been then performed via *SQL* queries via *Athena* services, to produce a series of dataset that will be subject of *SNA*, *Text Analysis* and *Sentiment Mining*.

Text Analysis and *Sentiment Mining* Calculation were executed using *Textblob* and *Pandas Python* libraries; *SNA* Calculation, on other hand, has been performed by a *Python* script leveraging *NetworkX* library. Considering the high number of edges, the two scripts, one related to the users' interaction graph *G01*, and the one related to bipartite graph *G02* between users and stocks, have been uploaded to an *AWS EC2 C5.18XL* instance, to leverage its enhanced calculation capacity; the tasks, however, took several days to complete for each of the dedicated script, being the bipartite the one that lasted the most, standing its biggest dimension.

Once analysis from experiment describe in sections 3.4 were consolidated, features obtained were again loaded into *AWS S3*, enriched with Financial data from *Yahoo*, transformed and loaded into the shape of the dataset described in section 3.5.

The Final Dataset was ready to go through the learning of the models, and *Keras Python* scripts on *TensorFlow* back-end, wrapped into a *ScikitLearn* pipelines, were deployed to an *AWS EC2 G3.4XL* instance, maximising Graphic computing power, to perform calculations that lasted several days.

The outcome went finally through Correlation Measures, Wilcoxon Signed ranked test and calculation of Kendall's Rank Coefficient, that have been performed on a *Jupyter* notebook via *Scipy Python* package.

Final visualisation that had as output the figures in the present works, were created in part with *Tableau* community edition, and in part with *Python* libraries such as *Seaborn* and *Matplotlib*.

4.2 Text Analytics Computation

The final dataset has been enriched by features coming by Text and Sentiment Analytics. The first activity was carried out simply analysing the message content, with regards to the self-tagging traces possibly left behind by the actors, the second took advantage of *TextBlob* library, to calculate the sentiment polarity, and then counting it as positive whether greater than 0, or negative in the opposite case.

4.2.1 Text Analysis

The dataset was mined for word containing either "Bull" or "Bear", and the sentiment self-tag by users were counted. In picture 4.1, the figure pulled per year: It's evident the constant optimistic interpretation from Social Network participant, supported in this sense by the long bull run of the capital markets.

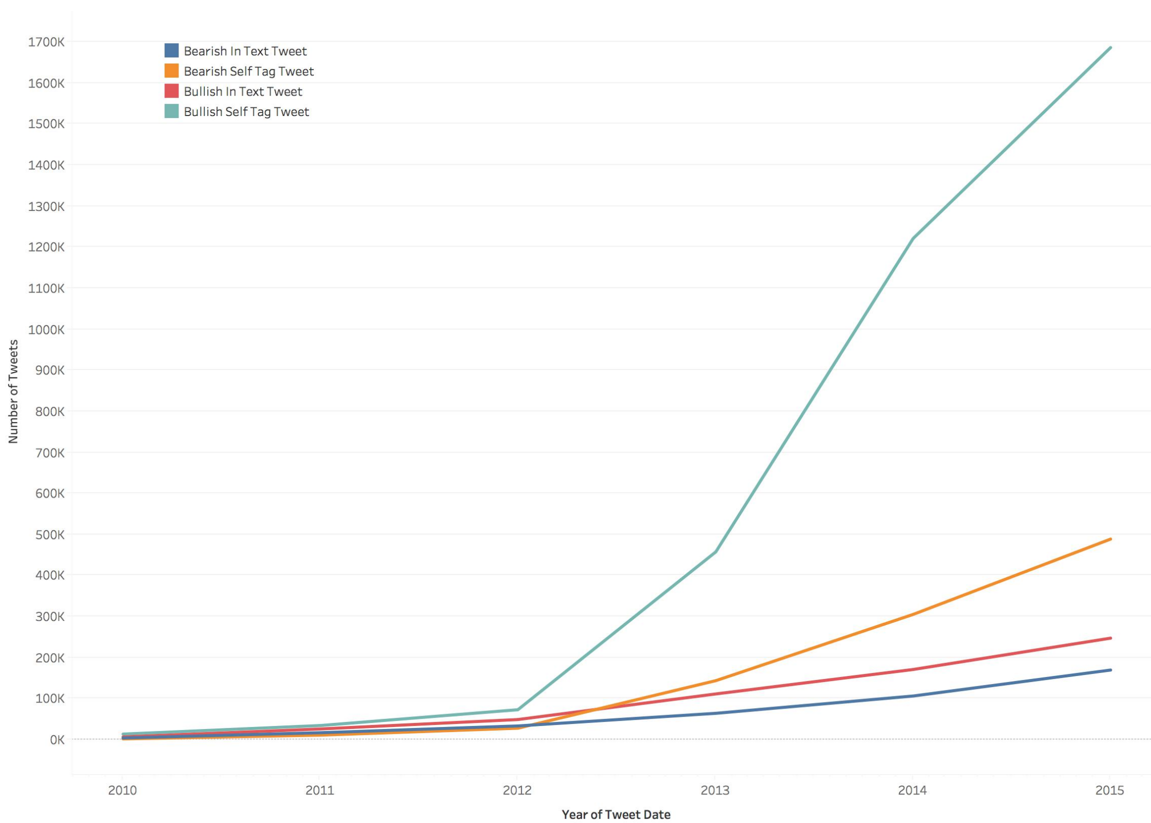


Figure 4.1: Text Analysis Results per Year

Bullish feelings outnumbered by far the Bearish ones, as is visible in the graph, and people preferred to express their opinion about market's direction via the self-tagging option, rather than writing it explicitly on the message. Just 11% of the Tweets, however, presented a self-tag, but this percentage tended to increased with the adoption of the platform.

Information about users' interactions were also counted: they were of paramount importance also in a different section of the experiment, where a daily graphs depicting user's interactions were built. The attributes extracted about user's participation on *StockTwits* were also giving insight on platform usage.

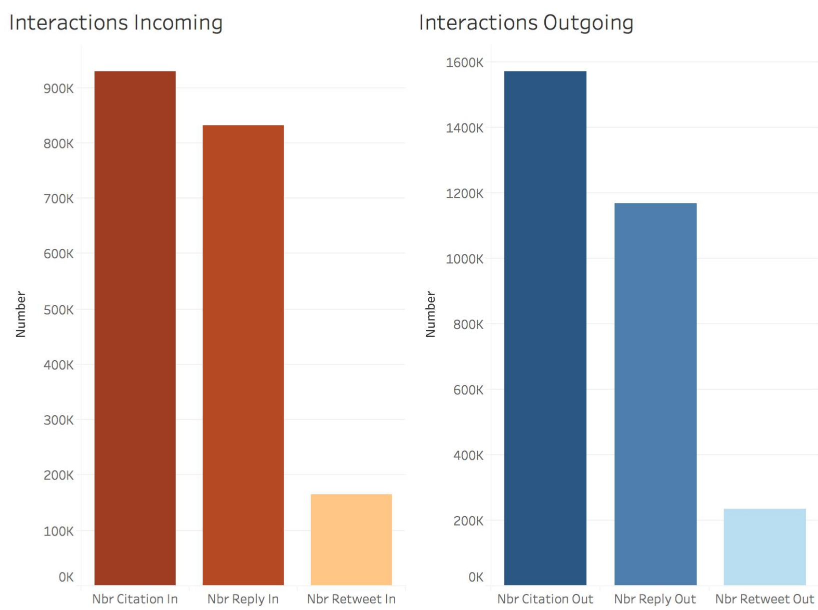


Figure 4.2: Interactions by type

From 4.2 is clear how user are more likely to interact replying, retweeting and citing someone, as opposite to receive a reply, a retweet or a citation; by far the citation(including someone in the message with a @ symbol) was the interaction method most used, both outgoing and incoming.

4.2.2 Sentiment Analysis

Outcome from Sentiment Analysis are confirming what already described in section 4.2.1; the actors showed a positive polarity that is never overtaken throughout the time by its negative counterpart. In the graph 4.3 is shown its trend against the time.

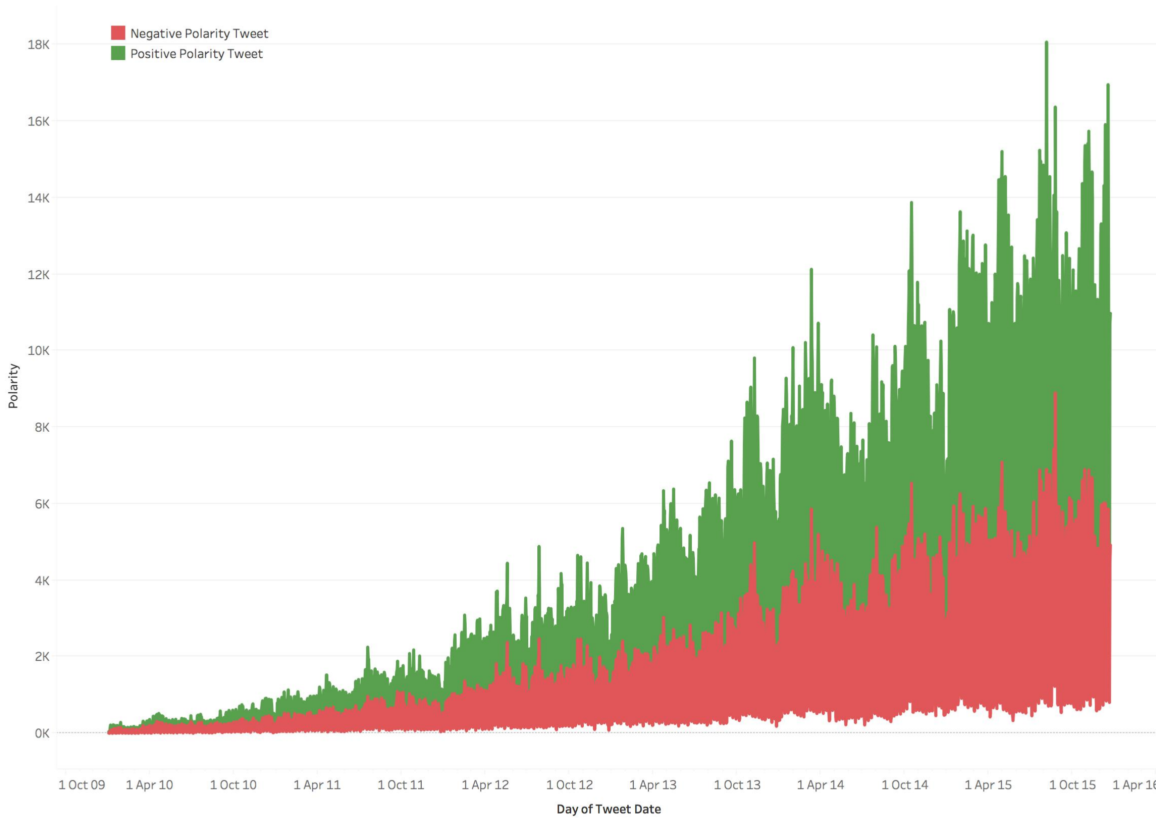


Figure 4.3: Sentiment Polarity by Date

4.3 Graphs features calculation

As detailed in section 3.3, two Graphs *G01* and *G02* were generated from *StockTwits.com* data, that were beforehand wrangled and cleaned up for the purpose.
As already described in details, for the graph *G01* were taken into account interactions between users, in terms of citation, reply and re-tweets, that were mapped as edges. Via a script offloaded to a server, were calculated for each day the metrics related to the network, and to each of its nodes.
The graphs built in this way didn't take into account directionality, didn't distinguish

about the type of Interaction and didn't weight its edges; a minority of the tweets could contribute to the graphs, and this was the reason to drop the majority of the data before the learning phase; from 87 thousands different users registered to the platform, only 79 thousands appeared as nodes in the interactions' network, in different days during the 6 years of observation.
A centrality degree distribution is shown in the chart 4.4 for a trading day with some volatility on a log scale; for all day a similar graph can be calculated, with most of the node having a degree of centrality equals to 1.

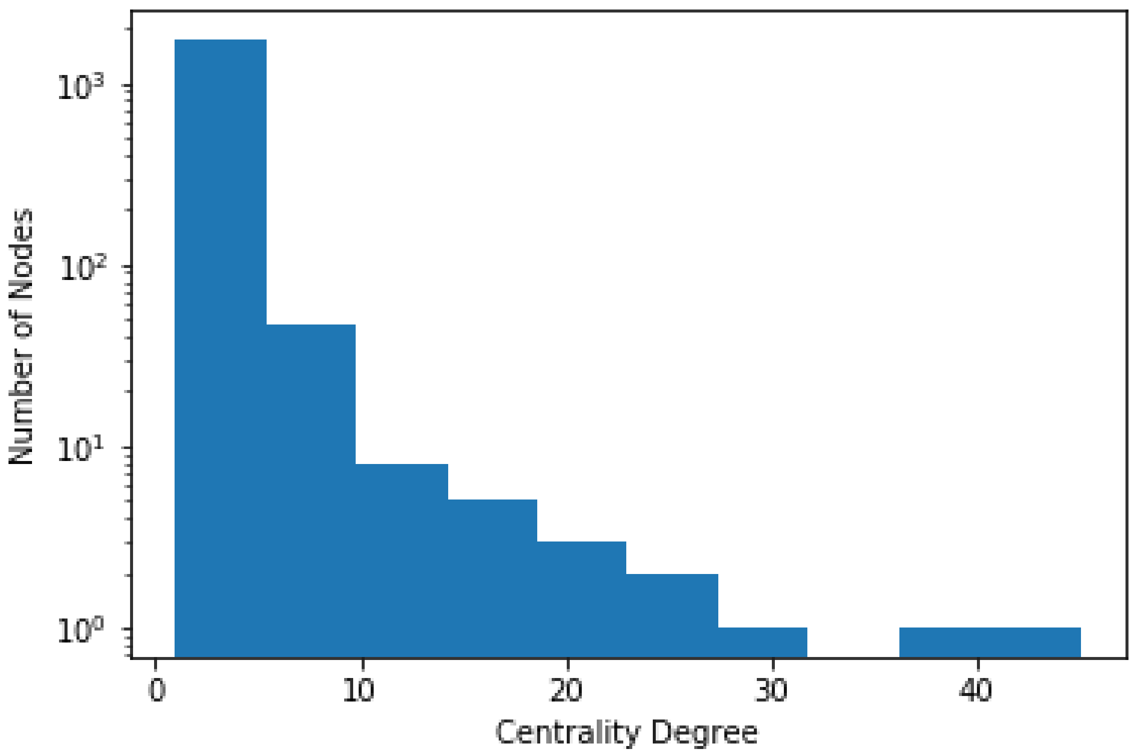


Figure 4.4: Distribution of degrees for a Volatile day

The second Graph *G02*, the one built on the interactions between stocks and users, was built with the same scripting criteria and wanted to document how users are twittering about stocks, and how they change preferences across the time. The graph was a bipartite one, with most of the edges connecting to a minority of stocks: in figure 4.5 are charted the edges per stock over the 6 years period; only the top 10, but is notable how some of them seem to gather most of the attention of the tweeters.

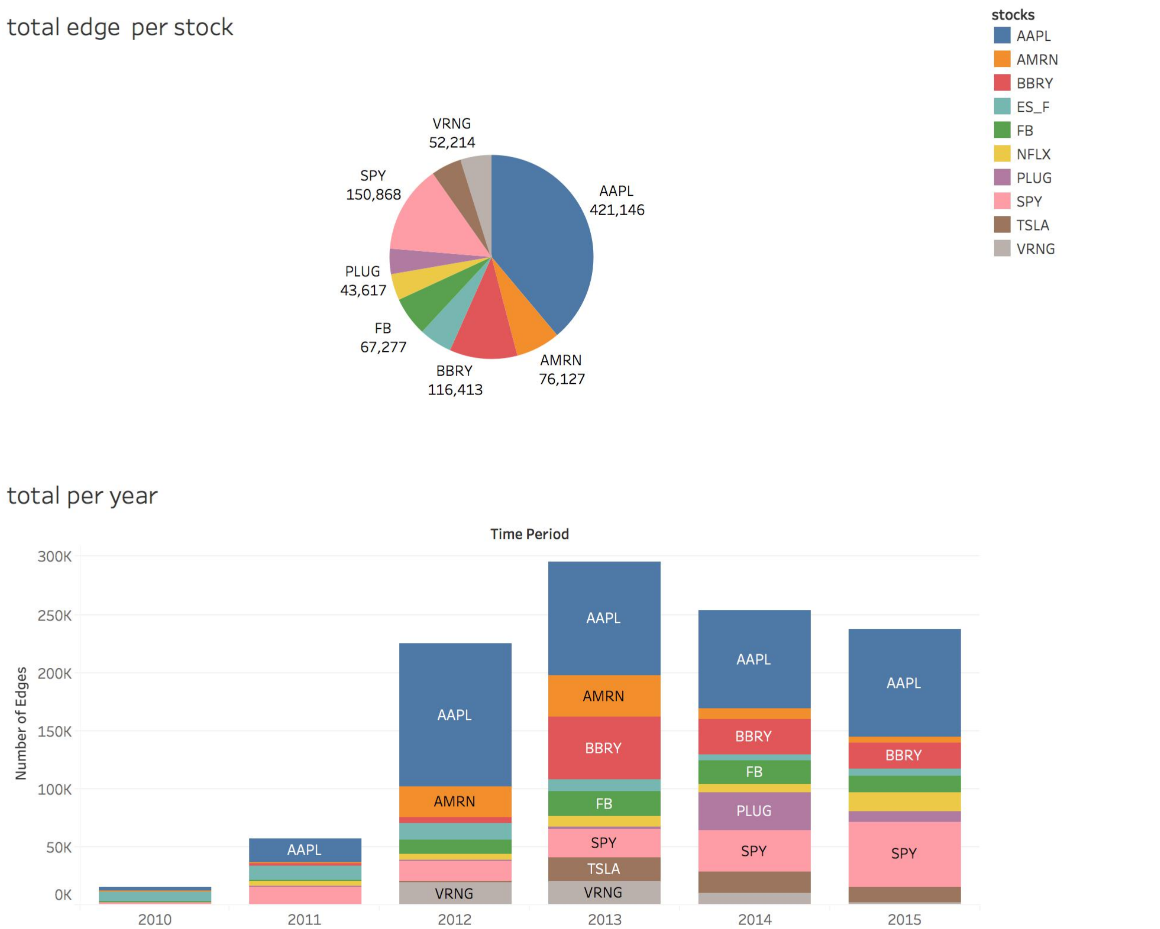


Figure 4.5: Number of connections per stock per year

In figure 4.5 can also be noted as some stocks are getting traction during some specific periods, while others are constantly interesting investors: AAPL, the ticker of Apple Corporations, maintains constantly the first position, throughout the time, and interest for SPY, ticker used for the S&P500 index is also alway kept up.
Other securities became popular in the recent past, as Tesla(TSLA); other rose into popularity for some particular condition that made them main characters in financial headlines for short period. It's the case of Vringo (VRNG), that was popular between 2012 and 2013 for the patent war it fought against Google, or Amarin(Amrn) that went through US Food and Drug Administration Department trial, for a patent on the only drugs it was producing.

Also for the Stock-Users Interaction graph *G02*, a good part of tweets were not con-

sidered: of the almost 37 Milions of tweets included in the original JSON blob, only 58% had included a Stock Tag, and therefore could be analysed in the present work. Of About 10 thousands stocks then referred by the users, around 40% where in more than 380 tweets, a number that will be used as a cut-off during the experiments. Some Tickers, moreover, were not referring to Securities quoted in some Stock Exchange all over the world, but were tag for Fundamental or Technical Analysis, such as 'ES_F', that was the fifth most recurring tag.
Also, some tweets were focused on other financial products, such forex, options, bitcoin and others; therefore they were excluded.

4.4 Correlation

The final dataset, containing the fields listed in table 3.3, and obtained by adding to the original data all the features coming from *Text Analytics*, *SNA* and *Sentiment Mining*, went through correlation testing.
The only features related to to financial market retained where Volume on the same day, Volume the day after and Volume in 5 days; obviously the target value, the difference between the price changes of S&P500 and the stock subject of the tweet was retained. All the attributes related to the date, to the user and to the name of the Stock were dropped.

In the figure 4.6, the correlation matrix between all the attributes included in the dataset.

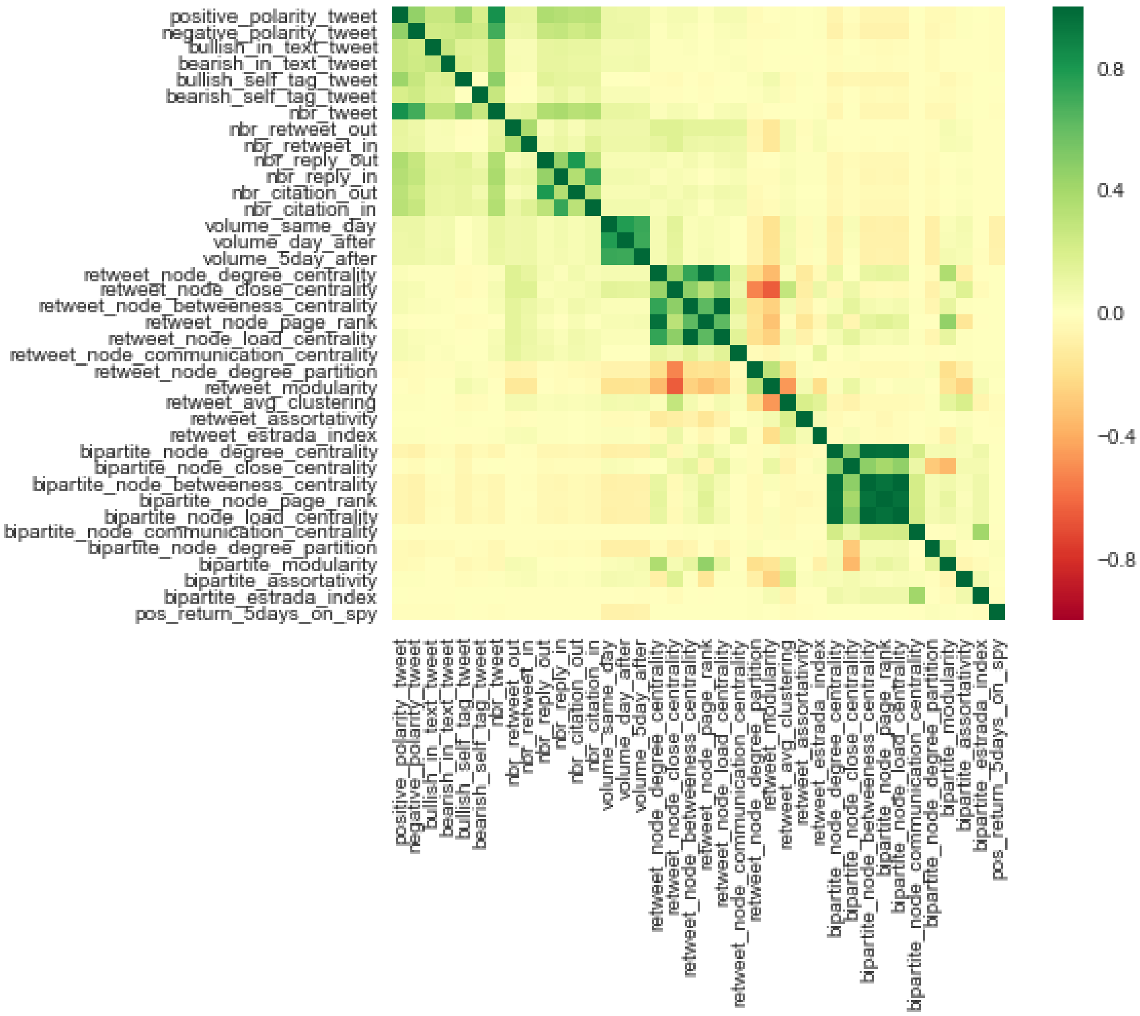


Figure 4.6: Correlation Matrix with Market Direction

Some Text and Sentiment-based features showed a weak-medium degree of correlation between them. For Instance Number of Positive polarity tweets had a correlation of 0.45 with Negative Polarity Tweets, meaning that a user in the same day can express both positive and negative feelings about a stock. While Positive polarity, obtained with Sentiment analysis, showed at least medium degree of correlation(> 0.3) with bullish feature obtained by Text Analysis, indicator of the correctness of the technique, negative polarity showed weak(> 0.1) degrees.

There were notable correlations between positive polarity and number of Reply In/Out and Citation Out (>0.36); This was true also for negative polarity but at a minor degree of correlation(> 0.1).

Bearish and Bullish in text had a weak(> 0.1) correlation between them; also notably was the medium correlation(> 0.3) that all the features related to replies, showed with the features related to Citations. All those features had also medium Correlation with positive polarity.
Differently than the work of Cha et al. (2010) and Sprenger et al. (2014), no significative correlation was found between number of re-tweet in and out for an user, and the target variable.

Regarding Network based features, none of them correlated with any Text-based or Sentiment-based feature, despite a low decorrelation between users' interactions graph modularity and number of retweet incoming and outgoing.
All the Network's measure of centrality, within the same graph, are strongly correlated with each other and they all decorrelate with modularity, from medium to strong intensity. This expected results, however, are coming from the definition of the measures, and the same can be said about the strong decorrelation measured between modularity and clustering.

Outcome of correlation measurement in relation with target variable and with Hypothesis H_{02}, will be discussed in the next chapter.

4.5 Models Training

The Final Dataset, before being processed by the Neural Network algorithms, faced a massive reduction of its instance, with the deletion of any rows bearing any null attributes. The actual number of instances passed from about 11.8 Milions to 5.2 Milions; the null value came only from the users who didn't take part to the graph of users interactions *G01* for the date he or she was twittering. So all the tweets the users made, when they were not Quoted, Retweeted and Replied to Someone, or when they were not Quoting, Retweeting or Replying to someone, didn't made the cut for the final experimentations.

In the figure 4.7, the value distribution for the Binary Class that will be predicted during the classification part of the experiment, with a good balance between the two, and a little prevalence of those Tweets about Securities, that actually *didn't* outperform the market. As describe in table 3.3, the Binary Class was containing the difference between stock subject of the tweet returns and S&P500 returns on the same 5 day period.

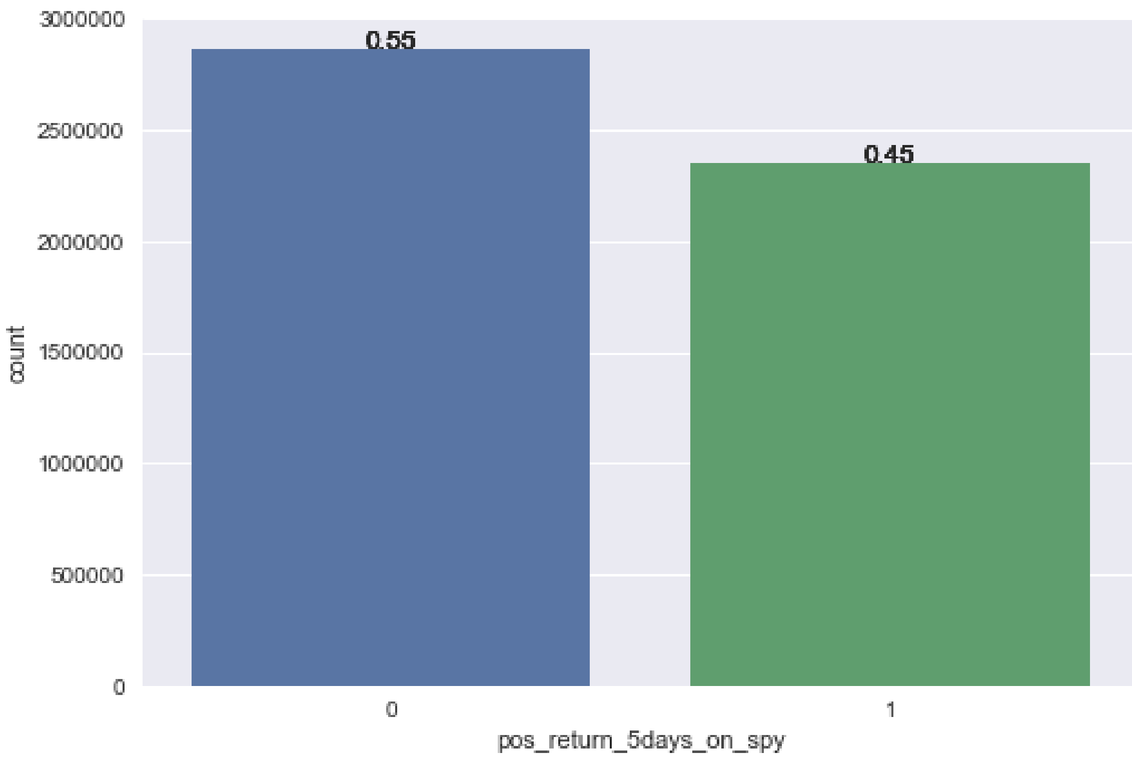


Figure 4.7: Tweets per Market Directions

In the figure 4.8, the Histograms of the difference between percentage returns of the stocks subject of Tweets and the percentage return of S&P500. Returns are measured in 5 days period, and populate the variable to predict by regression phase; most of the values are around the 0.

The Graph shows a slightly greater incidence of the negative region, with more bins left of the zero. It's worth to mention that few outliers were removed before ingestion:

they were probably due to the *Yahoo!* API error, used to retrieve financial values. The Histogram is anyway sheding light on the inclination of the actors to tweet about stocks having a mean returns over the S&P500 negative, on average around -0.6%.

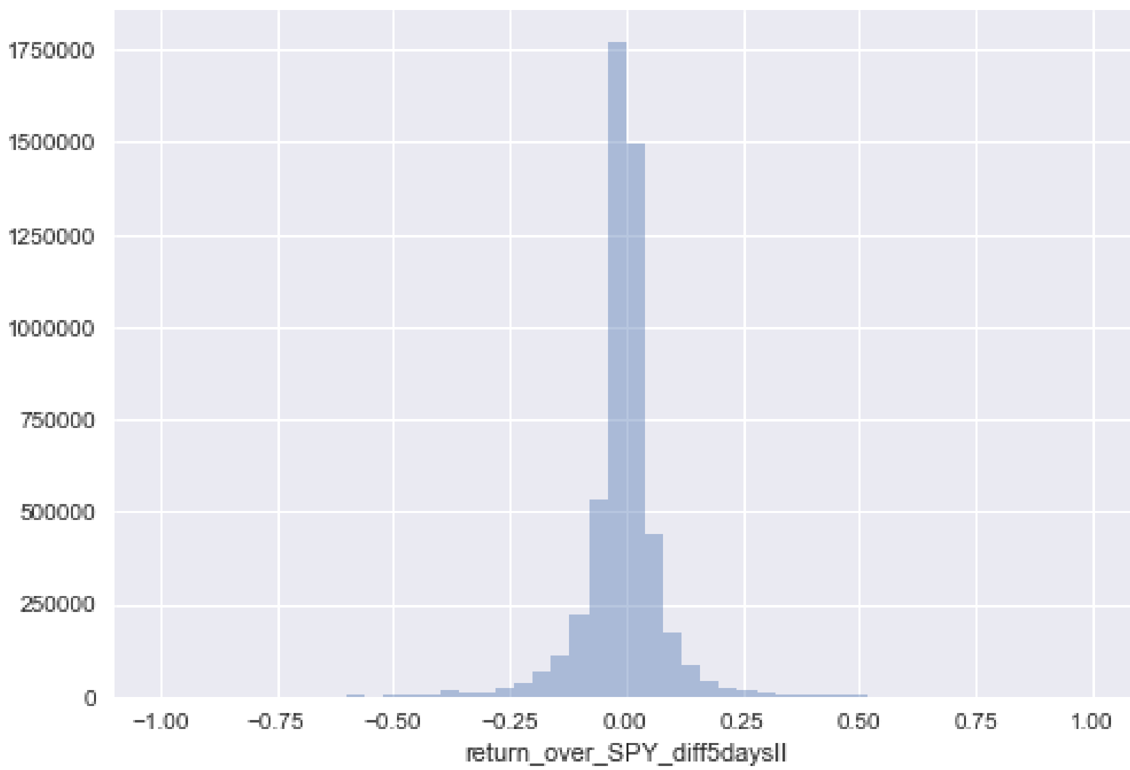


Figure 4.8: Histogram of Market Tweets Returns

4.5.1 Regressor Training and Results

The present work was aiming to test the effectiveness of ANN on Micro-Blog data, both as predicting returns and directions of the Market, with the purpose of rank the investors. Keras Classifiers and Regressors, however didn't yield the same successful outcome. Any try to tune and have the Regressor converging to values with acceptable Mean Squared Error, proved to be unsuccessful.

Other tests carried out on the single percentage stocks return, were unsuccessful to show acceptable results; a Regressor build targeting only the returns of S&P500, however, had good results, such as a value of 0.17 Mean Squared Error for the model *ANN01*, trained on all the instance, and a MSE ranging from 0.006 to 0.15 for the top

100 Investors.

In the Figure 4.9, a learning curve for model based on an user in the top 100, with a 100 Epochs training. The Mean Square Error on Train dataset, in red, decreased constantly, while the Mean Square Error for the test part, in blue, reached asymptotically a minimum. It's worth to mention that results weren't so good for most of the users.

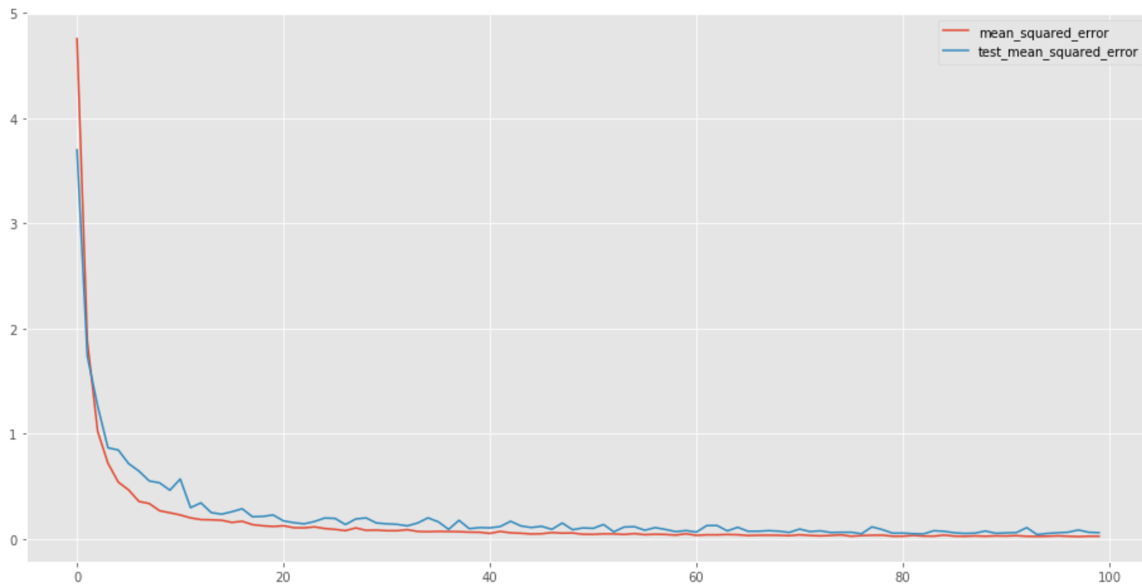


Figure 4.9: Learning the Regressor Model

Similar tests to those carried over on the Classifier Predictions were executed, yielding identical outcome; from now on however, will be described results from classifier only, since they were the foundation of the Trading Schema proposed in the present research.

4.5.2 Model tuning

the Artificial Neural Network topology was selected and tuned with a sample of 50 thousands rows from the final dataset; those rows have been not used any further in the research. All the possible Hyper-parameters have been grid-searched, with a cross fold validation consisting of 4 distinct folds; a limited number, but a good trade-off to achieve a consistent test in feasible amount of time.

As first, the activation function was tested keeping a first entry single layer of 38 nodes, the same number of the attributes fed to the algorithm, a hidden layer of 20 nodes, along with the final layer that was constantly kept a single node with sigmoid function for the classifier, and a basic node for the regressor, as experimented and suggested in Di Persio and Honchar (2016).

	Accuracy	**Standard Deviation**
Linear	0.505421	0.013563
Softmax	0.524840	0.000313
Softplus	0.532364	0.025631
Softsign	0.503351	0.001789
Relu	*0.544359*	*0.001251*
Tanh	0.543121	0.007709
Sigmoid	0.532921	0.004598
Hard Sigmoid	0.529545	0.010451

Table 4.1: Activation Function Selection

The Rectified Linear Unit(ReLU) function got an edge here, as in most of the recent experiment on Neural Network, as summarised in Di Persio and Honchar (2016). This function, that has found an increased adoption in the last few years, is computed in the following way:

$$f(x) = max(0, x) \tag{4.1}$$

This means that activation is linear above 0, and thresholded at zero for values below. Interesting to note also the low values for Standard deviations.

the optimiser tasked to minimise the loss function was tested next, keeping the same Topology and with ReLU activation function. The loss function was cross-binary entropy for classification and Mean Squared Error for regression.

	Accuracy	**Standard Deviation**
SGD	0.545420	0.011999
RMSprop	0.554540	0.000616
Adagrad	0.564160	0.004835
Adadelta	0.576640	0.001772
Adam	*0.578160*	*0.003258*
Adamax	0.575700	0.005503
Nadam	0.553020	0.002999

Table 4.2: Optimiser Selection

The best result were achieved with an Adam optimiser (Kingma & Ba, 2014), and that was used for the final model; subsequently combination Batch size and the number of Epochs was tested, where for number of Epochs is meant as the number of times the dataset is going back and forth through the ANN model to establish the best combination of weights. Results are shown in table 4.3

Table 4.3: Epochs number Selection

	Accuracy	**Standard Deviation**
epochs: 10 **batch_size**: 10	0.567260	0.007010
epochs: 50 **batch_size**: 10	0.571780	0.001779
epochs: 100 **batch_size**: 10	0.571560	0.007348
epochs: 10 **batch_size**: 20	0.570140	0.003117
epochs: 50 **batch_size**: 20	0.578600	0.009287
epochs: 100 **batch_size**: 20	*0.580480*	*0.003708*
epochs: 10 **batch_size**: 40	0.567480	0.005335
epochs: 50 **batch_size**: 40	0.578240	0.000439
epochs: 100 **batch_size**: 40	0.573580	0.007139
epochs: 10 **batch_size**: 60	0.567800	0.002418
epochs: 50 **batch_size**: 60	0.577680	0.002986

Continuation of Table 4.3		
	Accuracy	**Standard Deviation**
epochs: 100 **batch_size**: 60	0.573980	0.004517
epochs: 10 **batch_size**: 80	0.563660	0.003462
epochs: 50 **batch_size**: 80	0.573220	0.003037
epochs: 100 **batch_size**: 80	0.571540	0.006726
epochs: 10 **batch_size**: 100	0.566040	0.001462
epochs: 50 **batch_size**: 100	0.578940	0.002231
epochs: 100 **batch_size**: 100	0.574020	0.000444

By experimentation, it was evident that increasing the number of epochs had a beneficial effects, so the final model was designed to have 20 batches size, meant as number of instances processed per time, and 500 epochs. Higher numbers of the epochs were considered hard to manage in terms of duration of the learning, for the potential benefit brought.

Further Test were conducted, to find the optimal topology of the ANN, following advice by Chollet et al. (2015): the number of nodes in the first layer was raised till a beneficial effect was measured; then a new layer was introduced, and its nodes were let grown till an improvement was measured. Results are shown in table 4.4.

$\text{Accuracy}_{\text{Std}}$	**Layer 1**	**Layer 2**	**Layer 3**
nodes: 20	$0.570480_{0.003}$	$0.6108984_{0.003}$	$0.614532_{0.001}$
nodes: 50	$0.585494_{0.004}$	$0.620105_{0.006}$	$0.608956_{0.01}$
nodes: 100	$0.605696_{0.01}$	*$0.633287_{0.02}$*	$0.608467_{0.02}$
nodes: 150	$0.621988_{0.009}$	$0.618204_{0.03}$	$0.568687_{0.01}$
nodes: 200	$0.587612_{0.001}$	$0.601303_{0.001}$	$0.540898_{0.001}$

Table 4.4: Layer Selection

From the evidence, Improvement stopped after adding two layers to the entry one, and a third one was not needed; optimal configuration was reached with 150 nodes on the first hidden layer, and 100 on the second hidden one.

Kingma and Ba (2014) advice against testing other Hyperparameter, such as *learning rate* and *decay*; test performed found a severe decrease of performance in that sense. The variant AMSGrad, as introduced by Reddi, Kale, and Kumar (2018), has not been applied.

4.5.3 Final Neural Network Topology

The final network topology appears as in the figure 4.10: it's a Multi-Layer Perceptron with all the layers dense, where the number of input layers matches the number of features in the dataset processed. The output layer ends with a single node minimising the binary cross-entropy for classification, or Mean Square Error for regression.

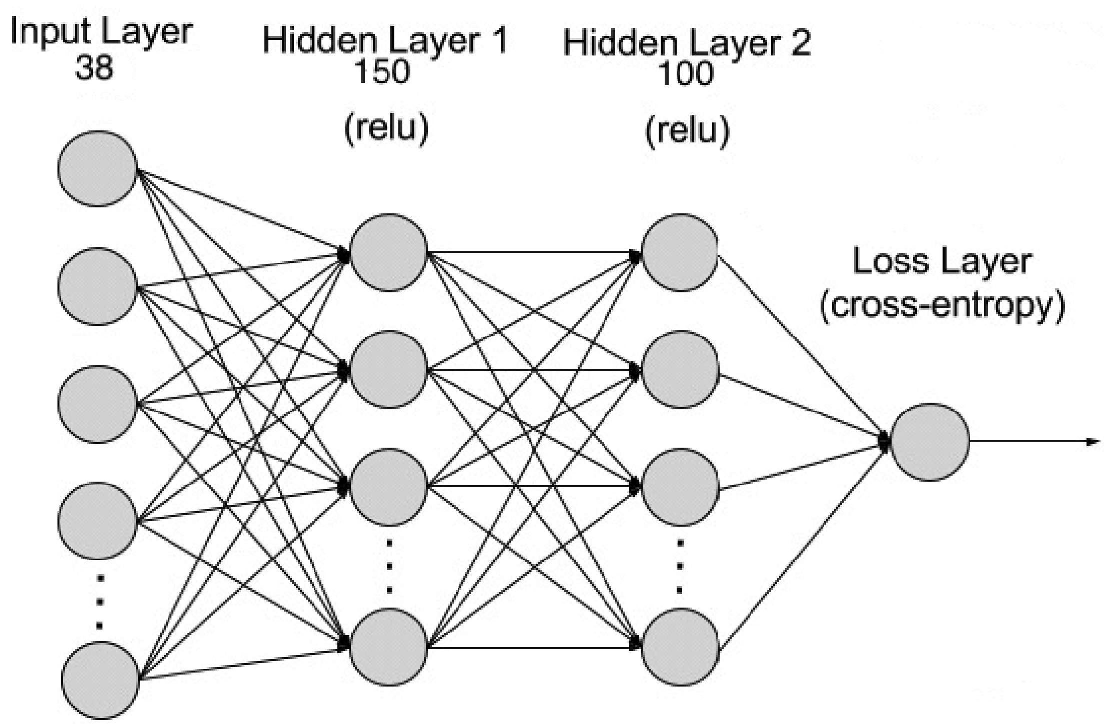


Figure 4.10: Neural Network Topology

4.5.4 Classifier Learning and Results

This section will focus on Classifier's result, since it was the foundational model for the Trading Schema. From the final dataset, all the instances with null values were dropped, decreasing the initial number of about 11 Millions of rows to just 5.5; it's important also to mention at this stage that multiple instances could potentially belong to a single tweet, since it could be tagged with multiple securities.
The results achieved for the model *ANN01*, trained on all the users for 500 epochs, are reported in table 4.5. The training was based on optimising accuracy, more than F1 score or Precision and Recall. This because of the greater value of classifying correctly the *alpha*.

ANN01	**Max Test Value**	**Max Train Value**
Accuracy	0.597	0.623
F1 score	0.303	0.342
Precision	0.559	0.599
Recall	0.231	0.264

Table 4.5: ANN01 Classification Results

After each epoch, the model trained was retained if it was scoring the best test accuracy so far, in order to be used for predictions. It's notable that results achieved on such big amount of data, were superseded by those achieved during the tuning phase, were a smaller random number of rows were used.

Results for the series of model *ANN02* and *ANN03*, trained on every single user on two different time frame, differed greatly from *ANN01*, trained on the mass, as in the paper from T. Wang et al. (2017) . First of all, more than 87 thousands distinct users logged into *StockTwits.com* platform during the 6 years period. Training models on activity of every single user however, requested to set up a threshold on them based on the number of instances each was producing; user with less instances were not trained, to not introduce bias or noise in the final results. The cut-off value has then been set

up to 380 instances, equal to 10 instances for each distinct feature of the dataset to train.(Raudys et al., 1991)
This reduced the training of *ANN02* to 1322 distinct actors, and the one of *ANN03* to 609.

Tables 4.6 and 4.7 present the result of the series of model *ANN02* improving results presents in other work in literature, such as Bar-Haim et al. (2011), where top 20 users don't score more than 0.65 and 0.54 accuracy. Best Results for users were in the same rage of those retrieved by T. Wang et al. (2017), with their expert users classifier.

ANN02	**Max Test Accuracy**	**Max Test F1 score**	**Max Test Precision**	**Max Test Recall**
Mean	0.613	0.533	0.552	0.633
Std	0.055	0.061	0.071	0.09
Min	0.503	0.086	0.263	0.057
25%	0.574	0.498	0.505	0.576
50%	0.601	0.535	0.547	0.628
75%	0.644	0.572	0.593	0.683
Max	0.886	0.758	0.806	0.961

Table 4.6: ANN02 Classification Test from 2010 to 2014

ANN02	**Max Train Accuracy**	**Max Train F1 score**	**Max Train Precision**	**Max Train Recall**
Mean	0.951	0.946	0.963	0.959
Std	0.047	0.056	0.048	0.055
Min	0.544	0.078	0.277	0.048
25%	0.934	0.926	0.946	0.943
50%	0.965	0.960	0.979	0.976
75%	0.985	0.983	0.995	0.995
Max	1	1	1	1

Table 4.7: ANN02 Classification Train from 2010 to 2014

Tables 4.8 and 4.9 present the result of the series of model *ANN03*.

ANN03	**Max Test Accuracy**	**Max Train F1 score**	**Max Test Precision**	**Max Test Recall**
Mean	0.62	0.544	0.576	0.624
Std	0.054	0.060	0.068	0.093
Min	0.497	0.345	0.362	0.323
25%	0.580	0.504	0.532	0.574
50%	0.612	0.544	0.574	0.619
75%	0.65	0.583	0.616	0.681
Max	0.842	0.760	0.800	0.927

Table 4.8: ANN03 Classification Test 2015

ANN03	**Max Train Acuracy**	**Max Train F1 score**	**Max Train Precsion**	**Max Train Recall**
Mean	0.943	0.947	0.953	0.952
Std	0.058	0.065	0.058	0.063
Min	0.662	0.484	0.638	0.377
25%	0.926	0.917	0.937	0.934
50%	0.961	0.956	0.974	0.974
75%	0.981	0.981	0.994	0.995
Max	1	1	1	1

Table 4.9: ANN03 Classification Train 2015

In the picture 4.11, an example of the Learning curve obtained for a user in the top 100 performers, and limited to 100 Epochs. Accuracy on the training dataset was improving with the number of epochs, faster at the beginning but then slower, with little or no benefit after 50 epochs.
A mirrored behaviour was observable in the loss function, *binary cross-entropy*, that was minimised by the algorithm. For the accuracy found on test dataset, was observable a different behaviour: it tended to reach a maximum value, and after reaching that, it decayed slowly. This is reflected in the loss function for the test and it could potentially be used to stop the execution of the algorithm, implementing a callback mechanism, to finish the learning if the test accuracy wasn't improving in a predefined number of epoch.
It's worth to mention that learning curve was peculiar to each user: for some of them decay was stronger, other reached best value of the accuracy at end of 500 epochs, other sooner.

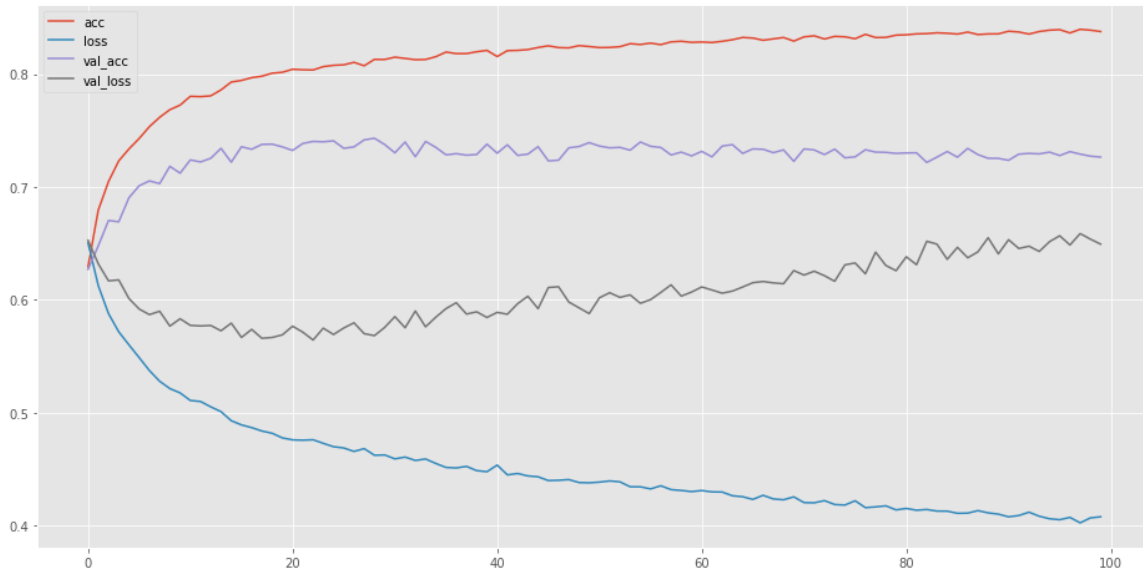


Figure 4.11: Classification Results example

The Keras Script that iterated the different Training Schema, was programmed in a way to retain the best performing models during the training, meant as those with best test accuracy, so effect of Test accuracy decay won't be observed in predictions.

Chapter 5

Evaluation and analysis

This Chapter is a review of the strength of research conducted and of the quality of the results, used for Hypotheses testing.
Conclusion will be presented on the generation of the features who enriched the final dataset, since most of them were not present in the original *JSON* blob, but derived from it, with the purpose of finding clues on the behaviour of end users, and on their relation with the Capital Markets.
Following section will be covered in this chapter:

- Results and Exploratory Data Analysis carried out in the previous Chapter will be evaluated, and its implications will be expanded. This will be done with an overview over the entire experiment and will be followed by a more focused analysis of the features under scrutiny.

- Statistical Test to reject, or fail to reject Hypotheses formulated in section 1.3 will be carried out. The significance of the results will be then outlined and discussed with respect to the existing literature.

- A Trading Schema, based on the evidence gathered in the other chapters, and on the results of the Hypotheses testing, will be introduced and measured in terms of percentage return over the capital Market.

- The final section will be then focusing over the strengths of the experiment, the findings, the weaknesses and the limitations that were encountered during the implementation, that led to possible source of bias.

At this stage is important to specify that primary metric to evaluate the results, and rank expert investors, will be the accuracy, as already in literature, like in Bar-Haim et al. (2011) and in Sohangir and Wang (2018). As written by Tosun, Aydin, and Bilgili (2016), comparing different Predicting technique in ANN, while not giving preference to any metric, such as MAE or MSE for regression problems, Accuracy can be a good choice for binary classification problem, where interest to find a single element of the class overtakes interest for the other, as it was in this case.

5.1 Text Analysis and Sentiment Mining

Results shown in section 4.2, showed a continuous optimism by tweeters: bullish self tagged tweets outnumbered bearish self tagged tweets 4 by 1 in year 2015, and this is consistent with what found by other authors, such as Dewally (2003). Consistent differences were found also in the number of Bearish and Bullish words in text, with the second present 1.5 times more, and in sentiment polarity of the tweets, where the positive polarity on daily average was twice the negative polarity.

The users seemed to react and recover positively also in case of Market crashes, like on the 2015 Black Monday, 24th of August, when Dow Index opened 1000 points down the previous closing (Denyer, 2015); in picture 5.1, users behaviour remains bullish, also in proximity of Black Monday, where we see bearish in text overtaking the bullish just for few days, and the bearish self-tags raising in volume, but then dropping quicker than the bullish ones.
Investors recognised the moment as a good one to enter into the market, in the pictures are also visible expected drop due to the weekends.

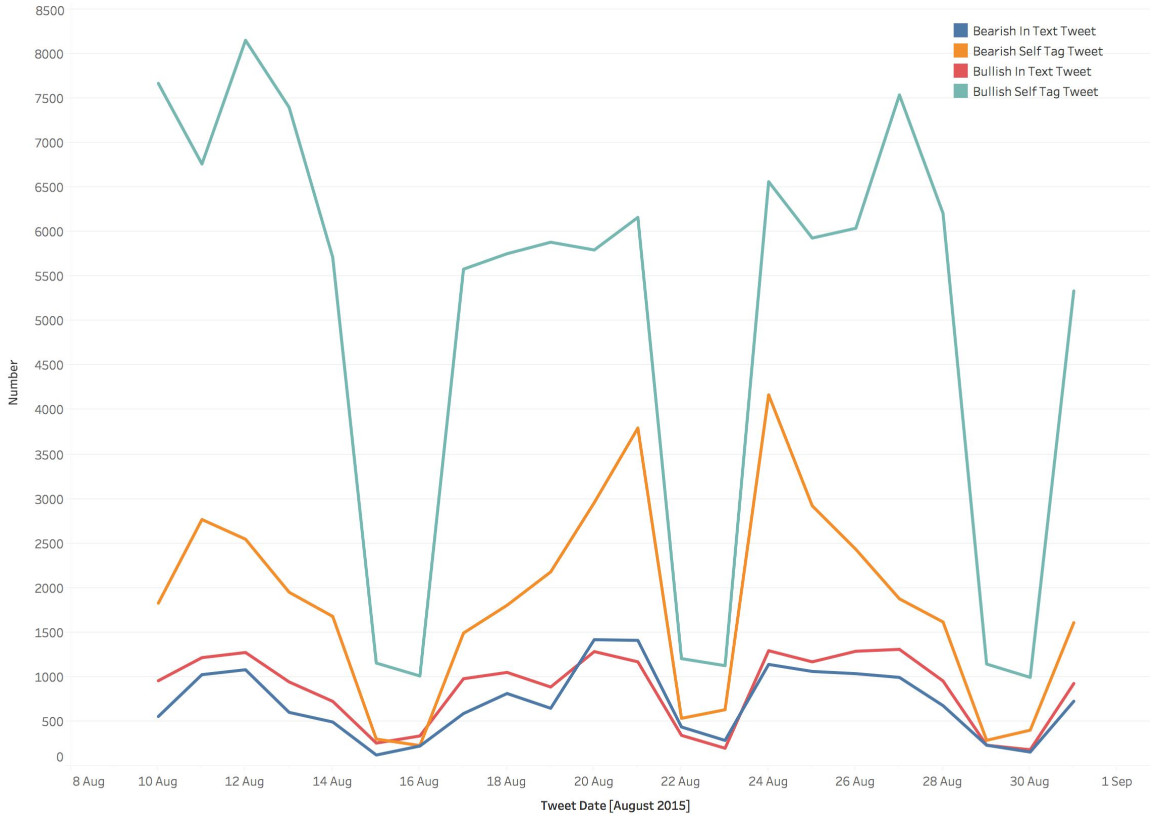


Figure 5.1: User Reactions at 2015 Black Monday

5.2 Social Network Analysis

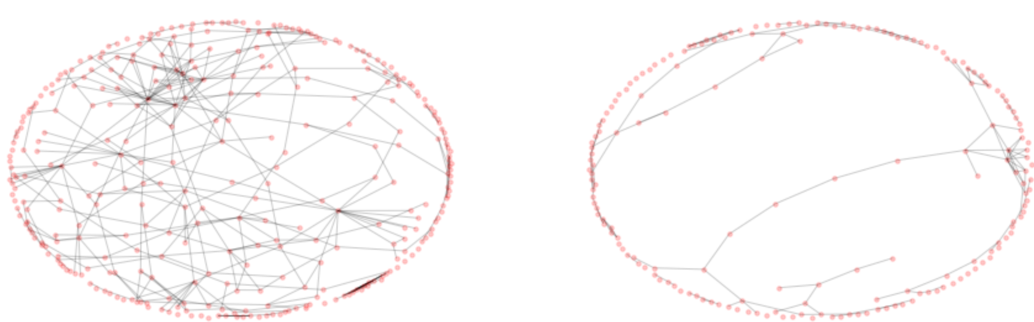
Observing the feature generated with graphs *G01* and *G02* computation, and putting them in relation with two market crashes, the falls of 1st of August 2011 (Jayech, 2016) and 24th of August 2015(Denyer, 2015), some consistent behaviours emerged. It's important to mention, however, that those 2 events were singularities in the continuous bull run that characterised years between 2010 and 2015, and were recovered in short period.

- **Increase of Number of Nodes and Edges**. As observed by Casnici et al. (2015), users participated more frequently in the network, increased their activity, and focused their joint-attention on event of sudden changes, to face the stress caused by it. For instance, in proximity of 24th of August, 2015, the number of Nodes and edges differed from 1782 nodes and 1763 edges, to 1324 nodes and 1110 edges measured in quieter conditions, 3 months later, as it is visible in the 2 snapshots represented in figure 5.2.

- **Modularity and Assortativity Drop**. In crisis period, the Graphs are less fragmented in community, but they coalesces around a single big question. Similar nodes are attaching to each other less frequently. Using the same market condition described in the point above, Modularity and Assortativity registered in a quiet market are 0.974 and 0.0247, while the same metrics, measured the 24^{th} of August 2015, are as low as 0.919 and -0.0343.

- **Average Between Centrality Increase**. As already discussed by Racca et al. (2016), some nodes tend to surge to hub role during this events, like a community gathering around its most expert and wiser members. It increased of 30% during the outburst of the 2011 Market Crisis, compared to the week before.

In the two pictures 5.2 the evident difference between a Network built on users' interaction during a Market Crisis, and the same network 3 months after, when the crisis was a past fact, can be seen also graphically. In the representation, nodes with a single degree of centrality have been cut off.



(a) 24-08-2015 (b) 24-11-2015

Figure 5.2: User Interactions Graph during and after Black Monday 2015

5.3 Correlation Analysis

As discussed in sections 5.1 and 5.2, some *Sentiment Mining* metrics correlated well with some *Text Analysis* metrics and *SNA* metrics in some case correlated with *SNA* metrics within the same graph type. The table 5.1 shows a deep dive on correlation of all the metrics with the target label; correlation extracted is extremely poor; full correlation matrix is available in section 4.4

Table 5.1: Correlation with Market directions.

Field	**correlation degree**
positive_polarity_tweet	-0.010222596
negative_polarity_tweet	-0.015462739
bullish_in_text_tweet	-0.011503922
bearish_in_text_tweet	-0.013037456
bullish_self_tag_tweet	0.001591819
bearish_self_tag_tweet	-0.007584005
nbr_tweet	-0.014631395
nbr_retweet_out	-0.000512497
nbr_retweet_in	-0.00024044
nbr_reply_out	-0.002143462
nbr_reply_in	-0.003531768
nbr_citation_out	-0.002339743
nbr_citation_in	-0.002270686
volume_same_day	-0.084279269
volume_day_after	-0.080698542
volume_5day_after	-0.074492391
retweet_node_degree_centrality	0.00421216
retweet_node_close_centrality	0.004950352
retweet_node_betweeness_centrality	0.002125584
retweet_node_page_rank	0.001869216

Continuation of Table 5.1	
Field	**correlation degree**
retweet_node_load_centrality	0.002109393
retweet_node_communication_centrality	0.000434633
retweet_node_degree_partition	0.008180217
retweet_modularity	0.003832521
retweet_avg_clustering	-0.016541348
retweet_assortativity	-0.012128715
retweet_estrada_index	-0.000465797
bipartite_node_degree_centrality	0.005427159
bipartite_node_close_centrality	0.002972054
bipartite_node_betweeness_centrality	0.00691083
bipartite_node_page_rank	0.005583399
bipartite_node_load_centrality	0.00682079377622
bipartite_node_communication_centrality	-0.00530612051078
bipartite_node_degree_partition	0.0230336601619
bipartite_modularity	0.007979003
bipartite_assortativity	-0.004458443
bipartite_estrada_index	-0.011679875
pos_return_5days_on_spy	1

Because of the results, H_{02} *must be rejected*, as largely expected from literature. (Oliveira et al., 2013b)

5.4 Predictive Algorithm Outcome Analysis

5.4.1 Investor Ranking

To compare how the investors ranked in the period 2010 - 2014 and just in 2015, a cut-off value of at least *380* observations was assumed, to create the two distinct populations. Only 492 users verified the conditions of having a sufficient number of observation in both periods of training.

Wilcoxon Signed Rank Statistical test, performed on the rankings produced from learning algorithm in the periods, brought the following outcome: a z value of 23829, with a *p value* of 0.939.
Because of a p value greater than 0.05, H_{03} *was rejected*, meaning that the mean difference between the results of the two groups is 0, and there is no statistical difference between the two rankings.
Moreover, a Kendall's τ measurement, gave a statistical significant value of 0.45 (with *p value* of 2.4e-10); meaning a medium degree of correlation between the two rankings.

5.4.2 Trading Schema

Leveraging the predictive algorithm results, an Investing Recommendation System has been put together. For the users who ranked in the top 100 for the 5 years period between 2010 and 2014, a set of forecast was produced for the financial years 2015, using models trained user by user on the 5 years windows.

The Trading Schema, partially based on the criteria proposed in the work of Ruiz et al. (2012), was composed by the following actions:

- When prediction was greater or equal to 0.5, the bid was placed on the stock related to the tweet, supposing it was beating S&P500 in 5 days timeframe.
- When prediction was lesser than 0.5, the bid was placed on S&P500, supposing the stock was not beating S&P500 in 5 days timeframe.
- A total commission fee of 0.1% was applied to users' investment, for every day of activity.

The amount used for the bid was constant for all the user across the day, so a user predicting 7 different stocks within a day, was placing the amount granted to him divided by 7. The amount was considered returned in 5 days to the the user, and the total amount invested was equals of said costant amount multiplied for the maximum number of investors active in consecutive 5 days period within the year.

Expert Investors Performance

The Top 100 Trading schema was having a return of 38.7% on the capital invested, with a standard deviation of 1.28; the baseline for comparison was a schema that was buying every time S&P500 for each predictions, no matter which one was the stocks subject of tweet, and respecting the same amount limitation. It performed sensitively worst, with a return of 3% but with an inferior Standard Deviation of 0.37%

The results measured on the baseline was consistent with the evolution of stock exchange indices; in graph 5.3, the market movement in 2015, with S&P500, generally used as principal comparison index, scoring a decrease of 0.69%, despite crossing period of great volatility, while NASDAQ index had a more consistent appreciation of 5.9% .

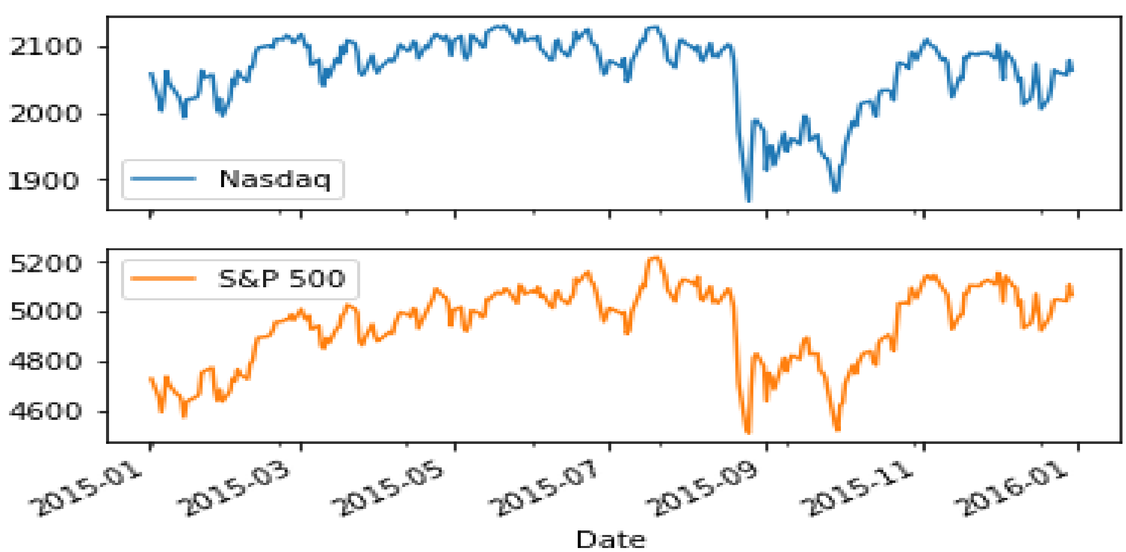


Figure 5.3: Principal Financial Indices

Parameters Tuning

Parameter of the Trading Schema went through testing of different combinations: in the chart 5.4, the percentage return is plotted against the buy signal change and the selection of top number of investors.

Incrementing the buy signal, translates into executing less transactions: a value of 0.6, for instance, means the predicted stock is bought above that value, while S&P500 is bought for less than 0.4.

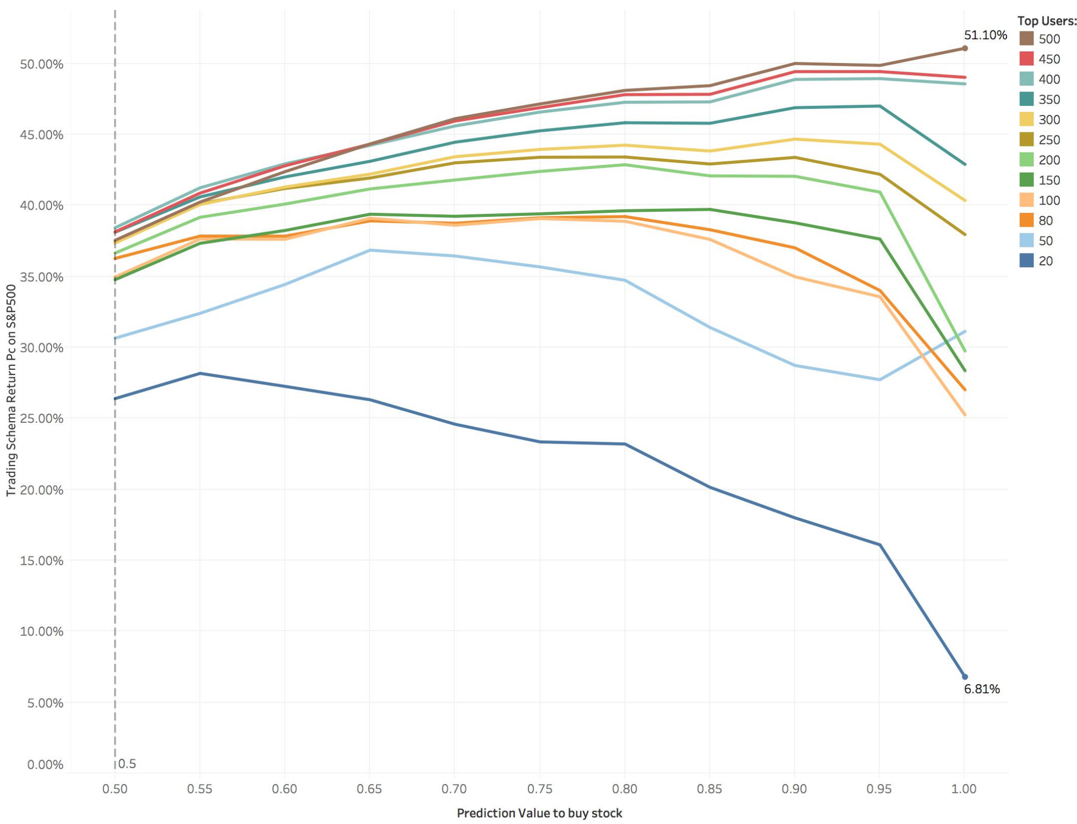


Figure 5.4: Trading Schema Return per Accuracy and Top Users

In terms of number of Top Investors, the results are showing evidences to prefer a larger number of expert investors than 100, since absolute gains is growing constantly with their number, beside rare exceptions. Incrementing the buy signal also as a positive effects on return: this effect is more evident increasing the number of users.

Chart 5.4 is a limitation of findings presented in section 5.4.1, where ranking were executed on models trained and test on separate time periods.

Wisdom of the Crowd Performance

To assess the performance of the Crowd, the trading schema has been applied 1000 times to random investors in 2015, chosen in a dataset obtained in the following way:

- Excluding Investors in the top 100 ranking between 2010 and 2014.
- Predicting market directions in 2015 with the model *ANN01* trained for all investors on years between 2010 and 2014.
- Using same number of investors and same investment capital used by top100 Trading Schema.

The total returns of the Crowd schema had moderate variance and rarely outperformed the baseline, based on buying constantly quotes of S&P500, as mentioned above. In table 5.2 some statistics on the returns generated over the S&P500 index, with the histogram of their distribution, 5.5.

Metric	**Value**
Mean	-2.300021
Standard Deviation	1.774120
Minum	-7.900811
Maximum	3.355164
25 percentile	-3.432755
50 percentile	-2.248228
75 percentile	-1.161931

Table 5.2: Summary Trading Schema applied to Crowd

It's interesting to notice that the mean value of the distribution is slightly negative, like the return of S&P500 for that year.

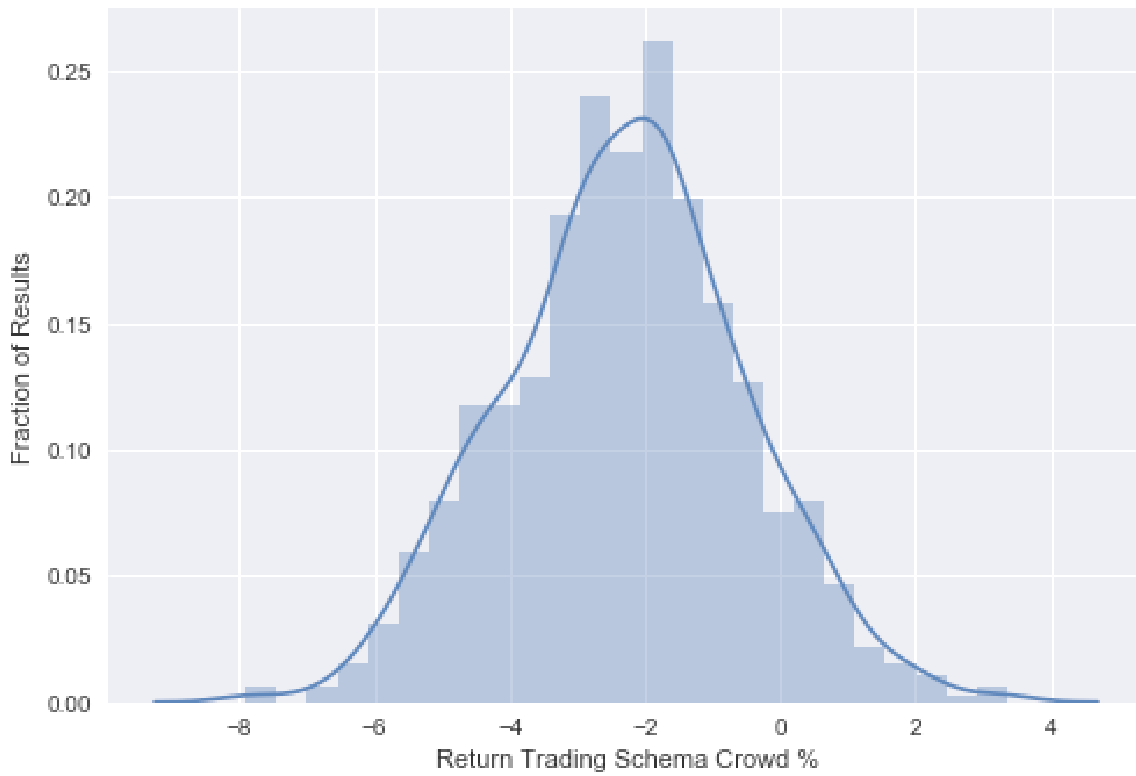


Figure 5.5: Distribution Returns over S&P500 Trading Schema applied to all

The evidence that the returns of Crowd were constantly below the returns obtained by expert Investors' schema, is the ground for rejecting hypotheses H_{01}, and also a confirmation of the Random Walk Theory.(Malkiel & McCue, 1985)

5.5 Results Limitations

Following limitations can be identified in the results:

- **Missing Convergence for Regressor**. As already mentioned, missing convergence from the regressor algorithm, pushed to the use of the Classifier for the Trading Schema, and might have led to the poor performance of model *ANN01* regarding the mass of the investors.

- **Laterality of Year 2015**. Stock year 2015 didn't present outstanding returns or the major financial indices, but it had a small decrease overall, despite some volatility was measured around August, due to the *2015 Stock Market Selloff*, caused by Shanghai Index sudden drop.(Denyer, 2015)

- **Big portion of dataset were dropped**. To let the algorithms work properly, the biggest portion of the dataset was dropped, because of no participation in the Graph of user interaction *G01* built for that day. It's worth to note, anyway, that in other work expert investors have been found to be those with more social interactions. (Sprenger et al., 2014)

- **Majority of user had few tweets**. As observed in many complex system, 80% of the effect, is triggered by 20% of the causes. Social Network don't make an exception in this sense, and small portion of the nodes is linked by the majority of the edge (Ediger, Jiang, Riedy, Bader, & Corley, 2010). Also the *StockTwits.com* final dataset presented a big portion of its users with less rows than the 380 instances needed to make the cut to the final ranking. From 87 thousands and more distinct users, only around 500 had the characteristics to be included in the Wilcoxon test on the rankings.

- **Many Instances derived from single tweets**. Since a user can tweet about more securities at once, a single tweets can generate more than one instance, and this might have possible led to unbalance in the final dataset.

- **Possible Unbalance of Target Variable at User level**. The target variable pos_return_5days_on_spy was balanced at level of the entire dataset, with a small prevalence of 0(the actors were twittering more about a stock that didn't out-perform S&P500). At single user level things might change, introducing a bias, that could only be potentially resolved by a stratified sampling(not used in this work).

- **Trading Gains not improving with top users**. Despite forecast produced by models trained on single users were definitively better than those produced training on all users, the trading schema didn't benefit of reducing the partici-pants to the better ranked.

Chapter 6

Conclusion

6.1 Research Overview

The Present work was putting together a predictive algorithm for Financial Market, based on Social Network, Sentiment and Text Analytics, elaborating a loosely structured dataset containing in the region of 37 Milions tweets, extracted from *Stock-Twits.com*.
The Dataset was cleaned and organised in structural and tabular way, and new Features were calculated, to enrich a final dataset that was feeding three different series of ANN model, based on different timeframe and repartitions of the users; the added features were based on Sentiment Polarity of the Text Message and indicators of traffic and of user activities. Moreover, features coming from Graphs generated for every day by users' interaction, and relation between users and stock, were also included.

The 3 distinct Series of Models helped to shed some light on the behaviour of the users on the market, and, in particular, whether expert investors were achieving constants results over time, in contrast from the rest of the crowd.
Finally, leveraging predictions user by user, a trading algorithm has been proposed, and evaluated in terms of its returns and compared against the returns of investment on the S&P500 financial index.

6.2 Problem Definition

In order to evaluate the research questions and deep dive in their faceted implications, it was necessary to overcome a series of problem:

- **Data Transformation of highly nested JSON blob.** The format of the initial dataset didn't allow expedite queries and join, so great part of the work was focusing in cleaning this up, and filtering out the greatest majority of tweets that didn't contribute to the final experiment.

- **Missing convergence**. The initial effort of this work, was to predict the intensity of the gain over the market, of the stocks subjects of twittering. Unfortunately, only S&P500 returns were possible to predict with an error small enough, and for this the problem was turned into a binary classification of market directions, with positive results.

- **Algorithms Performance Issues.** Graphs generation for a smaller part, and ANN for the biggest part, were impossible to handle on a single-CPU machine, so it was necessary to put together automation scripts and to upload them to a dedicated servers for computation.

- **Few significative rows per users.** Despite the large number of unique users of the platform, to train a model on each of them yielding significative results, it was necessary to filter only those that had at list 380 instances for the period of interest, and this limited diversity in the final dataset.

6.3 Design/Experimentation, Evaluation & Results

The present work was designed to verify the Hypotheses in section 1.3 via statistical testing on the evidences gathered by experimentation. Results summarise as following:

1. H_{01} *was rejected*. Accuracy measured on the best investors identified on *StockTwits.com* differed significantly from the mass of the crowd. These results was used to design a Trading Strategy.

2. H_{02} *failed to be rejected.* No notable degree of correlation was measured between any of Financial Micro-Blog features, existing or engineered, and the market direction.

3. H_{03} *was rejected.* Statistical Results showed that Investors that performed among the best in the 5 years period between 2010 and 2014, continued to achieve comparable rankings also in the trading year of 2015.

6.4 Contributions and impact

Original contribution to the rich body of knowledge of Stock Forecasting by Social Network can be synthesised as follow:

- Exploratory Data Analysis via Big Data Technologies of the characteristics of a Financial Microblog, *StockTwits.com* and its evolution along the time. A volume such as 37 Millions of tweets is massive compared to literature of the previous year, where the focus was ranging from thousands of rows(Bar-Haim et al., 2011) to hundred of thousands of rows (Casnici et al., 2015).

- Study of relation between user of a Financial Social Network via graph representation, and shedding some light on how the network metrics can be used in distinguish *alpha* investors.

- Adaptation of a Machine learning algorithm based on ANN to financial mixed features, engineered via techniques belonging to different domain of expertise.

6.5 Future Work & Recommendations

Following Recommendations can be done for future work, to improve the results here presented or to overcome its bias:

- **Describing Expert Investors by Features:** In the present work no systematic investigation has been taken on the characteristics that differentiate expert investors from the others within the network, like in Sprenger et al. (2014), where expert investors where found having a greater number of retweets. Moreover, no measure of correlation was carried out with specific focus on expert investors; the works of Sprenger et al. (2014) and T. Wang et al. (2017) suggested it might differ in a notable way from the crowd. Other possible analysis includes the role of the dimension of the message (Bakshy et al., 2011).

- **Comparison Index:** To Assess performance of user's Predictions, all the stocks were compared against S&P500, that had returns close to 0 that year. Would have been more realistic to compare them against a specific Index, reflecting the characteristic of the Company, or where the Company was listed. For instance NASDAQ for AAPL, and so on.

- **Bear Market:** The period were the algorithms were trained and tested, going from 1st of January 2010 and 31st December 2015, was a continuous bull-run that rarely can be matched with any past period. To create a more resilient Trading Schema, confrontation with a Bear Market, like the one that Impacted US Stock Exchanges between 2007 and 2008 would be necessary.

- **Analysis per Stock:** With such massive amount of stocks and tweets, analysis can be focused just on specific stocks, and see how users' tweets are moving the market, how long the information of their activity decays into noise, and how stocks and users are influencing each others.

- **Time Series:** In stock forecasting, the value in a moment depends on previous performance. Reducing the prediction of a single security to a time series will require a non stationary transformation, and the usage of ARIMA models.

- **Long Short Term Memory Network:** as employed by Chen, Zhou, and Dai (2015), where this kind of Neural Network has been used to predict China Securities behaviour after headlines in news, a similar techniques can be employed on the dataset, after transforming it into a Time Series.

References

Adamic, L. A., Zhang, J., Bakshy, E., & Ackerman, M. S. (2008). Knowledge sharing and yahoo answers: everyone knows something. In *Proceedings of the 17th international conference on world wide web* (pp. 665–674).

Al Nasseri, A., Tucker, A., & de Cesare, S. (2015). Quantifying stocktwits semantic terms? trading behavior in financial markets: An effective application of decision tree algorithms. *Expert Systems with Applications*, *42*(23), 9192–9210.

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

Bagnoli, M., Beneish, M. D., & Watts, S. G. (1999). Whisper forecasts of quarterly earnings per share. *Journal of Accounting and Economics*, *28*(1), 27–50.

Bakshy, E., Hofman, J. M., Mason, W. A., & Watts, D. J. (2011). Everyone's an influencer: quantifying influence on twitter. In *Proceedings of the fourth acm international conference on web search and data mining* (pp. 65–74).

Bar-Haim, R., Dinur, E., Feldman, R., Fresko, M., & Goldstein, G. (2011). Identifying and following expert investors in stock microblogs. In *Proceedings of the conference on empirical methods in natural language processing* (pp. 1310–1319).

Bartov, E., Faurel, L., & Mohanram, P. (2017). Can twitter help predict firm-level earnings and stock returns? *The Accounting Review*.

Bavelas, A. (1950). Communication patterns in task-oriented groups. *The Journal of the Acoustical Society of America*, *22*(6), 725–730.

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

Brooks, C. (1998). Predicting stock index volatility: can market volume help? *Journal of Forecasting*, *17*(1), 59–80.

Casnici, N., Dondio, P., Casarin, R., & Squazzoni, F. (2015). Decrypting financial markets through e-joint attention efforts: on-line adaptive networks of investors in periods of market uncertainty. *PLoS one*, *10*(8), e0133712.

Cha, M., Haddadi, H., Benevenuto, F., & Gummadi, P. K. (2010). Measuring user influence in twitter: The million follower fallacy. *Icwsm*, *10*(10-17), 30.

Chapman, P., Clinton, J., Kerber, R., Khabaza, T., Reinartz, T., Shearer, C., & Wirth, R. (2000). Crisp-dm 1.0 step-by-step data mining guide.

Chee, K. H. (2017). Twitter cashtags and sentiment analysis in predicting stock price movements.

Chen, K., Zhou, Y., & Dai, F. (2015). A lstm-based method for stock returns prediction: A case study of china stock market. In *Big data (big data), 2015 ieee international conference on* (pp. 2823–2824).

Chollet, F., et al. (2015). *Keras.* `https://keras.io`.

Curme, C., Preis, T., Stanley, H. E., & Moat, H. S. (2014). Quantifying the semantics of search behaviour before stock market moves. *Proceedings of the National Academy of Sciences*, *111*(32), 11600–11605.

de la Peña, J. A., Gutman, I., & Rada, J. (2007). Estimating the estrada index. *Linear Algebra and its Applications*, *427*(1), 70–76.

De Meo, P., Ferrara, E., Fiumara, G., & Provetti, A. (2011). Generalized louvain method for community detection in large networks. In *Intelligent systems design and applications (isda), 2011 11th international conference on* (pp. 88–93).

Denyer, S. (2015). China's "black monday" spreads stock market fears worldwide. *Washington Post*.

Dewally, M. (2003). Internet investment advice: Investing with a rock of salt. *Financial Analysts Journal*, *59*(4), 65–77.

Dimpfl, T., & Jank, S. (2016). Can internet search queries help to predict stock market volatility? *European Financial Management*, *22*(2), 171–192.

Di Persio, L., & Honchar, O. (2016). Artificial neural networks architectures for stock price prediction: Comparisons and applications. *International Journal of Circuits, Systems and Signal Processing*, *10*, 403–413.

Dondio, P. (2012). Predicting stock market using online communities raw web traffic. In *Web intelligence and intelligent agent technology (wi-iat), 2012 ieee/wic/acm international conferences on* (Vol. 1, pp. 230–237).

Dondio, P., Barrett, S., Weber, S., & Seigneur, J. M. (2006). Extracting trust from domain analysis: A case study on the wikipedia project. In *International conference on autonomic and trusted computing* (pp. 362–373).

Dondio, P., & Longo, L. (2011). Trust-based techniques for collective intelligence in social search systems. In *Next generation data technologies for collective computational intelligence* (pp. 113–135). Springer.

Dridi, A., Atzeni, M., & Recupero, D. R. (2018). Finenews: fine-grained semantic sentiment analysis on financial microblogs and news. *International Journal of Machine Learning and Cybernetics*, 1–9.

Du, Q., Hong, H., Wang, G. A., Wang, P., & Fan, W. (2017). Crowdiq: A new opinion aggregation model. In *Proceedings of the 50th hawaii international conference on system sciences.*

Ediger, D., Jiang, K., Riedy, J., Bader, D. A., & Corley, C. (2010). Massive social network analysis: Mining twitter for social good. In *Parallel processing (icpp), 2010 39th international conference on* (pp. 583–593).

Estrada, E., & Rodriguez-Velazquez, J. A. (2005). Subgraph centrality in complex networks. *Physical Review E*, *71*(5), 056103.

Fama, E. F. (1970). Efficient capital markets: A review of theory and empirical work. *The journal of Finance*, *25*(2), 383–417.

Fama, E. F. (1991). Efficient capital markets: Ii. *The journal of finance*, *46*(5), 1575–1617.

Felton, J., & Kim, J. (2002). Warnings from the enron message board. *The Journal of Investing*, *11*(3), 29–52.

Gilbert, C. H. E. (2014). Vader: A parsimonious rule-based model for sentiment analysis of social media text. In *Eighth international conference on weblogs and social media (icwsm-14). available at (20/04/16) http://comp. social. gatech. edu/papers/icwsm14. vader. hutto. pdf.*

Holland, P. W., & Leinhardt, S. (1971). Transitivity in structural models of small groups. *Comparative Group Studies*, *2*(2), 107–124.

Huang, A. (2016). Data mining twitter to model stock price movement. In *Proceedings of the the 3rd multidisciplinary international social networks conference on socialinformatics 2016, data science 2016* (p. 32).

Jayech, S. (2016). The contagion channels of july–august-2011 stock market crash: A dag-copula based approach. *European Journal of Operational Research*, *249*(2), 631–646.

Jones, A. L. (2006). Have internet message boards changed market behavior? *Info-The journal of policy, regulation and strategy for telecommunications*, *8*(5), 67–76.

Kendall, M. G. (1955). Rank correlation methods.

Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. *CoRR*, *abs/1412.6980*. Retrieved from `http://arxiv.org/abs/1412.6980`

Le, Q., & Mikolov, T. (2014). Distributed representations of sentences and documents. In *International conference on machine learning* (pp. 1188–1196).

Liu, Y., Qin, Z., Li, P., & Wan, T. (2017). Stock volatility prediction using recurrent neural networks with sentiment analysis. In *International conference on industrial, engineering and other applications of applied intelligent systems* (pp. 192–201).

Loria, S., Keen, P., Honnibal, M., Yankovsky, R., Karesh, D., Dempsey, E., et al. (2014). Textblob: simplified text processing. *Secondary TextBlob: Simplified Text Processing*.

Malkiel, B. G. (2003). The efficient market hypothesis and its critics. *Journal of economic perspectives*, *17*(1), 59–82.

Malkiel, B. G., & McCue, K. (1985). *A random walk down wall street* (Vol. 8). Norton New York.

Mao, H., Counts, S., & Bollen, J. (2011). Predicting financial markets: Comparing survey, news, twitter and search engine data. *arXiv preprint arXiv:1112.1051*.

Mao, Y., Wei, W., & Wang, B. (2013). Twitter volume spikes: analysis and application in stock trading. In *Proceedings of the 7th workshop on social network mining and analysis* (p. 4).

McNamara, S. (2016). ? identifying market indicators and content quality from a financial micro-blog platform.

Miller, G. A. (1995). Wordnet: a lexical database for english. *Communications of the ACM*, *38*(11), 39–41.

Newman, M. (2010). *Networks: an introduction*. Oxford university press.

Newman, M. E. (2001). Scientific collaboration networks. ii. shortest paths, weighted networks, and centrality. *Physical review E*, *64*(1), 016132.

Newman, M. E. (2002). Assortative mixing in networks. *Physical review letters*, *89*(20), 208701.

Newman, M. E. (2006). Modularity and community structure in networks. *Proceedings of the national academy of sciences*, *103*(23), 8577–8582.

Oliveira, N., Cortez, P., & Areal, N. (2013a). On the predictability of stock market behavior using stocktwits sentiment and posting volume. In *Portuguese conference on artificial intelligence* (pp. 355–365).

Oliveira, N., Cortez, P., & Areal, N. (2013b). Some experiments on modeling stock market behavior using investor sentiment analysis and posting volume from twitter. In *Proceedings of the 3rd international conference on web intelligence, mining and semantics* (p. 31).

Page, L., Brin, S., Motwani, R., & Winograd, T. (1998). The pagerank citation ranking: Bringing order to the web.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... others (2011). Scikit-learn: Machine learning in python. *Journal of machine learning research*, *12*(Oct), 2825–2830.

Racca, P., Casarin, R., Squazzoni, F., & Dondio, P. (2016). Resilience of an online financial community to market uncertainty shocks during the recent financial crisis. *Journal of Computational Science*, *16*, 190–199.

Randles, R. H. (1988). Wilcoxon signed rank test. *Encyclopedia of statistical sciences*.

Raudys, S. J., Jain, A. K., et al. (1991). Small sample size effects in statistical pattern recognition: Recommendations for practitioners. *IEEE Transactions on pattern analysis and machine intelligence*, *13*(3), 252–264.

Reddi, S. J., Kale, S., & Kumar, S. (2018). On the convergence of adam and beyond. In *International conference on learning representations.* Retrieved from https://openreview.net/forum?id=ryQu7f-RZ

Roy, R. B., & Sarkar, U. K. (2011). Identifying influential stock indices from global stock markets: A social network analysis approach. *Procedia Computer Science*, *5*, 442–449.

Ruiz, E. J., Hristidis, V., Castillo, C., Gionis, A., & Jaimes, A. (2012). Correlating financial time series with micro-blogging activity. In *Proceedings of the fifth acm international conference on web search and data mining* (pp. 513–522).

Sankar, C. P., Vidyaraj, R., & Kumar, K. S. (2015). Trust based stock recommendation system–a social network analysis approach. *Procedia Computer Science*, *46*, 299–305.

Sohangir, S., Petty, N., & Wang, D. (2018). Financial sentiment lexicon analysis. In *Semantic computing (icsc), 2018 ieee 12th international conference on* (pp. 286–289).

Sohangir, S., & Wang, D. (2018). Finding expert authors in financial forum using deep learning methods. In *Second IEEE international conference on robotic computing, IRC 2018, laguna hills, ca, usa, january 31 - february 2, 2018* (pp. 399–402). IEEE Computer Society. Retrieved from `https://doi.org/10.1109/IRC.2018.00082` doi: 10.1109/IRC.2018.00082

Sprenger, T. O., Tumasjan, A., Sandner, P. G., & Welpe, I. M. (2014). Tweets and trades: The information content of stock microblogs. *European Financial Management*, *20*(5), 926–957.

Surowiecki, J. (2004). The wisdom of crowds: Why the many are smarter than the few and how collective wisdom shapes business. *Economies, Societies and Nations*, *296*.

Tetlock, P. C., Saar-Tsechansky, M., & Macskassy, S. (2008). More than words: Quantifying language to measure firms' fundamentals. *The Journal of Finance*, *63*(3), 1437–1467.

Tosun, E., Aydin, K., & Bilgili, M. (2016). Comparison of linear regression and artificial neural network model of a diesel engine fueled with biodiesel-alcohol mixtures. *Alexandria Engineering Journal*, *55*(4), 3081–3089.

Tumarkin, R., & Whitelaw, R. F. (2001). News or noise? internet postings and stock prices. *Financial Analysts Journal*, *57*(3), 41–51.

Wang, G., Gill, K., Mohanlal, M., Zheng, H., & Zhao, B. Y. (2013). Wisdom in the social crowd: an analysis of quora. In *Proceedings of the 22nd international conference on world wide web* (pp. 1341–1352).

Wang, G., Wang, T., Wang, B., Sambasivan, D., Zhang, Z., Zheng, H., & Zhao, B. Y. (2015). Crowds on wall street: Extracting value from collaborative investing platforms. In *Proceedings of the 18th acm conference on computer supported cooperative work & social computing* (pp. 17–30).

Wang, T., Wang, G., Wang, B., Sambasivan, D., Zhang, Z., Li, X., … Zhao, B. Y. (2017). Value and misinformation in collaborative investing platforms. *ACM Transactions on the Web (TWEB)*, *11*(2), 8.

Watts, D. J., & Strogatz, S. H. (1998). Collective dynamics of "small-world" networks. *nature*, *393*(6684), 440.

Watts, J., Anderson, N., Asbill, C., & Mehr, J. (2017). Analyzing microblog feeds to trade stocks.

Weng, J., Lim, E.-P., Jiang, J., & He, Q. (2010). Twitterrank: finding topic-sensitive influential twitterers. In *Proceedings of the third acm international conference on web search and data mining* (pp. 261–270).

Wolfram, M. S. A. (2010). Modelling the stock market using twitter. *Scotland, UK: University of Edinburgh*.

Wysocki, P. (1998). Cheap talk on the web: The determinants of postings on stock message boards.