UNIVERSITY
OF TECHNOLOGY
SYDNEY

Multimodal Deep Learning for Stock Market Prediction

by Kamaladdin Fataliyev

Thesis submitted in fulfilment of the requirements for the degree of

Doctor of Philosophy

under the supervision of Associate Professor Wei Liu

University of Technology Sydney
Faculty of Engineering and Information Technology

June 2024

Certificate of Original Authorship

I, Kamaladdin Fataliyev, declare that this thesis is submitted in fulfilment of the requirements for the award of Doctor of Philosophy, in the Faculty of Engineering and Information Technology at the University of Technology Sydney.

This thesis is wholly my own work unless otherwise referenced or acknowledged. In addition, I certify that all information sources and literature used are indicated in the thesis.

This document has not been submitted for qualifications at any other academic institution.

This research is supported by the Australian Government Research Training Program.

Kamaladdin Fataliyev
27/09/2024

Abstract

Stock market movements are influenced by public and private information that includes historical market data, news articles, company reports, and social media discussions. Analyzing these vast sources of data can give market participants an edge to make profits. However, the majority of the studies in the literature are based on traditional approaches that come short in analyzing unstructured, vast multimodal data. Also, most existing works focus on singular data sources, which end up missing huge amounts of information from other data sources. Recently, while multimodal forecast models have been proposed, they mostly focus on either modality-specific or joint information in the modalities. Although there is a great opportunity for the use of multiple data sources for market analysis, challenges exist in efficiently modelling these modalities together. Another challenge lies in the multi-modal representation of these multiple modalities and capturing cross-modal information from the input data. To address the challenges of multi-modal learning in stock market prediction, we propose several innovative deep learning based learning models that can utilize both modality-specific and joint information across modalities. The main contributions of these models lie in their effective utilization of various data sources for price movement prediction, with advanced multi-modal representations and data feature fusion techniques. We propose novel models to predict both closing prices (regression models) and directional movements (classification models). We evaluate the performances using various metrics (accuracy, MCC for classification; MAE, MAPE, MDAPE, RMSE for regression) to evaluate the performances of our models and compare them against the state of the art analysis models. In the experiments, our proposed novel multimodal models outperform the baseline models and improve the prediction performances. Our experiments with real-world datasets show the effectiveness of the proposed multi-modal deep learning designs in stock market analysis and forecasting.

Acknowledgements

First of all, I would like to thank my supervisor, Associate Professor Wei Liu, who has provided enormous support during my time at UTS. Without his guidance and assistance, it would not have been possible to complete it, and I appreciate all his ideas and contributions. Wei has shown tremendous support and provided the push I needed not just to complete my degree, but also to tackle daily life obstacles. Even during the times when I lost my confidence in continuing and finishing the work, he provided much-needed support and guidance, and I am forever grateful for this.

I am also thankful for the support and feedback given by Mukesh Prasad and I appreciate all his contributions and assistance. I would also like to acknowledge the support of UTS FEIT and GRS staff and thank them for their assistance during my years at UTS.

Finally, I would like to thank my parents and sisters for their love and support during these years. I am ever thankful and grateful to my parents, who raised me with the love and passion of learning and supported me in all my pursuits.

Contents

List of Figures

List of Tables

List of Publications

Published Papers

1. K. Fataliyev and W. Liu: MMDL: A Novel Multi-modal Deep Learning Model for Stock Market Prediction. In Proceedings of the 2022 IEEE International Conference on Data Science and Advanced Analytics (DSAA), 2022.

2. K.Fataliyev and W. Liu: MStoCast: Multimodal Deep Network for Stock Market Forecast. In Proceedings of the 2023 Australian Data Mining Conference (AusDM), 2023.

3. K.Fataliyev and W. Liu: MCASP: Multi-Modal Cross Attention Network for Stock Market Prediction. In Proceedings of The 21st Annual Workshop of the Australasian Language Technology Association (ALTA), 2023.

Under Review Papers

1. MStoCast Journal Paper: Invited Submission to AusDM23 Special Issue in Data Science and Engineering (DSE), a Q1 Journal.

Chapter 1

Introduction

Stock markets are considered one of the pillars of the modern economic and financial system. Stock market is a financial market where the new issues of stocks, i.e., initial public offerings, are created and sold at the primary market, whereas the succeeding buying and selling are carried out at the secondary market [216]. Stock markets involve various market players where they invest to gain potential benefits. But these huge gains are also associated with high risks that can lead to losses. This makes stock market prediction an appealing area but also a challenging task as it is very difficult to predict stock markets with high accuracy due to high volatility, irregularity, and noise.

There are two main theories that assume the movements in the financial markets are random and thus unpredictable - Efficient Market Hypothesis (EMH) and Random Walk Theory(RWH). EMH [49] states that the price in the markets reflects the stock value accurately and responds only to new information, which consists of historical prices, public information and private information. Fama categorized markets as weak, semi-strong and strong based on the plausibility of prediction [49]:

- **Weak form:** states that stock market prediction based on historical prices won't lead to huge returns as the historical prices are already reflected in the current price. On the other hand, using private and public information can give excess returns.

- **Semi-strong form:** states that all of the past information - historical prices and public information - is already reflected in the current price and cannot be used for analysis. Private information can be used to predict markets.

- **Strong form:** states that all types of past information are reflected in the price; hence, using these for prediction won't yield good results.

RWH aligns with EMH and states that financial markets are stochastic, and prices follow random walk patterns, hence making it impossible to predict and outperform the market. In this way, it is consistent with EMH, especially with its semi-strong form.

In contrast to these theories, behavioural economists claim that investors can be emotional and thus, their behaviour can be explained using psychology-based theories. LeBaron [106] show that there is a delay between the time new information is being introduced and the market correcting itself by reflecting the new information, and Gidófalvi [63] report that this lag is approximately 20 minutes. These results support the idea that new information can be used in stock price prediction for a short duration.

Recently, a relatively new theory - the Adaptive Market Hypothesis - has been proposed to bridge EMH and behavioural finance - efficiency and inefficiency of the markets - in order to understand investor behaviour better. This theory assumes that markets can be predicted by analyzing investor behaviour. Behavioural finance mainly focuses on understanding the effect of investors' psychology on their trading strategy and on the market [268].

Various approaches have been developed to analyze and beat the market. Fundamental and technical analytical techniques and statistical and machine learning methods have been widely explored. The first two are mostly based on human knowledge and reasoning in areas such as locations of reversal patterns, market patterns, and trend forecasting [153]. Although these techniques take historical stock data into consideration, most of the existing research takes stock prices as stationary data. Moving averages, exponential smoothing linear regression, and other statistical methods have been widely implemented. However, they haven't been really effective as these models mostly assume linear relationships in the data.

The success of machine learning methods in other time series applications made them a good prospect for stock market analysis. Support vector machines and neural networks have been vastly applied to predict price movements in the financial markets. The performance of a prediction model mostly depends on the learned features. Early prediction models in the field focus on the features from historical stock data. To analyze the influence of public and private information, textual data from social media, news articles and official company announcements have been utilized. While the use of deep learning techniques in stock market prediction applications looks promising and brings a great opportunity, there are some existing limitations in incorporating multiple data sources to forecast the market movement.

1.1 Challenges in Modelling Stock Markets

Multimodal learning involves processing inputs from multiple data modalities (e.g. financial market data, social media and news data, corporate announcements). Although utilizing rich and multiple data sources can improve the performance of a learning model, it also brings more challenges, such as data alignment, representation and multimodal fusion. One of the main research areas in this domain includes designing an effective multimodal fusion technique, which is crucial in capturing the underlying information from the input modalities. Multimodal fusion is the process of gathering these multiple data modalities together for a learning task, and doing so effectively requires careful exploration of the data and learning algorithms. Two of the main fusion techniques are early and late fusion. In the early fusion, the input modalities are amalgamated as raw features at the first step, and then a representation and learning model is constructed to analyze these combined features. Although this approach can help to capture the inter-modal information, it mostly disregards the modality-specific features of each input data source.

On the other hand, in the late fusion approach, the input modalities are first processed separately, and the latent features are then brought together to create fusion features. This approach first focuses on the individual inputs and extracting modality-specific information, and then combining them to model their joint influence.

Another main challenge is the adoption of an effective methodology for multimodal representation. In the literature, this is usually done by combining multiple features into a compound vector, which is then further processed for the learning task. This process can help to optimize the learning process by forming a single feature vector. However, representing multi-characteristic features with a compound vector at an early stage may fail to encapsulate the deep hidden correlation of the various data modalities. Recently, tensors and encoders/decoders have also been utilized for representation, which overcomes some of the issues presented with the vector approach [120]. With the success of deep learning techniques, they are also being explored for multimodal representation.

With three or more modalities, bi-modal and cross-modal information among input pairs can also carry useful information and can improve the model's performance. On the other hand, trying to model every bit of information can over-complicate the learning process and may lead to over-fitting in the end. So, when designing multimodal learning architectures, it is important to find the right balance between keeping the model as simple as possible and not losing the important latent features in the input modalities.

It is also important to form correspondence among the input modalities, which involves the alignment of the data features at an early stage. This process often involves aligning the multiple data modalities based on an event or time (e.g. daily/weekly for stock markets), and it is usually done before multimodal fusion.

In this research we explore the existing works and propose multimodal deep learning models to address these issues for stock market prediction. The experiments with real-world datasets show the effectiveness of our proposed models in improving the efficiency of the market prediction models.

1.2 Aims and Objectives

Given the limitations above, this research focuses on multimodal deep learning and its implementation in utilizing various data sources for market prediction. We conduct research on multimodal representation and fusion techniques and experiment with various deep-learning algorithms for the effective utilization of various data sources. We explore the field and propose methodologies to answer the following research questions:

1. How to design a deep learning architecture to effectively implement multimodal fusion to utilize financial news and historical market data together for daily price prediction?

2. How to build a state of the art deep learning based multimodal representation model to effectively utilize multiple data sources together while not losing the modality-specific information from the input modalities?

3. How to analyze the cross-modal interactions of local representations in multiple modalities for stock market analysis while not disregarding the relationships among the input

data? How to utilize the dependencies between the data modalities and long sequences for an effective market prediction model?

In order to answer these questions, we focus on exploring some of the main challenges of multimodal learning in the stock market analysis domain. We experiment with multimodal fusion and propose deep learning based multimodal representation techniques. This research further utilizes deep learning algorithms to effectively build multimodal learning stock market prediction applications using various data sources.

We put three main points as the main objectives of this research in order to explore these questions and propose answers. These three objectives are:

1. To construct a deep learning architecture that effectively implements multimodal fusion to utilize financial news alongside historical market data to forecast the next day's closing price in stock markets.

2. To build deep learning (e.g. ResNet, Attention mechanism) based multimodal representation in order to learn joint and bi-modal relationships from multimodal web data in a classification task for stock market prediction.

3. To develop a multimodal learning application using Transformers and Attention mechanism in order to exploit the heterogeneity and interconnectedness of multiple data modalities for financial market analysis.

1.3 Research Contributions

This research develops several deep learning methods to answer the above questions and achieve the given objects. We present multimodal deep learning architectures to learn joint and unique information from multiple modalities for stock market analysis. We propose effective deep learning based architectures for multimodal fusion and representation. Our experiments show the effectiveness of the proposed techniques for both regression and classification tasks in stock market analysis.

The main contributions of this research are:

1. A novel multimodal deep learning model to capture the data-specific and joint influence

of financial news and stock market data for the prediction of the next day's closing price in stock markets.

2. Three novel multimodal learning models that utilize ResNets, Attention mechanism and Large Language Models for effective multimodal representations of multi-model web data for price movement forecast.

3. A novel Transformer-based multimodal cross-attention network to extract both modality-specific features and the cross-modal information from the data modalities in a unified framework for the prediction of the price movement direction.

1.4 Thesis Structure

We first review the existing works in the domain and provide a literature review in Chapter 2. Then we focus on adrresisng the gaps in the literature and providing answers for the given research questions. In chapter 3, we present our published work in multimodal deep learning, which is the first component of this research. The second component of our research is presented in Chapter 4 and then we present the third contribution in Chapter 5. We then conclude our research and provide our suggestions for future works in the area.

Chapter 2

Literature Review

In this chapter, we provide a review of the immense amount of existing literature on multi-modal stock market analysis models. We present input data types and cover the main textual data sources and variations, and provide discussions on feature representation techniques. Then, we cover both statistical analysis and machine learning based techniques and create a taxonomy of the major stock market forecast models. Importantly, we discuss representative work in each category of the taxonomy, analyzing their respective contributions. Finally, we show our findings on unaddressed open problems and give suggestions for future work. The aim of this study is to survey the main stock market analysis models, representation techniques for financial market prediction, shortcomings of existing techniques, and propose promising directions for future research.

2.1 Input Data

It's been shown that using multiple data sources together can improve the performance of a forecast model [5]. Li et al. [131] show that using both technical indicators and news data yields better results than only using either technical indicators or news sentiments, in both individual stock level and sector level. In this section, we present the existing works in stock market prediction domain based on the input data types, their sources and characteristics.

Quantitative stock data is a time series that can consist of daily closing price, opening price, price change, closing bid price, volume, and closing offer price. The existing research mostly focuses on stock market indexes such as DIJA, NASDAQ Index, NYSE Index, S&P

500, Hong Kong Index and others. These data can be extracted from various providers including Retuers, Bloomberg, Yahoo Finance, Google Finance and others in various frequencies such as daily, hourly, minute data etc..

Open, close, high and low prices are the opening, closing, highest and lowest prices in a given frequency, respectively, and volume shows the total number of traded shares in that duration. Another price component that is widely used is adjusted closing price which factors the stock's dividends, stock splits, and the new stock offerings. Thakkar and Chaudhari [218] take open, high, low, close and trading volume for S&P 500 and DIJA indices. Wang and Gao [231] focus on high and low prices and propose an LSTM model to predict high and low prices of soybean xfutures.

Historical data have been used in variuos frequencies in the literature. Zhong and Enke [267], Oztekin et al. [169] use daily stock data, and Ratto et al. [185] extract historical prices from Google Finance API with a frequency of a quarter of hour. Matsubara et al. [147] use daily data for Nikkei 225 and S&P 500 indices as input for a deep learning based system. Li et al. [129] use financial news articles to run experiments on the tick-by-tick price data in Hong Kong Stock Exchange. Xu and Cohen [249] categorize the stocks under nine industries: Basic Materials, Consumer Goods, Healthcare, Services, Utilities, Conglomerates, Financial, Industrial Goods and Technology. They extract the historical data for the period of two years for 88 stocks: all the 8 stocks in Conglomerates and the top 10 stocks in capital size in each of the other 8 industries. This data is used alongside twitter data for market prediction. Xu et al. [246] also use the same dataset for their research. They propose a novel stock movement predictive network via incorporative attention mechanisms that uses tweets and historical price data for stock movement prediction.

Textual data can come from various sources - social media, financial and general news, corporate announcements, blogging and micro-blogging websites. Textual content from news [194], financial blogs [33], [166] and social media and discussion boards [16], [151], [184], [209] have been used for stock market prediction. Dong et al. [48] compare the effect of mass media and social media on the stock market in various frequencies. They show that using mass media data can help with one-day horizon forecasting and social media can lead to better performance in two-to-five day horizon.

One of the main sources of the textual data is news data from websites, journals and newspapers. Shi et al. [204] and Peng and Jiang [175] use financial news data from Reuters and Bloomberg to predict stock price movements. Schumaker et al. [197] develop a sentiment analysis tool and use it with a financial news article prediction system called Arizona Financial Text (AZFinText). Wu et al. [240] use both news data and technical features for a regression model. Nam and Seong [155] propose a novel model for price movement prediction based on the financial news considering casual relationships between firms. They implement context-aware text mining based on the company-specific financial news rather than general news.

Although financial news is widely used as it is assumed to be less noisy, researchers have used filters in their data extraction such as all financial news, just company or product related news, stock related news and others. Tetlock et al. [215] and Tetlock [214] claim that general financial news has limited and short-lived predictive power on future stock prices. Li et al. [124] find that fundamental information of firm-specific news articles can enrich the knowledge of investors and affect their trading activities. Li et al. [128] work on both company-specific and market related news data and use the summary of the news content rather than the whole article in the analysis. Shynkevich et al. [207] use stock related news and allocate them to five categories for their sentiment analysis model. Vargas et al. [227] extract titles from financial news for their S&P 500 prediction model. Ding et al. [47] employed features from news headlines for market volatility prediction. Li et al. [124] show that stock prices are sensitive to restructuring and earning issues news.

The existing literature also varies based on the source of the news data. Most of the works take the financial news from Reuters and Bloomberg [227; 147; 47] and Yahoo! Finance [194; 197]. de Fortuny et al. [37] use data from the online versions of all major Flemish newspapers to predict commodity stock prices. Hong Kong Stock Exchange focused works mostly extract financial news from a financial information platform called FINET (www.finet.hk). Tokyo Stock Exchange related studies use Nikkei as their news source [5; 147], while some Chinese works extract data from a platform called Wind, which is a widely used financial information service provider [266].

The rise of the social media and micro-blogging websites created a huge source of data. It is not uncommon for influential people to share their opinions about political or financial situations on Twitter or other platforms and influence other market players' decisions. Some of the sources include Facebook, Twitter, Yahoo! Finance, Sina Weibo, Google Blog, Guba and Xueqiu.

Since social media data mostly reflect personal opinions, they have widely been used for sentiment analysis applications. The research by Bollen et al. [16] is one of the pioneers that uses Twitter data to evaluate the effect of public mood on stock markets. Smailovic et al. [211] and Li et al. [111] also import tweets to analyze the relationship between textual data and stock price movements. Wang et al. [237] use data from stock review blogs and employ SVM to classify the emotions using bootstrapping classifier.

Like news, the nature of social media data in the literature varies as well. Although most of the works extract Twitter data, some of the research focus on financial social media platforms like StockTwits or discussion boards like Yahoo! Finance boards. Mahmoudi et al. [145] use StockTwits data in their domain-specific sentiment analysis model with deep learning architecture. Nguyen and Shirai [159] extract stock related data from Yahoo! Finance message board and show that using sentiment information from social media can help to improve the stock prediction. Zhang et al. [266] use a discussion board named Guba (http://www.guba.com.cn) and Xueqiu, a Twitter-like investor social network in China to extract user sentiments.

Corporate disclosures are considered more trustworthy and less noisy as they usually include the latest official firm related information. These include quarterly earnings, legal and management information, past and current performance and challenges of the business [114; 178]. These documents, if decoded correctly, can give a major insight into a company's status, which can help to understand the future trend of the stock. Their influence on the stock returns have been analyzed in the short and long-term [56; 99; 230; 210].

Feuerriegel and Gordon [56] analyze company ad hoc announcements to understand their effect on short-term and long-term stock index predictions. Pröllochs et al. [179] use ad hoc announcements from Thomson and Reuters to build a decision support model. They build a sentiment analysis model with focus on negation scope detection. Hagenau et al. [71] use disclosures to get more expressive features for stock market prediction. Groth and Muntermann [65] also use ad hoc announcements to build and compare multiple classifiers for stock price forecasting.

Balakrishnan et al. [9] build a text classification model using data for narrative disclosures. Document-level information is processed for value-relevant information with features for risk sentiment and price momentum. Correlations are found between financial disclosure in text and financial characteristics and market performance of a firm. In terms of economic forecasts, the quality of disclosure documents and features is associated with future returns and confidence estimates of a firm. Thus, predictive analytics on financial disclosure can be used to condition the modelling parameters and product life cycles of a firm to particular industry segments and entrepreneurship ecosystems.

There are also studies that analyze reports and documents by regulatory organisations. Basu et al. [12] predict future investments by deriving industry-specific factors from SEC reports about financial performance of firms. They use Lasso and factor analysis as part of a supervised machine learning technique for driving corporate investments. Table 2.1 shows some of the research done in the textual analysis for finance area.

While working with social media data or official company disclosures, researchers usually take the whole content of the textual data. But there are various approaches to working with news articles such as taking just the news title, the whole article content, or the summary of the article. Shi et al. [204] compare using news title and content as input and conclude that their proposed deep learning model does not benefit from using additional content; so they focus on working with news titles. Li et al. [128] focus on the summaries of the articles and compare the techniques using news article summarization and full-length articles. They use both company-specific and market-related news articles and run experiments on five year data from Hong Kong Stock Exchange at at the individual stock, sector index, and market index levels. Their results show that using news article summarization for the prediction improves the performance and is better than using full-length articles.

In the literature, studies have extracted different kind of textual data for market analysis [128]: company related, both company and market related, product specific, sector related and data about related companies have been analyzed for prediction tasks. Li et al. [126] extract both company-specific and market related financial news written in English from a major financial news vendor called Finet. Seong and Nam [199] extract financial and economic news articles from Naver. Nuij et al. [165] use financial company-specific news articles alongside with technical indicators to build trading strategies. Their results indicate that combining news articles with technical indicators can lead to higher returns. Li et al. [124] use firm-specific financial news articles alongside with data from discussion boards to

Table 2.1: Stock market analysis using various types of textual content

Cat.	Ref	Type	Source	Market
News Content	[197]	News	Yahoo Finance	S&P 500
	[37]	News	Online Flemish newspapers	Brussels Stock Exchange
	[207]	News	LexisNexis database	S&P 500 Healthcare sector
	[87]	News	Reuters and Bloomberg	S&P 500
	[5]	News	Nikkei newspaper	Tokyo Stock Exchange
Social Media	[124]	Forum data (news and discussions)	Sina and EastMoney	CSI 100 Index
	[254]	Daily blog, forum, newspaper, social media data	Google blog, Google news, BoardReader, Twitter, newspapers and business magazines	824 public traded companies across 6 industries from COMPUSTAT and CRSP
	[160]	Message boards	Yahoo Finance	18 stocks from Yahoo Finance
	[266]	News and social media	Wind, Guba and Xueqiu	CSI 100 Index and Hong Kong Stock Exchange
	[30]	Social media based news	Sina Weibo	CSI 300 Index
	[16]	Social media	Twitter	DIJA
	[111]	Social media	Twitter	30 companies from NYSE and NASDAQ
	[145]	Social media	StockTwits	
Official docs	[113]	Corporate filings	SEC Edgar website	Corresponding stocks
	[56]	Ad hoc announcements	DGAP(Deutsche Gesellshaft fur Adhoc-Publizitat)	DAX index, CDAX index and STOXX 600 index
	[179]	Ad hoc announcements	Thomson Reuters	Corresponding stocks

analyze their effects on investors' trading activities. Li et al. [111] extract social media data from Twitter mentioning pre-selected 30 companies directly or indirectly (i.e. their products or services.). For example, for the company Apple, their lookup keywords include 'AAPL' (market code of the company), 'iPad', 'iPhone' (company products), etc.. Yu et al. [254] analyze data for a specific firm on a daily basis for a large range of firms rather than focusing on one company or sector. The research includes data from overall media instead of a specific business domain which can lead to high misclassification rate and spurious correlations.

Some applications in the existing literature analyze the data based on the categories within Global Industry Classification Standard (GICS). Nam and Seong [155] focus on asymmetric relationship of firms within GICS sector and analyze company-specific financial news considering casual relationships between firms (i.e target firm and casual firms). The proposed method is tested on Korean market dataset and outperforms traditional algorithms that assumes there is bidirectional influence between all firms. The research shows that, even if there is no firm related news, the model can use news related to the causal companies to predict the target firm's price directional movement. Shynkevich et al. [207] take news articles from LexisNexis database and allocate them to different categories according to their relevance to the target stock, its sub-industry, industry, group industry and sector (according to GICS as in Schumaker and Chen [193]). The results show that using multiple news categories with the Multiple Kernel Learning outperforms SVM and kNN models using single news category. Another finding is that using lower numbers of news categories reduces the model performance.

On the other hand, Seong and Nam [199] claim that GICS is limited in finding relevance regarding stock prediction and propose a model that incorporates heterogeneity and searches for homogeneous groups of companies which have high relevance. The research combines data from the target company and its homogeneous cluster (developed using k-means clustering) and the tests show that the proposed model outperforms the GICS system based and individual company level forecast systems.

The applications in the literature vary on the sources of the textual data as well. Shi et al. [204] use financial news data from Reuters and Bloomberg and stock-related social media data from Twitter for their deep learning based end-to-end model. Yu et al. [254] analyze the influence of social and conventional media on short-term stock market returns and risks. They show that when used separately data from social media gives better performance, but sentiment analysis based on both social and conventional data may increase the accuracy. Mahmoudi et al. [145] focus on social media data from finance-specific platform called StockTwits. The authors also include emojis in their data to explore their effect on investor sentiment analysis and show that it significantly improves sentiment classification in traditional algorithms. Chai et al. [24] build a multi-source heterogeneous data analysis

(MHDA) model by combining stock data, news event data and investor comments from financial discussion boards. The model is tested on the data of palm oil features and a future specific news events analysis module is developed to analyze palm oil related news articles.

2.2 Feature Representation

The representation of the given data - the features have a great effect on the performance of a prediction model. It is important to extract meaningful representations from the input data – let it be historical stock data or textual data. Hence, employing a good representation technique is a crucial task that affects the outcome of the whole model. Here, we cover the main representation techniques used in the literature for stock market analysis.

In the literature, fundamental and technical analysis are widely used to derive input features for an analysis model. The major difference between these two strategies is related to the nature of market features considered by each approach [153]. Fundamental analysis focuses on the fundamental values of a stock and analyzes the market based on these values, which covers financial statements, balance sheets, government policies, company market data, and political and geographical circumstances. This kind of systematic approach allows the investors to see the changes before they are reflected in the price and thus outperform the market. Researchers have used data from market trends, financial and political news, social media platforms to understand the fundamental value of a stock in order to improve prediction accuracy. On the other hand, technical analysts focus on historical stock data for their analysis as they believe that all available internal and external information is already reflected in the price, and any patterns in the market data would include the fundamental values as well. Technical analysis assumes that future price is tied to some patterns in the historical data and defines technical indicators to describe these behaviours in the data. It uses historical stock data such as stock prices, trading volume and breadth as their reference point to derive mathematical indicators [213]. Widely employed tools in the are include charting, relative strength index, moving average, exponential moving average, moving average convergence/divergence rules, relative-strength index, on balance volumes, momentum and rate of change, directional movement indicators [104; 222]. Guo et al. [69] and Gunasekaran and Ramaswami [67] use historical stock data to derive a set of technical indicators to get better results in the prediction process. Haq et al. [72] apply technical analysis for their deep learning based analysis model. They calculate forty-four technical indicators and use multiple feature selection approaches to find the best features for forecast. Shynkevich et al. [208] study the relationship between forecast horizon and the time frame used to calculate technical indicators. Using ten years daily price data for fifty stocks, they show that the best performance is achieved when the input window length is approximately equal to the forecast horizon. Patel et al. [172] test four prediction models for two different input approaches: using technical indicators as input and representing technical parameters as trend deterministic data. Using ten technical parameters from ten years of stock data, they show that their models perform better when they represent the technical indicators as trend deterministic data. Kara et al. [93] derive multiple technical indicators (simple moving average, weighted moving average, relative strength index and others) from ten years daily price data of ISE National 100 Index for machine learning based forecast models. Ratto et al. [185] employ technical indicators alongside dictionary based sentiment embeddings for market analysis.

The statistical approach is another widely used feature engineering technique for financial time series data. Statistical methods are more about dimensionality reduction and information compression. Principal component analysis, independent component analysis, singular value decomposition are among widely used techniques. In Guo et al. [69], independent component analysis is used for dimensionality reduction and canonical correlation analysis is used to extract feature set for the forecast of next day close price. Zhong and Enke [267] employ principal component analysis for ANN based market prediction model.

With the success of deep learning in various areas, researchers have also started exploring deep learning techniques for financial time series data. Studies either extract features beforehand and feed them into the network as input, or build an end-to-end model that can learn feature representations by itself. Long et al. [142] propose a novel end-to-end model named multi-filters neural network (MFNN) for feature extraction and price prediction on financial time series data. Using both convolutional and recurrent methods to capture different feature spaces, the model gives better performance than traditional machine learning techniques and single-structure deep learning techniques (CNN, LSTM, RNN). Gunduz et al. [68] build an end-to-end CNN-based model that combines feature selection and classification steps and it outperforms classifiers that rely on manually selected feature sets for market forecast.

In the literature, 2-g [71], noun phrases [193], sentiment words [124], topic modeling [160] and bag-of-words [207], [65] are widely used textual representation techniques. Bag-of-words approach is one of the most popular and basic approaches [156] that has been employed in analyzing financial texts [63], [105]. Unlike general tokenization, this method removes stop-words from the representations. Some implementations of BOW approach use stemming technique for better representation. But this approach sometimes lacks the ability to capture the semantics between words.

N-grams with various configurations have been studied for representation. Hagenau et al. [71] compare 3-gram and 2-gram for analysis of ad hoc announcements and find that 2-gram gives better results. The research also tests 2-word combinations with feedback-based feature selection method and finds that its accuracy is better than noun phrases and 2-gram. On the other hand, Pröllochs et al. [179] employ n-gram and test unigrams, bigrams, trigrams and 4-grams for negation scope detection in their reinforcement learning model. They show that reinforcement learning with 3-grams give better predictive performance.

Another technique is called noun phrases which extracts nouns and noun phrases from a given text [220]. An extension of this method that is called named entities that focuses on proper nouns from pre-defined categories. Semantic lexical hierarchy [198] and semantic tagging [148] are employed for categorization. Named entities allows for better generalization of previously unseen terms and does not possess the scalability problems associated with a semantics-only approach [197]. Proper nouns technique extracts specific nouns and named entities without a given category set. Schumaker and Chen [194] compare these techniques for news articles and show that using proper noun method yields better results.

In the last decade, deep learning methods have been successfully used for feature extraction from textual data to predict stock markets. Word embeddings is one of the techniques used to extract meaningful features from textual data. Researchers have used both domain-general and domain-specific embeddings to analyze the features. Without correct domain knowledge, it is a clear challenge to find effective matrices/measures to characterize the market movement, and such knowledge is often beyond the mind of the data miners [242]. Li and Shah [117] implement a model with CNN to train domain-specific word embeddings using StockTwits data. Shi et al. [204] employ a different technique that visually interprets text-based deep learning models. Peng and Jiang [175] use word embeddings to represent textual data. Lee and Soo [108] use Word2vec technique to get valuable features from text data. Mahmoudi et al. [145] employ deep learning with domain-specific word embeddings to explore investor sentiment classification in financial markets. They show that including emojis lead to a better sentiment classification. The proposed model was able to detect abstract-level feature types such as sarcasm and irony. In [147], a deep neural generative model (DGM) with news articles using paragraph vector algorithm is used for creation of the input vector to predict the stock prices.

Sentiment analysis has been a big part of textual data usage for financial market analysis. Researchers have developed various sentiment analysis methods using text-mining and natural language processing techniques to determine the sentiment of data with respect to a specific topic. Zhang et al. [262] employ Naive Bayes method to classify public sentiment as positive or negative. Li et al. [111] employ TF-IDF to build a sentiment word list and a novel technique named 'concept map' is built to capture the relationship between sentiment words from tweets about a company and its products. de Fortuny et al. [37] use TF-IDF to analyze news data for sentiment analysis and use sentiment polarity to assert the prediction of stock price movement. Li et al. [124] extract proper nouns to represent the event information in the news article and implement finance specific sentiment analysis (e.g. 'bearish' might mean something different in finance world). Kelly and Ahmad [94] show that a trading system can benefit from incorporating the sentiments from relevant news into next day trading decisions.

One of the techniques for sentiment extraction is lexicon-based unsupervised learning approach. It is based on word or phrase count in a given text. Deng et al. [40] use a sentiment lexicon - SentiWordNet 3.0 [6] - to analyze the sentiments of news, and predict stock prices by training a multiple kernel learning regression model with the fusion of sentiments and prices. Another widely used version is called SenticNet 5 [22].

Another technique is linguistics based which involves using manually marked dictionaries. Li et al. [126] employ the Harvard IV-4 Dictionary and the Loughran–McDonald Financial Dictionary in their news based sentiment analysis model. The research shows that sentiment analysis with the given dictionaries yield better results than bag-of-words model. Table 2.2 presents some applications from the literature with their representation techniques.

Market prediction applications have widely utilized bag-of-words, n-grams, named entities for textual representation. Shynkevich et al. [207] use simple bag-of-words, and Wang et al.

Table 2.2: Stock market analysis with textual representation techniques

Category	Ref	Content Type	Nature	Representation
Traditional techniques	[194]	News	Financial news	Bag of Words, Noun Phrases, and Named Entities
	[128]	News	Company-specific and market related news	Bag-of-words
	[197]	News	Financial news	Proper Nouns
	[155]	News	Company related news	Bag-of-words
	[179]	Ad-hoc announcements	Official filings	N-gram
	[124]	Forum	Firm-specific news	Proper Nouns
	[56]	Ad-hoc announcements	Official filings	Bag-of-words
	[71]	Ad-hoc announcements	Official filings	2-gram
Deep Learning	[47]	News	Financial news	Event embeddings
	[227]	News	Financial news	Word2Vec
	[5]	News	Financial news	Paragraph vector
	[266]	News and social media	Financial news	Domain-specific Word2Vec
	[30]	News	Financial news	Latent Dirichlet Allocation (LDA)
	[204]	News and social media	Financial news and stock related twits	Word embeddings
	[145]	Social media	Finance related tweets	Word embeddings
Sentiment Analysis	[160]	Message board	Stock related messages	Bag-of-words; Aspect-based sentiment
	[16]	Social media	General tweets	Sentiments using OpinionFinder and Google Profile of Mood States
	[126]	News	Company and market related news	Bag-of-words

[230] employ bag-of-words with TF-IDF and represent corporate disclosures as feature vectors. They show that utilizing textual data can improve prediction performance. In Li et al. [120], bag-of-words method is used to build a term vector from proper nouns and sentiment words. The authors propose a tensor-based stock information analyzer named TeSIA for market forecast.

Deep learning algorithms have also been explored for feature representation and engineering. Li et al. [123] build a tensor-based event-driven LSTM model for market movement forecasting. The authors use tensors to preserve the interconnections between different kind of features – technical indicators and media sentiment. The proposed model shows superiority when compared with state-of-the-art algorithms, including AZFinText, eMAQT, and TeSIA. Li et al. [129] propose a model using deep learning representation architecture for feature representations and extreme learning machine as the prediction model. The authors show that using deep learned feature representation together with extreme learning machines can improve the accuracy.

In Hogenboom et al. [80], events from news messages are weighted for their impact on the price movements. The model includes an extra step that allows for word sense disambiguation (WSD) to be incorporated in the event detection. The results show that including WSD reduces noise reduction that is introduced by high-impact ambiguous events.

Word2vec and Glove are two widely used word embedding algorithms [145] that have gained fame due to their success in textual representation. Vargas et al. [227] employ Word2Vec to represent financial news titles from Reuters. Zhang et al. [266] use Word2Vec to extract domain-specific word embeddings from Chinese financial news. The news corpus data is used to extract stock-related events. In Xu et al. [246], textual data are represented with word embeddings, which are fed into a bidirectional gated recurrent unit (Bi-GRU) network and obtain the tweet-level contextual embeddings. The authors propose a novel stock movement predictive network (SMPN) via incorporative attention mechanisms using Twitter data and historical stock data. The incorporative attention combines local and contextual attention mechanisms to clean the contextual embeddings by using local semantics which reduces the noise in the features and improves the model performance.

FastText and Bert are two other important embedding techniques. de Oliveira Carosia et al. [39] employ FastText to create word embeddings for Portuguese news data from Brazilian online news sources. In Cheng et al. [32], a pretrained BERT is employed for news embeddings which are later used for a novel multimodal prediction model. Colasanto et al. [34] employ BERT to determine the polarity score of the events that influence the market.

The efficiency of domain-specific and domain-general word embeddings have also been explored in the literature. Mahmoudi et al. [145] use GloVe and Word2Vec to extract domain-specific word similarities from StockTwits social media data. The research shows that domain-specific word embeddings capture the investor sentiment better than domain-general embeddings. Li and Shah [117] use CNN to build a large scale sentiment lexicon for stock market. The authors train domain-specific word embeddings using StockTwits data and show that domain-specific sentiment-oriented embeddings outperform domain-general word embeddings such as Word2Vec model. Kraus and Feuerriegel [99] employ representation learning approach (i. e. pre-training word embeddings) by using a different, but related, corpus with financial language and then transfer the resulting word embeddings to their dataset from German ad-hoc announcements in English from DGAP. The authors build a deep learning model with LSTM and compare it with traditional machine learning algorithms using bag-of-words approach. The results show that the LSTM model are superior, especially when the authors further pre-train word embeddings with transfer learning. Akita et al. [5] employ paragraph vectors to capture the distributed representations of news articles from the morning edition of the Nikkei newspaper. The authors show that using distributed representations outperforms bag-of-words based methods.

Some of the works focus on the event representation and employ various techniques to extract event information from textual data. Nuij et al. [165] extract events from Reuters news articles for the companies in the FTSE350 stock index. The research employs the ViewerPro tool for event extraction which relies on domain-specific knowledge. Ding et al. [47] develop a neural tensor network to learn event embeddings from financial news titles. The research represents events using dense vectors and analyze the combined influence of long-term events and short-term events on stock price movements.

Sentiment analysis has been widely studied for market analysis. Li et al. [126] incorporate Harvard IV-4 Dictionary and Loughran–McDonald Financial Dictionary sentiment dictionaries to create a sentiment space for their news-based model. They show that using sentiment dictionaries outperforms bag-of-words model at the individual stock, sector and index levels, but models using just the sentiment polarity don't perform well. Ratto et al. [185] use McDonald dictionary and AffectiveSpace 2 [21] to extract sentiment emebeddings from summaries of financial news articles for twenty most capitalized companies listed in the NASDAQ 100 index.

de Oliveira Carosia et al. [39] employ CNN for investor sentiment classification from Brazilian news. The proposed model outperforms MLP and LSTM based models models for sentiment analysis of Portuguese textual data. Jing et al. [91] also compare CNN against SVM, RNN, and LSTM and show that CNN outperforms others for investor sentiment classification.

Li et al. [130] propose a sentimental transfer learning approach that uses sentiments extracted from news-rich stocks (source stocks) to improve the prediction performance of the news-poor stocks (target stocks). The authors extract the financial news data from from Finet that includes both company-specific and market related news. The sentiment information is extracted using three dictionaries: Loughran-McDonald, Harvard IV-4, and SenticNet 3.0. The authors test the proposed technique on the data of Hong Kong Stock Exchange stocks from 2003 to 2008 and the results show that sentiment transfer learning can improve the prediction performance of the target stocks. Qian et al. [180] use Word2Vec to represent user comments and then employ CNN classifier to extract users' bullish-bearish tendencies. The experiments using data from Shanghai and Shenzhen 300 constituent stocks show that users' bearish tendencies are reflected in stronger market volatility and higher market returns, and the consistency of online users' tendencies has a positive impact on market volatility. Bollen et al. [16] employ two sentiment analysis tools to evaluate the daily Twitter feeds for sentiment extraction: OpinionFinder to get positive vs. negative sentiment values, and Google-Profile of Mood States (GPOMS) for more detailed analysis that classified the mood into 6 dimensions of sentiment (calm, alert, sure, viral, kind, and happy).

Studies also vary based on the level they focus for sentiment extraction. Yu et al. [254] extract sentence-level and document-level sentiment polarity from social media – blogs, forums and Twitter, and conventional media – newspapers, television and business magazines using Naive Bayes. Nguyen et al. [160] propose a new feature called 'topic sentiment' for market forecast. They extract topic sentiment which represents the sentiments of the specific topics of the company (product, service, dividend and so on), from Yahoo Finance Message Board using Latent Dirichlet Allocation (LDA).

2.3 Learning Models

In the literature, there are diverse sets of methodologies and techniques applied to stock market analysis. Statistical techniques, machine learning and deep learning algorithms have been implemented successfully. Figure 2.1 shows the categorization of the main techniques used in the area. Although deep learning is a subsection of machine learning, we cover it separately as these techniques have been growing in the recent years. Table 2.3 shows some of the studies using various statistical methods, machine learning and deep learning algorithms.

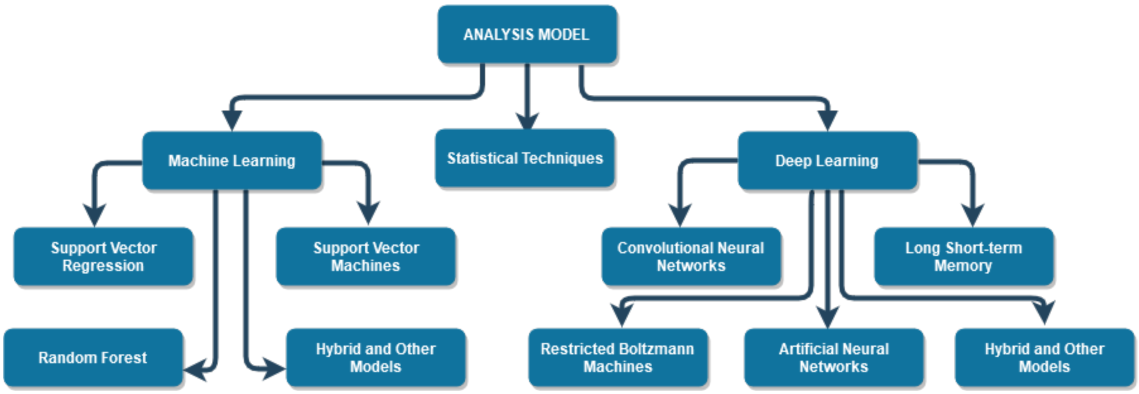

Figure 2.1: A taxonomy of stock market analysis models.

2.3.1 Statistical Techniques

Statistical methods including linear regression, auto-regressive moving average (ARMA), auto-regression integrated moving average (ARIMA), generalized autoregressive conditional heteroskedasticity (GARCH) and the smooth transition autoregressive (STAR) have been widely used in stock market forecasting. ARMA employs auto-regressive models to understand the momentum and mean aversion effects, while ARIMA was implemented to analyze and reduce the non-stationarity in the time series data to stationary series. GARCH [59] and STAR [190] have been employed as more advanced forms of statistical techniques. Franses and Ghijsels [59] use GARCH to focus on additive outliers (AO) – corrected returns. The research shows that the one-step ahead forecasts of volatility based on AO-corrected returns outperform the forecasts from GARCH and GARCH-t models for unadjusted returns. In Rounaghi and Zadeh [188], an ARMA based application is proposed for S&P 500 and London Stock Exchange stocks forecasting.

Hybrid models have also been proposed by implementing various techniques together. Pai and Lin [170] combine ARIMA with support vector machines for multiple stock forecasting and Zhang [261] use artificial neural networks with ARIMA for British pound/US dollar

Table 2.3: Stock market analysis using various analysis techniques

Category	Ref	Market	Input Type	Model
Statistical Techniques	[76]	S&P 500	Stock market data and technical indicators	GARCH
	[188]	S&P 500	Stock market data	ARMA
Machine Learning	[197]	S&P 500	News and stock market data	SVR
	[155]	Korean Stock Exchange	News and stock market data	Multiple kernel learning
	[65]	Corresponding stocks from announcements	Corporate filings and stock market data	Naive Bayes
	[185]	20 companies from NASDAQ 100	News, technical indicators and stock market data	RF, SVM, ANN
	[179]	Corresponding stocks from announcements	Ad hoc announcements and stock market data	Reinforcement learning
	[159]	Exxon Mobil, Dell, eBay, IBM, Coca-Cola Company stocks	Message board content and stock market data	SVM with a linear kernel
	[37]	Brussels Stock Exchange	Technical indicators, news and stock market data	SVM
Deep Learning	[30]	Shanghai-Shenzhen 300 Stock Index (HS300)	News and stock market data	RNN-Boost: RNN with GRU
	[99]	CDAX	Ad hoc announcements and stock market data	LSTM
	[47]	S&P 500	News, social media and stock market data	CNN
	[227]	S&P 500	News and stock market data	RNN + CNN
	[82]	Chinese Stock Market	News and stock market data	Hybrid Attention Network with self-paced learning mechanism
	[108]	Taiwan Stock Exchange	News, stock market data and technical indicators	CNN + LSTM
	[150]	Stock price of Tsugami Corporation	Event information, backlog and stock market data	LSTM-RNN

exchange rate prediction. Wang et al. [230] create a hybrid model by combining ARIMA with SVR. After getting feature vectors from textual data, ARIMA is used to analyze the linear part. Finally, an SVR model based on textual feature vector is developed for the nonlinear part. The authors compare the proposed model with a pure ARIMA model and a hybrid ARIMA and SVM model that uses stock data only. The results show that the proposed model outperforms the baseline techniques for the prediction of six pre-selected stocks.

Although there have been some success in applying statistical techniques, they can fall short in analyzing complex time series data. The main shortcoming of these methods is assuming linearity in the stock data and ignoring the stochasticity of stock markets.

2.3.2 Machine Learning

Financial markets have a non-stationary and non-linear nature and are considered dynamic, chaotic and noisy [1]. Machine learning algorithms are able to approximate non-linear functions and find underlying patterns which makes them useful in forecasting stock movements. On the other hand, these algorithms are prone to over-fitting. They often have difficulty finding the global optimum and can easily fall into local minima [102].

Various machine learning models have been tried for stock market analysis: support vector machines(SVM) [95], support vector regression (SVR) [69; 70], k-nearest neighbors (KNN) [264], decision tree classifiers [25], random forest (RF) [238], fuzzy system [201] and hybrid methods [232; 173]. Pröllochs et al. [179] employ reinforcement learning to build a tailored trading decision support model with a focus on negation scope detection. The results also show that negations affect investor's perception of the stock market and reinforcement learning can make the negation scope detection more accurate. Smailovic et al. [211] compare SVM with kNN and Naive Bayes based sentiment analysis models. They use company-specific social media data and develop a novel sentiment analysis model for stream-based active learning. They implement an active learning model using SVM classifier to analyze the relationship between company related twit data and their stock price. The results show that Naive Bayes under-performed compared to SVM and using kNN was not computationally efficient – the model was too slow.

Below, we cover some of these techniques used for financial market analysis. Although Deep Learning is a sub-field of Machine Learning, we put Deep Learning in Section 2.3.3 as an independent subsection due to its significant research advancements in the past decade.

Support Vector Machines

SVM is a supervised model that tries to minimize the upper threshold of the error of its classifications [86] by transforming the training samples into a greater space. The transformation is done with the help of kernel functions. It is a classification technique that maps training samples into points in a space and tries to create a gap between them, so new examples can be classified accordingly.

With ANN, SVMs are the most used machine learning techniques in stock market prediction [75]. Schumaker and Chen [194] employ an SVM-based model for market prediction using financial news and stock prices. Oztekin et al. [169] build three models – ANN, ANFIS and SVM - to predict daily stock price movements of Borsa Istanbul BIST 100 Index. In the experiments, SVM outperforms the other two models. Ni et al. [161] employ SVM with feature selection techniques for price index forecast. Li et al. [126] implement SVM based prediction model that incorporates financial news. Cao and Tay [23] experiment with adaptive parameters by incorporating the nonstationarity of financial time series into SVM. The results show that the SVM with adaptive parameters outperforms the standard SVM in financial forecasting. Yu et al. [253] propose an evolving least squares support vector machine (LSSVM) learning paradigm with a mixed kernel to predict the movement direction in financial markets. The model integrates a genetic algorithm (GA) for feature selection, and another GA for parameter optimization of the model.

Support Vector Regression

SVR operates similar to SVM, with the main difference being a regression technique rather than a classification one [224]. It attempts to transform training samples into multi-dimensional hyperplane in order to minimize the fitting error and maximize the goal function. Like SVM, SVR is also widely used for financial applications. Guo et al. [69] employ an SVR-based model for one-day-ahead prediction of closing prices.

An SVR model with Self-Organizing Feature Map (SOFM) is used by Huang and Tsai [85] for the analysis of Taiwan Stock Market Index (FITX). Patel et al. [173] propose to integrate SVR in the first stage and fused SVR, ANN, and random forest (RF) in the second stage that combined SVR-ANN, SVR-RF, and SVR-SVR models for prediction.

Random Forest

Random forest (RF) [18] is an ensemble learning algorithm that uses decision trees as the base learner. It integrates tree bagging [17] with random subspace technique. As the assumption of prior distribution is not necessary in RF, the technique has been well implemented in financial applications [100]. Patel et al. [172] build multiple prediction models using ANN, SVM, RF and NB. Experiments on the data of CNX Nifty and S&P Bombay Stock Exchange (BSE) Sensex price indices show that RF algorithm outperforms SVM in the market prediction.

Ballings et al. [10] compare three ensemble models – RF, AdaBoost, kernel factory – against single classifier models such as NN, SVM, kNN, and logistic regression. They test the models on the data from 5767 publicly listed European companies for one year ahead prediction. The results show that RF outperforms all the other algorithms. Weng et al. [238] employ RF regression for short-term price prediction using online data sources. Besides RF, they also test NN, SVR and boosted regression tree. Their results indicate the boosted regression tree (BRT) and the Random Forest Ensemble (RFR) as the best models for predicting the 1-day ahead stock price. Ji et al. [90] employ RF for forecasting of multiple stock indices using improved technical indicators based on wavelet denoising and a novel two-stage adaptive feature selection method.

2.3.3 Deep Learning

Deep learning is a particular kind of machine learning that achieves great power and flexibility by learning to represent the world as a nested hierarchy of concepts, with each concept defined in relation to simpler concepts, and more abstract representations computed in terms of less abstract ones [64]. The key advantage of deep learning is the capability of representation learning and the semantic composition empowered by both the vector representation and neural processing [115]. Deep nonlinear topology in neural networks can successfully model complex real-world data by extracting robust features that capture the relevant information [77] and achieve even better performance than traditional machine learning methods [15].

The rising success of these techniques in pattern recognition, image and speech processing and other fields, have made them a good choice for financial time series analysis. Artificial neural networks(ANN) [267], convolutional neural networks (CNN) [101], deep belief networks (DBN) [78], recurrent neural networks (RNN) [42], stacked autoencoders [14], a new model of RNN – Long short-term memory (LSTM) [79] have successfully been applied to different real-world applications. Deng et al. [42] use RNNs for feature learning and then apply reinforcement learning with deep representations to trading decision making. Xu et al. [248] propose a model with reinforcement learning based bidirectional gated recurrent unit that includes attention mechanism. They introduce two novel attention mechanisms based on global contextual and local semantic embeddings to deal with noise in textual data and learn news-level representation with more abundant semantics. Lee and Yoo [110] compare three RNN models for market analysis - SRNN, LSTM, GRU. The networks take open, adjusted close, high, low prices and volume as input and learn the features within the network. Akita et al. [5] build a news-based analysis model using LSTM to predict next day's prices and compare their model with SVM, MLP and RNN. In their trading simulations, LSTM outperforms the other models. Wang et al. [235] propose a novel model called Deep Random Subspace Ensembles (DRSE) that uses public sentiment and technical indicators for market forecasting. The proposed DRSE model combines deep learning algorithms and ensemble learning. They take Tsinghua Sentiment Dictionary 1.0 for sentiment analysis but extend it to enable more finance based word analysis. Shi et al. [204] build a deep learning based system called DeepClue for end-users - stock traders from public/private funds. The proposed deep regression model consists of four layers: a word representation layer, a bigram representation layer, a title representation layer, and a feed-forward regression layer. DeepClue bridges the analysis model and end-users by visually interpreting the main points of the prediction model. Matsubara et al. [147] develop a novel Deep Neural Generative model for the forecast of daily stock price movements using financial news. They employ paragraph vector technique to represent textual data, and the comparison against support vector machines and multi-layer perceptrons and show that the deep learning based model outperform the traditional techniques in the forecast of both mentioned markets. Li et al. [131] use LSTM network for analysis model and compare it with SVM and MKL based models. The authors use sentiment analysis to represent textual data to use for market prediction alongside with technical indicators from stock data. The proposed model outperforms the baseline models in terms of prediction accuracy when using both information sources. Cheng et al. [32] propose a novel multimodality graph neural network (MAGNN) to learn the lead-lag effects for financial time series forecasting, which preserves informative market information as inputs, including historical prices, raw news text and relations in knowledge graphs (KG). The model includes a two-phase attention mechanism (inner-modality and inter-modality attention) to infer the internal sequential patterns and inter-source lead-lag relations. Inner-modality attention mechanism is designed to automatically learn different contributions of graph-structured sources to the target node within each modality inputs. While inter-modality attention is proposed to learn weights among different modalities dynamically for a decent price movement prediction of target nodes, as different modality contributes differently in different time period. Yin et al. [251] propose a novel model called Graph Attention LSTM (GALSTM) to learn the correlations between stocks from Chinese A-share market and perform the prediction of these stocks' prices automatically.

Hybrid deep learning techniques have been proposed to improve the model performances. Park et al. [171] propose a prediction model called LSTM-Forest to overcome over-fitting by combining LSTM with RF. Vargas et al. [227] show that combining RNN and CNN (RCNN) is better than a CNN model and using textual data and technical indicators together has positive influence on the model performance. Another hybrid model called RNN-boost is used to predict the stock volatility [30]. LDA features and sentiment features are extracted from social media data to be used with technical indicators. The proposed model incorporates RNN and Adaboost and achieves an average accuracy of 66.54% and a best accuracy of 70.17%. The RNN model uses Gated Recurrent Units (GRU) to predict the stock price. Jing et al. [91] employ CNN for sentiment classification and then build an LSTM based model for one-day-ahead closing price prediction using technical indicators and investor sentiment as its input. They show that investor sentiment with technical indicators outperforms using technical indicators alone for price prediction. Ronaghi et al. [187] build a hybrid deep fusion framework for stock price prediction using a new dataset called COVID-19 related Price Movement prediction (COVID19 PRIMO). The model employs two parallel paths (hence hybrid), one based on CNN with Local/Global Attention modules, and one integrated CNN and Bi-directional LSTM path. Then, a multilayer fusion layer acting as a fusion centre is implemented to combine the localized features.

The main deep learning architectures used in stock market prediction literature are summarized below.

Artificial Neural Networks (ANNs)

ANNs try to imitate the learning process of humans and work on identifying patterns in the data. The basic unit is called a neuron which receives input and generates an output based on the given function. Interconnected units are attributed with weights and the model attempts to minimize the error by optimizing parameters.

The study by White [239] was the first to apply ANN models in the financial domain. Li and Ma [132] review the neural network applications in financial markets. Laboissiere et al. [103] implement a neural network model that takes stock prices and market index as input for Sao Paulo stock exchange (BOVESPA). An automatic trading system was proposed using ANNs by Vanstone and Finnie [223]. Gunasekaran and Ramaswami [67] use adaptive neuro-fuzzy inference systems (ANFIS) model for prediction of stock close prices. Ticknor [219] implement a Bayesian-regularized feed-forward ANN model for the one-day-ahead market trend forecasting. Zhong and Enke [267] use ANN with principal component analysis (PCA) for the prediction of S&P 500 ETF (SPY) returns.

Convolutional Neural Networks (CNNs)

CNN is a feed-forward neural network that contains more hidden layers than a standard neural network model. A typical CNN architecture includes convolutional layer, pooling layer and fully connected layers. The convolution, pooling and dropout operations makes it possible to have much deeper architectures without over-fitting the data. Compared to some of other deep learning models, CNN based models require more data as input.

After their success in areas such as image processing, Sezer and Ozbayoglu [200] build a 2-D deep CNN for stock market trend forecasting. They use 15 different technical indicators each with different parameter selections with stock price data, and transforms them into 2-D image. Then each indicator creates data for 15-day period and 15x15 sized 2-D images are created. Their algorithmic trading model is used for Dow 30 stocks and daily prices of nine Exchange-Traded Fund (ETFs). Gudelek et al. [66] also apply CNN to stock price movement forecast using ETFs. Wang [234] build a deep convolutional fuzzy systems (DCFS) and fast training algorithms for the DCFS for the forecast of Hang Seng Index (HSI).

Long Short-term Memory (LSTM) Models

RNNs can process raw text in sequential order to learn context specific features. But vanishing gradient problem and short context dependencies affect their real-world performance [13]. A new model of RNN – LSTM [79] has been proposed to overcome that issue. LSTM employs memory cells to process inputs with long dependencies [79]. It uses hierarchical structures and a large number of hidden layers for feature engineering.

LSTM has widely been used in financial and other time series models. Minami [150] implement an LSTM based model that takes event information, backlog and stock prices single stock price prediction. Liu et al. [137] combine CNN and LSTM for market prediction by using CNN for stock selection and LSTM for the prediction. Fischer and Krauß [57] show that LSTM outperform memory-free classification techniques such as random forest and logistic regression classifier for the prediction of a single stock market index. Akita et al. [5] represent textual data with paragraph vectors in their LSTM prediction model and test the model on the data from fifty companies listed in Tokyo Stock Exchange. Nelson et al. [157] also build and LSTM based model for IBovespa index prediction where they use technical indicators and stock data as input. They show that LSTM is able to learn even with input with very large dimension. Rezaei et al. [186] explore LSTM and CNNS with empirical mode decomposition (EMD) and complete ensemble empirical mode decomposition (CEEMD) algorithms for one-step ahead stock market prediction. They build three novel hybrid algorithms, i.e., CEEMD-CNN-LSTM and EMD-CNN-LSTM, to extract deep features and time sequences from stock price data. Liu et al. [138] use LSTM for market trend prediction using patent data and fundamental stock data where they transform the input data into wavelet transforms using discrete wavelet and feed into LSTM network.

Restricted Boltzmann Machines (RBM)

RBM is a different type of ANN model that can learn the probability distribution of the input set [181]. RBM consists of visible and hidden layers and the units in the layers are not connected. The learning process is performed multiple times on the network [181].

RBM has been successfully applied to financial market forecasting. Furao et al. [61] develop an improved deep belief network with continuous RBMs to exchange rate prediction. The model is tested with British pound/US dollar (GBP/USD), Indian rupee/US dollar (INR/USD), and Brazilian real/US dollar (BRL/USD) weekly exchange rates data sets. Their results show that the improved model works better than conventional neural networks such as feed forward nets.

Kuremoto et al. [102] propose a 3-layer deep belief networks for stock market prediction. Their DBN involves two restricted RBMs to capture the feature of input space of time series data. They employ particle swarm optimization algorithm to optimize the RBMs structure. Liang et al. [134] build an RBM based model for short-term stock market trend prediction.

2.4 Summary and Research Gaps

It has been shown that public mood and emotions have influence on trading strategies, which affects the prices [8]. Research in behavioral finance also shows that the stock market movements are affected by emotion [38] , financial news and social media content [197; 124]. Yu et al. [254] experiment with multiple textual data sources and show that the impact of different types of social media varies significantly.

In this section, we summarize the works from the previous sections and present the research gaps in the literature. We present some existing works in terms of the textual data they use, the representation types for the data and analysis models. We review some of the experiments and represent their results and comparison. Table 2.4 summarizes some of the representative works in the area.

With the success of machine learning and deep learning algorithms in other domains, researchers have also started to explore their applications for stock market analysis and forecasting. But most of the existing works in the area focus on single-mode applications with historical market data being the main source for a lot of applications, where deep learning techniques are applied to make predictions based on stock market data. Recently, with the increasing amount of web data, the incorporation of various data sources into forecast models has been explored [191; 28; 54]. It can be seen from Table 2.4 that various types of data modalities have been utilized for stock market analysis.

As mentioned in previous sections, modelling multiple modalities together is a challenging task, and in the literature, most of the works either focus on the input-specific features or joint features to make market predictions. However, it is important to capture both modality-specific and joint information. Also, multimodal representation and aligning data with varying characteristics for market analysis pose another challenge. Another focal point with time series analysis is modeling the dependency between long sequences and among the input modalities when using multiple data sources.

To address these challenges, we first propose a multimodal deep learning model where we effectively implement a multimodal fusion architecture to predict the next day's stock price. Then we build a novel multimodal representation network that utilizes ResNets, Attention networks and LLMs to effectively extract joint and modality-specific information from the input modalities. In our last contribution, a novel attention based design is proposed to model the interconnectedness of the web and market data for market analysis.

Table 2.4: Representative works in stock market analysis with text data

Cat.	Ref	Text type	Feature	Model	Notes
Data	[126]	News	Bag-of-Words	SVM	This research uses both company-specific and market-related news. It also builds a sentiment space using 2 well known dictionaries.
	[128]	News	Bag of Words	SVM	This research takes both company-specific and market related news. It extracts the summary of the text rather than using the whole article
	[207]	News	Bag-of-words	Multiple kernel learning	It shows that using multiple news categories helps with the forecast performance
Feature	[5]	News	Paragraph vector	LSTM	It shows that distributed representations of text are better than bag-of-words and numerical-data-only methods
	[124]	Forum	Proper nouns	SVR	This research uses domain specific sentiment analysis with firm-specific news data
	[197]	News	Proper nouns	SVR	It builds a sentiment analysis system and shows traders' contrarian character – buy after bad news and sell after good news.
	[179]	Official announcements	n-gram	Reinforcement Learning	This research shows that negations affect investor's perception of the stock market and reinforcement learning can make the negation scope detection more accurate.
Model	[227]	News	Word2Vec	RCNN	This research proposes a novel RCNN model by combining RNN and CNN for different tasks
	[266]	News and Social Media	Word2Vec	A novel tensor-based model	This research develops a novel tensor-based prediction model. It uses domain specific word embeddings.
	[204]	News aresnet filter and kernel sizend Social Media	Word Embeddings	Deep NN	It builds a visually interpreted end-to-end framework for end users

Chapter 3

MMDL: A Novel multimodal Deep Learning Model for Stock Market Prediction

In this chapter, we explore the multimodal applications in the stock market analysis space and explore the utilization of various data sources for a regression problem. We propose a novel deep-learning model for the effective fusion of various data modalities. Initial work in the area shows that textual data and historical market data can be analyzed, and both modality-specific and joint information can be effectively captured by multimodal deep learning models.

3.1 Introduction

It has been shown that the stock markets are influenced by various information sources, including transaction data, news [194] and social media [16], and official announcements [178]. Although early models in the literature focus on single-input models, researchers have also started to analyse multiple data sources for market prediction [5; 227]. Using news articles with fundamental data can lead to better results [37] and social media and market data can be analysed together to predict the stock volatility [30].

One of the main challenges encountered in the literature is modelling these various information sources. One approach that has widely been applied in the literature is early fusion, where multiple input types are directly merged together to create a common input. The common way of doing this is to concatenate these inputs to form a compound vector and feed this newly formed vector into a model to make predictions. Paragraph vectors from textual data and market data [5] and news sentiments and market data [131] have been concatenated together to be used in LSTM based prediction networks. Recently, researchers have started implementing tensors to capture the joint effect of multiple data sources together [120]. They have been employed to capture the interconnectedness between technical indicators and news sentiment data for market analysis [123].

Another technique for utilizing various data types is the late fusion approach, where each input is first analysed separately to capture the unique features. These features are then combined together for prediction. Two separate networks are used in [227] to analyse the input data – financial news and market data, and then are combined together for binary classification.

Although early fusion techniques are good in capturing the joint effect of multiple data sources, they tend to miss unique features that come with each input. On the other hand, late fusion models are able to extract modal-specific features by analysing each input separately. But their main disadvantage is that the joint effect of multiple data sources are mostly lost here. The architecture of a prediction model plays an important role in the end performance and a model with a good architecture can outperform other models even with a less powerful textual representation technique [227].

In this research, we propose a multimodal deep-learning model, which we call MMDL, that targets to capture data-specific information from stock market data and news data and the joint effect of both data sources. We aim to achieve this target by employing two unique sub-networks and one common network for feature extraction. We use an LSTM network for stock market unique network, and use a hybrid CONV-LSTM model [245] for the common network and the news text unique network. The main contributions of this chapter are the following:

- We construct a deep learning-based prediction model with a multimodal architecture consisting of multiple sub-networks to capture both input-specific and common information from financial news and fundamental data. The common network employs a hybrid deep network to analyze the inputs together.

- Glove embeddings are employed to represent the financial news titles. Since there are varying numbers of articles per day, we first create a single vector per article by summing all embeddings together and then take the average of a given day [47].

- We implement deep networks in our subnetworks and a fully connected layer in the output layer to predict the next day's closing price. The fundamental data network utilizes LSTM as they are able to learn from time series data. We use a hybrid CONV-LSTM model for the common and text unique networks, where the input goes through the convolution operation before being analysed by the recurrent layer.

- We evaluate the performance of MMDL against baseline models with price-only and multimodal architectures where the networks use the same input as the proposed model.

- We run tests on a new dataset that includes financial news from Reuters and S&P500 index fundamental data from 2013 to 2019. In our experiments on real-world data, MMDL outperforms current existing models in all popular evaluation metrics. The results show that utilizing financial news, with the architecture of the proposed MMDL network can significantly improve the stock market prediction performance.

  The results show that capturing both unique and common information can improve the prediction performance. We also show that although utilizing financial news can help the model, the architecture of the network also plays an important role in the performance.

The rest of the chapter is organized as follows. In Section 3.2, we show the related work in the multimodal market prediction research, and then introduce the architecture of the proposed model in Section 3.3. Section 3.4 covers the dataset, baseline models, experimentation details and results. In Section 3.5, we give conclusions and our recommendations for future work.

3.2 Related Work

In this section, we review the multimodal financial market prediction literature with a focus on utilizing financial news. Financial news data have been used with technical indicators for market prediction [227; 5]. It has also been shown firm specific news can improve trading strategies [37; 124] and lead to higher returns [165]. In a study by Shynkevich et al. [207], a Multiple Kernel Learning (MKL) based model is employed to analyse stock related news and market data for S&P500.

Various textual representation techniques have been employed to understand the relationship between the markets and breaking news [193]. In the novel news-based prediction system called Arizona Financial Text (AZFinText) [197], proper nouns are used to extract article sentiments. The experiments with S&P500 index prices show that investors can be contrarian: they tend to buy after bad news and sell after good news. It has been shown that using noun phrases for text representation can lead to better prediction results than named entities and bag-of-words in news-based market prediction models [194]. Various word embedding vectors have been employed to represent the textual news data for market prediction [227; 87].

Event representations and embeddings have been utilized to capture more complex features in news-based forecast models [47]. The authors employ a CNN based deep network to create event embeddings from news titles. They take financial news articles from Reuters and Bloomberg and use Word2Vec word embeddings to represent the text data. The authors first build a neural tensor network to learn event embeddings from the news data and then feed these to a CNN based prediction model for S&P500 movement prediction.

Paragraph vectors have also been implemented to encode the textual news data for market prediction [5]. The authors work with both financial news articles and market data to predict 10 company's closing stock price. They use paragraph vectors and build a regression model using LSTM. If a company does not have any article at a given time step, a zero vector is used. On the other hand, if there are multiple articles, an average of these vectors is taken. The authors use early fusion approach and concatenate the textual information and market data into a single compound vector to form the input for the prediction network. The textual data include the news articles from 2001 to 2008 from Nikkei newspaper and the market data is extracted for the 10 most frequently mentioned companies in these articles. The results show that LSTM outperforms the traditional machine learning techniques (MLP, SVR, simple RNN), while paragraph vectors seem to be a better representation technique than bag-of-words. They also show that using financial news and market data together improves the prediction performance.

Sentiment analysis models for stock market prediction have been widely explored. A research by Li et al. [126] builds an SVM based prediction model with dictionary based sentiment analysis model using Harvard IV-4 Dictionary and the Loughran–McDonald Financial Dictionary. Their data consists of company-specific and market related news from FINET (www.finet.hk) and five years daily stock prices from Hong Kong Stock Exchange. The results show that the proposed approach outperforms bag-of-words only representation. In another research, they show that using article summarization rather than the full article can improve the prediction model performance [128]. Another study by Li et al. [131] use market data and financial news data for sentiment analysis based prediction model. They generate news sentiments time series and technical indicators time series and concatenate them by date to create input for an LSTM network. In the experiments on HKSE data multimodal model outperforms both price-only and news-only models in stock level and sector level. They also show that LSTM network outperforms SVM and MKL based baseline models.

Researchers have utilized CNN for textual analysis and feature representation. CNNs employ convolving filters to capture local features in the data [107] and they have been shown to perform well with textual data [92; 98]. CNNs and LSTMs have been implemented together to utilize both networks advantages in hybrid market analysis models [227]. A research by Vargas et al. [227] uses financial news titles with market data and technical indicators for prediction of intraday directional movements of S&P 500 index. The research analyses the input information separately and later merge the extracted features for binary classification. The textual data is represented using Word2Vec word embeddings and fed into a CNN and later to an LSTM network, and the market data and technical indicators are fed directly into an LSTM network. The outputs of these two separate networks are fed into the output layer of the proposed hybrid model which generates a binary class for the movement direction. They conclude that textual data and technical indicators improve the performance and the proposed CONV-LSTM model is a better structure than CNN for the index prediction task. In their experiments, they show that although event embeddings (EB) are better representations than word embeddings, their model outperforms EBNN because of the network architecture.

In the study by Lee and Soo [108], a Recurrent Convolutional Neural Network (RCN) is built for market prediction using financial news and market data. The authors take a different approach and analyse the textual data separately before combining the features with price vector. The news titles are vectorized using Word2Vec word embeddings and fed into a CNN to capture local features of the input. The output of CNN is a single scalar and combined with market data to form the input for LSTM network. The experiments using the data of TWSE index and four stocks in TWSE show that the proposed RCN model is better than LSTM with lower RMSE.

A novel tensor-based model was proposed by Zhang et al. [266] to predict multiple correlated stocks at the same time by incorporating the joint effect of multiple data sources. They build a tensor to capture the relations between the events from news articles and investors sentiments from social media data. Two separate matrices – a stock quantitative feature matrix (market data) and a stock correlation matrix – are constructed to help with the tensor deconstruction. The experiments on the China A-share stock data and the HK stock data show the effectiveness of the proposed model.

3.3 Model Design

Stock market prices are affected by various data types such as historical stock data, financial news data, and sentiment on the social media. These data sources have been incorporated into deep learning models in different ways. One of the strategies is to merge data vectors from multiple data sources and then apply a deep prediction network. Another methodology is to implement separate networks to capture data-specific features and later concatenate these at the output layer for prediction.

These strategies either capture common features from multimodal networks or input-specific information from each data source. However, it is important to employ both of these information for the analysis model. To this end, we propose to utilize both inter-modality and intra-modality information in the analysis by creating two unique networks and a common network in the feature layer.

In this section, we introduce the design of the proposed multimodal architecture - MMDL for stock market prediction. We propose to utilize both inter-modality and intra-modality information by analysing the inputs both jointly and separately. The model, shown in Figure

3.1, consists of an input layer, a feature layer that includes three sub-networks to capture the unique and common information, and an output layer.

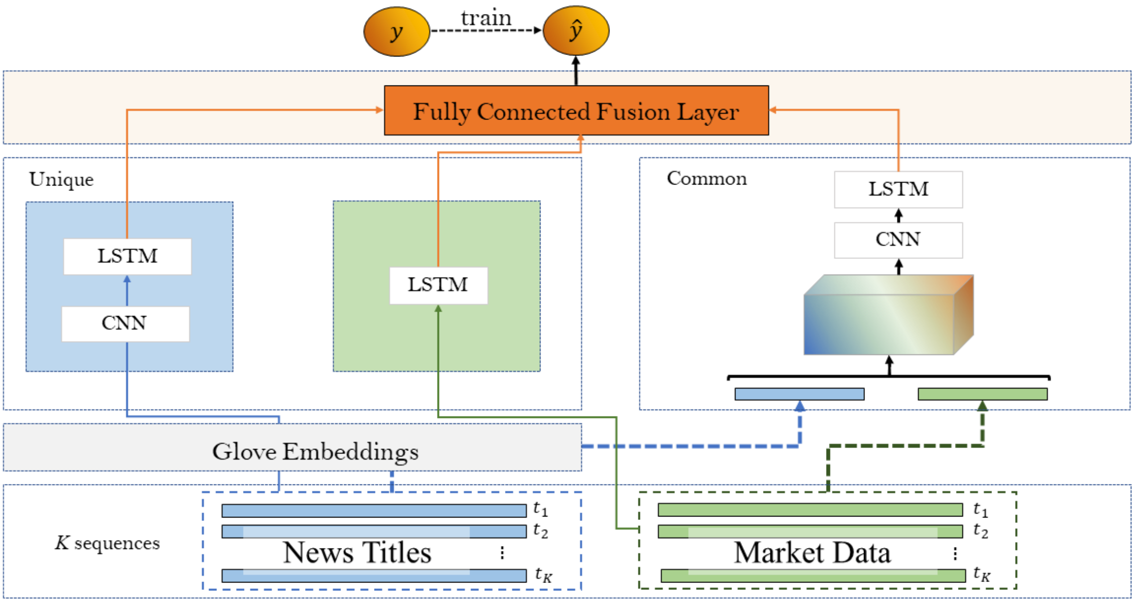

Figure 3.1: Demonstration of the MMDL architecture design.

3.3.1 Input Layer

The Input layer takes stock market data and textual news data for processing. We describe these below.

Text Data Layer

We use Glove embedding to represent the news. To manage the irregularity in the data, we first create a single vector per article by summing all embeddings together and then averaging the resulting sum across all embeddings in a day [47].

It has been shown that prediction models benefit from incorporating financial news data [5] and using news titles can lead to better results than the whole article body [204]. We take all financial news titles per trading day and clean the text for processing. To manage the irregularity in the number of articles per trading day, we average the titles' embedding vectors per day. First, we create encoded vectors of each title using pre-trained Glove embeddings [176] with the embedding dimension of 300. Then following [47], for each day in our sample, we sum up the vectors converted from each word in the word embedding and averaged the

Table 3.1: Description of market data attributes

Attribute	Description
Open/Close Price	The first and last price at which a stock is traded on a given trading day
High/Low Price	The highest and lowest price at which a stock is traded on a given trading day
Volume	The total number of shares traded for a stock on a given trading day

resulting sum across all embeddings associated with that day. Same as fundamental data, we take 10 days of financial news to predict the next day's closing price.

Stock Data Layer

In the stock market analysis literature Open, Close, High, Low prices and Volume are widely used for prediction as they show the price movements of a stock on a given day and carry some predictive value for understanding market patterns [149; 50]. We take these five attributes to capture the fundamental information of the market on a given day. We take 10 days of fundamental data and normalize them to be within the range of [0, 1] to predict the next day's closing price. These five attributes are explained in Table 3.1.

3.3.2 Feature Layer

The input data is processed through the Feature Layer which consists of three sub-networks: text news unique network, stock market unique network and common network. We describe these networks below.

Text News Unique Network

The aim of this sub-network is to extract modality-specific features from the textual news data. The daily average embedding vectors of news titles are taken to be analyzed to capture the influence of the financial news. These embedding vectors are fed into a hybrid CONV-LSTM deep network for processing. Here the inputs are first processed with CNN and then the outputs of CNN go through an LSTM network for further analysis. This technique aims to utilize both model's advantages. CNNs perform well in capturing local features [107] and LSTM networks are able to extract the context information and capture temporal dependencies [79].

The input of this sub-network is the averaged word embedding vectors of 10 days' news titles and each day's input is represented with 300 dimensional vector.

We first implement a one-dimensional convolutional layer, which is a temporal convolution. A temporal max-pooling layer is added which forces the network to retain only the most useful local features produced by the convolutional layer [47]. In order to add non-linearity to the model we implement an activation with a rectified linear unit (ReLU) function. Lastly, a dropout operation [212] is applied for regularization before the recurrent network.

After the dropout layer, we implement a recurrent network with LSTM layers to capture the temporal characteristics of the textual data.

Stock Market Unique Network

The second unique sub-network is for market data only and it takes inputs from the stock data layer and uses a recurrent network (RNN) with LSTM layers for feature extraction. LSTM networks are widely used in time series analysis as they are able to model long-term dependencies in sequence data and overcome the vanishing gradient problem of standard recurrent neural networks [227].

We use 10 days of fundamental data as the network input where each day is represented with the selected 5 attributes.

Common Network

It has been shown that analysing the joint effect of multiple data sources can improve the model performance [5]. After analysing the market data and financial news separately, we build a separate sub-network to capture their influence together. Similar to text unique layer, here we also employ CNN and LSTM together as they can help to capture the local features and temporal dependencies in the data. For the representation of two input sources, we experiment with two techniques which are explained below:

- **Concatenation**: We first concatenate the textual input and fundamental data input by date to form a compound vector. As shown in previous sections, the dimension of textual input is (10, 300) and the dimension of fundamental data input is (10, 5).

After concatenation, the compound matrix's dimension is (10, 305). This input is then fed to a CONV-LSTM deep network to obtain the common information. This network has the same structure as the text unique network.

- **Tensor**: Here the input of the common network is a 4d tensor formed by combining news embedding vectors and fundamental data. We first employ a two-dimensional convolution operation and then a two dimensional max-pooling layer is implemented to capture the most important features. We try ELU and ReLU as activation functions, and then apply a dropout operation for regularization before the LSTM based recurrent network.

3.3.3 Output Layer

The last layer in our architecture is a traditional fully connected layer. The input of this layer is the concatenation of the outputs of the three sub-networks: text unique network, fundamental unique network and common network. The objective of our work is to predict the next day's closing price and we employ a regular deeply connected dense layer for the forecast.

3.4 Experiments

To evaluate the effectiveness of our proposed architecture for stock market prediction, we conduct a series of experiments using actual S&P500 transaction data. In this section, we describe these experiments and their results.

3.4.1 Dataset

In our experiments, we create a new dataset that includes historical stock data for the S&P500 index from Yahoo! Finance and financial news headlines from Reuters between January 1, 2013, and December 31, 2019.

- **Stock Market Data**: The quantitative data includes fundamental information of S&P500 index that are extracted from Yahoo! Finance. We take daily Open, Close, High, Low prices as well as the daily Volume for the index. An excerpt of the market

Table 3.2: Sample extracted from the S&P index historical data. All prices are in US Dollars.

Open	High	Low	Close	Volume
1426.19	1462.43	1426.19	1462.42	4202600000
1462.42	1465.47	1455.53	1459.37	3829730000
3227.20	3240.08	3227.20	3239.91	2160680000
3247.23	3247.93	3234.37	3240.02	2428670000
3240.09	3240.92	3216.57	3221.29	3013290000

Table 3.3: Descriptive statistics of fundamental data. All prices are in US Dollars.

Variables	Min	Mean	Max	Std
Open	1426.19	2261.40	3247.23	435.47
Close	1457.15	2261.86	3240.02	435.18
High	1461.89	2271.11	3247.93	436.64
Low	1426.19	2250.89	3234.37	433.88
Volume	1296540000	3549008796	7609010000	661449626.47

data is given in Table 3.2. The descriptive statistics of the market data, given in Table 3.3, shows that there is huge variation in volume, whereas the price data doesn't seem to be too volatile as they tend to be around a company's fundamental value.

- **Text News Data**: The dataset includes financial news articles from Reuters for the same period and we only employ news titles in our experiments. Each row includes the title and publishing date of a news article. The date is used to align the articles with the daily market data. After pre-processing and removing duplicates and unrelated articles, the data includes 527047 news headlines. The minimum number of articles per day is 38 and the maximum number of articles per day is 998. News headlines from previous trading day are vectorized through pre-trained Glove word embeddings and used for analysis. We use Glove with dimension of 300. Table 3.4 shows an example from the textual data.

Before any pre-processing, we eliminate the non-trading days from the set which leaves us with 1761 trading days. The first 80% of the total data is used for training and validation, and the remaining part used for testing. We employ the same division for our proposed model and the baseline models.

Table 3.4: Sample extracted from the finanacial news data

Headline	Date
China Dec official factory PMI flat at 50.6-NBS	2013-01-01
Best Buy loses two board directors	2013-01-01
Repo rates lowest since March 2018 on collater...	2019-12-31
U.S. crude stocks fell in latest week -API	2019-12-31
Wall Street edges higher; S&P closes decade wi...	2019-12-31

Table 3.5: Network parameters for the experiments

Layer	Parameter Values
Convolution Layers	filter size = {32, 64. 128} and kernel size = {2, 3}
Maxpooling Layers	pool size = {2, 3}
Dropout Layers	rate = {0.2, 0.5}
Activation Layers	functions = {None, ReLU, ELU}
LSTM layers	neurons = {50, 64, 128}

3.4.2 Experimentation Settings

We implement MMDL and the baseline models using Keras and TensorFlow. The structure of MMDL's Tensorflow model is given in Figure 3.2. The goal of this network is to predict the next day's closing price using historical market data and financial news. As shown in the figure, MMDL employs multiple sub-networks for the analysis of inputs separately and together. As a result, it increases the number of parameters needed for the whole network. But this factor also enables the model to learn both intra-model and inter-model features which leads to better forecast results.

We adopt mean squared error (MSE) as our loss function to calculate the prediction errors and optimize the parameters of our network. MSE is the mean of the squared differences between the predicted and true values.

We employ a stochastic gradient descent (SGD) algorithm as our optimizer function with an initial learning rate of 0.1 and set the momentum to 0.9 to train the network. We try two different epoch sizes and batch sizes when we train the models, where we use 100 and 200 for the epoch size and 64 and 128 for the batch size.

The proposed network is constructed based on CNN and LSTM deep networks and use pooling and dropout operations. The network parameters used in the experiments are presented in Table 3.5.

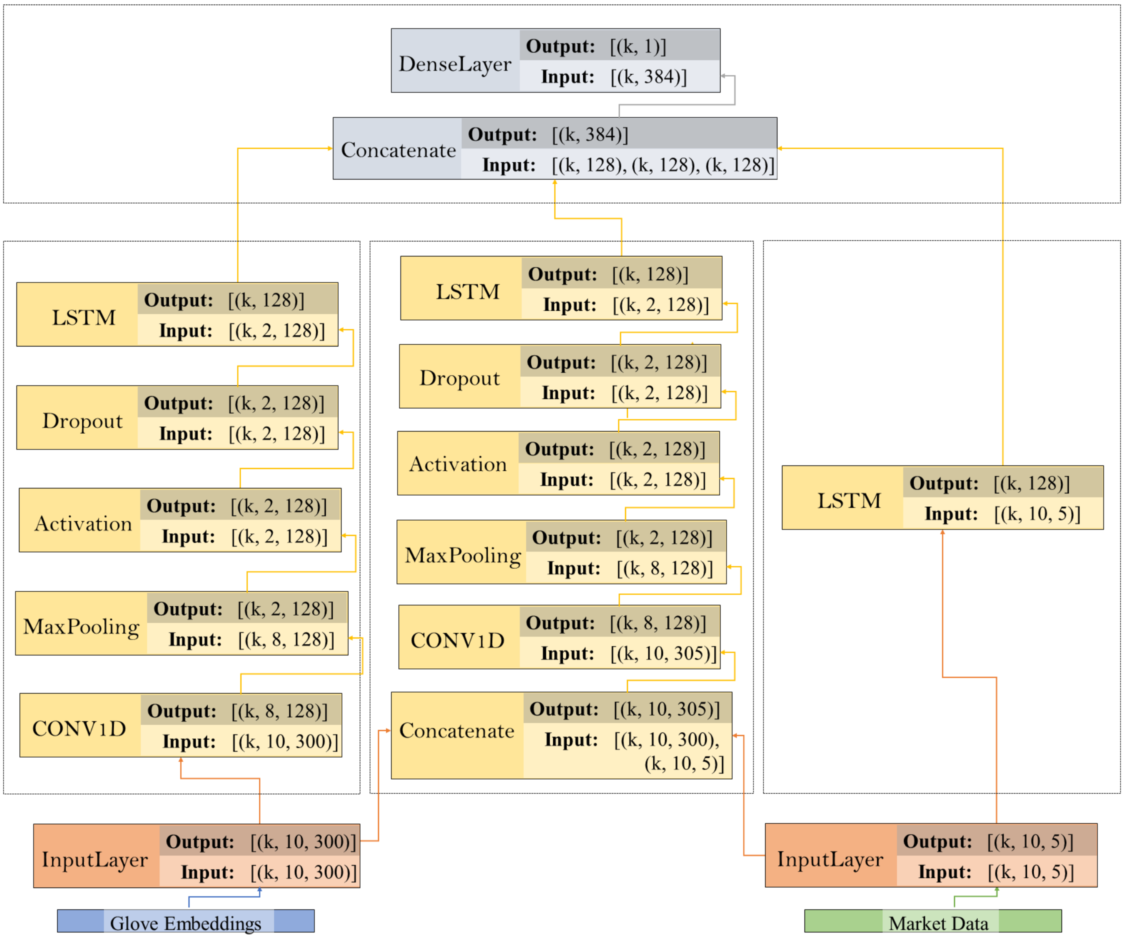

Figure 3.2: MMDL Keras model structure

3.4.3 Baseline Models

In our experiments, we build our baseline models based on three widely used architectures to compare our model against: a market data only model, a late-fusion multimodal network that uses market data and financial news, and an early-fusion multimodal prediction model that also uses market data and financial news. We give the details of these implementations below:

- **LSTM model** using only the market data [157]: Fundamental data has always been a crucial part of market analysis, and market data-only deep learning models have been widely implemented and tested for price prediction [69]. This baseline model employs LSTM network to predict the next day's price based on the fundamental data as LSTMs work better with time-series data. The architecture of the model is given in Figure 3.3A.

- **RCN model** using market data and financial news [108]: The word embeddings from the financial news titles are analyzed through CNN and RNN and then combined with the stock market data for the price prediction. The architecture of this model is presented in Figure 3.3B.

- **CONV-LSTM model** with late fusion using market data and financial news [227]: Hybrid models using CNN and LSTM have been employed to utilize the advantages of both networks: CNNs to capture the local features and LSTM to capture the temporal characteristics of the data. The CONV-LSTM model analyzes the textual and fundamental data separately, where the text news data is analyzed using CNN and LSTM, and the stock market data is analysed with LSTM. A fully connected output layer is employed to make a prediction based on the concatenated feature vector. Figure 3.3C shows the architecture of this model.

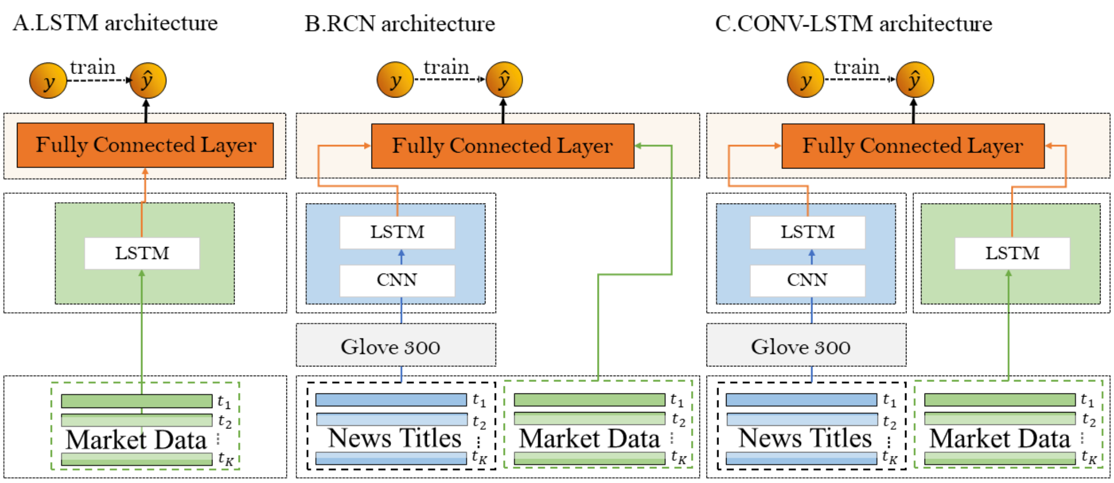

Figure 3.3: Three baseline models. A: The LSTM model using only market data [157]; B: The RCN model with both news and market data [108]; C: The CONV-LSTM model with both news and market data [227].

3.4.4 Evaluation Metrics

The aim of this research is to predict the next day's closing price, and we evaluate the performance of the baseline models and our proposed deep network for regression problems. We employ three error metrics that are used to evaluate the difference between the actual and predicted prices root mean squared error (RMSE), mean absolute error (MAE), mean

Table 3.6: Description of the selected evaluation metrics

Attribute	Description
MAE	The average of the absolute values of the differences between the predicted and true prices.
MAPE	It is similar to MAE, but unlike MAE, MAPE measures the prediction accuracy in percentages.
MDAPE	An error metric that is mostly used to measure the regression model performances where it calculates the median value of the errors (i.e. differences between the predicted and true prices).
RMSE	The square root of mean squared error (MSE). It is used to show the predictive power the model, where a low number means that the prediction is close to the actual price.

absolute percentage error (MAPE) and median absolute percentage error (MDAPE). For all these metrics, a lower value is better. We describe these metrics in Table 3.6.

3.4.5 Results and Discussion

The focal point of our experiments is to test the performance of MMDL in stock market prediction. Table 3.7 presents the details of our experimental results based on root mean squared error (RMSE) and mean absolute error (MAE). The results of the LSTM baseline model highlight the informative characteristics of the market data for price forecasts. Both the RCN model and CONV-LSTM model employ financial news alongside the market data for market analysis. The comparison with the LSTM model, which only uses market data, indicates that using financial news improves the prediction performance.

It can be observed from Figure 3.4 that MMDL outperforms the baselines in all metrics - RMSE, MAE, mean absolute percentage error (MAPE) and median absolute percentage error (MDAPE) significantly. The comparison with CONV-LSTM model, which gives the best results among the baselines, shows that a thorough multimodal network architecture can improve the model performance. The CONV-LSTM model only captures the data-specific features and combines them at a later stage for prediction, whereas, we propose to utilize both modal-specific features separately and the joint effect of the textual and market data together. The prediction plot for MMDL predicted prices vs. true prices is given in Figure 3.5. The results show that utilizing both unique and common networks improves the prediction performance and MMDL improves the state-of-the-art performance significantly.

Table 3.7: Results from our experiments

Model	LSTM	RCN	CONV-LSTM	MMDL-V	MMDL-T
MAE RMSE	27.74 36.89	20.22 36.8	18.23 35.39	15.48 31.07	14.54 27.69

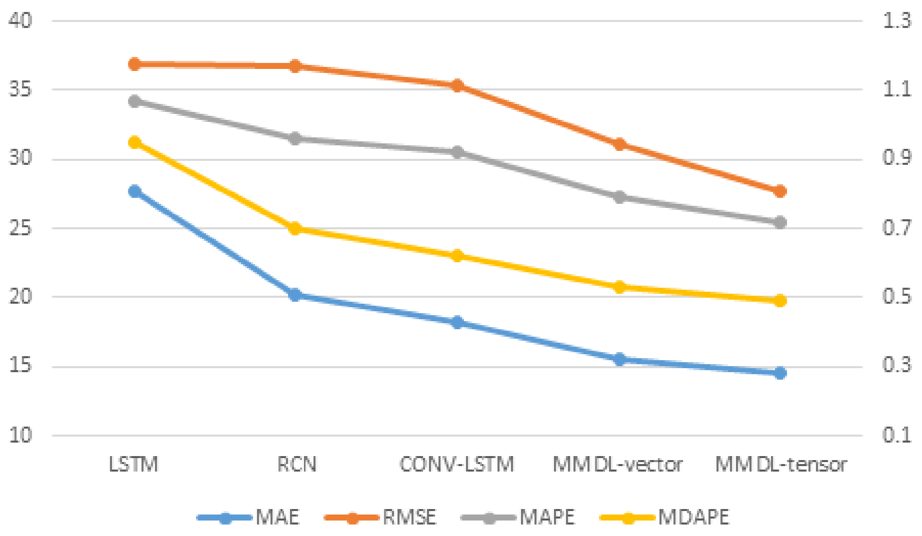

Figure 3.4: Comparisons of models based on different metrics.

We also experiment with options of using tensors or vectors to represent the common network input.

We aslo experiment with two main structures to form the common network input from two data sources: direct concatenation and tensor formation. Our inputs are given as:

```
Nt: [i, j]
Pt: [i, k]
```

where i is the number of days (10) we take to predict the next day's closing price, j is the dimension of Glove embeddings (300) and k is the number of attributes (5) we employ from the market data. We ran tests with the following models:

- **MMDL-V.**: here we directly concatenate the market data and title embeddings by date and feed the compound vector into the network. The network structure is explained in Section 3.3.

- **MMDL-T.-matmul**: We employ TensorFlow's matmul operation to form an input

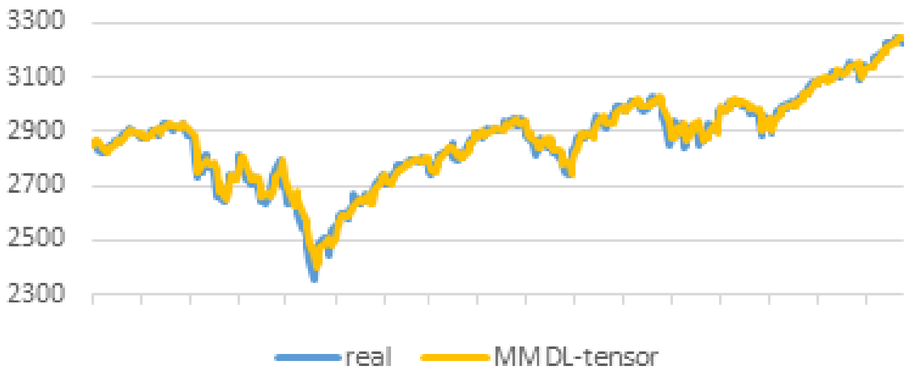

Figure 3.5: Plot of real stock prices vs MMDL-Tensor predicted prices.

tensor from reshaped fundamental and textual inputs. The formed tensor is analyzed through the same CONV-LSTM network structure as above.

- **MMDL-T.-einsum-3d**: Here we use einsum operation from TensorFlow to form a tensor with the rank 3 which also includes the batch size, and employ the same common network structure.

- **MMDL-T.-einsum-4d**: We again employ einsum operation to form a tensor but with the rank 4 which also includes the batch size. Unlike the previous ones, here we use 2 dimensional convolution and pooling operations., and employ the same common network structure.

Table 3.8 shows the comparison of prediction results with various input structures for the common network. Among these models, MMDL-T.-einsum-4d give the best results and the improves the results of model MMDL-V. where we concatenate the inputs directly to create a compound vector for the common network. The comparison of MMDL-T.-einsum-4d with other tensor based models shows that a tensor with a higher rank is more likely to preserve the input data during conversion. Overall, MMDL improves the prediction performance significantly both with tensor and direct concatenation structures.

We also experiment with varying network parameters for our proposed model. Table 3.9 shows the evaluation results of these tests. As can be seen from the table, the model gave the best performance with 128 LSTM layers and convolutional layers with 128 filters and kernel size of 3. Including an Activation layer with Relu function also improves the model and a dropout rate of 0.5 outperforms the models with a droupot rate of 0.2.

Table 3.8: Experiments with common network input. C - concatenate; T1 - matmul; T2 - einsum with rank 3; T3 - einsum with rank 4

Model	MAE	MAPE	MDAPE	RMSE
MMDL-V.	15.48	0.79%	0.53%	31.07
MMDL-T.-matmul	22.14	0.98%	0.76%	36.48
MMDL-T.-einsum-3d	17.69	0.78%	0.61%	29.26
MMDL-T.-einsum-4d	15.05	0.73%	0.51%	28.41

Table 3.9: Comparison of MMDL performances with different network parameters

LSTM	Conv.	Pool.	Actv.	Dropout	MAE	MAPE	MDAPE	RMSE
64	128/3	3	N/A	0.2	17.16	0.91	0.59	35.74
50	64/3	2	N/A	0.2	20.12	0.90	0.70	33.73
64	64/3	3	Relu	0.5	20.53	0.91	0.69	34.58
128	128/3	3	Relu	0.5	15.48	0.79	0.53	31.07

The results also show that even with the varying parameters, MMDL outperforms the baseline models in stock market prediction. This again confirms that employing financial news and market data together improves the performance and it is crucial to capture the modality-specific features alongside the interconnectedness of multiple data sources.

3.5 Conclusion

In this chapter, we proposed MMDL - a novel multimodal deep learning model for stock market prediction using financial news and stock market data. MMDL utilizes the unique information from both input types, and their joint effect on the market prices. We ran experiments on S&P 500 index data from 1 January 2013 to 31 December 2019 and financial news titles for the same period. We extracted the news articles from Reuters and employed Glove word embeddings to represent the titles. MMDL consists of three sub-networks: a unique network for each data source and a common network to analyse these inputs together. In our networks, we apply deep learning models to learn the information from the data. An LSTM network is used for market data, and a hybrid CONV-LSTM network is constructed for textual data and common network. We use CONV-LSTM network to take advantage of both models' capabilities – capturing local features with CNN and modelling time series data with LSTM.

The experiments on S&P500 index data showed that financial news plays an important role in making market decisions, and incorporating them influences the forecast model positively. The baseline models showed that the overall architecture of the model and the representation techniques of input sources play a crucial role. The experiments confirm that it is important to capture the unique information from data sources and their interconnected relationship. We evaluate the results using MAE, MAPE, MDAPE and RMSE and show that MMDL outperforms the baseline models in all of the metrics. These results show that our proposed architecture is effective in modeling the price movements in stock markets and can help to improve the financial returns.

We also ran tests using two different representation techniques for the common network input: vector and tensor. The tests show that tensors are better at representing the joint effect of the various data sources than a compound vector. Experiments with varying parameters show that the network structure affects the prediction performance.

For future work, we would like to incorporate more data sources, such as related social media data and technical indicators, to analyse their effect on the prediction performance. We showed that analysing multiple data sources separately and together can improve prediction, and with more data sources the model can become cumbersome. So the representation of these data will play an important role in the model performance. To test this, we would like to employ more complex textual representation techniques such as event embeddings and sentence embeddings.

Chapter 4

MStoCast: Multimodal Deep Network for Stock Market Forecast

Stock market analysis is a complex task that involves various types of data, such as web news, historical prices, and technical market indicators. Recent research in this area focuses on analyzing these modalities either separately or all together, but the underlying correlation patterns in the multimodal data were not captured. To address this issue, in this chapter, we propose three multimodal learning models: MStoCast (Multi-modal Stock Forecast), MStoCast-Attention and MStoCast-Sentiment. First, we propose MStoCast, which consists of a common network to capture joint information and a unique network that discovers bi-modal information from the inputs. In the second model, MStoCast-Attention, we utilize Attention mechanism to build a cross-modality module to capture intra-modal and inter-modal information from the modalities using cross-modal attention module and sentiment module. While cross-attention module focuses on market data and technical indicators, in the third model, MStoCast-Sentiment, we employ an LLM in the sentiment module to extract and utilize sentiments from financial news headlines. In all three models, the features are amalgamated together in the fusion layer and then processed through a fully connected layer to predict the direction of the closing price movement. Experiments using real-world datasets show that MStoCast, MStoCast-Attention and MStoCast-Sentiment models significantly outperform other state-of-the-art models.

4.1 Introduction

The price movements in stock markets are influenced by various data sources, including historical price data and technical indicators [226], financial news [192], social media [29], and official announcements [55]. It has been demonstrated that analyzing these multiple data modalities collectively can aid in capturing the underlying patterns of stock movements [122; 203], thereby making stock market prediction a multimodal learning task [217]. Employing effective multimodal representation and learning techniques to capture the influence of these diverse data modalities is crucial for model performance. Therefore, it has become essential to appropriately integrate these various types of data (e.g., financial news, stock market data, and technical indicators) to predict price movements [118; 122].

While the common approach involves concatenating raw features from input modalities [4], this vector-based method might struggle to extract the interconnectedness of the data sources [265]. Although matrices have been utilized to handle two modalities, they pose challenges when dealing with three or more input modalities, and they may also fail to capture intra-modal information. However, it's imperative to capture both intra-modal and inter-modal features [229].

Lastly, capturing bi-modal relationships among the input data is a crucial aspect that should also be taken into consideration, especially when dealing with three or more modalities. However, existing models often might fall short when dealing with three or more modalities.

To address the challenges of multimodal learning for stock market forecast, we construct three market analysis models: MStoCast (Multi-modal Stock Forecast), MStoCast-Attention and MStoCast-Sentiment. In these models, we utilize historical market data, financial news and technical indicators as our input modalities. These models aim to capture the modality specific and joint features from the input modalities, as well as the cross-modal information among various input modality pairs.

We first propose MStoCast, a stock movement prediction model that employs deep learning algorithms (such as ResNet [73] and Transformers [44]) to analyze multiple input modalities (i.e., financial news, stock market data, and technical indicators). It's important to note that, although we have chosen ResNet and BERT (two of the most advanced models at present) for this research, they can be substituted with other CNN and RNN models to accommodate the requirements of different problems. Our objective is to capture inter-modal and intra-modal information, as well as bi-modal relationships among the input data modalities, by introducing common and unique sub-networks.

MStoCast utilizes two types of information extracted from multimodal input sequences. The first type of information involves the bi-modal interactions among the pairs of input modalities, representing distinct and unique insights. The second type of information encompasses common patterns, capturing both inter-modal relationships and intra-modal details derived from all input modalities. To achieve this, our approach involves designing both a unique network and a common network, each aimed at extracting the respective kinds of information crucial for accurate stock market prediction.

The unique network begins by implementing early information fusion where we combine the inputs in pairs to form feature matrices. These matrices are then processed through three separate ResNets, enabling the extraction of bi-modal relations from the modalities. Conversely, the common network commences by employing Long Short-Term Memory networks (LSTMs) to extract modality-specific information from each input modality. These extracted intra-modal details are amalgamated into a multimodal tensor through their outer product. Subsequently, Convolutional Neural Networks (CNNs), specifically ResNets, are employed to extract inter-modal information from this tensor. The resulting information from both the common and unique networks is then concatenated, forming a compound vector. Through the utilization of global average pooling, a feature map is obtained. The final stages encompass the creation of two fully connected layers, responsible for analyzing the feature vector. Additionally, another fully connected network is employed to facilitate market movement prediction.

In our market prediction model, we incorporate historical market data, a selection of seven technical market indicators, and financial news. MStoCast effectively analyzes these three input modalities to anticipate the directional movement of closing prices. Textual data is encoded using BERT sentence embeddings, while a combination of CNNs and RNNs is leveraged to construct a robust multimodal representation model tailored for market prediction. It's worth emphasizing that, although we opt for ResNet and BERT—two of the most advanced models currently—in our approach, these models can be substituted with alternative CNN and RNN models that align with the unique requirements of other problems. Our primary focus remains on capturing the crucial inter-modal and intra-modal information, as well as comprehending the intricate bi-modal relationships within the input data, all achieved through the integration of the proposed common and unique sub-networks.

In the MStoCast-Attention model, we propose a cross-attention module that first analyzes the inputs separately and then processes them in pairs in the cross-attention module. We first build a representation layer, where we employ two LSTM networks to analyze technical indicators and market data modalities separately. We also use Transformer-based embeddings to encode the textual news data and feed them into CNN to capture modality-specific information. These features are used in two ways: fed as inputs to the cross-attention module, and also merged with the other features in the Fusion layer. This design aims to capture cross-modal information in the cross-attention module while not losing the intra-modal features and utilizing them in the fusion layer for market prediction.

Our proposed cross-attention module takes the outputs of the previous layer and forms six input pairs. Each pair is then processed separately using an attention network. The outputs of these networks are then merged with the modality-specific features in the Fusion layer. The structure of the fusion and output layers is similar to the MStoCast model, where we use a feed-forward network to process the newly formed feature vector for price movement prediction.

Lastly, we propose MStoCast-Sentiment market prediction model where we utilize Large Language Models (LLM) for sentiment extraction. We first process market data and technical indicators separately using two LSTM networks and then feed them into our cross-attention module to extract the cross-modal information. We also employ FinGPT [250] LLM in our sentiment module to extract sentiments from the financial news headlines. This module takes the news headlines in text format and feeds them into FinGPT API. The sentiments from the LLM are then fed into the fusion layer alongside the outputs of the cross-attention module. The next steps for price movement prediction are similar to the previous models.

The main contributions of this chapter are as follows:

- We propose a multimodal prediction model MStoCast, where we build a common network and unique network for feature representation. The common network focuses on modality specific and joint information, while a unique network consisting of three separate residual networks is constructed to capture the bi-modal relationships between pairs of input features.

- We build an attention based multimodal model MStoCast-Attention, where we employ Transformer based multi-head attention network. We first process the inputs separately to capture the intra-modal information and then analyze them using our cross-attention module to extract cross-modal features.

- In our last model MStoCast-Sentiment, we build a cross-attention module to process the numerical input modalities and employ LLMs in the sentiment module to analyze the financial news headlines. The extracted sentiments and the outputs of our cross-attention module are then utilized together for price forecasting.

- In the tests using real world datasets, all three models outperform the baseline models in all evaluation metrics both on S&P500 index and individual stock prediction. The results also show that Attention based models perform better in market prediction.

The rest of this chapter is organized as follows. In Section 4.2, we show the related work in the stock market prediction domain. We then introduce the specifics of the proposed models in Sections 4.3, 4.4 and 4.5. Sections 4.6 and 4.7 cover experiments that include market prediction tests. In Section 4.8, we give conclusions and our recommendation for future work.

4.2 Related Work

Quantitative indicators, such as historical market data and technical indicators, have been widely explored and have shown to be effective for stock market prediction [20; 162]. A novel State Frequency Memory recurrent network is proposed by Zhang et al. [263] to make long and short-term predictions using the historical market data. The study by Fischer and Krauss [58] employ LSTM to predict the movement direction of S&P500 index prices where they show that LSTM outperforms Random Forest, DNN and Logistic Regression Classifier based models. Another study by Wang et al. [233] uses raw financial trading data in their novel hybrid model called CLVSA: A Convolutional LSTM Based Variational Sequence-to-Sequence Model with Attention.

The fusion of market data and technical indicators has also garnered attention in the stock market prediction research. The research by Nelson et al. [158] delve into the application of LSTM networks to forecast price trends by leveraging historical price data and technical indicators. Employing LSTM networks with an attention mechanism, the study by Chen and Ge [27] focuses on predicting Hong Kong stock movements through market data and technical indicators, highlighting the effectiveness of the attention mechanism in LSTM-based prediction models. In a similar vein, Li et al. [112] propose an RNN-based strategy for predicting three prominent Chinese stock market indexes, using a multi-task RNN for feature extraction from raw market data.

Taking a different approach, Schumaker et al. [196] introduced a novel market prediction model, AZFinText, which harnesses proper nouns for new data representation. Leveraging the success of deep learning in other domains, researchers have begun exploring their utility in Natural Language Processing (NLP) tasks. A study by Vargas et al. [226] utilize Word2Vec word embeddings to encode textual news data for CNN and LSTM-based prediction model.

The integration of financial news data, using NLP techniques, alongside quantitative indicators has become a prevalent theme in stock market prediction. The research by Chen et al. [29] incorporated sentiment dictionaries to extract features from social media news, while Li et al. [127] demonstrated the superiority of using article summaries over complete article bodies for prediction. Analyzing events extracted from news articles in conjunction with technical indicators, Nuij et al. [164] predict FTSE 350 index prices. The utilization of CNN-based event embeddings for market prediction via financial news articles is explored by Ding et al. [46]. In a noteworthy multimodal study, Zhang et al. [265] employ tensors to jointly model news articles and social media sentiments, providing predictive insights for market prices in China A-share and Hong Kong Stock Market.

In recent years, the Attention mechanism has also been implemented for stock market analysis. A hybrid attention network (HAN) was developed by Hu et al. [84] that analyzes related news data from market prediction. In another market analysis study, Xu et al. [247] utilize social media and market data in their attention-based prediction model. A study by Daiya and Lin [36] employ Transformer model and build multi-head attention modules for market analysis. Ramos-Pérez et al. [183] use Transformer model for volatility prediction in stock markets.

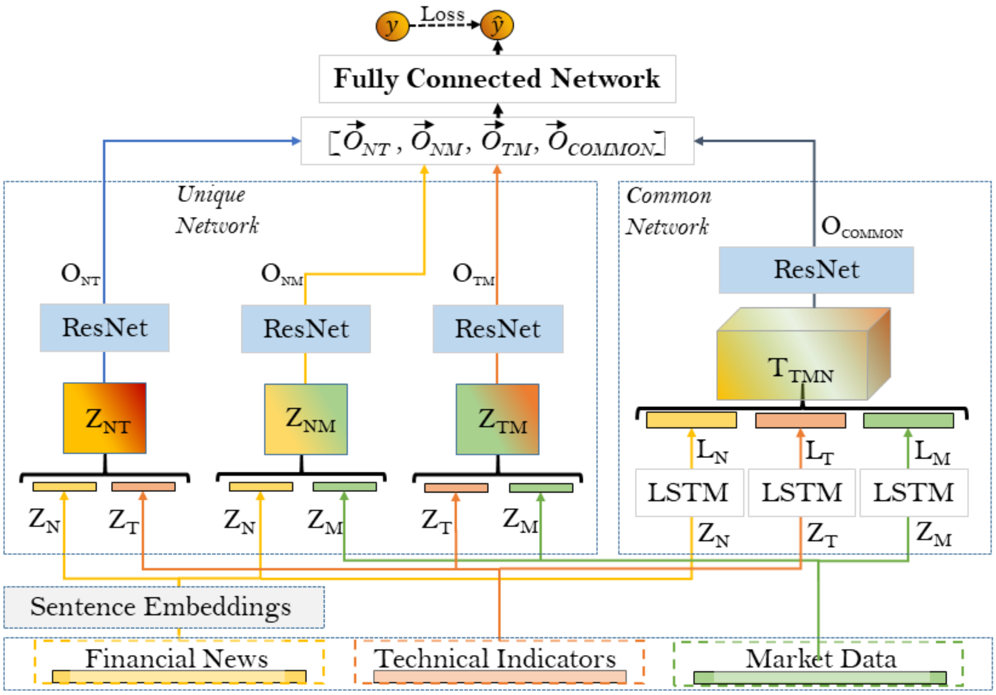

Figure 4.1: The framework of our MStoCast model

4.3 MStoCast

In the design of our MStoCast method, we aim to utilize both common and unique types of information. This entails capturing modality-specific as well as joint information, while also delving into the modeling of bi-modal relationships across various data modalities. We initially take historical market data and financial news as our main data sources. Then we derive a list of seven technical indicators from the market data and utilize three data modalities: market, technical indicators and financial news, where sentence embeddings are employed to encode the textual financial news data. The three raw input features are denoted as Z_T, Z_M, Z_N, representing technical indicators, market price data, and news data, respectively. These feature vectors are analysed with three separate LSTMs in the common network and concatenated as raw features to form bi-modal matrices in the unique network. By performing early information fusion in the unique network and feature fusion in the common network, MStoCast aims to capture joint and cross-modal information from the input modalities.

The information gathered from both the common and unique sub-networks is then amalgamated through a fusion layer to perform multimodal market movement prediction. The design of our method is presented in Figure 4.1.

4.3.1 Unique Network

In the unique network, raw concatenated input matrices are analyzed through convolutional networks, such as ResNets, to capture the bi-modal relationships among the input data, which we detail as follows.

We begin by pairing up the raw input features, thereby generating three distinct input matrices. These matrices are as follows:

$$\begin{aligned} Z_{TM} =& Z_T \oplus Z_M \\ Z_{NM} =& Z_N \oplus Z_M \\ Z_{NT} =& Z_N \oplus Z_T \end{aligned} \tag{4.1}$$

where Z_{NM} is formed by the outer product (denoted by $\oplus$) of raw features from the market data Z_M and news embeddings Z_N; Z_{TM} by using raw features from the market data Z_M and technical indicators Z_T; and Z_{NT} is formed using news embeddings Z_N and technical indicators Z_T.

We then proceed to construct three distinct ResNets, each responsible for extracting underlying cross-modal information from the input matrices. Each network in the unique sub-network includes two residual blocks. In each residual block, the input is first processed using convolutional layers, and then we apply batch normalization and analyze the output with an activation layer using Rectified Linear Unit (ReLU) [2] function to add non-linearity. Batch normalization lets the network to train fast by keeping the mean output close to 0 and the output standard deviation close to 1 and the standard ReLU function returns the element-wise maximum of 0 and the input. The output goes through another round of convolution and batch normalization processes. We then add the initial input to the output of second batch normalization layer and feed it into the last activation layer.

We then produce three feature matrices O_{NT}, O_{NM}, O_{TM} by processing the three bi-modal matrices through three ResNets:

$$\begin{aligned} O_{TM} =& ResNet(Z_{TM}) \\ O_{NM} =& ResNet(Z_{NM}) \\ O_{NT} =& ResNet(O_{NT}) \end{aligned} \tag{4.2}$$

The unique network can be further extended to analyze these outputs for market movement prediction but it is still not as effective as utilizing the information captured with both unique and common sub-networks, and the results of the ablation studies showing the comparison are presented in the experiments section.

4.3.2 Common Network

We propose a common network to obtain the intra-modal and inter-modal information from the inputs. The common network includes three key elements: LSTM layers, feature tensors, and ResNet blocks. First, the inputs are separately processed using LSTMs to capture modality specific features.

$$\begin{aligned} L_M &= LSTM(Z_M) \\ L_T &= LSTM(Z_T) \\ L_N &= LSTM(Z_N) \end{aligned} \tag{4.3}$$

where L_M is the output of LSTM layer using market input features Z_M, L_T is the output of LSTM layer using the technical indicators, and L_N is the output of LSTM layer using news embeddings Z_N.

The latent features obtained from with the three LSTMs are then brought together via an outer product to form a multi-mode tensor. These tensors represent the inter-modal and intra-modal information within the multiple modalities.

$$\begin{aligned} T_{TM} &= \sum_j L_{T_{(ijk)}} \times L^T_{M_{(ijk)}} \\ T_{TMN} &= \sum_k T_{TM_{(ijk)}} \times L_{N_{(ikl)}} \end{aligned} \tag{4.4}$$

Lastly, the joint information from the tensor is retrieved through using a ResNet. In this common network, the ResNet takes T_{TMN} as its input and produces the output O_{COMMON}:

$$O_{COMMON} = ResNet(T_{TMN}) \tag{4.5}$$

4.3.3 Fusion Layer

The feature vectors from the common and unique sub-networks are integrated through vector concatenation to form O_{merged}.

$$O_{merged} = O_{NT} \oplus O_{NM} \oplus O_{TM} \oplus O_{COMMON} \tag{4.6}$$

We first flatten this feature vector via global average pooling operation and then process it with two fully connected layers using ReLU as the activation function. At the last step another fully connected layer is used to make a prediction. The overall network is a binary classification model that predicts the movement direction of the stock prices and the weights are optimized by minimizing the binary cross entropy loss given below:

$$L = -(y \log(\hat{y}) + (1 - y) \log(1 - \hat{y})) \tag{4.7}$$

where y denotes the target class of the movement direction and $\hat{y}$ is the prediction obtained from our network. The direction of the movement is defined as the difference between Close prices on day t+1 and day t. The labels are classified into two classes, where Class 1 indicates an upward movement and Class 0 indicates a downward movement in the closing prices.

$$Direction = \begin{cases} 1, & Close_{t+1} \geq Close_t \\ 0, & Close_{t+1} < Close_t \end{cases} \tag{4.8}$$

4.4 MStoCast-Attention

Similar to MStoCast, the goal is to capture the inter-modal and intra-modal information from the input modalities. We initially take financial news and historical market data as our data sources, and then compute seven technical indicators from the market data. So, our input modalities are historical market data, technical indicators and financial news headlines. The inputs are first processed separately with modality-specific encoders. Then the proposed

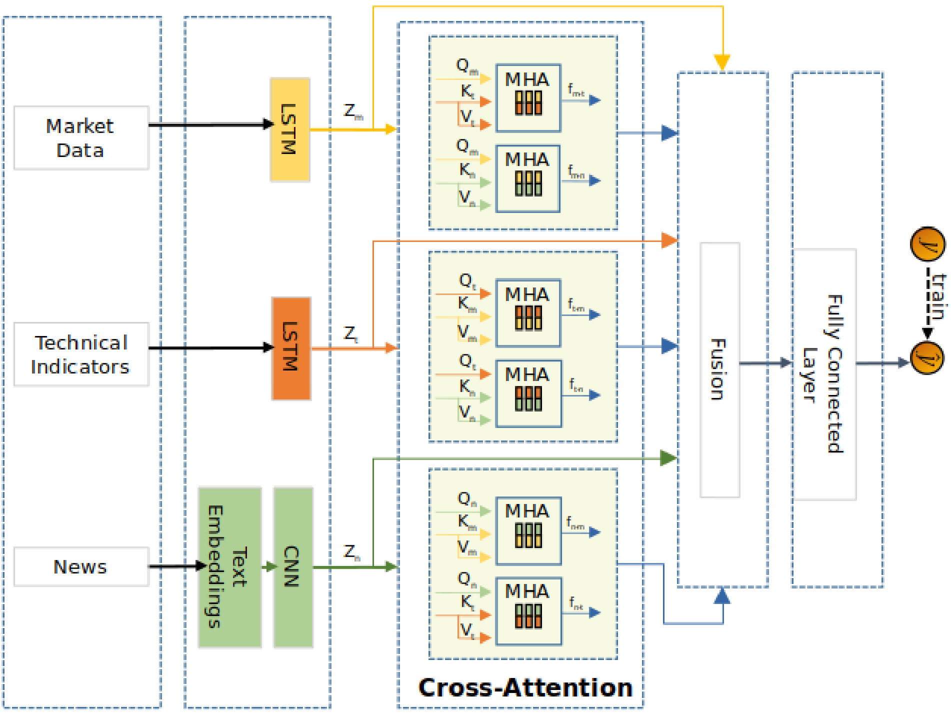

Figure 4.2: The framework of our MStoCast-Attention model

cross-attention module analyzes these encoded features and temporal average layer is applied to each resulting feature. The outputs are then amalgamated in the fusion layer to form a feature vector.

The latent features from the cross-attention and representation layers are then concatenated in the fusion layer to form a feature vector where it is utilized for market movement prediction. The design of MStoCast-Attention is presented in Figure 4.2.

The input modalities are first processed using three separate networks to capture the intra-modal information. We construct two separate LSTMs to process the market data and technical indicators. For the news modality, we first utilize textual embeddings to encode the textual data and then employ a convolutional layer to process these embeddings. The latent features from these networks are then fed into the cross-attention network for further processing.

In the proposed cross-attention module, the goal is to capture cross-modal information by applying attention network. We construct six multi-head attention networks where each consists of the features from the source and target modalities.

4.4.1 Representation Layer

The representation layer takes the input modalities and processes through three separate networks. To process the financial time series data, i.e. technical indicators and historical market data, we utilize two separate LSTM networks. On the other side, we employ text embeddings to encode the textual data and then feed these encoded representations into a Convolutional layer. The goal of these networks is to capture the modality-specific information from the inputs. The outputs of these networks, Z_m, Z_t and Z_n for market data, technical indicators and news modalities, are further analyzed in the cross-attention module. These features are also fed directly into fusion layer to be form a feature vector alongside the outputs of the cross-attention module.

4.4.2 Cross-attention Module

The primary objective of our cross-attention module is to capture the joint influence of the data modalities by modeling the cross-modal relationships among them. We model both modality-specific information and inter-modal interactions by constructing three cross-attention modules, where each module consists of two separate multi-head attention networks.

The attention mechanism focuses on the most crucial features in the input modalities by using higher weights. The attention mechanism can be summarized as mapping a query (Q) and a set of key-value (K, V) pairs to an output where the query and key define the weight matrix of the values.

We take the latent features from the representation layer and feed them into our cross-attention module. The data representations of every two modalities are processed together through these networks using attention function where we employ the scaled dot product to get the weight matrix. We first establish the input pairs for each network from these representations where we take Key and Value from one modality, and Query from another modality, such as market-news, news-technical indicators and so on. With three separate input modalities, we get the following six pairs: market-news, market-technical indicators, technical indicators-market, technical indicators-news, news-market, news-technical indicators. These pairs are then processed separately where we use softmax function to calculate the attention values for market-news (Q_m, K_n, V_n) and news-market (Q_n, K_m, V_m) pairs. These values are then used to calculate the values for each head which are then concatenated together to get the multi-head attention. Our cross-attention module consists of six multi-head attention networks and using the same calculations we can also get all the outputs (f_{m-t}, f_{m-n}, f_{t-m}, f_{t-n}, f_{n-m}, f_{n-t}). The outputs are then passed into the fusion layer to be processed alongside the modality-specific features from the input layer.

4.4.3 Fusion Layer

We take the outputs of the cross-attention module and the latent features from the representation layer and concatenate them together. Similar to MStoCast fusion layer, the feature vector is formed as below:

$$F_{merged} = f_{m-n} \oplus f_{m-t} \oplus f_{t-m} \oplus f_{t-n} \oplus f_{n-m} \oplus f_{n-t} \oplus Z_m \oplus Z_t \oplus Z_n \tag{4.9}$$

The next steps of MStoCast-Attention is the same as MStoCast, which was explained in detail in Section 4.3.3.

4.5 MStoCast-Sentiment

MStoCast-Sentiment aims to model the price movements in the stock markets by utilizing historical market data and the effects of financial news by analyzing their sentiment scores. Here, the goal is to capture the inter-modal and intra-modal information from the input modalities. We first take financial news and historical market data as our data sources, and then compute seven technical indicators from the market data. The inputs are first processed separately with modality-specific encoders. Then the proposed cross-attention module analyzes these encoded features and temporal average layer is applied to each resulting feature. The outputs are then merged in the fusion layer to create a feature vector which is later utilized for price movement forecast.

The design of MStoCast-Sentiment is presented in Figure 4.3. The numerical input modalities - market data and technical indicators - are first processed with two separate networks to capture the intra-modal information. We construct two separate LSTMs to

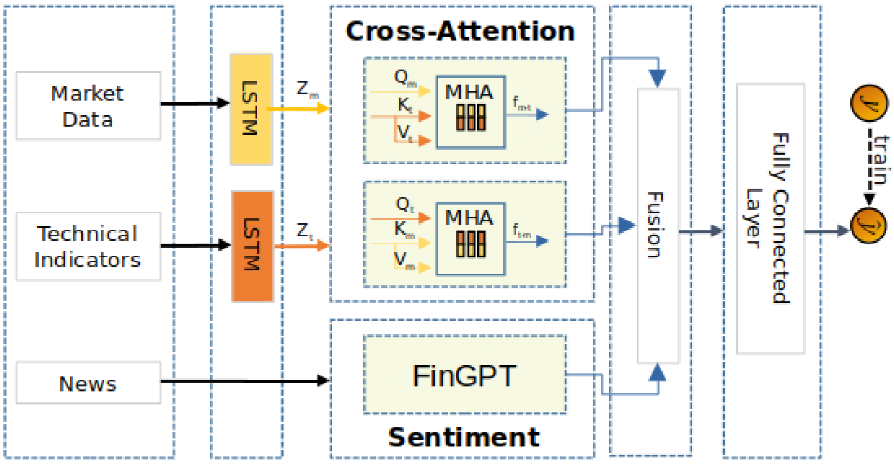

Figure 4.3: The framework of our MStoCast-Sentiment model

process the market data and technical indicators.

In the proposed cross-attention module, the goal is to capture cross-modal information by applying attention network. We construct six multi-head attention networks where each consists of the features from the source and target modalities. The news data goes through a sentiment module where an LLM is employed to extract sentiments from the news headlines.

4.5.1 Cross-attention Module

Our cross-attention module aims to extract cross-modal information from technical indicators and market data modalities. Following a similar design to MStoCast-Attention, we utilize Attention mechanism and construct two multi-head attention networks to process the modality-specific features from the previous layer.

We first take the outputs of the LSTM networks and prepare two sets of inputs to the attention modules. The inputs are formed by taking Query from one modality and Key and Value pairs from another modality: market-technical indicators (Q_m, K_t, V_t) and technical indicators-market (Q_t, K_m, V_m). The structure of the attention modules are similar to MStoCast-Attention modules and these inputs are processed using the same approach.

4.5.2 Sentiment Module

We construct a sentiment module to analyze the effect of the news sentiments on the market movements. The module takes cleaned news headlines in textual format and uses a Large

Language Model (LLM) to extract sentiments from the input data. We employ Transformer library from HugginFace [1] to utilize FinGPT LLM for sentiment analysis.

Large Language Models (LLMs) such as GPT-3, GPT-4 have made major breakthrough in NLP tasks [19]. They utilize transformer-based approach and have shown to be great performers in various generative tasks. BloombergGPT [241] which was trained using a financial and general data sources, can be considered as the first example in the financial sector. Unlike BloomberGPT, FinGPT takes a data-centric approach where it utilizes already existing LLMs and finetunes them to adapt to financial domain. The data sources include financial news, social media, corporate filings, trends and academic datasets.

In our sentiment module, we employ FinGPT with Llama2 [2] model, which is a model released by Meta. We use transformer library from HuggingFace platform which enables calls to the model through their API. We use "fingpt-sentiment-llama2-13b-lora" [3] model and make calls using Transformer library to extract sentiments for our financial news headlines. These sentiment values are then fed into the fusion layer for further analysis.

Following the guideline from FinGPT [4], we use the following prompt format:

```
prompt = [
'''Instruction: What is the sentiment of this news?
Please choose an answer from {negative/neutral/positive}
Input: <financial news headline>.
Answer: ''']
```

Following the same design from MStoCast given in Section 4.3.3, the outputs of the Cross-attention and Sentiment modules are later amalgamated together to form a feature vector. This feature vector is later processes using a feed-forward layer to predict the price movements.

[1] https://www.huggingface.co
[2] https://llama.meta.com/llama2/
[3] https://huggingface.co/FinGPT/fingpt-sentiment_llama2-13b_lora
[4] https://huggingface.co/FinGPT/fingpt-sentiment_llama2-13b_lora

4.6 Experimental Settings

In our experiments, we utilize real-world datasets comprising financial news, market data, and technical indicators spanning from January 1, 2013, to December 31, 2019, encompassing 1761 trading days. The financial news was sourced from Reuters[5], where each article included a title, body, and publishing date. The date served to synchronize the articles with the daily market data. Following preprocessing, including the removal of duplicates and unrelated articles, the dataset consisted of 527,047 news headlines. The minimum number of articles per day was 38, while the maximum was 998.

We employed the headlines from the financial news, as research has indicated that utilizing news titles can yield superior prediction results compared to using the entire article body [206]. Our approach involved utilizing 5-day windows as input, meaning that data from five consecutive days were used at each step to predict the movement of the closing price for the subsequent day. As the number of news titles per trading day varied, we amalgamated all the titles for a given day into a single coherent sentence. Subsequently, we harnessed BERT to encode the textual data into feature vectors, ultimately generating a singular sentence embedding vector per trading day.

Market data pertaining to the S&P 500 index and individual stock data for five companies within the index were extracted for the corresponding dates from Yahoo Finance[6]. These market data were represented using five attributes: Open, High, Low, Close prices, and Volume. The data were normalized to fall within the [0, 1] range.

In addition, we computed seven technical indicators for each trading day, derived from the prices over the preceding five days. These indicators encompassed Stochastic %K, Stochastic %D, Momentum, Rate of Change, William's %R, A/D Oscillator, and Disparity 5. These particular indicators have demonstrated effectiveness in market prediction [97]. Refer to Table 4.1 for the list of selected technical indicators and their descriptions.

Our experiments were centered on predicting the directional movement of the S&P index price and the individual stock prices of five companies (AAPL, MSFT, AMZN, TSLA, GOOGL). For index prediction, we initially employed an 80-20% split for training and testing. Additionally, we evaluated yearly performance by utilizing the first 10 months of each year

[5] https://www.reuters.com/business/finance/
[6] https://finance.yahoo.com/

Table 4.1: Technical indicators and their descriptions

Indicator	Description
Stochastic %K	The %K is the percentage of the difference between its highest and lowest values over a certain time period.
Stochastic %D	Moving average of Stochastic %K.
Momentum	The change in a security's price over a given time period.
Rate of Change	The percentage difference between the current price and the price n days ago.
William's %R	A momentum indicator that measures overbought/oversold levels.
A/D Oscillator	A momentum indicator that associates changes in price.
Disparity 5	The distance of current price and the moving average of 5 days.

for training and the last 2 months for testing. In the case of individual stock prediction, the 80-20% split was again used for training and testing purposes.

In this study, the standard measure of accuracy (Acc) and Matthews Correlation Coefficient (MCC) are selected to evaluate the performance of the models for S&P500 index and individual stock prediction [46; 119; 122; 265; 203]. MCC is generally employed when the sizes of classes y = 1 and y = 0 differ:

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \tag{4.10}$$

where TP, TN, FP, and FN are the number of true positives, true negatives, false positives, and false negatives, respectively.

For our LSTM networks, we employ 64 LSTM layers to analyze the sentence embeddings from the news titles, while the technical indicator and market data inputs are processed using 32 LSTM layers.

In the MStoCast model, the design of the ResNet is consistent across both the unique and shared networks. In the common network, we utilize 64 filters and set the kernel size to 3 in the convolutional layers. For the unique network, we employ convolutional layers with 32 filters and a kernel size of 2. In both sub-networks, a stride of 1 is applied, and we perform padding convolution.

The initial two fully connected layers in the fusion layer contain 32 output neurons each and employ the ReLU activation function to process the inputs. The Adam optimizer is utilized to optimize the network parameters, and we set the epoch size to 500 epochs with a batch size of 64.

4.6.1 Baseline Methods

We compare our approach with the following baselines on predicting individual stocks and S&P500 index.

- **Recurrent Convolutional Neural Network (RCNN)** [226]: a CNN and LSTM based market forecast model that utilizes technical indicators and financial news.

- **Event Embeddings-RCN (EB-RCN)** [168]: a similar model to RCNN that uses market data alongside with event embeddings [46] from news data.

- **Bidirectional Gated Recurrent Unit (BGRU)** [89]: a market forecast model with financial news and historical market data.

- **LSTM-based Recurrent State Transition (ANRES)** [140]: a market movement prediction model utulizing events from news data.

- **Hybrid Attention Network (HAN)** [84]:a state-of-the-art stock market forecast model with hierarchical attention utilizing news data.

- **Adversarial Attentive LSTM (Adv-LSTM)** [53]: a market forecast model utilizing attentive LSTMS with adversarial training strategy.

These machine learning models are built for stock market prediction. But in order to test the effectiveness of MStoCast as a multimodal learning system, we compare it against the following two state of the art multimodal learning models that have been successful in other domains. Both of these models have been proposed for sentiment analysis but we adopt their architectures and customize them for stock market prediction.

- **Tensor Fusion Network (TFN)** [257] utilizes the bi-modal and unimodal information from the input modalities.

- **Early Fusion LSTM (EF-LSTM)** [259] concatenates the inputs from different modalities and employs a single LSTM to analyse the combined input.

We also perform ablation studies by creating two models using our common (MStoCast-Common) and unique (MStoCast-Unique) sub-networks separately and evaluate their performances.

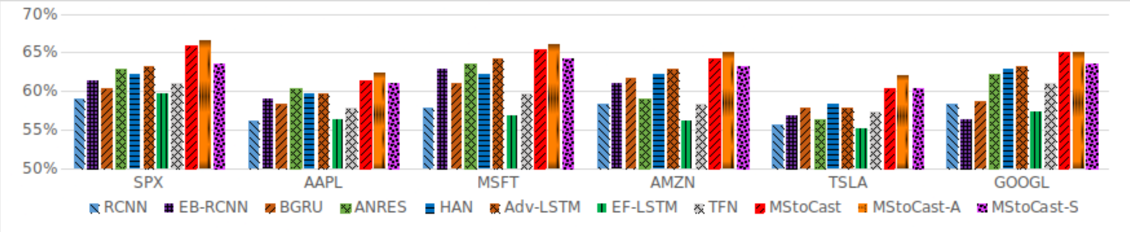

Figure 4.4: Accuracy on index and individual stock prediction (the higher, the better).

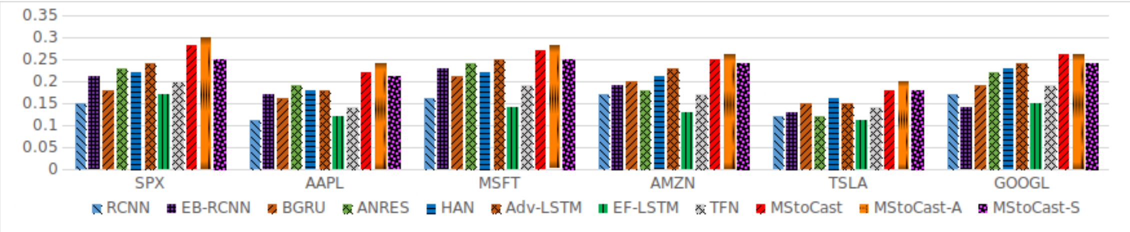

Figure 4.5: MCC results on index and individual stock prediction (the higher, the better).

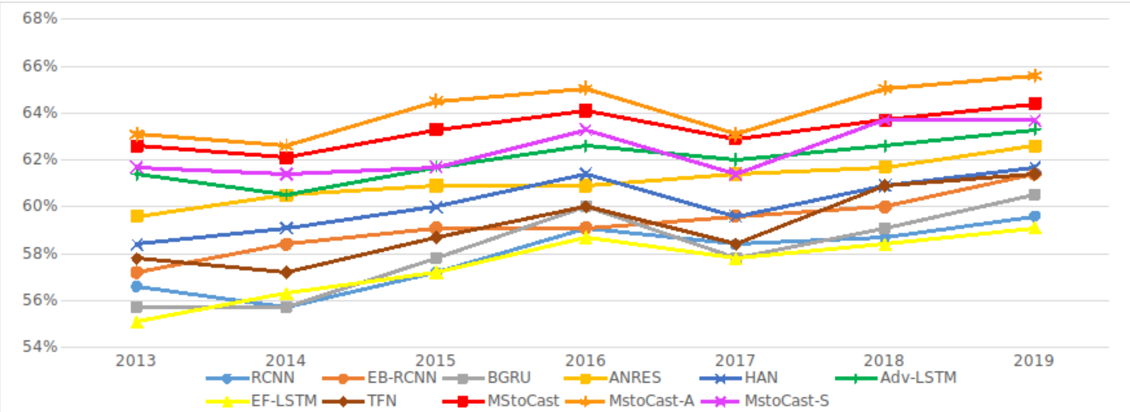

Figure 4.6: Yearly accuracy results on S&P500 index prediction (the higher, the better).

4.7 Results and Analysis

The primary contributions of this research are given in the three innovative multimodal models, which leverage both common and unique information from various data modalities to predict price movements in stock markets. To demonstrate the efficacy of the proposed models, we conducted experiments using real-world datasets. Initially, we performed ablation tests to ascertain the significance of the novel multimodal design. Then, we compared the performance of MStoCast against several state-of-the-art models from stock market prediction domain literature. The results of the experiments are given in Figure 4.4 and Figure 4.5.

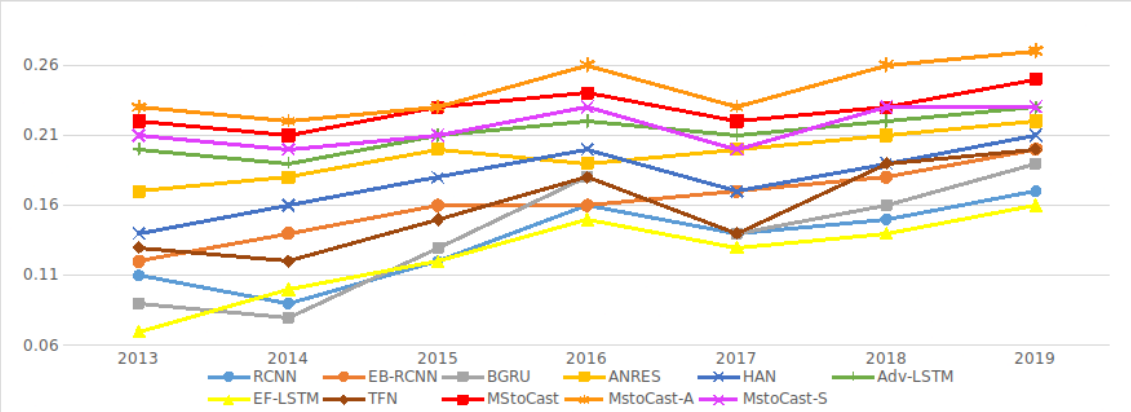

Figure 4.7: Yearly MCC results on S&P500 index prediction (the higher, the better).

4.7.1 Ablation Study

To assess the effectiveness of utilizing common and unique information, we conduct ablation studies using MStoCast's unique (MStoCast-Unique) and common (MStoCast-Common) sub-networks. We run these sub-networks as separate market forecast models and compare the results against the complete MStoCast architecture. We focused on predicting the movement direction of both the S&P index and five individual stocks. The outcomes of these experiments are presented in Figure 4.8 and Figure 4.9.

A notable observation is that MStoCast consistently outperforms both the unique and common sub-networks in both stock index prediction and individual stock trend prediction. These results underscore the effectiveness of our proposed design, which capitalizes on inter-modal and intra-modal information as common knowledge, while exploiting bi-modal relationships among input modalities as unique information, leading to improved prediction accuracy.

Among our findings, the common sub-network achieved superior results in both metrics compared to the unique sub-network. This emphasizes that while modeling bi-modal relationships among modalities through early fusion enhances model performance, the primary focus should remain on capturing the inter-modal and intra-modal dynamics inherent in the data modalities.

4.7.2 Comparison with Baselines

We conducted our experiments under two distinct data split configurations. Initially, we employed an 80-20% split and performed tests to predict the movement direction of the

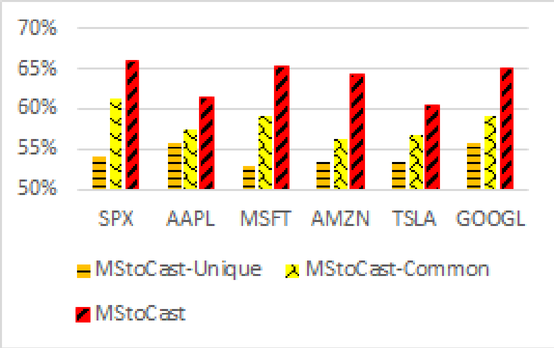

Figure 4.8: Ablation accuracy results.

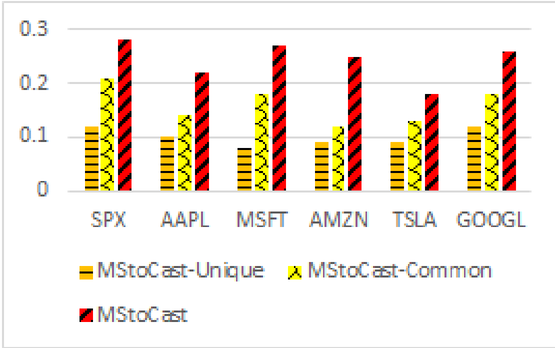

Figure 4.9: Ablation MCC results.

S&P 500 index price and the price of five individual stocks. We evaluated the results using two key metrics: accuracy and MCC. Our findings, as presented in Figure 4.4 and 4.5, clearly indicate that our proposed MStoCast, MStoCast-Attention and MStoCast-Sentiment models consistently outperforms baseline stock market prediction models across both metrics. Although Furthermore, the results demonstrate our models' superiority over two state-of-the-art multimodal deep learning models, TFN and EF-LSTM. The consistent improvement in performance against baseline models underscores the efficacy of the proposed designs for both index and individual stock prediction, leveraging multimodal data.

Among our proposed models, MStoCast-Attention outperforms other two models. This is also firstly due to the effectiveness of the proposed design, we utilize modality-specific features using separate LSTM and CNN networks, as well as cross-modal information using multi-head attention modules that analyzes the input modalities in pair. The results also show the effectivenes of Attention mechanism in capturing the most important information from the inputs.

On the other hand, MStoCast-Sentiment performs worse than the other two models in both metrics. This can be attributed to the loss of intra-modal information after the cross-attention module and failing to model the relationship between the news modality and the others. But in the experiments, MStoCast-Sentiment model also outperforms the baseline models in both metrics.

Among the baseline models, attention-based HAN and Adv-LSTM stand out, demonstrating better performance in most tickers in terms of both accuracy and MCC. These findings highlight the significance of the attention mechanism in focusing on crucial data aspects when capturing latent features. However, even against these models, MStoCast exhibits substantial enhancements in both accuracy and MCC metrics. We attribute this to the innovative multimodal learning design that models intra-modal and inter-modal relationships within the input data.

We expanded the comparative analysis to include two successful multimodal deep learning models from other domains: TFN and EF-LSTM. MStoCast outperforms these models significantly in predicting both index and individual stock price movements. While TFN outperforms EF-LSTM across all tickers, both models are often outperformed by other baseline methods.

In our experiments, models like EB-RCN and ANRES aimed to extract event-based information from financial news and integrate it into prediction models. The results highlighted that this approach yields superior performance compared to employing basic word embeddings for textual data encoding. This underscores the crucial role of effective textual representation techniques in multimodal stock market prediction.

Another observation is that MStoCast-Common consistently outperforms the RCNN model in most tickers, although it is often surpassed by EB-RCN. This suggests that while tensors and sentence embeddings contribute to prediction accuracy, employing more sophisticated textual representation methods, such as event embeddings, can yield even better results.

Among the baseline models, EF-LSTM and MStoCast-Unique networks employed early information fusion, where raw input features were concatenated at the input stage prior to analysis. Conversely, other baseline methods adopted a late fusion technique, wherein models independently analyzed input modalities before fusing data at a later stage. The outcomes indicated that late fusion techniques produce better results by capturing more crucial latent features from input modalities.

We also evaluated the models' yearly prediction performance for S&P 500 index prediction, using the first 10 months of each year for training and the last two months for testing. The results, illustrated in Figure 4.6 and 4.7, showcase the proposed models' consistent superiority over baseline models for each year, in both directional accuracy and MCC. The same trend from the above tests also continues here, where Adv-LSTM performs the best among the baselines. Although the yearly results are slightly lower than the initial test results, this can be attributed to the smaller test sample size inherent in the yearly setup.

4.7.3 Experiments with Text Representation Techniques

We assess the performances of GloVe and BERT embeddings for the representation of the news headlines and compare the results. We use the same textual data and test the performance of our MStoCast and MStoCast-Attention models, and the baseline methods for the prediction of S&P500 index price movement. The results of these experiments are presented in Figure 4.10.

It can be observed from the results that BERT embeddings outperform GloVe for all the methods except RCNN. This shows that BERT sentence embeddings are able to represent the textual data better, and they also work better than GloVe embeddings in our proposed model. So we choose BERT sentence embeddings as the main textual representation technique. However, it's worth noting that our models with GloVe embeddings still outperformed baseline models utilizing both GloVe and BERT embeddings. This underscores that while the choice of textual representation technique can positively influence model performance and BERT sentence embeddings contribute to better results, the primary factor driving improved outcomes is the proposed multimodal design.

4.7.4 Trading Simulation

We run a trading simulation to demonstrate the performance of the models in making profits. We employ the strategy proposed by **?**] to build a real stock trading simulation. In the strategy, if the model predicts that the price will go up the following day, then the trader will buy $10,000 worth of that stock at the opening price. The stock will be hold for one day, in which if the trader can make a profit of 2% or more, then the stock will be sold immediately. Otherwise, the stock will be sold at the closing price. We use the same strategy if the model indicates a negative trend in the price movement.

We use the same dataset for index and individual trading simulation. Using the aforementioned strategy, in index trading, we averaged $4,945 profit using MStoCast which is higher than the profit we made using the baseline models.

On individual stock level, we run the simulation to trade five stocks in the market. Trading with MStoCast makes more profit for all five stocks than trading with the baseline models. The results on index and individual stock levels are given in Table 4.2.

Table 4.2: Trading simulation results for index and individual stocks using the test dataset.

Model	SPX	AAPL	MSFT	AMZN	TSLA	GOOGL
RCNN	$2250	$1259	$2123	$632	$2338	$2475
EB-RCNN	$3310	$2317	$2494	$1427	$2778	$3522
BGRU	$2775	$1827	$2921	$1910	$2801	$3870
ANRES	$3540	$2920	$3095	$932	$2466	$4155
MStoCast	$4945	$3319	$3273	$2110	$3264	$4431

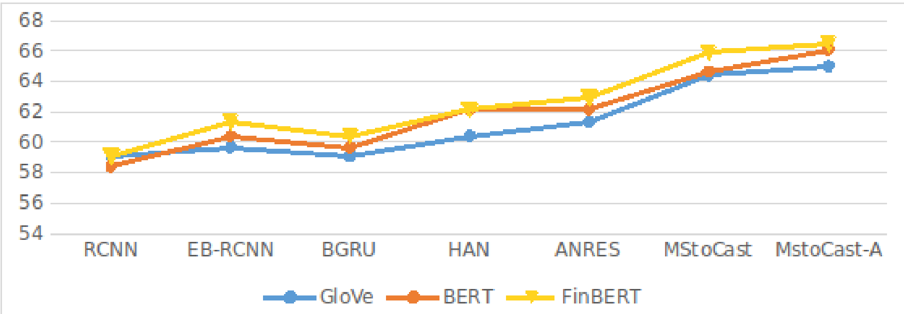

Figure 4.10: Accuracy comparison of three embedding techniques (the higher, the better).

4.8 Conclusion and Future Work

In this chapter, we proposed three novel multimodal stock market prediction model named MStoCast, MStoCast-Attention and MStoCast-Sentiment. These models effectively utilize stock market data, financial news data, and technical indicator data to predict the price movements. The models focus on capturing modality-specific and joint information, as well as the cross-modal and bi-modal features among the inputs modalities. In the experiments with real-world datasets, our proposed models considerably outperform the baseline models. The results show the strong potential of the proposed model architectures in multimodal learning. In future, we plan to explore and include more modalities of data, such as social media data and macroeconomic indicators, to further improve the stock market prediction performance.

Chapter 5

MCASP: multimodal Cross Attention Network for Stock Market Prediction

Stock market prediction is considered a complex task due to the non-stationary and volatile nature of the stock markets. With the increasing amount of online data, various information sources have been analyzed to understand the underlying patterns of the price movements. However, most existing works in the literature mostly focus on either the intra-modality information within each input data type, or the inter-modal relationships among the input modalities. Different from these, in this chapter, we propose a novel multimodal Cross Attention Network for Stock Market Prediction (MCASP) by capturing both modality-specific features and the joint influence of each modality in a unified framework. We utilize financial news, historical market data and technical indicators to predict the movement direction of the market prices. After processing the input modalities with three separate deep networks, we first construct a self-attention network that utilizes multiple Transformer models to capture the intra-modal information. Then we design a novel cross-attention network that processes the inputs in pairs to exploit the cross-modal and joint information of the modalities. Experiments with real world datasets for S&P500 index forecast and the prediction of five individual stocks, demonstrate the effectiveness of the proposed multimodal design over several state-of-the-art baseline models.

5.1 Introduction

Stock market movements are inherently affected by a multitude of data sources, encompassing historical price data, technical indicators [225], financial news [191], social media [28], and official announcements [54]. It has been established that analyzing these multiple data modalities together enables the capture of underlying patterns in stock movements, rendering stock market prediction a multimodal learning task [3]. The efficacy of employing effective multimodal representation and learning techniques to uncover the joint influence of these data modalities is pivotal for model performance [121]. Simultaneously, it is important to extract the intra-modal information within each data source. Early information fusion techniques combine raw input features initially and then construct a prediction model, which aids in capturing the combined influence of modalities but neglects intra-modal information. Late fusion techniques, conversely, analyze input features separately and subsequently employ a fusion layer for prediction. While this approach facilitates a focus on modality-specific features, it may overlook inter-modal information. Balancing the capture of intra-modal and inter-modal information from input modalities is essential.

Researchers have identified that pairs of data modalities, such as financial news and market prices, as well as market prices and technical indicators [225], both impact price movements. However, existing models, while striving to capture the joint influence of all modalities together, may overlook the underlying bi-modal relationships between various data inputs. Therefore, in addition to capturing their collective influence, it is also crucial to understand the bi-modal relationships among pairs of input modalities.

To address these challenges, various methods have been developed, primarily categorized as inter-modality and intra-modality-based techniques. Inter-modality methods aim to capture the underlying relationships among input modalities but may miss the connections within each modality. Conversely, intra-modality techniques focus on uncovering modality-specific relations but tend to disregard the inter-modal connections across input modalities. Combining modality-specific features with inter-modal connections can synergize and enhance overall analysis. Hence, exploring a unified framework capable of capturing both inter-modality and intra-modality relations within the input data is imperative.

Motivated by these challenges, we present a novel multimodal Cross-Attention Network for Stock Market Prediction (MCASP). MCASP forecasts the direction of price movements by jointly modeling inter-modality and intra-modality relationships within the input data (i.e., financial news, market data, and technical indicators) within a unified deep learning framework. To achieve this, we construct two distinct attention networks: a self-attention network and a cross-attention network, designed to capture intra-modal and inter-modal relationships, respectively. Different from our previous works, here we utilize the attention mechanism to effectively capture both cross-modal information and modality-specific information from the input modalities in the form of cross-attention and self-attention modules. Having these information helps us to better predict the market movements which we show in our experiments using a real-world dataset.

The self-attention module focuses on extracting modality-specific features from the input modalities. We first employ two separate Long Short-Term Memory (LSTM) networks to extract latent features from market data and technical indicators. Simultaneously, we leverage FinBERT [141] to encode textual data (i.e., financial news). Within the self-attention network, the LSTM network outputs are processed by two Transformer [228] units, while the encoded textual data undergo analysis via a Convolutional Neural Network (CNN).

The cross-attention module involves creating three pairs by concatenating representations of news and market data, news and technical indicators, and market data and technical indicators. These pairs are then fed into three separate Transformer units. The outputs from the self-attention and cross-attention modules converge in the Fusion Layer to generate a combined feature vector. Finally, we employ a fully connected layer to predict the direction of price movements.

5.2 Related Work

In this section, we review related work in stock market prediction, multimodal machine learning and the attention mechanism.

5.2.1 Stock Market Prediction

Financial news, market data, social media data, and official company announcements have been widely used for market analysis research. It has been shown by Shi et al. [202] that using only news titles is better than using the whole article text. Schumaker et al. [195] proposed the Arizona Financial Text (AZFinText) system, focusing on sentiment analysis using proper nouns. In another study, Vargas et al. [225] represented news headlines using Word2Vec word embeddings and constructed a multimodal prediction model using Convolutional Neural Networks (CNN) and Long Short-term Memory (LSTM) networks. Meanwhile, Huynh et al. [88] designed a prediction model using the Bidirectional Gated Recurrent Unit (BGRU) architecture, extracting news headlines and representing them using word embedding vectors.

The paper by Nuij et al. [163] used ViewerPro to extract events from news articles and incorporated them with technical indicators. Matsubara et al. [146] employed paragraph vectors for news data representation, and Ding et al. [45] introduced a CNN-based event embeddings model where the authors constructed a neural tensor network to learn event embeddings from financial news data.

5.2.2 Multimodal Machine Learning

Multimodal learning architectures have been widely utilized in various fields including robotics [109], healthcare [62], multimedia [133], and sentiment analysis [258]. A multimodal paper by Barnum et al. [11] applies early fusion in the multimodal representation of audio and visual inputs and another research [51] employs structured image and textual to construct multimodal concept taxonomies. Researchers have also utilized various RNN structures for multimodal representations for different kinds of applications such as human behaviour analysis [182] and time-series data analysis [133; 209].

One popular technique for combined utilization of multimodal data is early fusion [152; 177]. Early fusion concatenates low-level features from individual modalities to be utilized with any learning framework for downstream machine learning tasks. Moreover, early fusion performs poorly when feature fusion among non-interacting modalities (such as voice and fingerprint) is performed. These limitations are slightly addressed in Zadeh et al. [256], where shared embeddings (latent space) among individual modalities are learned. These shared representations outperform the early fusion but require careful parameter tuning.

There also exists a stream of work that perform outer-product-based neural frameworks for multimodal data fusion. In Lin et al. [135] a bilinear-CNN is proposed to obtain bi-modal interactions among features obtained from two heterogeneous CNNs. This is accomplished by taking a neural-based bilinear product of high-level features. The bilinear layer required parameter estimation of a quadratic number of neurons and hence prone to over-fitting. This limitation is alleviated in Fukui et al. [60]; Hu et al. [81] which introduced an alternate formulation of the bilinear layer and obtains its compact representation by utilizing sophisticated neural-based factorization schemes.

5.2.3 Attention

The attention mechanism has found success in a wide range of domains, including natural language processing (NLP) [7; 228], image captioning [252], image classification [244], visual question answering [143], and more [189; 116]. Notably, the Transformer model [228] introduced the self-attention mechanism, which can capture long dependencies in sequential data [35]. This allows the model to effectively explore the intra-modal relationships, such as the relationships between words in machine translation. Cross-attention mechanism is used in a neural net to capture the interconnectedness of multiple data modalities or sequences [26]. Different from self-attention, cross-attention mechanism is able to model cross-modal information by creating mappings between the source and target modalities [221].

Taking inspiration from the Transformer model [228], the self-attention mechanism has been applied in various works, extending its utility to visual question answering [255], video analysis [236], and image-text matching [243].

In recent years, attention mechanisms have also made their way into multimodal learning problems. While architectures like BERT [43] were originally designed for NLP tasks, they have been adapted for multimodal challenges as well [31; 144]. For instance, some approaches, like the dual attention network in Nam et al. [154], focus on learning inter-modal relationships between visual regions and textual elements within sentences. Others, like the co-attention framework in Lu et al. [143], tackle tasks like visual question answering by jointly learning image and question attentions. Additionally, in Paulus et al. [174], a combination of inter-modal and intra-modal attentions is leveraged within deep reinforcement learning for text summarization.

The existing works in the literature try to address the multimodal representation of the various input modalities. Some of the works attempt to capture the joint information by employing early [227] or late fusion techniques [5],**21.** and some others mainly focus on

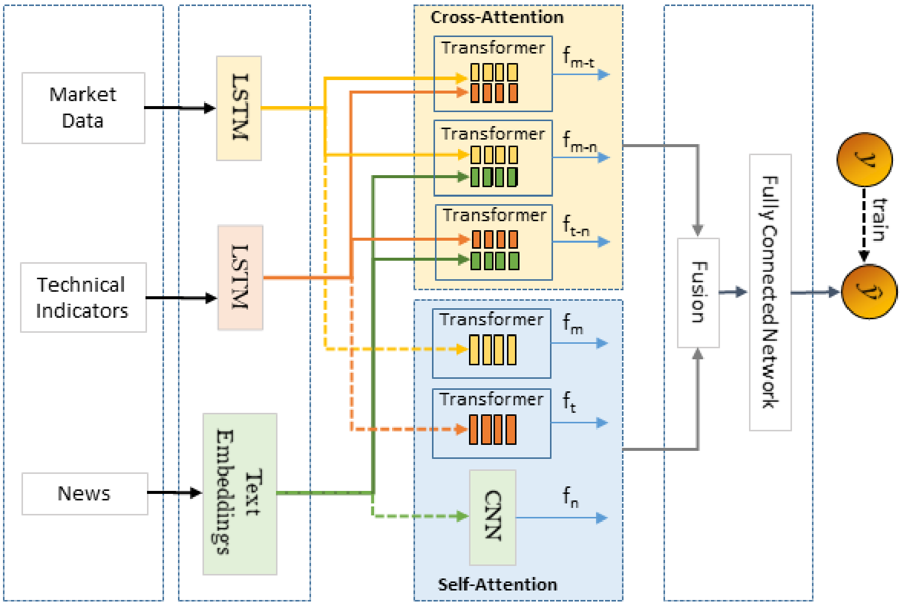

Figure 5.1: Demonstration of the MCASP architecture design. Solid lines represent the inputs to the Cross-Attention network while the dotted lines are the inputs to the Self-Attention network.

modality specific information from each data source. Although these works show some good results, they mostly fail to utilize the potential of using multiple data modalities together. When using variuous input modalities, it is crucial to capture the joint information of these modalities as well as the modality-specific features. Also, when using three or more modalities, it is beneficial to explore the bi-modal relationships among the modality pairs (e.g. the relationship between market data and technical indicators and market data and financial news should also be analyzed separately). To address these issues, we propose MCASP, a unified deep learning model that utilizes intra-modal and inter-modal information for market prediction.

5.3 Model Design

In this section, we provide a detailed description of the architecture of the proposed MCASP model. The design of our MCASP model is demonstrated in Figure5.1.

5.3.1 Input Representation

We start by using historical market data and financial news as our primary data sources. From the market data, we derive a set of seven technical indicators. We employ three distinct data modalities for stock market prediction: market data, technical indicators, and financial news. To process these modalities, we employ three separate deep networks.

We construct two LSTM networks to handle the market data and technical indicator modalities, respectively. Additionally, we utilize text embeddings to encode the news data. For this purpose, we leverage BERT and FinBERT embeddings.

The latent features obtained from the LSTM networks and the sentence embeddings from FinBERT are then fed into the self-attention and cross-attention modules to capture both intra-modal and inter-modal relationships.

5.3.2 Self-Attention Module

The primary objective of the attention process is to discern the relationship between two states and focus on the most crucial features. This is achieved by assigning higher weights to the most pertinent elements within the input vectors. The attention layer consists of three key components: the query, keys, and values, with these elements being identical in the self-attention context. The attention mechanism can be conceptualized as mapping a query and a set of key-value pairs to an output, where the output is a weighted sum of the values. The weight matrix, determining the weight assigned to each value, is defined using the query and the key. Several options for the attention function are available, including the dot product, multi-layer perceptron, and scaled dot product.

The self-attention network is used to capture intra-modality relations, employing two separate Transformer units [125] for market data and technical indicators, along with a CNN for financial news data. In the Transformer model, we employ the scaled dot product to compute the weight matrix. This module encompasses both multi-head self-attention and position-wise feed-forward layers, as depicted in Figure 5.2. The term 'multi-head attention' implies that attention is computed multiple times. The attention calculation is as follows:

$$A(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V \tag{5.1}$$

Where the d_k represents the dimension of the queries and the keys. In the Transformer module, multiple parallel attention values are computed where each output is called a head.

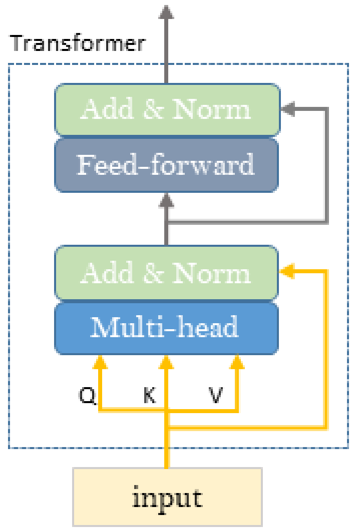

Figure 5.2: Design of the Transformer model

The i^th head is calculated as:

$$head_i = A(QW_i^Q, KW_i^K, VW_i^V) \tag{5.2}$$

We then concatenate these heads to obtain the multi-head attention.

$$MT(Q, K, V) = Concat(head_1, ., head_h)W^0 \tag{5.3}$$

In our self-attention module, the two Transformers for market data and technical indicators modalities, we get the following two outputs:

$$\begin{aligned} f_m &= MT(Q_m, K_m, V_m) \\ f_t &= MT(Q_t, K_t, V_t) \end{aligned} \tag{5.4}$$

For the textual data modality, we utilize the outputs of the BERT embeddings. The BERT model incorporates multiple Transformers and is proficient at capturing intra-modality information. Subsequently, we employ a CNN to extract local latent features denoted as f_n.

These three outputs from our self-attention module, namely f_m, f_t, and f_n, are later employed to predict the movement of closing prices.

5.3.3 Cross-Attention Module

We introduce a novel cross-attention to model both intra-modality information and the interconnectedness of the modalities, achieved by implementing three separate Transformer units. By modeling both intra-modality and inter-modality relationships, we aim to capture the joint effect of the input modalities while retaining modality-specific features.. Our aim is to capture the interactions across the input modalities by applying the cross-attention function to the outputs of the input representation layer. Initially, we establish three distinct pairs from the modalities to implement the attention mechanism: from market data to technical indicators ($m - t$), from market data to financial news ($m - n$), and from technical indicators to financial news ($t - n$). Market data and the derived technical indicators have a significant influence on market movements, which justifies prioritizing these pairings with higher weights.

The calculation of these three cross-attention values is as follows:

$$
\begin{aligned}
A_{m-n}(Q_m, K_n, V_n) &= softmax(\frac{Q_m K_n^T}{\sqrt{d_k}})V_n \\
A_{m-t}(Q_m, K_t, V_t) &= softmax(\frac{Q_m K_t^T}{\sqrt{d_k}})V_t \\
A_{t-n}(Q_t, K_n, V_n) &= softmax(\frac{Q_t K_n^T}{\sqrt{d_k}})V_n
\end{aligned}
\tag{5.5}
$$

Here, A_{m-n}, A_{m-t}, and A_{t-n} represent the cross-attention between market data and news, market data and technical indicators, and technical indicators and news modalities, respectively. Furthermore, Q_m and Q_t denote the query vectors for the market data and technical indicators modalities, while K_t and K_n represent the key vectors, and V_t and V_n denote the value vectors for the technical indicators and news modalities, respectively.

With these cross-attention terms in place, we proceed to compute the attention values for each head as follows:

$$
\begin{aligned}
head^i_{m-n} &= A_{m-n}(Q_m W_i^{Q_m}, K_n W_i^{K_n}, V_n W_i^{V_n}) \\
head^i_{m-t} &= A_{m-t}(Q_m W_i^{Q_m}, K_t W_i^{K_t}, V_t W_i^{V_t}) \\
head^i_{t-n} &= A_{t-n}(Q_t W_i^{Q_t}, K_n W_i^{K_n}, V_n W_i^{V_n})
\end{aligned}
\tag{5.6}
$$

These terms represent each head in each cross-attention pair. Subsequently, we combine these head values for each pair to obtain the multi-head attention for each cross-attention block:

$$\begin{aligned} MT_{m-n} &= Concat(head^1_{(m-n)}, .., head^h_{(m-n)})W^0_{m-n} \\ MT_{m-t} &= Concat(head^1_{(m-t)}, .., head^h_{(m-t)})W^0_{m-t} \\ MT_{t-n} &= Concat(head^1_{(t-n)}, .., head^h_{(t-n)})W^0_{t-n} \end{aligned} \tag{5.7}$$

Putting all these together, our cross-attention module produces the following three outputs:

$$\begin{aligned} f_{m-n} &= MT_{m-n} \\ f_{m-t} &= MT_{m-t} \\ f_{t-n} &= MT_{t-n} \end{aligned} \tag{5.8}$$

5.3.4 Fusion Layer

In the fusion layer, we amalgamate the feature vectors from the self-attention and cross-attention modules to form a combined feature vector.

$$f_{merged} = [f_m, f_t, f_n, f_{m-n}, f_{m-t}, f_{mcsp-t}] \tag{5.9}$$

We then employ a fully connected layer with ReLU as the activation function to process the feature vector f_{merged}. In the final step, another fully connected layer is employed to make predictions. The overall network is a binary classification model used for predicting the movement direction of stock closing prices, and the model weights are optimized by minimizing the binary cross-entropy loss:

$$L = -(y \log(\hat{y}) + (1 - y) \log(1 - \hat{y})) \tag{5.10}$$

where y represents the target class for the movement direction, and $\hat{y}$ signifies the prediction obtained from MCASP. The movement direction is defined as the difference between the closing prices on day $t + 1$ and day t. The labels are categorized into two classes: Class 1 indicating an upward movement and Class 0 indicating a downward movement in the closing prices.

5.4 Experimental Settings

In our experiments, we utilized real-world datasets encompassing financial news, market data, and technical indicators spanning from January 1, 2010, to December 31, 2019, encompassing a 10-year period. The financial news was sourced from Reuters[1], with each article containing a title, body, and publication date. The publication date was employed to align the articles with the daily market data. We specifically focused on the headlines from the financial news, as research has demonstrated that using news titles can yield superior prediction results compared to using the entire article body [205]. The number of news titles per trading day varied; hence, we aggregated all the titles for a given day into a single extended sentence and employed FinBERT to encode the textual data into feature vectors. Consequently, we obtained a single sentence embedding vector for each trading day.

We utilize historical market data for S&P index and individual stocks from Yahoo Finance[2] for the corresponding dates. These five companies included Google, Tesla, Amazon, Apple, and Microsoft and the data includes Open, High, Low, Close prices, and Volume. We normalize the market data to be within the range of [0, 1].

We initially employ an 80-20% split for training and testing for index price prediction as this technique keeps the temporal characteristic of the dataset and lets us to quickly assess different models' performances. In 80-20 split technique, the first portion (80%) of the dataset is used for training and the rest of the dataset is utilized for model evaluation.

We also evaluate the yearly performances of the models by utilizing the first 10 months of each year for training and the last 2 months for testing. We utilize the 80-20% split again for training and testing purposes for individual stock prediction.

Based on the literature [96], we computed seven technical indicators for each trading day using the market data over the preceding five days.

We employ accuracy (Acc) and Matthews Correlation Coefficent (MCC) to evaluate the performance of different models. MCC is generalyy employed when the sizes of classes y = 1 and y = 0 differ.

[1] https://www.reuters.com/business/finance/
[2] https://finance.yahoo.com/

5.4.1 Baseline Methods

We compare our approach with the following baselines on predicting individual stocks and S&P500 index:

Recurrent Convolutional Neural Network (RCNN) [225] is a CNN and RNN based stcok forecast model that utilizes technical indicators and financial news.

Event Embeddings (EB-RCN) [167] is another LSTM and CNN based model that also includes market data and employ event embeddings from [45].

Bidirectional Gated Recurrent Unit (BGRU) [88] uses both online financial news and historical price data to predict the stock movements.

LSTM-based Recurrent State Transition (ANRES) [139] uses only news events for market movement prediction.

Hybrid Attention Network (HAN) [83] is a state-of-the-art stock trend prediction model with hierarchical attention that utilizes news data.

multimodality Attention Network (MMAN) [74]

Attention-Based Recurrent Neural Network (At-LSTM) [136]

Adversarial Attentive LSTM (Adv-LSTM) [52] is a market prediction model using historical market data, where the authors employ attentive LSTMs and utilize adversarial training strategy.

Other than these methods, we also perform ablation studies by constructing different variants of the proposed MCASP model.

5.5 Results and Analysis

In order to test the effectiveness of our model, we run experiments using real-world dataset including financial news data, historical market data and technical indicators.

5.5.1 Main Results

We use our dataset to conduct tests for forecasting of the price movements of S&P500 index and five individual stocks. The accuracy results are illustrated in Figure 5.3, showing that MCASP improves upon the baseline models. The MCC results, presented in Figure 5.4,

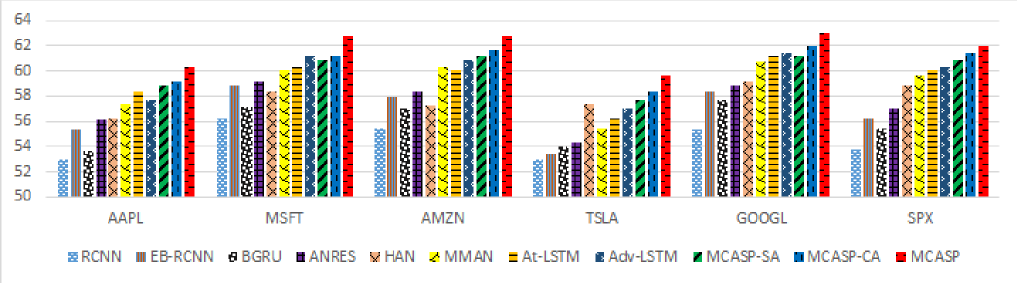

Figure 5.3: Accuracy results on index and individual stock prediction (the higher, the better).

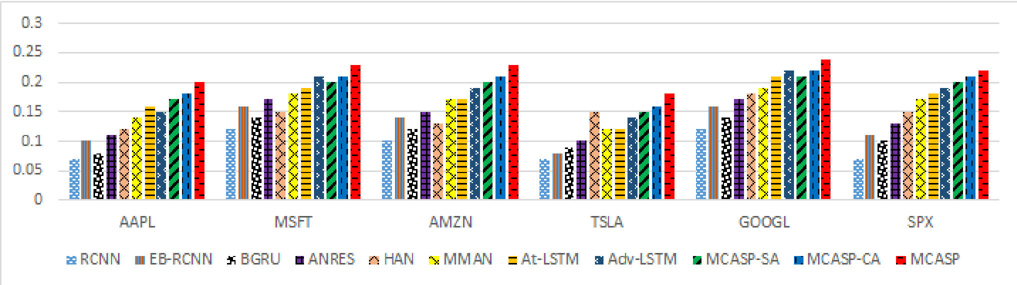

Figure 5.4: MCC results on index and individual stock prediction (the higher, the better).

echo the same trend, with MCASP exhibiting superior prediction performance for the price movement directions of all five stocks and S&P index compared to the baseline models.

Overall, in our experiments, MCASP consistently achieves the best results in terms of both accuracy and MCC. When compared to the baselines, MCASP demonstrates improvements in prediction performance for both index and individual stock predictions, underscoring the effectiveness of the proposed multimodal attention design in leveraging intra-modal and inter-modal information from multiple input sources.

Among the baseline models, attention-based prediction models perform better than other baselines in both accuracy and MCC. These results underscore the significance of the attention module in capturing critical latent features from the input data. However, MCASP surpasses the attention-based baseline models, suggesting that its enhanced performance stems not only from the use of the self-attention module but also from its ability to extract inter-modal relationships among input modalities through the novel cross-attention module.

We also asses the models' yearly prediction performances for S&P 500 index prediction, where we use the first 10 months of each year for training and the last two months for testing. The accuracy results, given in Figure 5.5, demonstrate that MCASP consistently outperforms all the baseline models for each year. Although the yearly results are slightly

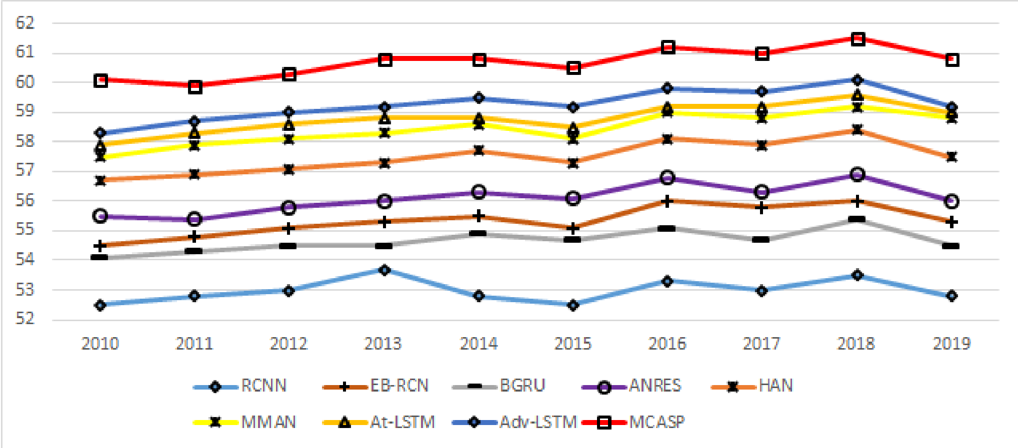

Figure 5.5: Yearly ACC results on S&P index prediction (the higher, the better).

lower than the initial test results, this can be attributed to the smaller test sample size inherent in the yearly setup.

Collectively, the experiments involving S&P500 index prediction and the prediction of price movements for five individual stocks demonstrate that the MCASP model is adept at learning meaningful representations from multiple input modalities, capitalizing on the self-attention network and the innovative cross-attention module.

5.5.2 Ablation Study

To assess the impact of different components of the MCASP model, we conducted an ablation study using the same real-world dataset. Initially, we evaluated the effectiveness of our two attention modules independently by creating two distinct models. Subsequently, we explored three text embedding techniques to demonstrate the influence of the textual representation method on the overall performance.

Self-attention and cross-attention modules. This experimental study elucidates the individual performance of each module and underscores the significance of capturing both intra-model and inter-model information, in contrast to the prevalent approach of focusing solely on either modality-specific or joint influence of input modalities, as seen in most existing works. To this end, we developed two distinct models - MCASP-SA (MCASP with the self-attention module only) and MCASP-CA (MCASP with the cross-attention module only) - and subjected them to testing using our original dataset.

In our experiments, MCASP consistently outperforms both MCASP-SA (which exclusively employs the self-attention module) and MCASP-CA (which relies solely on the cross-attention module) across both accuracy and MCC metrics. This substantiates the effectiveness of our proposed design in addressing multimodal problems.

Notably, MCASP-CA yields superior results compared to MCASP-SA. We postulate that this is attributed to the cross-attention module's design, which initially extracts modality-specific features and subsequently captures inter-modal relationships among modalities using the attention mechanism.

Moreover, when compared to the baseline models, both MCASP-SA and MCASP-CA consistently demonstrate improved accuracy and MCC results in the majority of the tests. This underscores the success of the proposed sequential design for both modules. The results further affirm that leveraging multiple modalities (i.e., financial news, historical market data, and technical indicators) can enhance model performance.

MCASP with various text embeddings. We subsequently examined the impact of various textual embeddings (Transformer-based BERT and GloVe) on the overall model performance. We employed three distinct textual embedding methods to encode and represent the financial news data, namely GloVe word embeddings, Transformer-based BERT embeddings, and FinBERT embeddings. Our experimental results underscore the significance of selecting an appropriate text embedding method when utilizing financial news data.

The results, presented in Table5.1 show that Transformer-based BERT and FinBERT embeddings consistently outperformed GloVe embeddings across both accuracy and MCC metrics for S&P index prediction. Furthermore, FinBERT showed improved results compared to BERT embeddings, underscoring the value of domain-specific knowledge in textual data representation.

Table 5.1: The impact of different text embedding methods.

Embedding Method	Accuracy	MCC
GloVe	60.91%	0.208
BERT	61.60%	0.215
FinBERT	62.03%	0.228

Notably, predictions using FinBERT as our text embedding method exhibited improvement compared to GloVe and BERT embeddings. This highlights the utility of domain knowledge in comprehending and representing textual data. However, even without domain knowledge and when employing RNN-based GloVe embeddings and general BERT embeddings, MCASP consistently outperformed all baseline methods across both metrics for S&P500 index prediction. These results affirm that while a robust textual representation technique can enhance model performance, the primary factor contributing to improved results lies in the novel multimodal design, which incorporates both self-attention and cross-attention modules to capture latent features from the input modalities.

5.6 Conclusion

In this chapter, we proposed a novel multimodal cross attention network for stock market prediction that models the intra-modal and inter-modal information from the input modalities in a unified framework. We first analyze the input modalities via three separate deep networks to extract the salient features. We then process these features with the proposed self-attention and cross-attention modules to jointly model the intra-modal and inter-modal information. We analyze financial news, historical market data and technical indicators to predict the movement direction of S&P500 index prices and the prices of five individual stocks. We test the effectiveness of the proposed multimodal design using real-world dataset from Reuters and Yahoo! Finance and compare its performance against multiple state-of-the-art baseline models. Experimental results show that our model achieves improved performance in stock market prediction.

Chapter 6

Conclusion and Future Work

Stock markets play a fundamental role in modern day economy and it is important to understand the underlying patterns in order to successfully predict the price movements. These movements are influenced by multiple factors, including historical market data, social media data, financial news and ad-hoc announcements. It is important to explore the relationships between these various data sources and how they affect the price movements in the markets. But this multimodality also brings some additional challenges which need to be addressed for successful applications. One of the main challenges is the fusion of various data sources. Effective multimodal fusion techniques are essential in successful utilization of these data sources for market prediction.

Multimodal representation is another area that poses some questions and hugely affects the performance of a learning model. In multimodal learning, it is crucial to capture both modality-specific information and the joint influence of the input data modalities and utilize them together. It is also important to extract the most important latent features and cross-modal information for successful models. When utilizing three or more modalities, bi-modal relationships can also help to understand the underlying patterns of the market movements. So, modelling them alongside the common influence of the input data can improve the learning model's performance.

In this thesis, we pose questions based on these challenges and present our goals for research contributions. This research starts by presenting a comprehensive review of the recent developments in the stock market analysis space. We review the literature based on the input data, representation techniques and analysis models. We then present a multimodal market analysis model that proposes a novel architecture to effectively implement multimodal fusion of multiple data modalities for a regression task. In the experiments with real world dataset, our novel model outperforms the baselines which shows the effectiveness of the proposed design in multimodal fusion for stock market prediction.

We then investigate various deep learning algorithms for multimodal representation. We experiment with the latest vector embeddings and build three different multimodal representation models using recent deep-learning algorithms. We utilize LLMs for sentiment analysis and employ ResNets and Attention mechanisms for multimodal representation. The experiments in stock market prediction domain show the importance of capturing both inter-modal and intra-modal information. We present the comparison results with the baselines, which show that all three models outperform both stock market prediction and multimodal learning models.

Lastly, we employ Transformers to build multimodal deep learning model to utilize the Attention mechanism in capturing cross-modal and intra-modal information from the data sources. We propose cross-attention and self-attention modules to effectively exploit multimodal web data for classification-based market prediction problems. The results show the effectiveness of the proposed design for multimodal learning.

6.1 Future Work

In our research, we explored the existing literature in the stock market analysis domain and identified the main gaps in the are. Based on these gaps we posed some research questions and proposed novel multimodal learning models to address these questions. But there are also some areas in the stock market analysis domain that need further research. Especially, with the success of LLMs (large language models) in NLP and other domains, they could also be utilized for market prediction. Although we also employed LLMs for sentiment analysis, it could also be interesting to analyze them as end-to-end market prediction models. BloombergGPT [175] and FinGPT [250] showed some initial works in the domain. BloombergGPT utilized its vast amount of financial data to train a model for financial purposes. Although it showed some success in early days, with its cost and implementation complexity showed some of the disadvantages of training a finance-focused LLM from scratch.

On the other hand, FinGPT took a different approach and focused on fine-tuning existing successful LLMs with financial data. Their utilization in financial sentiment analysis have shown good results. [260] employ LLMs to build a sentiment analysis model where they use instruction fine-tuning. In the experiments, the proposed model outperforms state of the art machine learning and deep learning models, as well as other widely used LLMS. This shows the effectiveness of LLMs in financial sentiment analysis and overall market analysis applications. [41] use an LLM to genereate financial sentiment labels for Reddit posts and feeds that data into a small model in production.

Another aspect is exploring more data sources and more data in general. The applications of deep learning in other domains show that, in most use cases, learning models can perform better with more data. Although these are interesting questions that can be explored further, they were not in the scope of this research.

References

[1] Y. Abu-Mostafa and A. F. Atiya. Introduction to financial forecasting. *Applied Intelligence*, 6:205–213, 2004.

[2] Abien Fred Agarap. Deep learning using rectified linear units (relu). *ArXiv*, abs/1803.08375, 2018.

[3] R. Akita, A. Yoshihara, T. Matsubara, and K. Uehara. Deep learning for stock prediction using numerical and textual information. *2016 IEEE/ACIS 15th International Conference on Computer and Information Science (ICIS)*, pages 1–6, 2016.

[4] R. Akita, A. Yoshihara, T. Matsubara, and K. Uehara. Deep learning for stock prediction using numerical and textual information. *2016 IEEE/ACIS 15th International Conference on Computer and Information Science (ICIS)*, pages 1–6, 2016.

[5] R. Akita, A. Yoshihara, T. Matsubara, and K. Uehara. Deep learning for stock prediction using numerical and textual information. *2016 IEEE/ACIS 15th International Conference on Computer and Information Science (ICIS)*, pages 1–6, 2016.

[6] S. Baccianella, A. Esuli, and F. Sebastiani. Sentiwordnet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. In *LREC*, 2010.

[7] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *CoRR*, abs/1409.0473, 2014.

[8] M. Baker and J. Wurgler. Investor sentiment and the cross-section of stock returns. 2003.

[9] R. Balakrishnan, Xinying Qiu, and P. Srinivasan. On the predictive ability of narrative disclosures in annual reports. *Eur. J. Oper. Res.*, 202:789–801, 2010.

[10] M. Ballings, D. Poel, N. Hespeels, and R. Gryp. Evaluating multiple classifiers for stock price direction prediction. *Expert Syst. Appl.*, 42:7046–7056, 2015.

[11] George M. Barnum, Sabera Talukder, and Yisong Yue. On the benefits of early fusion in multimodal representation learning. *ArXiv*, abs/2011.07191, 2020.

[12] S. Basu, Xinjie Ma, and Hoa Tran. Measuring investment opportunities using financial statement text. 2019.

[13] Y. Bengio, P. Simard, and P. Frasconi. Learning long-term dependencies with gradient descent is difficult. *IEEE transactions on neural networks*, 5 2:157–66, 1994.

[14] Y. Bengio, P. Lamblin, D. Popovici, and H. Larochelle. Greedy layer-wise training of deep networks. In *NIPS*, 2006.

[15] Y. Bengio, A. C. Courville, and P. Vincent. Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35:1798–1828, 2013.

[16] J. Bollen, H. Mao, and X. Zeng. Twitter mood predicts the stock market. *ArXiv*, abs/1010.3003, 2011.

[17] L. Breiman. Bagging predictors. *Machine Learning*, 24:123–140, 1996.

[18] L. Breiman. Random forests. *Machine Learning*, 45:5–32, 2004.

[19] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeff Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. *ArXiv*, abs/2005.14165, 2020. URL https://api.semanticscholar.org/CorpusID:218971783.

[20] Oscar Bustos and Alexandra Pomares Quimbaya. Stock market movement forecast: A systematic review. *Expert Syst. Appl.*, 156:113464, 2020. URL https://api.semanticscholar.org/CorpusID:219097702.

[21] E. Cambria, Fu J, F. Bisio, and S. Poria. Affectivespace 2: Enabling affective intuition for concept-level sentiment analysis. In *AAAI*, 2015.

[22] E. Cambria, S. Poria, D. Hazarika, and K. Kwok. Senticnet 5: Discovering conceptual primitives for sentiment analysis by means of context embeddings. In *AAAI*, 2018.

[23] L Cao and F Tay. Support vector machine with adaptive parameters in financial time series forecasting. *IEEE transactions on neural networks*, 14 6:1506–18, 2003.

[24] L. Chai, H. Xu, Z. Luo, and S. Li. A multi-source heterogeneous data analytic method for future price fluctuation prediction. *Neurocomputing*, 418:11–20, 2020.

[25] S. W. Chan and J. Franklin. A text-based decision support system for financial sequence prediction. *Decis. Support Syst.*, 52:189–198, 2011.

[26] Chun-Fu Chen, Quanfu Fan, and Rameswar Panda. Crossvit: Cross-attention multi-scale vision transformer for image classification. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 347–356, 2021. URL https://api.semanticscholar.org/CorpusID:232404237.

[27] Shun Chen and Lei Ge. Exploring the attention mechanism in lstm-based hong kong stock price movement prediction. *Quantitative Finance*, 19:1507 – 1515, 2019. URL https://api.semanticscholar.org/CorpusID:199364412.

[28] W. Chen, C. Yeo, C. Lau, and B. S. Lee. Leveraging social media news to predict stock index movement using rnn-boost. *Data Knowl. Eng.*, 118:14–24, 2018.

[29] W. Chen, C. Yeo, C. Lau, and B. S. Lee. Leveraging social media news to predict stock index movement using rnn-boost. *Data Knowl. Eng.*, 118:14–24, 2018.

[30] W. Chen, C. Yeo, C. Lau, and B. S. Lee. Leveraging social media news to predict stock index movement using rnn-boost. *Data Knowl. Eng.*, 118:14–24, 2018.

[31] Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. Uniter: Learning universal image-text representations. *ArXiv*, abs/1909.11740, 2019.

[32] Dawei Cheng, Fangzhou Yang, Sheng Xiang, and Jin Liu. Financial time series forecasting with multi-modality graph neural network. *Pattern Recognit.*, 121:108218, 2022.

[33] M. D. Choudhury, H. Sundaram, A. John, and D. Seligmann. Can blog communication dynamics be correlated with stock market activity? In *HT '08*, 2008.

[34] Francesco Colasanto, Luca Grilli, Domenico Santoro, and Giovanni Villani. Albertino for stock price prediction: a gibbs sampling approach. *Information Sciences*, 2022.

[35] Zihang Dai, Zhilin Yang, Yiming Yang, Jaime G. Carbonell, Quoc V. Le, and Ruslan Salakhutdinov. Transformer-xl: Attentive language models beyond a fixed-length context. In *Annual Meeting of the Association for Computational Linguistics*, 2019. URL https://api.semanticscholar.org/CorpusID:57759363.

[36] Divyanshu Daiya and Che Lin. Stock movement prediction and portfolio management via multimodal learning with transformer. *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3305–3309, 2021. URL https://api.semanticscholar.org/CorpusID:235771535.

[37] E. J. de Fortuny, T. D. Smedt, D. Martens, and W. Daelemans. Evaluating and understanding text-based stock price prediction models. *Inf. Process. Manag.*, 50: 426–441, 2014.

[38] J. B. de Long, A. Shleifer, L. Summers, and R. Waldmann. Noise trader risk in financial markets. *Journal of Political Economy*, 98:703 – 738, 1990.

[39] Arthur Emanuel de Oliveira Carosia, Guilherme Palermo Coelho, and Ana Estela Antunes da Silva. Investment strategies applied to the brazilian stock market: A methodology based on sentiment analysis with deep learning. *Expert Syst. Appl.*, 184:115470, 2021.

[40] S. Deng, Takashi Mitsubuchi, K. Shioda, T. Shimada, and A. Sakurai. Combining technical analysis with sentiment analysis for stock price prediction. *2011 IEEE Ninth International Conference on Dependable, Autonomic and Secure Computing*, pages 800–807, 2011.

[41] Xiang Deng, Vasilisa Bashlovkina, Feng Han, Simon Baumgartner, and Michael Bendersky. What do llms know about financial markets? a case study on reddit market sentiment analysis. *Companion Proceedings of the ACM Web Conference 2023*, 2022. URL https://api.semanticscholar.org/CorpusID:254974468.

[42] Y. Deng, F. Bao, Y. Kong, Z. Ren, and Q. Dai. Deep direct reinforcement learning for financial signal representation and trading. *IEEE Transactions on Neural Networks and Learning Systems*, 28:653–664, 2017.

[43] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *ArXiv*, abs/1810.04805, 2019.

[44] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *ArXiv*, abs/1810.04805, 2019.

[45] X. Ding, Y. Zhang, T. Liu, and J. Duan. Deep learning for event-driven stock prediction. In *Proceedings of the 24th International Joint Conference on Artificial Intelligence (IJCAI)*, 2015.

[46] X. Ding, Y. Zhang, T. Liu, and J. Duan. Deep learning for event-driven stock prediction. In *Proceedings of the 24th International Joint Conference on Artificial Intelligence (IJCAI)*, 2015.

[47] X. Ding, Y. Zhang, T. Liu, and J. Duan. Deep learning for event-driven stock prediction. In *IJCAI*, 2015.

[48] Hang Dong, Jie Ren, Balaji Padmanabhan, and Jeffrey V. Nickerson. How are social and mass media different in relation to the stock market? a study on topic coverage and predictive value. *Information & Management*, 59(2):103588, 2022. ISSN 0378-7206. doi: https://doi.org/10.1016/j.im.2021.103588.

[49] E. F. Fama. Random walks in stock market prices. *Financial Analysis Journal*, 51: 75–80, 1995.

[50] Eugene F. Fama and Kenneth R. French. Common risk factors in the returns on stocks and bonds. *Journal of Financial Economics*, 33:3–56, 1993.

[51] Marco Federici, Anjan Dutta, Patrick Forr'e, Nate Kushman, and Zeynep Akata. Learning robust representations via multi-view information bottleneck. *ArXiv*, abs/2002.07017, 2020.

[52] Fuli Feng, Huimin Chen, Xiangnan He, Ji Ding, Maosong Sun, and Tat-Seng Chua. Enhancing stock movement prediction with adversarial training. In *International Joint Conference on Artificial Intelligence*, 2018.

[53] Fuli Feng, Huimin Chen, Xiangnan He, Ji Ding, Maosong Sun, and Tat-Seng Chua. Enhancing stock movement prediction with adversarial training. In *International Joint Conference on Artificial Intelligence*, 2018.

[54] S. Feuerriegel and J. Gordon. Long-term stock index forecasting based on text mining of regulatory disclosures. *Decis. Support Syst.*, 112:88–97, 2018.

[55] S. Feuerriegel and J. Gordon. Long-term stock index forecasting based on text mining of regulatory disclosures. *Decis. Support Syst.*, 112:88–97, 2018.

[56] S. Feuerriegel and J. Gordon. Long-term stock index forecasting based on text mining of regulatory disclosures. *Decis. Support Syst.*, 112:88–97, 2018.

[57] T. Fischer and C. Krauß. Deep learning with long short-term memory networks for financial market predictions. *Eur. J. Oper. Res.*, 270:654–669, 2018.

[58] Thomas G. Fischer and Christopher Krauss. Deep learning with long short-term memory networks for financial market predictions. *Eur. J. Oper. Res.*, 270:654–669, 2017. URL https://api.semanticscholar.org/CorpusID:207640363.

[59] P. H. Franses and H. Ghijsels. Additive outliers, garch and forecasting volatility. *International Journal of Forecasting*, 15:1–9, 1999.

[60] Akira Fukui, Dong Huk Park, Daylen Yang, Anna Rohrbach, Trevor Darrell, and Marcus Rohrbach. Multimodal compact bilinear pooling for visual question answering and visual grounding. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 457–468, 2016.

[61] S. Furao, J. Chao, and J. Zhao. Forecasting exchange rate using deep belief networks and conjugate gradient method. *Neurocomputing*, 167:243–253, 2015.

[62] Muhammad Ghulam, Fatima Alshehri, Fakhri Karray, Abdulmotaleb El Saddik, Mansour Alsulaiman, and Tiago H. Falk. A comprehensive survey on multimodal medical signals fusion for smart healthcare systems. *Inf. Fusion*, 76:355–375, 2021.

[63] G. Gidófalvi. Using news articles to predict stock price movements. 2001.

[64] I. Goodfellow, Y. Bengio, and Aaron C. Courville. Deep learning. *Nature*, 521:436–444, 2015.

[65] S. S. Groth and J. Muntermann. An intraday market risk management approach based on textual analysis. *Decis. Support Syst.*, 50:680–691, 2011.

[66] M. U. Gudelek, S. A. Boluk, and A. Ozbayoglu. A deep learning based stock trading model with 2-d cnn trend detection. *2017 IEEE Symposium Series on Computational Intelligence (SSCI)*, pages 1–8, 2017.

[67] M. Gunasekaran and K. S. Ramaswami. A fusion model integrating anfis and artificial immune algorithm for forecasting indian stock market. *Journal of Applied Sciences*, 11:3028–3033, 2011.

[68] Hakan Gunduz, Y. Yaslan, and Z. Cataltepe. Intraday prediction of borsa istanbul using convolutional neural networks and feature correlations. *Knowl. Based Syst.*, 137:138–148, 2017.

[69] Z. Guo, H. Wang, Q. Liu, and J. Yang. A feature fusion based forecasting model for financial time series. *PLoS ONE*, 9, 2014.

[70] D. Gupta, Mahardhika Pratama, Zhenyuan Ma, J. Li, and M. Prasad. Financial time series forecasting using twin support vector regression. *PLoS ONE*, 14, 2019.

[71] M. Hagenau, M. Liebmann, and D. Neumann. Automated news reading: Stock price prediction based on financial news using context-capturing features. *Decis. Support Syst.*, 55:685–697, 2013.

[72] A. Haq, Adnan Zeb, Zhenfeng Lei, and De fu Zhang. Forecasting daily stock trend using multi-filter feature selection and deep learning. *Expert Syst. Appl.*, 168:114444, 2021.

[73] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016.

[74] Shwai He and Shihao Gu. Multi-modal attention network for stock movements prediction. *ArXiv*, abs/2112.13593, 2021.

[75] B. Henrique, V. A. Sobreiro, and H. Kimura. Literature review: Machine learning techniques applied to financial market prediction. *Expert Syst. Appl.*, 124:226–251, 2019.

[76] H. Herwartz. Stock return prediction under garch — an empirical assessment. *International Journal of Forecasting*, 33:569–580, 2017.

[77] G. E. Hinton and R. Salakhutdinov. Reducing the dimensionality of data with neural networks. *Science*, 313:504 – 507, 2006.

[78] G. E. Hinton, S. Osindero, and Y. Teh. A fast learning algorithm for deep belief nets. *Neural Computation*, 18:1527–1554, 2006.

[79] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural Computation*, 9: 1735–1780, 1997.

[80] Alexander Hogenboom, Alex Brojba-Micu, and Flavius Frasincar. The impact of word sense disambiguation on stock price prediction. *Expert Syst. Appl.*, 184:115568, 2021.

[81] Guosheng Hu, Yang Hua, Yang Yuan, Zhihong Zhang, Zheng Lu, Sankha S Mukherjee, Timothy M Hospedales, Neil Martin Robertson, and Yongxin Yang. Attribute-enhanced face recognition with neural tensor fusion networks. In *ICCV*, pages 3764–3773, 2017.

[82] Z. Hu, W. Liu, J. Bian, X. Liu, and T. Liu. Listening to chaotic whispers: A deep learning framework for news-oriented stock trend prediction. *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*, 2018.

[83] Ziniu Hu, Weiqing Liu, Jiang Bian, Xuanzhe Liu, and Tie-Yan Liu. Listening to chaotic whispers: A deep learning framework for news-oriented stock trend prediction. *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*, 2017.

[84] Ziniu Hu, Weiqing Liu, Jiang Bian, Xuanzhe Liu, and Tie-Yan Liu. Listening to chaotic whispers: A deep learning framework for news-oriented stock trend prediction. *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*, 2017.

[85] C. Huang and C. Y. Tsai. A hybrid sofm-svr with a filter-based feature selection for stock market forecasting. *Expert Syst. Appl.*, 36:1529–1539, 2009.

[86] W. Huang, Y. Nakamori, and S. Wang. Forecasting stock market movement direction with support vector machine. *Comput. Oper. Res.*, 32:2513–2522, 2005.

[87] H. D. Huynh, L. M. Dang, and D. Duong. A new model for stock price movements prediction using deep neural network. In *SoICT 2017*, 2017.

[88] Huy D. Huynh, L. Minh Dang, and Duc Duong. A new model for stock price movements prediction using deep neural network. *Proceedings of the Eighth International Symposium on Information and Communication Technology*, 2017.

[89] Huy D. Huynh, L. Minh Dang, and Duc Duong. A new model for stock price movements prediction using deep neural network. *Proceedings of the Eighth International Symposium on Information and Communication Technology*, 2017.

[90] Gang Ji, Jingmin Yu, Kai Hu, Jie Xie, and Xunsheng Ji. An adaptive feature selection schema using improved technical indicators for predicting stock price movements. *Expert Systems with Applications*, 2022.

[91] Nan Jing, Zhao Wu, and Hefei Wang. A hybrid model integrating deep learning with investor sentiment analysis for stock price prediction. *Expert Syst. Appl.*, 178:115019, 2021.

[92] Nal Kalchbrenner, Edward Grefenstette, and Phil Blunsom. A convolutional neural network for modelling sentences. *ArXiv*, abs/1404.2188, 2014.

[93] Yakup Kara, M. A. Boyacioglu, and Ö. K. Baykan. Predicting direction of stock price index movement using artificial neural networks and support vector machines: The sample of the istanbul stock exchange. *Expert Syst. Appl.*, 38:5311–5319, 2011.

[94] Stephen Kelly and K. Ahmad. Estimating the impact of domain-specific news sentiment on financial assets. *Knowl. Based Syst.*, 150:116–126, 2018.

[95] K. Kim. Financial time series forecasting using support vector machines. *Neurocomputing*, 55:307–319, 2003.

[96] Kyoungjae Kim. Financial time series forecasting using support vector machines. *Neurocomputing*, 55:307–319, 2003.

[97] Kyoungjae Kim. Financial time series forecasting using support vector machines. *Neurocomputing*, 55:307–319, 2003.

[98] Yoon Kim. Convolutional neural networks for sentence classification. In *EMNLP*, 2014.

[99] M. Kraus and S. Feuerriegel. Decision support from financial disclosures with deep neural networks and transfer learning. *Decis. Support Syst.*, 104:38–48, 2017.

[100] C. Krauß, X. A. Do, and N. Huck. Deep neural networks, gradient-boosted trees, random forests: Statistical arbitrage on the s&p 500. *Eur. J. Oper. Res.*, 259:689–702, 2017.

[101] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *CACM*, 2017.

[102] T. Kuremoto, S. Kimura, K. Kobayashi, and M. Obayashi. Time series forecasting using a deep belief network with restricted boltzmann machines. *Neurocomputing*, 137:47–56, 2014.

[103] L. A. Laboissiere, R. Fernandes, and G. G. Lage. Maximum and minimum stock price forecasting of brazilian power distribution companies based on artificial neural networks. *Appl. Soft Comput.*, 35:66–74, 2015.

[104] M. Lam. Neural network techniques for financial performance prediction: integrating fundamental and technical analysis. *Decis. Support Syst.*, 37:567–581, 2004.

[105] V. Lavrenko, M. D. Schmill, D. Lawrie, P. Ogilvie, D. D. Jensen, and J. Allan. Language models for financial news recommendation. In *CIKM '00*, 2000.

[106] B. LeBaron. Time series properties of an artificial stock market. *Economic Dynamics and Control*, 23(9–10):1487–1516, 1999.

[107] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proc. IEEE*, 86:2278–2324, 1998.

[108] C. Y. Lee and V. Soo. Predict stock price with financial news based on recurrent convolutional neural networks. *2017 Conference on Technologies and Applications of Artificial Intelligence (TAAI)*, pages 160–165, 2017.

[109] Michelle A. Lee, Yuke Zhu, Krishna Parasuram Srinivasan, Parth Shah, Silvio Savarese, Li Fei-Fei, Animesh Garg, and Jeannette Bohg. Making sense of vision and touch: Self-supervised learning of multimodal representations for contact-rich tasks. *2019 International Conference on Robotics and Automation (ICRA)*, pages 8943–8950, 2018.

[110] S. Lee and Seong Joon Yoo. Threshold-based portfolio: the role of the threshold and its applications. *The Journal of Supercomputing*, pages 1–18, 2018.

[111] B. Li, K. Chan, C. X. Ou, and R. Sun. Discovering public sentiment in social media for predicting stock movement of publicly listed companies. *Inf. Syst.*, 69:81–92, 2017.

[112] Chang Li, Dongjin Song, and Dacheng Tao. Multi-task recurrent neural networks and higher-order markov random fields for stock price movement prediction: Multi-task rnn and higer-order mrfs for stock price classification. *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2019. URL https://api.semanticscholar.org/CorpusID:198952502.

[113] F. Li. The information content of forward-looking statements in corporate filings—a naïve bayesian machine learning approach. *Journal of Accounting Research*, 48:1049–1102, 2010.

[114] F. Li. Textual analysis of corporate disclosures: A survey of the literature. 2011.

[115] J. Li, A. Sun, J. Han, and C. Li. A survey on deep learning for named entity recognition. *IEEE Transactions on Knowledge and Data Engineering*, pages 1–1, 2020. doi: 10.1109/TKDE.2020.2981314.

[116] Jiwei Li, Minh-Thang Luong, and Dan Jurafsky. A hierarchical neural autoencoder for paragraphs and documents. In *Annual Meeting of the Association for Computational Linguistics*, 2015.

[117] Q. Li and S. Shah. Learning stock market sentiment lexicon and sentiment-oriented word vector from stocktwits. In *CoNLL*, 2017.

[118] Q. Li, T. Wang, P. Li, L. Liu, Q. Gong, and Y. Chen. The effect of news and public mood on stock movements. *Inf. Sci.*, 278:826–840, 2014.

[119] Q. Li, Y. Chen, L. Jiang, P. Li, and H. Chen. A tensor-based information framework for predicting the stock market. *ACM Trans. Inf. Syst.*, 34:11:1–11:30, 2016.

[120] Q. Li, Y. Chen, L. Jiang, P. Li, and H. Chen. A tensor-based information framework for predicting the stock market. *ACM Trans. Inf. Syst.*, 34:11:1–11:30, 2016.

[121] Q. Li, J. Tan, J. Wang, and H. Chen. A multimodal event-driven lstm model for stock prediction using online news. *IEEE Transactions on Knowledge and Data Engineering*, pages 1–1, 2020.

[122] Q. Li, J. Tan, J. Wang, and H. Chen. A multimodal event-driven lstm model for stock prediction using online news. *IEEE Transactions on Knowledge and Data Engineering*, pages 1–1, 2020.

[123] Q. Li, J. Tan, J. Wang, and H. Chen. A multimodal event-driven lstm model for stock prediction using online news. *IEEE Transactions on Knowledge and Data Engineering*, pages 1–1, 2020.

[124] Qing Li, Tiejun Wang, Ping Li, Ling Liu, Qixu Gong, and Yuanzhu Peter Chen. The effect of news and public mood on stock movements. *Inf. Sci.*, 278:826–840, 2014. URL https://api.semanticscholar.org/CorpusID:37195305.

[125] X. Li, H. Xie, L. Chen, J. Wang, and X. Deng. News impact on stock price return via sentiment analysis. *Knowl. Based Syst.*, 69:14–23, 2014.

[126] X. Li, H. Xie, L. Chen, J. Wang, and X. Deng. News impact on stock price return via sentiment analysis. *Knowl. Based Syst.*, 69:14–23, 2014.

[127] X. Li, H. Xie, Y. Song, S. Zhu, Q. Li, and F. L. Wang. Does summarization help stock prediction? a news impact analysis. *IEEE Intelligent Systems*, 30:26–34, 2015.

[128] X. Li, H. Xie, Y. Song, S. Zhu, Q. Li, and F. L. Wang. Does summarization help stock prediction? a news impact analysis. *IEEE Intelligent Systems*, 30:26–34, 2015.

[129] X. Li, J. Cao, and Z. Pan. Market impact analysis via deep learned architectures. *Neural Computing and Applications*, pages 1–12, 2018.

[130] X. Li, H. Xie, R. Y. K. Lau, T. Wong, and F. Wang. Stock prediction via sentimental transfer learning. *IEEE Access*, 6:73110–73118, 2018.

[131] X. Li, P. Wu, and W. Wang. Incorporating stock prices and news sentiments for stock market prediction: A case of hong kong. *Inf. Process. Manag.*, 57:102212, 2020.

[132] Y. Li and W. Ma. Applications of artificial neural networks in financial economics: A survey. *2010 International Symposium on Computational Intelligence and Design*, 1: 211–214, 2010.

[133] Paul Pu Liang, Liu Ziyin, Amir Zadeh, and Louis-Philippe Morency. Multimodal language analysis with recurrent multistage fusion. In *Conference on Empirical Methods in Natural Language Processing*, 2018.

[134] Q. Liang, W. Rong, J. Zhang, J. Liu, and Z. Xiong. Restricted boltzmann machine based stock market trend prediction. *2017 International Joint Conference on Neural Networks (IJCNN)*, pages 1380–1387, 2017.

[135] Tsung-Yu Lin, Aruni Roy-Chowdhury, and Subhransu Maji. Bilinear cnn models for fine-grained visual recognition. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1449–1457, 2015.

[136] Huicheng Liu. Leveraging financial news for stock trend prediction with attention-based recurrent neural network. *ArXiv*, abs/1811.06173, 2018.

[137] S. Liu, C. Zhang, and J. Ma. Cnn-lstm neural network model for quantitative strategy analysis in stock markets. In *ICONIP*, 2017.

[138] Weidong Liu, Zhiwei Yang, Yan Cao, and Jinchen Huo. Discovering the influences of the patent innovations on the stock market. *Information Processing & Management*, 2022.

[139] Xiao Liu, Heyan Huang, Yue Zhang, and Changsen Yuan. News-driven stock prediction with attention-based noisy recurrent state transition. *ArXiv*, abs/2004.01878, 2020.

[140] Xiao Liu, Heyan Huang, Yue Zhang, and Changsen Yuan. News-driven stock prediction with attention-based noisy recurrent state transition. *ArXiv*, abs/2004.01878, 2020.

[141] Zhuang Liu, Degen Huang, Kaiyu Huang, Zhuang Li, and Jun Zhao. Finbert: A pre-trained financial language representation model for financial text mining. In *International Joint Conference on Artificial Intelligence*, 2020.

[142] Wen Long, Zhichen Lu, and L. Cui. Deep learning-based feature engineering for stock price movement prediction. *Knowl. Based Syst.*, 164:163–173, 2019.

[143] Jiasen Lu, Jianwei Yang, Dhruv Batra, and Devi Parikh. Hierarchical question-image co-attention for visual question answering. *ArXiv*, abs/1606.00061, 2016.

[144] Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. In *Neural Information Processing Systems*, 2019.

[145] N. Mahmoudi, P. Docherty, and P. Moscato. Deep neural networks understand investors better. *Decis. Support Syst.*, 112:23–34, 2018.

[146] T. Matsubara, R. Akita, and K. Uehara. Stock price prediction by deep neural generative model of news articles. *IEICE Trans. Inf. Syst.*, 101-D:901–908, 2018.

[147] T. Matsubara, R. Akita, and K. Uehara. Stock price prediction by deep neural generative model of news articles. *IEICE Trans. Inf. Syst.*, 101-D:901–908, 2018.

[148] D. McDonald, H. Chen, and R. P. Schumaker. Transforming open-source documents to terror networks: The arizona terrornet. In *AAAI Spring Symposium: AI Technologies for Homeland Security*, 2005.

[149] R. David McLean and Jeffrey Pontiff. Does academic research destroy stock return predictability. *Journal of Finance*, 71:5–32, 2016.

[150] S. Minami. Predicting equity price with corporate action events using lstm-rnn. *Journal of Mathematical Finance*, 08:58–63, 2018.

[151] A. Mittal. Stock prediction using twitter sentiment analysis. 2011.

[152] Louis-Philippe Morency, Rada Mihalcea, and Payal Doshi. Towards multimodal sentiment analysis: Harvesting opinions from the web. In *Proceedings of the 13th international conference on multimodal interfaces*, pages 169–176, 2011.

[153] J. Murphy. Technical analysis of the financial markets. 1999.

[154] Hyeonseob Nam, Jung-Woo Ha, and Jeonghee Kim. Dual attention networks for multimodal reasoning and matching. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2156–2164, 2016.

[155] K. Nam and N. Seong. Financial news-based stock movement prediction using causality analysis of influence in the korean stock market. *Decis. Support Syst.*, 117:100–112, 2019.

[156] A. K. Nassirtoussi, S. Aghabozorgi, T. Y. Wah, and D. Ngo. Text mining for market prediction: A systematic review. *Expert Syst. Appl.*, 41:7653–7670, 2014.

[157] D. Nelson, A. M. Pereira, and R. A. de Oliveira. Stock market's price movement prediction with lstm neural networks. *2017 International Joint Conference on Neural Networks (IJCNN)*, pages 1419–1426, 2017.

[158] David M. Q. Nelson, Adriano M. Pereira, and Renato A. de Oliveira. Stock market's price movement prediction with lstm neural networks. *2017 International Joint Conference on Neural Networks (IJCNN)*, pages 1419–1426, 2017. URL https://api.semanticscholar.org/CorpusID:206919491.

[159] T. H. Nguyen and K. Shirai. Topic modeling based sentiment analysis on social media for stock market prediction. In *ACL*, 2015.

[160] T. H. Nguyen, K. Shirai, and J. Velcin. Sentiment analysis on social media for stock movement prediction. *Expert Syst. Appl.*, 42:9603–9611, 2015.

[161] L. P. Ni, Z. W. Ni, and Y. Z. Gao. Stock trend prediction based on fractal feature selection and support vector machine. *Expert Syst. Appl.*, 38:5569–5576, 2011.

[162] Isaac Kofi Nti, Adebayo Felix Adekoya, and Benjamin Asubam Weyori. A systematic review of fundamental and technical analysis of stock market predictions. *Artificial Intelligence Review*, 53:3007–3057, 2019. URL https://api.semanticscholar.org/CorpusID:201064622.

[163] W. Nuij, V. Milea, F. Hogenboom, F. Frasincar, and U. Kaymak. An automated framework for incorporating news into stock trading strategies. *IEEE Transactions on Knowledge and Data Engineering*, 26:823–835, 2014.

[164] W. Nuij, V. Milea, F. Hogenboom, F. Frasincar, and U. Kaymak. An automated framework for incorporating news into stock trading strategies. *IEEE Transactions on Knowledge and Data Engineering*, 26:823–835, 2014.

[165] W. Nuij, V. Milea, F. Hogenboom, F. Frasincar, and U. Kaymak. An automated framework for incorporating news into stock trading strategies. *IEEE Transactions on Knowledge and Data Engineering*, 26:823–835, 2014.

[166] C. Oh and O. Sheng. Investigating predictive power of stock micro blog sentiment in forecasting future stock price directional movement. In *ICIS*, 2011.

[167] Pisut Oncharoen and Peerapon Vateekul. Deep learning for stock market prediction using event embedding and technical indicators. *2018 5th International Conference on Advanced Informatics: Concept Theory and Applications (ICAICTA)*, pages 19–24, 2018.

[168] Pisut Oncharoen and Peerapon Vateekul. Deep learning for stock market prediction using event embedding and technical indicators. *2018 5th International Conference on*

Advanced Informatics: Concept Theory and Applications (ICAICTA), pages 19–24, 2018.

[169] A. Oztekin, R. Kizilaslan, S. Freund, and A. Iseri. A data analytic approach to forecasting daily stock returns in an emerging market. *Eur. J. Oper. Res.*, 253:697–710, 2016.

[170] P. Pai and C. S. Lin. A hybrid arima and support vector machines model in stock price forecasting. *Omega-international Journal of Management Science*, 33:497–505, 2005.

[171] Hyun Jun Park, Youngjun Kim, and Ha Young Kim. Stock market forecasting using a multi-task approach integrating long short-term memory and the random forest framework. *Applied Soft Computing*, 114:108106, 2022. ISSN 1568-4946. doi: https://doi.org/10.1016/j.asoc.2021.108106.

[172] J. Patel, S. Shah, P. Thakkar, and K. Kotecha. Predicting stock and stock price index movement using trend deterministic data preparation and machine learning techniques. *Expert Syst. Appl.*, 42:259–268, 2015.

[173] J. Patel, S. Shah, P. Thakkar, and K. Kotecha. Predicting stock market index using fusion of machine learning techniques. *Expert Syst. Appl.*, 42:2162–2172, 2015.

[174] Romain Paulus, Caiming Xiong, and Richard Socher. A deep reinforced model for abstractive summarization. *ArXiv*, abs/1705.04304, 2017.

[175] Y. Peng and H. Jiang. Leverage financial news to predict stock price movements using word embeddings and deep neural networks. *ArXiv*, abs/1506.07220, 2016.

[176] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. Glove: Global vectors for word representation. In *EMNLP*, 2014.

[177] Verónica Pérez-Rosas, Rada Mihalcea, and Louis-Philippe Morency. Utterance-level multimodal sentiment analysis. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 973–982, 2013.

[178] N. Pröllochs and S. Feuerriegel. Business analytics for strategic management: Identifying and assessing corporate challenges via topic modeling. *Inf. Manag.*, 57, 2020.

[179] N. Pröllochs, S. Feuerriegel, and D. Neumann. Negation scope detection in sentiment analysis: Decision support for news-driven trading. *Decis. Support Syst.*, 88:67–75, 2016.

[180] Y. Qian, Z. Li, and H. Yuan. On exploring the impact of users' bullish-bearish tendencies in online community on the stock market. *Inf. Process. Manag.*, 57:102209, 2020.

[181] X. Qiu, L. Zhang, Y. Ren, P. Suganthan, and G. Amaratunga. Ensemble deep learning for regression and time series forecasting. *2014 IEEE Symposium on Computational Intelligence in Ensemble Learning (CIEL)*, pages 1–6, 2014.

[182] Shyam Sundar Rajagopalan, Louis-Philippe Morency, Tadas Baltruaitis, and Roland Göcke. Extending long short-term memory for multi-view structured learning. In *European Conference on Computer Vision*, 2016.

[183] Eduardo Ramos-Pérez, Pablo J. Alonso-González, and José Javier Núñez-Velázquez. Multi-transformer: A new neural network-based architecture for forecasting s&p volatility. *ArXiv*, abs/2109.12621, 2021. URL https://api.semanticscholar.org/CorpusID:237716022.

[184] T. Rao and S. Srivastava. Analyzing stock market movements using twitter sentiment analysis. In *ASONAM 2012*, 2012.

[185] A. Ratto, S. Merello, Y. Ma, L. Oneto, and E. Cambria. Technical analysis and sentiment embeddings for market trend prediction. *Expert Syst. Appl.*, 135:60–70, 2019.

[186] Hadi Rezaei, Hamidreza Faaljou, and Gholamreza Mansourfar. Stock price prediction using deep learning and frequency decomposition. *Expert Syst. Appl.*, 169:114332, 2021.

[187] Farnoush Ronaghi, Mohammad Salimibeni, Farnoosh Naderkhani, and Arash Mohammadi. Covid19-hpsmp: Covid-19 adopted hybrid and parallel deep information fusion framework for stock price movement prediction. *Expert Systems with Applications*, 187:115879, 2022.

[188] M. M. Rounaghi and F. N. Zadeh. Investigation of market efficiency and financial stability between s&p 500 and london stock exchange: Monthly and yearly forecasting of time series stock returns using arma model. *Physica A-statistical Mechanics and Its Applications*, 456:10–21, 2016.

[189] Alexander M. Rush, Sumit Chopra, and Jason Weston. A neural attention model for abstractive sentence summarization. *ArXiv*, abs/1509.00685, 2015.

[190] N. C. Sarantis. Nonlinearities, cyclical behaviour and predictability in stock markets: international evidence. *International Journal of Forecasting*, 17:459–482, 2001.

[191] R. P. Schumaker and H. Chen. Textual analysis of stock market prediction using breaking financial news: The azfin text system. *ACM Trans. Inf. Syst.*, 27:12:1–12:19, 2009.

[192] R. P. Schumaker and H. Chen. Textual analysis of stock market prediction using breaking financial news: The azfin text system. *ACM Trans. Inf. Syst.*, 27:12:1–12:19, 2009.

[193] R. P. Schumaker and H. Chen. A quantitative stock prediction system based on financial news. *Inf. Process. Manag.*, 45:571–583, 2009.

[194] R. P. Schumaker and H. Chen. Textual analysis of stock market prediction using breaking financial news: The azfin text system. *ACM Trans. Inf. Syst.*, 27:12:1–12:19, 2009.

[195] R. P. Schumaker, Y. Zhang, C. Huang, and H. Chen. Evaluating sentiment in financial news articles. *Decis. Support Syst.*, 53:458–464, 2012.

[196] R. P. Schumaker, Y. Zhang, C. Huang, and H. Chen. Evaluating sentiment in financial news articles. *Decis. Support Syst.*, 53:458–464, 2012.

[197] R. P. Schumaker, Y. Zhang, C. Huang, and H. Chen. Evaluating sentiment in financial news articles. *Decis. Support Syst.*, 53:458–464, 2012.

[198] S. Sekine and C. Nobata. Definition, dictionaries and tagger for extended named entity hierarchy. In *LREC*, 2004.

[199] N. Seong and K. Nam. Predicting stock movements based on financial news with segmentation. *Expert Systems With Applications*, page 113988, 2020.

[200] O. B. Sezer and A. Ozbayoglu. Algorithmic financial trading with deep convolutional neural networks: Time series to image conversion approach. *Appl. Soft Comput.*, 70: 525–538, 2018.

[201] L. Shen and H. Loh. Applying rough sets to market timing decisions. *Decis. Support Syst.*, 37:583–597, 2004.

[202] L. Shi, Z. Teng, L. Wang, Y. Zhang, and Alexander Binder. Deepclue: Visual interpretation of text-based deep stock prediction. *IEEE Transactions on Knowledge and Data Engineering*, 31:1094–1108, 2019.

[203] L. Shi, Z. Teng, L. Wang, Y. Zhang, and Alexander Binder. Deepclue: Visual interpretation of text-based deep stock prediction. *IEEE Transactions on Knowledge and Data Engineering*, 31:1094–1108, 2019.

[204] L. Shi, Z. Teng, L. Wang, Y. Zhang, and Alexander Binder. Deepclue: Visual interpretation of text-based deep stock prediction. *IEEE Transactions on Knowledge and Data Engineering*, 31:1094–1108, 2019.

[205] Lei Shi, Zhiyang Teng, Le Wang, Yue Zhang, and Alexander Binder. Deepclue: Visual interpretation of text-based deep stock prediction. *IEEE Transactions on Knowledge and Data Engineering*, 31:1094–1108, 2019.

[206] Lei Shi, Zhiyang Teng, Le Wang, Yue Zhang, and Alexander Binder. Deepclue: Visual interpretation of text-based deep stock prediction. *IEEE Transactions on Knowledge and Data Engineering*, 31:1094–1108, 2019.

[207] Y. Shynkevich, T. Mcginnity, S. Coleman, and A. Belatreche. Forecasting movements of health-care stock prices based on different categories of news articles using multiple kernel learning. *Decis. Support Syst.*, 85:74–83, 2016.

[208] Yauheniya Shynkevich, T. McGinnity, S. Coleman, A. Belatreche, and Yuhua Li. Forecasting price movements using technical indicators: Investigating the impact of varying input window length. *Neurocomputing*, 264:71–88, 2017.

[209] J. Si, A. Mukherjee, B. Liu, Q. Li, H. Li, and X. Deng. Exploiting topic based twitter sentiment for stock prediction. In *ACL*, 2013.

[210] M. Siering and J. Muntermann. The role of misbehavior in efficient financial markets: Implications for financial decision support. In *FinanceCom*, 2012.

[211] J. Smailovic, M. Grcar, N. Lavrac, and M. Znidarsic. Stream-based active learning for sentiment analysis in the financial domain. *Inf. Sci.*, 285:181–203, 2014.

[212] Nitish Srivastava, Geoffrey E. Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.*, 15:1929–1958, 2014.

[213] L. A. Teixeira and A. Oliveira. A method for automatic stock trading combining technical analysis and nearest neighbor classification. *Expert Syst. Appl.*, 37:6885–6890, 2010.

[214] P. C. Tetlock. Giving content to investor sentiment: The role of media in the stock market. *Journal of Finance*, 62:1139–1168, 2007.

[215] P. C. Tetlock, M. Saar-Tsechansky, and S. A. Macskassy. More than words: Quantifying language to measure firms' fundamentals. *Journal of Finance*, 63:1437–1467, 2008.

[216] A Thakkar and K Chaudhari. Fusion in stock market prediction: A decade survey on the necessity, recent developments, and potential future directions. *Information Fusion*, 65:95–107, 2021.

[217] Ankit Thakkar and Kinjal Chaudhari. Fusion in stock market prediction: A decade survey on the necessity, recent developments, and potential future directions. *An International Journal on Information Fusion*, 65:95 – 107, 2020. URL https://api.semanticscholar.org/CorpusID:221323781.

[218] Ankit Thakkar and Kinjal Chaudhari. Predicting stock trend using an integrated term frequency-inverse document frequency-based feature weight matrix with neural networks. *Appl. Soft Comput.*, 96:106684, 2020.

[219] J. L. Ticknor. A bayesian regularized artificial neural network for stock market forecasting. *Expert Syst. Appl.*, 40:5501–5506, 2013.

[220] K. Tolle and H. Chen. Comparing noun phrasing techniques for use with medical digital library tools. *Journal of the Association for Information Science and Technology*, 51: 352–370, 2000.

[221] Yao-Hung Hubert Tsai, Shaojie Bai, Paul Pu Liang, J. Zico Kolter, Louis-Philippe Morency, and Ruslan Salakhutdinov. Multimodal transformer for unaligned multimodal language sequences. *Proceedings of the conference. Association for Computational Linguistics. Meeting*, 2019:6558–6569, 2019. URL https://api.semanticscholar.org/CorpusID:173990158.

[222] P. Tsinaslanidis and D. Kugiumtzis. A prediction scheme using perceptually important points and dynamic time warping. *Expert Syst. Appl.*, 41:6848–6860, 2014.

[223] B. Vanstone and G. Finnie. An empirical methodology for developing stockmarket trading systems using artificial neural networks. *Expert Syst. Appl.*, 36:6668–6680, 2009.

[224] V. N. Vapnik. The nature of statistical learning theory. In *Statistics for Engineering and Information Science*, 2000.

[225] M. R. Vargas, B. S. L. P. De Lima, and A. Evsukoff. Deep learning for stock market prediction from financial news articles. *2017 IEEE International Conference on Computational Intelligence and Virtual Environments for Measurement Systems and Applications (CIVEMSA)*, pages 60–65, 2017.

[226] M. R. Vargas, B. S. L. P. De Lima, and A. Evsukoff. Deep learning for stock market prediction from financial news articles. *2017 IEEE International Conference on Computational Intelligence and Virtual Environments for Measurement Systems and Applications (CIVEMSA)*, pages 60–65, 2017.

[227] M. R. Vargas, B. S. L. P. De Lima, and A. Evsukoff. Deep learning for stock market prediction from financial news articles. *2017 IEEE International Conference on*

Computational Intelligence and Virtual Environments for Measurement Systems and Applications (CIVEMSA), pages 60–65, 2017.

[228] Ashish Vaswani, Noam M. Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NIPS*, 2017.

[229] Sunny Verma, Chen Wang, Liming Zhu, and Wei Liu. Deepcu: Integrating both common and unique latent information for multimodal sentiment analysis. In *International Joint Conference on Artificial Intelligence*, 2019.

[230] B. Wang, H. Huang, and X. Wang. A novel text mining approach to financial time series forecasting. *Neurocomputing*, 83:136–145, 2012.

[231] Chenhao Wang and Qiang Gao. High and low prices prediction of soybean futures with lstm neural network. *2018 IEEE 9th International Conference on Software Engineering and Service Science (ICSESS)*, pages 140–143, 2018.

[232] J. Z. Wang, J. J. Wang, Z. Zhang, and S. P. Guo. Forecasting stock indices with back propagation neural network. *Expert Syst. Appl.*, 38:14346–14355, 2011.

[233] Jia Wang, Tong Sun, Benyuan Liu, Yu Cao, and Hongwei Zhu. Clvsa: A convolutional lstm based variational sequence-to-sequence model with attention for predicting trends of financial markets. *ArXiv*, abs/2104.04041, 2019. URL https://api.semanticscholar.org/CorpusID:199465738.

[234] L-X. Wang. Fast training algorithms for deep convolutional fuzzy systems with application to stock index prediction. *IEEE Transactions on Fuzzy Systems*, 28:1301–1314, 2020.

[235] Q. Wang, W. Xu, and H. Zheng. Combining the wisdom of crowds and technical analysis for financial market prediction using deep random subspace ensembles. *Neurocomputing*, 299:51–61, 2018.

[236] X. Wang, Ross B. Girshick, Abhinav Kumar Gupta, and Kaiming He. Non-local neural networks. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7794–7803, 2017.

[237] X. Wang, J. Wang, H. Li, S. Liu, and H. Liu. Research and implementation of svm and bootstrapping fusion algorithm in emotion analysis of stock review texts. *DEStech Transactions on Computer Science and Engineering*, 2019.

[238] B. Weng, L. Lu, X. Wang, F. Megahed, and W. Martinez. Predicting short-term stock prices using ensemble methods and online data sources. *Expert Syst. Appl.*, 112:258–273, 2018.

[239] H. White. Economic prediction using neural networks: the case of ibm daily stock returns. *IEEE 1988 International Conference on Neural Networks*, pages 451–458 vol.2, 1988.

[240] J. L. Wu, C. Su, L. Yu, and P. C. Chang. Stock price predication using combinational features from sentimental analysis of stock news and technical analysis of trading information. In *Economics Development and Research*, 2012.

[241] Shijie Wu, Ozan Irsoy, Steven Lu, Vadim Dabravolski, Mark Dredze, Sebastian Gehrmann, Prabhanjan Kambadur, David Rosenberg, and Gideon Mann. Bloomberggpt: A large language model for finance. *ArXiv*, abs/2303.17564, 2023. URL https://api.semanticscholar.org/CorpusID:257833842.

[242] X. Wu, X. Zhu, G. Wu, and W. Ding. Data mining with big data. *IEEE Transactions on Knowledge and Data Engineering*, 26:97–107, 2014.

[243] Yiling Wu, Shuhui Wang, Guoli Song, and Qingming Huang. Learning fragment self-attention embeddings for image-text matching. *Proceedings of the 27th ACM International Conference on Multimedia*, 2019.

[244] Tianjun Xiao, Yichong Xu, Kuiyuan Yang, Jiaxing Zhang, Yuxin Peng, and Zheng Zhang. The application of two-level attention models in deep convolutional neural network for fine-grained image classification. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 842–850, 2014.

[245] Yuteng Xiao, Hongsheng Yin, Yudong Zhang, Honggang Qi, Yundong Zhang, and Zhaoyang Liu. A dual-stage attention-based conv-lstm network for spatio-temporal

correlation and multivariate time series prediction. *International Journal of Intelligent Systems*, 36(5):2036–2057, 2021.

[246] H. Xu, Chai L, Z. Luo, and S. Li. Stock movement predictive network via incorporative attention mechanisms based on tweet and historical prices. *Neurocomputing*, 418:326–339, 2020.

[247] Hongfeng Xu, Lei Chai, Zhiming Luo, and Shaozi Li. Stock movement predictive network via incorporative attention mechanisms based on tweet and historical prices. *Neurocomputing*, 418:326–339, 2020. URL https://api.semanticscholar.org/CorpusID:225334278.

[248] Hongfeng Xu, Lei Chai, Zhiming Luo, and Shaozi Li. Stock movement prediction via gated recurrent unit network based on reinforcement learning with incorporated attention mechanisms. *Neurocomputing*, 467:214–228, 2022.

[249] Yumo Xu and Shay B. Cohen. Stock movement prediction from tweets and historical prices. In *ACL*, 2018.

[250] Hongyang Yang, Xiao-Yang Liu, and Chris Wang. Fingpt: Open-source financial large language models. *ArXiv*, abs/2306.06031, 2023. URL https://api.semanticscholar.org/CorpusID:259129734.

[251] Tao Yin, Chenzhengyi Liu, Fangyu Ding, Ziming Feng, Bo Yuan, and Ning Zhang. Graph-based stock correlation and prediction for high-frequency trading systems. *Pattern Recognit.*, 122:108209, 2022.

[252] Quanzeng You, Hailin Jin, Zhaowen Wang, Chen Fang, and Jiebo Luo. Image captioning with semantic attention. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4651–4659, 2016.

[253] L Yu, H Chen, S Wang, and K K. Lai. Evolving least squares support vector machines for stock market trend mining. *IEEE Trans. Evol. Comput.*, 13:87–102, 2009.

[254] Y. Yu, W. Duan, and Q. Cao. The impact of social and conventional media on firm equity value: A sentiment analysis approach. *Decis. Support Syst.*, 55:919–926, 2013.

[255] Zhou Yu, Jun Yu, Yuhao Cui, Dacheng Tao, and Qi Tian. Deep modular co-attention networks for visual question answering. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6274–6283, 2019.

[256] Amir Zadeh, Rowan Zellers, Eli Pincus, and Louis-Philippe Morency. Multimodal sentiment intensity analysis in videos: Facial gestures and verbal messages. *IEEE Intelligent Systems*, 31(6):82–88, 2016.

[257] Amir Zadeh, Minghai Chen, Soujanya Poria, E. Cambria, and Louis-Philippe Morency. Tensor fusion network for multimodal sentiment analysis. *ArXiv*, abs/1707.07250, 2017.

[258] Amir Zadeh, Paul Pu Liang, Navonil Mazumder, Soujanya Poria, E. Cambria, and Louis-Philippe Morency. Memory fusion network for multi-view sequential learning. In *AAAI Conference on Artificial Intelligence*, 2018.

[259] Amir Zadeh, Paul Pu Liang, Soujanya Poria, E. Cambria, and Louis-Philippe Morency. Multimodal language analysis in the wild: Cmu-mosei dataset and interpretable dynamic fusion graph. In *Annual Meeting of the Association for Computational Linguistics*, 2018.

[260] Boyu Zhang, Hongyang Yang, and Xiao-Yang Liu. Instruct-fingpt: Financial sentiment analysis by instruction tuning of general-purpose large language models. *ArXiv*, abs/2306.12659, 2023. URL https://api.semanticscholar.org/CorpusID:259224880.

[261] G. Zhang. Time series forecasting using a hybrid arima and neural network model. *Neurocomputing*, 50:159–175, 2003.

[262] G. Zhang, L. Xu, and Y. Xue. Model and forecast stock market behavior integrating investor sentiment analysis and transaction data. *Cluster Computing*, 20:789–803, 2017.

[263] Liheng Zhang, Charu C. Aggarwal, and Guo-Jun Qi. Stock price prediction via discovering multi-frequency trading patterns. *Proceedings of the 23rd ACM SIGKDD*

International Conference on Knowledge Discovery and Data Mining, 2017. URL https://api.semanticscholar.org/CorpusID:28307599.

[264] N. Zhang, A. Lin, and P. Shang. Multidimensional k-nearest neighbor model based on eemd for financial time series forecasting. *Physica A-statistical Mechanics and Its Applications*, 477:161–173, 2017.

[265] Xi Zhang, Yunjia Zhang, Senzhang Wang, Yuntao Yao, Bin Xing Fang, and Philip S. Yu. Improving stock market prediction via heterogeneous information fusion. *Knowl. Based Syst.*, 143:236–247, 2017. URL https://api.semanticscholar.org/CorpusID:3708860.

[266] Xi Zhang, Yunjia Zhang, Senzhang Wang, Yuntao Yao, Binxing Fang, and Philip S. Yu. Improving stock market prediction via heterogeneous information fusion. *Knowl. Based Syst.*, 143:236–247, 2018.

[267] X. Zhong and D. Enke. Forecasting daily stock market return using dimensionality reduction. *Expert Syst. Appl.*, 67:126–139, 2017.

[268] Z. Çopur. Handbook of research on behavioral finance and investment strategies: Decision making in the financial industry. 2015.