CHAPTER 1

1. Introduction

An important feature of financial market structure is the trading volume. Volume and its changes not only play an important role in better understanding the financial markets, but also are an important component of any algorithmic strategy used to trade and manage portfolios.

The volume of trade consists of the total number of shares of contracts of a security exchanged between buyers and sellers in a specified time frame, i.e. it would include every share that is bought and sold during the time period considered.

Volume is used as an indicator to measure the market activity and liquidity over a given period and usually higher trading volumes can be regarded as more positive than lower trading volume since they imply more liquidity and better order execution.

In view of the significant impact on price trends and market movements, the effective forecast of volumes can definitively help to level up the performances in many areas of finance, such as algorithmic trading and asset management.

In light of the recent explosion in computation and information technology in the past decades that has made available vast amounts of data in various domains, more complex and advanced predictive models started becoming the main subject of research in various disciplines, including finance. Data science methods are increasingly playing an essential role in many industries today, and the financial industry, which deals with huge amount of data on a daily basis, has surely realized the importance and the utility of the Data Science methods applied to its field. Across all areas of finance, the vast amount of data available today presents economists and analysts with enormous opportunities. One of the main advantages of using predictive models belonging to Machine Learning (ML) and Deep Learning (DL) is the opportunity to uncover economic relationships that are often not evident when variables are aggregated over many products, individuals, or time periods.

Currently, most of the literature on stock market prediction is mainly focused on price prediction, and only a handful of studies have been conducted on the use of ML and DL methods in volume forecasting, although predicting and generally better understanding volume remains important because many market players and traders are affected by the trading volume.

Today, the industry still relies to a large extent on traditional statistical approaches when it is required to predict asset volume over the lifetime of a market order.

It should be noted that the behaviour of investors, and therefore the volume of trading, is correlated with both the psychology and the behaviour of human beings. For example, the overconfidence of investors can have an impact on the volume of trading and push up its level. Accordingly, factoring it into predictive models might help to add additional relevant information to be used for the trading volume forecasting. The present study will be focusing on the impact of the news on the volume of the asset being traded as well. Today the role of information has never been more important. The advantage of having access to information as early as possible and the ability to process the

information is an important advantage in financial markets. Existing algorithms for textual analysis and news-based trading are capable to determine if the current news flow about companies and their stock prices, currencies or commodities are positive or negative.

The purpose of this study is to further contribute to the research on volume trading forecasting by using ML and DL models and obtain further knowledge regarding the performance of using Data Science models for predicting volume trading.

The analysis is divided into the following research questions:

1) What may the accuracy of prediction of daily stock market trading volume be by using Data Science methods such as ML and DL?
2) Can ML and DL models perform better than a naive model?
3) Can the sentiment analysis, based on news, improve the prediction accuracy?

A general overview of the main issues to be addressed in this study is provided in the first part of this thesis (Chapter 1).

In Chapter 2, the main concepts related to the financial markets and the role of trading volume in them are described. It also highlights the application of data science methods to finance, as well as the use of sentiment analysis in financial markets. A literature review on predicting trading volumes using ML and DL models concludes the second part.

In the third part (Chapter 3) the methodology of the research is presented. The data used in the work and its engineering process are illustrated. Moreover, a comprehensive description of the predictive model selected for the purpose of this work and the metrics used to evaluate and compare the models are described.

The final part (Chapter 4) of the paper is a report on the results of the experiment and the discussions around them.

CHAPTER 2

Background and literature review

2.1 Financial market and Trading volume

Nowadays, numerous assets can be traded in the financial market, such as stocks, bonds, derivatives contracts and more. Although the technology for trading assets over the years continues to evolve, the fundamental market mechanism of buying and selling assets remain the same. In simple terms, sellers need to find buyers, and vice versa, as quickly and as efficiently as possible. Corporations and governments issue assets in order to raise capital required to meet their needs. On the other hand, investors and speculators must be able to easily buy and sell assets in order to obtain a return out of their capital. The ease with which such trading can occur is commonly referred to as liquidity; highly liquid markets are more active and so usually much easier and cheaper to trade in. In order to augment the liquidity, dedicated venues, such as exchanges, have been established [1]. Nonetheless, there may not be always a natural seller and buyer to trade with, therefore, markets rely on intermediaries to facilitate the trading of assets, whose main aim is to make a short-term profit.

Brokers act as agents to execute the orders of their clients. When clients want to buy or sell their investments, they often pay a small commission or brokerage fee to the broker. The broker then executes the order.

Dealers act as a counterparty for both buyers and sellers by setting bid and asks prices for the security in question, and will trade with any investor willing to accept those prices. By doing this, the dealer provides liquidity to the market at a small premium. In other words, dealers will often set bid prices lower than the market and ask prices higher. The spread between these prices is the profit the dealer makes. In return, the dealer assumes counterparty risk.

Financial markets can exist in a physical form, such as stock exchanges, or in a virtual form through electronic trading systems and they have a paramount role in the global economy through their influence on economic policy and the development of financial issues worldwide.

Depending on the financial instrument that is traded, the financial market can be divided into a number of different categories:

- **Stock markets:** In stock markets investors can buy and sell shares of public companies, which grants shareholders ownership in a company (giving them voting rights in major corporate decisions as well) and provides investors with the opportunity to profit if a company is doing well. However, investing in stocks requires an understanding of market trends and company performance, financial analysis, and serious risk assessment.
- **Over-the-counter markets:** Over-the-counter (OTC) markets refer to decentralised platforms that do not have the oversight of an exchange. In these markets, financial instruments are bought and sold directly between the counterparties. OTC markets are not

as strictly regulated as exchanges, which might imply potentially higher risks and less transparency

- **Bond markets:** Bond markets deal with debt securities from governments and companies, whose main aim is to finance themselves for public projects or to compensate budget deficits. In return, investors receive a fixed amount of interest over a predetermined period of time in addition to the principal amount (the bond's face value) at maturity.
- **Money markets:** The money market is related to short-term borrowing and lending funds, usually under a year. It aims at facilitating short-term funding in order to help to maintain economic stability. It allows banks, governments, and companies to have access to funds to meet their short-term needs and preserve a stable cash flow.
- **Derivatives markets:** The derivatives market deals with trading futures, options contracts, and other complex financial products, whose value derives from underlying assets such as bonds, commodities, currencies, interest rates, market indexes, and stocks. Aside from speculating on the future price movements of assets, the derivatives markets allow investors to manage better market risks. For example, a company can use futures contracts to lock in the price of raw materials, protecting against potential price increases.
- **Foreign exchange (forex) market:** The foreign exchange allows traders to exchange domestic currencies against each other. It is one of the largest and most liquid markets in the world. The traders not only necessarily intend to take physical possession of the currencies themselves but also, they may simply be speculating about or hedging against future exchange rate fluctuations.
- **Commodities markets:** Commodities trading involves buying and selling commodities such as agricultural products (wheat, corn, coffee), metals (gold, silver, copper), energy resources (crude oil, natural gas), and other natural resources.

Several factors might impact the daily movement of financial markets. Amongst the most important there are:

- Actions and sentiments of investors, which affect the prices of assets being traded. Supply and demand levels, expectations and change in the level of trust placed in a specific industry are among the core factors that impact the value of financial instruments.
- Individual business conditions and the strength of its larger industry affect the values of the related assets. Profits earned, volume of sales, and even the time of year can affect the investors behaviour.
- Government actions by making decisions on new regulations on a business, interest rates, tax rates, trade policy and more.
- Events around the world, such as changes in currency values, trade barriers, wars, natural disasters, and changes in governments. All these events can impact the view and the expectations of investors, which in turn shape how the financial markets will move.

- Economic indicators, which are closely monitored by investors in order to detect any signal about changes in the economy and try to predict what might happen in the future. Some of the most important indicators are gross national product, the inflation rate, the budget deficit, and the unemployment rate, which can provide some insights about economy performances and its direction.

In order to make informed financial decision, investors need to have deep knowledge of what financial market is and how it works. It is required to crunch numerous financial data before making a trading or investment decision. One of most relevant factors is the trading volume.

Trading volume measures the amount of a given financial asset that is actively traded within a defined time frame. For stocks, volume is measured in the number of shares bought and sold. For futures and options, volume is based on how many contracts have moved from seller and buyer and vice versa.

Volume is a strong indicator of the liquidity of a stock and changes in volume can be used in conjunction with technical indicators to make trading decisions. It is important to note that trading volume will be high when there is a lot of trading in the stock and low when there is not. A highly liquid stock gives traders the flexibility to buy and sell shares more easily because there are a significant number of buyers and sellers for the stock.

It goes without saying that trading volume is an important factor to consider when making financial decisions and analysing stocks and their returns.

Trading volume can somehow reflect the investor disagreement since a trade can take place when a seller and a buyer meet in the financial markets. It implies that at the same point in the time and with roughly the same amount of information two investors have a different valuation of the same financial asset. The researcher W. Beaver in 1968 first associated the trading volume with the level of disagreement between investors [2]. In analysing the trading volume of the week during earning announcements, he found evidence that during the announcement's week the trading was higher because it reflected a lack of consensus on price, due to different interpretations of the report. Moreover, W. Beaver noticed that during the weeks before the announcement, the volume was lower than usual, which could indicate that investors are delaying their decisions to wait for additional information to be released regarding earnings.

In line with W. Beaver's findings, Bamber found that trading volume movements can be caused by effects that financial disclosures have on disagreement and asymmetric information that exists among investors [3] and that the recorded higher trading volume than the firm's average in normal period (non-announcement) can be explained by different views of the investors [4].

It has also been found that volume-based metrics are a better measure of the reaction of investors to public disclosure than are return-based metrics as reported by Cready W. & Hurtt D. [5]

Not only the disagreement degree might cause variation in trading volume, but also the period of the time. It has been found that trading activity might decrease or increase seasonally indeed. Hong &

Yu found that the level of trading activity drops in summer period because the investors, quoting the authors, “are gone fishing” [6]. Moreover, the authors Lakonishok & Vermaelen noticed that trading volume is higher as it gets closer to ex-dividend days [7].

Aside from the link to the level of disagreement between investors, the analysis of trading volume allows to discover insights regarding other relevant factors, such as price and returns of the financial assets. There is a large body of literature on the relationship between trading volume and price, and hence returns, of traded financial assets. Caginalp & Desantis [8] provided statistical evidence in support of the idea that an increase in trading volumes has a positive effect on price fluctuations. In addition, they found that when the price is rising but trading volume is falling, investors perceive the price growth of the financial asset to be unstable.

Karpoff in 1987 oberved that there is a positive correlation between volume and the magnitude of the price change and the price itself in the case of equity markets [9]. In line with Karpoff’s findings, Westerfield (1977) found that there is a positive relationship between the absolute value of daily price changes and daily volume traded [10]. Gervais et al. found that financial instrument with anomalous high (low) trading volumes during a certain period (from one day to one week) tend to rise (fall) during the following month [11].

It is worth to mention the research work of Gunasekarage [12], who observed the existence of a certain asymmetry in the relationship between the stock returns and trading volume. The researcher found that returns play an important role in predicting the future dynamics of trading volume, but that trading volume does not have a very significant impact on the future dynamics of stock returns.

Llorente et al. [13] found evidence that stocks associated with high levels of informed trading have higher return persistence on high-volume days, and stocks associated with low levels of informed trading have higher return reversals on high-volume days, which is consistent with the idea that trading volume contains information about future price changes.

Finally, trading volume may also have a relationship with specific return patterns such as momentum and reversal. In Lee’s and Swaminathan’s research work [14], it was shown that an important link between momentum and value strategies is established by trading volume. The study also suggests that the volume of trading in the past is predictive of the persistence and magnitude of price momentum.

2.2 Machine Learning and Deep Learning in finance

During the past decades, the development of computation and information technology has made available huge amounts of data in various fields. These huge amounts of data represent a valuable source of information, which can surely be used for improving processes and helping business decisions. The necessity of handling this massive amount of data and extracting meaningful information brought on the birth of a new discipline, the so-called Data Science, whose main target

is getting meaningful and useful information out of raw data. Data Science is a cross-disciplinary discipline, which encompasses different subjects such as statistics, Machine Learning (ML), Artificial Intelligence (AI), Deep Learning (DL).

One of the industries that widely benefits from using Data Science is the financial one, which, by nature, is considered one of the most data-intensive sectors. The financial industry handles a massive amount of data on a daily basis, and the employment of Data Science techniques has been regarded as a huge opportunity to process and analyse the data for getting important insight into them across all areas of finance. It is believed that the employment of Data Science-related techniques by the financial industry can increase the competitive advantage, allowing the financial institutions to decrease costs and improving the quality of their services.

The use of data science is being successfully employed across all areas of finance. In the following, is reported a broad picture about how Data science applications can be applied in different sectors of finance.

Algorithm trading

Algorithmic trading concerns the use of algorithms used to carry out trading operations in an automated way. By using complex algorithms, financial institutions are able to perform trading processes in an extremely fast and objective way.

Data Science, through ML and AI, can furtherly improve the performances of the trading algorithms. It can offer the opportunity to employ more advanced trading strategies, which are able to adapt themselves in real time by spotting insight into market movements. Data Science techniques can reinforce the ability of the models to adapt themselves to changing market conditions without any human intervention and instantly, unlike the traditional models that still rely heavily on human actions, which lead to a loss in performance from a time standpoint [15].

Asset management

Asset management departments also started realizing the potential benefits from using Data science solutions for improving their investment process and decision.

By analysing efficiently historical data, it is possible to improve accuracy of investment process, enhance performance, strengthen risk management, and improve the customer experience [16].

The investment strategies of Asset Management have always relied on the capacity to gather information and analyse all the structured data available in the market. However, the use of Data Science methodologies enables to get new and useful information out of vast amounts of raw or unstructured/semi-structured data to be used in the implementation of the investment strategies. The asset managers can now digest massive amounts of data from multiple sources and uncover significant patterns inside the data and adapt their strategies at very short timeframes. In addition, ML and AI solutions can provide automated investment solutions for customers. Special algorithms

can be built to manage financial portfolios in order to achieve the goals and the risk tolerance of a specific client.

Credit Risk Management

Data Science is being successfully employed by financial institutions to evaluate the creditworthiness of borrowers in order to establish the feasibility of credit lending. Data Science models are trained on huge consumer data in order to forecast the probability of default for each borrower. That way, a financial institution, based on the default probability, can hedge itself against the credit risk by not granting the credit or charging properly this risk with higher interest rates. The algorithms of Data Science can reveal financial trends that might influence lending and underwriting risks in the future, allowing the financial institutions to make the best business decision to mitigate these risks.

Nowadays, ML models provide better performance in predicting counterparty default compared to standard statistical models (e.g. logic regressions) especially when limited information is available [17].

Fraud detection

One of the greatest problems for financial institution is fraud. Considering the massive amount of money that every day flows across the financial world, it is necessary to counteract properly the fraud risk and not let the guard down.

AI, ML and DL models can monitor, manage and evaluate data for detecting frauds in real-time. They could identify bad transactions and spot fraud signals by analysing huge amounts of data in very short time, allowing the financial institutions to prevent the frauds from happening instantly.

Unlike the standard financial fraud detection systems, which relied heavily on complex and robust sets of rules, the fraud detection systems, based on Data science model, go beyond following a checklist of risk factors and they can actively learn and calibrate to new potential fraud threats [18].

In the previous sections we realized the great potential and advantage of Data Science applied to the financial sector. However, alongside the developments of Data Science technology in Finance, a greater number of challenges and risks are being little by little identified.

A first potential impediment to fully exploiting the Data Science methodology is “Data Accessibility”. In real-world, the data accessibility is often restricted in order to protect sensitive information and obey privacy rules. Therefore, it is required to achieve a good balance between accessibility and protection in Data Science application [19].

Another important issue that financial institutions employing Data Science methods have to face to is related to the necessity of employing new IT infrastructure for handling massive amounts of data. All the data used in Data science models require to be stored in special computing environments, which are specifically built to process large amount of data in an efficient way. Financial institutions that want to exploit the Data Science methodologies will have to massively invest in acquiring proper

IT environments in order to effectively enable reliable execution of machine learning algorithms and AI techniques [20].

A further dark side in Data Science is related to potential discrimination issues. It is well-known that Data Science methods have the potential to help avoid discrimination based on human interactions in financial services. By delegating decision part to the algorithm, the user of the ML/AI-based model avoids biases attached to human judgement. However, at the same time, the use of ML/AI applications may introduce bias or discrimination if biases are found in the data. For this reason, in using Data Science techniques, it is not possible to fully rely on their outcomes. The assessment of humans in the decision-making process is still critical in order to identify and correct potential biases present into the data or in the model design [20].

Related to the outcomes of the algorithms used in Data Science, one of the main difficulties is understanding why and how the model generates results. This difficulty in justifying or rationalising model decisions and outputs is generally described by the term 'explainability'. The main reason for that is obviously due to the inherent complexity of ML/AI-based models. Due to this limited understanding of the underlying logic of the models, the users have still limited room to predict how their models may affect market conditions, and whether they contribute to financial market disruption. There exists a trade-off between explainability and performance of the model and for this reason financial institutions need to achieve the right balance between explainability of the model and performance in term of accuracy. Some degree of insight into the models prevent them from being considered as 'black boxes'. The lack of explainability of model used can be a serious challenge when faced to regulatory framework. It is well-known that the financial industry is arguably one of the most heavily regulated industries worldwide. When it comes to data science, one of the main concerns of the regulator is a strong need for explainability, particularly when it comes to models. It is well-known that there exists a trade-off between explainability and model performance. Put in other words, models that can give more accurate predictions tend to be the more complex black box models, like a deep learning neural network model. This potential difficulty to understand the underlying theory of the model and this lack of explainability surely raises concerns with regulators

2.3 Sentiment analysis in financial market

Sentiment analysis is a Natural Language Processing (NLP) technique, which attempts to assess emotional tone by processing large amounts of unstructured data. This is also known as opinion mining. Sentiment analysis can use data from a variety of sources, such as comments on social networks, news articles, consumer reviews and more [21].

By identifying and categorising opinions in textual data, companies and organisations can gauge public sentiment about products, services or issues and make informed decisions.

From brand monitoring to customer feedback analysis, sentiment analysis has many practical applications. The following are some of the most important areas in which sentiment analysis can be used [21]:

- **Business and Marketing:** Companies use sentiment analysis on social media, review platforms and forums to monitor brand reputation and customer feedback, which can provide useful information for adjusting marketing strategies, developing products, and improving customer service.
- **Financial Markets:** Based on the sentiment expressed in news articles, analyst reports and social media, sentiment analysis might be used to predict market trends. For example, positive news about a company can lead to a rise in its share price, while negative news can have a downward effect.
- **Politics and Public Opinion:** During an election, sentiment analysis can help you gauge public opinion on a candidate or issue based on the discourse on social media and in the news. That way, it might allow to properly design campaign strategies and predict election outcomes.

There are a number of ways in which sentiment analysis can be performed on text data, with varying degrees of complexity and accuracy. The most common methods are as follows: a lexicon-based approach, a machine learning (ML) based approach, a deep learning (DL) approach and hybrid approach as illustrated in Figure 1.

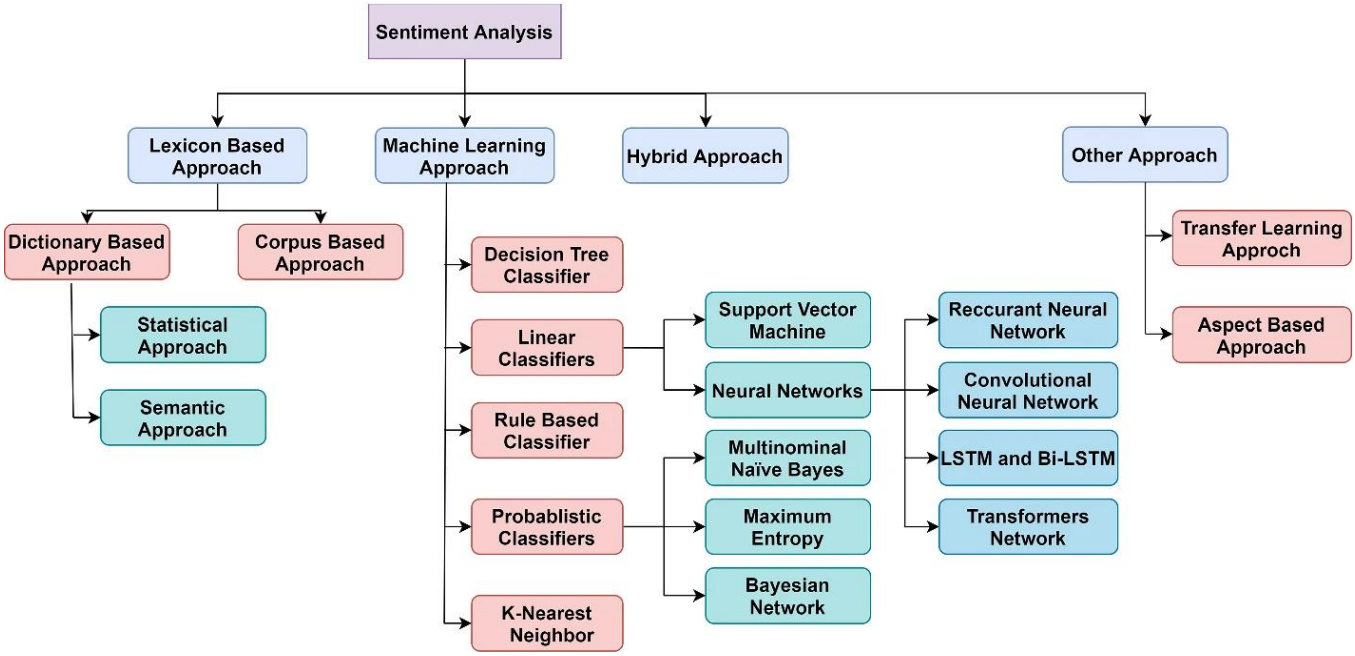


Figure 1. Break-down of the main methods for sentiment analysis [34].

Lexicon-based sentiment analysis is a popular technique for extracting the emotional polarity of text, which can be classified as positive, negative or neutral [22]. It is based on pre-defined dictionaries (lexicons) of words and phrases (tokens) to which sentiment scores or categories have been assigned. Tokens are scored based on polarity, as follows: +1, 0, −1 for positive, neutral, negative,

respectively, or the score can be assigned on the basis of the intensity of the polarity and its values range from [+1, −1] where +1 represents highly positive, and −1 represents highly negative.

The scores of each token are aggregated. This means that positive, negative and neutral scores are summed separately. Finally, based on the highest value of the individual scores, an overall polarity is assigned to the text. To summarise, the text is first broken down into tokens of individual words. The polarity of each token is then calculated and finally aggregated and the overall sentiment of a text is returned.

It could be described as an unsupervised technique as no training data is required [23], which can be regarded as an advantage. However, the main disadvantages of this technique are its domain dependency [24] and it fails to consider context and sarcasm, which can lead to inaccuracies. Words can have multiple meanings and senses, and therefore a positive word in one domain may be negative in another. For instance, the word 'big' may be positive or negative based on the domain in which it is being used. In "the obstacle to overcome is big" the word may assume negative meaning whereas, in "the company obtained big revenue" the word can be considered positive. Therefore, the polarity of words should be carefully considered in relation to the domain.

Some of the advantages in using Lexicon-based method are that they are not expensive and do not rely on advanced sentiment analysis algorithms. In addition, there is no need for training data, especially if companies use a dictionary-based approach. With respect to their limitations, aside from the aforementioned incapacity to identify sarcasm, negation, grammar mistakes, misspellings, or irony and their domain-dependency, lexicon-based approach might be time-consuming and prone to human bias since the labelling is handled manually.

To sum up, Lexicon-based methods can be useful in cases where there is a limited amount of training data, or where domain-specific knowledge is required, but they can be limited by not taking context into consideration [25].

In machine learning methods, algorithms are trained on the labelled data in order to learn the patterns and relationships in textual frameworks. The uncovered patterns are then used to assess and classify the sentiment of a given text by associating specific features of the text (like words or phrases) with sentiment labels. Needless to say, the success of this approach is a function of the quality of the training data set and the quality of the algorithm. Unlike lexicon-based methods, ML algorithms for sentiment analysis can be trained to go beyond the understanding of simple definitions and understand contextual information, sarcasm and misused words.

Traditional machine learning classifier such as Support Vector Machines (SVM), Naive Bayes, decision trees (DT) and K-Nearest neighbours (KNN) can be employed for sentiment analysis, each with advantages and disadvantages.

SVM is a non-probabilistic supervised learning algorithm that separates two classes with maximum margin and it is widely used in sentiment analysis due to its effectiveness in the classification and evaluation of sentiment across a variety of data sets.

Naive Bayes (NB) is a supervised, probabilistic classification approach based on Bayes' Theorem and is used for feature extraction. This approach can handle misspellings and grammatical errors because it assumes that each token or feature is independent of the others. By applying the Bayes's theorem it estimates the likelihood of data points belonging to different categories under the feature independence assumption.

Decision tree (DT) classifiers use a tree-like model of decisions and their possible consequences to categorize sentiment. This method recursively splits data based on feature values, and it can be combined with techniques such as Random Forest, which helps to reduce overfitting and improve accuracy.

K-Nearest neighbours (KNN) classifies data based on the majority sentiment of its nearest neighbours. Though less common in sentiment analysis, KNN can be effective when tuned properly, leveraging proximity in feature space to determine sentiment scores.

Among the main advantages in using automated ML method for sentiment analysis there are: the possibility to be trained to detect sarcasm, irony, or negation in sentiment analysis, their ability to learn the valence of the words without requiring a pre-determined dataset and more accuracy of the results. On the other hand, the main disadvantages in using these methods are the need of large and high-quality dataset in order to obtain accurate results and higher costs when compared to traditional methods like the lexicon-based method, although these methods show the potential to be more accurate than rule-based methods [26, 27].

In Figure 2 a comparison of automated and lexicon-based sentiment analysis methods is displayed.

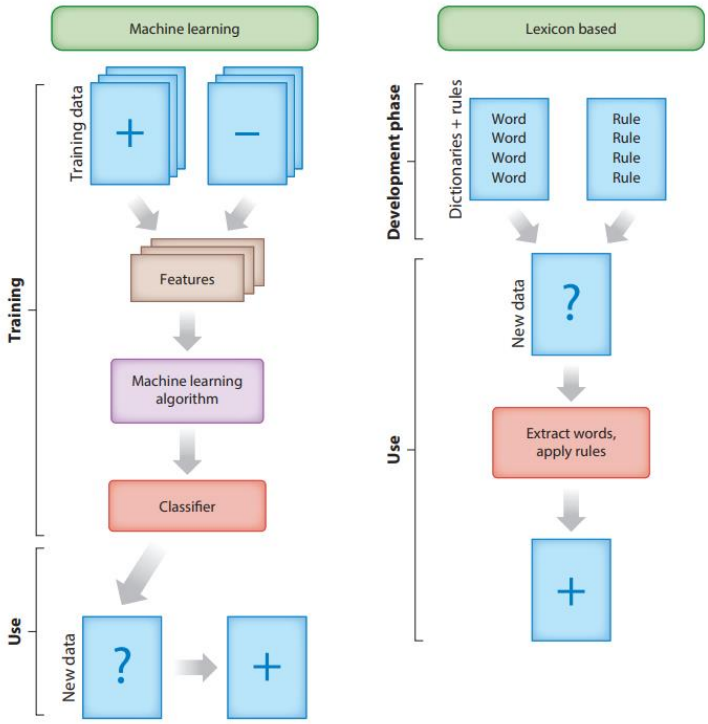


Figure 2. Comparison between Machine Learning and Lexicon-based methods for sentiment analysis (source: Taboada M. (2016). Sentiment analysis: An overview from linguistics. *Annual Review of Linguistics*)

The aforementioned methods, lexicon-based and automated methods, have advantages and disadvantages. Thus, combining lexicon-based and machine learning approaches can leverage the strengths of both, improving accuracy and context sensitivity.

Other approaches include more advanced methods than traditional machine learning, such as deep learning methods that use complex neural networks to capture the sentiment in data. When it comes to understanding context and the influence of word order on sentiment, these models are particularly effective. They can be used to classify text at a more granular level, such as identifying specific emotions or opinions. However, they require vast amounts of labelled data and can be computationally expensive [28].

Overall, the specific application, available data and resources will determine the choice of method or technique for sentiment analysis. It is important to carefully evaluate the strengths and limitations of each approach before deciding which one to use.

2.4 Literature review on trading volume forecast with ML and DL models

Compared to price prediction, the literature on volume prediction is still not plentiful. However, as reported in Section 1.2, factoring in trading volumes in business decisions might be important because many investor strategies are affected by the trading volume, therefore carrying on with research on this topic might reveal precious and return further relevant information to be employed when it comes to investment decision. In the following, some of the most interesting research works on trading volume prediction are mentioned.

With respect to traditional approaches, it is worth to mention a study by Thomas Lux & Taisei Kaizoji [29], whose focus is on the evaluation of the performance of long memory time series models (FIGARCH and ARFIMA) in comparison to their short memory counterparts (GARCH and ARMA). They found that the long memory models outperformed the short-memory models. Despite a number of cases with dramatic failures of their forecasts, the FIGARCH and ARFIMA do not suffer from this shortcoming and their accuracy practically is always higher than the traditional models based on historical volatility.

Moving on learning model, as they got increasingly popular and common, several interesting studies have been performed related to market volume prediction.

In 1995, Iebeling Kaastra & Milton Boyd performed a study to predict the futures trading of six commodities for Winnipeg Commodity Exchange by using a backpropagation neural networks model, which is benchmarked to traditional ARIMA model [30]. The authors of the study found that neural networks model is able to forecast up to nine months ahead and to outperform the naive model for all commodities except barley and rye.

Support Vector Regression model (SVR) and Partial Least Squares (PLS) for daily trading volume prediction have been tested out for daily volume prediction, by Leandro G. M. Alvim et al. [31]. The authors designed dynamic model for daily volume forecasting, which uses the partial volume information during the day for obtaining better daily volume prediction.

The authors concluded that SVR and PLS have got better performances in comparison to naive strategy, which simply sums the *n* prior volume intervals of the current day interval.

Long Short-Term Memory (LSTM) networks have been used in conjunction with Support Vector Regression (SVR) and Autoregressive (AR) models for forecasting the changes in intraday trading volume in the research work of Daniel Liban et al. [32]. The authors factored in the hour into the feature vector in order to consider the typical U-shape of the intraday trading volume and improve the forecast accuracy. The results shown that LSTM networks can achieve a more accurate forecast, particularly when they are part of a hybrid model with other models such as AR.

The study of Xiaojie Xu et al. [33], in an attempt of providing response to some academic question related to volume prediction, concluded that a relatively simple deep learning model (based upon ten hidden neurons and thirty lags) can achieve stable and accurate prediction results and tackle the prediction problem related to rather irregular volume. In addition, the study found that further including predictive information from trading volumes of the futures did not benefit predictions.

Bin Gao & Jun Xie in their research work focus on the ability of investor sentiment to forecast excess returns and abnormal trading volumes in Chinese stock index futures market [34]. By using Ordinary Least Squares (OLS) regression, the authors show that over a daily horizon, stock index futures sentiment has a significant explanatory power for the stock index futures excess returns and trading volume. Moreover, the sensitivity of returns and trading volumes to sentiment is more significantly positively related to the highly volatile sentiment period.

Another interesting study by Liang Zhao et al. [35], based on Correlation-powered Graph-based Multi-view (CGM) modelling method, try to incorporate multi-view information, i.e., long-term stock trend, short-term fluctuation and sudden events information jointly into a temporal heterogeneous graph. The results obtained show how this method outperforms baselines methods by large margin.

CHAPTER 3

Methodology

The goal of the present work is to predict the average medium-term percentage change (10 days) of volume traded of TESLA and APPL stock. The percentage change of the volume is regarded as the target dependent variable to be predicted. The independent variables considered are some of the daily market data related to the assets such as: volume, adjusted close price of stock, adjusted close price of stock index and several market indicators, which are extracted from the existing data. In addition to them, the daily sentiment score computed on the ground of the stock news will be factored in as well. The volume percentage change prediction will be performed by both not factoring and factoring in the sentiment score for each of the predictive model. That way, the impact of introducing sentiment information on prediction of volume change will be assessed.

The following is an overview of the methodology of the present work. The data used and the theoretical background are described.

3.1 Datasets description

The data employed for the present work can be grouped into two main classes: financial data and textual data for the sentiment analysis.

The data are collected from external open sources, which can provide the right amount of data for free and without the need of costly subscription. This condition surely may stand for a limitation and reduce the scenarios covered by this study. With this limitation in mind, the data required to perform the analysis are retrieved through the following channels:

- Financial data: the platform used is *Yahoo! Finance*, which is part of the *Yahoo!* network. It makes available different type of financial data in tabular format, among which stock quotes and stock volumes for different time framework [36].
- News headlines for sentiment analysis: for sentiment analysis, the platform used for retrieving the data is '*Kaggle*', which is an online community platform for data scientists. It allows users to collaborate with other users, find and publish datasets and use GPU integrated notebooks. '*Kaggle*' makes available numerous data set containing news and headlines over the time provided by different sources (for example Reuters, Guardian, Twitter, etc.) [37].

The main datasets collected from the aforementioned sources are described below:

- a dataset containing the time serie of historical news/headlines related to the both Tesla (TSLA) and Apple (APPL) stocks that will be used for sentiment analysis purpose. The Tesla data set consists of 22274 instances and covers the time period from 8 Jul 2008 to 10 Apr 2023. The Apple news data consist of 2517 instances and cover the time period from 1 Dec 2006 to 30 Nov 2016.

- a dataset containing time serie of historical market variables related to both Tesla and Apple stocks, such as prices and volume. The Tesla dataset consists of 3530 instances and covers the time period from 1 Jul 2010 to 11 Apr 2024. The dataset related to Apple consists of 2517 instances and covers the time period from 1 Dec 2006 to 30 Nov 2016.
- a dataset containing the time serie of historical market variables related to the index Nasdaq100 and Vix index. The original data sets of Nasdaq100 and Vix index contains 4657 and 4829, respectively, and cover a time period from 3 Jan 2006 to 5 Jul 2024.

The datasets related to the stocks and market indices contain the following features:

- **Date:** the date in daily format,
- **Open:** the daily open price,
- **High:** the highest price recorded during the day,
- **Low:** the lowest price recorded during the day,
- **Close:** the daily closing price,
- **Adj Close:** the daily adjusted closing price
- **Volume:** the daily traded volume.

The datasets related to the stock news/headlines contain the following features:

- **Date:** the date in daily format,
- **Headlines:** the daily news/headlines related to the stock,
- **Source:** the source where the news/headlines come from.

In order to have a consistent time interval across the different datasets, the longer time series are processed in order to meet the length of the shortest dataset available. Looking at the available datasets, the main discriminant in defining the length of the time period considered for each two stocks is the sentiment data available. With this in mind, the final datasets related to the two stocks will be designed as follows:

- **Tesla:** The data covering the time period from 1 Jul 2010 to 10 Apr 2023 (3215 instances) are considered
- **Apple:** The data covering the time period from 1 Dec 2006 to 30 Nov 2016 (2517 instances) are considered

The corresponding time period above-defined for the two stocks will be considered in selecting the data of Nasdaq100 and Vix index for Tesla and Apple, respectively.

To sum up, in Table 1 are reported the time interval considered for the two stocks and their associated datasets

	Financial data (price, volume)	Nasdaq100 index	Vix Index	News/headlines
Tesla	1 Jul 2010-10 Apr 2023	1 Jul 2010-10 Apr 2023	1 Jul 2010-10 Apr 2023	1 Jul 2010-10 Apr 2023
Apple	1 Dec 2006-30 Nov 2016	1 Dec 2006-30 Nov 2016	1 Dec 2006-30 Nov 2016	1 Dec 2006-30 Nov 2016

Table 1. Time interval considered for the data in relation to the TSLA and APPL stocks.

3.2 Sentiment Analysis with NTLK

Sentiment analysis is a natural language processing technique that is used to identify and extract the feelings and opinions that are expressed in textual data. The main goal of sentiment analysis is to uncover the underlying sentiment of text and classify it into negative, neutral, and positive categories [38].

As described in Section 1.4, there are different approaches to implementing sentiment analysis, i.e. Rule-Based Approach, Machine Learning Approach and Hybrid Approach.

The tool used to perform the sentiment analysis is the python package Natural Language Toolkit (NLTK), which relies on the rule-based method VADER (Valence Aware Dictionary and Entiment Reasoner). VADER is based on the use of a dictionary called Lexicon, which contains various words that are labelled as either positive or negative according to their semantic orientation [39]. Below in Table 2 is provided an example.

Word	Sentiment rating
Tragedy	-3.4
Insane	-1.7
Disaster	-3.1
great	3.1

Table 2. Example of Lexicon dictionary for VADER.

In order to identify the sentiment underlying a text, the VADER performs the following steps [40]:

1. The text is broken down into individual words.
2. A score from lexicon dictionary is assigned to each word in order to identify if it is positive, negative or neutral.
3. Based on the previous assigned scores, VADER finally calculates the overall sentiment score of the text
4. The scores returned by VADER range from -1 to 1. -1 implies very negative sentiment. On the other hand, 1 implies very positive sentiment.

The use of VADER method for sentiment analysis offers several advantages. Firstly, unlike ML approaches, it does not require training phase and the underlying logic is more intuitive and

transparent. In addition, the use of a pre-defined dictionary makes the VADER faster than any other rule-based or ML-based sentiment analysis algorithm

Despite its relevant advantages, VADER method has some limitations due to its simplicity. It might struggle with detecting sarcasm and irony in the text. Moreover, negations are not easily identified by VADER as well. For example, “not bad" is a positive word but can be classified as unfavourable based on the lexicon.

In Figures 1 and 2 are reported the score of the sentiment analysis performed on the news/headlines datasets of the two stocks considered, TSLA and APPL, respectively.

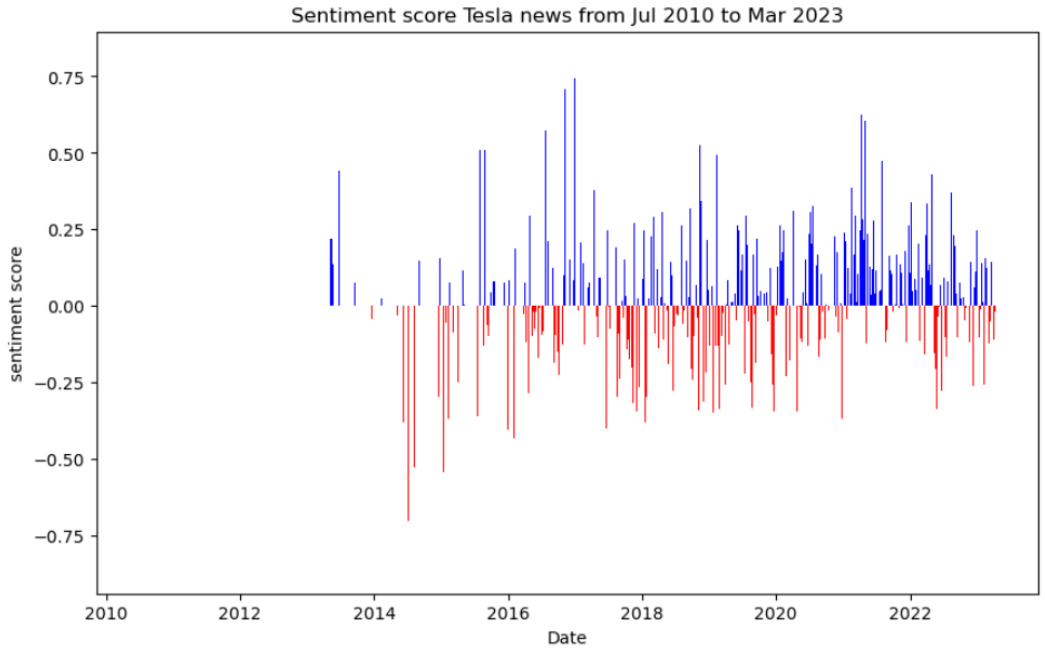


Figure 1. TSLA sentiment score over the time considered

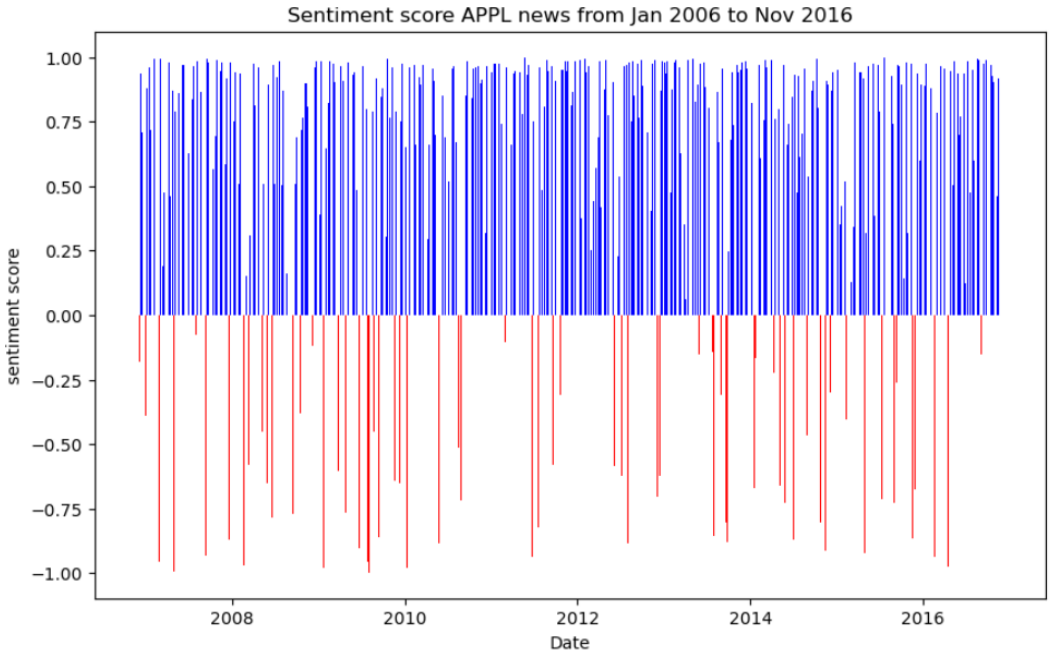


Figure 2. APPL sentiment score over the time considered

In TSLA plots, it is observable a low density of score values during the time period from 2010 to about 2015 due to scarcity of data related to news about the company. It is expected since the TESLA company was listed on the stock exchange in June 2010 and during the initial period, it might have been challenging to collect large samples of news related to it. Where the news was missing the score has been set to neutral value equal to 0.

3.3 Data pre-processing

In order to obtain the final data frame that will be fed into the predictive models employed in this work, pre-processing and feature engineering steps are performed on the data.

The main data frames of the two stocks are merged on the 'date' feature with their corresponding datasets containing sentiment analysis and indices (Nasdaq and Vix) information.

After the merge, the stock data frames will consist of 11 features: the date (daily format), daily open price of the stock, daily highest prices of the stock, daily lowest prices of the stock, daily close price of the stock, adjusted close price of the stock, daily volume of the stock, daily sentiment score of the stock, daily close price of Nasdaq index, daily volume of Nasdaq index and daily Vix index value.

3.3.1 Feature Engineering

The main goal in feature engineering is the creation of new input or target features from existing features of dataset, in order to add more information to be fed into the predictive models and improve the accuracy of the predictions.

The data frames of the two stock are enriched with the daily price variation and daily percentage change of volume, which are computed as follows:

$$daily\ price\ variation_i = closing\ price_i - \ closing\ price_{i-1} \quad (1)$$

$$daily\ pct\ volume\ change_i = \frac{volume_i - volume_{i-1}}{volume_{i-1}} \quad (2)$$

The $daily\ pct\ volume\ change_i$ represents the target variable to be predicted in our research work.

In addition, other market indicators will be computed out of the available data. The market indicators are widely used by investors in order to detect any relevant clue about whether a financial instrument is trending, the probability of its direction and more. The market indicators are the building block of technical analysis, which is used to forecast the direction of security prices through the analysis of the past market data. They are quantitative tools obtained by mathematical transformation of market variable such as price and volume and can indicate the buy and the sell signals in order to maximise the profit and mitigate the risk [41].

There are numerous market indicators available, each one providing information about specific aspect of the market, such as momentum, buy and sell level condition, price trend, volatility and so

on. Considering the volume-related nature of the target variable to be predicted, for the present work the majority of the market indicators selected are linked to volume trading. Specifically, the following market indicators are computed:

- **On-Balance Volume:** OBV can be regarded as a momentum indicator, which relies on volume flow to predict changes in stock price. The daily OBV indicator is computed as follows [42]:

$$\text{If the closing price is above the prior close price:} \quad OBV_i = OBV_{i-1} + volume_i \qquad (3.a)$$

$$\text{If the closing price is below the prior close price:} \quad OBV_i = OBV_{i-1} - volume_i \qquad (3.b)$$

$$\text{If the closing prices equals yesterday}' \text{closing price:} \quad OBV_i = OBV_{i-1} \qquad (3.c)$$

- **Relative strength index:** RSI is a momentum oscillator that can be used to detect overbought and oversold conditions in the market. It oscillates between 0 and 100. The steps for computing the daily RSI are [43]:

  - Computing the upward change (U) or downward change (D):

$$U_i = \begin{cases} closing\ price_i - closing\ price_{i-1} & if\ closing\ price_i > closing\ price_{i-1} \\ 0 & if\ closing\ price_i \leq closing\ price_{i-1} \end{cases} \qquad (4.a)$$

$$D_i = \begin{cases} closing\ price_{i-1} - closing\ price_i & if\ closing\ price_{i-1} > closing\ price_i \\ 0 & if\ closing\ price_{i-1} \leq closing\ price_i \end{cases} \qquad (4.b)$$

  - The two averages are computed:

$$MA_{Ui} = U_i + MA_{Ui-1}\ \frac{period - 1}{period} \qquad (4.c)$$

$$MA_{Di} = D_i + MA_{Di-1}\ \frac{period - 1}{period} \qquad (4.d)$$

  - The final RSI formula is:

$$RSI_{\boldsymbol{i}} = 100 - 100\ \frac{1}{1 + \frac{MA_{Ui}}{MA_{Di}}} \qquad (4.e)$$

The period considered for the computation of the RSI in the present work is equal to 14 days.

- **The Average True Range:** ATR is a technical indicator that measures the volatility of the financial market by decomposing the entire range of the price of a stock for a particular period. The formula for computing the daily ATR is [44]:

$$ATR_i = ATR_{i-1}\ + \frac{TR_i}{n} \qquad (5.a)$$

  Where n is the number of periods and TR_i:

$$TR_i = Max(high\ price_i - low\ price_i,$$
$$|high\ price_i - closing\ price_{i-1}|, |low\ price_i - closing\ price_{i-1}|\ ) \qquad (5.b)$$

The number of periods considered in computing ATR here is 14.

- **Ease of Movement Value:** EMV is an indicator that attempts to quantify both price and volume into one quantity. Since it considers both price and volume, this indicator might prove useful in determining the strength of a trend. The steps for computing the daily EMV involves different calculation [45]:

$$EMV = \frac{(high\ price - low\ price)}{\frac{(\frac{volume}{10000})}{(high\ price - low\ price)}} \qquad (6)$$

  Where $high\ price$ and $low\ price$ refers to the highest and lowest price during a specified period, respectively. $volume$ denotes the total trading volume during that period. The period considered for the computation of the EMV is equal to 14 days.

- **Volume-weighted average price:** VWAP provide a measure of the level the average price at which a specific asset has been traded over a specified time period by considering the trading volume at each price level. When the VWAP indicator goes up, it might signal an increasing trend of the price [46]. The formula for VWAP is:

$$VWAP_i = \frac{\sum_{n=1}^{i}(volume_n * price_n)}{\sum_{n=1}^{i} volume_n} \qquad (7.a)$$
$$Price_n = \frac{high\ price_n + low\ price_n + closing\ price_n}{3} \qquad (7.b)$$

  The period considered for computing the VWAP is equal to 14 days.

3.3.2 Data smoothing

One characteristic of time series data is the presence of random variation, which could prevent the models from achieving satisfactory performance accuracy if it is highly marked. Looking at the target variables (the daily volume percentage change of volume), which are displayed in Figures 3 and 4 for Tesla and APPL, respectively, the daily percentage changes of volume resemble closely a white noise, which might be more challenging to be predicted for the models.

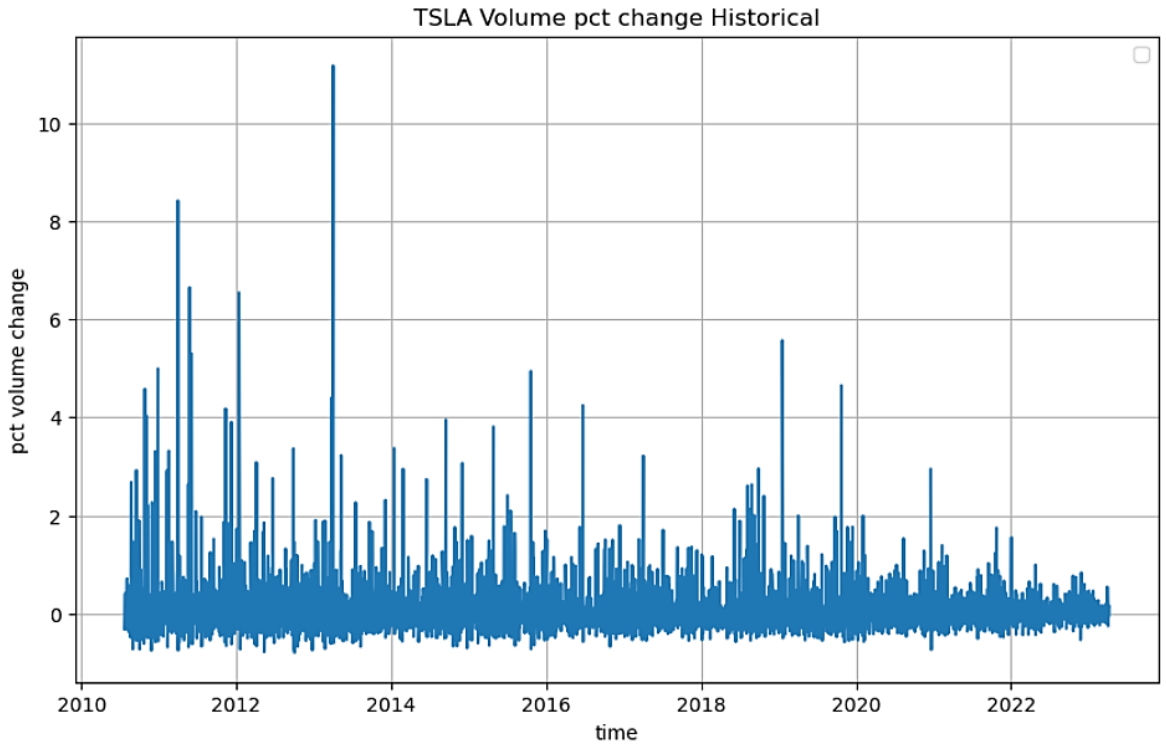


Figure 3. Time Serie of the daily percentage change of volume for TSLA stock

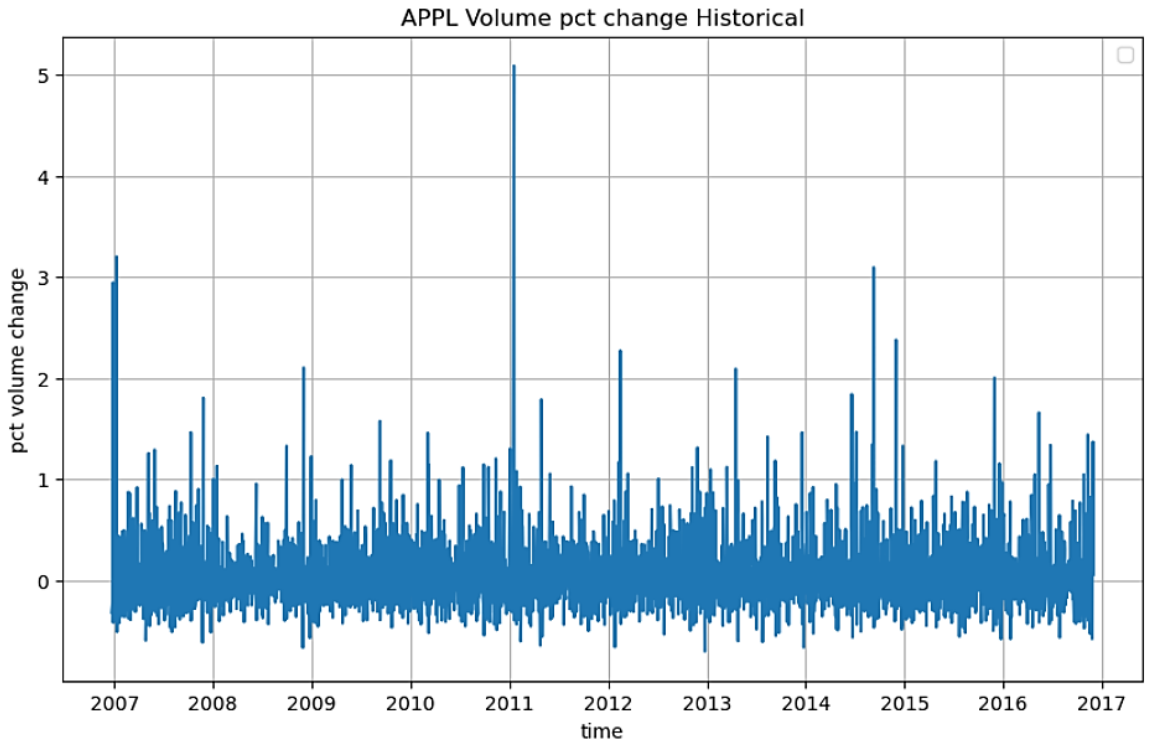


Figure 4. Time Serie of the daily percentage change of volume for APPL stock

In order to try to improve the quality of the predictions, one commonly used technique for reducing such random variation is smoothing. One approach is simply computing a moving average in order to compute the medium-term averages.

The size of the rolling window chosen is 10 days. For a given data series $x_1, x_2, x_3, x_4, \dots, x_n$, after computing the moving average with previous periods considered $k = 10$, the new moving average serie become:

$$\left[\frac{x_1, x_2, x_3, x_4, \dots, x_{10}}{10}\right], \left[\frac{x_2, x_3, x_4, x_5, \dots, x_{11}}{10}\right], \dots, \left[\frac{x_{n-9}, x_{n-8}, x_{n-7}, x_{n-6}, \dots, x_n}{10}\right] \qquad (8)$$

From economical perspective, this would mean that we are going to focus more on medium-term changes of volume instead of random daily fluctuations.

In the Figure 5 and 6, the smoothed time series of the percentage change of volume for TSLA and APPL, respectively.

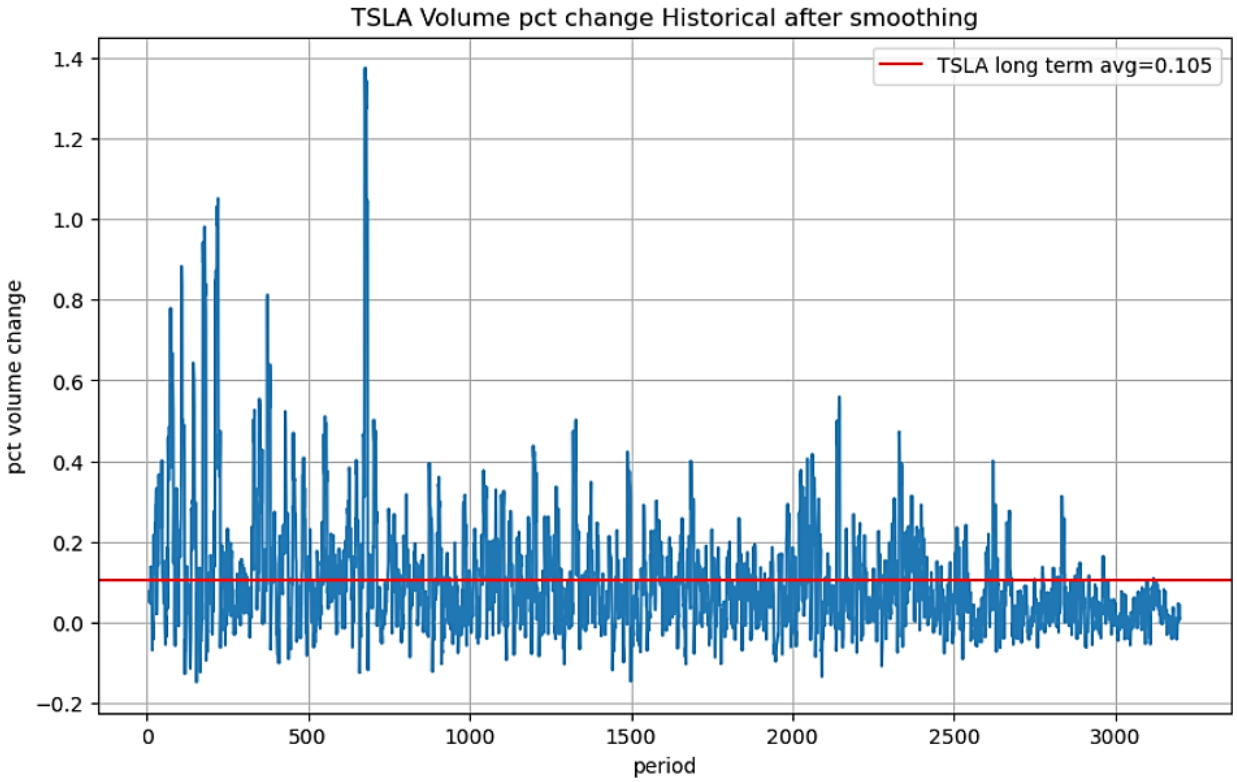


Figure 5. Time Serie of the daily percentage change of volume for TSLA stock after smoothing process

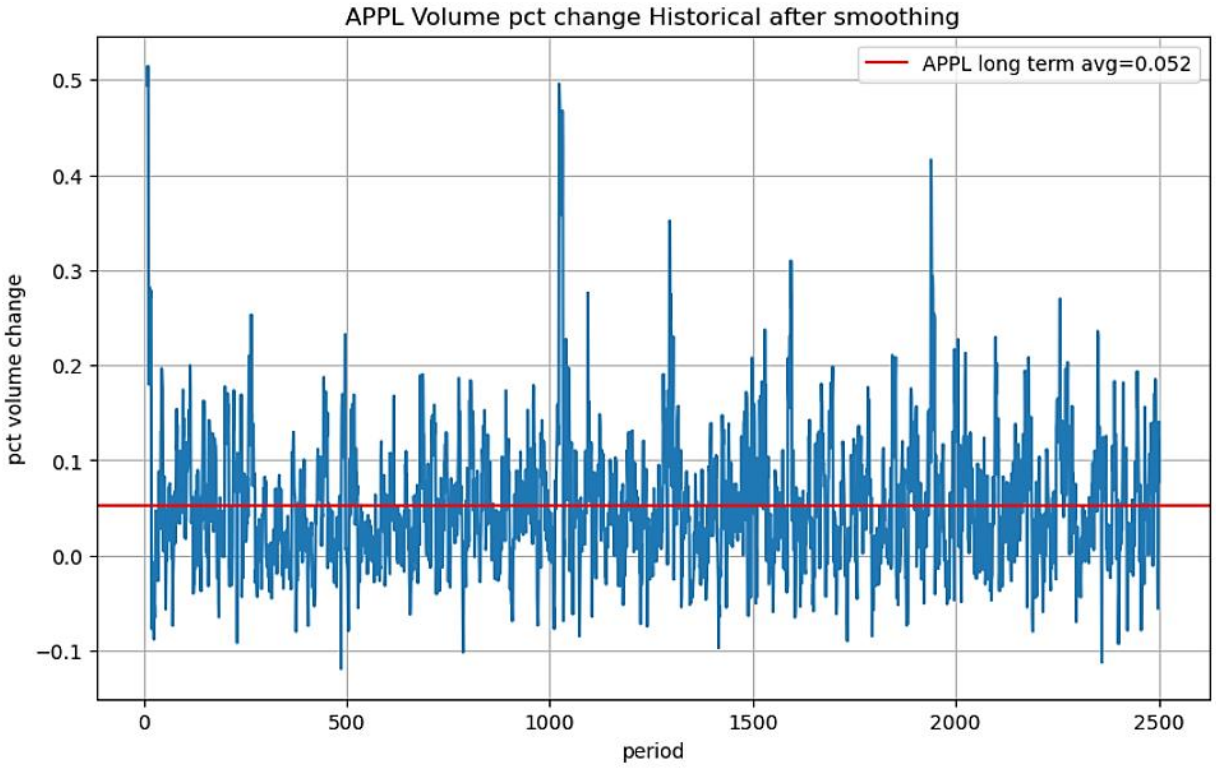


Figure 6. Time Serie of the daily percentage change of volume for TSLA stock after smoothing process

When compared to Figures 3 and 4, it is noticeable a relevant reduction of the random fluctuations, which might help to improve the accuracy of the predictions.

3.4 Experimental setup

The programming language employed to work out the present work is Python (v. 3.10.14).
Several built-in libraries of Python have been used to perform the different tasks during the analysis. The most relevant are:

- Scikit-learn (v. 1.4.2), which is an open-source library designed for Machine Learning. It provides tool for data analysis and development of both supervised and unsupervised ML models.
- TensorFlow (v. 2.10.0), which is an open-source library for artificial intelligence, that focuses on deep learning.
- Keras (v. 2.9.0), which is open-source library that provides a powerful tool to implement neural networks and it relies on TensorFlow library.
- Nltk (v. 3.7), which, as already described in Section 2.3, is an open source Python library that supports research and development in Natural Language Processing.
- Pandas (v. 2.2.2), which is a open-source library for data manipulation and analysis.
- Matplotlib (v. 3.8.4), which is an open-source library designed for data visualization.

3.5 Feature Selection

Feature selection is one of the core concepts in Machine Learning and Deep Learning. The main goal is identifying the related features from a set of data and removing the irrelevant or less important features that do not contribute much to our target. By removing irrelevant features might help to achieve better accuracy for our model. The feature selection here will be performed by relying on the results coming from VIF analysis and Pearson correlation.

Variance Inflation Factor

VIF is a method to measure the existing multicollinearity between the independent variables, whose presence might adversely affect the results of the predictive model. VIF provide a measure about how much of an independent feature is influenced by its correlation with the other independent features. In general, we have the following thresholds [47]:

- -VIF equal to 1 = variables are not correlated
- -VIF between 1 and 5 = variables are moderately correlated
- -VIF greater than 5 = variables are highly correlated

One approach for reducing multicollinearity issues is to eliminate the independent variables whose VIF is high (greater than 5), although for the present work only features with very relative high values of VIF will be eliminated. In Table 3 is reported the result of VIF analysis that has been performed on both the stocks.

TSLA		APPL	
Feature	**VIF**	**Feature**	**VIF**
Open	6669.21	Open	32215.10
High	7984.82	High	44408.14
Low	6428.71	Low	36136.02
Close	inf	Close	54985.80
Adj Close_tsla	inf	Adj Close_appl	6410.97
Volume_tsla	3.98	Volume_appl	8.14
sent_score_tsla	1.05	sent_score_appl	1.59
Adj Close_nasdaq	32.52	Adj Close_nasdaq	323.86
Volume_nasdaq	25.11	Volume_nasdaq	30.23
Vix_index	9.53	Vix_index	13.76
Price_variation_tsla	3.53	Price_variation_appl	2.44
OBV_tsla	38.04	OBV_appl	30.47
VWAP_tsla	20.41	VWAP_appl	188.90
RSI_tsla	9.11	RSI_appl	26.00
EMV_tsla	1.62	EMV_appl	1.49
ATR_tsla	25.02	ATR_appl	21.50

Table 3. Varian Inflation Factor scores for TSLA and APPL stocks

It observable a really highly multicollinearity between the stock price features in both TSLA and APPL, which is to be expected since they provide approximatively the same information.
In addition to VIF analysis, measuring the Pearson coefficient between the independent variable can provide further information about how strong a relationship is between data. The correlation coefficient ranges from -1 to 1 [48]:

- 1 indicates a strong positive relationship.
- -1 indicates a strong negative relationship.
- -A result of zero indicates no relationship at all.

It is worth to mention that the Person coefficient measures the level of the linear relationship, which implies that if the Pearson correlation is zero, there might still exist a non-linear relationship.
If two variables are highly correlated, it implies that they contribute with the same information and it would be redundant including all of them in a model. High correlation can cause multicollinearity issue again, and it might be difficult to determine the independent effect of each variable on the target variable. Discarding variables with high correlation is always advisable whenever possible since it might help to improve the efficiency and accuracy of the prediction model.
In Figures 7 and 8 are displayed the correlation matrices related to the stock TSLA and APPL, respectively.

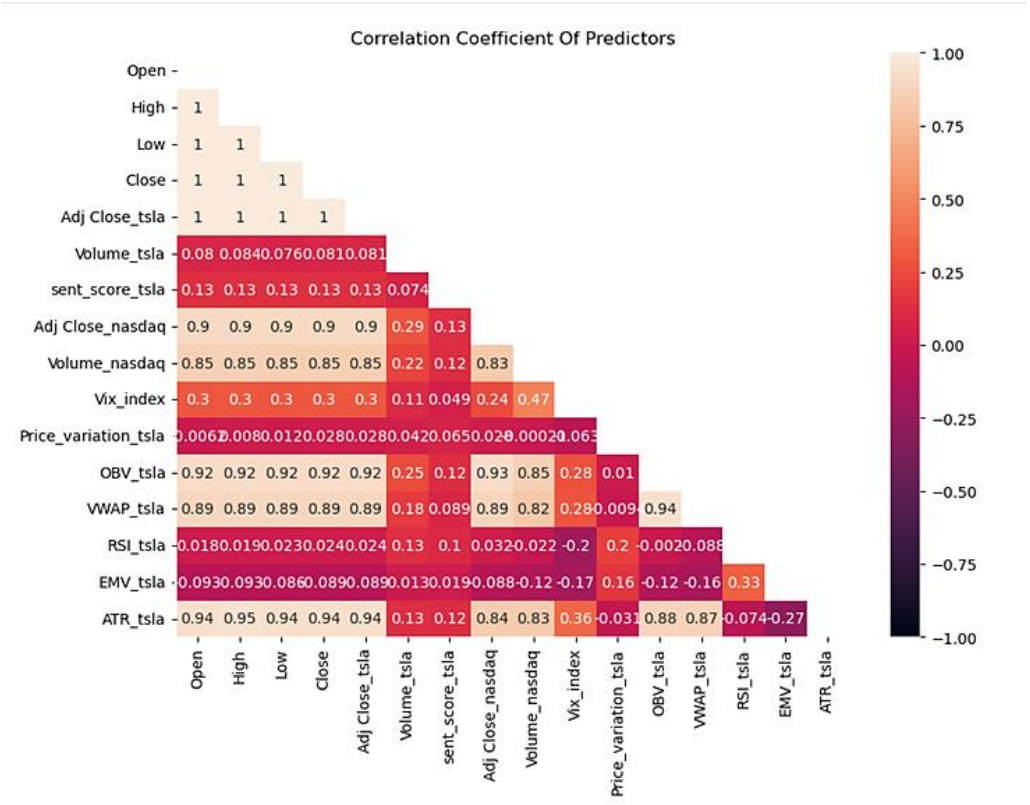


Figure 7. Correlation matrix for TSLA.

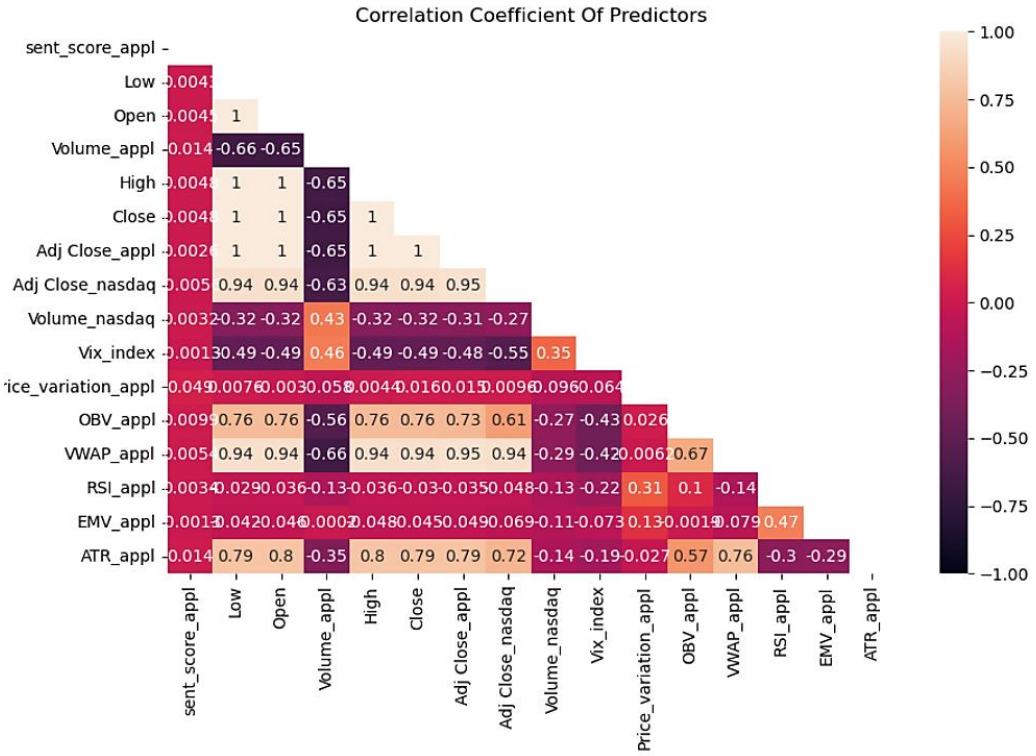


Figure 8. Correlation matrix for APPL

Looking at the above matrix correlations, the results of both TSLA and APPL approximately reflect what was observed in VIF analysis with high correlation between the stock price features (Open, Close, High, etc.) which is to be expected. A relevant correlation between stock price and index price variables is observed as well.

According to the results of VIF and Pearson correlation, the following features are discarded for both of the stocks: Open, High, Low, Close, and VWAP. In Table 4 are summarized the selected features that will be fed into the models in order to predict the target variable, which is the percentage change of the volume.

Variable type	Feature
dependent	% change of stock Volume
independent	Adj Close stock price
independent	Price variation of stock
independent	Volume of stock
independent	Sentiment score
independent	Adj Close price Nasdaq100 index
independent	Volume of Nasdaq100 index
independent	Vix index
independent	OBV indicator
independent	RSI indicator
independent	EMV indicator
independent	ATR indicator

Table 4. List of the features that will be used for the present work.

3.6 Models

This chapter presents the predictive models chosen for this research and their basic theoretical background.

The models range from simplest predictive model, such as Simple Moving Average (SMA) model and Autoregressive (AR) model, which are considered the benchmark model, to more advanced models related to ML and DL field.

The ML models used in the present work fall all into the supervised learning models.

Supervised learning is an ML paradigm in which the algorithms are trained on input data (independent variables) that has been labelled to produce a particular output (dependent variable) so that they can uncover underlying patterns and relationships between the dependent and the independent variables. In Figure 9 a schematic representation of the process involved in supervised learning.

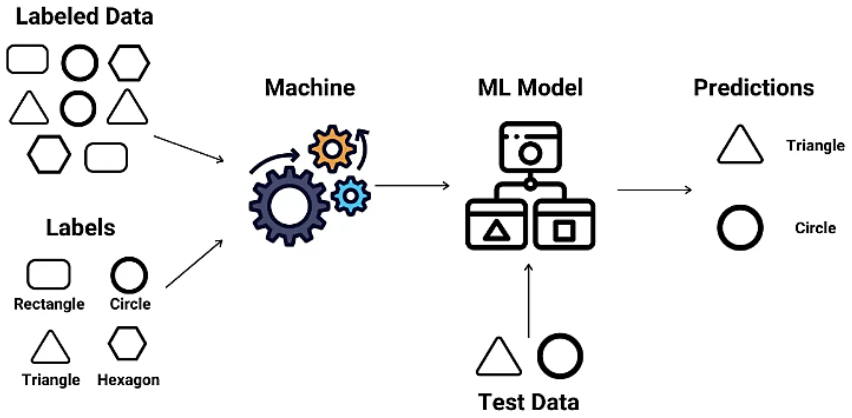


Figure 9. Schematic rapresentation of the steps inviolved in designing Supervised Learning model (source: https://medium.com/@ngneha090/a-guide-to-supervised-learning-f2ddf1018ee0)

In the present work, we will try to assess the performances of some of the most common and widely used ML models and investigate a model coverage that includes different natures of ML predictive models. That way, we could assess the suitability of the various model to the prediction problems that is being investigated.

The supervised learning models selected for this work are: Linear Regression (LR), Support Vector Regressor (SVR) and Random Forest Regressor (RFR).

In addition to the ML models, the Long Short-Term Memory Networks (LSTM) has been employed for predicting the medium-term percentage change of the stocks' volume. LSTM is part of DL models and can be regarded as an improved type of recurrent neural network (RNN) aimed at dealing with the vanishing gradient problem [49]. DL is a subset of machine learning that attempts to mimic the complex decision process of the human brain, and is inspired by the brain's neuron organisation and connectivity.

DL models are derived from Artificial Neural Networks (ANN). They consist of multilayer artificial neural networks (Figure 10a). Figure 10b shows a schematic representation of the mathematical model of an artificial neuron. The main components are the inputs (X_i), weights (w_i), bias (b), the processing element (summation function Σ and activation function f) and the corresponding output signal (y). They take input, calculate a weighted value, sum, and evaluate the result using an activation function that transforms it non-linearly.

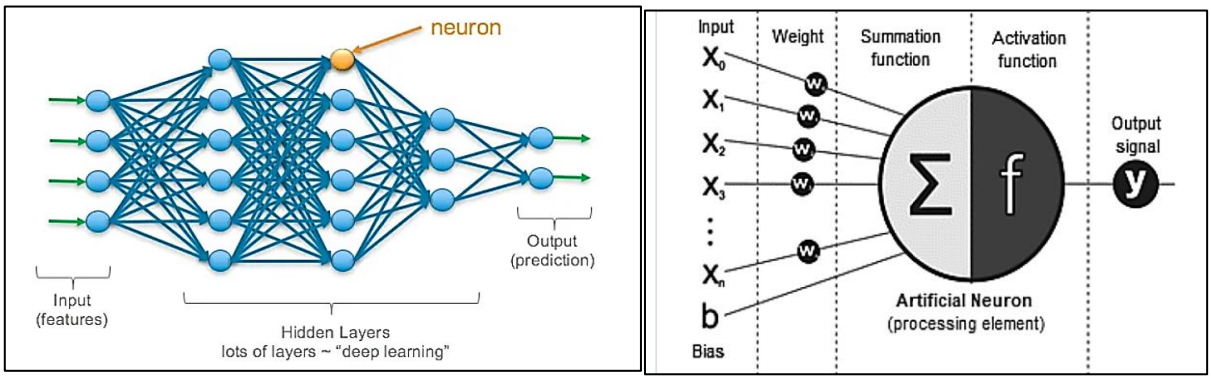


Figure 10a. Figure 10b.

Figure 10. On the left (Fig 10a) a multilayer artificial neural network architecture, On the right (Fig. 10b) the mathematical model of a single artificial neuron (source: https://link.springer.com/article/10.1007/s42979-021-00815-1)

3.6.1 Benchmark models

In order to assess the potential benefits in using ML and DL models, the latter will be compared against some naive models, the Simple Moving Average (SMA) and Autoregressive (AR), which will serve as our benchmarks.

A SMA is simply an arithmetic moving average, which is computed by adding the *n* past values of the variable being considered and then dividing that sum by the total time periods *n* considered as described by the following expression [50]:

$$SMA = \frac{X_1 + X_2 + \cdots + X_n}{n} \qquad (9)$$

Where:

X_i is the value of the variable at previous time period i and n is the total number of the past time periods considered.

There are a number of different versions of moving averages, which are popular indicators used in technical analysis. SMA is the easiest moving average to construct since it averages the values of the considered variable over the specified period.

SMA is often used as a measure of trend direction. It smooths out volatility and makes it easier to uncover the trend of the variable over the time. If the simple moving average points up, this might imply an increasing trend. On the other hand, if it is pointing down, it means a decreasing trend.

The longer the time frame for the moving average, the smoother the simple moving average will be. As per design, The SMA can be regarded as a really simple model, which implies a high degree of explainability and ease of application. However, being a simple method, it has several limitations.

It is worth to mention that since it is constructed by using past values, SMA is a lag indicator and it might not reflect current market conditions accurately. In other words, it is simply an indication of past performance and is not a predictor of future values. In addition, SMA is sensitive to spikes and this might give us false signal.

The length of the past time period considered for computing the simple moving average must be selected before the method is applied. This is usually done using the forecaster's judgement. For the present work the length of the time window is set to 30 days.

The second benchmark model is the Autoregressive model (AR), which is a fairly used model in the field of time series forecasting.

The main assumption underlying AR is that the current value of a time series depends linearly on its own previous values and on a stochastic error term. The model turns out being powerful and effective when it is applied to time series data where current observations are correlated with past observations [51]. The mathematical expression of the $AR(p)$ of order p can be written as:

$$X_t = \sum_{i=1}^{p} \varphi_i X_{t-i} + \varepsilon_t \quad (10)$$

Where $\varphi_1, \dots, \varphi_p$ are the parameters of the model, and ε_t is the white noise (independent and identically distributed with a mean of zero).

Unlike standard regressive models, the term autoregression indicates that it is a regression of the variable against itself.

It is worth noting that, similarly to many other time series models, stationarity is a basic assumption of the AR model.

In addition, another assumption of AR model is that future patterns might mirror past trends, and, under this assumption they can give us valuable insights for market predictions, although their accuracy can be limited during volatile conditions like financial crises, where historical patterns may not be valuable anymore.

One common approach to decide about order p of the AR model is assessing the Autocorrelation Function (ACF) and Partial Autocorrelation Function (PACF). The ACF describe the correlation of a time series with itself at different lags, while the PACF provide the same information, but after removing the effects of the previous lags.

From Figure 11 to Figure 14, the plots of both ACF and PACF for the time series of the volume percentage change for the stocks of TSLA and APPL are displayed.

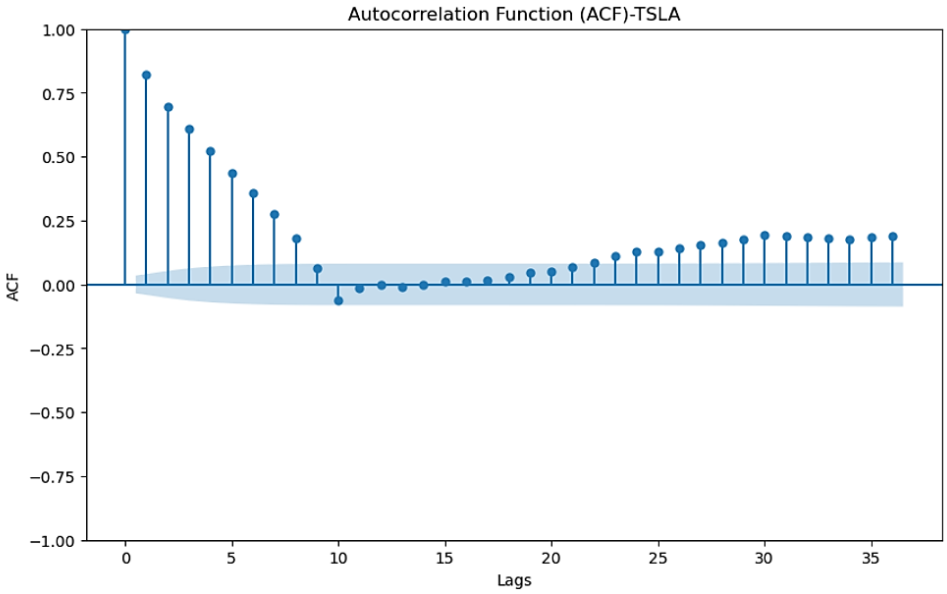


Figure 11. ACF plots for the Time Serie of the pct change of volume of TSLA

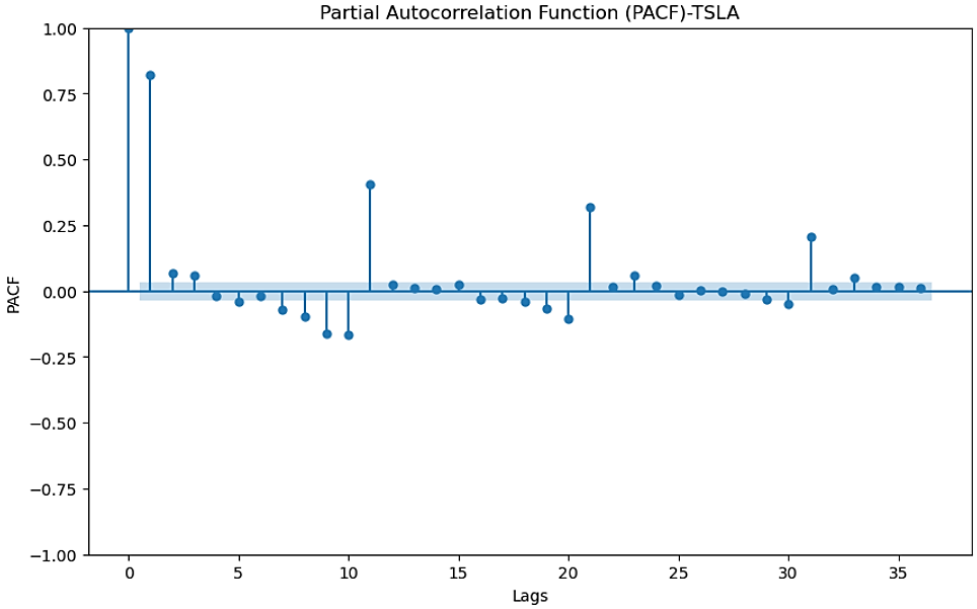


Figure 12. PACF plots for the Time Serie of the pct change of volume of TSLA

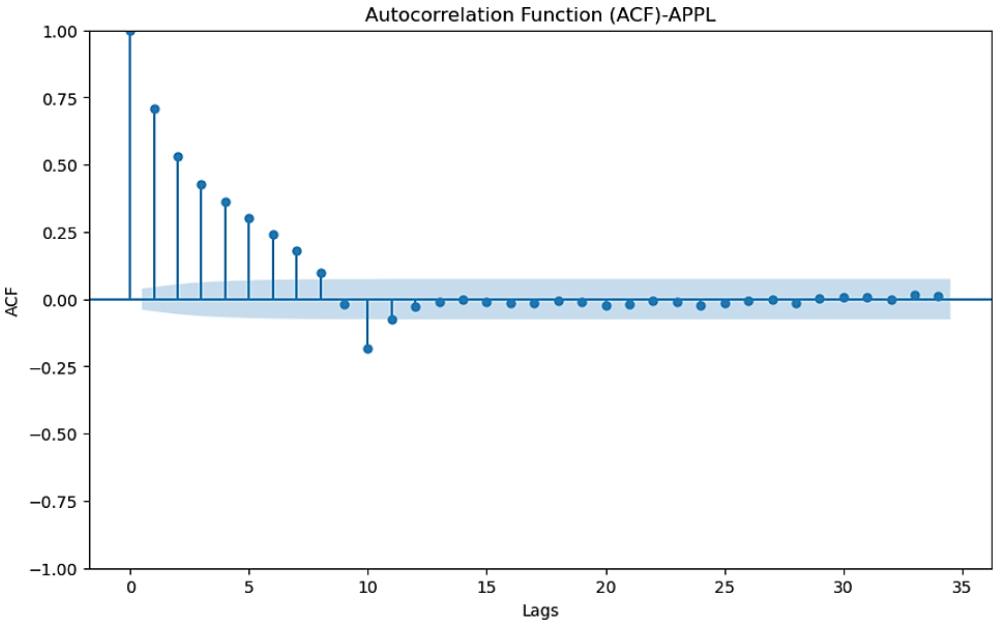


Figure 13. ACF plots for the time Serie of the pct change of volume of APPL

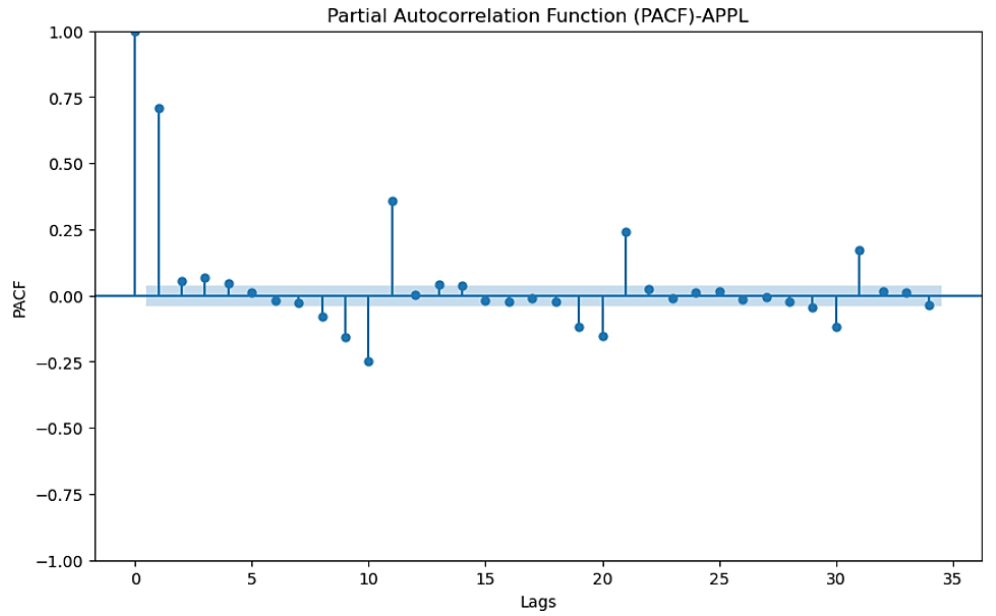


Figure 14. PACF plots for the Time Serie of the pct change of volume of APPL

Looking at the figures, it is observable for both of the stocks an approximately geometric decay of ACF and two significant terms in the PACF plot. This would suggests to design an AR(2) model.

3.6.2 Linear Regression model

According to J. M. Stanton (2001) [52] the concept of linear regression is dated back to 19th century, during which Sir Francis Galton applied it to problems of heredity. Nowadays, Linear Regression analysis is widely employed in numerous areas of research.

Linear Regression (LR) is a supervised learning approach in which the given set of data is labelled. The main goal of LR models is building a relationship between a dependent variable and one or

more independent variables. To achieve this, LR models attempt to find a line that best fits a given set of data points, usually by minimising the sum of squared prediction errors [53].

The regression analysis performed by using only one independent variable is called a simple regression. If more independent variables are considered, the regression analysis is called multiple regression, and its mathematical expression can be written as:

$$y = \varphi_0 + \varphi_1 x_1 + \varphi_2 x_2 + \cdots + \varphi_i x_i \qquad (11a)$$

$$\hat{y} = y + \varepsilon \qquad (11b)$$

where the φ_i are regression coefficients that describe the starting point and slope of the fitting line. The term ε is the residual (the difference between the predicted value y and the actual observation $\hat{y}$.) and it is supposed to be normally distributed with mean 0 and variance σ^2,

The term linear is not meant to $\hat{y}$ as a linear function of the x_i, but to the linearity of the parameters $\varphi_0,\ \varphi_1, \varphi_2, \ldots, \varphi_i$ [53]. In this way, the fitting line drawn by linear regression does not necessarily have to be straight, but can be fitted to curves in the data by including non-linear predictors.

The aim in LR is to compute the values of the coefficients φ_i, such that the error term ε is minimized, and y will be as much close to $\hat{y}$ as possible. In Figure 15 is displayed a simple linear regression model with one independent variable.

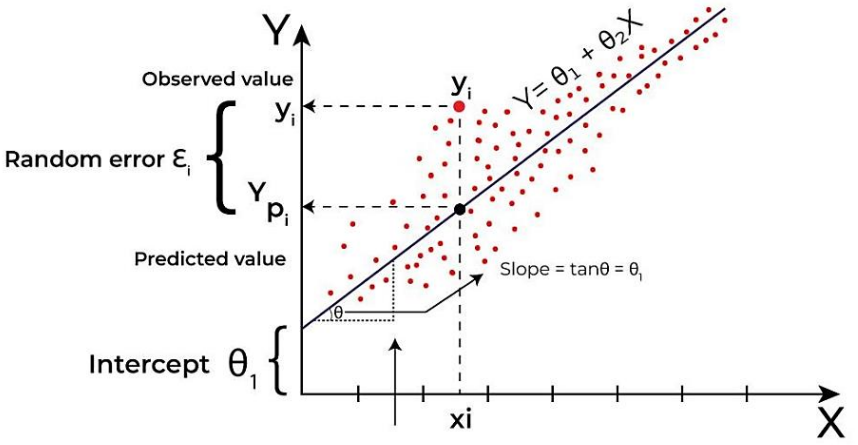


Figure 15. Representation of a simple linear regression model with one independent variable (source: https://www.geeksforgeeks.org/ml-linear-regression/)

The most commonly used method is ordinary least squares (OLS), which aims to minimise the sum of squares of the residuals, i.e. to find a set of coefficients that minimises the difference between the predicted value and the actual value.

By using a matrix form of the expression (11b) and solving for ε,

$$\varepsilon = \boldsymbol{Y} - \boldsymbol{X}\beta \qquad (12)$$

We want to minimize the sum of the squared residual

$$\sum \varepsilon_i^2 = [\varepsilon_1 \varepsilon_2 \dots \varepsilon_n]^T [\varepsilon_1 \varepsilon_2 \dots \varepsilon_n] = \varepsilon^T \varepsilon = (\boldsymbol{Y} - \boldsymbol{X}\beta)^T (\boldsymbol{Y} - \boldsymbol{X}\beta) \qquad (13)$$

Where the symbol "T" denotes the transpose of the matrix. In order to minimize the above expression (13), the derivatives with respect to the vector β are considered and set equal to 0. By solving for β adn after some mathematical passages, the final solution vector β is obtained:

$$\beta = (X^T X)^{-1} X^T y \qquad (14)$$

In order to ensure the validity of the linear regression model and the properties of the best linear some assumptions need to be met [54]:

- There is a linear relationship between independent variables and dependent variables.
- The sample observations are independent of each other
- The error term has the same variance (σ^2) for all independent variable levels.
- The error terms are normally distributed.
- There is no complete or highly linear relationship between the independent variables

With respect to the present work, the model employed is a multiple Linear Regression, whose target variable is the medium-term volume percentage change and independent variables are the ones defined in Section 2.6.

3.6.3 Support Vector Regressor model

Support Vector Regression (SVR) is an extension of the Support Vector Machine (SVM), which was developed in 90's by Vapnik and his colleagues [55]. The promising results obtained in real world problems made the underlying principle of SMV be subsequently extended to regression problems [56].

Unlike SVM model, which is designed for classification problems, SVR is designed to predict continuous numerical values, making it suitable for tasks such as time series forecasting, stock price prediction and more.

SVR uses the same concept of a hyperplane and margin as SVM, although some differences are present between their definitions (Figure 16). In SVR, the margin is defined as the error tolerance of the model. This is also referred to as the ε-insensitive tube (Figure 16b).

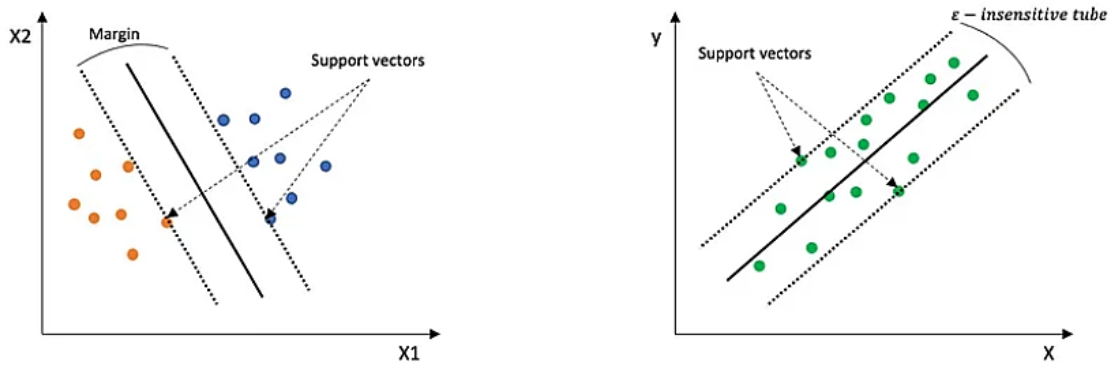


Figure 16a

Figure 16b

Figure 16. On the left (Fig 16a) the margin representation of SVM model. On the right (Fig. 16b) the representation of the tolerance margin of a SVR model (source: https://medium.com/@niousha.rf/support-vector-regressor-theory-and-coding-exercise-in-python-ca6a7dfda927)

The target of the SVR model is finding the hyperplane that best fit the data falling in the ε-insensitive tube

Minimising the coefficient w of the model, rather than the squared error, is the objective function of the SVR model. As shown in the Figure 17 (left) below, the SVR model uses an error term, denoted ε, in the problem constraint.

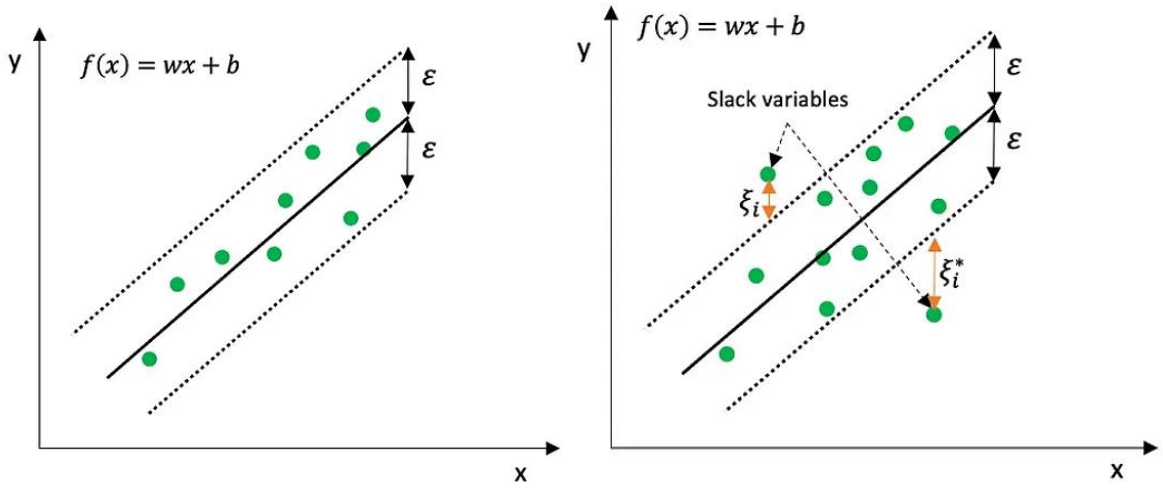


Figure 17. Mathematical description of the tolerance margin of a SVR model. On the left SVR with zero tolerance for error, on the left SVR allowing some errors with slack variables. (source: https://medium.com/@niousha.rf/support-vector-regressor-theory-and-coding-exercise-in-python-ca6a7dfda927)

This constraint states that the absolute error must be less than or equal to ε. The value of ε can be adjusted to achieve desired regression model accuracy.

The simplified form of the constraint optimisation problem of the SVR can be expressed in the following way [57]:

$$minimize: \left(\frac{1}{2}\|w\|^2\right) \qquad (15a)$$

$$given: |y_i - f(x_i)| \leq \varepsilon \; and \; f(x_i) = w_i x_i \qquad (15b)$$

In order to consider error values greater than ε for the points outside the margins, slack variables might be introduced (Figure 17, right).

The point falling out the margin ε are charged the cost C, and the new version of the constraint optimisation problem for SVR is described below:

$$minimize: \left(\frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}|\xi_i + \xi_i^*|\right) \qquad (16a)$$

$$given: |y_i - f(x_i)| \leq \varepsilon + |\xi_i| \; and \; |f(x_i) - \; y_i| \leq \varepsilon + |\xi_i| \; and \; f(x_i) = w_i x_i \qquad (16b)$$

The additional cost parameter C allows some tuning in order to manage the tolerance to points outside of ε. It is worth to mention that the tuning of the parameter C triggers a trade-off between the flatness of the function f and the amount up to which deviations larger than ε are tolerated. As C decreases, the tolerance for points outside ε increases, and as the value of C approaches zero, the constraint collapses into the simplified form.

A relevant building block of SVR are the kernel functions, which allow the model to be suitable for both linear and non-linear regression problems. The kernel function, in cases where the data is non-linearly separable, helps in finding function $f(x)$ in higher dimensional space where a linear regression problem can be solved. Kernel functions are powerful tool that allow to resolve in easier way regression problems that might show complex relationships. Some of the most common kernel functions used in SVR are linear, polynomial, radial basis function (RBF), and sigmoid.

With respect to the SVR model used in the present work, the RBF kernel function has been selected. The RBF kernel is well-designed for non-linear problems and it is widely used in SVR models. It provides good results especially when there is no prior knowledge of the data, allowing to capture complex relationships between data.

Below the RBF mathematical function and its two-dimensional plot in Figure 18.

$$k(x_1, x_2) = \exp(-\gamma\|x_1 - x_2\|^2) \qquad (17)$$

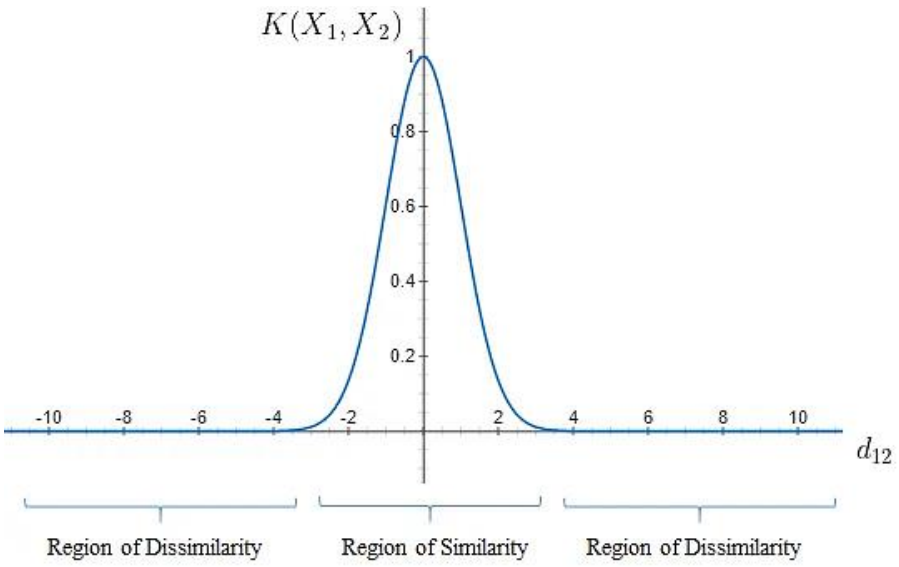


Figure 18. RBF kernel function representation (source: https://towardsdatascience.com/radial-basis-function-rbf-kernel-the-go-to-kernel-acf0d22c798a)

3.6.4 Random Forest Regression model

The Classification and Regression Tree model has been introduced by Leo Breiman in his book *Classification and Regression Tree* in 1984 [58].

The building block of Classification and Regression tree are the decision trees, which can be used to design model for classification and regression problems. Over the time, more and more different approaches have been proposed largely in the field of Classification and Regression Tree.

Random Forest Regression, which is being employed for forecast purpose in the present work, is part of the supervised learning paradigm of ML. RF models for regression consist of growing trees depending on numerical values as opposed to class labels related to Random Forest Classification. A regression tree (RT) can be thought of as a set of conditions that are structured in a hierarchical manner. They are successively applied from a root to a terminal node or leaf of the tree [59].

The Figure 19 display the schematic structure of a Random Forest model, which combines several simple trees that run in parallel with no interaction amongst them.

Each tree in the forest is built from a different subset of data and makes its independent prediction. The final prediction is based on the average or weighted average of all individual predictions.

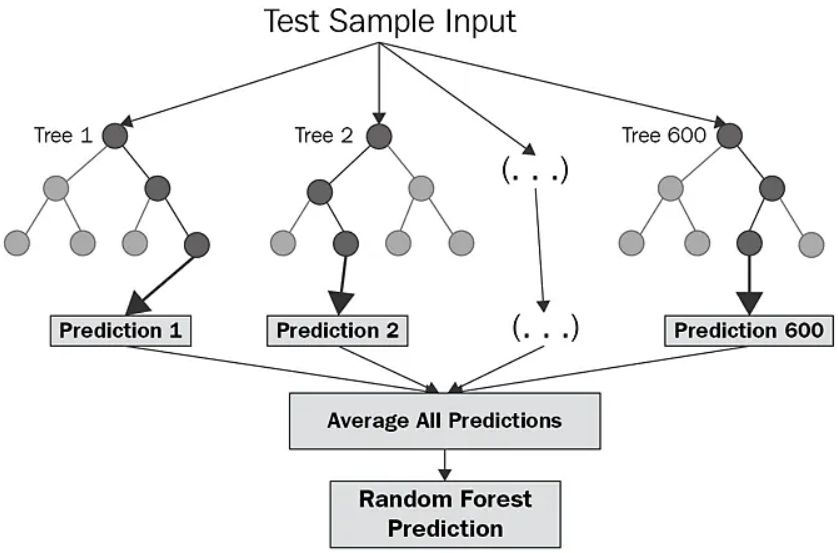


Figure 19. Schematic representation of Random forest (source: https://corporatefinanceinstitute.com/resources/data-science/random-forest/)

In other words, the rationale behind the random RFR is to combine multiple decision trees in determining the final output rather than relying on individual decision trees. This method is known as Ensemble learning, in which multiple models are trained independently on random subsets of the training data. By averaging the predictions from the individual models, it is expected to obtain a more accurate prediction than a single model. The main advantages of using this technique are the reduction of variance by averaging multiple samples, the improvement of accuracy when using random features, and continuous estimation of the generalization error of the combined ensemble of trees, as well as estimates for the strength and correlation.

The Random Forest Regression model is used to predict continuous values, such as stock price forecasting, time series forecasting, selling price forecasting, etc. Some of the main advantages in using RFR are its ability to deal with large datasets and capture non-linear relationships between input and target variables.

Furthermore, unlike other regression models like LR, Random Forest makes no assumptions about the underlying data distribution and can handle nonlinear relationships better. Finally, it is less prone to overfitting because it can randomly select different subsets of the data to train on and average its results.

On the other hand, a relevant limitation of the RFR might be its low interpretability, which could lead to some issues especially when it is employed in the industry.

The split criterion in decision trees plays a relevant role and depending on the nature of target variable, we have different criterions for evaluating the splitting of the nodes in decision tree. A decision tree makes decisions by splitting nodes into sub-nodes to create relatively pure nodes. This process is performed multiple times in a recursive manner during the training process until only homogenous nodes are left.

For Regression trees some of the common criterion used for node splitting are MSE, MAE log_loss and SSE.

For the present work, the criteria used is the Sum of Squared Error (SSE), which is the default method that is set into the python sklearn package.

Assuming that we are required to split the dataset S into two groups S_1 and S_2 so that the selection of S_1 and S_2 needs to minimize the sum of the squared errors [60]:

$$SSE = \sum_{i \epsilon S_1} (y_i - \bar{y}_1)^2 + \sum_{i \epsilon S_2} (y_i - \bar{y}_2)^2 \qquad (18)$$

Where $\bar{y}_1$ and $\bar{y}_2$ are the average of the sample in S_1 and S_2. The way the regression tree grows is to automatically decide on the split variables and split points that can achieve maximum SSE reduction. In addition to set the splitting criteria, the following parameters have been defined in building the RFR used for this research work:

- **n_estimators:** it refers to the number of trees included in the forest. The selected number for the model used in this work is 200.
- **random_state**: it refers to the seed for random number generator. Setting the *random_*state ensures that the same tree is selected each time model.fit() is called. It is worth to set this hyperparameter since the training procedure of trees is inherently random. The selected number for the model used in this work is 42.

3.6.5 Long Short-Term Memory Networks

Long Short-Term Memory (LSTM) is a type of artificial recurrent neural network (RNN) architecture used in the field of deep learning, and was introduced by Hochreiter and Schmidhuber in 1997 [61]. Unlike RNNs that have no capacity to find long-term time dependencies, LSTMs have feedback connections, which allows them to learns what data to forget and what data to remember when predicting. LSTMs are designed to exploit temporal dependencies across sequences of data and this proves to be useful and precious when dealing with real case problems that require the time dependencies from the data not only from recent time but also from further back.

In addition to this, another powerful feature of the LSTMs comes from controlling and memorizing information over long sequences is their ability to deal with the problem of vanishing or exploding gradients., which can occur when training traditional RNNs on sequences of data.

Unlike traditional RNNs, where each cell has an extremely simple structure, LSTMs have gates that can regulate the flow of information. Instead of having a single neural network layer like traditional RNNs, LSTMs have four layers interacting in a very special way.

In Figure 20 and 21 a schematic representation of the architecture of a RNN and LSTM, respectively [66].

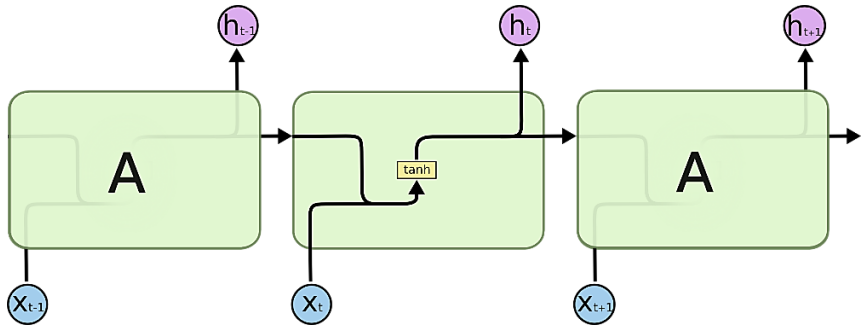


Figure 20. The repeating module in a standard RNN contains a single layer. (source: [66])

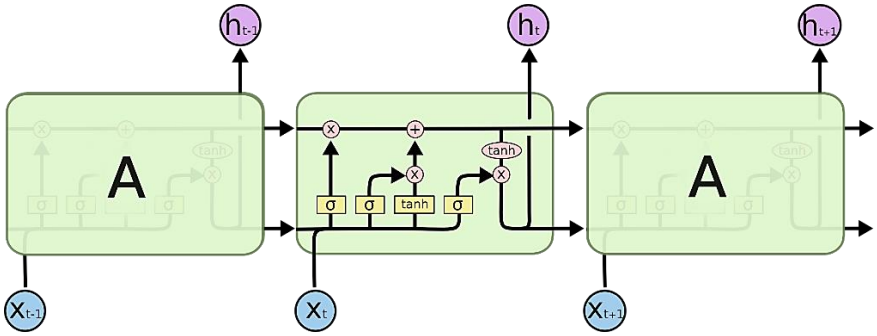


Figure 21. The repeating module in an LSTM contains four interacting layers. (source: [66])

A typical LSTM cell has three gates: forget, input, and output. In Figure 21 the aforementioned gates are represented by the three sigmoid layers. The basic functions of each gate can be described as follows:

- The forget gate detects the previous information that is not needed and drop that information from the cell state.
- The input gate feeds the cell with new information.
- After the above-mentioned steps, the cell state contains information about what should be forgotten or kept from prior steps and what should be added as useful new information. The output gate controls how much of the new memory cell's content should be used for making a prediction.

The gates allow LSTM networks to store, update and retrieve information in a selective manner over long sequences, making them particularly effective as model for problems that require mode long-term dependencies. Common use of LSTMs are speech recognition, language translation, and sentiment analysis.

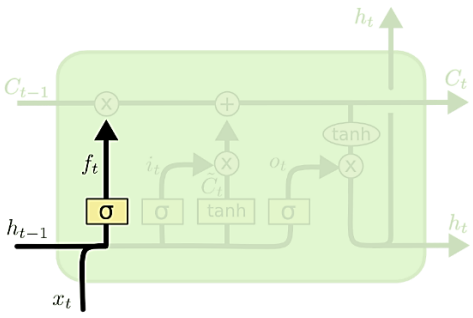
As displayed in Figure 21, each gate (forget, input and output) receive the same inputs, which are the input of current timestep (x_t) and the output from the previous time period (h_{t-1}).

The cell state C_t runs through the LSTM cell with only minor linear interactions with the gates and its main task is to convey the memory of the cell.
In the following, the LSTM shown in Figure 21 has been deconstructed in order to describe in more detail how it works.
From Figure 22 to Figure 25, the mathematical notations can be described as follows:

- W_f, W_i, W_C, W_o: they refer to the weight matrices associated with gates and cell state.
- b_f, b_i, b_C, b_o: they refer to the biases associated with gates and cell state
- σ: it refers to the Sigmoid activation function, which returns outputs with values between 0 and 1 for any given input.
- $tanh$: it refers to $tanh$ activation function, which returns outputs with values between -1 and 1 for any given input. It has a steeper gradient as compared to sigmoid.
- $*$: it refers to Hadamard product, an operation used to multiple two matrices of the same dimensions.

Forget gate and Information to be dropped at each time period

As already mentioned, the main task of the forget Gate (f_t) is selecting what part of information coming from the previous timestep (C_{t1}) must be forgotten. The sigmoid activation used during this phase return output values between 0 and 1. 1 implies to completely preserving the information, while 0 implies to discard the previous information. In Figure 22 is displayed the forgot process along with its mathematical equation.



$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] \; + \; b_f\right)$$

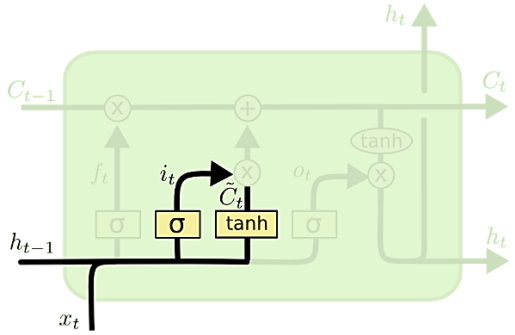
Figure 22. Representation of forget gate along with its mathematical equation. (source: [66])

Input gate and Information to be conveyed into the next time period

Once the forget gate processed the information from previous period and eliminated the unnecessary data, the input gate evaluates what data should be saved to the cell state and carried to the next timestep. This process consists of two parts (Figure 23):

1) the input gate (i_t), relying on a sigmoid layer, determines what data that is present in the cell state must be updated and carried forward to the next timestep.

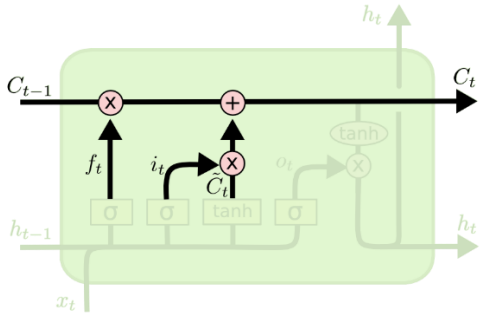
2) The second part is performed through a $tanh$ layer, which creates a vector $\tilde{C}_t$ containing new information. The new information can be added to the current cell state through the $tanh$ activation function which filters what new information can be added or discarded.



$$i_t = \sigma\left(W_i \cdot [h_{t-1}, x_t] \; + \; b_i\right)$$
$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] \; + \; b_C)$$

Figure 23. Input Gate and new values added to cell state alongside with their mathematical expression. (source: [66])

Subsequently, as displayed in Fig 24, the final information carried to the next timestep is the overall result that comes from the outputs of the input gate, the new values added to the cell state, the forget gate and the cell state from the previous timestep.

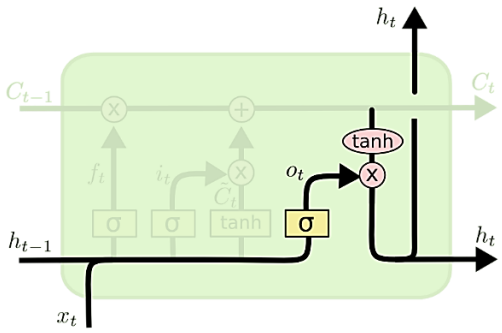


$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

Figure 24. Cell state computation. (source: [66])

Output at each timestep

The Output Gate (o_t) and the Cell state select the output at each period. The output gate, by using sigmoid activation function, identify which parts of the cell state can be considered as output. Then, the cell state C_t is passed through a $tanh$ and it is multiplied by the output previously identified in order to determine the hidden state for the next LSTM cell (h_t). The process involing the Output Gate is schematically displayed in Figure 25.



$$o_t = \sigma\left(W_o\left[h_{t-1}, x_t\right] + b_o\right)$$
$$h_t = o_t * \tanh\left(C_t\right)$$

Figure 25. Output computation. (source: [26])

The main architecture of the LSTM designed for the present work consists of:

- ***1 Input Layer*** *(11 input features, 64 neurons, activaction function=relu)*
- ***1 Hidden Layer*** *(32 neurons, activaction function=relu)*
- ***1 Dropout Layer*** *(pct=0.2)*
- ***1 Output Layer*** *(1 neuron, activaction function=no)*

The dropout layer entered into the LSTM architecture helps layer to reduce the overfitting of the model [67]. During the training, the Dropout process randomly selected neurons that are ignored, which means that their contribution to the activation of downstream neurons is temporally removed. In the present case, the value 0.2 implies that 20% of the neurons are randomly excluded from each update cycle.

The Output Layer, unlike the other layers, is a simple non-recurrent neural network layer with 1 neuron and no activation function, whose functionality is to return our output, namely the predicted volume percentage change.

With respect to the activation function, the rectified linear activation function (ReLU) has been selected. The ReLU function is widely used in neural networks since offers important advantages. It is simple to be implemented and above all, unlike other activation functions, it is less sensitive to one of the main issues during the training phase, which is the vanishing gradients [68].

Another important element to be defined in neural network architectures is the optimizer, which is a crucial component that help neural networks learn efficiently and converge to optimal solutions.

The selected optimizer for the LSTM model adopted in this work is Adam, which stands for Adaptive Moment Estimation and is one of the most popular optimization algorithms used in training deep neural networks. Adam is able to convey together the advantages of the “gradient descent with momentum” and the “RMSP” algorithms [69]. It calculates individual learning rates for different parameters using an adaptive learning rate method. Some of the main advantages in using *adam* optimizer are easy implementation, efficiency in term of computation and memory usage and suitability for problems with a large amount of data or parameters,

3.7 Evaluation metrics

In literature, in order to evaluate the accuracy of predictive model, several common metrics are available. However, certain metrics might be more suitable to be employed in financial time series forecasting [70]

Most of the metrics evaluate the model accuracy in prediction in terms of the difference between the actual and predicted values, while others assess the ability of the models to correctly predict the trend/direction in the series.

Each of these performance measures has its own advantages and its own limitations, therefore it is advisable to not rely on a single measure for a particular forecasting problem.

In the present work, three different evaluation metrics are considered: the Root-Mean-Squared Error (RMSE), Mean Absolute Error (MAE) and Mean directional accuracy (MDA).

Root-Mean-Squared Error (RMSE) and Mean Absolute Error (MAE)

Widely used measures for evaluating models are the Root-Mean-Square-Error (RMSE) and the Mean Absolute Error (MAE).

Given n observations y with and n corresponding model predictions, the RMSE and MAE are:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \qquad (18)$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \qquad (19)$$

The RMSE is the square root of the mean squared error (MSE). The square root allows to obtain a metric with the same units of the variable investigated. The MSE and MAE can be regarded as the Euclidean and Manhattan distance, respectively [71].

Due to the squaring operation, RMSE relevant weight is given to larger errors, which leads to higher sensitivity to outliers. It implies that RMSE can prove very useful when large errors are particularly undesirable.

Contrary to RMSE, MAE is a linear score, i.e. all the errors are weighted equally. This makes MAE less sensitive to outliers [71].

The RMSE will always be larger or equal to the MAE. The magnitude of the difference between RMSE and MAE can provide further information: the greater difference between them, the greater the variance in the individual errors in the sample. If the RMSE equals to MAE, then all the errors are of the same magnitude

Mean directional accuracy (MDA)

Another useful metric used for measuring the prediction accuracy is the Mean directional accuracy (MDA). The MDA is mainly employed to assess the accuracy of forecast direction. It compares the forecast direction (upward or downward) against the actual realized direction.

MDA is a popular metric for forecasting performance in economics and finance [72], where we are often interested in the directional movement of variables of interest. It is defined by the following formula:

$$\frac{1}{N}\sum_{t}\mathbf{1}_{sgn}(A_t - A_{t-1}) = sgn(F_t - A_{t-1}) \qquad (20)$$

where A_t is the actual value at time t and F_t is the forecast value at time t. Variable N represents number of forecasting points. The function $sgn(\cdot)$ is sign function and 1 is the indicator function.

MDA assesses the directional accuracy and can be regarded as a binary evaluation. The metric only considers the upward or downward direction in the time series, irrespective of the quantitative value of increase or decrease.

3.8 Feature Importance

Feature importance analysis is performed to assess the contribution of each feature to the model prediction. Each feature is assigned with a score, which represent its importance in prediction of the target variable. The higher the score, the larger the effect of the feature on the model that is used to predict a certain variable

Depending on the predictive model employed, there might be many ways of calculating feature importance.

3.8.1 Linear Regression Feature Importance

One of the main advantages of using LR model is its high interpretability. As described in Section 2.5.1, the main goal is to obtain the coefficients of expression (11a) so that the line can best fit the observed data. These coefficients can be regarded as a measure of the relationship between each independent variable and the dependent variable and can be used to identify the importance of the features [73]. The absolute values of the coefficients are ranked and the features that have the largest magnitude are supposed to contribute more to the model prediction.

3.8.2 Support Vector Regression Feature Importance

For the Support Vector regression, one common method used to evaluate the feature importance is based on permutation technique. Permutation feature importance method was first introduced by by Breiman in 2001 for Random Forest models [74].

The underlying logic of the Permutation feature importance method is pretty straightforward. The importance of the features is computed by assessing the increase in the error of the prediction after permuting the feature. If permuting the feature an increase of the error prediction is observed, it implies a relevant importance of the feature since the model relies on it for the prediction. On the other hand, if the error remains unchanged, it implies low importance of the feature since the model did not take the feature into account for the prediction.
Below is presented the algorithm outline for the feature importance calculation [75]:

- The inputs are the predictive model m and the dataset (training or validation) in tabular format D.
- Computation of the reference score s (for example, accuracy for a classifier or R^2 for a regressor) of the model m on data D.
- For each feature j of the dataset D:
    - For each repetition k in $1, \dots, K$:
        - Randomly shuffle column j of dataset D to generate a corrupted version of the data named $\widetilde{D}_{k,j}$.
        - Compute the score $s_{k,j}$ of the model m on corrupted data $\widetilde{D}_{k,j}$.
    - Compute importance i_j for feature f_j:

$$i_j = s - \frac{1}{K}\sum_{k=1}^{K} s_{k,j} \qquad (21)$$

For the present work, the built-in function ‘permutation_importance’ from the sklearn Python package is used to perform the feature importance analysis.

3.8.3 Random Forest Regression Feature Importance

The logic behind the feature importance analysis in decision trees is based on the decrease in node impurity weighted by the probability of reaching that node. The above probability can be easily calculated by dividing the number of samples that reach the node by the total number of samples. The criteria used to assess the decrease in node impurity can be based on Gini or entropy measurements.
The built-in function ‘feature_importances’ of the Scikit-learn package is based on above mentioned principles [76] and is utilized for computing the feature importance with respect to the Random Forest Regression model. In the following the main steps for measuring the feature importance in Random Forest trees.
For each decision tree, Scikit-learn calculates a nodes importance using Gini Importance, assuming only two child nodes (binary tree):

$$ni_j = w_j C_j - w_{left(j)} C_{left(j)} - w_{right(j)} C_{right(j)} \quad (22)$$

Where ni_j is the importance of node j, w_j is the weighted number of samples reaching node j, C_j is the impurity value of node j and the terms with the $left$ and $right$ pedix refers to the child node from the left and right split on node j, respectively.

The importance for each feature on a decision tree is then calculated as follows:

$$fi_i = \frac{\sum_{j:node\ j\ splits\ on\ feature\ i} ni_j}{\sum_{k\ \epsilon\ all\ nodes} ni_k} \quad (23)$$

Where fi_i is the importance of feature i and ni_j is the importance of node j.

fi_i is then normalized to a value between 0 and 1 by dividing by the sum of all feature importance values:

$$normfi_j = \frac{fi_i}{\sum_{j\ \epsilon\ all\ features} fi_j} \quad (24)$$

The final feature importance, at the Random Forest level, is it's average over all the trees. The sum of the feature's importance value on each tree is calculated and divided by the total number of trees:

$$RFfi_i = \frac{\sum_{j\ \epsilon\ all\ trees} normfi_{ij}}{T} \quad (25)$$

Where $RFfi_i$ is the importance of feature i calculated from all trees in the RF model, $normfi_{ij}$ is the normalized feature importance for i in tree j and T is the total number of trees.

3.8.4 LSTM Feature Importance

With respect to the LSTM, one available way to evaluate the contribution of the features to the prediction is based on the SHAP (SHapley Additive exPlanations) method. Similar to the previous feature importance method, SHAP assigns each feature an importance value representing its contribution to the model's output.

SHAP method is model-agnostic, which implies that it can be used to interpret any predictive model. The method is inspired by the coalitional game theory, where the feature values of an instance act as a player in coalition. The Shapley values computed provide information about how to fairly distribute the "pay-out" (prediction) among the features and represent the average magnitude of each feature across all the possible combination of features [77].

In SHAP method, the values are obtained by comparing the predictions of the model with and without a specific feature and this process is performed iteratively for each feature and each sample of the data set.

By representing the Shapley value as a linear mode, SHAP is able to connect LIME and Shapley values. The SHAP compute the feature explanation as follows:

$$g(z') = \varphi_0 + \sum_{j=1}^{M} \varphi_j z_j' \qquad (26)$$

Where g is the explanation model, z' is the coalition vector, M is the maximun coalition size and φ_j is the Shapley values for a feature.
In the following some of the main features of SHAP method are presented:

- **Local accuracy:** by adding up to the difference between the expected and the actual output of the model, SHAP values can return an accurate and local interpretation of the model prediction for a given input.
- **Missingness:** when a feature is missing or irrelevant, SHAP assign a value of zero, which makes the method insensitive to missing data and ensures that irrelevant features do not distort the interpretation.
- **Consistency:** SHAP values are insensitive to model changes, which implies that the computed SHAP values return a consistent and robust interpretation of the prediction of the model, irrespective of potential change in parameters or architecture of the model.

For the present work, in order to compute the feature importance for the LSTM model, the built-in shap package of Python is used.

CHAPTER 4

Results and Discussion

In this chapter we describe the results and performances obtained from the implementation of the ML and DL models selected for the present research work, including the accuracy metric measurement and the Feature Importance analysis.

We start with the results related to the benchmark models and then the results for both the experiment 1 (sentiment analysis information not included) and experiment 2 (sentiment analysis information included) are described.

For each model, the prediction curve is plotted against the actual volume percentage change recorded for the period considered, the values of the accuracy metrics (MAE, RMSE and MDA) are reported and the histograms containing the feature importance scores are displayed.

4.1 Results related to benchmark models

In Figures 26 and 27 the prediction results of the volume percentage change for TSLA and APPL by using SMA are displayed, respectively.

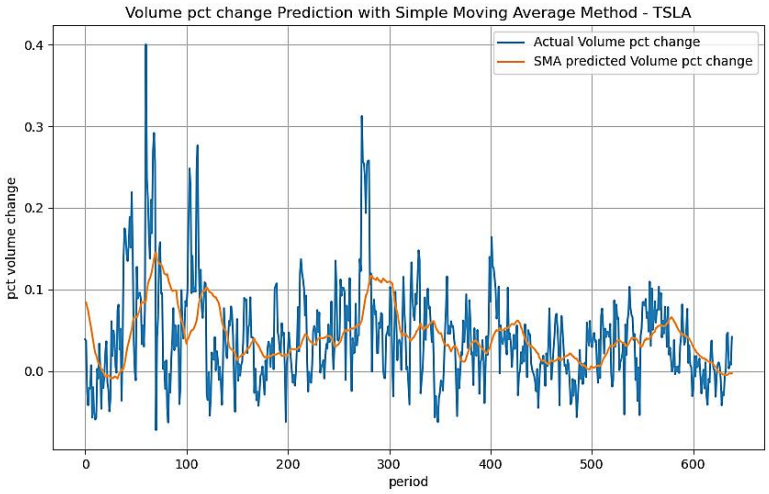


Figure 26. TSLA percentage volume change prediction through SMA model.

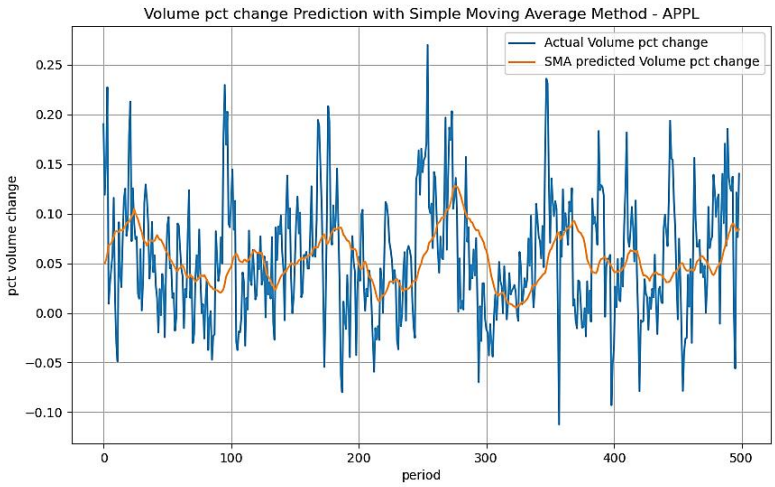


Figure 27. APPL percentage volume change prediction through SMA model.

Looking at both of figures, it is observable that the SMA model is able to reflect the main trends of the time series related to the actual values, although it is not able to predict all the abrupt direction changes. The latter is to be expected since the designed SMA considers 30 previous period, which leads to smooth out the volatility. The longer the time frame for the moving average, the smoother the simple moving average will be.

The above conclusions are confirmed by a low value of MDA (Table 5), which measures the accuracy of the direction forecast. Although the main trends in the time series are detected, the values of 0,403 and 0,410 imply that less than half of variable directions have been properly forecasted.

Performance metric	TSLA	APPL
Mean Absolute Error (MAE)	0.044	0.047
Root Mean Squared Error (RMSE)	0.059	0.060
Mean Directional Accuracy (MDA)	0.403	0.410

Table 5. Performance metrics for the SMA model.

Moving on the other performance metrics reported in Table 1, the SMA model obtains slightly better performance for the TSLA asset than the APPL asset. The magnitude of the error prediction, both for MAE and RMSE, is far lower than the actual average volume percentage change recorded for TSLA, which is around 10.5%. Conversely, the accuracy metrics for APPL stocks show a magnitude of the error close to the actual long-term average volume percentage change for APPL, which is about 5.2%, which suggests poor prediction ability. The higher difficulty in predicting the volume change for APPL might be explained by the nature of the its time serie. Unlike TSLA, where more marked trends are observable, APPL time serie still shows resemblance to a white noise.

In Figures 28 and 29 the prediction results by using the AR model are displayed for TSLA and APPL, respectively.

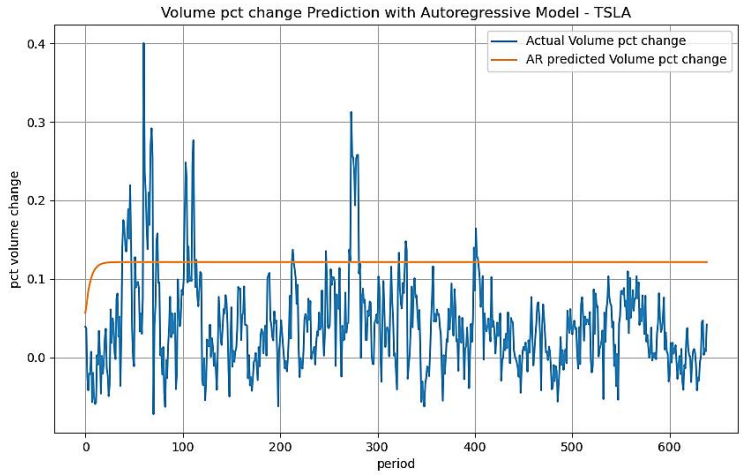


Figure 28. TSLA percentage volume change prediction through AR model.

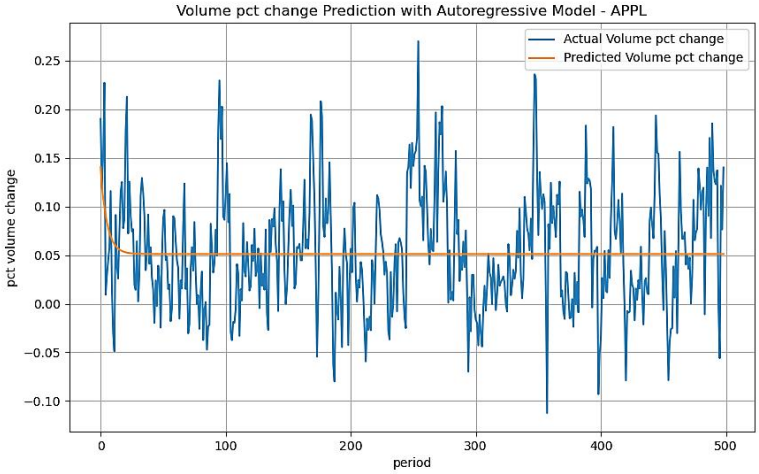


Figure 29. APPL percentage volume change prediction through AR model.

It is observable in both figures a tendency of the prediction curves to flatline quickly. The behaviour is to be expected since the AR model tends to converge to the long-term mean, with the influence of past values diminishing over time. The AR model is supposed to provide best results when there's a clear trend or seasonality in the time serie. In the present case, both TSLA and APPL time series don't show marked trend or seasonality, which might cause not optimal AR performances. The poor performances of the model are confirmed by values of the accuracy metrics shown in Table 6.

In term of both directional accuracy and prediction error, the AR shows overall lower performance than the SMA model.

Performance metric	TSLA	APPL
Mean Absolute Error (MAE)	0.090	0.047
Root Mean Squared Error (RMSE)	0.099	0.060
Mean Directional Accuracy (MDA)	0.399	0.184

Table 6. Performance metrics for the AR model.

4.2 Results related to experiment 1 – No sentiment analysis included

In this section we report the results related to the first stage of the present research work. During this phase, the sentiment analysis information is not fed into the predictive models. The independent variables that will be used to predict the target variable (volume percentage change) are adjusted close stock price, price variation of stock, volume of stock, adjusted Close price of Nasdaq100, volume of Nasdaq100, Vix index, OBV indicator, RSI indicator, EMV indicator and ATR indicator.

In the following, the results and performances of the ML and DL under the above scenario described are reported.

4.2.1 Linear Regression model – No sentiment

In Figures 30 and 31 the plot of the predictions by using the LR model are displayed for TSLA and APPL, respectively.

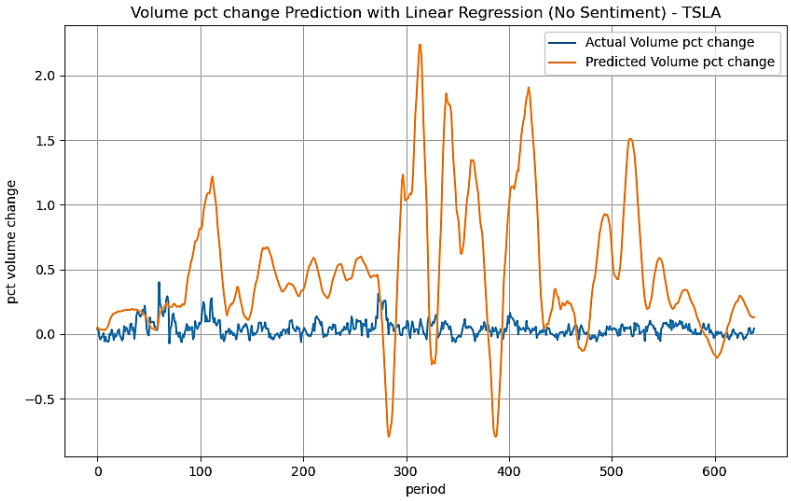


Figure 30. TSLA percentage volume change prediction through LR model – no sentiment analysis.

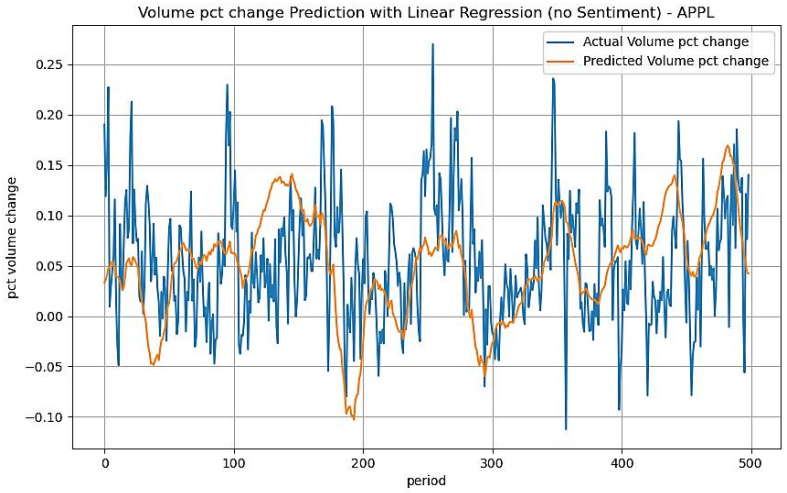


Figure 31. APPL percentage volume change prediction through LR model – no sentiment analysis.

Looking at the figures, completely different results are observable. Unlike the APPL case, where the LR model seems to provide good predictions in relation to the main trends, the TSLA case shows anomalous results with the prediction curve being not able to reflect the pattern and the trend of the actual. The root cause of such a behaviour might be explained looking at the plots of the residuals (Figure 32), which shows the difference between residuals on the vertical axis and the dependent variable on the horizontal axis.

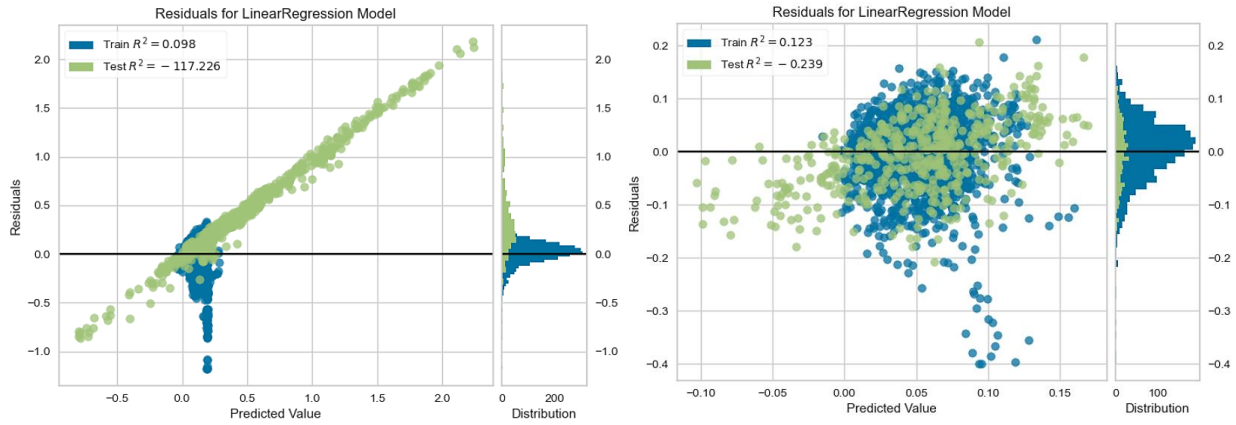


Figure 32. Residual plot of LR model – no sentiment analysis. On the left the TSLA residual plot, on the right the APPL residual plot.

It is clearly observable an increasing pattern in the residual error of TSLA, which is against one of LR assumption that the errors are independent and normally distributed. Therefore, the use of LR models might be not appropriate for the data associated with TSLA. On the other hand, the residuals randomly dispersed in the right plots of Figure 7 suggest that a LR model might be more appropriate for the APPL data.

The above findings are confirmed by the values of the performance metrics, which are reported in Table 7.

Performance metric	TSLA	APPL
Mean Absolute Error (MAE)	0.491	0.053
Root Mean Squared Error (RMSE)	0.663	0.067
Mean Directional Accuracy (MDA)	0.531	0.532

Table 7. Performance metrics for the LR model – No sentiment analysis.

As expected, the TSLA case show very poor performances, which are outperformed by a large extent not only by APPL case, but also by the benchmark models. The performance of APPL case are in line with the ones recorded for the benchmark models.

When compared to the observed long-term percentage volume change, for TSLA the magnitude of the errors obviously suggests very poor prediction capacity, while APPL obtains better results, although its prediction errors are still high. With respect to the MDA metrics, slightly better values are obtained than the previous models already investigated.

In Figure 33 the histograms related to the feature importance are displayed.

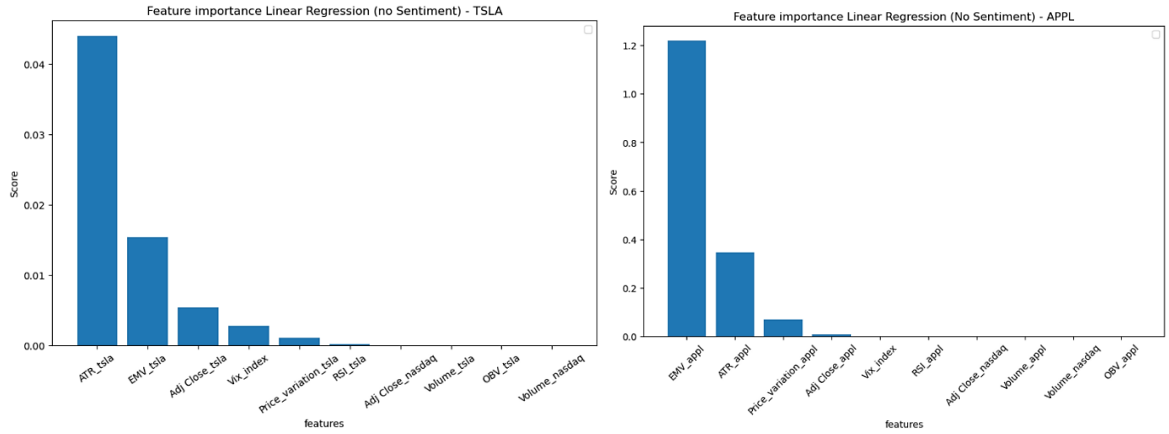


Figure 33. Feature importance histogram for LR model – no sentiment analysis. On the left and on the right, the TSLA and the APPL feature importance, respectively.

According to the figures, the ATR indicator, the EMV indicator, the close price of the stock and its variation price provide the main contribution for the target variable prediction, while the contribution of the remaining features is almost zero, with the exception of the Vix index that obtain a not negligible score in TSLA case.

4.2.2 Support Vector Regression model – No sentiment

In Figures 34 and 35 the plots of the predictions by using the SVR model are displayed for TSLA and APPL, respectively.

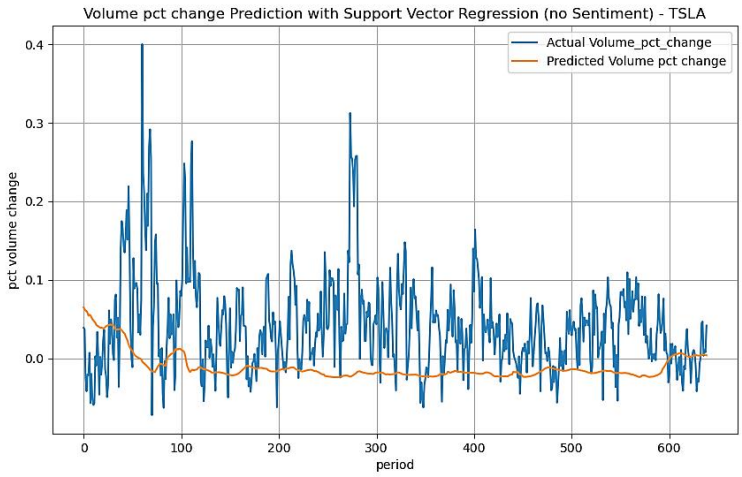


Figure 34. TSLA percentage volume change prediction through SVR model – no sentiment analysis.

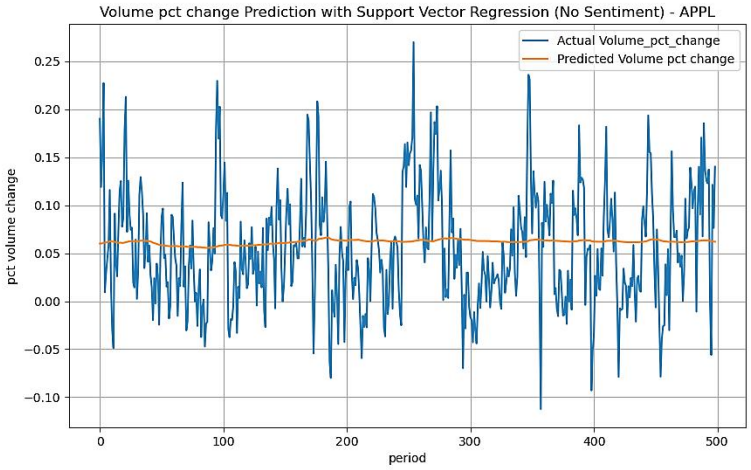


Figure 35. APPL percentage volume change prediction through SVR model – no sentiment analysis.

We observe that in both cases, TSLA and APPL, the curve prediction flatlines and is not able to reflects the upside and downside movements of the volume percentage change time serie. The main trends over the time are not detected either, showing similar results to the ones observed for the AR model in Section 3.1. The values of the prediction accuracy reported in Table 8 show that the SVR model performances are worse than other predictive models such as LR (with the exception of LR TSLA case) and SMA. Its performances are closely similar to the ones recorded for the AR model.

Performance metric	TSLA	APPL
Mean Absolute Error (MAE)	0.063	0.048
Root Mean Squared Error (RMSE)	0.083	0.060
Mean Directional Accuracy (MDA)	0.525	0.652

Table 8. Performance metrics for the SVR model – No sentiment analysis.

A potential explanation for the results obtained from the SVR model might be related to the relevant noise present. SVR model is sensitive to class overlapping, which might impact adversely on the prediction capacity of the SVR model. Class overlapping might occur when the features have similarities, although they belong to different classes and our dataset present features that are highly correlated as shown in Section 2.6.

With respect to the directional accuracy, values in line with the ones observed for other models investigated so far.

Looking at the feature analysis figure displayed in Figure 36, the OBV seems to have the most important contribution to the target variable prediction in both TSLA and APPL case. In the latter case, in addition to OBV indicator, the volume of the stock has equal importance to the prediction of the target variable.

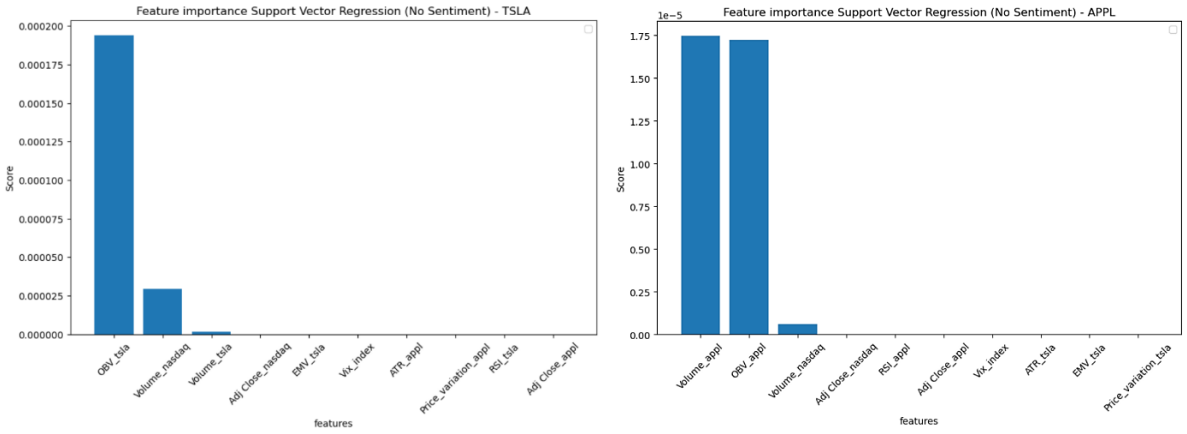


Figure 36. Feature importance histogram for SVR model – no sentiment analysis. On the left and on the right, the TSLA and the APPL feature importance plots, respectively.

4.2.3 Random Forest Regression model – No sentiment

The prediction curves against the actual records of volume percentage change for TSLA and APPL with no sentiment analysis are plotted in Figures 37 and 38, respectively.

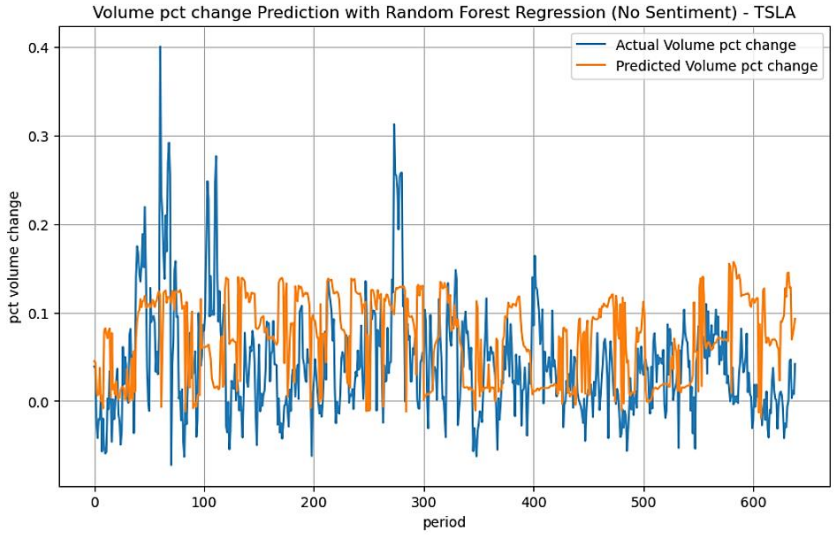


Figure 37. TSLA percentage volume change prediction through RFR model – no sentiment analysis.

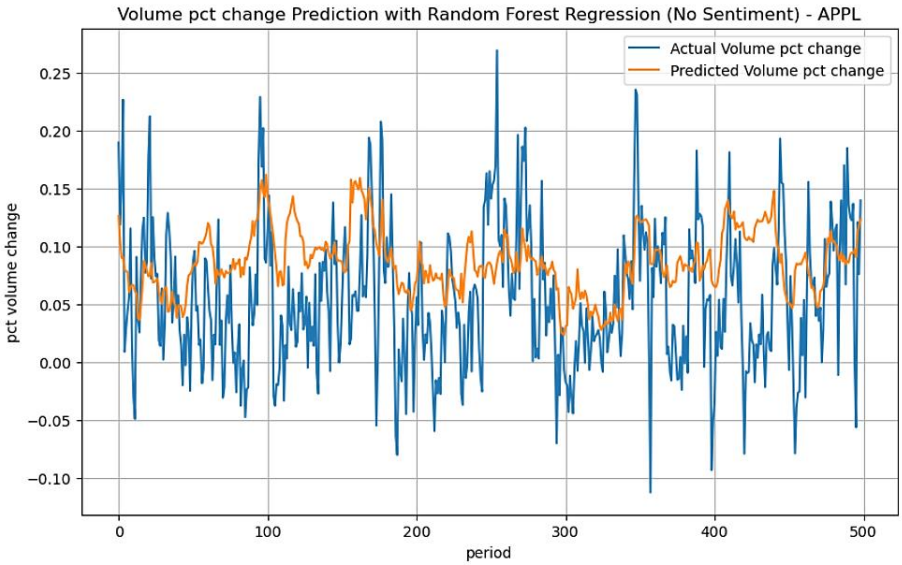


Figure 38. APPL percentage volume change prediction through RFR model – no sentiment analysis.

From the above figures, it is observable a good capacity of the model to reflect the main up/down trends, although, similarly to what already observed for other models investigated, the model is not able to predict all the abrupt movement of the volume percentage change.

RFR models can suffer from data noise as well, which might prevent the model from getting good prediction accuracy. In addition, in situations where the training and prediction inputs differ in their range and/or distributions (covariate shift), RFR model has difficult to handle this situation, especially because it can't extrapolate. The aforementioned situation is pretty marked for both TSLA and APPL as shown in Figures 39 and 40. We observe that for most of the independent features there are relevant differences between the train and test distributions.

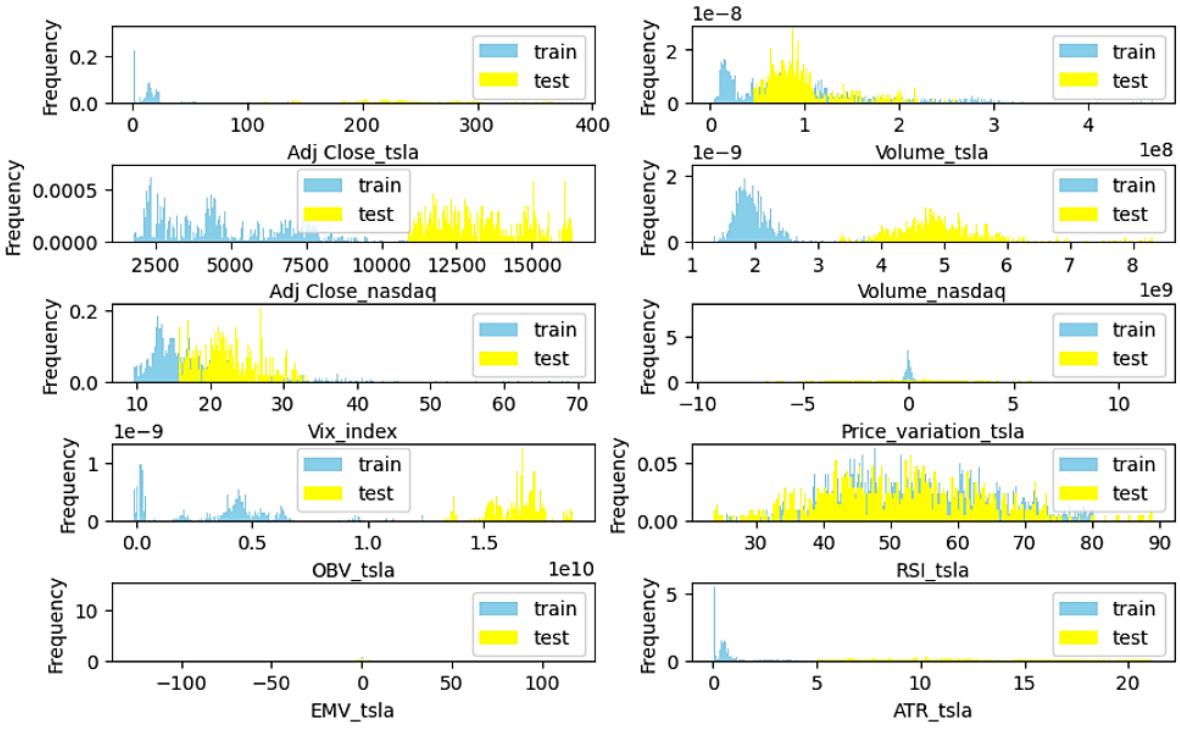


Figure 39. Distribution of independent features for train and test samples - TSLA

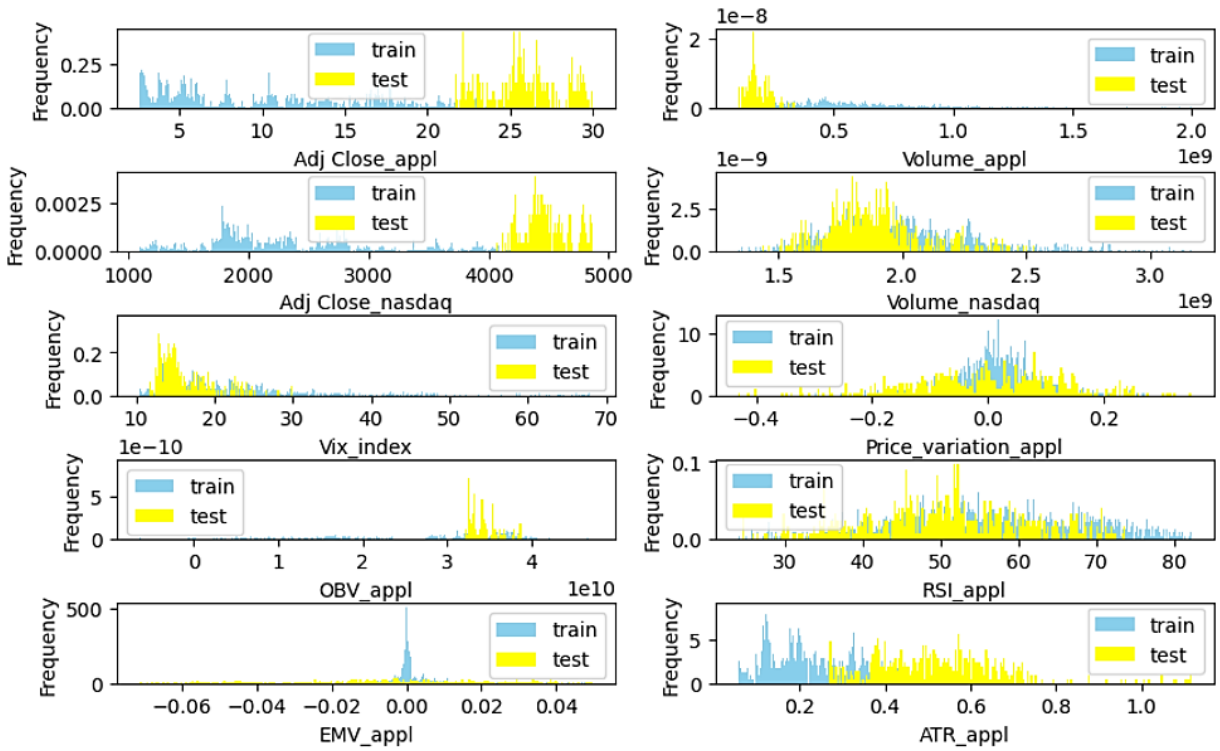


Figure 40. Distribution of independent features for train and test samples - APPL

In Table 9, the values of the performance metrics of RFR model are reported. No improvement is recorded and the results are closely in line with the results observed for most of the model already investigated.

Performance metric	TSLA	APPL
Mean Absolute Error (MAE)	0.064	0.053
Root Mean Squared Error (RMSE)	0.078	0.066
Mean Directional Accuracy (MDA)	0.493	0.536

Table 9. Performance metrics for the RFR model – No sentiment analysis.

In Figure 41, the feature importance histograms for RFR are displayed.

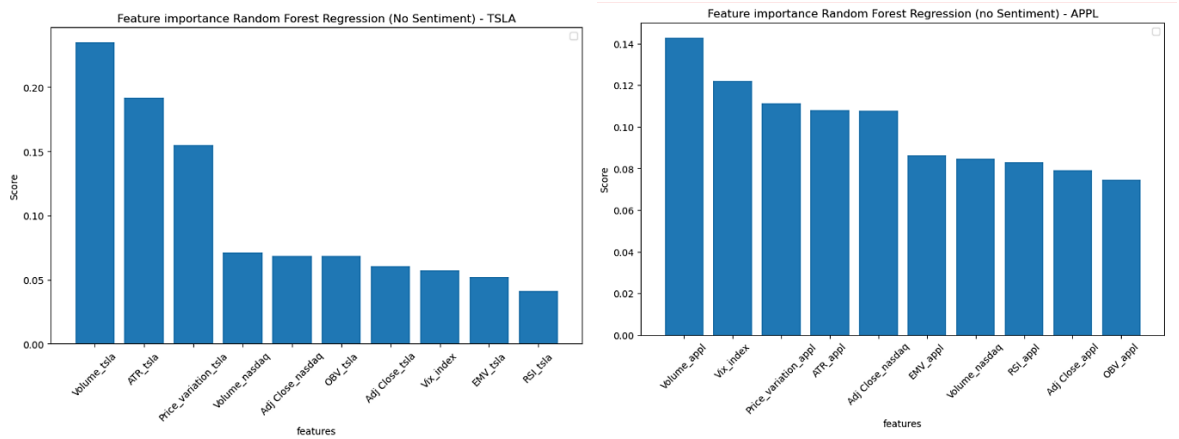


Figure 41. Feature importance histogram for RFR model – no sentiment analysis. On the left and on the right, the TSLA and the APPL feature importance plots, respectively.

Unlike what observed for previous predictive models, it is notable how all the features provide a good contribution to target variable prediction, with the volume stock recoding the highest score for both TSLA and APPL.

4.2.4 LSTM model – No sentiment

In the following we analyse the results related to the only DL model employed in this work under the "no sentiment" scenario. In Figures 42 and 43 the prediction curve obtained from LSTM model are plotted for TSLA and APPL, respectively.

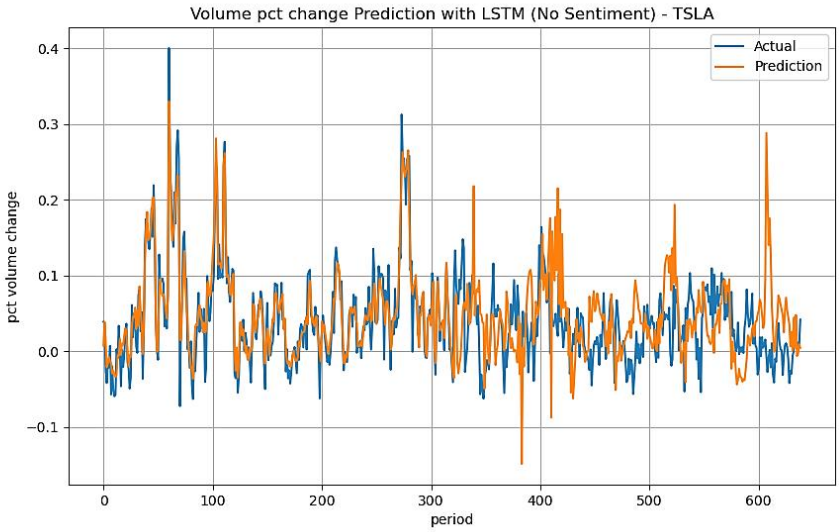


Figure 42. TSLA percentage volume change prediction through LSTM model – no sentiment analysis.

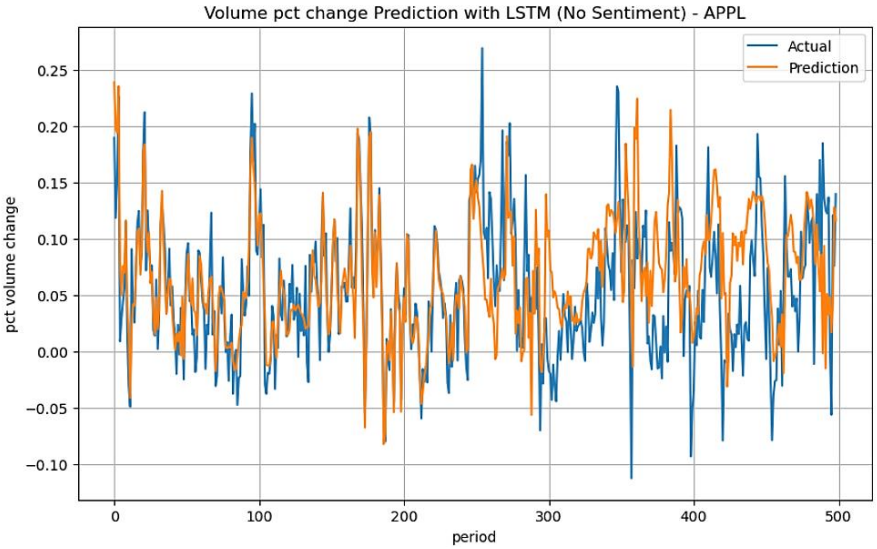


Figure 43. APPL percentage volume change prediction through LSTM model – no sentiment analysis.

The first relevant observation steaming from looking at the both figures is good ability of the LSTM model to predict not only the main up/down trend (as already observed in other previous model), but

also most of the abrupt movement changes. It is a relevant result when compared to the remaining models considered and suggests a good prediction ability of the LSTM model.

As per design, the LSTM has the advantage of being able to capture long-range dependencies and information from earlier steps and maintain important information while discarding the irrelevant ones. These particular features surely play an important role in time series and the good results are confirmed by looking at the Table 10.

Performance metric	TSLA	APPL
Mean Absolute Error (MAE)	0.034	0.036
Root Mean Squared Error (RMSE)	0.048	0.050
Mean Directional Accuracy (MDA)	0.559	0.558

Table 10. Performance metrics for the LSTM model – No sentiment analysis.

LSTM model obtains overall the best performances, with the value of MAE far lower than the values recorded for the other models. The good performances of the model are reflected by the comparison with the long-term average values of the percentage volume change. Both TSLA and APPL show an average error of prediction that is lower than the actual recorded average volume change that are 10.5% 5.2% for TSLA and APPL, respectively. With respect to the MDA, not relevant improvement is observed.

In Figure 44, the feature importance analyses performed by using SHAP method is show.

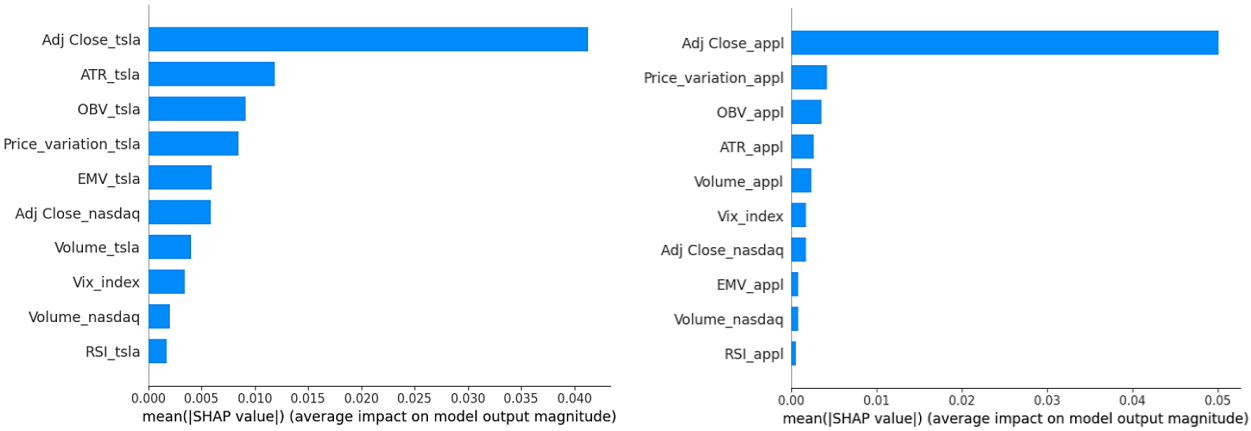


Figure 44. Feature importance (SHAP) histogram for LSTM model – no sentiment analysis. On the left and on the right, the TSLA and the APPL feature importance plots, respectively.

4.3 Results related to experiment 2 – Sentiment analysis included

In this section we report the results related to the second stage of the present research work.

In addition to the independent variable considered during the experiment 1, the sentiment analysis information is now fed into the predictive models.

Once again, the results and performances of the ML and DL under this new scenario including the sentiment analysis are reported in the following.

4.3.1 Linear Regression model – sentiment included

In Figures 45 and 46 the plots of the predictions by using the LR model and the sentiment information are displayed for TSLA and APPL, respectively.

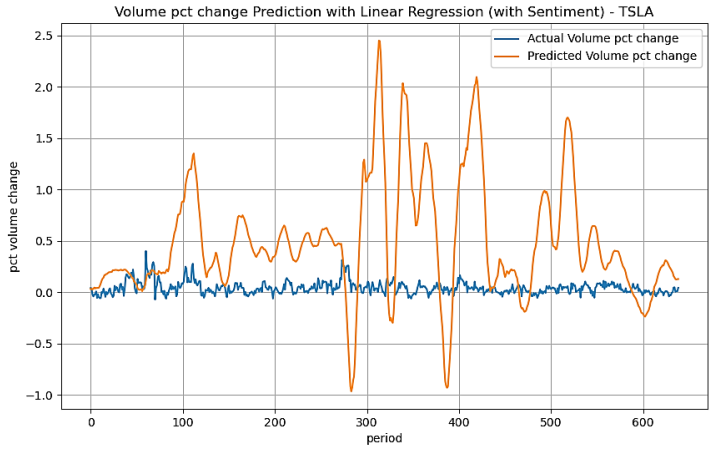


Figure 45. TSLA percentage volume change prediction through LR model – with sentiment analysis.

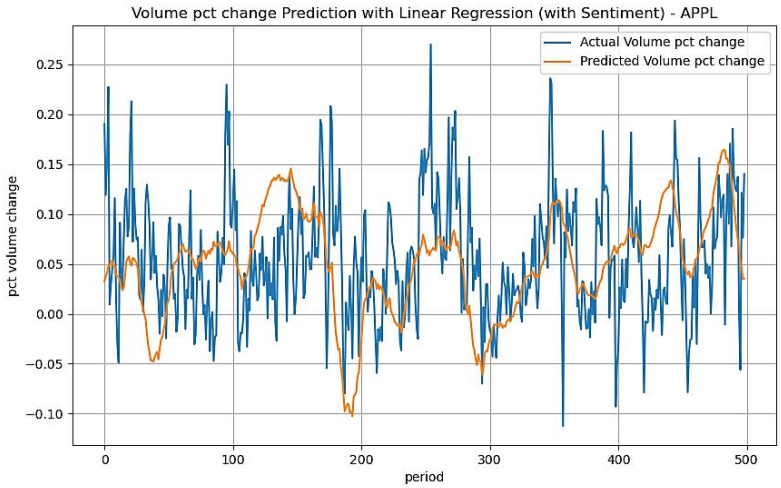


Figure 46. APPL percentage volume change prediction through LR model – with sentiment analysis.

Similar conclusions to the ones of the case with no sentiment analysis in Section 3.2.1 can be drawn. The TSLA predictions suffer from errors dependency, which impact adversely on the model's performance, unlike the APPL case, whose residual errors seem to be independent and normally distributed.

The values of the metric performances reported in Table 11 are in line with the values observed under the scenario with no sentiment analysis and no relevant improvement is observed by adding the sentiment information for our predictions.

Performance metric	TSLA	APPL
Mean Absolute Error (MAE)	0.532	0.053
Root Mean Squared Error (RMSE)	0.722	0.067
Mean Directional Accuracy (MDA)	0.517	0.550

Table 11. Performance metrics for the LR model – with sentiment analysis.

The feature importance analysis displayed in Figure 47 confirms the main contribution to prediction of the features ATR indicator, EMV indicator, variation price of the stock and the adjusted close price of the stock. It is worth to mention that the additional feature related to the sentiment analysis obtain relevant score in both cases, with the highest score recorded in feature importance for the TSLA case.

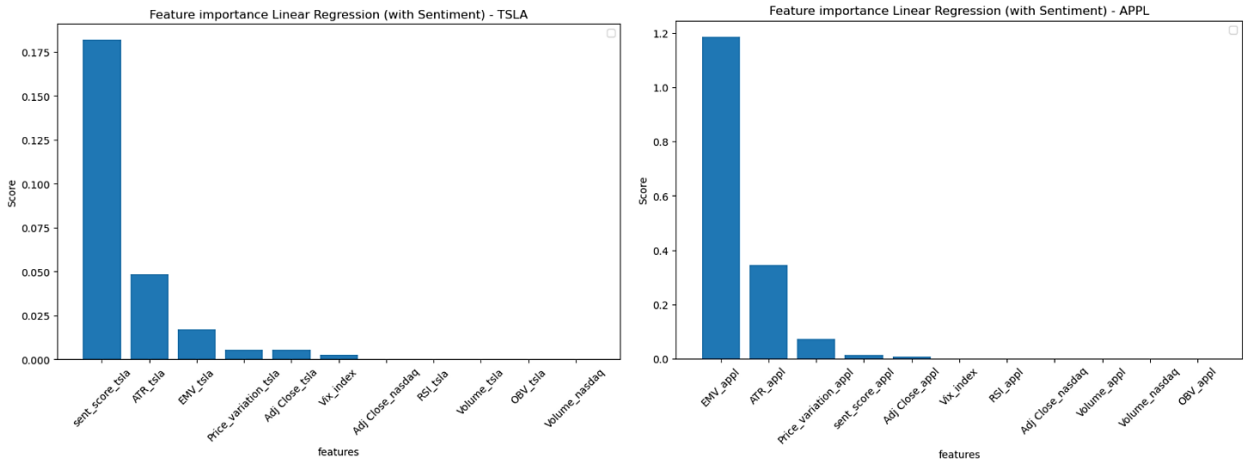


Figure 47. Feature importance histogram for LR model – with sentiment analysis. On the left and on the right, the TSLA and the APPL feature importance, respectively.

4.3.2 Support Vector Regression model – sentiment included

In Figures 48 and 49 the plots of the predictions by using the SVR model under the "sentiment" scenario are displayed for TSLA and APPL, respectively.

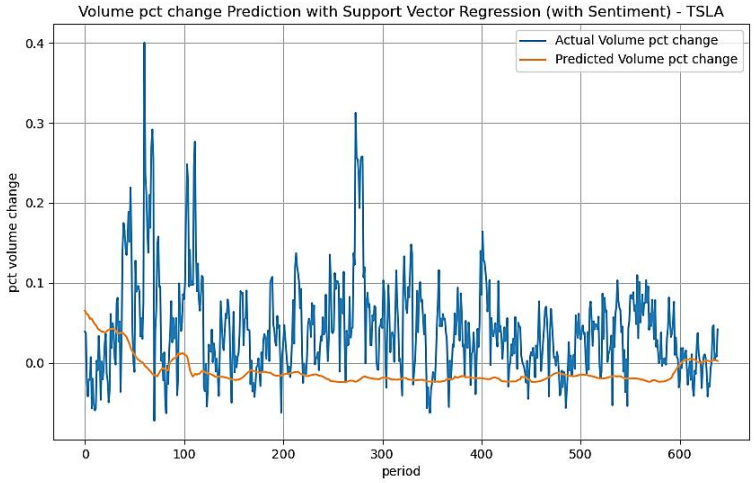


Figure 48. TSLA percentage volume change prediction through SVR model – with sentiment analysis.

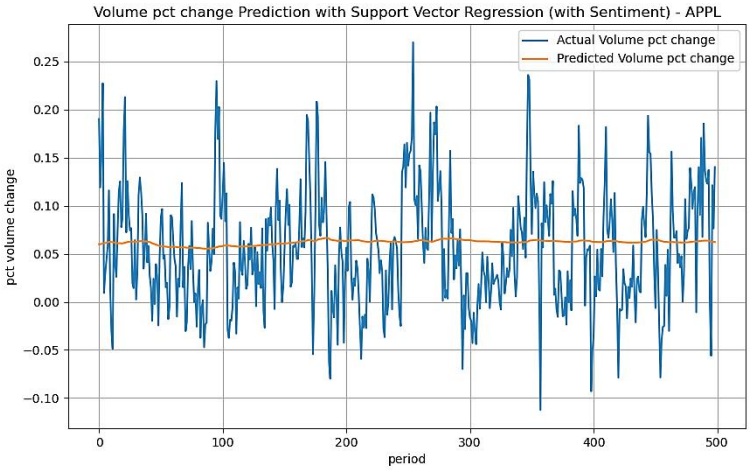


Figure 49. APPL percentage volume change prediction through SVR model – with sentiment analysis.

In Table 12 the performances of the SVR model implemented by adding the sentiment information are reported.

Performance metric	TSLA	APPL
Mean Absolute Error (MAE)	0.063	0.048
Root Mean Squared Error (RMSE)	0.083	0.060
Mean Directional Accuracy (MDA)	0.529	0.650

Table 12. Performance metrics for the SVR model – No sentiment analysis.

The same pattern of the results shown for the case with “no sentiment” analysis is observed. Because of the noise present in the data, the SVR model is not able to provide high level of the accuracy and its performance are similar to less sophisticated models already investigated in the previous sections.

The addition of the sentiment information seems to not provide any relevant benefit to the prediction accuracy.

In Figure 50 the feature importance scores for the SVR with sentiment analysis are displayed.

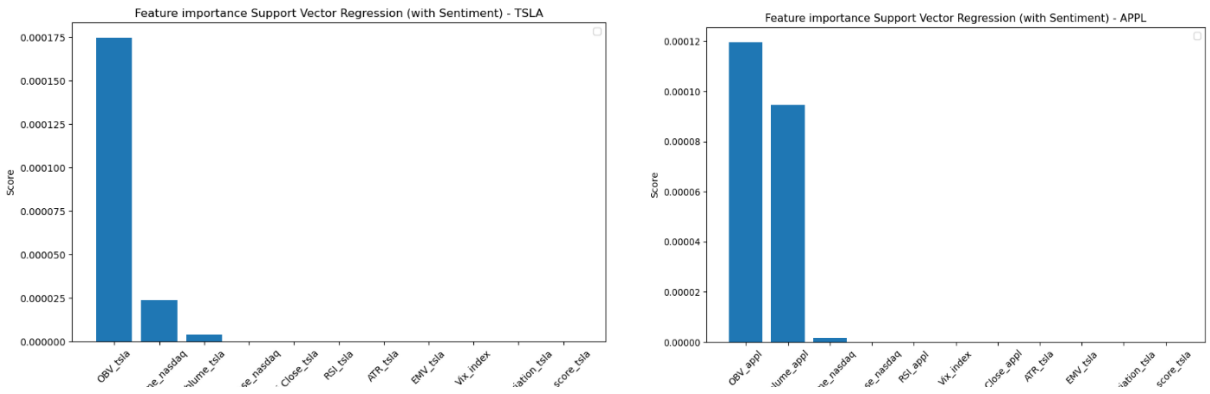


Figure 50. Feature importance histogram for SVR model – with sentiment analysis. On the left and on the right, the TSLA and the APPL feature importance plots, respectively.

Similar to the corresponding case with no sentiment analysis, for both of the stock, the OBV indicator obtain the highest importance score, far higher than the ones related to the other features. Only the volume apple feature obtains a compatible score to the OBV indicator. It is worth to mention that the contribution of the sentiment feature to target prediction in SVR model is practically nothing.

4.3.3 Random Forest Regression model – sentiment included

In line with what already observed for other model, the RFR model don't shown relevant improvement by adding the sentiment score information. In Figures 51 and 52 the prediction curves under "sentiment" scenario are shown, while in Table 13 the accuracy prediction metrics are reported.

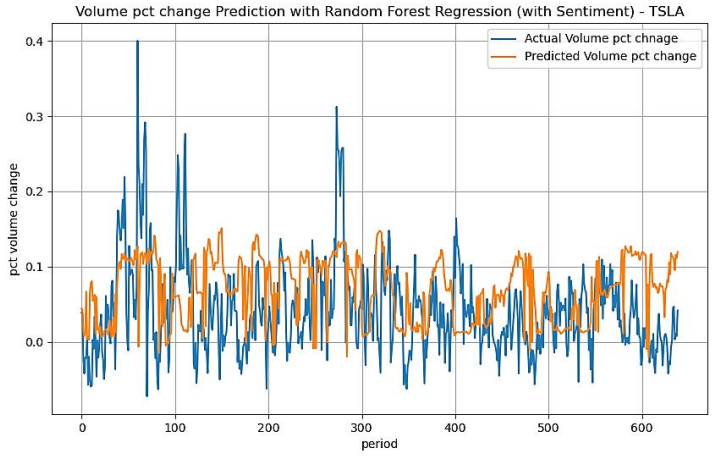


Figure 51. TSLA percentage volume change prediction through RFR model – with sentiment analysis.

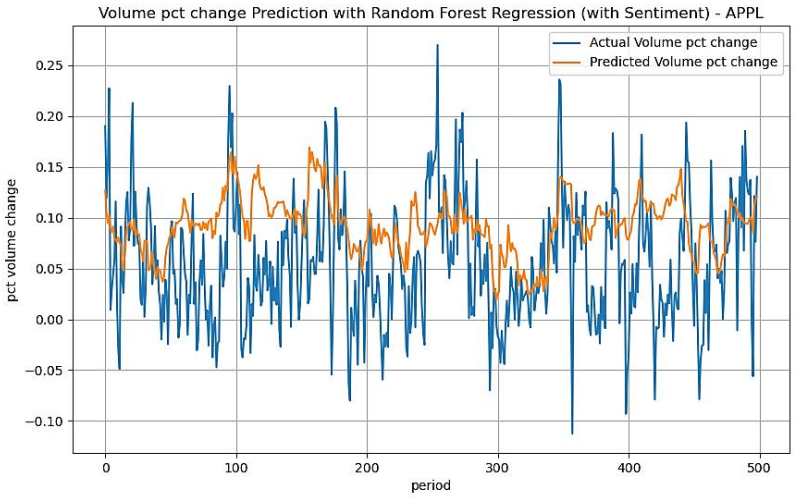


Figure 52. APPL percentage volume change prediction through RFR model – with sentiment analysis.

Performance metric	TSLA	APPL
Mean Absolute Error (MAE)	0.061	0.056
Root Mean Squared Error (RMSE)	0.075	0.069
Mean Directional Accuracy (MDA)	0.481	0.544

Table 13. Performance metrics for the RFR model – with sentiment analysis.

Analysing the metrics, only a small reduction of the error is observed when compared to the "no sentiment" scenario. However, as already pointed out in the "no sentiment" scenario, the independent features show significant differences between the distribution of the train and test samples, which can affect adversely the capacity of prediction of the RFR model.

The same patterns in the corresponding "no sentiment" case is observed for the feature importance analysis performed for the RFR, which is shown in Figure 53.

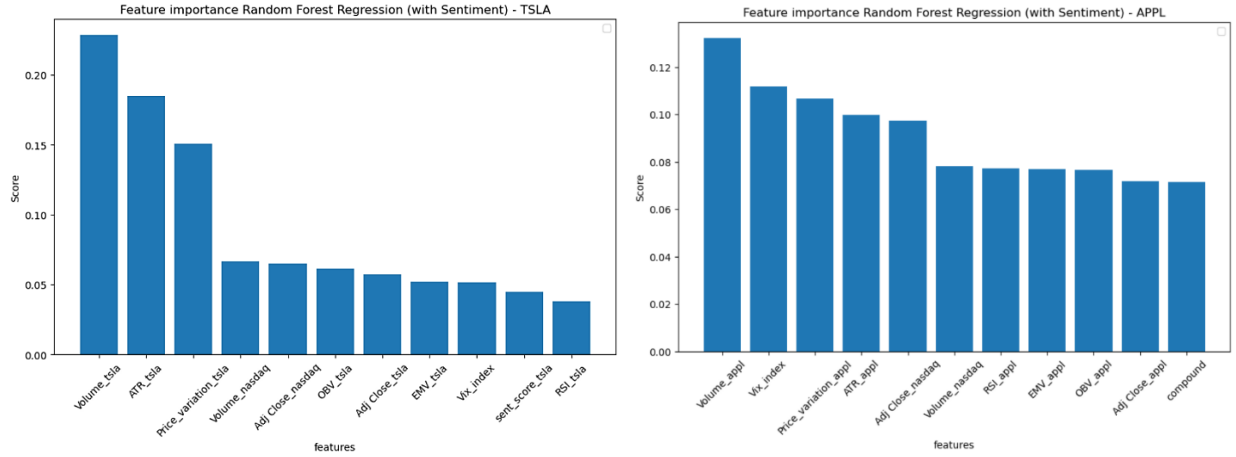


Figure 53. Feature importance histogram for RFR model – with sentiment analysis. On the left and on the right, the TSLA and the APPL feature importance plots, respectively.

All the features provide a discrete contribution for the target variable prediction, with the volume stock having the highest score for both TSLA and APPL.

4.3.4 LSTM – sentiment included

In the following, the results of the LSTM model applied under "sentiment" scenario is presented.

In Figures 54 and 55 the prediction curves of TSLA and APPL against the observed values are displayed, respectively.

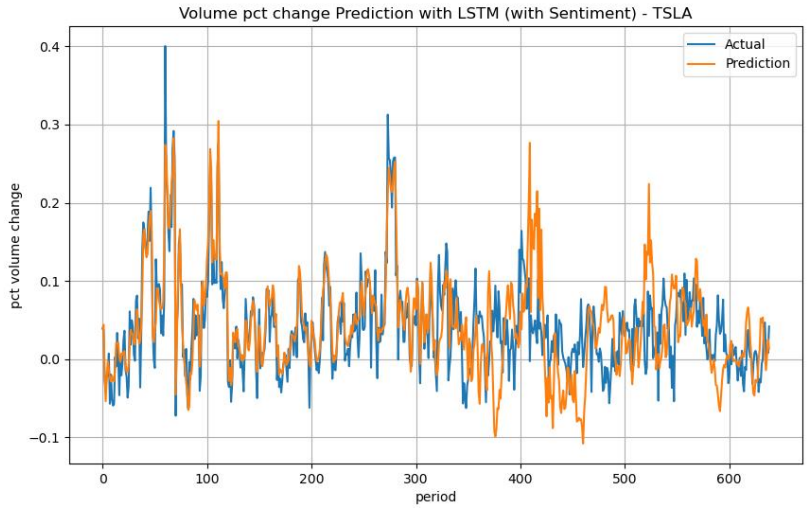


Figure 54. TSLA percentage volume change prediction through LMST model – with sentiment analysis.

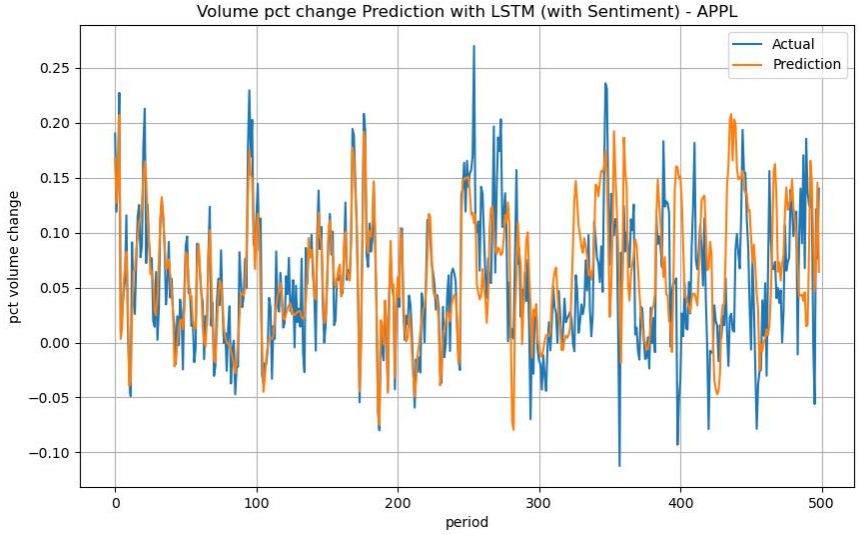


Figure 55. APPL percentage volume change prediction through LMST model – with sentiment analysis.

In Table 14 the accuracy prediction metrics of LSTM model that includes sentiment information are reported.

Performance metric	TSLA	APPL
Mean Absolute Error (MAE)	0.035	0.035
Root Mean Squared Error (RMSE)	0.050	0.051
Mean Directional Accuracy (MDA)	0.592	0.622

Table 14. Performance metrics for the LSTM model – with sentiment analysis.

The good results already observed in the corresponding scenario without sentiment information are confirmed. We observe how the curve prediction of LSTM model is able to reflect with good accuracy the pattern of the observed values.

The LSTM model still shows the best overall performance in this second stage of the experiment. However, it is worth to highlight that the performances are closely in line with what already observed in the corresponding “no sentiment” scenario, which suggests that no relevant improvement is obtained out of adding the sentiment information.

In Figure 56 the plots of feature importance analysis performed by using SHAP method are displayed.

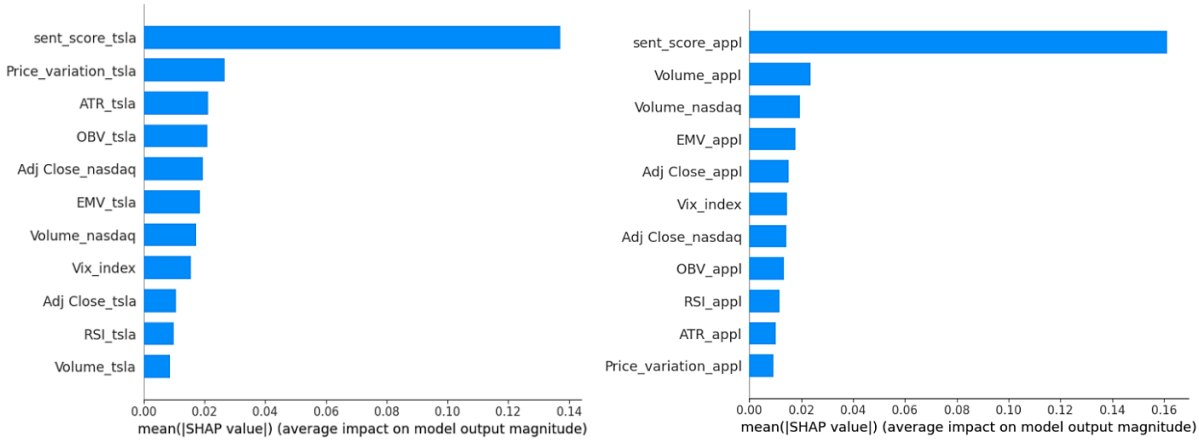


Figure 56. Feature importance histogram for LSTM model – with sentiment analysis. On the left and on the right, the TSLA and the APPL feature importance plots, respectively.

It is observable that most of the features seem to contribute almost equally to the model prediction, although the sentiment features stand out and obtain a far higher score than the remaining feature.

4.4 Conclusion

The main goal of the present research work was to compare the capacity of several predictive models to predict a medium-term average volume percentage change. The research involved predictive model pertaining to different level of complexity and class. From simple models such as SMA and AR, we moved to more advanced predictive models. We investigated some of the most widely used models in ML, such as LR, SVR and RFR. In addition, the promising LSTM from DL has been subjected to our analysis. That way we had a proper coverage of model investigated from complexity and classes perspective.

The experiment consisted of two main phases: the first was about evaluating the performances of the models without including any sentiment information, the second was about the behaviour of the same models by feeding into them data related to the sentiment analysis that had been worked out.

In Table 15 are summarized the results obtained from the analysis and for each model, the prediction accuracy performances are reported.

Model	Mean Absolute Error (MAE)	Root Mean Squared Error (RMSE)	Mean Directional Accuracy (MDA)
SMA - TSLA	0.044	0.059	0.403
AR - TSLA	0.090	0.099	0.399
SMA - APPL	0.047	0.060	0.410
AR - APPL	0.047	0.060	0.184
LR (no sentiment)-TSLA	0.491	0.663	0.531
LR (with sentiment)-TSLA	0.532	0.722	0.517
LR (no sentiment)-APPL	0.053	0.067	0.532
LR (with sentiment)-APPL	0.053	0.067	0.550
SVR (no sentiment)-TSLA	0.063	0.083	0.525
SVR (with sentiment)-TSLA	0.063	0.083	0.529
SVR (no sentiment)-APPL	0.048	0.060	0.652
SVR (with sentiment)-APPL	0.048	0.060	0.650
RFR (no sentiment)-TSLA	0.064	0.078	0.493
RFR (with sentiment)-TSLA	0.061	0.075	0.481
RFR (no sentiment)-APPL	0.053	0.066	0.536
RFR (with sentiment)-APPL	0.056	0.069	0.544
LSTM (no sentiment)-TSLA	0.034	0.048	0.559
LSTM (with sentiment)-TSLA	0.035	0.050	0.592
LSTM (no sentiment)-APPL	0.036	0.050	0.558
LSTM (with sentiment)-APPL	0.035	0.051	0.622

Table 15. Summary of the performances of the model investigated.

Looking at the performances of the models, we can draw the following main conclusions:

- Contrary to what might be expected, the addition of the sentiment information does not contribute to relevant improvement in the performances of any model for both of the stocks. Comparing the values of the experiment 1 (no sentiment) against the experiment (with sentiment), we observe closely similar values, which suggests that the additional feature does not provide further information that might help to better predict the target variables considered.

  Although the investor sentiment might wield influence within the stock market and on trading volume [78], this effect seems to not showing up in our research. One of the limits of the present work was related to the text data available for performing the sentiment analysis. We were forced to rely on data retrieved from open source and that were old and scarce in some cases like the TSLA case during the initial period considered. For future works, it might be worth to further investigate the quality of data are being used for sentiment analysis and try to retrieve data from specialized providers such as Reuters and Bloomberg.

- Overall, the best performance is recorded for the DL model, the LSTM model. It is in line with expectations since the LSTM model can be regarded as the most advanced model among the ones investigated in the present work. The design of the LSTM model ensures an efficient management of the information that is being processed. LSTM can capture long-term dependencies and select only the relevant information, while discarding the irrelevant ones [66]. The aforementioned feature of LSTM can surely play an important role in prediction

problems that deal with time series. The good results are confirmed by the accuracy metrics reported in table 11, which show a value around 0.035 of MAE for both of the stock. This value is even more valuable when compared against the actual long-term averages of the volume percentage change, which are 0.105 and 0.052 for TSLA and APPL, respectively.

- With respect to the predictive models other than LSTM, not relevant results are observed. The accuracy performances are closely similar to each other and the predictive models LR, SVR and RFR are not able to outperform the simpler benchmark models. We have already pointed out some of the potential reasons that might have impacted on the performances of the predictive model considered. For LR TSLA, which obtains the worst performances, we observed a high correlation between the residuals, which is against one of the assumptions of LR model. As regards SVR, potential factors that might have impacted adversely on their performances are noise of the data and overlapping class, while the relevant differences between the train and the test distribution of the independent features might have impacted adversely on the performances of the RFR.
  The results become less valuable when compared to the long-term averages. Although the error magnitude of TSLA is lower than is corresponding, overall, we deem the error values still high for both of the stock, which might imply no relevant benefit from the predictions.

- The direction accuracy, represented by the metric MDA, reflects the results that have been already observed. The best performance is recorded for the LSTM model alongside the SVR model, which are able to predict properly about 60% of the variable direction. The remaining models shown closely similar values.

It is worth to mention that the nature of the target variable to be predicted might have caused additional prediction difficulties to the most of models. Despite the smoothing process, the percentage volume change variable of both of the stocks resembles closely to white noise, with numerous abrupt up/down movements. The prediction of these random fluctuation might be challenging for most of the predictive models. In the present work, standard setting of the model has been used, therefore it might be worth to perform new analysis by changing some of the hyperparameters of the model in order to assess any improvement in the performances.
The only model to obtain promising results is the LSTM model, whose complexity and design enable to uncover relevant information in the time serie. Similar to other model, only a configuration of the LSTM model has been used in this work. Therefore, in order to assess the full potentiality of the LSTM, additional analysis with new settings of the model are recommended.

4.5 Future works

In this study, we covered many of the most commonly predictive models used in research and industry in order to evaluate their potential to deal with the prediction problem of the medium-term percentage change of the trading volume in the financial market. The most promising results are returned by the DL model, the LSTM. As already mentioned in the previous Section 4.4, only a configuration of the LSTM model has been employed in the present work.

Hence, a future study that will be focusing solely on the LSTM is recommended, which should include different LSTM architectures and different configurations of the model hyperparameters in order to try to improve further the performances of the model.

In addition, it might be worth to perform analysis by using different market variables as independent variables, which might provide more information for predicting the percentage volume change.

Considering the importance of the investor sentiment to the financial market dynamic, further investigations of its potential impact on the model prediction would be required. It would be important to perform sentiment analysis by relying on more robust and valid data, which might be obtained from specialized providers such Reuters and Bloomberg.

Finally, an alternative approach might be designed in order to mitigate the problem related to the random nature of target variable currently considered, the percentage volume change. The trading volume of the stock could be directly considered as the target variable to be predicted rather than its percentage change.

REFERENCES

[1] Johnson B. (2010). Algorithmic Tradind & DMA – An introduction to direct access trading strategies. *4Myeloma Press.*
[2] Beaver W. H. (1968). The Information Content of Annual Earnings Announcements. *Journal of Accounting Research*.
[3] Bamber L. S. et al. (2011). Trading Volume Around Earnings Announcements and Other Financial Reports: Theory, Research Design, Empirical Evidence, and Directions for Future Research. *Contemporary Accounting Research.*
[4] Bamber L. S. et al. (1999). Differential interpretations and trading volume. *Journal of Financial and Quantitative Analysis.*
[5] Cready W. M. and Hurtt D. N. (2002). Assessing investor response to information events using return and volume metrics. *Accounting Review.*
[6] Hong H. and Yu J. L. (2009). Gone fishin': Seasonality in trading activity and asset prices. *Journal of Financial Markets.*
[7] Lakonishok J. and Vermaelen T. (1986). Tax-Induced Trading around Ex-Dividend Days. *Journal of Financial Economics.*
[8] Caginalp G. and Desantis M. (2011). Stock price dynamics: Nonlinear trend, volume, volatility, resistance and money supply. *Quantitative Finance.*
[9] Karpoff J. M. (1987). The Relation between Price Changes and Trading Volume - a Survey. *Journal of Financial and Quantitative Analysis.*
[10] Westerfield R. (1977). The Distribution of Common Stock Price Changes: An Application of Transactions Time and Subordinated Stochastic Models. *The Journal of Financial and Quantitative Analysis.*
[11] Gervais S. et al. (2001). The high-volume return premium. *Journal of Finance.*
[12] Gunasekarage A. (2007). Causal and Dynamic Relationships among Stock Returns, Return Volatility and Trading Volume: Evidence from Emerging markets in South-East Asia. *Asia-Pacific Finan Markets.*
[13] Llorente G. et al. (2002). Dynamic volume-return relation of individual stocks. *Review of Financial Studies.*
[14] Lee C. M. C. and Swaminathan B. (2000). Price momentum and trading volume. *Journal of Finance*
[15] BIS Markets Committee (2020). Markets Committee FX execution algorithms and market Functioning.
[16] Deloitte (2019). Artificial intelligence: The next frontier for investment management firms.
[17] Bank of Italy (2019). Corporate default forecasting with machine learning.
[18] Hariom Tatsat et al. (2020). Machine Learning & Data Science Blueprints for Finance - From Building Trading Strategies to Robo-Advisors Using Python, *O'Reilly*.
[19] Barbaglia L. and Consoli S. (2021). Data Science Technologies in Economics and Finance. *In book: Data Science for Economics and Finance (pp.1-17).*
[20] Report from "The Organisation for Economic Co-operation and Development (OECD)" (2021). Artificial Intelligence, Machine Learning and Big Data in Finance - Opportunities, Challenges and Implications for Policy Makers.
[21] Mayur W. et al. (2022). A survey on sentiment analysis methods, applications, and challenges. *Artificial Intelligence Review*
[22] Kiritchenko S. et al. (2014). Sentiment Analysis of Short Informal Texts. *Journal of Artificial Intelligence Research*.
[23] Yan-Yan Z. et al. (2010). Integrating intra-and inter-document evidences for improving sentence sentiment classification. *Acta Automatica Sinica*
[24] Moreo A. et al. (2012). Lexicon-based comments-oriented news sentiment analyzer system. *Expert Systems with Applications.*
[25] Appiahene P. et al. (2022). Understanding the Uses, Approaches and Applications of Sentiment Analysis*. Research Square*
[26] Poornima A. and Priya K.S. (2020). A Comparative Sentiment Analysis of Sentence Embedding Using Machine Learning Techniques. *2020 6th International Conference on Advanced Computing & Communication Systems (ICACCS)*

[27] Bhatt N.T. and Swarndeep S.J. (2020). Sentiment Analysis using Machine Learning Technique: A Literature Survey. International *Research Journal of Engineering and Technology*
[28] Tan K.L. et al. (2023). A Survey of Sentiment Analysis: Approaches, Datasets, and Future Research. *Applied Sciences*
[29] Thomas L. and Taisei K. (2006). Forecasting Volatility and Volume in the Tokyo Stock Market: Long Memory, Fractality and Regime Switching.
[30] Iebeling K. and Milton B. (1995). Forecasting Futures Trading Volume Using Neural Networks,
[31] Alvim L. et al. (2010). Daily Volume Forecasting using High Frequency Predictors.
[32] Daniel L. et al. (2019). Volume prediction with Neural Networks.
[33] Xiaojie X. and Yun Z. (2023). A high-frequency trading volume prediction model using neural networks.
[34] Bin G. and Jun X. (2020). Forecasting Excess Returns and Abnormal Trading Volume using Investor Sentiment: Evidence from Chinese Stock Index Futures Market.
[35] Liang Z. et al. (2021). Long-term, Short-term and Sudden Event: Trading Volume Movement Prediction with Graph-based Multi-view Modelling.
[36] https://finance.yahoo.com/
[37] https://www.kaggle.com/
[38] Pansy N. and Rupali V. (2021). A review on sentiment analysis and emotion detection from text. *Social Network Analysis and Mining.*
[39] Hutto C.J. and Gilbert E. (2014). VADER: A Parsimonious Rule-based Model for Sentiment Analysis of Social Media Text. *Association for the Advancement of Artificial Intelligence.*
[40] Elbagir S. and Yang J. (2019). Twitter Sentiment Analysis Using Natural Language Toolkit and VADER Sentiment. *Proceedings of the International MultiConference of Engineers and Computer Scientists.*
[41] Srushti D. (2022). Study of Market Indicators used for Technical Analysis. *International Journal of Engineering and Management Research*
[42] Tsang W. W. H. and Chong T. T. L. (2009). Profitability of the On-Balance Volume Indicator. *Economics Bulletin*.
[43] Moroan A. (2011). The relative strength index revisited. *African journal of business management.*
[44] Yamanaka S. (2012). Average True Range. *Stocks & Commodities V. 20:3 (76-79).*
[45] Chavarnakul T. and Enke D. (2006). Stock Trading using Neural Networks and the Ease of Movement Technical Indicator. *IISE Annual Conference.*
[46] Zarattini C. and Aziz A. (2023). Volume Weighted Average Price (VWAP) The Holy Grail for Day Trading Systems. *Peak Capital Trading.*
[47] Marcoulides K. and Raykov T. (2019). Evaluation of Variance Inflation Factors in Regression Models Using Latent Variable Modelling Methods. *National Library of Medicine.*
[48] Roemer J. J. et al. (2021). Conducting correlation analysis: important limitations and pitfalls. *Clinical Kidney Journal.*
[49] Hochreiter S. and Schmidhuber J. (1997). Long short-term memory. *Neural Computation*
[50] Svetunkova I. and Petropoulos F. (2017). Old dog, new tricks: a modelling view of simple moving averages. *International Journal of Production Research*
[51] Ullrich T. (2021). On the Autoregressive Time Series Model Using Real and Complex Analysis. *Forecasting.*
[52] Stanton J.M. et al. (2001). A Brief History of Linear Regression for Statistics Instructors. *Journal of Statistics Education*.
[53] Montgomery D. et al. (2007). Introduction to Linear Regression Analysis. *Wiley-Interscience.*
[54] Kecheng Q. (2024). Research on Linear regression algorithm. *MATEC Web of Conferances.*
[55] Vapnik V.N. (1995). The Nature of Statistical Learning Theory. *New York, Springer-Verlag*.
[56] Smola A.J. and Scholkopf B. (1998). A tutorial on support vector regression. *NeuroCOLT Technical Report*.
[57] Muthukrishnan. R and Maryam J. S (2020). Predictive Modelling Using Support Vector Regression. *International journal of scientific & technology research.*
[58] Breiman L. (1984). Classification and Regression Tree. *CRC Press.*
[59] Breiman L. (1993). Classification and Regression Trees. *Wadsworth Statistics/Probability Series*.
[60] Breiman Leo. (2001) “Random Forests.” In: Machine Learning.

[61] Hochreiter and Schmidhuber (1997). Long short-term memory. *Neural Computation*
[66] Colah.github.io. 2020. Understanding LSTM Networks -- Colah's Blog. [online] Available at: https://colah.github.io/posts/2015-08-Understanding-LSTMs/
[67] Srivastava N. et al. (2014). Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *Journal of Machine Learning Research.*
[68] Agarap A. F. M. (2019). Deep Learning using Rectified Linear Units (ReLU). *arXiv:1803.08375.*
[69] Kingma D. P. and Ba J. L. (2017). Adam: a method for stochastic optimization. *arXiv: 1412.6980.*
[70] Abecasis S.M.et al. (1999). Performance metrics for financial time series forecasting. *Journal of computational intelligence in finance*
[71] Hodson T. O. (2022). Root-mean-square error (RMSE) or mean absolute error (MAE): when to use them or not. *European Geosciences Union.*
[72] Pesaran M. H. and Timmermann A. (2004). How costly is it to ignore breaks when forecasting the direction of a time series?. *International Journal of Forecasting*
[73] Grömping U. (2015). Variable importance in regression models. *WIREs Comput Stat.*
[74] Breiman L. (2001). “Random Forests.” *Machine Learning 45 (1). Springer.*
[75] https://scikit-learn.org/dev/modules/permutation_importance.html
[76] https://www.geeksforgeeks.org/feature-importance-with-random-forests/
[77] Lundberg S. M. and Lee S. (2017). A Unified Approach to Interpreting Model Predictions. *31st Conference on Neural Information Processing Systems*
[78] Wang Z. (2023). Investor Sentiment and The Stock Market. *Highlights in Business, Economics and Management.*