Essays on Finance and Deep Learning

A thesis presented in partial fulfillment of the requirements for

the degree of Doctor of Philosophy in Finance

School of Economics and Finance

Massey University

Guoyao Pan

Supervisor: Associate Professor Yafeng Qin

Professor Nuttawat Visaltanachoti

July 2025

Abstract

This thesis aims to broaden the application of deep learning techniques in financial research and comprises three essays that make meaningful contributions to the related literature.

Essay One integrates deep learning into the Hub Strategy, a novel chart pattern analysis method, to develop trading strategies. Utilizing deep learning models, which analyze chart patterns alongside data such as trading volume, price volatility, and sentiment indicators, the strategy forecasts stock price movements. Tests on U.S. S&P 500 index stocks indicate that Hub Strategy trading methods, when integrated with deep learning models, achieve an annualized average return of approximately 25%, significantly outperforming the benchmark buy-and-hold strategy's 9.6% return. Risk-adjusted metrics, including Sharpe ratios and Jensen's alpha, consistently demonstrate the superiority of these trading strategies over both the buy-and-hold approach and standalone Hub Strategy trading rules. To address data snooping concerns, multiple tests validate profitability, and an asset pricing model with 153 risk factors and Lasso-OLS (Ordinary Least Squares) regressions confirms its ability to capture positive alphas.

Essay Two utilizes deep learning techniques to explore the relationships between the abnormal return and its explanatory variables, including firm-specific characteristics and realized stock returns. Trained deep learning models effectively predict the estimated abnormal return directly. We evaluate the effectiveness of detecting abnormal returns by comparing our deep learning models against three benchmark methods. When applied to a

random dataset, deep learning models demonstrate a significant improvement in identifying abnormal returns within the induced range of -3% to 3%. Moreover, their performance remains consistent across non-random datasets classified by firm size and market conditions. In addition, a regression of abnormal return prediction errors on firm-based factors, market conditions, and periods reveals that deep learning models are less sensitive to variables like firm size, market conditions, and periods than the benchmarks.

Essay Three assesses the performance of deep learning predictors in forecasting momentum turning points using the confusion matrix and comparing them to the benchmark model proposed by Goulding, Harvey, and Mazzoleni (2023). Tested on U.S. stocks from January 1990 to December 2023, deep learning predictors demonstrate higher accuracy in identifying turning points than the benchmark. Furthermore, our deep learning-based trading rules yield higher mean log returns and Sharpe ratios, along with lower volatility, compared to the benchmark. Two models achieve average monthly returns of 0.0148 and 0.0177, surpassing the benchmark's 0.0108. These gains are both economically and statistically significant, with consistent annual results. Regression analysis also shows that our models respond more effectively to changes in stock and market return volatility than the benchmark.

Overall, these essays expand the application of deep learning in finance research, demonstrating high predictive accuracy, enhanced trading profitability, and effective detection of long-term abnormal returns, all of which hold significant practical value.

Acknowledgments

I extend my heartfelt gratitude to my supervisors, Associate Professor Yafeng Qin and Professor Nuttawat Visaltanachoti, for their steadfast support, expert guidance, and invaluable insights throughout my research and thesis writing. Their expertise and inspiration have been pivotal in shaping the direction of this thesis. The journey to becoming a researcher would have been far more challenging without their constant encouragement, thoughtful feedback, and unwavering patience.

I am profoundly thankful to the faculty and staff of the Massey University School of Economics and Finance, particularly Professor Sasha Molchanov, Associate Professor Hung Do, and Dr. Jasmine Fang, for their constructive feedback during my course confirmation process. Their support, along with the encouraging academic environment and resources they provided, has been instrumental in facilitating my research efforts.

I would like to express my sincere gratitude for the financial support provided by Massey University through the Massey University Doctoral Scholarship. This generous assistance has been essential in enabling me to pursue my research. My heartfelt thanks also go to Mark Woods for his invaluable technical support, as well as to Myrah Corrales and Muharram Azizova for their outstanding administrative assistance.

I am deeply grateful to my friends and family, with special thanks to my wife, Juan Lu. This journey would not have been possible without her tremendous sacrifices, patience, and unwavering encouragement. Her support has been invaluable, and I am truly appreciative of her dedication throughout this entire process.

Table of Contents

List of Tables

List of Figures

Chapter One

Introduction

This section summarizes the content of this thesis, which explores the application of deep learning (DL) techniques across three key research areas: chart pattern trading strategies, the abnormal return (AR) estimation, and enhancements to time series momentum (TSMOM) trading rules.

1.1 Introduction

Deep Learning (DL), a type of Artificial Neural Network (ANN) with multiple layers, offers the key advantage of automatically extracting high-quality features from input data through a general-purpose learning process. Recent advancements in computer hardware and software, such as graphics processing units (GPUs) and TensorFlow, have fueled the rapid growth of DL applications. DL has been widely adopted in diverse fields, including autonomous driving for cars and drones, computer vision, audiovisual recognition, image processing, classification, and prediction tasks (e.g., Collobert, Weston, Bottou, Karlen, Kavukcuoglu, and Kuksa, 2011; Guo, Liu, Oerlemans, Lao, Wu, and Lew, 2016; Chai and Li, 2019; Chai and Ngai 2020; Partheepan, Sanati, and Hassan, 2023; Botezatu, Burlacu, and Orhei, 2024; Zhang, Yang, Chen, Zhang, Leng, and Zhao, 2024).

Similarly, DL has garnered significant attention in the finance sector. Its applications, particularly in predicting financial movements, processing textual information, and enhancing trading strategies, are rapidly evolving, highlighting DL's growing influence in finance (e.g., Jurgovsky, Granitzer, Ziegler, Calabretto, Portier, Guelton, and Caelen, 2018; Matsubara, Akita, and Uehara, 2018; Hambly, Xu, and Yang, 2023; Jiang, Kelly, and Xiu, 2023; Nazareth, and Reddy, 2023; Billah, Sultana, Bhuiyan, and Kaosar, 2024; Chen, Pelger, and Zhu, 2024; Chowdhury, Nabi, Rana, Shaima, Esa, Mitra, and Nazni, 2024; Feng, He, Polson, and Xu, 2024; Zheng, Wu, Song, Guo, and Xu, 2024). Heaton, Polson, and Witte (2016) identify several advantages of DL models over traditional financial methods, including their ability to account for nonlinearities and complex interactions in data, effectively manage overfitting, utilize all available data as input, and operate without requiring assumptions about the data.

They stress that theoretical models based on existing axiomatic foundations are unlikely to match the predictive power of DL models. The implications for the future of financial economics remain uncertain, but applying DL models to enhance predictability in financial markets represents both a promising opportunity and a meaningful challenge.

This thesis aims to broaden the application of Deep Learning (DL) techniques in financial research, with a particular focus on three areas. Specifically, it explores the integration of DL into chart pattern trading strategies, the estimation of the abnormal return (AR), and the enhancement of time series momentum (TSMOM) rules. These topics are examined in detail across the three essays that comprise the thesis:

- Essay One investigates the use of DL models to develop advanced chart pattern-based trading strategies, highlighting their potential to improve predictive accuracy and trading outcomes.
- Essay Two focuses on employing DL techniques to refine the estimation of AR, emphasizing the advantages of DL in handling complex financial data and improving estimation reliability.
- Essay Three examines the role of DL in enhancing TSMOM rules, showcasing its ability to adapt to market dynamics and deliver superior risk-adjusted returns.

By addressing these distinct yet interconnected topics, this thesis demonstrates the versatility and transformative potential of DL techniques in advancing financial research and practice.

1.2 Essay One

Chart pattern analysis, a subset of technical analysis (TA), has gained academic attention in recent decades. However, most related studies highlight the limited profitability of traditional chart patterns—such as spikes, flags, pennants, triangles, head-and-shoulders, and various tops and bottoms (e.g., Edwards and Magee, 1996; Schwager, 1995; Pring, 2002). Additionally, few studies directly extract structural features of chart patterns to feed into deep learning (DL) models in order to enhance their performance.

Essay One introduces a novel approach to chart pattern analysis, Hub Strategy (HS), and integrates DL techniques to develop HSDL trading strategies. We aim to demonstrate the superior profitability of HSDL strategies and showcase how AI can revitalize chart pattern analysis in investment. Our key research question is whether HSDL trading rules can significantly outperform both the buy-and-hold (BH) strategy and standalone Hub Strategy (HS) rules.

We evaluate the HSDL, HS, and buy-and-hold (BH) strategies using U.S. S&P 500 index constituent stocks. Our findings reveal that HSDL achieves higher profitability compared to HS and BH, as measured by annualized average returns, risk-adjusted metrics, and performance after accounting for transaction costs. Analyzing performance on the 10 worst market days with significant declines, we find HSDL is resilient to crash risk, delivering substantially higher returns than the BH benchmark. To address data snooping concerns, we employ the "false discovery rate" (FDR) control method presented by Benjamini and Hochberg (1995) and Giglio, Liao, and Xiu (2021). Tests of six null hypotheses confirm that HSDL buyers outperform BH, sellers generate profits on most stocks,

and HSDL shows clear improvement over HS. Additionally, using a 153-factor asset pricing model, we demonstrate that HSDL strategies capture positive alphas across a majority of stocks.

Our study offers several key contributions to the literature. First, we integrate deep learning (DL) techniques, focused on cross-sectional data, with the Hub Strategy (HS)-a novel chart pattern analysis method-to develop the HSDL trading strategy. This integration enables traders and analysts to perform chart analysis quantitatively and in a multidimensional framework, introducing a new type of input: chart-based features derived from HS patterns. Additionally, the automated method we developed for identifying chart patterns provides a valuable contribution to the field of traditional chart pattern analysis. Finally, we conduct a comprehensive series of tests to rigorously validate the robustness of HSDL's predictive superiority over the buy-and-hold (BH) and standalone Hub Strategy (HS) approaches. Our findings demonstrate that HSDL's enhanced profitability offers significant value for investors in practical applications.

1.3 Essay Two

In event studies, researchers analyze abnormal stock returns over periods of one to five years following significant corporate event disclosures. The key focus is estimating normal returns or benchmarks, which are used to identify abnormal returns by subtracting normal returns from realized returns. For instance, Lyon, Barber, and Tsai (1999) use reference portfolios as benchmarks, while Bessembinder, Cooper, and Zhang (2019) suggest using fitted values from regressions on firm characteristics as benchmarks.

Building on Bessembinder, Cooper, and Zhang (2019), our study introduces two key innovations. First, we replace their traditional linear regression method with DL techniques. Second, we redefine the estimation of abnormal returns by directly modeling their relationship with firm characteristics and realized returns. Unlike their approach, which predicts normal returns via regression, our method uses DL to estimate abnormal returns directly for a given stock over the target period based on known returns and characteristics. The research question is whether our DL estimator can outperform the benchmark from Bessembinder, Cooper, and Zhang (2019).

To explore the factors driving the superior performance of DL models, we conduct a simulation comparing the abnormal return prediction errors of our DL model with those of the Bessembinder benchmark (as proposed by Bessembinder, Cooper, and Zhang (2019)). Additionally, we perform regression analysis on the prediction errors of abnormal returns using relevant firm characteristics to gain deeper insights.

Our study makes several important contributions to event studies. First, we introduce deep learning (DL) as a novel, data-driven method for estimating abnormal returns, moving beyond traditional linear regression models. The DL model is trained on cross-sectional data. Second, we present a fresh perspective by directly modeling the relationship between abnormal returns, firm characteristics, and realized returns, enhancing the understanding of this complex interaction. Additionally, we highlight the strong connection between firm characteristics and future stock returns, offering new insights into stock return dynamics. Furthermore, we demonstrate the superior performance of DL models over benchmark methods in power tests, advancing research on abnormal return estimation and stock return

analysis. Finally, this study presents a method framework for training a practical AR estimator, providing researchers with a valuable tool for conducting event study analyses.

1.4 Essay Three

This research aims to accurately predict the occurrence of TSMOM turning points for stock price movements by leveraging deep learning (DL) techniques and consequently achieve better performance than the method (called GHM2) presented by Goulding, Harvey, and Mazzoleni (2023) (hereafter, GHM). Our research question is whether DL trading rules can beat the benchmark GHM2.

We use a confusion matrix to compare the prediction accuracy of turning points in U.S. stocks from 1990 to 2023 between our DL models and GHM2. Additionally, we calculate four performance metrics—Accuracy, Precision, Recall, and F1-score—based on the confusion matrix results. To present our findings, we evaluate the prediction of turning points for relevant stocks annually and examine the profitability improvements of our models compared to GHM2. Both the DL model and GHM2 benchmark are applied to all test stocks over the study period.

This paper advances time-series momentum (TSMOM) research in several key ways. First, we apply deep learning (DL) techniques to advance TSMOM strategies, outperforming the GHM benchmark by more accurately predicting turning points and enabling more effective long and short position adjustments. The DL predictor is trained on time-series data. Second, confusion matrix analysis of annual predictions (1990–2023) shows our models achieve more true positives, fewer false negatives, and significantly better Accuracy,

Precision, Recall, and F1-score metrics than GHM2. Third, applying DL and GHM2 to TSMOM trading rules reveals that DL models deliver higher monthly mean returns with lower volatility, resulting in superior Sharpe ratios. Lastly, results from cross-sectional and time-series regressions demonstrate that DL models respond more effectively to changes in stock and market return volatility, confirming their superior performance.

Overall, this study enhances the existing TSMOM trading rules by integrating deep learning strategies, which we hope will prove valuable for practical investment applications.

1.5 Structure of The Thesis

The remainder of the thesis proceeds as follows. Chapter Two explores the application of deep learning (DL) models in developing advanced trading strategies based on chart patterns, highlighting their improved predictive accuracy and trading performance. Chapter Three utilizes DL techniques to enhance the abnormal return (AR) estimation, highlighting DL's strengths in managing complex financial data and increasing estimation accuracy. Chapter Four examines the role of DL in enhancing time series momentum (TSMOM) rules, demonstrating its adaptability to market dynamics and its capacity to achieve superior risk-adjusted returns. Chapter Five concludes the thesis by presenting the important findings, meaningful implications, and limitations of each of the three essays.

Chapter Two

Chart Analysis with Deep Learning

This chapter presents the first essay, which incorporates the deep learning (DL) prediction model into the Hub Strategy (HS) to develop the HSDL trading rule. The essay evaluates its profitability using a sample of constituent stocks from the US S&P 500 index. Section 2.1 presents a brief overview of the essay and explains the motivation to conduct this study. Section 2.2 summarizes the literature review. Section 2.3 develops our methodology relative to HS and HSDL prediction models. Section 2.4 analyzes the empirical results. Section 2.5 concludes the paper. Appendices and the reference list of this chapter are reported at the end of this thesis.

Abstract

We integrate the deep learning (DL) prediction model into the Hub Strategy (HS), a novel method of chart analysis, and generate HSDL trading strategies. DL models study HS chart patterns and additional information, including trading volumes, price volatility, and sentiment indicators, to predict price movements of stocks. Our empirical tests on US S&P 500 index constituent stocks show that HSDL strategies can realize an annualized average return of around 25%, compared to an equivalent return of 9.6% from the benchmark, the buy-and-hold (BH) rule. The risk-adjusted measures, such as Sharpe ratios and Jensen alphas, consistently demonstrate that the HSDL strategy outperforms both the BH and pure HS strategies. To address data snooping concerns, we control for the false discovery rate (FDR) and conduct multiple tests to verify the profitability of HSDL relative to the benchmarks. We also demonstrate its ability to capture significantly positive alphas based on an asset pricing model with 153 risk factors, using Lasso, and Ordinary Least Squares regressions.

Keywords: Deep Learning; Technical analysis; Chart analysis; Hub Strategy; Return predictability

2.1 Introduction

Chart analysis is a key technique in technical analysis (TA) used to identify recurring price patterns in financial markets. These chart patterns, derived from historical price movements, help traders and investors anticipate potential future price trends. The underlying principle of chart analysis is that market psychology and investor behavior tend to repeat, forming recognizable structures on price charts.

In recent decades, academic research has also explored chart analysis; however, most studies highlight its limited profitability, likely due to the subjectivity and weak predictive signals of traditional chart patterns. With advancements in artificial intelligence (AI), this study introduces a novel chart analysis method, the Hub Strategy (HS), and integrates deep learning (DL) techniques to develop HSDL trading strategies that incorporate DL-based predictors. Our research aims to demonstrate the superior profitability of HSDL trading strategies, to significantly outperform both the buy-and-hold (BH) rule and standalone Hub Strategy (HS) trading rules.

Hub Strategy, introduced by Biao Li, a well-known Chinese trader, was published on his blog between 2006 and 2008. It quickly gained widespread attention in China due to its high success rate. Evidence of its popularity can be seen in a "Baidu" search, which returns over 10 million relevant results, including publications and tutorials in China[1, 2].

The Hub Strategy (HS) views stock price movements as a series of hubs, where each

1. https://www.baidu.com/s?ie=utf-8&f=3&rsv_bp=1&rsv_idx=1&tn=baidu&wd=%E7%BC%A0%E4%B8%AD%E8%AF%B4%E7%A6%85&fenlei=256&rsv_pq=f9153042004d66df&rsv_t=462eo%2FZBEMEBFZKgPLlL6w5luCT3LvweUFrILqEKIWIxj6iozAaaiFp7OYA&rqlang=cn&rsv_dl=ts_0&rsv_sug3=10&rsv_enter=1&rsv_sug1=2&rsv_sug7=100&rsv_sug2=1&rsv_btype=i&prefixsug=chanzhon&rsp=0&inputT=6616&rsv_sug4=6616

2. https://www.google.com/search?sxsrf=AOaemvJJfTCvadwTPmNdHZVMjka8h-o_CQ:1641266246255&q=%E7%BC%A0%E4%B8%AD%E8%AF%B4%E7%A6%85&tbm=isch&chips=q:%E7%BC%A0+%E4%B8%AD+%E8%AF%B4+%E7%A6%85,online_chips:%E5%9B%BE%E4%B9%A6:VsxR8C1V6tw%3D&usg=AI4_-kRDg-4IMhjiQBDzMbS7ss7aXMxgAA&sa=X&ved=2ahUKEwixwNuckZfl AhVaSGwGHV40BF0QgIoDKAF6BAgnEBQ&biw=1463&bih=759&dpr=2.63

hub represents a defined price range over a specific period. Stock prices follow a cyclic process of hub creation, extension, expansion, and termination—though not every cycle includes extension and expansion, creation and termination always occur. The author introduced three key chart patterns: the reversal pattern, the first following pattern, and the second following pattern. These patterns are considered objective, repeatable, and strong trading signals. HS establishes criteria for defining the HS line segment, which helps identify emerging hubs in price movements. The structured nature of HS chart patterns makes them well-suited for feature extraction and integration into deep learning models for training.

Figure 2.1.1 illustrates examples of the three HS patterns, each represented by its characteristic HS line segments. The selected examples appear successively on the same stock—Amazon Company—during the period from July 2010 to June 2011. All three patterns shown are buyer-type signals.

In the top panel of Figure 2.1.1, we observe a reversal pattern. Here, two hubs are moving downward, and the most recent line segment exits the second hub with less strength compared to the segment that entered it. This weakening movement triggers a buy signal according to the HS rule. After the reversal point, the subsequent segment re-enters the overlapping region of the second hub but fails to make a new low. This behavior forms what is known as the first following pattern, as shown in the middle panel.

The bottom panel of Figure 2.1.1 presents the second following pattern. In this case, the leaving segment emerges from the most recent hub, and the returning segment does not re-enter the overlapping part of that hub—this distinguishes it as the second following pattern.

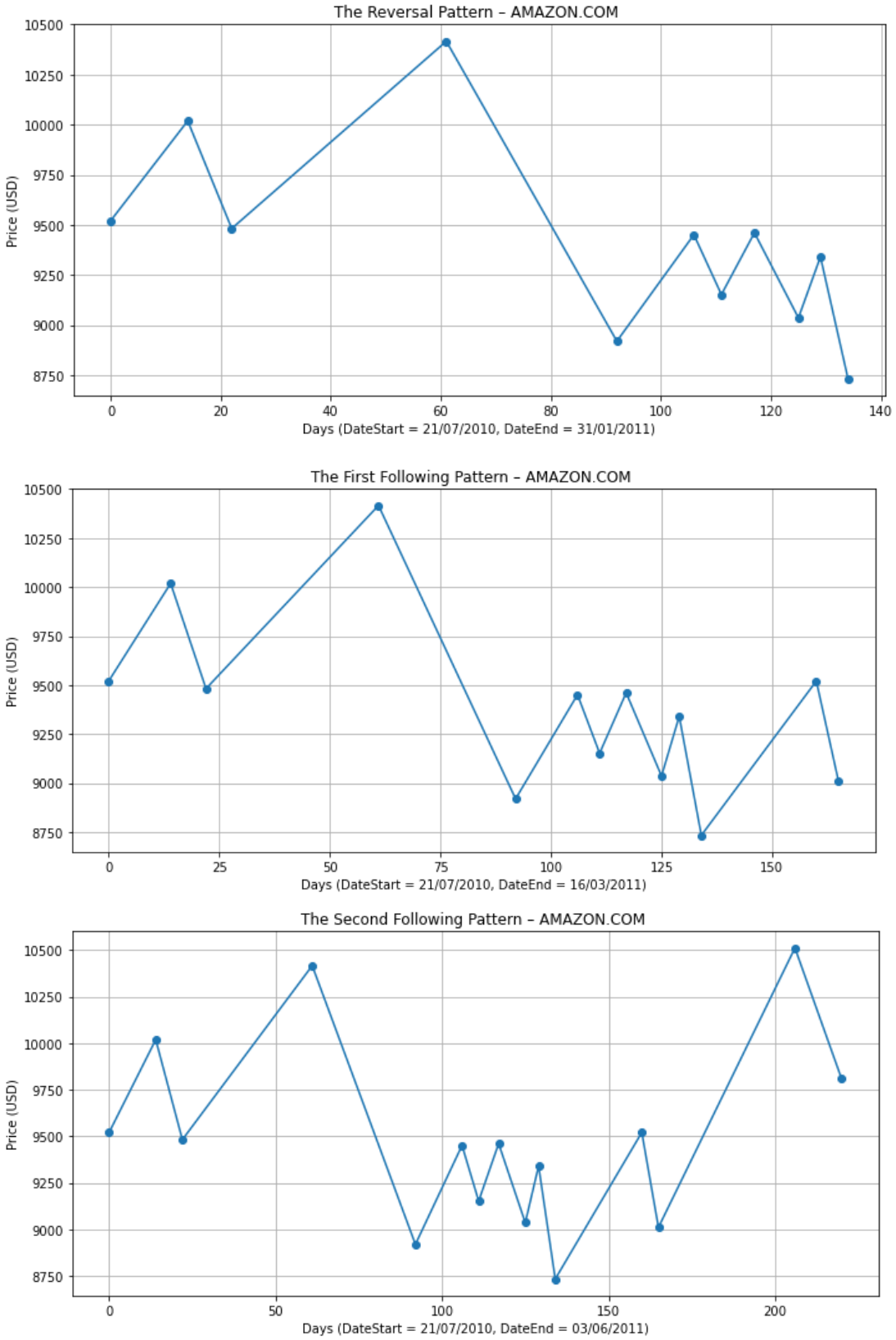

Figure 2.1.1 Examples of the reversal, first following, and second following patterns

Deep learning (DL) is an advanced technique of machine learning (ML) that achieves great power and flexibility by learning to represent the world as a nested hierarchy of concepts (Goodfellow, 2016). DL has been widely adopted in diverse fields, including autonomous driving for cars and drones, computer vision, audiovisual recognition, image processing, classification, and prediction tasks (e.g., Chai and Li, 2019; Chai and Ngai, 2020; Partheepan, Sanati, and Hassan, 2023; Botezatu, Burlacu, and Orhei, 2024; Zhang, Yang, Chen, Zhang, Leng, and Zhao, 2024). Recently, deep learning (DL) has gained interest in stock forecasting. While many studies show DL models using time-series price data can be profitable, few prove they outperform the buy-and-hold (BH) benchmark (Sezer et al., 2020; Li & Bastos, 2020; Nti et al., 2020; Huang et al., 2020). A key challenge is that traditional time-series data (Open, High, Low, Close) is often noisy, chaotic, volatile, and non-stationary, making it ineffective in capturing historical trends.

In our DL models, inputs focus on features of HS chart patterns instead of the time series data, which are expected to better detect the price direction. Recent researches support this expectation. Gudelek, Boluk, and Ozbayoglu (2017), Sezer, Ozbayoglu, and Dogdu (2018), and Sezer and Ozbayoglu (2019) transform time series price data to 2-D images and feed them to the Convolutional Neural Network (CNN) model. Choi and Renelle (2019) suggest a “deep learning price momentum” (DLPM) model that takes a one-year history of total returns as input. Furthermore, Jiang, Kelly, and Xiu (2021) directly feed images of U.S. stock price patterns into CNNs, including sets of Japanese candlesticks, moving average lines, and volume bars, expecting the models to learn chart patterns that are most predictive of future returns.

Our input differs from Jiang, Kelly, and Xiu (2021) in two key ways. First, while they use fixed-length price chart images, our DL input captures HS chart patterns with variable sizes, reflecting their full development. Second, their CNN model extracts predictive patterns from raw market data, whereas we focus on HS price plots, incorporating volatility, trading sentiment, and volume. We propose that stock prices are shaped by investor expectations and trading behavior, with hubs acting as proxies for consensus on intrinsic value. Thus, HS price charts serve as more effective DL inputs.

Despite the popularity of HS among Chinese practitioners, it has received little attention from academia, possibly because of the difficulty in quantitatively describing HS chart patterns. In this study, we programmatically recognize HS chart patterns using the Python programming language and take on the challenge of empirically testing the validity and profitability of this trading strategy. And more importantly, we apply DL to such a prediction model and see how much we can further enhance the profitability of this trading strategy.

DL has obvious advantages in computational processing power and data availability. The works in Athey and Imbens (2019), and Coulombe, Leroux, Stevanovic, and Surprenant (2020) center on a comparison between DL and traditional econometric methods. They argue that DL provides a toolset that allows for flexible functional forms, which can be used to capture non-linearities between the dependent variable (output) and independent variables (input). Unlike statistical techniques, DLs are data-driven and do not need any explicit assumption on the model between the input and output. Furthermore, techniques about cross-validation and regularization are normally used in DLs, which are helpful for DLs to

achieve high predictive power. Therefore, we integrate DL into HS and expect DL can effectively find the non-linear relationship between HS chart patterns and future prices of stocks.

The HSDL strategy (the HS strategy integrated with the DL model) is expected to be superior to the pure HS strategy. First, the pure HS can only detect and analyze HS chart patterns roughly in a qualitative manner and cannot distinguish the subtle but important differences among the same kind of chart patterns appearing at varying times and levels. This means HS likely ignores some information that is valuable in discovering the relationship between the chart pattern and the future price direction. In contrast, the HSDL strategy can collect more details of chart patterns and utilize DL computer power to learn this information to enhance predictability. For example, in our study, DL models automatically collect the information of a certain HS chart pattern, say the second following, including the highest and lowest prices of the hub and the hub core, and the strength of the leaving and returning HS line segments. Hence, based on the quantitative analysis of features about details of chart patterns, the HSDL strategy can be reasonably expected to perform better than the pure HS. Second, the DL model extends its input to adding additional information, such as the price volatility, trading volume, and sentiment index, standing in a multiple-dimensional rather than a 2-dimensional perspective, with only plane geometry pictures. The value of this extension can be observed in the prediction study of the DL model using time series data. Abe and Nakayama (2018) and Zhang and Tan (2018) not only use the raw time series price (OCHLV: open, close, high, low, and transaction volume) data, they also add fundamental data of the company, information relevant to the market sentiment, and features, such as the

Sharpe-ratio, standard deviations, and fund alphas, into DL models. Their empirical results have shown that extended information can enhance the accuracy in the prediction of price movements. Our study adds to this stream of literature and also shows that DL models generate predictions with higher accuracy.

Our study contributes to the literature in the following ways. First, we combine deep learning (DL) techniques, applied to cross-sectional data, with the Hub Strategy (HS)—a novel chart pattern analysis approach—to create the HSDL trading strategy. This combination could allow traders and analysts to perform the method of chart analysis in a quantitative and multiple-dimensional manner and cause a new type of input, a chart-based feature of HS chart patterns. In addition, the method that we developed for automatically identifying chart patterns also can contribute to the study of traditional chart pattern analysis. Furthermore, we conduct a series of rigorous tests checking the robustness in the higher predictability of HSDL, compared to the BH and pure HS rules.

In this study, we estimate the annualized average returns of HSDL on the buy and sell strategies applied to the US S&P 500 index constituent stocks. We compare these conditional mean returns of HSDLs to both the unconditional BH mean returns and those from the associated pure HS strategies. Our empirical results show the buy rules of HSDLs, reaching a high of about 25% annualized average return, not only fairly outperform the 9.6% BH benchmark return but also show a significant improvement over the pure HS rules, which produce returns in the range of 15.93% to 19.58%. This conclusion holds when we use risk-adjusted measurements, such as the Sharpe ratio and Fama and French (1995) five-factor alphas. In addition, the average returns of all sell HSDL strategies are negative, which means

these sell- rules can achieve gains if we short stocks on these sell signals.

We also evaluate the effect of transaction costs on HSDL strategies. Financial academics and practitioners are interested in transaction costs because the net gains from trading rules can be considerably impacted by such costs, and for some high-frequency rules, returns could change from positive to negative due to these costs. Roll (1984) states that transaction costs mainly include brokerage commissions and the bid-ask spread, and the corresponding measurement is painful but direct. In our test, we first suppose there is a 0.5% fixed brokerage commission imposed on a round-trip trade (opening and closing the position) to calculate the average return, and then consider the effect of the ask-bid spread on the returns. We find that the effect of the price spread is negligible for the estimation of the average returns of HSDLs due to the low frequency of signals generated by HSDLs. The mean values of break-even transaction costs (BETCs) from all HSDL strategies range from 155 to as high as 566 basis points. Jones (2002) calculated the one-way transaction costs (half spread +NYSE commission) that are 100 basis points in 1972 and 20 in 2000, respectively. Our results are much larger than these values, even after imposing the 0.5% fixed cost, which consistently suggests that the profitability of HSDLs is reliable.

Academics are concerned that the trading rule is susceptible to periods related to the 'crash risk'. In papers from Marshall, Nguyen, Visaltanachoti (2017), Daniel and Moskowitz (2011), and Santa-Clara (2014), they all analyze this issue in the application of the moving average and time series momentum strategies. We use the method of Marshall, Nguyen, Visaltanachoti (2017) to examine the returns of trading rules on the 10-worst days when the market has huge drops. We show that the HSDL strategy is not susceptible to crash risk. On

these worst days, returns from the HSDL strategy are much greater than those of the benchmark, the buy-and-hold (BH). If the market allows for shorting stocks, our HSDL strategy can even realize positive returns these days.

Another issue we are interested in is whether the performance of HSDL models depends on the firm-specific or trading-specific properties of stocks. Hence, we sort all sample stocks based on the market value, illiquidity, and volatility of returns, respectively, and create five sorted groups to observe the relationship patterns between the performance of HSDLs and these variables. We find there is no obvious pattern in the sorted groups of the market value and illiquidity, but the volatility of returns can affect the performance of HSDLs. The higher the volatility, the higher the annualized average returns for HSDLs.

Our multiple testing that if HSDL strategies applied to all sample stocks are superior to benchmarks, pure HS and BH rules, comes with the data snooping concern. An unguarded use of single-inference procedures could result in “false discoveries”. When we conduct more and more tests, an increasing number of significances is attributed to chance. Once the “false discovery proportion” (FDP) reaches 100%, the significance of the individual tests is of no use. Benjamini and Hochberg (1995) develop a false discovery rate controlling approach. This approach implements multiple testing by controlling the false discovery rate (FDR) and thus can avoid the data snooping issue. Giglio, Liao, and Xiu (2021) add the screening step and show a solid improvement. We employ the same method in our study. We set six null hypotheses to test if HSDL buyers can beat the associated pure HS and BH rules and if HSDL sellers can gain profits under the false discovery controlling approach. Our results show that the average returns of HSDL buyers are significantly greater than those of the BH rule on a

large fraction of stocks, HSDL sellers realize the profits on the majority of sample stocks, and the fractions, which rejects the null hypothesis that the HSDL return is less than or equal to the HS return, reflect an improvement of HSDLs over HSs.

Lastly, we use the recent asset pricing model with 153 risk factors to evaluate the capability of the HSDL strategy in capturing significantly positive alphas. These factors have been suggested by some researchers (e.g., Cochrane, 2011; Harvey, Liu, and Zhu, 2015; McLean and Pontiff, 2016) to explain the cross-section of expected stock returns. Feng, Giglio, and Xiu (2020) summarize these hundreds of risk factors covering most of the variables in the literature, including the two new factors introduced by Fama and French (2015) and Hou, Xue, and Zhang (2015), and the intermediary-based factors from He, Zhang, Kelly, and Manela (2017). Due to the potential multicollinearity among 153 factors, we use a two-stage regression to implement the alpha study. Lasso regression is well-suited to resolve this intercorrelation problem. Therefore, in the first stage, following Feng, Giglio, and Xiu (2020), we use Lasso regression to remove redundant variables. In the second stage, we calculate the alphas and their t-statistics through ordinary least squares regression on the selected factors by Lasso. We adopt the HSDL synthetic rules to construct switching strategies for all sample stocks, taking long or short positions when HSDL signals appear, and allocating to risk-free assets when no HSDL signals are present. Our results demonstrate that HSDL strategies can gain positive alphas on the majority of stocks.

The remainder of this essay proceeds as follows: Section 2.2 summarizes the literature review. Section 2.3 develops our methodology relative to HS and HSDL prediction models. Section 2.4 analyzes empirical results. Section 2.5 concludes the paper.

2.2 Literature Review

Chart analysis is a fundamental technique in technical analysis (TA) used to identify recurring price patterns in financial markets. These patterns, derived from historical price movements, assist traders and investors in predicting potential future price trends. The core idea behind chart analysis is that market psychology and investor behavior often repeat, creating recognizable formations on price charts. Studies by Smidt (1965), Schwager (1993, 1995), and Lo and Hasanhodzic (2009) surveyed investors and found that more than half of amateur traders use charts to identify trends, while many top traders and fund managers also rely on chart analysis. In recent decades, chart analysis has gained increasing attention from academics. Nazário, Silva, Sobreiro, and Kimura (2017) note that, despite its complexity, the stock chart is a vital tool in technical analysis and is often displayed on professional traders' multi-monitor workstations.

Edwards and Magee (1996), Schwager (1995), and Pring (2002) analyze the profitability of various chart patterns, which are typically named after their shapes in bar charts. These include gaps, spikes, flags, pennants, wedges, triangles, head-and-shoulders formations, and various tops and bottoms. Chang and Osler (1999) assess the performance of the head-and-shoulders pattern in currency markets and find that trading rules based on this pattern yielded statistically significant annual returns of approximately 13% for the German mark and 19% for the Japanese yen. These returns are higher than those generated by the buy-and-hold (BH) strategy applied to the S&P 500 index over the same period. Similarly, Sant'Anna and Vancin (2015) demonstrate that the head-and-shoulders pattern could also generate positive returns in the Brazilian stock market. Savin, G., Weller, P., & Zvingelis, J.

(2007) also find that the pattern of ‘head-and-shoulders’ had power to predict the price direction.

Lo and Hasanhodzic (2000) expand the analysis to a broader range of chart patterns in stock markets, including head-and-shoulders, inverse head-and-shoulders, broadening tops and bottoms, triangle tops and bottoms, rectangle tops and bottoms, and double tops and bottoms. For NYSE/AMEX stocks, goodness-of-fit test results indicate that five out of the ten chart patterns showed statistically significant differences in the relative frequencies of returns compared to unconditional returns. In contrast, all ten patterns are found to be statistically significant for Nasdaq stocks. Additionally, Osler and Chang (1995) and Boainain and Pereira (2009) explore the potential of developing conditional trading strategies using the Head-and-Shoulders chart pattern. Their findings suggest that this pattern can generate profitable strategies based solely on historical data, demonstrating its effectiveness in predicting market movements.

When reviewing the literature on chart pattern analysis, particularly regarding line chart patterns, significantly fewer studies are found compared to research on technical indicators such as moving averages, Moving Average Convergence Divergence (MACD), and the Relative Strength Index (RSI). This discrepancy may be due to the ease with which technical indicators can be mathematically expressed, whereas identifying and interpreting chart patterns like "Head-and-Shoulders" and "Flags" often involve a high degree of subjectivity.

Despite this challenge, considerable efforts have been made over the years to automate the identification of technical patterns, leading to the development of various

methodologies. For example, the template-based (TB) approach (e.g., Arévalo, García, Guijarro, & Peris, 2017) measures the temporal and amplitude distance between a chart pattern template and a query subsequence. Similarly, the rule-based (RB) approach (e.g., Tsinaslanidis and Guijarro, 2020) classifies patterns using a predefined set of rules. These automated techniques enable broader empirical investigations into chart pattern analysis.

Specifically, Arévalo, García, Guijarro, and Peris (2017) develop and validate a trading rule based on flag pattern recognition, applying it to 91,309 intraday observations of the DJIA index. Their approach demonstrates improved profitability compared to the buy-and-hold strategy. Similarly, Tsinaslanidis and Guijarro (2021) apply automating chart pattern recognition and design a trading system that incorporates any generic pattern proven to be profitable in the past. Their findings indicate that, on average, the proposed system outperforms the market index in terms of mean-variance efficiency. Moreover, Norasaed and Siriborvornratanakul (2024) propose an algorithm that extracts candlestick wave patterns and applies transformer modeling. Their results indicate that the test set achieved a 19.8% return, outperforming the MACD signal, which yielded 12.5%.

Other researchers have focused on studying bar chart patterns, particularly candlestick patterns. Marshall, Young, and Rose (2006) conducted the first comprehensive study on candlestick charting, the oldest known form of technical analysis. Candlestick analysis is a short-term timing technique that generates signals based on the relationship between open, high, low, and close prices. Their findings indicate that candlestick trading strategies do not provide value for Dow Jones Industrial Average (DJIA) stocks.

In recent years, many researchers have integrated candlestick patterns with machine

learning techniques to improve price direction predictions. Andriyanto (2020) develop a model using candlestick chart patterns and a convolutional neural network (CNN) algorithm to predict market movements. Jearanaitanakij and Passaya (2019) propose a method for short-term trading prediction based on convolutional networks and candlestick patterns, achieving better results in training time and accuracy compared to ResNet-18. Udagawa (2019) introduces a model with six parameters designed to retrieve similar candlestick patterns, enhancing stock price prediction accuracy. Thammakesorn and Sornil (2019) propose a method for generating stock trading strategies based on characteristics proven to detect patterns in candlestick charts reliably. Kusuma et al. (2019) explored the predictive capabilities of candlestick charts in the Taiwanese and Indonesian stock markets. Using datasets and advanced machine learning techniques—such as CNNs, residual networks, and visual geometry group (VGG) networks—they aimed to evaluate and improve predictive accuracy.

Deep Learning (DL) is a type of Artificial Neural Network (ANN) with multiple layers, and the key advantage of DL models is automatically extracting the good features of input data by a general-purpose learning procedure. The popular models of DL include Deep Multi-Layer Perceptron (DMLP), Recurrent Neural Network (RNN) with Long Short-Term Memory (LSTM), Convolutional Neural Networks (CNNs), and Reinforcement Learning (RL).

Currently, DL is booming with the developments of computer hardware and software, such as graphics processing units (GPUs) and Tensor Flow software, and thus DL has been applied comprehensively in computer visions, audio-visual recognition, image processing,

classification, and prediction tasks (e.g., Collobert, Weston, Bottou, Karlen, Kavukcuoglu, and Kuksa, 2011; Guo, Liu, Oerlemans, Lao, Wu, and Lew, 2016; Chai and Li, 2019; Chai and Ngai 2020; Partheepan, Sanati, and Hassan, 2023; Botezatu, Burlacu, and Orhei, 2024; Zhang, Yang, Chen, Zhang, Leng, and Zhao, 2024). The price prediction in finance is no exception, and the vast number of papers relevant to DL models have been published in the appropriate conferences and journals.

Using deep learning (DL) models with time series data for price prediction is increasingly popular in financial markets. Research focuses on applying DL to various asset classes or developing advanced models. Recent studies use DL to forecast price movements in stocks, indices, commodities, bonds, currencies, and cryptocurrencies. Dezsi and Nistor (2016) applied DL to predict stock trends in Romanian markets, while Das, Mokashi, and Culkin (2018) used neural networks for S&P 500 price prediction. Other studies focus on commodities, such as Lasheras et al. (2015), who used an Elman RNN for COMEX copper prices, and Chen, He, and Tso (2017), who applied DL to WTI crude oil. DL-based forecasting of forex and cryptocurrency prices is also widely explored (e.g., Zhang, Shen, and Zhao, 2014; Shen, Chao, and Zhao, 2015).

Researchers take two main approaches to improving deep learning (DL) model performance: model-oriented and feature-oriented. The model-oriented approach focuses on optimizing model selection and combining multiple models for better accuracy. Studies compare various DL models, such as LSTM, RNN, CNN, and MLP, for financial forecasting (Hiransha et al., 2018; Selvin et al., 2017). Some researchers build hybrid models, like ensemble networks (Yang et al., 2017) or LSTM-MLP models (Zhou, 2018). Sezer, Gudelek,

and Ozbayoglu (2020) found LSTM was most used for price prediction, while CNN and DMLP were favored for classification tasks.

The feature-oriented approach optimizes input data to enhance model performance. Early studies used raw price data (OCHLV) (Chen et al., 2015; Dezsi & Nistor, 2016), while later research incorporated technical indicators, macroeconomic data, and sentiment analysis. For instance, Zhang and Tan (2018) added technical indicators to LSTM models for Chinese stock predictions, and Fan, Xue, and Yao (2017) used macroeconomic data like GDP and unemployment rates. Matsubara, Akita, and Uehara (2018) applied a deep neural generative model with news data, outperforming traditional classifiers.

The feature-oriented approach introduces innovative methods to improve financial time series predictions, addressing their complex and volatile nature. Researchers enhance feature quality by transforming time series into alternative data formats or replacing them with new inputs.

Some studies convert price data into 2D images for CNN models. Gudelek, Boluk, and Ozbayoglu (2017) generated 2D images from technical analysis values, achieving 72% accuracy and outperforming the buy-and-hold (BH) strategy. Sezer and Ozbayoglu (2018, 2019) applied similar methods with prices and indicators, showing superior results for stocks and ETFs. Others refine input features. Shen and Shafiq (2020) customized feature engineering in an LSTM-based model, reaching 93% accuracy and surpassing previous models. Choi and Renelle (2019) introduced a deep learning price momentum (DLPM) model that improved market-neutral annual returns from -0.49% to +1.93%, outperforming traditional momentum strategies.

Some researchers replace time series data entirely. Patil et al. (2020) framed stock prediction as a graph problem, reducing errors compared to traditional techniques. Jiang, Kelly, and Xiu (2021) used raw price, volume, and indicator chart images as CNN inputs, demonstrating superior performance over benchmarks like moving averages and time series momentum. In addition, we previously introduced price pattern recognition in this section, where researchers utilized recognized patterns as inputs for deep learning, aligning with this development direction.

Our work aims to integrate deep learning (DL) techniques with chart analysis, specifically the Hub Strategy (HS). We utilize Deep Multi-Layer Perceptron (DMLP) to learn structural features of HS patterns that indicate potential future price directions. This approach is expected to significantly enhance the predictive power of DL models and improve profitability for HSDL trading rules, which combine HS patterns with DL-based predictions.

2.3 Methodology

2.3.1 Hub Strategy (HS)

Hub Strategy (HS) is a method of price chart pattern analysis to detect future price movements. HS claims that the stock's price movement is the hub movement process that is a cyclic process of creation, extension, expansion, and termination of a series of hubs, where a hub is defined as a price range of a stock over a certain period. HS generates HS line segments based on a set of rules. Using HS line segments can characterize the hub and help traders identify the HS trading patterns. There are three principal trading chart patterns in HS, including the reversal, the first following, and the second following chart patterns. The

reversal pattern tends to find the location of the stock price where this pattern indicates that the movement of a prior hub could be finished, and the evolution of a potentially new hub will initiate. Specifically, if more than two hubs move in the down (up) direction, the leaving line segment leaves the latest hub weakly, compared to the counterpart entering into this hub. The endpoint of the leaving line segment is a reversal buy (sell) point, and this related chart pattern is the reversal pattern. The reversal pattern predicts that there will be more than one hub that will appear in the opposite direction relative to the prior hubs. Investors can put a long or short position on the market and then capture a trend. The second following pattern could occur after the price moves forward two HS line segments from the reversal point and tells traders the price of the stock is still located in the early or middle period of the creation phase of a potential hub. The second following pattern is related to the one where a line segment leaves the current hub, and then the price in the next line segment does not return to the core region of the hub. This pattern forecasts that a new hub will initiate or just stay in the creation phase.

In our view, the hub can be a proxy for the consensus on the intrinsic value of the stock. Thus, the HS trading pattern has to do with how participants come to a consensus on the value of a stock over a specific period.

In the following sub-section, we will introduce the concepts and three principal trading chart patterns in detail.

2.3.1.1 Concepts in HS

HL Chart

HL comes from the traditional chart, OHLC (open, high, low, close), and hides the prices of

open and close in charts. Thus, the HL chart only consists of a vertical line, with the high being the period's high and the low of the vertical line being the period's low. The high and low of the HL chart i are denoted as $HL_{H,i}$ and $HL_{L,i}$, respectively.

Here are examples of HL charts:

Figure 2.3.1 Examples of HL Charts and the Covering Relationship

Covering Relationship Between Adjacent HL Charts

Given HL chart i and i+1, we call these two charts having a covering relationship if one of the following conditions is satisfied (See, the right part of Figure 2.3.1).

Conditions of the covering relationship:

- $HL_{H,i} >= HL_{H,i+1}$ and $HL_{L,i} <= HL_{L,i+1}$. (The third and fourth HL in the right part of Figure 2.3.1)
- $HL_{H,i} <= HL_{H,i+1}$ and $HL_{L,i} >= HL_{L,i+1}$. (The first and second HL in the right part of Figure 2.3.1)

Clean HL Chart

A clean HL chart is related to the situation where there is no covering relationship between this HL chart and its adjoining HL charts. In HS, we first deal with the covering

relationship between neighbor HL charts and generate a sequence of clean HL charts for analysis. If a pair of HL charts has the covering relationship, we combine them to produce a new HL chart, replacing the originals. The combination rule is described as follows:

- First, find the HL chart i which has a covering relationship with i+1 rather than i-1. We will join the HL chart i and i+1 as a new HL chart, denoted as HL_{i_i+1}.
- Second, determine the combination direction. If $HL_{H,i-1} > HL_{H,i}$ and $HL_{L,i-1} > HL_{L,i}$, we adopt a downward combination (the top part in Figure 2.3.2); and if $HL_{H,i-1} < HL_{H,i}$ and $HL_{L,i-1} < HL_{L,i,}$, the upward combination is taken (the bottom part in Figure 2.3.2).
- The downward combination: the high and low of the new HL chart, HL_{i_i+1} are:

  $HL_{H,\, i\_i+1} = \min (HL_{H,i}, HL_{H,i+1})$

  $HL_{L,\, i\_i+1} = \min (HL_{L,i}, HL_{L,i+1})$
- The upward combination: the high and low of the new HL chart, HL_{i_i+1} are:

  $HL_{H,\, i\_i+1} = \max (HL_{H,i}, HL_{H,i+1})$

  $HL_{L,\, i\_i+1} = \max (HL_{L,i}, HL_{L,i+1})$
- If the new HL_{i_i+1} still has a covering relationship with HL_{i+2}, repeat the above steps to create a new one, denoted as $HL_{i\_i+1\_i+2}$ to replace the HL_{i_i+1}, and so on, up to the new chart having no covering relationship with the adjacent chart.
- If there is a covering relationship between the first and second HL charts in our whole sequential HL chart, we can choose either a down or an upward combination.

Note: In the following context, HL charts all refer to the clean HL charts.

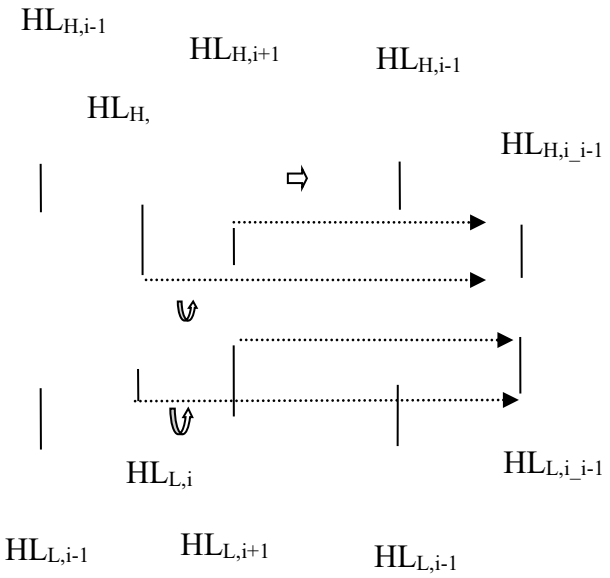

Figure. 2.3.2 The Rule of Dealing with Containing Relationship

Line Segment

Before explaining the line segment, we need to introduce the concept of the top and bottom structure based on HL charts.

Top structure and bottom structure

A top structure conditional on:

$HL_{H,i} > HL_{H,i-1}$, $HL_{L,i} > HL_{L,i-1}$, $HL_{H,i} > HL_{H,i+1}$, and $HL_{L,i} > HL_{L,i+1}$.

A bottom structure conditional on:

$HL_{H,i} < HL_{H,i-1}$, $HL_{L,i} < HL_{L,i-1}$, $HL_{H,i} < HL_{H,i+1}$, and $HL_{L,i} < HL_{L,i+1}$.

Here is an example:

Figure 2.3.3 Examples of Top and Bottom Structure

Line Segment Description

A line segment is a line starting from a high of an HL chart in a top structure (a low of an HL chart in a bottom structure) and ending a low of an HL chart in a bottom structure (a high of an HL chart in a top structure) and with one more additional HL charts between the top structure and bottom structure, and the start and end in this line segment is either the highest price or lowest price. The line segment starting at the low of the HL chart in the bottom structure is called the upward line segment, while the line segment starting at the high of the HL chart is the downward line segment.

The definition of the line segment reflects the clear and solid change in price direction. A line segment always experiences a formation of top and bottom structure and includes at least more than one additional HL chart, representing a consistent move along this existing direction. Thus, the HS line segment is expected to truly indicate the upward or downward movement of prices.

Requirements of line segments:

- Each line segment contains a minimum of 5 HL charts
- The price of the starting point of line segment i is denoted as $LS_{S,i}$, and $LS_{E,i}$ is the

price of the ending point of line segment i. If $LS_{S,i} < LS_{E,i}$, the line segment i is an upward line segment; and vice versa.

- A downward (an upward) line segment is always followed by an upward (a down) line segment.
- The starting and ending of a line segment always correspond to either the highest or lowest price of this line segment.

Here are examples of line segments:

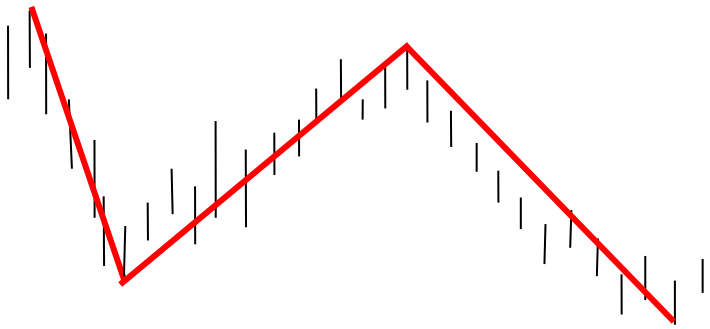

Figure 2.3.4 Examples of Line Segments

Close Point of the Line Segment

The close point of an ongoing line segment i is the one where we can determine the end point of this line segment or the start point of the next line segment at the earliest time. The end of the ongoing line segment i or the start point of the next line segment i+1 is the same point. The end point of each line segment can only be determined at a lagged time until the corresponding close point appears. The price of the close point is denoted as $LS_{C,i}$, and the

close point meets the following requirements.

(1) If $LS_{S,i} < LS_{E,i}$, $LS_{C,i}$ is the lowest price between the end point of the line segment i and its close point, and if $LS_{S,i} > LS_{E,i}$, $LS_{C,i}$ is the highest price between the end point of the line segment i and its close point.

(2) There is a minimum of five HL charts from the end of line segment i to the corresponding close point, including the end and close point.

(3) The close point is the nearest point to the end of line segment i, compared to other ones satisfying the requirements of (1) and (2) above.

Here is an example showing several close points ($LS_{C,i}$).

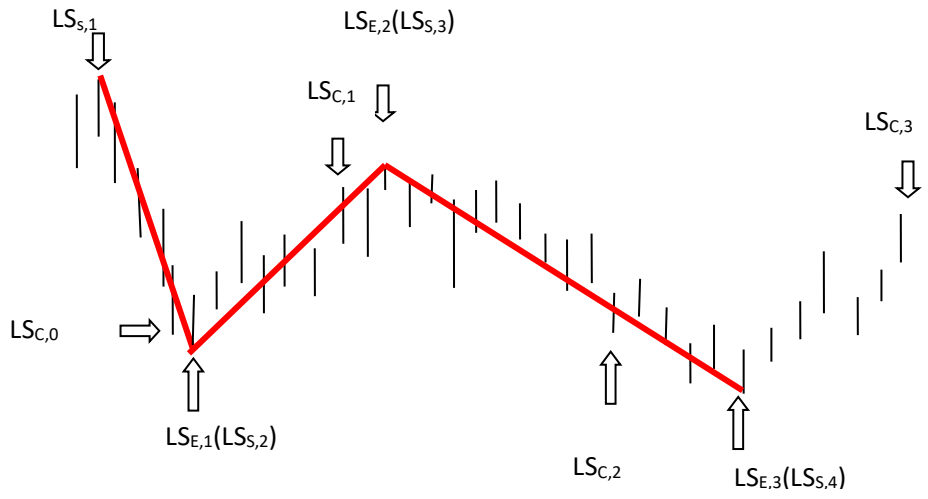

Figure 2.3.5 An Example of the Close Points

Using daily HL Charts, we apply programs to identify line segments and plot the HS price chart for Dow Jones Industrial Index in a specific time (see Figure 2.3.6).

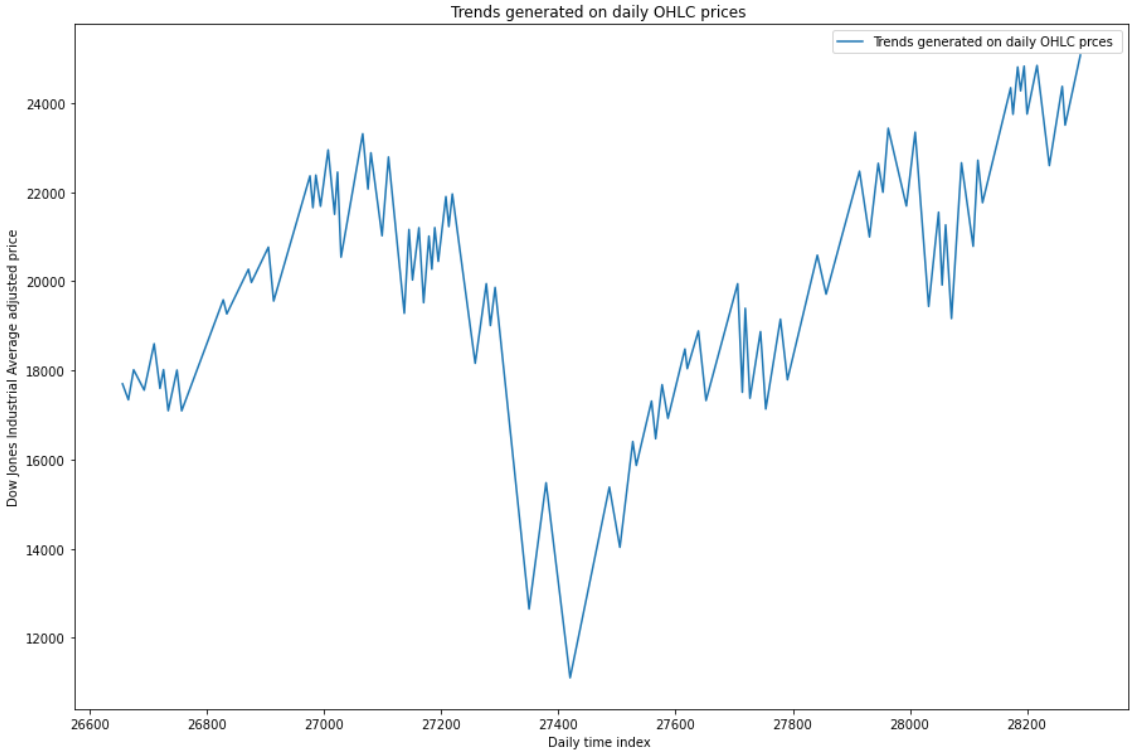

Figure 2.3.6 An Example of the Price Chart in HS

Hub

A hub is defined as a price range of a stock over a certain period. From our perspective, the hub can be precisely viewed as an asymptotical proxy for the participant consensus on the intrinsic value of a stock. A hub is composed of at least three line-segments that have a common overlapping part in prices. In practice, if prices are moving upward in a whole perspective, the hub is initiated into a downward segment and vice versa.

Hub features:

- $Hub_{HH:}$ the highest price in the hub (the HH point in Figure 2.3.7).
- Hub_{LL}: the lowest price in the hub (the LL point in Figure 2.3.7).

- Hub_H: the high price of the overlapping part in the hub (the H point in Figure 2.3.7).
- Hub_L: the low price of the overlapping part in the hub (the L point in Figure 2.3.7).

Here are two examples:

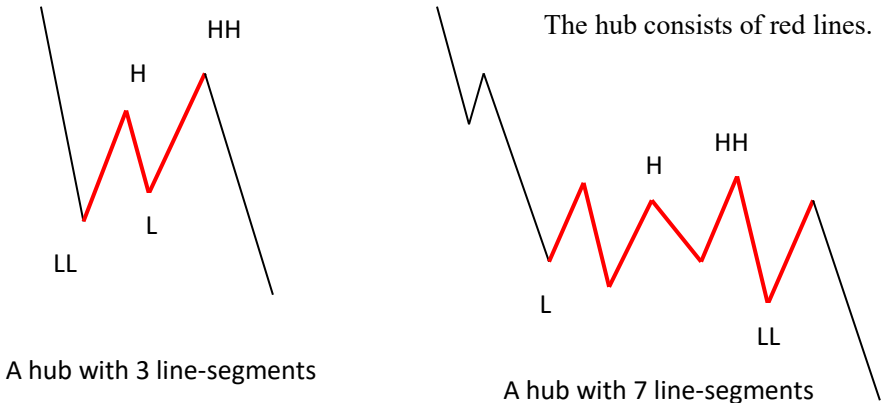

Figure 2.3.7 Examples of Hubs

The type of movement in HS

HS only involves two types of movement, trend (down or upward) and fluctuation. A trend movement includes two or more hubs arraying in the same direction and without overlapping between hubs, while the fluctuation movement only contains one hub.

Here are examples of trends and fluctuations:

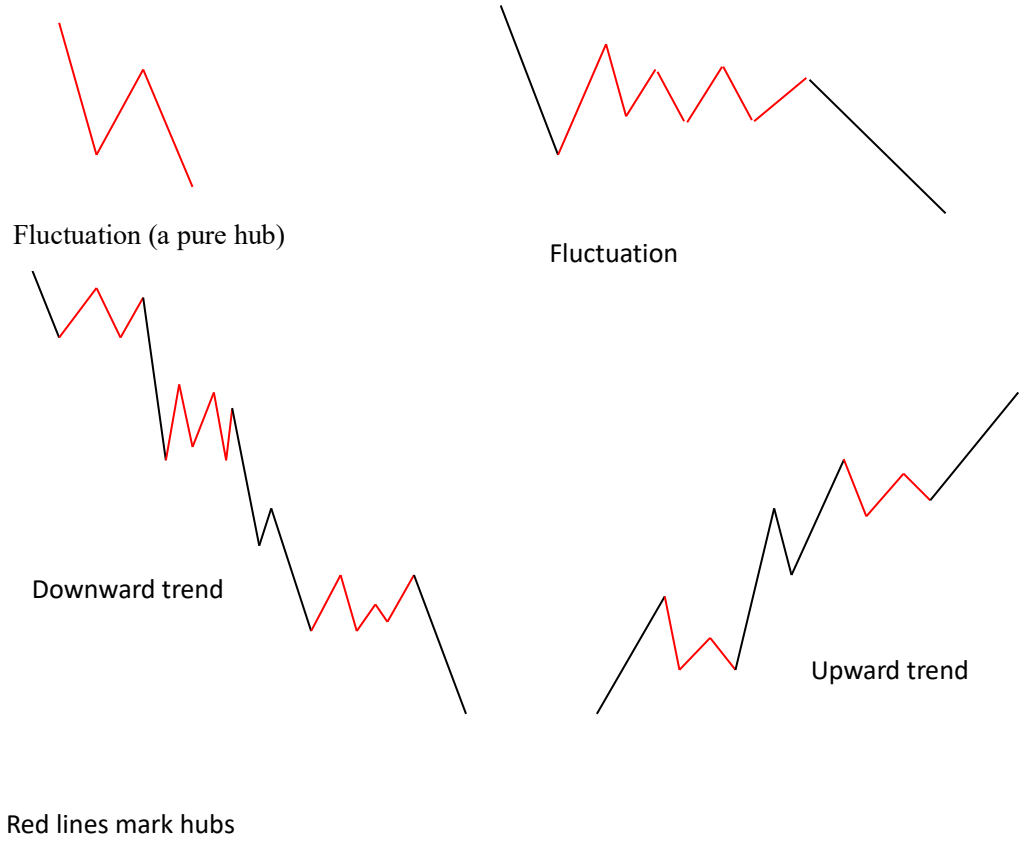

Figure 2.3.8 Types of Price Movements in HS

2.3.1.2 Trading Chart Patterns in HS

According to specific locations of stock prices associated with prior hubs, HS presents a series of trading chart patterns. For the sake of brevity, our research only introduces three kinds of trading rules, called the reversal pattern, first following pattern, and second following pattern. In general, HS rules attempt to search the stage where either the prior hub will be terminated, or a potential new hub is just initializing or in the early phase of the hub

creation. When traders invest funds in this specific time where the price is moving in the junction part of the prior hub and next hub, they can hold the position till to either the extension phase in the opposite side related to your entering point or a simple fixed sub-trend step. In this study, we adopt holding the number of three line-segments.

Reversal Chart Pattern

The reversal chart pattern aims to detect the reversal point of the ongoing HS trend and trigger the trading signal. To better describe this pattern, we first show the example of the reversal chart pattern and then explain it.

Here is an example of the reversal chart pattern:

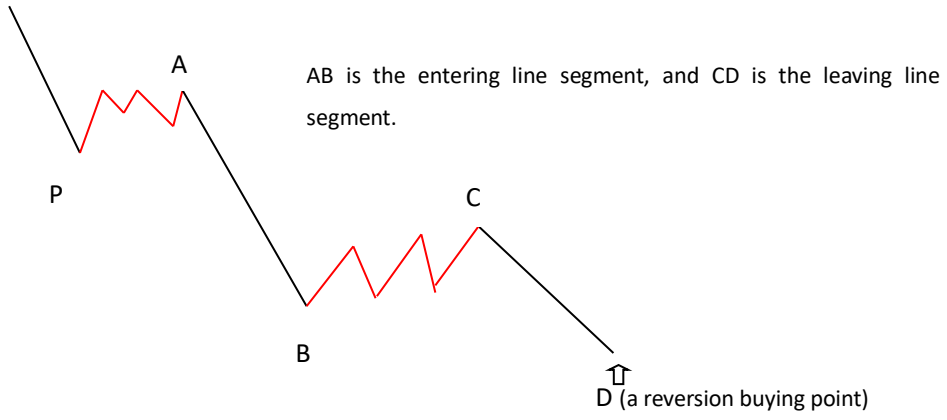

Figure 2.3.9 Reversal Chart Pattern

According to the above graph (Figure 2.3.9), we can understand the following:

- Hub1 (up) consists of five line-segments from the point P to A.
- Hub2 (down) consists of five line-segments from point B to C.
- AB is the entering line segment, and CD is the leaving line segment.

Explanation of the reversal chart pattern

Two or more hubs (e.g., Hub1 and Hub2) move in the down (up) direction, the leaving line segment (CD) leaves the latest hub (Hub2) weakly, compared to the counterpart (AB) entering into this hub. The end point (D) of the leaving line segment is the **potential** reversal point, and thus it is a buy (sell) point. The potential reversal point could imply the termination of the prior hub. This kind of chart pattern is called the reversal chart pattern if the following condition1 and condition2 are both satisfied.

Condition1: $\text{Strength}_{\text{entering}} > \text{Strength}_{\text{leaving}}$.

Condition2(buy): $LS_{E,CD} < \text{Hub2}_{LL}$.

Condition2(sell): $LS_{E,CD} > \text{Hub2}_{HH}$.

$\text{Strength}_{\text{entering}}$(height, slope) and $\text{Strength}_{\text{leaving}}$ (height, slope) are the strength functions for the entering line segment (AB) and leaving line segment (CD), respectively. $LS_{E,CD}$ is the price of the end point of the leaving line segment CD. Hub2_{LL} and Hub2_{HH} are the lowest and highest prices in the Hub2, respectively.

First Following Chart Pattern

Here is an example of the first following chart pattern:

The following information can be collected based on the graph above (Figure 2.3.10).

- Hub1(up) consists of five-line segments from point P to A.
- Hub2 (down) consists of five-line segments from point B to C.
- CD is the leaving line segment, denoted as line segment i-2.
- EF is the re-leaving line segment, denoted as line segment i.

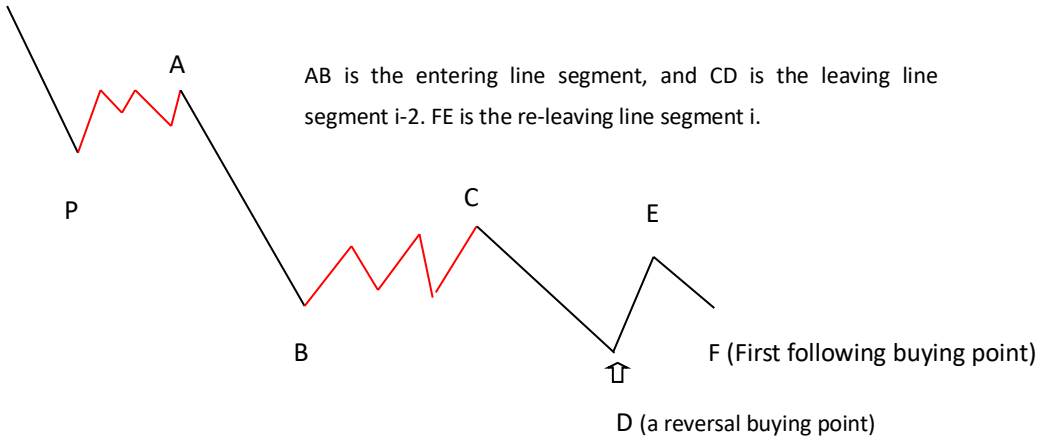

Figure 2.3.10 First Following Chart Pattern

Explanation of the first following chart pattern

Once the reversal point is determined, which means the line segment i-1 bounces back into the overlapping part of the prior hub, the end point (F) of re-leaving line segment i is the first following point if this leaving does not create a new low for the buy rule or new high for the sell rule, compared to the price of end point (D) the line segment i-2. The first following trading point reflects an early stage of the creation phase of a new hub. This chart pattern described above is the first following chart pattern if the following condition is met.

Condition(buy): $LS_{E,i} > LS_{E,i-2}$ and $LS_{S,i} > Hub_L$ (2.3.1.1),

Condition(sell): $LS_{E,i} < LS_{E,i-2}$ and $LS_{S,i} < Hub_H$ (2.3.1.2),

where $LS_{E,i}$ and $LS_{E,i-2}$ are the price of the end point of the re-leaving line segment i and the line segment i-2, respectively; and the Hub_H and Hub_L are the high and low price of overlapping parts in the latest hub (Hub2), respectively.

Second Following Chart Pattern

Here are examples of the second following chart patterns:

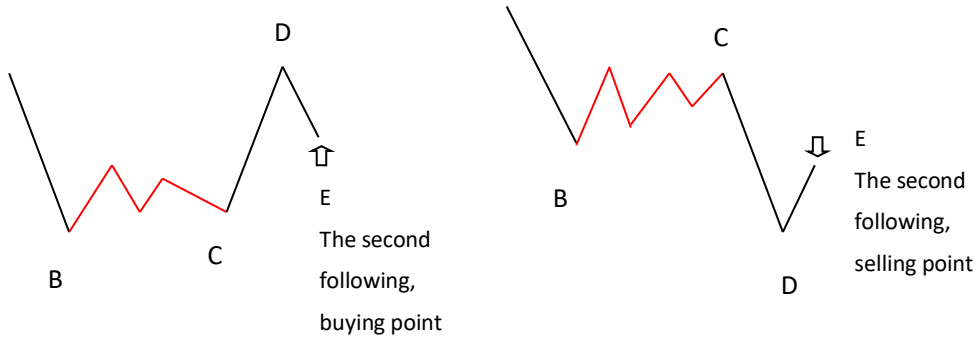

Figure 2.3.11 Second Following Chart Patterns

Based on examples of the Second following chart patterns (Figure 2.3.11), we can

include the following information.

- Hub3 (left) consists of four line segments from point B to C.
- Hub4 (right) consists of five line-segments from point B to C.
- CD is the leaving line segment, denoted as line segment i-1.
- DE is the returning line segment, denoted as line segment i.

Explanation of the second following chart pattern

The line segment i-1 is leaving the latest hub (e.g., Hub3 and Hub4) and creates a new high or new low, compared to the hub, and the next line segment i could not return to the overlapping part of the hub (Hub3 or Hub4). This end point (E) of the returning line segment i is the trading point of the second following pattern. The relevant chart pattern is the second following chart pattern if the condition (2.3.1.3 or 2.3.1.4) is satisfied.

Condition (buy): $LS_{E,i} > Hub_H$ and $LS_{S,i} > Hub_{HH}$, (2.3.1.3)

Condition (sell): $LS_{E,i} < Hub_L$ and $LS_{S,i} < Hub_{LL}$, (2.3.1.4)

where $LS_{E,i}$ and $LS_{S,i}$ are the price of the end and start point of the line segment i, respectively, and Hub_H, Hub_{HH}, $Hub_{L,}$ and Hub_{LL} are features for the latest hub (Hub3 or Hub4).

The trading point of the second following pattern claims that the prior hub has been terminated and the new hub will be formed or is at the stage of creation.

2.3.1.3 HS Trading Rules in Our Research

We will apply four trading rules of HS on individual stocks, including the reversal, first

following, second following, and synthetic rule. The synthetic rule is the combination of these three rules. Considering buy and sell, eight rules of HS will be conducted, which are denoted as follows.

- The reversal rule for buying, denoted as HS_Re_B.
- The first following rule for buying, denoted as HS_FF_B.
- The second following rule for buying, denoted as HS_SF_B.
- The synthetic rules for buying, denoted as HS_Syn_B.
- The reversal rule for selling, denoted as HS_Re_S.
- The first following rule for selling, denoted as HS_FF_S.
- The second following rule for selling, denoted as HS_SF_S.
- The synthetic rules for selling, denoted as HS_Syn_S.

Specifically, the trading steps are described as follows:

1. We generate programs on the Python platform to detect the corresponding chart patterns, including the reversion, first following, and second following patterns.
2. Once the ‘end’ point of the last line segment, segment i, appears, we put a long or short position, based on these buy chart patterns or sell chart patterns. The trading price is the close price of the corresponding original HL chart.
3. After generating positions, all rules a fixed-length to close the trade position. The fixed length refers to a three-line-segment step. For instance, if we buy a stock at the ‘end’ point of line segment i, we will close this buy position in the ‘end’ point of line segment i+3.

Here, we need to explain the ‘end’ of line segment i and the “end” point of line segment i+3. In reality, one cannot determine the end point of an ongoing line segment at first-hand time other than by delaying several HL charts. Hence, we have to use a proxy to replace the real end of the line segment i and i+3. For maximizing profits, we aim to find the proxy that is as close to the end of the line segment as possible. In HS, there are many ways to find these proxy points, and some are very complex. Our study selects a simple way to locate the proxy point. That is, we choose the close point (See, *Close point of the line segment* in 3.1.1.4), $LS_{C,\ i-1}$, of the line segment i-1 as the proxy for the end point of line segment i. Likewise, the close point, $LS_{C,\ i+2}$, of the line segment i+2 is adopted as the proxy for the end point of the line segment i+3.

In this research, the close point represents the end point of the last line segment in a given chart, at which we identify trading patterns and decide to move in or out of markets.

We choose the fixed length that is a three-line-segment step. The reason could be explained as follows. Firstly, we need to have odd numbers, keeping the direction with the desired one. Secondly, our choice to use three consecutive line segments is inspired by a philosophical idea from a classic Chinese historical text: “一鼓作气，再而衰，三而竭”, which roughly means "The first effort brings momentum, the second weakens, and the third exhausts it." Therefore, selecting three steps allows us to capture the first and second stages of the overall momentum for the reversal pattern, while the first and second following patterns can ride on the second and third stages process of the whole momentum, just before it begins to fade. In the latter cases, the first stage of momentum is the signal stage for the first and second following patterns.

2.3.2 Deep Learning (DL) Models

Deep Multi-Layer Perceptron (DMLP) is a fundamental deep learning model designed to learn complex, nonlinear relationships between inputs and outputs. It is composed of layers of interconnected nodes (neurons) and is well-suited for structured, tabular data. In prediction tasks such as stock price direction, DMLP can effectively learn from financial indicators or engineered features, making it a flexible and powerful tool for classification and regression problems. A DMLP is a supervised feed-forward neural network and consists of an input layer, multiple hidden layers, and an output layer. The input layer matches the feature space so that there are as many input neurons as predictors. The output layer is either a classification or regression layer to match the output space. There is no standard formula to calculate the number of hidden layers and nodes needed in the hidden layer (Wang and Sun (1996)). DMLP is a computational structure, and DMLP models can perform well in the field of stock price prediction, with sufficient training data and appropriate training strategies.

Here is an example of DMLP:

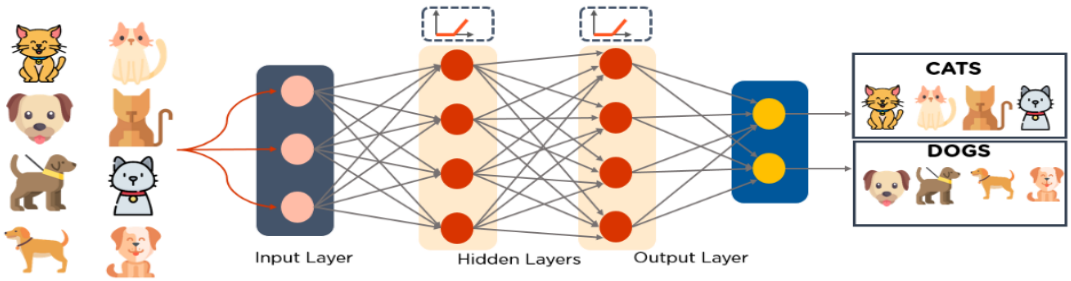

Figure 2.3.12 DMLP Architecture

This DMLP model (Figure 2.3.12) has one input layer, 2 hidden layers, and one output

layer. Input layer: Receives features (e.g., the related structure elements, transaction volume, sentiment, etc.). Hidden layers: Multiple fully connected layers with activation functions like ReLU. Output layer: Produces final predictions (e.g., price direction, probability of a trend). The dots in Figure 2.3.12 are the neurons. Each neuron within a layer could compute a linear combination of its inputs, followed by a nonlinear transformation. In general, the function of 2.3.2.1 can be used to describe the model of DMLP.

$$y_i = f\left(\sum_{j=1}^{N} w_{ij}x_j + b_i\right), \qquad i, j = 1, 2, \ldots, N \qquad (2.3.2.1)$$

The principle of this model is based on summarizing the input variables with weights. The sum is activated by the activation function. Input variables are transferred from node j to node i: x_j means independent variables, and w_{ij} means weights. The function f is referred to as the activation function. In this study, we use an activation function as the ReLU function (Glorot, Bordes, and Bengio, 2011). y_i is a target of the line segment rise (1) and fall (0). X_j is the feature related to the chart pattern.

2.3.2.1 Inputs and Outputs of DMLPs

Inputs

Features related to HS chart patterns are extracted by the program and fed as the input to DMLPs. Features are mainly involved in the structure of chart patterns, price volatility, trading volume, and trading sentiment. Incorporating additional information could further enhance the accuracy of DL predictions. Several studies support this idea—Zhang and Tan

(2018) improved LSTM predictions for Chinese stocks by adding technical indicators, Fan, Xue, and Yao (2017) integrated macroeconomic data like GDP and unemployment rates, and Matsubara, Akita, and Uehara (2018) leveraged news data in a deep neural generative model, outperforming traditional classifiers. The data relative to features is the cross-sectional data.

For describing the trading sentiment, we create our sentiment index values for the start and end points of line segments, based on comparing the prices of the start or end points of the line segments with the values of the moving average of past prices (see the Table 2.3.1: Sentiment Index Value), and then generate the hub sentiment index.

Table 2.3.1 Sentiment Index Value

$LS_{E,i}$ and $LS_{S,i+1}$ are the end point of line segment i and the start point of line segment i+1, respectively. MV(50), MV(150), and MV(250) are the moving average of past 50, 150, and 250 days' prices.

If $LS_{E,i}$ or $LS_{S,i+1}$	Sentiment Index Value
>= MV(50) and >=MV(150) and >=MV(250)	3
<MV(50) and >=MV(150) and >=MV(250)	2
>=MV(50) and <MV(150) and >=MV(250)	1
<MV(50) and <MV(150) and >=MV(250)	0
>=MV(50) and >=MV(150) and <MV(250)	-1
<MV(50) and >=MV(150) and <MV(250)	-2
>=MV(50) and <MV(150) and <MV(250)	-3
<MV(50) and <MV(150) and <MV(250)	-4

For a start or end point of a line segment, we first calculate the values of MV(50), MV(150), and MV(250) and then determine the value of the sentiment index for this point according to the criteria described in Table 2.3.1. The hub sentiment index value is the average of all endpoints of line segments within the hub. We use the following table to show features related to our chart patterns (Table 2.3.2).

Table 2.3.2 Features for the Reversal, First Following, and Second Following Chart Patterns

This table lists the features of all HS chart patterns. Hub_{HH} and Hub_{LL} are the highest and lowest prices of the latest hub, respectively. Hub_H and Hub_L are the highest and lowest prices of the overlapping part of the latest hub, respectively. $LS_{E,CD}$ and $LS_{S,CD}$ are the price of the end point and start point of the line segment CD, respectively. $LS_{E,DE}$ and $LS_{S,DE}$ are the price of the end point and start point of the line segment DE, respectively. $LS_{E,EF}$ and $LS_{S,EF}$ is the price of the end point and start point of the line segment EF, respectively. Std is the standard deviation on log daily returns.

Feature	Description
Panel A: features for all HS chart patterns	
HubThick	$\log(Hub_{HH}/Hub_{LL})$, Hub (hereafter) refers to the latest hub.
HubCoreThick	$\log(Hub_H/Hub_L)$
LeavingHeight	$\log(LS_{E,CD}/LS_{S,CD})$, CD is the leaving line segment (see Fig. 2.3.9, 2.3.10, and 2.3.11)
LeavingSlope	LeavingHight/lasting days for the leaving line segment (CD)
LeavingStd/HubStd	Std of log daily return for CD/Std of the counterpart for the Hub
LeavingT/HubT	The average of daily turnover for CD/the counterpart for the Hub
Leaving/HubHorL	$\log(LS_{E,CD}/Hub_H)$ for buy patterns; $\log(LS_{E,CD}/Hub_L)$ for sell patterns
HubSentiment	Average of Hub sentiment index
Panel B: supplement features for the first and second following patterns	
ReturningHeight	$\log(LS_{E,DE}/LS_{S,DE})$, DE is the returning line segment (see Fig. 2.3. 10, and 2.3.11)
ReturningSlope	ReturningHight /lasting days for the returning line segment (DE)
Returning/leaving	ReturningHight /LeavingHeight
Returning/HubHorL	$\log(LS_{E,DE}/Hub_H)$ for buy patterns; $\log(LS_{E,DE}/Hub_L)$ for sell patterns.
ReturningStd/HubStd	Std of log daily return for DE/Std of the counterpart for the Hub
ReturningT/HubT	The average daily turnover for DE/the counterpart for the Hub
Panel C: supplement features for the reversal pattern	
LeavingH/Comparison	LeavingHight/ the height of the comparison (AB, see Fig. 2.3.9)
LeavingS/Comparison	LeavingSlope/the slop of comparison (AB)
LeavingSenti/HubS	The sentiment index of the leaving line segment (CD)/HubSentiment
Panel D: supplement features for the first following pattern	
ReleavingHieght	$\log(LS_{E,EF}/LS_{S,EF})$, FE is the re-leaving line segment (see Fig. 2.3.10)
ReleavingSlope	ReleavingHight /lasting days for the re-leaving line segment (EF)
ReleavingH/ReturningH	ReleavinghHeight/RetruningHeight
ReleavingStd/HubStd	Std of log daily return for EF/Std of the counterpart for the Hub
ReleavingT/HubT	The average daily turnover for EF/the counterpart for the Hub
ReleavingSenti/HubS	The sentiment index of the re-leaving line segment (EF)/HubSentiment
Panel E: supplement features for the second following pattern	
ReturningSenti/HubS	The sentiment index of the returning line segment (DE)/HubSentiment

Output

The label for the output is either zero or one. Suppose the index of the last line segment in an HS chart pattern is i, and our DL model predicts the direction of the end point price of the line segment i+3, which is proxied by the close point of the segment i+2. When the price of the end point of the line segment is larger than the price of the end point of line segment i-1, this means prices experience an upward trend and the output is labeled as one, and vice versa.

2.3.2.2 Construction of DMLPs

TensorFlow, which is the premier open-source deep learning framework developed and maintained by Google, is utilized to build and train our six DMLP models for six chart patterns, including the buy and sell reversal patterns, the buy and sell first following patterns, and the buy and sell second following patterns.

The activation function in our models is the ReLU function (Glorot, Bordes, and Bengio, 2011), and Adam (Kingma and Ba (2014)) for the optimization algorithm. Additional hyperparameters, such as the learning rate and momentum, varying across models.

Our Deep Multi-Layer Perceptron (DMLP) model integrates deeper architecture, residual connections, and advanced optimization. It begins with an input layer of varying neurons (matched to different chart patterns) representing features such as HS pattern structure, price volatility, volume, and sentiment. This is followed by five hidden layers with 256, 128, 64, 32, and 16 neurons, using ReLU and Swish activations. Batch normalization is applied after the second layer, and a residual connection links the input to the third layer to retain key information. To prevent overfitting, we apply 30% dropout after each layer and L2 regularization ($\lambda = 0.001$). The output layer uses a sigmoid activation for classification. The model uses Adam optimization with a learning rate scheduler, Huber loss for outlier

robustness, a batch size of 64, and early stopping after 20 stagnant validation epochs. For classification tasks, we use a binary cross-entropy loss function.

We split our sample period into the in-sample for training and out-of-sample sub-periods for testing. The in-sample period starts on the date of 31/12/1992 and ends on the date of 29/12/2006, while the out-sample period ranges from 03/01/2007 to 31/12/2020.

3.2.3 Integration of HS and DL

We combine the DMLP models and HS trading rules and propose the HSDL trading strategies. HSDL trading rules can be explained in the following steps.

- Detect HS chart patterns through the programs.
- Apply the corresponding DMLP model to predict the direction of price movements.
- If DL model predicts the direction we desire, place the trading position and keep the fixed length, a 3 line-segments step.

Considering buy and sell, eight rules of HSDL will be conducted, which are displayed as follows.

- The reversal rule for buying, denoted as HSDL_Re_B.
- The first following rule for buying, denoted as HSDL_FF_B.
- The second following rule for buying, denoted as HSDL_SF_B.
- The synthetic rules for buying, denoted as HSDL_Syn_B.
- The reversal rule for selling, denoted as HSDL_Re_S.
- The first following rule for selling, denoted as HSDL_FF_S.
- The second following rule for selling, denoted as HSDL_SF_S.
- The synthetic rules for selling, denoted as HSDL_Syn_S.

2.4 Results

2.4.1 Data and Sample Periods

In this study, we focus on examining HSDL strategies on individual stocks, in particular, the US S&P 500 index constituent stocks, and comparing them with the benchmarks, HS strategies and the buy-and-hold (BH) rule. Taking into account that daily OHLC prices of these stocks were announced in 1992, we set our sample period to be 31/12/1992 to 31/12/2020. We collect the price, price index, market value, trading volume, as well as S&P 500 index information from the DataStream database. We split our sample period into the in-sample for training and out-of-sample sub-periods for testing. The in-sample period starts on the date of 31/12/1992 and ends on the date of 29/12/2006, while the out-sample period ranges from 03/01/2007 to 31/12/2020. The prices of OHLC are adjusted by the price index that considers dividend reinvestment. The line segments are generated on daily HL price charts, and HS chart patterns are detected by using this daily level of HS line segments.

2.4.2 Overall Performance of HSDL Strategies

First, we trained and tested six DL models, corresponding to three buy rules and three sell rules, using the in-sample data sets. Then the performance of DL models is examined over the out-of-sample data. The prediction performance of DLs is documented in Appendix A.1. The buy DL models can achieve the precision of prediction from 71.11% to 77.54%, while the prediction precisions for the two sell DL models are close to 70% (see Table A.1.1

in Appendices).

After the processing mentioned above, we integrate our DL prediction models into HS strategies and generate a novel trading strategy, the HSDL strategy. The purpose is to use trained DL models to forecast the future direction of assets' prices and make final trade decisions when programs detect the HS chart patterns.

Then, we directly apply HSDL strategies to 505 individual stocks over an out-of-sample period. Following the method used by Brock, Lakonishok, and BeBaron (1992), the average returns conditional on HSDL buy and sell rules are recorded and compared with those conditional on Benchmarks (HS rules and the buy-and-hold (BH) rule). Table 2.1 reports the statistics of annualized average returns from HSDL, HS, and BH strategies. For the one-way transaction cost for stocks, Lynch and Balduzzi (2000) suggest a value of 25 basis points, and Glabadanidis (2015) uses 50 basis points. In all our experiments, we assume the one-way trading costs are 50 basis points for stocks.

The results in Table 2.1 are intriguing. HSDL buy strategies, HSDL_Re_B, HSDL_FF_B, HSDL_FS_B, and HSDL_Sy_B, can realize the annualized average returns of 25.75%, 23.26%, 25.46%, and 25.26%, respectively (see, column 2 of Table 2.1), which are considerably greater than the annualized return of the BH strategy (9.64%), and these differences between the HSLD strategies and the BH strategy vary from 13.62% to 16.11% and are all significant at the level of 1% (see, the column 3 of Table 2.1). Furthermore, the HSDLs have around 6% improvements in the annualized returns, compared to their relevant HS strategies, and the differences between HSDLs and HSs are still significant at the level of 1% (see, column 4 of Table 2.1). Likewise, HSDL sell strategies, including HSDL_Re_S,

HSDL_FF_S, HSDL_FS_S, and HSDL_Sy_S, show their excellent performance with fairly

Table 2.1 Statistics of Annualized Average Returns of HSDL Strategies

Table 2.1 presents the summary statistics of annualized average returns for HSDLs, HSs, and the BH, applied to 505 individual stocks in out-of-sample. The sample period ranges from 03/01/2007 to 31/12/2020. μ is the annualized average return. *, **, and *** denote statistically significantly different from the equivalent metric at the 10%, 5%, and 1% levels, respectively. Re, FF, and SF refer to the reversal, first following, and second following rules, respectively. B and S respectively represent buy and sell rules. Syn is the brevity of synthetic rule and means the combination rule of these three rules. For example, HSDL_SF_B refers to the second following buy rule of the HSDL strategy. DiffBH refers to the difference in the annualized average return between HSDL and BH, and DiffHS represents the difference in the annualized average return between HSDL and HS.

	Mean	DiffBH	DiffHS	Q3	Median	Q1
Panel A: Annualized average return μ (%) of HSDL strategies						
HSDL_Re_B	25.75	16.11***	6.17***	40.35	20.66	7.01
HSDL_FF_B	23.26	13.62***	7.33***	38.93	17.91	-1.24
HSDL_SF_B	25.46	15.82***	6.62***	34.80	19.41	7.72
HSDL_Syn_B	25.26	15.62***	6.67***	33.37	22.22	11.06
HSDL_Re_S	-4.72	-14.36***	-3.60**	10.32	-4.41	-20.81
HSDL_FF_S	-11.67	-21.31***	-5.53***	3.26	-13.44	-32.36
HSDL_SF_S	-14.04	-23.68***	-6.82***	7.31	-15.39	-39.66
HSDL_Syn_S	-10.13	-19.77***	-5.55***	4.13	-9.30	-30.09
Panel B: Annualized average return μ (%) of the BH strategy						
BH	9.64			14.83	8.48	3.25
Panel C: Annualized average return μ (%) of HS strategies						
HS_Re_B	19.58	9.94***		33.80	16.71	3.15
HS_FF_B	15.93	6.29***		32.82	12.52	-7.97
HS_SF_B	18.83	9.20***		28.37	14.55	3.42
HS_Syn_B	18.58	8.95***		28.44	16.99	6.16
HS_Re_S	-1.12	-10.76***		10.62	-5.37	-19.88
HS_FF_S	-6.14	-15.78***		7.50	-7.57	-21.58
HS_SF_S	-7.22	-16.86***		7.35	-8.61	-24.07
HS_Syn_S	-4.58	-14.22***		4.83	-4.30	-19.18

negative average returns ranging from -4.72% to -14.04% (see column 2), which means the HSDL sell rules can receive profits if putting short positions. HSDL selling strategies again significantly outperform three corresponding HS sell strategies at the level of 1% (see column 4).

We visualized the annualized average return from HSDL, HS, and BH trading rules in Figure 2.1.

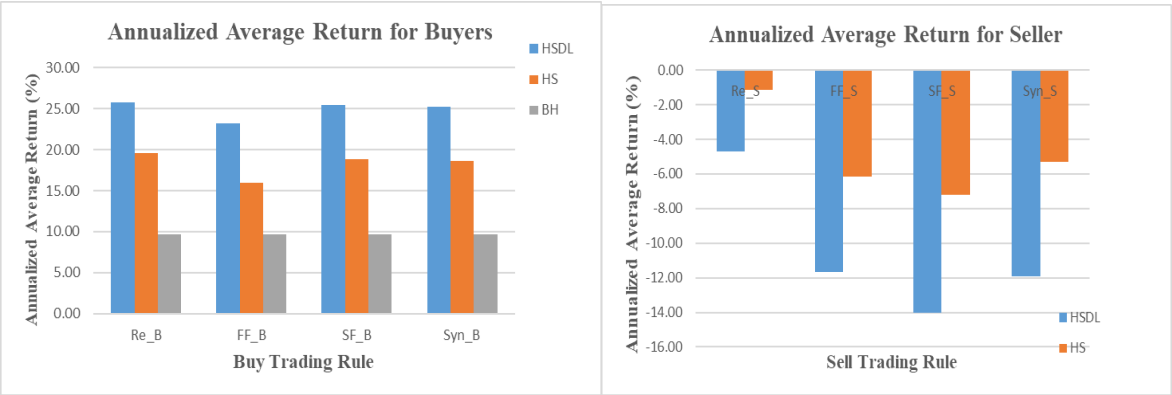

Figure 2.1 Annualized Average Return for Trading Rules

This figure displays the annualized average return from HSDL, HS, and BH trading rules. Re, FF, and SF refer to the reversal, first following, and second following rules, respectively. B and S respectively represent buy and sell rules. Syn is the brevity of synthetic rule and means the combination rule of these three rules. For example, SF_B refers to the second following buy rule, and FF_S refers to the first following sell rule.

Table 2.2 presents the statistics of annualized standard deviations of returns from HSDL, HS, and BH strategies. The average annualized standard deviations (Std) of HSDL_FF_B, HSDL_SF_B, and HSDL_Sy_B are significantly lower than that of the BH strategy at the 1% level, and the rest one, HSDL_Re_B, has a slightly larger Std, compared to the BH (see columns 2 and 3, in Table 2.2). Average standard deviations of all HSDL buy strategies are smaller than those of HS counterparts, and differences between HSDLs and HSs are significant at the level of 1% (see rows 3 to 6 in column 4 of Table 2.2). These results suggest that HSDL buy strategies are less risky than HSs and BH. In contrast, the majority of HSDL sell strategies increase the values of the average standard deviations, compared to HSs and BH, which is interesting. In stock markets, the standard deviations of returns normally tend to be larger when markets move in the down direction. We can conclude that this case reflects the ability to capture the downward trends for our HSDL selling strategies.

Table 2.2 Statistics of the Annualized Standard Deviation of Returns

Table 2.2 presents the summary statistics of annualized standard deviations of returns for HSDLs, HSs, and the BH, applied to 505 individual stocks in out-of-sample. The sample period ranges from 03/01/2007 to 31/12/2020. σ is the annualized standard deviation of the return. *, **, and *** denote statistically significantly different to the equivalent metric at the 10%, 5%, and 1% levels, respectively. Re, FF, and SF refer to the reversal, first following, and second following rules, respectively. B and S respectively represent buy and sell rules. Syn is the brevity of synthetic rule and means the combination rule of these three rules. For example, HSDL_SF_B refers to the second following buy rule of the HSDL strategy. DiffBH refers to the difference of annualized standard deviations of returns between HSDL and BH, and DiffHS represents the difference of annualized standard deviations of returns between HSDL and HS.

	Mean	DiffBH	DiffHS	Q3	Median	Q1
Panel A: Annualized standard deviation σ (%) of HSDL strategies						
HSDL_Re_B	36.78	1.19	-3.44***	42.90	35.45	28.72
HSDL_FF_B	33.48	-2.11***	-4.08***	38.47	31.38	25.29
HSDL_SF_B	28.08	-7.51***	-3.14***	32.20	27.23	
HSDL_Syn_B	33.40	-2.19***	-3.74***	38.12	32.41	26.75
HSDL_Re_S	27.94	-7.65***	0.55	32.57	25.28	19.91
HSDL_FF_S	34.46	-1.13	3.22***	41.56	30.50	24.08
HSDL_SF_S	47.23	11.64***	5.91***	57.40	42.81	31.11
HSDL_Syn_S	42.85	7.26***	5.30***	51.78	38.13	30.23
Panel B: Annualized standard deviation σ (%) of the BH strategy						
BH	35.59			40.96	34.01	22.71
Panel C: Annualized standard deviation σ (%) of HS strategies						
HS_Re_B	40.23	4.64***		47.21	37.20	29.95
HS_FF_B	37.56	1.97**		43.71	33.92	26.98
HS_SF_B	31.22	-4.37**		35.67	28.78	24.67
HS_Syn_B	37.14	1.55**		43.38	34.72	28.61
HS_Re_S	27.39	-8.20***		31.46	24.79	19.90
HS_FF_S	31.24	-4.35***		36.84	28.47	22.84
HS_SF_S	41.32	5.73***		49.24	38.47	30.04
HS_Syn_S	37.54	1.95**		44.59	35.15	28.25

Table 2.3 reports the Sharpe ratio for HSDL, HS, and BH strategies as an alternative performance measure. The results are consistent with those from Table 2.1. For HSDL buying strategies, HSDL_Re_B, HSDL_FF_B, HSDL_FS_B, and HSDL_Sy_B, can achieve the average Sharpe ratio of 0.688, 0.658, 0.797, and 0.695, respectively (see column 2 in Table 2.3), which are around three times as large as that of the BH strategy (0.221), and the differences between the HSDLs and BH are all significant at the level of 1% (see columns 3 and 4). The more meaningful results show that the Sharpe ratios of HSDL buying stategies

Table 2.3 Statistics of Annualized Sharpe Ratio of HSDL Strategies

Table 2.3 presents the summary statistics of annualized Sharpe ratio for HSDLs, HSs, and the BH, applied to 505 individual stocks in out-of-sample. The sample period ranges from 03/01/2007 to 31/12/2020. We use the formula of Sharp ratio: $(abs(\mu)-r_f)/\sigma$, where μ is the average annualized return, r_f is the average risk-free turn, and σ is the standard deviation of returns. *, **, and *** denote statistically significantly different to the equivalent metric at the 10%, 5%, and 1% levels, respectively. Re, FF, and SF refer to the reversal, first following, and second following rules, respectively. B and S respectively represent buy and sell rules. Syn is the brevity of synthetic rule and means the combination rule of these three rules. For example, HSDL_SF_B refers to the second following buy rule of the HSDL strategy. DiffBH refers to the difference in annualized Sharpe ratios between HSDL and BH, and DiffHS represents the difference of annualized Sharpe ratios between HSDL and HS.

	Mean	DiffBH	DiffHS	Q3	Median	Q1
Panel A: Annualized Sharpe ratio of HSDL strategies						
HSDL_Re_B	0.688	0.47***	0.08	1.136	0.550	0.120
HSDL_FF_B	0.658	0.44***	0.09	1.228	0.545	-0.092
HSDL_SF_B	0.797	0.58***	0.17***	1.228	0.704	0.211
HSDL_Syn_B	0.695	0.47***	0.13**	1.040	0.634	0.284
HSDL_Re_S	0.058	-0.16**	0.11*	0.751	0.212	-0.474
HSDL_FF_S	0.169	-0.05	0.04	0.824	0.390	-0.193
HSDL_SF_S	0.090	-0.13**	0.02	0.743	0.323	-0.234
HSDL_Syn_S	0.124	-0.10	0.00	0.633	0.240	-0.183
Panel B: Annualized Sharpe ratio of the BH strategy						
BH	0.221			0.386	0.193	0.028
Panel C: Annualized Sharpe ratio of HS strategies						
HS_Re_B	0.605	0.38***		1.021	0.456	0.092
HS_FF_B	0.570	0.35***		1.108	0.373	-0.205
HS_SF_B	0.625	0.40***		0.999	0.524	0.117
HS_Syn_B	0.567	0.35***		0.895	0.499	0.178
HS_Re_S	-0.049	-0.27***		0.718	0.097	-0.542
HS_FF_S	0.124	-0.10		0.652	0.278	-0.286
HS_SF_S	0.072	-0.15**		0.547	0.188	-0.294
HS_Syn_S	0.125	-0.10		0.484	0.191	-0.159

are all greater than those of relevant HS buy strategies, and two of four differences are significant at the 5% level. All HSDL selling strategies realize positive Sharpe ratios, but only the Sharpe ratio of HSDL_Re_S is significantly larger than that of HS at the level of 10%.

We visualized the annualized Sharpe ratio from HSDL, HS, and BH trading rules in Figure 2.2.

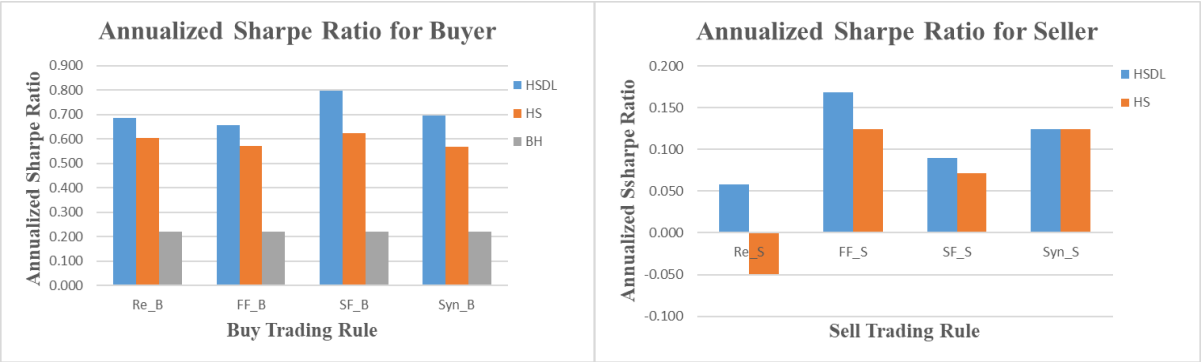

Figure 2.2 Annualized Sharpe Ratio for Trading Rules

This figure displays the annualized Sharpe ratio from HSDL, HS, and BH trading rules. Re, FF, and SF refer to the reversal, first following, and second following rules, respectively. B and S respectively represent buy and sell rules. Syn is the brevity of synthetic rule and means the combination rule of these three rules. For example, SF_B refers to the second following buy rule, and FF_S refers to the first following sell rule.

For furtherly examining the adjusted returns for HSDL strategies, we calculate the Jenson Alphas from the Fama and French (2015) five-factor model. Relevant results are shown in Table 2.4. As such, these alphas are net of market, size, value, profitability, and investment patterns. Generally, the results in Table 2.4 demonstrate that HSDLs, including buy and sell rules, can capture positive alphas. For HSDL buy rules, these daily alphas exhibit a similar pattern to the Sharpe ratio. The average annualized alphas of HSDL buy rules, ranging from 10.31% to 16.56%, are significant at the level of 1%, and significantly greater than that from the BH rule. Specifically, differences of the average alphas between HSDLs and BH are valued from 6.15%-15.4% (see the column of DiffBH in Table 2.4). In addition, results also reflect that HSDL has a significant improvement to HS, meaning the average alphas of HSDL buyers are significantly larger than those of HSs (see the column of DiffHS in Table 2.4). For sell strategies, alphas of HSDLs also achieve high values, from 7.45% to 18.91%, and are significantly greater than that from the benchmark of BH, which means HSDL sell rules can

Table 2.4 Statistics of Annualized Jenson Alpha

Table 2.4 presents the summary statistics of annualized Alpha for HSDLs, HSs, and the BH, applied to 505 individual stocks in out-of-sample. The sample period ranges from 03/01/2007 to 31/12/2020. We use the formula of the Fama French five-factor model: $R_{it} - RF_t = a_i + b_i(RM_t - RF_t) + s_iSMB_t + h_iHML_t + r_iRMW_t + ciCMA_t + e_{it}$, where R_{it} is the return of the trading strategy, RF_t is the risk-free return, RM_t is the market return, SMB_t is the return spread of small minus large stocks, HML_t is the return spread of cheap minus expensive stocks, RMW_t is the return spread of the most profitable firms minus the least profitable, CMA_t is the return spread of firms that invest conservatively minus aggressively, a_i is the Alpha, and e_{it}, is the disturbance term. *, **, and *** denote statistically significantly different from the equivalent metric at the 10%, 5%, and 1% levels, respectively. Re, FF, and SF refer to the reversal, first following, and second following rules, respectively. B and S respectively represent buy and sell rules. Syn is the brevity of synthetic rule and means the combination rule of these three rules. For example, HSDL_SF_B refers to the second following buy rule of the HSDL strategy. DiffBH refers to the difference of annualized Alpha between HSDL and BH, and DiffHS represents the difference of annualized Alpha between HSDL and HS.

	Mean	DiffBH	DiffHS	Q3	Median	Q1
Panel A: Annualized Alpha (%) of HSDL strategies						
HSDL_Re_B	10.31***	6.15***	3.74***	19.74	6.76	-3.69
HSDL_FF_B	16.56***	15.40***	8.45***	27.78	10.79	-3.02
HSDL_SF_B	11.38***	7.22***	4.05***	16.61	6.57	-2.10
HSDL_Syn_B	11.97***	7.81***	4.24***	18.13	8.53	0.95
HSDL_Re_S	10.83***	6.67***	0.34	20.81	6.43	-5.96
HSDL_FF_S	18.91***	17.75***	4.16***	33.74	14.04	-2.82
HSDL_SF_S	7.45***	3.29**	-0.13	20.61	5.14	-4.31
HSDL_Syn_S	10.21***	6.05***	4.78***	18.56	8.76	-0.34
Panel B: Annualized Alpha (%) of the BH strategy						
BH	4.16***			8.05	2.92	-1.52
Panel C: Annualized Alpha (%) of HS strategies						
HS_Re_B	6.57***	2.89*		19.06	5.61	-2.81
HS_FF_B	8.11***	4.43***		19.73	3.68	-9.42
HS_SF_B	7.33***	3.65**		13.27	4.21	-4.74
HS_Syn_B	7.73***	4.05***		15.21	6.52	-0.44
HS_Re_S	10.49***	6.81***		19.87	3.95	-6.47
HS_FF_S	14.75***	11.07***		21.46	7.57	-4.85
HS_SF_S	7.58***	3.90***		18.05	3.91	-8.92
HS_Syn_S	5.43***	1.75		12.70	3.79	-4.59

effectively capture the positive Jenson alphas and are rewarded fairly adjusted returns. Sharpe ratios, however, do not reflect this property for HSDL sell strategies. This may tell us the indication of the Jenson alpha likely outperforms the Sharpe ratio when sell strategies are evaluated.

In summary, HSDL trading rules generate significant annualized returns,

outperforming HS and BH benchmarks. This strong performance remains consistent across risk-adjusted metrics, such as Sharpe ratios and Jensen's alpha, making it highly relevant for practical investment applications. Investors can efficiently program HS pattern identification, train the DL predictor, and implement HSDL trading strategies for automated trading. For HSDL buyers, these strategies apply to long trades, while for sellers, they can be utilized for short positions.

2.4.3 Trading frequencies and transaction costs

Table 2.5 exhibits a summary of trading frequencies of buying (or longing) and selling (or shorting) for HSDL strategies applied to 505 individual stocks. First, HSDLs are trading strategies with low trading frequency. The majority of rules create fewer than one trading signal per year on average. Second, HSDL buy rules generate more trading signals than corresponding sell rules. For example, HSDL_Syn_B produces 1.311 trading signals per year on average, whereas HSDL_Syn_S only signals 0.544 times per year. Among six individual HSDLs, excluding synthetic rules, HSDL_SF_B has the highest frequency of trading, which achieves 0.843 times per year. For exploring how many days each HSDL rule holds its position over the out-of-sample period, we compute the trading days for all HSDLs and compare these with the BH rule (see Table 2.6). The second row in Table 2.6 indicates the summary of trading days for the BH rule over 505 stocks. The average trading of BH is 3219.6 days. The rest of the rows display the statistics of trading days for HSDLs. The column of Mean shows the average trading days from HSDL strategies. For example, the average trading days of HSDL_Syn_B are 1154.3, and its proportion to BH' is 35.9%.

Table 2.5 Trading Frequency of HSDL Strategies

Table 2.5 reports the summary of trading frequency, the annualized time, for HSDL strategies applied to 505 individual S&P stocks in the out-sample period. The sample period ranges from 03/01/2007 to 31/12/2020. Annualized times mean how many times on average a strategy generates buy or sell signals during a year. TotalNum is the total number of stocks on which the specific HSDL strategy generates at least one trading signal. Re, FF, and SF refer to the reversal, first following, and second following rules, respectively. B and S respectively represent buy and sell rules. Syn is the brevity of synthetic rule and means the combination rule of these three rules.

	Annualized trading times						
	Min	Q1	Median	Mean	Q3	Max	TotalNum
HSDL_Re_B	0.073	0.375	0.515	0.527	0.654	1.154	474
HSDL_FF_B	0.074	0.366	0.488	0.508	0.589	1.109	493
HSDL_SF_B	0.146	0.726	0.871	0.843	0.961	1.474	490
HSDL_Syn_B	0.636	1.171	1.316	1.311	1.448	1.974	498
HSDL_Re_S	0.072	0.206	0.289	0.282	0.366	0.917	475
HSDL_FF_S	0.072	0.219	0.294	0.319	0.398	1.16	484
HSDL_SF_S	0.072	0.219	0.292	0.307	0.372	1.301	491
HSDL_Syn_S	0.209	0.436	0.515	0.544	0.655	1.302	498

Table 2.6 Statistics of Trading Days for HSDL Strategies

Table 2.6 reports statistics of trading days for HSDL and BH applied to 505 individual stocks in the out-sample period. The sample period ranges from 03/01/2007 to 31/12/2020. Re, FF, and SF refer to the reversal, first following, and second following rules, respectively. B and S respectively represent buy and sell rules. Syn is the brevity of synthetic rule and means the combination rule of these three rules. The column of HSDL/BH shows the percent value that is the trading days of HSDL divided by those of BH.

Rule	Mean	HSDL/BH(%)	Q3	Median	Q1
BH	3219.6		3457	3453	3398
HSDL_Re_B	549.4	17.1	714	539	387
HSDL_FF_B	308.9	9.6	400	304	210
HSDL_SF_B	558.7	17.4	670	577	452
HSDL_Syn_B	1154.3	35.9	1345	1197	1021
HSDL_Re_S	247.2	7.7	330	230	158
HSDL_FF_S	256.2	8.0	324	229	152
HSDL_SF_S	244.1	7.6	308	228	157
HSDL_Syn_S	603.5	18.7	549	417	318

One of the issues that investors are concerned about is the transaction cost, which determines the net profits of a trading strategy. In the previous analysis, we assumed a fixed transaction cost of 0.5% for the stock investment. In this session, we take a closer look at this

issue and examine how the transaction cost would affect the performance of the HSDLs. In so doing, we first estimate the breakeven transaction costs (BETC). For each of the HSDL buy strategies, we reduce returns to those of the BH rule to calculate their BETCs, while for HSDL sell rules, we assume to short stocks when they generate signals and then directly compute BETCs. The results of BETCs are reported in Table 2.7.

From Table 2.7, we can see that the values of the mean of BETCs from all HSDL strategies range from 155 to as high as 566 basis points. This is a very large level of transaction cost, and bear in mind that the BETCs are in addition to the 50-basis-point one-way transaction cost imposed in calculating the HSDL returns. Jones (2002) estimates one-way transaction costs of around 100 basis points in 1970 and 20 basis points in 2000 for NYSE stocks. Given transaction costs have likely declined further since 2000, it seems reasonable to assume an average transaction cost of around 50 basis points over our out-of-sample period. After imposing a fixed transaction cost of 50 basis points, HSDLs still have such large values of BETCs. Therefore, based on BETCs, transaction costs should not be a concern for the HSDL traders.

In addition, the performance of HSDL strategies shows low sensitivity to varying transaction costs (e.g., 0.75% or 0.1%), largely due to their low trading frequency of 0.3 to 1.3 times per year. Increasing the transaction cost from 0.5% to 0.75% or 1% alters the annualized mean returns by only 0.08%–0.33% or 0.15%–0.65%, respectively. These changes are negligible compared to the baseline returns of HSDL buyers (23.26% to 25.75%) and sellers (–4.72% to –14.04%).

Further, we consider another critical issue relevant to the transaction cost, spread,

Table 2.7 BETCs for HSDL Strategies

Table 2.7 in panel A reposts statistics of one-way breakeven transaction costs (BETCs) of HSDL buy strategies which reduce returns to those of the BH rule, in the out-sample period, ranging from 03/01/2007 to 31/12/2020. Panel C reports BETCs for HSDL sell strategies. Panel B and D show the average number of days each position is held for. Re, FF, and SF refer to the reversal, first following, and second following rules, respectively. B and S respectively represent buy and sell rules. Syn is the brevity of synthetic rule and means the combination rule of these three rules.

Rule	Mean	Q3	Median	Q1
Panel A: BETC (basis points)				
HSDL_Re_B	566	945	388	12
HSDL_FF_B	272	536	187	-172
HSDL_SF_B	367	511	203	2
HSDL_Syn_B	464	686	342	105
Panel B: Holding periods (days)				
HSDL_Re_B	81	91	79	68
HSDL_FF_B	49	54	47	41
HSDL_SF_B	51	57	50	45
HSDL_Syn_B	69	75	68	61
Panel C: BETC (basis points)				
HSDL_Re_S	155	531	121	-270
HSDL_FF_S	321	739	318	-64
HSDL_SF_S	408	1006	407	-168
HSDL_Syn_S	345	802	277	-106
Panel D: Holding periods (days)				
HSDL_Re_S	68	78	66	56
HSDL_FF_S	61	69	58	50
HSDL_SF_S	62	71	61	51
HSDL_Syn_S	63	70	62	54

which is the difference between the bid price and the ask price for a particular stock. Here, we tend to check whether the spread affects HSDLs performance, average returns. In our investigation, we use the daily close price of the stock to calculate returns for our trading rules. The daily close price is quoted in a value that could be the ask price, bid price, or the value between the ask and bid price, which depends on the regulation of the trading organization at the last trading moment of the day. In this subsection, we did an aggressive treatment that is the ask price is used for the buy signal and the bid price is collected for the selling signal, instead of the daily close price, and then re-calculate the annualized average

returns overall sample stocks for all HSDL rules and observe the difference between before and after consideration of the spread. The results are reported in Table 2.8. The second and third columns of Table 2.8, respectively, show the values of the annualized mean of returns in percentage after and before considering the spread, and the column of Diff gives the difference between the former and the latter. The differences are very small. For HSDL buy strategies, the annualized average returns only have a slight drop (0.4%, 0.6%, 0.5%, and 0.9% for the four HSDL buy rules, respectively). Likewise, the annualized average returns from HSDL sell strategies also increase by a small percentage (around 1%). Therefore, the evidence again indicates that the transaction cost has a minor impact on the performance of HSDLs. These rules appear to beat the BH or generate positive returns after transaction costs.

Table 2.8 Considering the Effect of the Spread

Table 2.8 presents the summary statistics of annualized average returns in percentage for SHDLs after considering the effect of the spread, over the out-of-sample period. Re, FF, and SF refer to the reversal, first following, and second following rules, respectively. B and S respectively represent buy and sell rules. Syn is the brevity of synthetic rule and means the combination rule of these three rules. For example, HSDL_SF_B refers to the second following buy rule of the HSDL strategy. MeanBefore refers to the annualized average returns before the consideration of the spread effect, which are the same as in Table 2.1 for corresponding trading rules. Diff refers to the difference in the annualized average return between before and after consideration of the spread effect.

Rule	Mean	MeanBefore	Diff	Q3	Median	Q1
HSDL_Re_B	25.33	25.75	-0.4	39.76	20.31	6.75
HSDL_FF_B	22.64	23.26	-0.6	38.01	17.43	5.23
HSDL_SF_B	24.95	25.46	-0.5	34.32	18.91	7.35
HSDL_Syn_B	24.39	25.26	-0.9	32.79	20.77	10.43
HSDL_Re_S	-4.58	-4.72	0.1	10.46	-4.19	-20.73
HSDL_FF_S	-10.57	-11.67	1.1	1.98	-12.65	-29.52
HSDL_SF_S	-13.91	-14.04	0.1	7.43	-15.32	-39.53
HSDL_Syn_S	-11.05	-12.13	1.1	6.86	-8.86	-26.33

2.4.4 Performance on the worst days

In the following context, we attempt to test if the HSDL strategy is susceptible to 'crash risk'. These periods of persistent negative returns can be a substantial risk for investors.

In Table 2.9, we focus on the lowest daily returns generated by a BH investor and by HSDL strategies and compare these with the returns earned by an investor using an alternative approach on that day.

In this experiment, we have three rules that are expressed as follows.

The BH rule: Invest equal money in all sample stocks. Buy and hold stocks.

The HSDL_Syn_B_RF rule: Invest equal money in all sample stocks. Long stocks when the HSDL synthetic buy rule generates buy signals; otherwise, invest in risk-free assets.

The HSDL_Syn_B_S_RF rule: Invest equal money in all sample stocks. Long stocks when the HSDL synthetic buy rule generates buy signals; and short stocks if HSDL synthetic sell rule produces sell signals and HSDL synthetic buy rule has no trading signals; otherwise invest in risk-free assets.

Results in Panel A of Table 2.9 indicate both HSDL_Syn_B_RF and HSDL_Syn_B_S_RF rules have considerably less downside risk than a BH rule. The lowest BH return (-12.83%) was on 16 March 2020. However, investors adopting the HSDL_Syn_B_RF rule would have invested the majority of fund in the risk-free item on this day and therefore would only lose a small amount of money (-2.86%), and investors applying the HSDL_Syn_B_S_RF strategy not only have invested in the risk-free security and also short some stocks and thus furtherly reduce their loss (-1.27%). A similar pattern is evident in each of the other nine days. The interesting thing is that the HSDL_Syn_B_S_RF rule even helps investors to achieve positive returns of 2.57% and 1.73% in the third and eighth worst days when the BH rule lost big money (-10.33% and -7.65%).

Table 2.9 Worst Day Performance

In this table, Re, FF, and SF refer to the reversal, first following, and second following rules, respectively. B, S, and RF respectively represent buy, sell, and risk-free. Syn is the brevity of the synthetic rule and means the combination rule of these three rules.

Date (yyyymmdd)	HSDL_Syn_B_RF(%)	HSDL_Syn_B_S_RF(%)	BH(%)
Panel A: Rank by BH			
20200316	-2.86	-1.27	-12.82
20200312	-2.34	-1.06	-10.50
20081201	-1.72	2.57	-10.33
20081015	-2.54	-0.48	-9.52
20200309	-1.78	-0.46	-9.19
20200318	-1.33	-0.19	-7.96
20080929	-2.45	-0.58	-7.89
20081119	-1.46	1.73	-7.65
20081009	-2.22	-0.42	-7.45
20110808	-2.01	0.04	-7.38
Panel B: Rank by HSDL_Syn_B_RF			
20090302	-3.42	-2.52	-5.66
20090305	-2.97	-1.91	-5.31
20200611	-2.88	-1.62	-6.78
20200316	-2.86	-1.27	-12.82
20090420	-2.72	-1.94	-5.35
20090120	-2.68	-0.52	-6.71
20090622	-2.63	-2.51	-3.47
20081015	-2.54	-0.48	-9.52
20080929	-2.45	-0.58	-7.89
20200312	-2.34	-1.06	-10.50
Panel C: Rank by HSDL_Syn_B_S_RF			
20090302	-3.42	-2.52	-5.66
20090622	-2.63	-2.51	-3.47
20090420	-2.72	-1.94	-5.35
20090305	-2.97	-1.91	-5.31
20160624	-1.97	-1.85	-4.10
20090702	-1.91	-1.83	-3.22
20090513	-2.25	-1.81	-3.88
20090615	-1.89	-1.79	-2.70
20090223	-2.30	-1.77	-3.65
20090624	-2.08	-1.74	-3.35

In Panels B and C, we compare the downside risk of the HSDL_Syn_B_RF and

HSDL_Syn_B_S_RF rules. In all the 10 worst days ranked by these two rules, investors employing the HSDL_Syn_B_S_RF rule always had less loss than those using the HSDL_Syn_B_RF rule, which means the former did a better job than the latter when facing the downside risk. The results also show that investors using the BH rule lost more money than investors utilizing HSDL strategies in these two 10 worst days.

2.4.5 Performance Over Several Sorted Stock Groups

We sort all sample stocks annually on the first trading day by the market value (MV), illiquidity (ILLI) (the measure Amihud and Mendelson (2015)), and standard deviation (Std), respectively, and split them into five groups, low, 2, 3, 4, and high, with the equal number of 101 stocks. We observe the patterns of the change in the performance of the annualized average return of stock groups as these measures increase or decrease.

From Panel A of Table 2.10, we can observe that the annualized average returns of BH in the sorted group of low are much lower than the other four groups, while the HSDL_Syn_B rule significantly increases the return in the low group, meaning this rule is fairly independent of market values.

In contrast, for the HSDL_Syn_S rule, there is a clear pattern, showing the smaller the market value of the groups, the larger the negative returns, which means the HSDL sell rule can achieve more profits as the market values of groups decrease.

Panel B of Table 2.10 illustrates that there is no obvious pattern between returns and groups sorted by illiquidity for HSDL rules and their benchmarks. For example, the HSDL_Syn_B rule gains similar annualized returns over four groups (low, 2, 3, 4), except a

Table 2.10 Performance Across MV, ILLI, and Std Sorted Groups

Table 2.10 presents the annualized average returns in percentage for HSDLs and benchmarks, HS and BH. MV, ILLI, and Std refer to the market value of the stock, illiquidity measured by the measure Amihud and Mendelson (2015), and standard deviation of stock returns, repectively. Low, 2, 3, 4, high are groups sorted respectively by MV, ILLI, and Std, with equal number of 101 stocks. Re, FF, and SF refer to reversion, first following, and second following rules, respectively. B and S respectively represent buy and sell rules. Syn is the brevity of synthetic rule and means the combination rule of three rules, reversion, first following, and second following rules. For example, HSDL_Syn_B refers to the synthetic buy rule of the HSDL strategy.

	Annualized average return (%)				
Sorted group	low	2	3	4	high
Panel A: sorted by the market value					
HSDL_Syn_B	17.06	28.52	24.15	25.07	21.78
HS_Syn_B	6.23	17.96	18.08	21.76	20.03
HSDL_Syn_S	-14.59	-12.84	-10.27	-9.53	-5.40
HS_Syn_S	-10.74	-9.21	-7.32	-6.11	-1.97
BH	3.23	9.00	12.60	11.04	12.24
Panel B: sorted by illiquidity					
HSDL_Syn_B	24.15	20.41	22.58	19.72	29.24
HS_Syn_B	18.87	18.40	18.15	9.07	19.15
HSDL_Syn_S	-12.08	-3.81	-8.65	-15.98	-11.73
HS_Syn_S	-6.58	-3.40	-4.14	-12.05	-8.90
BH	11.13	13.04	9.13	4.49	10.25
Panel C: sorted by standard deviation of the stock return					
HSDL_Syn_B	13.28	21.44	23.99	25.34	32.74
HS_Syn_B	14.53	21.08	19.12	17.48	11.60
HSDL_Syn_S	0.94	-2.10	-7.05	-9.84	-30.45
HS_Syn_S	-1.43	-2.98	-5.02	-6.89	-17.79
BH	8.57	11.85	10.62	9.04	7.93

higher return in the group of high.

Panel C of Table 2.10 shows the interesting relationship between the return and the standard deviation of the stock. Both HSDL and HS buy rules attain higher returns as values of the standard deviation increase, associated sell rules have larger negative returns when groups show more volatility of returns, while BH does not display the same pattern. Hence, we can acknowledge that the performance of HSDL strategies is dependent on the volatility of stock returns. The more volatile stocks are, the more profitability HSDLs can achieve.

2.4.6 Valuation of HSDL under the FDR Controlling Procedure

2.4.6.1 Annualized Mean Returns of HSDL Rules

So far, we have examined the average of annualized mean returns of HSDLs and their corresponding benchmarks, HS and BH strategies, applied on our sample stocks, and the results suggest that HSDL outperforms the HS and BH rules. To verify this conclusion, we further conducted multiple tests based on all individual tests among 505 stocks. We tend to answer the following two questions to fulfill this task.

- How many stocks do the HSDL and HS strategies outperform or underperform relative to the benchmarks, namely the Buy-and-Hold strategy and the zero-return baseline?
- How many stocks do HSDL strategies outperform or underperform HS strategies?

In our cases, the standard approach is to perform individual statistical tests on stocks, checking the difference of annualized mean returns between HSDL, HS, and benchmarks, and then make a selection based on the significance of these individual tests. However, this unguarded use of single-inference procedures results in the data snooping concern that as more and more tests are performed, an increasing number of them will be positive purely due to chance. This means that a potentially large fraction of the tests that ex-post appear positive could be "false discoveries". In the extreme case, the significance of the individual tests becomes completely uninformative when the "false discovery proportion" (FDP) reaches 100%. Benjamini and Hochberg (1995) proposed a false discovery controlling approach that controls the expectation of FDP, the false discovery rate (FDR), to implement multiple testing

and then resolve this data snooping issue. The FDR control procedure aims to control the overall false discovery rate below a prespecified level (say, 5%) by optimally setting different significance thresholds across different individual tests. Giglio, Liao, and Xiu (2021) developed the FDR control procedure by adding the screening step and displaying solid improvement. Our methodology follows these two papers to execute the multiple testing. We now describe the details of the FDR controlling approach in our study.

First, we explain the definition of FDR. Consider the problem of testing simultaneously N (null) hypotheses, H_0^i (i,…,N). Suppose t_j is a test statistic for the null H_0^j. R is the number of hypotheses rejected, and V is the number of false rejections in the sample. R is an observable random variable, whereas V cannot be observed. Thus, the false discovery proportion (FDP) can be written as FDP = V/R, and FRD is defined as the expectation of FDP in the equation of 2.4.6.1.

$$DR = E(FDP) = E(V/R) \qquad (2.4.6.1)$$

Analogously to standard individual tests (that control the size of Type I error), FDR controlling procedure controls the size of the FDR: it ensures that FDR$\leqslant \tau$. In this study, we set $\tau = 5\%$.

Second, this method aims to identify a critical value p^*, so that the null H_0^i is rejected for all $p_i < p^*$. Specifically, $p^* = \tau (i / N)$, where i is the index of p_i, which is sorted in ascending order. For each sample i, it has a corresponding threshold, p^*.

Third, screen samples having “deep in the null” and then decrease the number of N to *I* for identifying the *p**. We use a formula suggested by Giglio et al. (2021) to estimate *I*. The formula is the equation of 2.4.6.2.

$$\widehat{\mathcal{I}} = \left\{ i \le N : t_i > -\log(\log T)\sqrt{\log N} \right\}, \qquad (2.4.6.2)$$

where T is the average number of the target parameter. In our study, T is the average of the number of returns on stocks having HS or HSDL trading signals. Therefore, p* is adjusted to the one, $\tau(i / I)$. We will reject the null H_0^i for all $p_i < \tau(i / I)$.

Finally, we can calculate the fraction, i/N, which expresses how many null hypotheses are rejected in the total tests.

Algorithm of the FDR controlling approach

S1. Carry out the t-test for target parameters of HSDL and HS strategies, and then calculate *p*-values and sort them in ascending order, $\{p_i : i = 1,\ldots,N\}$, of the t-test statistics $\{t_i\}$. N is the number of stocks appearing HS or HSDL signals. Denote $p_{(1)} \le \ldots \le p_{(N)}$ as the sorted *p*-values.

S2. Screen stocks having “deep in the null” to decrease the number of N to *I*.

Thus, reduce the sorted sequence of the p-value to: $p_{(1)} \le \ldots \le p_{(I)}$, for $\{ p_i : i \in I \}$.

S3. For $i \in I$, reject $\mathbf{H_0^i}$ if $p_i \le p_{(k)}$, where $k = \max\{ i \in I : p_{(i)} \le \tau(i / I) \}$. Accept all other $\mathbf{H_0^i}$.

S4. Calculate the fraction: i / N. $\tau = 5\%$.

We totally conduct six multiple texts. Details of six hypotheses are as follows.

1. $\mathbf{H_0^1}$ **:** $\mu_{i,HS\text{ or }HSDL} - \mu_{i,BH} \leq 0$, and $\mathbf{H_1^1}$ **:** $\mu_{i,HS\text{ or }HSDL} - \mu_{i,BH} > 0$.
2. $\mathbf{H_0^2}$ **:** $\mu_{i,HS\text{ or }HSDL} - \mu_{i,BH} \geq 0$, and $\mathbf{H_1^2}$**:** $\mu_{i,HS\text{ or }HSDL} - \mu_{i,BH} < 0$.
3. $\mathbf{H_0^3}$ **:** $\mu_{i,HS\text{ or }HSDL} \leq 0$, and $\mathbf{H_1^3}$**:** $\mu_{i,HS\text{ or }HSDL} > 0$.
4. $\mathbf{H_0^4}$ **:** $\mu_{i,HS\text{ or }HSDL} \geq 0$, and $\mathbf{H_1^4}$ **:** $\mu_{i,HS\text{ or }HSDL} < 0$.
5. $\mathbf{H_0^5}$**:** $\mu_{i,HSDL} - \mu_{i,HS} \leq 0$, and $\mathbf{H_1^5}$ **:** $\mu_{i,HSDL} - \mu_{i,HS} > 0$.
6. $\mathbf{H_0^6}$ **:** $\mu_{i,HSDL} - \mu_{i,HS} \geq 0$, and $\mathbf{H_1^6}$**:** $\mu_{i,HSDL} - \mu_{i,HS} < 0$.

$\mu_{i,HS}$, $\mu_{i,HSDL}$ and $\mu_{i,BH}$ are the mean of returns of the HS, HSDL, and BH strategies, respectively, on stock i.

Table 2.11 displays the fractions of involved stocks rejecting the null hypotheses under the FDR controlling procedure. To explain these results clearly, we categorize the buy and sell strategy groups to discuss.

First, we focus on results related to HSDL and HS buy strategies. For buy strategies, we care for the extent to which HSDLs and HSs can beat the benchmark of the BH or reach positive annualized mean returns on total stocks. The results of tests of hypotheses (from 1 through 4) in Table 2.11 respond to this concern. The second and third columns of Panel A exhibit the fractions of stocks for HSDL and HS strategies, rejecting the null hypotheses of $\mu_{i,HS\text{ or }HSDL} - \mu_{i,BH} \leq 0$ and $\mu_{i,HS\text{ or }HSDL} - \mu_{i,BH} \geq 0$. Overall, each HSDL or HS strategy has a higher fraction value in the second column than its associated value in the third column. Specifically, we can see all HS and HSDL buy strategies have higher mean returns than the BH rule on the majority of stocks, and only there is a smaller number of stocks on which HSs and HSDLs

underperform the BH rule. For example, the second column shows HS_Re_B, HS_Sy_B, HSDL_Re_B, and HSDL_Sy_B strategies reach the fractions of 0.608, 0.683, 0.715, and 0.807 of total stocks, respectively, which means these strategies significantly outperform the BH rule on 60.8%, 68.3%, 71.5%, and 80.7% of total stocks, respectively. In contrast, in the third column, we can find that HS_Re_B, HS_Sy_B, HSDL_Re_B, and HSDL_Sy_B rules significantly underperform the BH rule only on 29%, 19.9%, 19.4%, and 11.2% of total stocks, respectively. Based on the outcomes of multiple tests, we can solidly conclude that HSDL and HS strategies both overall beat the BH rule. For further evaluating our strategies, we also make sure how many stocks can gain profits through these buy rules. Results in the fourth column indicate that on more than 70% of stocks, HS and HSDL buy strategies can achieve positive mean returns, except for the 61.3% for the HS_FF_B strategy. The best one is the strategy of HSDL_Sy_B, which can get positive mean returns on 90.9% of stocks. In addition, comparing the fraction values between HSDLs and HSs, we also notice that HSDLs can achieve higher values than corresponding HSs do across all tests of four null hypotheses, which suggests that HSDL strategies could perform better, relevant to ES strategies.

Second, the sales strategies of HS and HSDL seek to gain negative returns, which allows the strategies can realize the profits. Therefore, here we emphasize the fractions of stocks rejecting the null of $\mu i_{\text{HS or HSDL}} \geq 0$. The fifth column of Panel B lists the fractions of stocks on which HS and HSDL strategies attain negative mean returns. The annualized mean returns from all HS and HSDL strategies are smaller than zero on at least more than 52.8% of stocks. Specifically, HSDL_FF_S, HSDL_SF_S, and HSDL_Sy_S rules increase these proportions to 64.7%, 62.7%, and 63.4%, respectively. With the multiple testing aspects, the

HS sell strategies on average can get negative returns, and HSDLs effectively reach this goal of achieving negative mean returns, also suggesting HSDLs are better, compared to ESs.

We continue to conduct multiple tests to verify if HSDL strategies outperform HS strategies. Table 2.12 illustrates the fractions of stocks on which two nulls that are $\mathbf{H_0^5}$: $\mu_{i,HSDL}-\mu_{i,HS}\leq 0$ and $\mathbf{H_0^6}$: $\mu_{i,HSDL}-\mu_{i,HS}\geq 0$ are rejected. If the null of $\mu_{i,HSDL}-\mu_{i,HS}\leq 0$ is rejected, this means the HSDL buy strategy outperforms the HS buy strategy, and likewise, if rejecting the null of $\mu_{i,HSDL}-\mu_{i,HS}\geq 0$, this represents the HSDL buy strategy underperforms the HS buy strategy. For sell strategy, the logic is just the opposite. If rejecting $\mathbf{H_0^6}$, we can say the HSDL sell strategy outperforms the corresponding HS strategy, and similarly, the HSDL strategy underperforms the associated HS strategy when rejecting $\mathbf{H_0^5}$. From the results in Panel A of Table 2.12, we can find that all four buy rules of HSDL have larger fractions for outperforming than fractions for underperforming. For example, HSDL_SF_B outperforms the HS_SF_B on the 52.9% stock and underperforms the HS_SF_B on the 24.8%, which reflects that the HSDL strategy reaches an improvement relevant to the corresponding HS strategy. Thus, we could verify that the HSDL buy strategy is better than its associated HS buy strategy, based on the individual tests on all 505 stocks, controlled by FDR.

Panel B of Table 2.12 demonstrates a similar outcome to Panel A. The fractions rejecting $\mathbf{H_0^6}$ are larger than the fractions rejecting $\mathbf{H_0^5}$ for three sellers, HSDL_FF_S, HSDL_SF_S, and HSDL_Sy_S, which suggests that the HSDL could beat the HS.

Table 2.11 Multiple Testing of HS and HSDL over the BH and Zero Return

Table 2.11 presents the fractions rejecting the null hypotheses through the FDR controlling procedure across all sample stocks, over the out-of-sample period. $\mu_{i,HS}$,$\mu_{i,HSDL}$ $\mu_{i,BH}$ are the annualized mean returns for the HS, HSDL, and BH strategies, respectively. Re, FF, and SF refer to reversion, The first following rules, and second following rules, respectively. B and S, respectively, represent buy and sell rules. Syn is the brevity of the synthetic rule and means the combination rule of these three rules. τ is the control rate for FDR.

	Fraction rejecting H_0 using the FDR controlling procedure (at the level of 5%)				Total number
	$\mathbf{H_0^1}$ **:** $\mu_{i,HS \text{ or } HSDL} - \mu_{i,BH} \leq 0$	$\mathbf{H_0^2}$ **:** $\mu_{i,HS \text{ or } HSDL} - \mu_{i,BH} \geq 0$	$\mathbf{H_0^3}$ **:** $\mu_{i,HS \text{ or } HSDL} \leq 0$	$\mathbf{H_0^4}$ **:** $\mu_{i,HS \text{ or } HSDL} \geq 0$	
Panel A : Buy rules					
HS_Re_B	0.608	0.291	0.752	0.202	492
HS_FF_B	0.499	0.406	0.613	0.295	497
HS_SF_B	0.566	0.304	0.776	0.146	500
HS_Syn_B	0.683	0.199	0.821	0.135	501
HSDL_Re_B	0.715	0.194	0.796	0.121	474
HSDL_FF_B	0.588	0.31	0.704	0.211	493
HSDL_SF_B	0.679	0.188	0.849	0.078	490
HSDL_Syn_B	0.807	0.112	0.909	0.053	498
Panel B: Sell rules					
HS_Re_S	0.221	0.677	0.369	0.535	493
HS_FF_S	0.177	0.727	0.308	0.588	496
HS_SF_S	0.184	0.723	0.327	0.569	499
HS_Syn_S	0.121	0.78	0.311	0.592	501
HSDL_Re_S	0.242	0.684	0.366	0.528	475
HSDL_FF_S	0.169	0.764	0.241	0.647	484
HSDL_SF_S	0.201	0.733	0.263	0.627	491
HSDL_Syn_S	0.132	0.779	0.261	0.634	498

Table 2.12 Multiple Testing of HSDL Over HS Return

Table 2.12 presents the fractions rejecting the null hypotheses through the FDR controlling procedure across all sample stocks, over the out-of-sample period. $\mu_{i,HS}$,$\mu_{i,HSDL}$ $\mu_{i,BH}$ are the annualized mean returns for the HS, HSDL, and BH strategies, respectively. Re, FF, and SF refer to reversion, The first following, and second following rules, respectively. B and S, respectively, represent buy and sell rules. Syn is the brevity of the synthetic rule and means the combination rule of these three rules. τ is the control rate for FDR.

	Fraction rejecting H_0 using the FDR controlling procedure (at the level of 5%)		Total number
	$\mathbf{H_0^5}$ **:** $\mu_{i,HSDL}$-$\mu_{i,HS}\leq 0$	$\mathbf{H_0^6}$ **:** $\mu_{i,HSDL}$-$\mu_{i,HS}\geq 0$	
Panel A: Buy rules			
HSDL_Re_B	0.362	0.259	493
HSDL_FF_B	0.321	0.318	496
HSDL_SF_B	0.529	0.248	490
HSDL_Syn_B	0.490	0.284	498
Panel B: Sell rules			
HSDL_Re_S	0.413	0.369	475
HSDL_FF_S	0.341	0.452	484
HSDL_SF_S	0.303	0.475	491
HSDL_Syn_S	0.321	0.488	498

2.4.6.2 Alpha Exploration in the Factor Zoo

In this subsection, we use the asset pricing model containing hundreds of factors from the 'factor zoo' to explore whether there are positive alphas for our HSDL strategies. Our factor library contains 153 risk factors that are collected from the website of Professor BRYAN KELLY, at a daily frequency over the out-of-sample period. In recent research, hundreds of factors have been suggested to explain the expected stock returns, as noted by Jenson, Kelly, and Pedersen (2021). Feng, Giglio, Xiu (2020) summarize that these factors include, among others, the two new factors introduced by Fama and French (2015) and Hou, Xue, and Zhang (2015), and the intermediary-based factors of He, Kelly, and Manela (2017). Therefore, this means our asset pricing model incorporates five factors of Fama and French model, intermediary factors from He, Kelly, and Manela (2017) and Jenson, Kelly, and Pedersen (2021).

Due to the potentially high intercorrelation and redundant factors among the factor zoo, we adopt the two-stage regressions, Lasso plus ordinary least squares (OLS) regressions. In the first stage, we applied Lasso regression to automatically select adequate factors. Lasso regression is well-suited for models showing high levels of multicollinearity among independent variables. Lasso regression sets an L1 penalty in its loss function that has the effect of shrinking the coefficients for those independent variables that do not contribute much to the prediction task. This penalty can decrease some coefficient values to zero, allowing variables to be effectively removed from the model, and consequently, a type of automatic feature selection is implemented. A cross-validation (CV) procedure is chosen to tune the parameters of Lasso and select proper risk factors based on the prediction results. In the second stage, we regress the selected risk factors on stocks' returns from the HSDL strategy using an OLS regression model with the correction of Newey-West HAC to gain the values and t-statistics of the intercepts, alphas. If the alpha is significantly larger than zero on an individual stock, we could evaluate the HSDL rules work on this stock. Due to about 500 individual stocks in

Table 2.13 Annualized Alpha

Table 2.13 presents the annualized alphas in percent from the two-stage regression and fractions rejecting the null hypotheses through the FDR controlling procedure across all sample stocks, over the out-of-sample period. B, S, and RF, respectively, represent buy, sell, and risk-free. Syn is the brevity of the synthetic rule and means the combination rule of these three rules related to HSDL or HS. τ is the control rate for FDR, and its value is 5%.

	Annualized Alpha %				Fraction rejecting
	Mean	Q3	Median	Q1	H_0: alpha ≤ 0
Trading rules					
HSDL_Syn_B_S_RF	11.09	16.77	9.34	4.08	0.589
HS_Syn_B_S_RF	6.41	10.86	5.38	0.63	0.336
HSDL_Syn_B_RF	7.92	11.47	7.04	3.61	0.505
HS_Syn_B_RF	4.02	8.08	3.56	-0.09	0.312
BH	3.51	6.59	2.16	-1.48	0.117

the investigation, the statistical assessment involves multiple tests, so we again employ the FDR controlling approach to gain the fraction of stocks on which the alpha is significantly positive and fulfill the task of alpha exploration for HSDL strategies.

We study two HSDL switching rules, buy signals only and both buy and sell signals, in comparison with benchmarks, the corresponding HS and BH rules.

- Buy only: We long stocks when the HSDL synthetic buy rule generates buy signals; otherwise, invest in risk-free assets. This rule is denoted as HSDL_Syn_B_RF and its associated benchmark from HS is recorded as HS_Syn_B_RF.
- Buy and sell: We long stocks when the HSDL synthetic buy rule creates buy signals; and short stocks if the HSDL synthetic sell rule produces sell signals and the HSDL synthetic buy rule has no trading signals; otherwise, invest in risk-free assets. The rule and its corresponding benchmark are denoted as HSDL_Syn_B_S_RF and HS_Syn_B_S_RF, respectively.

Table 2.13 reports the statistics of the annualized alphas for two trading rules of HSDL and their benchmark rules. From the results in Table 2.13, we can see that both buy-only and buy and sell trading rules of HSDL achieve positive means of annualized alphas overall sample stocks. The HSDL_Syn_B_S_RF and HSDL_Syn_B_RF attain 11.09% and 7.92% of the average alphas, respectively, while their corresponding HS rules only achieve 6.41% and 4.02% of the mean of alphas. The fifth column in Table 2.13 displays the fractions that reject the hypothesis of alpha ≤ 0. The HSDL rules achieve significantly positive alphas on more than 52% of stocks. The values of the fractions are 0.589 and 0.525 for HSDL_Syn_B_S_RF and HSDL_Syn_B_RF, respectively, which are larger than those from benchmarks.

2.5 Conclusions

In this study, we integrate deep learning (DL) techniques with an advanced chart pattern approach, the Hub Strategy (HS), to develop a novel trading strategy called HSDL. In this framework, DL models are trained on cross-sectional datasets that include features derived from HS chart patterns, along with additional inputs such as trading volume, price volatility, and market sentiment, to accurately predict the direction of price movements. The HSDL rules can achieve substantial annualized average returns on S&P 500 stocks, ranging from 23.26% to 25.75%, and considerably outperform the buy-and-hold (BH) rule (9.64%). and further, there is an improvement for HSDLs over corresponding pure HS strategies. This shows DL techniques could be successfully applied to enhance the predictability of chart analysis.

Our empirical results consistently exhibit the excellent performance of HSDL strategies relevant to some risk-adjusted measures, the Sharpe ratio, and Jensen's alpha. For examining the effect of transaction costs on the HSDL performance, we also estimate the breakeven transaction costs (BETC) and analyze the influence of the price spreads (the difference between bid and ask prices) on the performance of HSDL models. These outcomes ascertain that HSDL strategies are less sensitive to transaction costs.

HSDL strategies are not particularly susceptible to crash risk. On 10 worst days, returns of HSDL are much smaller than the market return, and the HSDL rule with the short function even gains a positive return on these worst days.

The issue of data mining in multiple tests is addressed in the study through a false discovery rate (FDR) controlling approach. Based on this approach, we calculate the fractions rejecting the null and closely look at all individual sample stocks to confirm that HSDL strategies can outperform their benchmarks, pure HS and BH rules. These multiple tests solidly show that the HSDL strategy is superior to the pure HS strategy and

can significantly beat the BH rule on the proportion of 80.7% of all sample stocks.

We last use the recent asset pricing model that contains 153 risk factors, and a two-stage regression is conducted, which involves the Lasso regression at the first stage and the OLS in the second stage. Also, under the FDR controlling approach, the result displays that the HSDL strategy can achieve significantly positive alphas on the majority of sample stocks and verifies the profitability of the HSDL strategy, compared to the pure HS and BH strategy.

In summary, our study contributes to the literature in several key ways. First, we integrate deep learning (DL) techniques with a novel chart analysis, the Hub Strategy (HS), introducing the HSDL trading strategy. This approach enables traders and analysts to apply chart analysis in a quantitative and multidimensional way, incorporating chart-based features of HS patterns as a new type of input. Additionally, our automated method for identifying chart patterns advances traditional chart pattern analysis. Finally, we conduct a series of rigorous tests to verify the robustness and superior predictability of HSDL compared to BH and pure HS rules.

Lastly, we acknowledge several limitations in our study. The limited number of HS chart patterns constrains the learning efficiency of deep learning (DL) models. Expanding to more stocks and markets would provide a larger dataset, improving model performance. Additionally, integrating multiple timeframes—such as daily, weekly, and 30-minute HS chart patterns—could enhance trade entry and exit decisions. Furthermore, our current DL models have a basic architecture. Developing more advanced models could further improve the accuracy of the DL predictor.

Chapter Three

Long Run Event Study with Deep Learning

This chapter presents the second essay, which utilizes deep learning (DL) techniques to enhance the abnormal return (AR) estimation, highlighting DL's strengths in managing complex financial data and increasing estimation accuracy. Section 3.1 presents a brief overview of the essay and explains the motivation to conduct this study. Section 3.2 summarizes the literature review. Section 3.3 develops our methodology relative to the estimation of AR. Sections 3.4 and 3.5 analyze the data and empirical results. Section 3.6 concludes the paper. Appendices and the reference list of this chapter are reported at the end of this thesis.

Abstract

We utilize deep learning (DL) techniques to examine the relationships between the abnormal return (AR) and its associated explanatory variables, which encompass firm-based characteristics and the realized return of stocks. The trained DL models are built on cross-sectional data and are well-suited for directly predicting estimated abnormal returns. When comparing the power of testing the abnormal return between DL models and the other three benchmarks, the DL models exhibit significant enhancement in performance. Moreover, our DL approaches contribute to mitigating the extent of misspecifications in non-random samples. By conducting a regression of the error of predicted AR on relative firm-based factors, market conditions, and periods, we discover that the DL model displays significantly reduced sensitivity to factors such as firm size, market conditions, and periods, as opposed to the benchmark.

Keywords: Long-run abnormal return, Event study, Deep Learning, Firm characteristics.

3.1 Introduction

Abnormal return (AR) in the context of stock markets refers to the deviation of an actual return from the expected return over a specific period. In event studies, abnormal returns are crucial for assessing the impact of specific events—such as earnings announcements, mergers, regulatory changes, or macroeconomic shocks—on stock prices. Within the domain of event studies, researchers investigate the potential occurrence of abnormal stock returns over extended periods ranging from one to five years, subsequent to pivotal corporate event disclosures. Typically, their focus centers on precise estimation of the normal return or benchmark. This serves as the foundation for subsequently discerning abnormal returns, achieved by isolating the normal returns from the realized returns of the companies under scrutiny. This methodological framework has found widespread utilization across both earlier and contemporary literature (e.g., Brown and Warner (1980, 1985); Campbell and Wasley, 1993; Barber and Lyon, 1997; Fama, 1998); Lyon, Barber, and Tsai 1999; Mitchell and Stafford, 2000; Boehme and Sorescu, 2002; Kolari and Pynnönen, 2011; Bessembinder, Cooper, and Zhang, 2019). For example, Lyon, Barber, and Tsai (1999) calculate abnormal returns using carefully construct reference portfolios as benchmarks, while recently, Bessembinder, Cooper, and Zhang (2019) propose that fitted values from market-wide regression of firm returns on lagged firm characteristics provide useful benchmarks (normal returns) for assessing whether average returns to certain stocks are abnormal.

Our investigation is inspired by Bessembinder, Cooper, and Zhang (2019), yet it unveils two distinctive facets. Primarily, we embark on a journey of utilizing a specific

machine learning technique known as Deep Learning (abbreviated as DL hereafter), in lieu of the conventional linear regression method employed in their study. In practical terms, scholars posit the existence of a nonlinear relationship amid financial variables. DL techniques have been demonstrated to adeptly uncover these nonlinear relationships between input (independent variables) and output (dependent variable). A comparative analysis undertaken by Athey and Imbens (2019) and Coulombe, Leroux, Stevanovic, and Surprenant (2022) focuses on pitting DL against conventional econometric methods. Their argument highlights the flexibility of DL in handling various functional forms, allowing it to capture complex non-linear relationships between dependent and independent variables. Unlike traditional statistical methodologies, DL approaches thrive on data-driven insights and avoid explicit assumptions concerning the model linking inputs to outputs. Moreover, DL methodologies typically integrate cross-validation and regularization techniques, augmenting their potential to achieve robust predictive capabilities.

Secondly, we reevaluate the conceptual framework of estimating abnormal returns from an alternative philosophical standpoint. Here, we employ DL to directly construct the relationship between abnormal returns and variables encompassing firm-level characteristics and realized returns. In contrast, Bessembinder, Cooper, and Zhang (2019) focus on forecasting predicted returns as normal returns using regression models. Given the known return and firm characteristics for a specific stock, our design incorporates DL models capable of directly calculating the value of abnormal returns spanning the interest period. Our concentration is on discerning the long-term abnormal returns over spans of one, three, and five years. We simply start with the assumption that the DL model is trained to recognize the

prevailing “normal” market conditions. Therefore, if a stock and a month are randomly selected, the expected abnormal return for that stock in that month is zero. In the training process of the DL model, inputs encompass the realized return and firm-level variables. In this scenario, the output is associated with an abnormal return set at zero. Furthermore, we supplement realized returns with additional returns, spanning from -3% to 3%, with increments of 0.2% on an annual basis. Consequently, the outputs (abnormal returns) align with the corresponding supplementary returns. This augmentation extends the scope of the input-output relationship. We anticipate that this approach augments DL's proficiency in identifying patterns interlinking abnormal returns, firm characteristics, and realized returns.

Our investigation further unveils compelling evidence that the 14 firm characteristics proposed by Lewellen (2015) exhibit a significant correlation with future stock returns. This finding aligns harmoniously with the outcomes observed by Lewellen (2015) and the conclusions reached by Bessembinder, Cooper, and Zhang (2019). In the course of this study, we direct our attention towards these pivotal 14 firm characteristics, positioning them as the core inputs for our DL models. Our chosen methodology encompasses two distinct forms of DL techniques: Artificial Neural Networks (ANNs) and Long Short-Term Memory Networks (LSTMs). The ANN, as a foundational DL model, has enjoyed widespread application within prior literature, serving as an effective tool for forecasting stock prices and returns. Meanwhile, LSTMs exhibit a remarkable capacity to grasp intricate long-term dependencies, rendering them well-suited for handling pooled or sequential data sets.

Recent research highlights the superiority of DL techniques over linear regression models in stock return forecasting. For instance, the work of Gu, Kelly, and Xiu (2020)

engages in a comprehensive comparative analysis of machine learning approaches aimed at predicting one-month returns. Their study unveils that machine learning forecasts, underpinned by a rich reservoir of stock-level predictive attributes, yield substantial economic gains for investors when compared with conventional regression methods. Similarly, the research undertaken by Chen, Pelger, and Zhu (2021) employs DL techniques to forecast individual stock returns. Capitalizing on an abundance of conditioning information, their approach maintains a remarkable degree of flexibility and accommodates time-varying dynamics. Furthermore, Geertsema and Lu (2023) underscore the ascendancy of machine learning-based predictions in the realm of individual stock returns. Their findings indicate a marked superiority of machine learning over traditional regression-based approaches, spanning horizons from one month to a decade.

These convincing findings inspire us to replace linear regression with DL techniques as we aim to create a stronger and more detailed connection between returns and firm traits. By utilizing the capabilities of DL methods, we strive to uncover complex patterns hidden within the complex interaction of financial factors, enhancing our understanding of the diverse forces that shape stock returns.

Specifically, the DL models take in two main components from our training data. One part encompasses 14 specific characteristics (referred to as C14 hereafter), while the other includes an actual return coupled with an arbitrary return. This arbitrary return is chosen from a range of -3% to 3%, with increments of 0.2%. As a result, for a given stock's actual return at a particular time, we generate a set of 31 'realized' returns (actual return + arbitrary return). When the arbitrary return is zero, the 'realized' return remains the same as the actual return.

This setup allows us to treat the outputs as abnormal returns, which are equivalent to the corresponding arbitrary returns. We input the combination of C14 and the 'realized' return into our DL models and label the corresponding outputs. The DL models then learn the pattern between C14+realized returns and abnormal returns. During the testing phase, when we input C14 and an actual return for a stock at a given time, our DL models can predict an abnormal return. To ensure the robustness of our approach, we conduct a test where we input C14 and the actual return, using C14 as the input and the actual return as the output for the DL models. In this scenario, the output can be interpreted as the anticipated return. DL models aim to uncover the relationship between C14 and the expected return. During testing, DL models initially provide expected returns, after which we deduce an expected return from an actual return to calculate an abnormal return for a stock over a specific period.

To facilitate a comprehensive comparison between our DL models and the regression models employed by Bessembinder, Cooper, and Zhang (2019), we take measures to ensure the similarity of our C14 dataset to theirs. In pursuit of this, we employ the Bessembinder approach to compute the predicted monthly log return based on our C14 dataset. Subsequently, we retrieve the monthly returns data of Bessembinder from the Wharton research data services platform. The resulting correlation between the two datasets is 0.9768, indicating a meaningful alignment and enabling a robust performance comparison between our DL models and those of Bessembinder.

The outcomes of our study are indeed compelling. We gauge the effectiveness of producing well-specified test statistics in random samples for both DL models and benchmark models. These benchmarks encompass 'Bessembinder,' as proposed by

Bessembinder, Cooper, and Zhang (2019), 'Lyon' as utilized by Lyon, Barber, and Tsai (1999), and 'FF5F' from Fama-French (2015).

Our DL models exhibit superior performance across various dimensions and time horizons, compared to the benchmark models. While the Bessembinder benchmark demonstrates a high level of performance in this context, akin to the findings of Bessembinder, Cooper, and Zhang (2019), our DL models exhibit further improvement over Bessembinder. Notably, the DL_LSTM model emerges as the most potent performer, yielding the most robust t-statistic when we introduce a consistent level of abnormal return within the range of -3% to 3% at intervals of 0.2% for event firms on an annual basis. For instance, with an annual abnormal return of -2.8%, the rejection rates are 100% for DL_LSTM, 98.3% for DL_ANN, 91.6% for Bessembinder, 28.3% for FF5F, and 22.5% for Lyon, utilizing the '1month-3years' method to conduct the power test. Furthermore, the enhanced t-statistics exhibited by the DL models over the benchmark models remain consistent across different periods, including 1-, 3-, and 5-year intervals, employing two distinct methods, namely '1month-year(s)' and 'year-year.'

Data mining remains a persistent concern within the realm of empirical asset pricing. In our study, we undertake measures to mitigate the potential impact of data mining in the formulation of our DL model. Our approach involves a deliberate effort to ensure the robustness of our model design. We commence with the random selection of 200 stocks without replacement, followed by the random selection of a specific month denoted as "t" for each of these 200 stocks. This process is iterated 1,000 times, resulting in a cumulative total of 200,000 observations. As a consequence, only modest proportions, ranging from 8.6% to

11.3% across distinct time horizons, are drawn from the entire dataset for model training. This strategic selection of stocks of interest and the utilization of a smaller subset of data for model training collectively serve to curtail the potential for data mining.

We extend our scrutiny to assess the specifications of DL methods across both random and nonrandom samples. Lyon's documentation reveals that their refined methodologies yield aptly specified tests within random samples. However, their techniques often generate mis-specified statistical results in non-random samples, such as those featuring atypical pre-event returns or a concentration of cases within a single industry. Lyon attributes these misspecifications to the inadequacy of firm size and the book-to-market ratio in capturing the entirety of the misalignment inherent in the Capital Asset Pricing Model. In contrast, our approach, alongside that of Bessembinder, is anchored in the utilization of C14, which boasts a more comprehensive array of firm-related information. As a result, our methods exhibit well-specified test statistics in random samples, with fewer misspecifications in nonrandom samples. We conjecture that the enhancement in our study can be ascribed to the broader coverage of potential misalignments afforded by C14. Notably, both DL and Bessembinder models improve the degree of misalignment in samples characterized by unusual pre-event returns or industry concentration. Additionally, in the context of samples featuring overlapping returns, both DL and Bessembinder techniques yield appropriately specified test statistics, while the Lyon and FF5F methods exhibit signs of misalignment. Furthermore, relative to DL models in conjunction with Bessembinder, our DL models also exhibit enhanced degrees of proper specification. For instance, the DL_LSTM model attains well-specified test statistics in the non-random sample corresponding to calendar clustering,

whereas misalignments persist in the results produced by Bessembinder's approach.

To delve into the mechanisms underlying the enhanced performance of DL models, we undertake a simulation to compare the prediction errors of abnormal returns between our DL model and the Bessembinder benchmark. Through this comparative analysis, we carefully examine the impact of various characteristics on prediction error within both methodologies. Our findings reveal a distinct disparity in the Root Mean Squared Error (RMSE) metrics between the DL model and Bessembinder, measuring 0.0137 and 0.0214, respectively. This difference translates to a substantial reduction of approximately 27% in RMSE achieved by the DL model. A similar trend is discernible in the Mean Absolute Error (MAE) measurement.

To probe further, we subject the prediction error of abnormal returns to regression analysis based on relevant characteristics. The outcomes illuminate notable contrasts between the two methodologies. Within the Bessembinder framework, variables related to market conditions, firm size, and temporal indicators exhibit a significant influence on the prediction error of abnormal returns. Intriguingly, these same variables, except one of the three temporal indicators, no longer yield significant t-statistics in the DL_LSTM model. This shift implies that the performance of the DL model is relatively less affected by firm size, prevailing market conditions, and temporal factors. The patterns explained above contribute to an understanding of why our DL models outperform benchmarks in power tests across different horizons and specification experiments involving non-random samples.

Our study brings several meaningful contributions to the realm of event studies and stock return analysis. Firstly, we introduce a novel application of deep learning techniques,

specifically artificial neural networks (ANNs) and long short-term memory networks (LSTMs), to the realm of estimating abnormal returns. This introduces a more versatile and data-centric approach, diverging from the conventional linear regression models. The DL model is trained using cross-sectional data. Secondly, we offer a fresh philosophical standpoint by directly modeling the intricate interplay between abnormal returns, firm characteristics, and realized returns. This novel perspective not only sheds new light on the estimation process but also widens the scope of understanding. Thirdly, we illuminate the significant link between firm characteristics and future stock returns, thereby enriching our comprehension of the driving forces behind stock return dynamics. Moreover, we conduct a comprehensive performance comparison between our deep learning models and benchmark models, showcasing their superiority in terms of power tests. Finally, this study introduces a method and framework for training a practical AR estimator, providing researchers with a valuable tool for conducting event study analyses.

The remainder of this essay is organized as follows. Section 3.2 summarizes the literature review. Section 3.3 develops our methodology relative to ANN and LSTM models. Sections 3.4 and 3.5 analyze the data and empirical results. Section 3.6 concludes the paper.

3.2 Literature Review

Assessing the magnitude of abnormal returns and discerning their presence for specific equities constitutes a pivotal aspect within the domain of event studies. Moreover, considerable scholarly attention has been directed toward comprehending the long-run effects of such phenomena. For instance, Kothari and Warner (2008) collected 200 research papers

that have employed measurement windows extending to 12 months or more, capturing the enduring impact of events, as featured in many finance journals. This computational task hinges upon the computation of abnormal returns as the disparity between actual returns realized and expected returns. The latter requires researchers to state the expected returns, often expressed as normal returns or benchmark returns. In the context of evaluating abnormal returns, the matter of fitting methodology remains vigorously debated. Fama (1997) argues that it is hard for a single model to estimate the abnormal returns that can reflect a complete description of the systematic patterns in average returns. Kothari and Warner (2007) emphasize that the question of which model of abnormal returns is optimal remains controversial.

To date, several methodological paradigms have gained prominence, including the buy-and-hold abnormal return methodology (hereafter BHAR), the calendar time portfolios abnormal returns (hereafter CTARs), and the characteristic-based asset pricing model. A comprehensive exposition of these methodologies ensues.

Primarily, the BHARs, as elaborated by Ritter (1991), enclose the differential between the cumulative returns over one year or more for a firm's stock and the collective performance of a benchmark composed of stocks sharing akin risk attributes, tracked across the same temporal window. Ritter claims that the BHARs capture the experience of an investor who buys and holds the stock for a pre-determined period. Nevertheless, empirical investigations by Barber and Lyon (1997) and Kothari and Warner (1997) reveal that the rejection levels empirically witnessed often exceed theoretical expectations in cases where BHARs are tested for long-run abnormal stock returns. This misspecification is attributed to

three sources, namely, new listing or survivor bias, rebalancing bias, and skewness bias. In dissent, Fama (1998) raises objections against the BHAR approach, underscoring the statistical problems inherent to its utilization. Notably, Eckbo, Masulis, and Norli (2000) raise practical reservations by contending that BHARs do not offer feasible portfolio strategies due to the unavailability of the total stock count in advance. Seeking to enhance BHARs, Lyon, Barber, and Tsai (1999) elaborately devise reference portfolios that are free of the new listing and rebalancing bias and apply either a skewness-adjusted t-statistic or an empirically derived distribution of prolonged abnormal returns. While these refinements offer robust performance in random samples, they encounter pronounced misspecification issues in nonrandom samples.

Subsequently, CTARs, a frequently employed methodology in financial literature for gauging long-run abnormal returns, assume the forefront. We usually estimate CTARs by first grouping an event portfolio in which stocks share the same event and regressing the monthly returns of the resulting portfolio on factors that are known to predict stock returns, such as size, book-to-market, and momentum (Fama and French, 1993; Carhart 1997), and then abstracting the intercept that is viewed as the estimated average monthly abnormal return of the portfolio of event-firms over the T-month post-event period. Noteworthy among the advantages of the calendar time approach is its inherent accommodation of cross-sectional return correlations (Lyon, Barber, and Tsai, 1999). Arguably, Fama (1998) and Mitchell and Stafford (2000) prefer CTARs over BHARs, positing that the former's reliance on monthly returns mitigates susceptibility to model-related deficiencies. Additionally, by aggregating monthly calendar time portfolios, all cross-correlations of event-firm abnormal returns are

inherently subsumed within portfolio variance. Nonetheless, CTARs are not exempt from limitations. Loughran and Ritter (2000) caution against over-interpretation of CTAR findings, citing elevated susceptibility to Type II errors. They also contend that calendar time portfolio regressions possess low discriminatory power in identifying abnormal performance, as they combine 'hot' and 'cold' event activity over months. Furthermore, Lyon, Barber, and Tsai (1999) establish that CTARs are prone to misspecification in nonrandom samples, whereas the BHAR approach evinces relative robustness. Chava and Reguly (2024) use a modified staggered synthetic control approach to study the long-run stock market performance of acquirer firms. Their methodology matches multiple acquirer characteristics before mergers and acquisitions. Jacobson and Mizik (2009) use a variation of the calendar-time portfolio approach that incorporates time-varying risk factors, thereby abandoning the restriction that risk factor loadings remain constant over the estimation period.

Lastly, the groundbreaking work by Bessembinder, Cooper, and Zhang (2019) advances a novel methodological framework. Their approach pivots upon utilizing fitted values from market-wide regressions that capture firm returns in relation to lagged firm characteristics, thus laying the foundation for normal returns. The resultant insights hold promise, exhibiting enhanced discriminatory power for assessing the emergence of abnormal returns within a specified temporal window. Notably, this methodology derives power from contemporary studies, indicating that firm characteristics have significant explanatory and predictive power for the cross-section of stock returns. Haugen and Baker (1996) showcase the predictive potency of a set of 46 firm-level variables, while Lewellen (2015) underscores the predictability of realized returns for the subsequent month based on 15 distinct firm

characteristics. Bessembinder, Cooper, and Zhang (2019) highlight the versatility of their approach, appropriately adapting to existing relationships between characteristics and expected returns in the broader stock market context. This perspective stands as the cornerstone for comparing realized returns among event firms with their characteristic-based projected returns.

In recent years, the application of sophisticated techniques, particularly machine learning, has boomed in research endeavors. Gu, Kelly, and Xiu (2020) venture into the realm of empirical asset pricing, showcasing the substantial economic gains provided by machine learning forecasts, surpassing conventional regression-based strategies. Geertsema and Lu (2023) tread a parallel path, exploiting machine learning to predict long-run expected stock returns across diverse temporal horizons. Azevedo, Hoegner, and Velikov, M. (2023) estimate the expected returns of machine learning-based anomaly trading strategies, and their findings contrast with previous literature suggesting that machine learning strategies are unprofitable after accounting for economic constraints. Hunt, Myers, and Myers (2022) applied machine learning algorithms, specifically random forests, to predict future earnings changes and associated abnormal returns. Their approach leverages increased computing power and extensive datasets to improve forecast accuracy beyond traditional statistical methods. Similarly, Li (2023) demonstrated that combining fixed effects models with machine learning approaches, such as XGBoost, improved the accuracy, robustness, and efficiency of abnormal return estimations. These findings suggest that integrating machine learning techniques can offer more precise assessments of abnormal returns in event studies.

Traditional methods often rely on predefined financial metrics, whereas machine

learning approaches can process large, diverse datasets, capturing complex, non-linear relationships. Machine learning models adapt to data patterns without strict parametric assumptions, offering flexibility over traditional linear models. Empirical evidence may suggest that machine learning techniques can enhance predictive accuracy and abnormal return estimation compared to traditional methods.

3.3 Methodology

In our research, we use deep learning (DL) techniques, a kind of machine learning method, to investigate two relationships. Firstly, we aim to establish a connection between abnormal stock returns and factors such as company characteristics and realized stock returns. This means that given the firm's characteristics and realized return in time of t, our DL models can directly output the value of abnormal return over the periods of interest.

Secondly, we adopt a method similar to the one used by Bessembinder, Cooper, and Zhang (2019) to understand the link between anticipated returns and characteristics. The key difference is that, instead of the conventional regression approach they used in their study, we are concentrating on employing DL techniques.

In the upcoming sections, we will provide a brief overview of the relevant DL techniques and explain the DL models we've developed for our study.

3.3.1 Deep Learning (DL) Techniques

We choose two kinds of classic deep learning (DL) models for our DL-based abnormal estimation molders, which are Long Short-Term Memory (LSTM) and Artificial Neural Network (ANN).

3.3.1.1 LSTM

The Long Short-Term Memory (LSTM) network, introduced by Hochreiter and Schmidhuber (1997), is a specialized type of recurrent neural network (RNN) developed to overcome limitations of traditional RNNs, such as vanishing gradients and difficulty modeling long-term dependencies. Its unique architecture enables effective learning over extended sequences, making it well-suited for tasks involving temporal or sequential data.

LSTMs have proven particularly valuable in applications like language processing and time series analysis. In financial forecasting, for example, LSTMs can capture patterns and dynamics in historical stock prices across multiple time steps, enhancing the prediction of directional trends with improved accuracy.

The transformative potential of LSTMs is evidenced by their notable efficacy across diverse domains, a consequence of their inherent competence in managing extensive temporal dependencies and mitigating the challenge posed by vanishing gradients. The practical impact of LSTMs spans an array of domains, encompassing speech recognition, language translation, financial analysis, medical diagnostics, and beyond. Their pervasive adoption within academic and industrial spheres attests to their broad utility, as they have evolved into versatile instruments embraced by researchers and data scientists alike. The ongoing evolution of LSTMs has engendered the emergence of specialized iterations attuned to

particular problem contexts, thereby underscoring their adaptability to heterogeneous challenges.

The Long Short-Term Memory (LSTM) model, as illustrated in Figure 3.1, is designed around a recurring memory block, or cell, that efficiently regulates information flow through three gating mechanisms: the forget gate, the input gate, and the output gate. These gates work in conjunction with a cell state, which serves as a memory unit, allowing the model to retain, update, or discard information as needed. The forget gate determines what portion of past information should be discarded, while the input gate decides which new information should be added to the cell state. The output gate then regulates what information is passed forward as the next hidden state h_t.

In this structure, the present input x_t and the previous hidden state h_{t-1} interact through weight matrices and activation functions—such as the hyperbolic tangent (tanh)-to modulate the memory cell's state. This enables LSTMs to effectively capture long-term dependencies in sequential data, making them highly suitable for tasks such as time-series forecasting, natural language processing, and financial market predictions. The gating mechanism

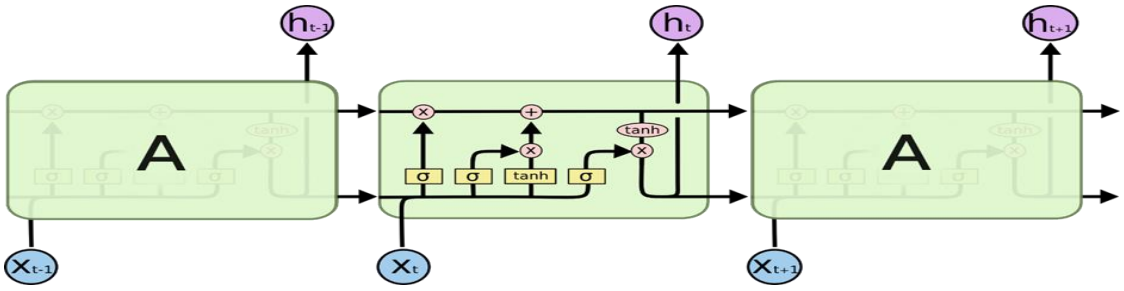

Figure 3.1 The Internal Structure of an LSTM (Colah (2015))

mitigates the vanishing gradient problem, a common issue in standard recurrent neural

networks (RNNs), thereby enhancing LSTMs' ability to learn complex temporal patterns over extended sequences.

3.3.1.2 ANN

We employ a multi-layer Artificial Neural Network (ANN) as our deep-learning framework. First formalized by Rumelhart, Hinton, and Williams (1988), ANNs routinely deliver state-of-the-art performance across varied machine-learning tasks by learning complex, nonlinear relationships between inputs and outputs. Their feed-forward architecture—an input layer, several hidden layers, and an output layer—makes them ideal for structured, tabular data such as financial indicators used to forecast stock-price direction.

In an ANN, the input layer mirrors the feature set, the hidden layers extract higher-order patterns, and the output layer returns either class probabilities or continuous values. Because no universal rule dictates the optimal number of hidden layers or neurons (Wang & Sun 1996), network size is tuned empirically. Inspired by biological neurons, ANNs function as computational approximations of neural behavior and, with sufficient data and careful training, offer a flexible, powerful alternative to traditional linear models in stock-return prediction.

Here is an example of an ANN (Figure 3.2):

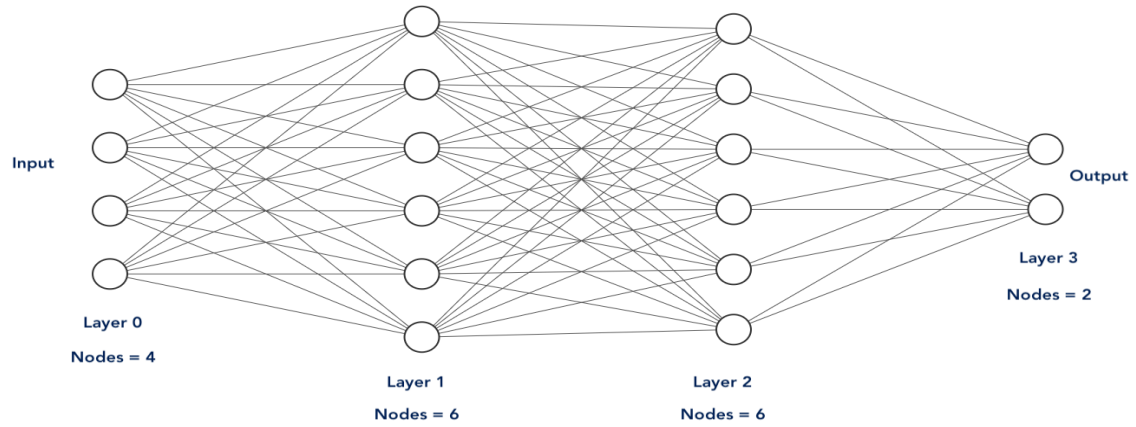

Figure 3.2 ANN Architecture

This ANN model (Figure 3.3.2) has 2 hidden layers. Each neuron within a layer could compute a linear combination of its inputs, followed by a nonlinear transformation. In general, the function of 3.3.1 can be used to describe the model of an ANN.

$$y_i = f\left(\sum_{j=1}^{N} w_{ij}x_j + b_i\right), \qquad i, j = 1,2,\ldots,N \qquad (3.3.1)$$

Central to this model is the summation of input variables, influenced by associated weights, and subsequently activated by an activation function. The transfer of input variables from node "j" to node "i" is described by "x_j" (representing independent variables) and "w_{ij}" (representing weights). The activation function, denoted as "f," governs this process, culminating in a dynamic and adaptive model capable of grasping intricate relationships within data.

3.3.2 DL Model Design

We initiate the training of our Deep Learning (DL) models employing a designated cross-sectional training dataset. The structure of this training dataset is as follows:

(a) We commence by the random selection, without replacement, of 200 distinct stocks.

(b) Subsequently, we proceed to randomly designate a specific month denoted as "t" of interest for each of the aforementioned 200 selected stocks.

(c) This process is iteratively executed 1,000 times, thus culminating in an accumulation of 200,000 observations.

(d) To augment the dataset, we undertake the extension of the aforementioned observations. This augmentation involves the deliberate addition of varying percentages, spanning from -3% to 3%, to the realized returns. This step effectively diversifies the dataset and fosters its richness.

As a result of these cumulative operations, the training dataset attains a substantial magnitude, comprising a total of 6,200,000 entries.

Our dataset spans from January 1970 to December 2021, with a specific focus on the interval from January 1980 to December 2021.

It is relevant to acknowledge that while the training data is a valuable subset, it constitutes merely a fractional representation of the complete dataset. Specifically, this training data accounts for a proportion ranging between 8.6% to 11.3% of the entirety. This selection is tailored towards facilitating the prediction of returns across disparate time horizons. In order to gauge the efficacy of the DL models comprehensively, we conduct an evaluation utilizing the entire dataset.

3.3.2.1 Artificial Neural Networks (ANN) Input and Output

The input and output are as follows:

The input X = C14 + 'realized' return, where the 'realized' return equals the realized return plus an arbitrary return. Here, C14 is the 14 firm characteristics proposed by Lewellen (2015). The selection of 14 firm-specific characteristics as explanatory variables in deep learning (DL) models is justified based on their empirical relevance in asset pricing and return prediction, as demonstrated in prior research by Lewellen (2015). These characteristics—such as size, book-to-market ratio, momentum, profitability, and investment—are widely recognized for their explanatory power in capturing systematic variations in stock returns.

Lewellen (2015) highlights their significance in explaining cross-sectional return differences, while Bessembinder et al. (2019) further emphasize their role of C14 in distinguishing between high- and low-performing stocks. Furthermore, these firm-specific attributes align with economic theories, such as the Fama-French multifactor models and behavioral finance perspectives, which suggest that return anomalies stem from fundamental firm characteristics.

The output y = abnormal return = arbitrary return

Specifically, our Artificial Neural Networks (ANN) designed for turning point prediction consists of an input layer with 15 neurons, corresponding to 15 variables representing C14 and 'realized' return. It includes three hidden layers with 128, 64, and 32 neurons, respectively, each using the ReLU (Rectified Linear Unit) activation function to

introduce non-linearity. To prevent overfitting, dropout layers with a rate of 0.2 are applied after each hidden layer. The model is trained using the Adam optimizer with a learning rate of 0.001 and employs mean squared error (MSE) as the loss function. Batch normalization is incorporated to stabilize learning, and L2 regularization (lambda = 0.01) is used to control model complexity. The output layer consists of a single neuron with a linear activation function, SoftMax, predicting the arbitrary return. The model is trained over 100 epochs with a batch size of 32 to ensure optimal generalization and performance.

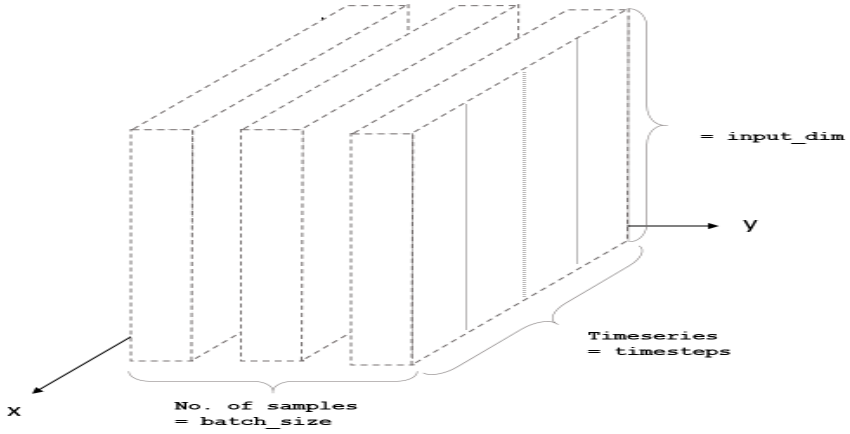

Figure 3.3 Configuration of Input and Output

3.3.2.2 LSTM Input and Output

The diagram (Figure 3.3) presented above illustrates the configuration of input and output within the Long Short-Term Memory (LSTM) framework. In our research context, the parameter "input_dim" pertains to the input denoted as "X" and the time step "τ" is set at 12 months. Consequently, for a given time "t=12", an observation encompasses the sequence (X_{τ}, $X_{\tau-1}$, $X_{\tau-2}$, ..., Xτ-11) as the input, which is the sequence of (C14 + 'realized' return). Output is

the same as one in the ANN model, an arbitrary return.

In our specific Long Short-Term Memory (LSTM) model configuration, we use 15 features as input variables and a time step of 12, meaning the model processes sequential data across 12 time periods to capture temporal dependencies effectively. The architecture consists of an input layer that feeds data into two stacked LSTM layers, each with 64 hidden units, enabling the model to learn long-term dependencies in financial time series. A dropout layer with a rate of 0.2 follows each LSTM layer to prevent overfitting. The output from the LSTM layers is passed through three fully connected dense layers, with 128, 64, and 32 neurons, respectively, followed by a final output layer with a single neuron for the prediction of the. The model is optimized using the Adam optimizer with a learning rate of 0.001, and the loss function is set to mean squared error (MSE) to measure prediction accuracy.

In summary, our analysis encompasses four distinct target returns, expressed in logarithmic form, spanning four different time horizons: 1 month, 1 year, 3 years, and 5 years. This leads to the creation of a total of four Artificial Neural Network (ANN) models and four Long Short-Term Memory (LSTM) models – each correspondingly aligned with a specific target return horizon.

In essence, the integration of "C14" (variables) and the realized return into our Deep Learning (DL) models culminates in a significant outcome. Specifically, the DL models, whether ANN or LSTM, are adept at predicting abnormal returns, underscoring their predictive prowess and capacity to unveil meaningful insights within our analytical framework.

3.4 Data

We gather stock data from the CRSP monthly file and the Compustat annual file, accessible through the Wharton Research Data Services (WRDS) platform. Our dataset spans from January 1970 to December 2021, with a specific focus on the interval from January 1980 to December 2021. This narrower period facilitates the incorporation of up to a decade of preceding data for the construction of C14, aligning with Bessembinder's temporal approach. Summary statistics for firm characteristics, measured monthly, are provided in Table A.2.2 within the Appendix.

To ensure the comparability of coefficients across different firm characteristics and time periods, we employ a normalization technique inspired by Bessembinder. This involves subtracting the cross-sectional mean for each month and then dividing it by the cross-sectional standard deviation for that same month. Consequently, all firm characteristics exhibit a mean of zero and a standard deviation of one for each month. Table 3.1 presents the average coefficients associated with firm characteristics derived from the regression outlined in equation (3.4.1). The obtained results exhibit a notable similarity to prior findings, as presented by Levellen (2015) and Haugen and Baker (1996). Notably, the signs of coefficients for the 1-month horizon align with those documented by Bessembinder, Cooper, and Zhang (2019).

$$R_{it} = \alpha_t + \beta_t X_{i,t-1} + \varepsilon_{it} \qquad (3.4.1)$$

where R_{it} is stock i's realized log return in month t, and $X_{i,t-1}$ is a vector of firm i's characteristics measured at the end of month t-1.

Table 3.1 Average Coefficients on Each Firm Characteristic Across the Sample Period, January 1980 to December 2021

Each month, we estimate cross-sectional regressions of firm monthly simple and log stock returns on characteristics measured at the end of the preceding month as specified in Equation (3.4.1). This table presents average coefficients over time. Firm characteristics are winsorized within each month at the upper and lower 1% and are normalized by subtracting the mean and dividing by the standard deviation. See Table A.2.1 for variable definition. The Fama-MacBeth standard errors are based on the time-series variability of the estimates, incorporating a Newey-West correction with four lags. The associated t-statistics are reported in the parentheses below each coefficient. ***, **, and * correspond to statistical significance at the 1%, 5%, and 10% levels, respectively.

Dep. Var.	Monthly or annually mean log return			
	1 month	1-year	3- year	5-year
log size	-0.1315***	-0.9399***	-1.3149***	-0.8544***
	-2.68	-5.08	-12.46	-4.21
log BM	0.4415***	1.1722***	1.6137***	1.2584***
	12.63	6.77	15.31	7.62
Momentum	0.3539***	0.4052***	0.1589***	0.2581***
	52.79	19.15	14.24	14.24
ROA	0.0656**	0.3855***	0.2874***	0.3785***
	2.15	8.46	4.58	12.58
Asset growth	-0.4918***	-2.4171***	-1.7504***	-1.3842***
	-3.81	-11.85	-10.22	-6.19
Beta	0.0114	-0.1425	-0.4725***	-0.1757
	0.07	-0.54	-4.49	-1.08
Accrual	-0.0337	-0.3139***	-0.1544***	-0.2607***
	-1.59	-5.61	-4.15	-4.87
Dividend	0.1245***	0.1121	-0.0319	-0.0155
	4.75	1.2	-0.56	-0.21
log LR return	-0.1814***	-0.9381***	-0.0057	-0.6175***
	-4.51	-4.38	-0.56	-3.56
Idio. Risk	-0.5583***	-1.09381***	-0.9031***	-0.4155***
	-7.22	-4.38	-5.65	-9.62
Illiquidity	0.5374***	2.5357***	1.3852***	1.0129***
	8.34	16.01	16.81	14.95
Turnover	-0.2987***	-2.246***	-1.4545***	-0.8854***
	-4.59	-16.65	-14.55	-6.21
Leverage	-0.0591***	-1.1049***	-0.6811***	-0.4428***
	-2.04	-10.03	-10.93	-7.59
Sales/price	0.1401***	0.9147***	0.4023***	0.5214***
	5.56	6.39	4.11	5.04
Constant	0.3471	4.3362***	3.4129***	2.871***
	1.39	4.56	7.91	2.69
R2	0.081	0.075	0.084	0.068
Number	2,322,253	2,242,253	2,094,279	1,970,654

To ensure the similarity of our C14 dataset with that employed by Bessembinder, we undertake a two-fold verification process. Firstly, we replicate the Bessembinder methodology to compute the predicted monthly log return using our own C14 dataset. Second, we obtain Bessembinder's monthly return data from the WRDS platform and compare it to our calculated returns. The resulting correlation coefficient between the returns obtained from these two sources is 0.9768, attesting to a significant level of similarity between the data that we have employed and the dataset organized by Bessembinder.

3.5 Results

3.5.1 Power Test

3.5.1.1 Power Test in Random Samples

To assess the power of the two DL methods and three other benchmark models that exhibit well-specified test statistics in random samples, we conduct a comprehensive experiment. First, we estimate abnormal returns for all stock-month observations using CRSP data, applying all estimators, including DL models, the Bessembinder approach, and other benchmark methods. Next, for each firm/month selected as a pseudo-event, we artificially introduce a constant abnormal return—ranging from +3% to -3% in 0.2% increments on an annualized basis—added to its estimated abnormal return. We run the regression of 3.5.1 and then proceed to document the empirical rejection rates at the theoretical significance level of 5 percent. The null hypothesis under scrutiny asserts that the mean sample long-run (1 year, 3 years, and 5 years) abnormal return is zero. This assessment is carried out across the 1,000

simulations. The outcomes of this experiment are visually presented in Figure 3.4.

In accordance with the approach advocated by Bessembinder, Cooper, and Zhang (2019), we adopt a two-stage method to scrutinize the abnormality of average returns for specific firms during time periods of interest that are abnormal.

Specifically, in the initial stage, we compute the abnormal return (AR) for all stocks in each month. Employing DL methods, we utilize pre-trained DL_LSTM and DL_ANN models. These models are employed to estimate the logarithmic abnormal return across four distinct time spans: 1 month, 1 year, 3 years, and 5 years into the future, encompassing all stock months. For the remaining three benchmark models, we adhere to the methodologies outlined in their respective research papers. In these cases, we calculate the anticipated logarithmic return and subsequently subtract it from the realized logarithmic return. This differential value provides the abnormal return for all stock months across the mentioned four-time horizons for Bessembinder and Lyon, and FF5F. Comprehensive details about the benchmark methodologies are described in Appendix A.2.2.

In the second stage, we regress, again using all stocks, estimated abnormal returns from the DL models or others on a constant and an indicator variable that is set to one for firms/months of interest, and zero for other firms/months, which is expressed in equation 3.5.1.

$$AR_{it} = a + b^{*} D_{it} + u_{it}\,, \qquad (3.5.1)$$

where D_{it} is an indicator that equals one if firm i experienced a corporate event, and zero otherwise. AR_{it} is the estimated abnormal log return for stock i, at time t.

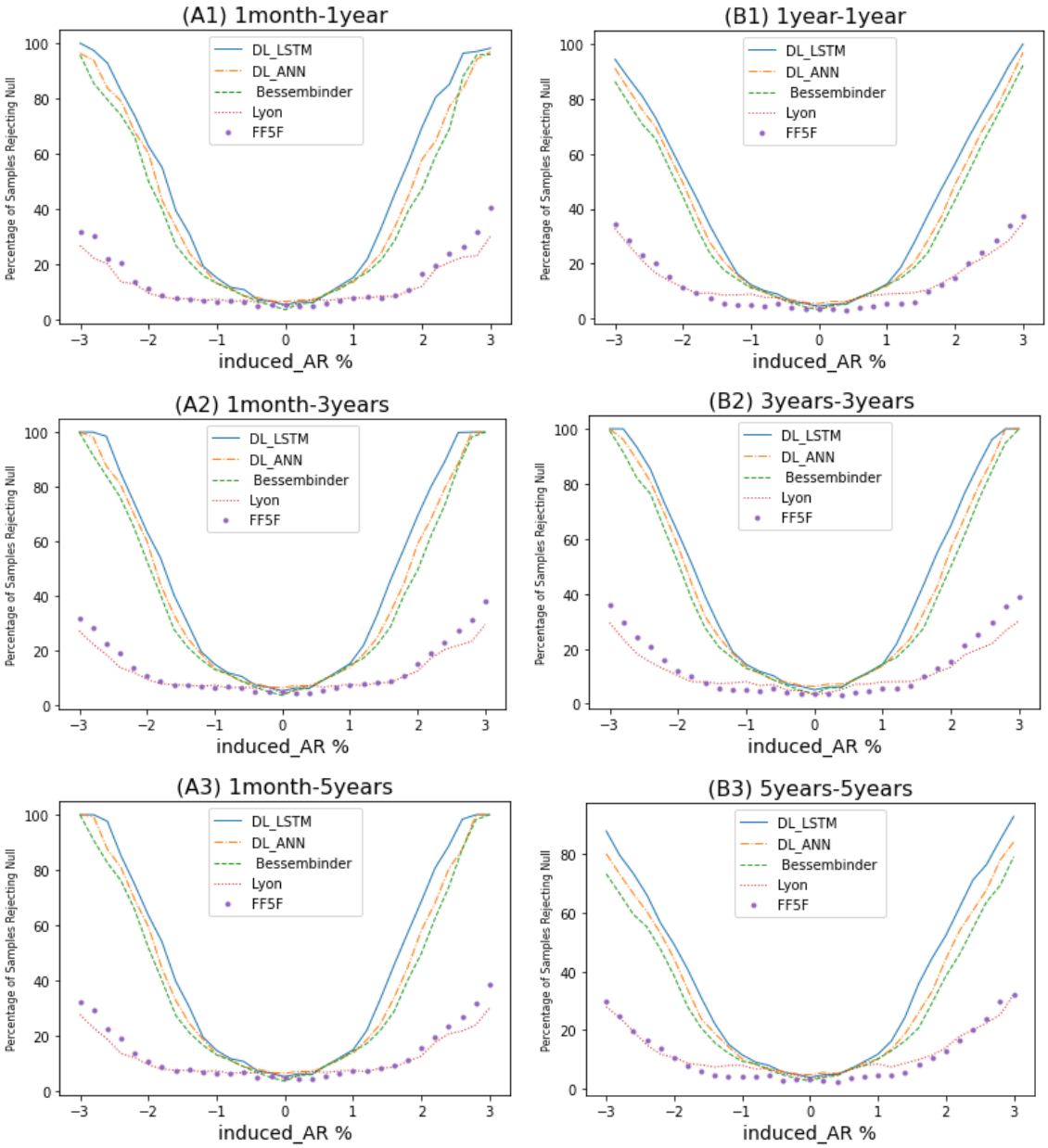

Figure 3.4 The Power of Test Statistics in Random Samples

This figure represents the percentage of 1,000 random samples of pseudo-event firms rejecting the null hypothesis of no annualized abnormal return at various yearly induced levels of abnormal return (horizontal axis). We conduct a two-stage method. In the first stage, we calculate the abnormal return for all stock months over various periods, including 1 month, 1 year, 3 years, and 5 years. For yearly returns, we use annualized returns. DL_LSTM and DL_ANN refer to DL models with LSTM and ANN, respectively. Bessembinder refers to the method proposed by Bessembinder and Cooper, and Zhang (2019). Lyon refers to the method of buy-and-hold size/book-to-market portfolio used by Lyon, Barber, and Tsai (1999). FF5F refers to the method from Fama-French (2015). In the second stage, we conduct the following regression: $AR_{it} = a + b^* D_{it} + u_{it}$, Where D_{it} is an indicator that equals one if firm i experienced corporate event, and zero otherwise. AR_{it} is the log abnormal return for stock i, at time t. For (A1-A3) in this figure, AR_{it} is the monthly return for the methods of 1month-1year, 1month-3years, and 1month-5years. We randomly choose ten pseudo-event firms in each month from January 1980 to December 2021. The indicator variable (D_{it}) equals one if the firm was selected as a pseudo-event firm during any of the prior 12 months, 36 months, or 60 months for the horizon of 1 year, 3 years, or 5 years, respectively, and zero otherwise. For (B1-B3) in Figure 3.4, we randomly choose 360 pseudo-event firms in each month from January 1980 to December 2016. In these cases, AR_{it} is the annualized one-, three-, or five-year abnormal return, respectively for the method of 1year-1year, 3years-3years, and 5years-5years, and D_{it} equals one if the firm was selected as a pseudo-event firm, and zero otherwise.

Regarding (A1–A3) in Figure 3.4, our evaluation uses monthly logarithmic abnormal returns to assess performance over 1-, 3-, and 5-year horizons, denoted as 1month–1year, 1month–3years, and 1month–5years. Here, AR_{it} represents the monthly abnormal return. The simulation proceeds as follows: from January 1980 to December 2021, ten pseudo-event firms are randomly selected each month. The indicator variable D_{it} equals one if a firm was selected as a pseudo-event firm in the past 12, 36, or 60 months—corresponding to the respective horizons—and zero otherwise. For each selected firm/month as a pseudo-event firm, an arbitrary abnormal return between -3% and +3% (in 0.2% annual increments) is added to its estimated abnormal return. We then run regression (3.5.1) to estimate the coefficient b on D_{it}. A significant t-statistic associated with $\hat{b}$ suggests that the null hypothesis of no abnormal return can be rejected. This process is repeated 1,000 times.

In relation to (B1-B3) within Figure 3.4, our focus shifts to the adoption of annualized 1-year, 3-year, or 5-year abnormal returns. These returns are respectively scrutinized to ascertain the presence of abnormal returns within the corresponding temporal horizons. For instance, the scenario "3years-3years" denotes the utilization of annualized 3-year abnormal returns computed using five distinct methods. This assessment aims to determine if abnormal returns are observable within the subsequent 3-year period. In these three instances, we randomly select 360 pseudo-event firms each month from January 1980 to December 2016. The indicator variable (D_{it}) adopts a value of one if the firm is chosen as a pseudo-event firm in specific month, and zero otherwise. AR_{it} denotes the annualized one-, three-, or five-year abnormal return for the respective cases of 1year-1year, 3years-3years, and 5years-5years.

Figure 3.4 presents the following key insights: (a) DL models consistently exhibit

superior performance in comparison to Bessembinder, Lyon, and FF5F methods across all six plotted scenarios. (b) Upon contrasting the two groups labeled as "A" and "B," a visible pattern emerges. The models within group "A" not only perform better but also display greater stability compared to those in group "B." To illustrate, in scenario A3, both DL_LSTM and Bessembinder models confidently reject the null hypothesis at a 100% rate when the induced level of abnormal return is set at -3%. However, in scenario B3, the corresponding rejection percentages decline to 87.69% for DL_LSTM and 73.16% for Bessembinder. Furthermore, while DL and Bessembinder models maintain relatively consistent rejection levels within the "A" group (A1, A2, and A3), these levels exhibit substantial variation within the "B" group. Notably, B2 outperforms B1 and B3, particularly when the absolute induced abnormal returns surpass the threshold of 2%. (c) Across all six scenarios, DL_LSTM consistently emerges as the top performer, exhibiting the most favorable performance outcomes.

To quantitatively assess the performance of power, we opt for the two best scenarios in Figure 3.4, which are ‘1month-3years’ and ‘3years-3years’. Table 3.2 displays the rejection ratios on the null hypothesis of no abnormal return for three types of models and their differences in the rejection ratio as well as these two scenarios. Column 1 of Table 3.2 lists the principal induced abnormal returns (denoted as Induced_AR in Table 3.2), and columns 2, 3, and 4 exhibit rejection ratios for the models of DL_LSTM, ANN, and Bessembinder, respectively. Columns 5, 6, and 7 report the differences in the rejection ratio according to these models for the comparison of performance among them.

Results in column 4 of Table 3.2 reveal that the differences in rejection ratio are

notably significant at the 1% level when the percentages of induced abnormal returns are -2.6, -2.0, -1.6, 1.6, 2.0, and 2.6. By considering Figure 3.4 alongside this analysis, it becomes

Table 3.2 Rejection Ratio on the Null Hypothesis in Random Samples

This table presents rejection ratios on the null hypothesis of no annualized abnormal return at the 3-years period when inducing levels of abnormal returns (Induced_AR in this table). LSTM, ANN, and Be stand for rejection ratios for models of DL_LSTM, DL_ANN, and Bessembinder, respectively. LSTM-Be, LSTM-ANN, and ANN-Be stand for the difference in the rejection ratio between corresponding methods. For instance, LSTM-Be represents the difference in rejection ratio for the DL_LSTM and Bessembinder models. ***, **, and * correspond to statistical significance at the 1%, 5%, and 10% levels, respectively. The associated t-statistics are reported in parentheses.

	Rejection ratio on the null hypothesis (%)					
	LSTM	ANN	Be	LSTM-Be	LSTM-ANN	ANN-Be
Induced_AR (%)						
				Panel A: 1month-3years		
-3.0	100	100	100	0	0	0
-2.6	98.5	87.6	83.9	14.6*** (11.93)	10.9*** (9.80)	3.7** (2.37)
-2.0	63.5	60.1	52.8	10.7*** (4.88)	3.4 (1.56)	7.3*** (3.31)
-1.6	40.4	33	27.7	12.7*** (6.06)	7.4*** (3.44)	5.3*** (2.58)
-1.0	15.1	13.6	13.1	2.0 (1.28)	1.5 (0.96)	0.5 (0.32)
0.0	5.2	5.5	4.3	0.9 (0.95)	-0.3 (-0.31)	1.2 (1.24)
1.0	15.2	13.8	14.8	0.4 (0.25)	1.4 (0.88)	-1 (-0.63)
1.6	45.8	34.2	28.8	17*** (7.98)	11.6*** (5.32)	5.4*** (2.61)
2.0	69.3	59.1	49.4	19.9*** (9.25)	10.2*** (4.78)	9.7*** (4.37)
2.6	99.8	88.6	87.4	12.4*** (11.7)	11.2*** (11.03)	1.2 (0.82)
3.0	100	100	100	0	0	0
				Panel B: 3years-3years		
-3.0	100	100	99.4	0.6** (2.45)	0	0.6** (2.45)
-2.6	93.3	88.7	82.1	11.2*** (7.71)	4.6*** (3.61)	6.6*** (4.2)
-2.0	62.3	57.5	51.5	10.8*** (4.91)	4.8** (2.19)	6.0*** (2.70)
-1.6	38.9	27.6	31.7	7.2*** (3.38)	11.3*** (5.40)	-4.1** (-2.01)
-1.0	14.4	12.8	13.6	0.8 (0.52)	1.6 (1.04)	-0.8 (-0.53)
0.0	4.9	5.2	3.9	1.0 (1.09)	-0.3 (-0.31)	1.3 (1.40)
1.0	14.1	13.7	14.3	-0.2 (-0.13)	0.4 (0.26)	-0.6 (-0.39)
1.6	43.3	33.1	27.5	15.8*** (7.49)	10.2*** (4.73)	5.6*** (2.73)
2.0	64.8	56.7	49.9	14.9*** (6.81)	8.1*** (3.72)	6.8*** (3.05)
2.6	96.1	88.6	84.7	11.4*** (8.82)	7.5*** (6.37)	3.9*** (2.57)
3.0	100	100	100	0	0	0

evident that the power of test statistics for DL_LSTM is significantly larger than the one for Bessembinder when Induced_AR values fall within the absolute range of 1.6 to 2.6. This holds true for both scenarios of ‘1month-3years’ and ‘3years-3years’.

The data in columns 5 and 6 show a similar pattern to column 4. The performance of the power of LSTM dominates that of ANN, and ANN has a better performance in power test than Bessembinder, when Induced_AR values lie within the absolute range of 1.6 to 2.6. This observation remains consistent for both scenarios ‘1month-3years’ and ‘3years-3years’. Furthermore, we computed these differences in rejection ratios for an additional four scenarios, and the pattern remains consistent. For the sake of conciseness, the outcomes are not presented in this context.

Our DL model outperforms the Bessembinder models, which rely on linear regression for prediction. This superiority probably stems from DL's ability to handle diverse functional forms, enabling it to capture complex, non-linear relationships between dependent and independent variables. Unlike linear models that assume a fixed relationship, DL models learn patterns directly from data without imposing strict assumptions about the input-output linkage. Additionally, DL approaches often incorporate cross-validation and regularization techniques, enhancing their robustness and improving predictive accuracy.

Researchers can apply this methodological approach to train more advanced DL models and make them available for others conducting event studies. By leveraging DL's ability to capture complex, non-linear relationships, these models can enhance the accuracy of abnormal return estimations. Providing well-trained models allows researchers to bypass the need for extensive model development, streamlining the analysis process and ensuring

consistency across studies. This collaborative approach fosters methodological advancements and improves the reliability of event study findings in financial research.

3.5.1.2 Power Test in Non-Random Samples

Our objective is to ascertain whether the performance of DL models surpasses that of the benchmark Bessembinder within two non-random sample sets. In this context, we shall present findings related to two distinct sample categories, categorized by firm sizes and prevailing market conditions.

To begin, we organize the entire dataset according to the firm sizes, represented as the natural logarithm of the market capitalization of stocks. Following this arrangement, we label the top and bottom deciles as indicators of large and small firms, respectively. Subsequently, we utilize the '1 month-3 years' approach, applying monthly predictions of either predicted return or predicted abnormal return in a simulation process akin to that described in section 3.5.1.1. This methodology is implemented for both sets of samples, encompassing both large and small firms.

We adopt the procedure outlined in Bessembinder, Cooper, and Zhang (2019) to categorize the entire dataset into two distinct groups: "Hot" and "Cold," which symbolize different market conditions. Specifically, a month is classified as "Hot" if the Bake and Wurgler (2006) investor sentiment index surpasses the median value across the period of 1980-2018; otherwise, it is designated as "Cold." Given the potential for data gaps in stock-specific monthly records within the sample set influenced by hot or cold markets, we opt for the '3years-3years' approach elaborated upon in section (3.5.1.1) to evaluate the power

performance.

Figure 3.5 depicts the outcomes of the power of test statistics conducted on two distinct samples categorized by firm size and market condition. In general, the rejection ratios for

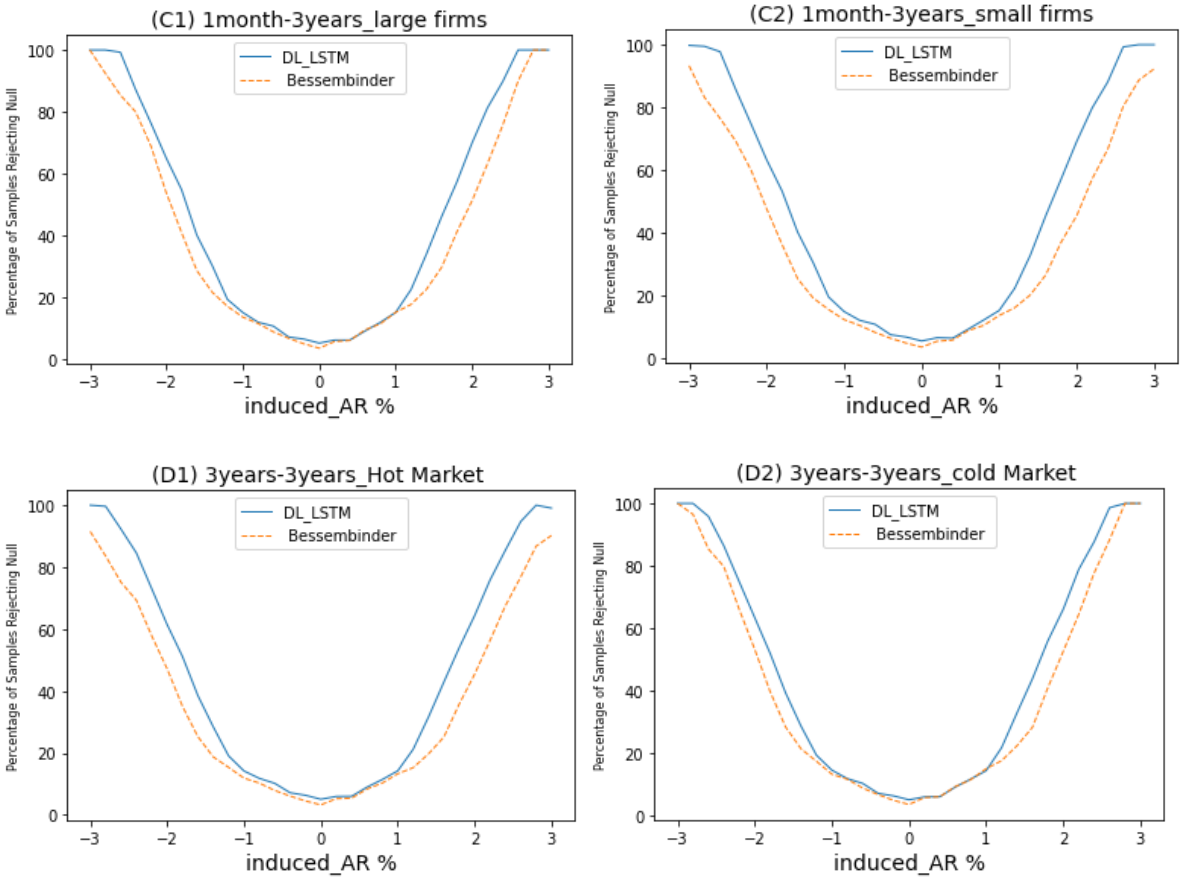

Figure 3.5 The Power of Test Statistics in Non-Random Samples

This figure represents the percentage of 1,000 random samples of pseudo-event firms rejecting the null hypothesis of no annualized abnormal return at 3-year horizon yearly induced levels of abnormal return (horizontal axis). We conduct a two-stage method. In the first stage, we calculate the abnormal return for all stock months over two periods, including 1 month and 3 years. For 3-year returns, we use annualized returns. DL_LSTM refers to DL models with LSTM. Bessembinder refers to the method proposed by Bessembinder and Cooper, and Zhang (2019). In the second stage, we conduct the following regression: $AR_{it} = a + b^* D_{it} + u_{it}$, Where D_{it} is an indicator that equals one if firm i experienced corporate event, and zero otherwise. AR_{it} is the log abnormal return for stock i, at time t. For (C1-C2) in this figure, AR_{it} is the monthly return for the method of 1month-3years. We randomly choose ten pseudo-event firms in each month from January 1980 to December 2021. The indicator variable (D_{it}) equals one if the firm was selected as a pseudo-event firm during any of the prior 36 months, and zero otherwise. For (D1-D2) in Figure 8, we randomly choose 360 pseudo-event firms in each month from January 1980 to December 2016. In these cases, AR_{it} is the annualized three-year abnormal return for the method of 3years-3years, and D_{it} equals one if the firm was selected as a pseudo-event firm, and zero otherwise.

DL_LSTM exhibits a notable superiority over those for Bessembider when induced AR values lie within the absolute range of 1.6 to 2.6. This pattern holds across both sample pairs, aligning consistently with the findings observed in the random samples. Furthermore, by

examining C1 and C2, it becomes evident that discernible disparities exist between large and small firms concerning the power performance of the Bessembinder model. Notably, the Bessembinder model performs more effectively within the subset of larger firms compared to its performance within smaller firms. Similarly, the potency of the Besembinder model demonstrates variation across distinct market conditions, with higher rejection levels attained during months characterized by colder market conditions. Of particular interest is the remarkable consistency of the DL_LSTM model's rejection levels across both samples, signifying a stable performance across varying firm sizes and market conditions. To provide a comprehensive assessment of the power's stability, we present the corresponding data in Table 3.3, focusing on the relative rejection ratios where induced AR values lie within the absolute range of 1.6 to 2.6.

Referring to the insights collected from Figure 3.5, we proceed to reinterpret the rejection ratios data concerning the null hypothesis of annualized abnormal return across various non-random samples during the 3-year timeframe. Specifically, we focus on instances where induced AR assumes values of -2.6, -2.0, -1.6, 1.6, 2.0, and 2.6. Within Panel A of the representation, the fourth column delineates the distinctions in rejection ratios between the sample comprising large firms and the one involving small firms, utilizing the method of DL_LSTM. Conversely, the seventh column shows the corresponding differences for bessembinder. It becomes apparent that the variances in rejection ratios for method DL_LSTM are markedly narrower in comparison to those seen for Bessembinder. This observation indicates that DL_LSTM is capable of maintaining a consistent performance across both the subsets of large and small firms, thereby showcasing stability in its testing

outcomes. On the contrary, significant disparities in rejection ratios exist between these two subsamples for Bessembinder, underscoring that Bessembinder 's performance differs between the scenarios and lacks stability.

Table 3.3 Rejection Ratio on the Null Hypothesis in Non-Random Samples

This table presents rejection ratios on the null hypothesis of no annualized abnormal return about non-random samples, over the 3-years period when inducing levels of abnormal returns (Induced_AR in this table). Panel A is relative to the non-random samples with large and small firms, while Panel B displays results for other non-random samples related to hot and cold market conditions. Diff(L-S) refers to the difference in rejection ratios between the large and small firms, Diff(C-H) represents the difference in rejection ratio between the cold and hot market conditions. ***, **, and * correspond to statistical significance at the 1%, 5%, and 10% levels, respectively. The associated t-statistics are reported in parentheses.

	Rejection ratio on the null hypothesis (%)					
Induced_AR (%)	DL_LSTM			Bessembinder		
	Panel A: Samples of Large and Small Firms					
	Large	Small	Diff（L-S)	Large	Small	Diff(L-S)
-2.6	99.26	97.75	1.51***	84.59	76.44	8.15***
			(2.79)			(4.63)
-2.0	65.09	63.23	1.86	53.37	47.56	5.81***
			(0.87)			(2.60)
-1.6	40.2	40.01	0.19	28.50	25.19	3.31*
			(0.00)			(1.69)
1.6	46.17	45.26	0.91	29.51	26.34	3.17
			(0.41)			(1.58)
2.0	70.08	69.03	1.05	50.89	45.29	5.6**
			(0.51)			(2.51)
2.6	100	99.27	0.73***	89.05	80.24	8.81***
			(2.71)			(5.51)
	Panel A: Samples of Hot and Cold Market					
	Hot	Cold	Diff(C-H)	Hot	Cold	Diff(C-H)
-2.6	93.25	95.85	2.60***	77.09	85.41	8.32***
			(2.57)			(4.79)
-2.0	61.43	63.49	2.06	47.14	53.35	6.21***
			(0.95)			(2.78)
-1.6	38.5	39.44	0.94	25.24	28.61	3.37*
			(0.43)			(1.75)
1.6	42.89	43.89	1.00	25.12	28.93	3.81*
			(0.45)			(1.87)
2.0	64.33	66.16	1.83	45.41	52.7	7.29***
			(0.86)			(3.27)
2.6	96.73	98.62	1.89***	78.84	88.36	9.52***
			(2.81)			(5.8)

Moving on to Panel B of Table 3.3, we present the rejection ratios for two subgroups categorized under hot or cold market conditions. The results in columns 4 and 7 mirror the patterns observed in Panel A. This consistency emphasizes the likelihood that method DL_LSTM might exhibit a more uniform power in statistical testing across samples characterized by distinct market conditions, compared to Bessembinder.

3.5.2 Specification

3.5.2.1 Specification in Random Samples

We follow the two-stage method narrated in section 3.5.1. First, we estimate the monthly log abnormal return (AR) for the five methods mentioned in the above section. Second, we conduct the regression shown in equation (3.5.1). We randomly choose ten pseudo-event firms in each month from January 1980 to December 2021. The indicator variable (D_{it}) equals one if the firm was selected as a pseudo-event firm during any of the prior 12, 36, or 60 months, respectively for one-, three-, and five-year horizons, and zero otherwise. AR_{it} is the monthly abnormal return. Then, repeat the selection 1,000 times.

The specification of the five methods at one-, three-, and five-year horizons is presented in Table 3.4. Findings reveal that all the t-statistics based on methods of DL_LSTM, DL_ANN, and Bessembinder are well specified without negative or positive bias over three horizons of one-, three-, and five-year, in 1%, 5%, and 10% significant levels. In other words, the methods of DL_LSTM, DL_ANN, and Bessembinder can yield well-specified test statistics in random samples. For the other two benchmarks, Lyon and FF5F also deliver well-specified t-statistics in all our conditions in Table 3.4, except for the condition of the

Table 3.4 Specification (Size) of Test Statistics in Random Samples

The numbers presented in this table represent the percentage of 1,000 samples with pseudo-event firms selected randomly, which reject the null hypothesis of no one-year (Panel A), three-year (Panel B), and five-year (Panel C) abnormal returns (ARs) at the theoretical significant levels of 1 percent, 5 percent, or 10 percent in favor of the alternative hypothesis of a significantly negative ARs (i.e., the calculated p-value is less than 0.5 percent at the 1 percent significance level) or a significantly positive AR (calculated p-value is greater than 99.5 percent at the 1 percent significance level). We conduct a two-stage method. In the first stage, we calculate the monthly abnormal return for all stock months. DL_LSTM and DL_ANN refer to DL models with LSTM and ANN, respectively. Bessembinder refers to the method proposed by Bessembinder and Cooper, and Zhang (2019). Lyon refers to the method of buy-and-hold size/book-to-market portfolio used by Lyon, Barber, and Tsai (1999). FF5F refers to the method from Fama-French (2015). In the second stage, we conduct the following regression: $AR_{it} = a + b^* D_{it} + u_{it}$, Where D_{it} is an indicator that equals one if firm i experienced a corporate event, and zero otherwise. AR_{it} is the log abnormal return for stock i, at time t. We randomly choose ten pseudo-event firms in each month from January 1980 to December 2021. The indicator variable (D_{it}) equals one if the firm was selected as a pseudo-event firm during any of the prior 12, 36, or 60 months, respectively for one-, three-, and five-year horizons, and zero otherwise. AR_{it} is the monthly abnormal return. The Number marked * suggest that the empirical size is significantly different from the theoretical significance level at 1 the 1 percent level, one-sided binomial test statistic.

	Two-Tailed Theoretical Significance Level					
	1%		5%		10%	
	Theoretical Cumulative Density Function (%)					
	0.5	99.5	2.5	97.5	5.0	95.0
	Panel A: One-Year ARs					
DL_LSTM	0.5	0.7	2.3	2.4	4.8	5
DL_ANN	0.7	0.6	2.8	2.5	4.4	4.6
Bessembinder	1.2	0.8	2.1	2.2	5.5	4.8
Lyon	0.3	0.6	2.6	2.5	5.4	4.3
FF5F	1.1	0.8	2.1	1.7	6.5	5.9
	Panel B: Three-Year ARs					
DL_LSTM	0.4	0.6	2.6	2.3	4.4	5.1
DL_ANN	0.6	0.7	2.8	2.9	5.2	5.1
Bessembinder	0.9	0.7	2.0	2.1	4.8	5.3
Lyon	0.7	0.8	3.2	2.9	5.9	5.6
FF5F	1.0	0.8	1.5	2.1	6.3	5.9
	Panel C: Five-Year ARs					
DL_LSTM	0.3	0.8	2.0	2.6	5.2	5.3
DL_ANN	0.7	0.6	2.8	2.7	5.4	5.0
Bessembinder	0.9	1.1	2.3	2.1	5.5	5.5
Lyon	0.5	1.3*	2.2	3.0	4.9	5.7
FF5F	1.4*	1.1	1.9	2.1	6.1	6.7

five-year horizon at the 1% significance level, in which Lyon yields a positive bias and FF5F generates a negative one.

3.5.2.2 Specification in Nonrandom Samples

Our methods of DL_LSTM and DL_ANN have proven to yield well-specified t-statistics and substantially augment the power of tests concerning long-run abnormal returns in random samples. Nonetheless, our curiosity extends to the adaptability of these innovative approaches within nonrandom samples. Although we acknowledge that achieving optimal performance across all sampling scenarios might not be feasible for our methods, it remains crucial to deeply grasp the underlying sources and extent of potential challenges, particularly in the presence of sampling biases. This comprehension stands as a pivotal foundation for constructing a sound test statistic tailored to specific sample situations.

Firm Size

To evaluate the influence of size-based sampling biases on all methods, we perform an analysis involving 1000 randomly drawn samples from both the largest size decile (decile 10, as shown in Table 3.5, Panel A) and the smallest size decile (refer to Table 3.5, Panel B). Within each sample, we replicate the procedural steps outlined in section 3.5.2.1.

The specifications of the five methods applied to these samples are detailed in Table 3.5. The findings from this assessment reveal that all methods, with the exception of FF5F, demonstrate accurate specification. Notably, the t-statistics derived from the FF5F approach exhibit negative skewness when applied to the smallest size sample, particularly over the 1-year horizon.

Book-to-Market Ratio

We select ten pseudo-event firms in each monthly interval spanning from January 1980

Table 3.5 Specification (Size) of Test Statistics in Size-Based Samples

The numbers presented in this table represent the percentage of 1,000 random samples with pseudo-event firms selected randomly in the largest firm size group (Panel A) and smallest firm size group (Panel B) that reject the null hypothesis of no one-, three-year, and five-year abnormal return (AR) at the theoretical significant levels of 5 percent in favor of the alternative hypothesis of a significantly negative AR (i.e., the calculated p-value is less than 2.5 percent at the 5 percent significance level) or a significantly positive AR (calculated p-value is greater than 97.5 percent at the 5 percent significance level). We conduct a two-stage method for two size-based groups. In the first stage, we calculate the monthly abnormal return for all stock months. DL_LSTM and DL_ANN refer to DL models with LSTM and ANN, respectively. Bessembinder refers to the method proposed by Bessembinder and Cooper, and Zhang (2019). Lyon refers to the method of buy-and-hold size/book-to-market portfolio used by Lyon, Barber, and Tsai (1999). FF5F refers to the method from Fama-French (2015). In the second stage, we conduct the following regression: $AR_{it} = a + b^* D_{it} + u_{it}$, Where D_{it} is an indicator that equals one if firm i experienced a corporate event, and zero otherwise. AR_{it} is the log abnormal return for stock i, at time t. We randomly choose ten pseudo-event firms in each month from January 1980 to December 2021. The indicator variable (D_{it}) equals one if the firm was selected as a pseudo-event firm during any of the prior 12, 36, or 60 months, respectively for one-, three-, and five-year horizons, and zero otherwise. AR_{it} is the monthly abnormal return. The Number marked * suggest that the empirical size is significantly different from the theoretical significance level at 1 the 1 percent level, one-sided binomial test statistic.

	Horizon					
	1 Year		3 Year		5 Year	
	Theoretical Cumulative Density Function (%)					
Model	2.5	97.5	2.5	97.5	2.5	97.5
Panel A: Samples of Largest Firms						
DL_LSTM	2.5	2.6	2.5	2.7	2.4	2.8
DL_ANN	2.3	2.7	3.0	2.2	2.4	2.7
Bessembinder	2.5	2.5	2.6	3.0	2.3	2.8
Lyon	3.4	3.5	3.3	2.8	2.5	2.1
FF5F	2.1	1.9	1.5	3.1	1.7	2.9
Panel B: Samples of Smallest Firms						
DL_LSTM	2.4	2.5	2.3	2.9	2.3	3.1
DL_ANN	2.6	2.8	2.2	3.1	2.1	3.4
Bessembinder	2.3	2.9	1.8	3.2	1.7	3.2
Lyon	2.7	2.8	2.6	2.2	4.2	2.8
FF5F	4.1*	1.5	3.1	1.2	2.8	2.1

to December 2021. This selection process is conducted independently for the low book-to-market decile (decile 10, as illustrated in Table 3.6, Panel A) and the high book-to-market decile (decile 1, as detailed in Table 3.6, Panel B). It is noteworthy that firms with zero book-to-market values are excluded from our dataset. The subsequent step involves the execution of the regression specified in equation (3.5.1). In this context, the indicator variable (D_{it}) takes on a value of one if a firm qualifies as a pseudo-event firm within the

Table 3.6 Specification (Size) of Test Statistics in Book-to-Market Samples

The numbers presented in this table represent the percentage of 1,000 random samples with pseudo-event firms selected randomly in the lowest book-to-market decile (Panel A) and largest book-to-market decile (Panel B) that reject the null hypothesis of no one-, three-year, and five-year abnormal return (AR) at the theoretical significant levels of 5 percent in favor of the alternative hypothesis of a significantly negative AR (i.e., the calculated p-value is less than 2.5 percent at the 5 percent significance level) or a significantly positive AR (calculated p-value is greater than 97.5 percent at the 5 percent significance level). We conduct a two-stage method for two Book-to-Market groups. In the first stage, we calculate the monthly abnormal return for all stock months. DL_LSTM and DL_ANN refer to DL models with LSTM and ANN, respectively. Bessembinder refers to the method proposed by Bessembinder and Cooper, and Zhang (2019). Lyon refers to the method of buy-and-hold size/book-to-market portfolio used by Lyon, Barber, and Tsai (1999). FF5F refers to the method from Fama-French (2015). In the second stage, we conduct the following regression: $AR_{it} = a + b^* D_{it} + u_{it}$, Where D_{it} is an indicator that equals one if firm i experienced a corporate event, and zero otherwise. AR_{it} is the log abnormal return for stock i, at time t. We randomly choose ten pseudo-event firms in each month from January 1980 to December 2021. The indicator variable (D_{it}) equals one if the firm was selected as a pseudo-event firm during any of the prior 12, 36, or 60 months, respectively for one-, three-, and five-year horizons, and zero otherwise. AR_{it} is the monthly abnormal return. The Number marked * suggests that the empirical size is significantly different from the theoretical significance level at 1 the 1 percent level, one-sided binomial test statistic.

	Horizon					
	1 Year		3 Year		5 Year	
	Theoretical Cumulative Density Function (%)					
Model	2.5	97.5	2.5	97.5	2.5	97.5
Panel A: Samples of Firms with Low Book-to-Market Ratios						
DL_LSTM	3.9*	2.1	3.1	3.2	3.5	2.7
DL_ANN	4.1*	2.4	2.1	2.9	3.0	2.4
Bessembinder	4.2*	2.6	2.9	3.0	4.2*	2.5
Lyon	3.9*	2.9	4.1*	2.9	3.4	2.4
FF5F	6.2*	2.3	5.4*	0.8	6.1*	1.4
Panel B: Samples of Firms with High Book-to-Market Ratios						
DL_LSTM	3.3	2.5	2.4	2.6	2.4	3.1
DL_ANN	2.3	2.7	3.5	3.5	2.7	3.1
Bessembinder	3.6	3.0	2.4	3.6	2.6	4.1*
Lyon	2.6	2.2	3.6	2.3	2.9	2.1
FF5F	1.9	4.1*	1.8	4.5*	2.3	4.1*

preceding 12, 36, or 60 months – corresponding to one-, three-, and five-year, or zero otherwise. The monthly abnormal return is represented as AR_{it}. This process is repeated 1,000 times.

The specifications of the five methods applied to these selected samples are presented in Table 3.6. When considering the sample characterized by a high book-to-market ratio, the DL_LSTM, DL_ANN, and Lyon methods produce reasonably well-defined test statistics. In

contrast, both Bessembinder and FF5F exhibit a positively biased result. Conversely, in the sample featuring a low book-to-market ratio, all methods demonstrate inferior performance compared to the high BTM Ratio sample. For instance, both DL_LSTM and DL_ANN display a singular negative bias, whereas the remaining methods exhibit more than two biases. Importantly, the findings from these non-random samples align closely with those reported in previous literature (e.g., Lyon, Barber, and Tsai (1999)).

Pre-Event Return Performance

A typical characteristic observed in firms analyzed in event studies is a period of abnormal stock returns preceding the focal event. For instance, equity issuance is typically preceded by a phase of high stock returns, while share repurchases are commonly preceded by a period of low stock returns. Furthermore, Jegadeesh and Titman (1993) have documented the presence of persistency in stock returns, which cannot be adequately explained by factors such as firm size and book-to-market ratio, as suggested by Fama and French (1996). Therefore, it is crucial to consider the prior return performance of the sample firms as a significant factor in long-term event studies.

We begin by calculating the preceding six-month buy-and-hold return for all firms every month, covering the period from July 1980 to December 2021. Subsequently, we rank all firms based on their six-month return and group them into deciles each month. To create pseudo-event firms, we randomly select ten firms separately from the lowest decile (decile 1, as shown in Table 3.7, Panel A) and the highest decile (decile 10, as shown in Table 3.7, Panel B) of the Six-Month Pre-Event return for each month between January 1980 and

December 2021. Next, we perform the regression analysis outlined in equation (3.5.1). Here, we adopt the method called month-year(s) that is mentioned in section 3.5.1.1. we randomly choose 36 pseudo-event firms in each month from January 1980 to December 2016. The indicator variable (D_{it}) equals one if the firm was selected as a pseudo-event firm, and zero otherwise. AR_{it} is the annualized one-, three-, or five-year abnormal return. This selection process is repeated 1,000 times.

Table 3.7 Specification (Size) of Test Statistics in Samples Based on Pre-Event Return Performance

The numbers presented in this table represent the percentage of 1,000 random samples with pseudo-event firms selected randomly from the lowest Six-Month Pre-Eevent return decile (Panel A) and the highest Six-Month Pre-Eevent return decile (Panel B) that reject the null hypothesis of no one-, three-year, and five-year abnormal return (AR) at the theoretical significant levels of 5 percent in favor of the alternative hypothesis of a significantly negative AR (i.e., the calculated p-value is less than 2.5 percent at the 5 percent significance level) or a significantly positive AR (calculated p-value is greater than 97.5 percent at the 5 percent significance level). We conduct a two-stage method for two groups. In the first stage, we calculate the monthly abnormal return for all stock months. DL_LSTM and DL_ANN refer to DL models with LSTM and ANN, respectively. Bessembinder refers to the method proposed by Bessembinder, Cooper, and Zhang (2019). Lyon refers to the method of buy-and-hold size/book-to-market portfolio used by Lyon, Barber, and Tsai (1999). FF5F refers to the method from Fama-French (2015). In the second stage, we conduct the following regression: $AR_{it} = a + b* D_{it} + u_{it}$, Where D_{it} is an indicator that equals one if firm i experienced a corporate event, and zero otherwise. AR_{it} is the log abnormal return for stock i, at time t. We randomly choose ten pseudo-event firms in each month from January 1980 to December 2021. The indicator variable (D_{it}) equals one if the firm was selected as a pseudo-event firm, and zero otherwise. AR_{it} is the annualized one-, three-, or five-year abnormal return. The Number marked * suggests that the empirical size is significantly different from the theoretical significance level at 1 the 1 percent level, one-sided binomial test statistic.

	Horizon					
	1 Year		3 Year		5 Year	
	Theoretical Cumulative Density Function (%)					
Model	2.5	97.5	2.5	97.5	2.5	97.5
Panel A: Samples of Firms with Low Six-Month Pre-Event Returns						
DL_LSTM	2.5	2.7	2.3	2.4	2.2	2.3
DL_ANN	2.9	3.0	2.8	3.0	2.2	2.9
Bessembinder	2.8	2.8	2.4	2.9	2.0	2.3
Lyon	4.9*	1.1	2.5	3.2	2.9	3.0
FF5F	3.9*	1.3	3.1	2.8	1.3	2.4
Panel B: Samples of Firms with High Six-Month Pre-Event Returns						
DL_LSTM	2.4	5.1*	2.2	2.3	2.2	2.8
DL_ANN	2.1	3.5	2.2	2.6	2.6	3.9*
Bessembinder	4.9*	2.4	3.5	2.5	3.3	2.4
Lyon	8.2*	1.2	6.2*	1.9	1.7	5.8*
FF5F	3.1	2.5	4.2	3.2	1.4	2.9

The results of this analysis are presented in Table 3.7. For firms with high six-month pre-event returns, FF5F shows the best performance without any bias at all three horizons, DL models and Bessembinder each only yield one biased result, and all four methods achieve an improvement, compared to Lyon. While in the group of firms with high six-month pre-event returns, DL models and Bessembider can yield well-specified results at all horizons, Lyon and FF5F have negative biases at the one-year horizon. These results indicate that four of five methods can reduce the effect of the pre-event return performance when we observe long-run abnormal returns.

Industry Clustering

Assuming the existence of varying expected returns across industries and an imprecise asset pricing model, the null hypothesis of zero long-run abnormal returns in favor of both positive and negative long-run abnormal returns would be frequently rejected. To evaluate the impact of industry clustering on the sample firms, we conduct 1,000 iterations where each iteration involves drawing a pseudo-event firm portfolio with ten firms each month, all sharing the same two-digit SIC code. This sampling process consists of two steps: initially, a random selection of a two-digit SIC code takes place; subsequently, a pseudo-event firm portfolio is drawn from this specific SIC code. In scenarios where the chosen two-digit SIC code does not encompass a minimum of 200 firms throughout our analysis timeframe, we bolster the sample by introducing arbitrarily selected firms from an alternative second two-digit SIC code. The findings of this analysis are presented in Table 3.8.

Table 3.8 Specification (Size) of Test Statistics in Samples with Industry Clustering

The numbers presented in this table represent the percentage of 1,000 random samples with pseudo-event firms selected randomly from stocks with the same two-digit SIC that reject the null hypothesis of no one-, three-year, and five-year abnormal return (AR) at the theoretical significant levels of 5 percent in favor of the alternative hypothesis of a significantly negative AR (i.e., the calculated p-value is less than 2.5 percent at the 5 percent significance level) or a significantly positive AR (calculated p-value is greater than 97.5 percent at the 5 percent significance level). We conduct a two-stage method. In the first stage, we calculate the monthly abnormal return for all stock months. DL_LSTM and DL_ANN refer to DL models with LSTM and ANN, respectively. Bessembinder refers to the method proposed by Bessembinder, Cooper, and Zhang (2019). Lyon refers to the method of buy-and-hold size/book-to-market portfolio used by Lyon, Barber, and Tsai (1999). FF5F refers to the method from Fama-French (2015). In the second stage, we conduct the following regression: $AR_{it} = a + b * D_{it} + u_{it}$, Where D_{itk} is an indicator that equals one if firm i experienced a corporate event, and zero otherwise. AR_{it} is the log abnormal return for stock i, at time t. We randomly choose ten pseudo-event firms in each month from January 1980 to December 2021. The indicator variable (D_{it}) equals one if the firm was selected as a pseudo-event firm during any of the prior 12, 36, or 60 months, respectively for one-, three-, and five-year horizons, and zero otherwise. AR_{it} is the monthly abnormal return. The Number marked * suggests that the empirical size is significantly different from the theoretical significance level at 1 the 1 percent level, one-sided binomial test statistic.

	Horizon					
	1 Year		3 Year		5 Year	
	Theoretical Cumulative Density Function (%)					
Model	2.5	97.5	2.5	97.5	2.5	97.5
DL_LSTM	3.2	2.9	3.4	2.8	4.8*	2.9
DL_ANN	2.5	3.2	2.7	3.0	5.3*	3.4
Bessembinder	3.1	3.0	2.6	2.8	4.9*	2.5
Lyon	4.3*	5.9*	7.1*	6.5*	8.5*	6.8*
FF5F	3.5	3.1	4.1*	3.5	4.4*	5.9*

The results in Table 3.8 indicate that DL models and Bessembinder can yield less misspecification in the non-random samples across industries, having an improvement compared to other benchmarks. The t-statistics for Lyon in Table 3.8 are similar to those reported by Lyon, Barber, and Tsai (1999), in which they explained that controlling for size and book-to-market alone is not sufficient when samples are drawn from a single two-digit SIC code.

Cross-Sectional Dependence of Sample Observations

As outlined by Brav (2000), when there exists cross-sectional dependence among the

gathered data within a sample, it can lead to inaccurately specified test statistics. This dependence causes test statistics to be inflated, as the number of sample firms overestimates the number of independent observations. To gauge the extent of cross-sectional dependence, we examine two extreme sample scenarios: (1) calendar clustering, and (2) overlapping return calculations.

Calendar Clustering

We operate under the reasonable presumption that the returns of firms occurring at the same time are more prone to displaying cross-sectional relationships, compared to returns from different periods. If this assumption remains valid, the issue of cross-sectional dependence will be most pronounced when all sample firms share the same event date. To assess the influence of calendar clustering on event dates, we undertake 1,000 iterations, each involving the creation of samples comprising 1,000 firms. It is ensured that all 1,000 firms within each specific sample share the same event date. Precisely, we proceed by randomly selecting a year-month and subsequently selecting a pseudo-event portfolio comprising 1,000 firms for that designated month. This is followed by the two-stage process narrated in the previous context. AR_{it} is the annualized one-, three-, or five-year abnormal return; thus, we adopt the method called month-year(s) that is mentioned in section 3.5.1.1.

Table 3.9 presents the percentage of 1,000 random samples with pseudo-event firms selected randomly in a common event month that reject the null hypothesis of no abnormal return over three horizons. Results show that DL_LSTM and Lyon can generate well-specified t-statistics without any bias at all three horizons. A singular negative bias is

present in both DL_ANN and Bessembinder.

Table 3.9 Specification (Size) of Test Statistics in Samples with a Common Event Month (Calendar Clustering)

The numbers presented in this table represent the percentage of 1,000 random samples with pseudo-event firms selected randomly in a common event month that reject the null hypothesis of no one-, three-year, and five-year abnormal return (AR) at the theoretical significant levels of 5 percent in favor of the alternative hypothesis of a significantly negative AR (i.e., the calculated p-value is less than 2.5 percent at the 5 percent significance level) or a significantly positive AR (calculated p-value is greater than 97.5 percent at the 5 percent significance level). We conduct a two-stage method. In the first stage, we calculate the monthly abnormal return for all stock months. DL_LSTM and DL_ANN refer to DL models with LSTM and ANN, respectively. Bessembinder refers to the method proposed by Bessembinder, Cooper, and Zhang (2019). Lyon refers to the method of buy-and-hold size/book-to-market portfolio used by Lyon, Barber, and Tsai (1999). FF5F refers to the method from Fama-French (2015). In the second stage, we conduct the following regression: $AR_{it} = a + b^* D_{it} + u_{it}$, Where D_{it} is an indicator that equals one if firm i experienced a corporate event, and zero otherwise. AR_{it} is the log abnormal return for stock i, at time t. We randomly choose one thousand pseudo-event firms in a given month randomly selected from the time period of January 1980 to December 2021. The indicator variable (D_{it}) equals one if the firm was selected as a pseudo-event firm, and zero otherwise. AR_{it} is the annualized one-, three-, or five-year abnormal return. The Number marked * suggests that the empirical size is significantly different from the theoretical significance level at 1 the 1 percent level, one-sided binomial test statistic.

	Horizon					
	1 Year		3 Year		5 Year	
	Theoretical Cumulative Density Function (%)					
Model	2.5	97.5	2.5	97.5	2.5	97.5
DL_LSTM	3.4	2.8	3.1	2.5	3.2	2.4
DL_ANN	3.8*	3.3	2.1	2.8	3.5	2.9
Bessembinder	2.1	2.9	3.5	3.1	3.9*	3.5
Lyon	3.0	3.5	2.1	2.5	3.5	2.8
FF5F	3.9*	3.4	4.3*	2.5	3.7*	5.9*

Overlapping Return Calculations

When conducting event studies aimed at analyzing long-term abnormal returns, a significant challenge emerges from the overlapping periods of return calculation for the same firm. Take, for example, the scenario of Microsoft's common stock split taking place in April 1990, June 1991, and June 1992. It becomes apparent that returns computed over three or five-year periods, with respect to each of these event months, lack independence due to the presence of overlapping returns. This showcases a pronounced form of cross-sectional

dependence, presenting researchers with a substantial obstacle when investigating long-term abnormal stock returns within an event study context.

Table 3.10 Specification (Size) of Test Statistics in Samples with Overlapping Returns

The numbers presented in this table represent the percentage of 1,000 random samples with pseudo-event firms selected randomly, which reject the null hypothesis of no one-, three-year, and five-year abnormal return (AR) at the theoretical significant levels of 5 percent in favor of the alternative hypothesis of a significantly negative AR (i.e., the calculated p-value is less than 2.5 percent at the 5 percent significance level) or a significantly positive AR (calculated p-value is greater than 97.5 percent at the 5 percent significance level). We conduct a two-stage method for two size-based groups. In the first stage, we calculate the annualized log abnormal return over one, three-, and five-year periods for all stock months. DL_LSTM and DL_ANN refer to DL models with LSTM and ANN, respectively. Bessembinder refers to the method proposed by Bessembinder, Cooper, and Zhang (2019). Lyon refers to the method of buy-and-hold size/book-to-market portfolio used by Lyon, Barber, and Tsai (1999). FF5F refers to the method from Fama-French (2015). In the second stage, we conduct the following regression: $AR_{it} = a + b^* D_{it} + u_{it}$, Where D_{it} is an indicator that equals one if firm i experienced a corporate event, and zero otherwise. AR_{it} is the log abnormal return for stock i, at time t. Selecting these samples consists of two steps. First, we randomly sample 500 firms from our population. Second, for these same 500 firms, we randomly select a second event month that falls within H-1 periods of the original event month (either before or after), where H equals 12, 36, or 60. Hence we have 1000 firms with 1000 event months where the same firm appears in the sample twice and this generates the issue of overlapping returns. We repeat this procedure 1000 times. The indicator variable (D_{it}) equals one if the firm was selected as a pseudo-event firm during any of the prior 12, 36, or 60 months, respectively for one-, three-, and five-year horizons, and zero otherwise. AR_{it} is the monthly abnormal return. The Number marked * suggests that the empirical size is significantly different from the theoretical significance level at 1 the 1 percent level, one-sided binomial test statistic.

	Horizon					
	1 Year		3 Year		5 Year	
	Theoretical Cumulative Density Function (%)					
Model	2.5	97.5	2.5	97.5	2.5	97.5
DL_LSTM	3.0	2.2	2.2	2.3	1.6	2.8
DL_ANN	2.6	1.9	2.5	2.1	2.3	1.6
Bessembinder	2.2	2.1	3.0	1.3	1.8	2.2
Lyon	4.7*	5.5*	3.1	4.5*	3.9*	4.8*
FF5F	3.1	2	1.4	1.9	2.4	1.7

To evaluate how the calculation of overlapping returns impacts examinations of long-term abnormal stock returns, our approach comprises two key steps. Initially, we draw a random sample of 500 firms from our population. Subsequently, for these identical 500 firms, we randomly choose a second selection of an event, positioned within H-1 periods of the original event month (either before or after), where H represents the span over which returns are compounded (12, 36, or 60). Hence, we have 1000 firms with 1000 event months where the same firm appears in the sample twice, and this generates the issue of overlapping returns. We repeat this procedure 1000 times, and the results are presented in Table 3.10.

Subsequently, we conduct the regression analysis as detailed in equation (3.5.1). The

indicator variable (D_{it}) equals one if the firm was selected as a pseudo-event firm during any of the prior 12, 36, or 60 months, respectively for one-, three-, and five-year horizons, and zero otherwise. AR_{it} represents the monthly abnormal return.

Table 3.10 presents the percentage of 1000 random samples, rejecting the null over three horizons. Results show four of five methods, except for Lyon, can produce well-specified t statistics across all spans.

3.5.3 Robust Experiments

In the previous sections of 3.5.1 and 3.5.2, we evaluate DL models that can outcome the abnormal return directly when their inputs are the C14 and realized return. In this section, we follow the logic in Bessembinder to train our DL_LSTM_B models by providing the input of just C14. Consequently, the trained DL_LSTM_B could forecast the predicted return when we feed it with C14. The calculation process of the abnormal return is the same as the method in Bessembinder or other benchmarks by deducting the predicted return from the realized return. We conduct these robust experiments, including the power test and specifications, both in the random and non-random samples, and the results are presented in Tables 3.11 and 3.12.

Table 3.11 presents the proportions derived from a set of 1,000 event samples from pseudo-event corporations, indicating the instances in which the null hypothesis of no annualized abnormal return at various yearly induced levels of abnormal return. This is examined across various annual levels of induced abnormal returns (depicted on the

Table 3.11 The Power of Test Statistics in Random Samples for the Robust Check

This table presents the percentage of 1,000 random samples of pseudo-event firms rejecting the null hypothesis of no annualized abnormal return at various yearly induced levels of abnormal return (horizontal axis) for DL_LSTM_B and Bessembinder, over the three-year horizon. We conduct a two-stage method. In the first stage, we calculate the monthly abnormal return for all stock months. DL_LSTM_B is the model that can estimate the stock predicted return when C14 is fed into the model. Bessembinder refers to the method proposed by Bessembinder, Cooper, and Zhang (2019). In the second stage, we conduct the following regression: $AR_{it} = a + b* D_{it} + u_{it}$, Where D_{it} is an indicator that equals one if firm i experienced a corporate event, and zero otherwise. AR_{it} is the monthly log abnormal return for stock i, at time t. We randomly choose ten pseudo-event firms in each month from January 1980 to December 2021. The indicator variable (D_{it}) equals one if the firm was selected as a pseudo-event firm during any of the prior 36 months, and zero otherwise.

induced_AR %	DL_LSTM_B	Bessembinder	Difference/(t-stat)
-3.0	100.00	100.00	0.00
-2.8	100.00	89.67	10.33***/(10.73)
-2.6	90.56	82.08	8.48***/(5.56)
-2.4	80.24	75.94	4.31**/(2.33)
-2.2	70.31	65.35	4.96**/(2.38)
-2.0	61.02	51.52	9.50***/(4.30)
-1.8	51.47	39.59	11.88***/(5.37)
-1.6	37.69	27.40	10.29***/(4.94)
-1.4	28.29	20.84	7.45***/(3.88)
-1.2	18.45	16.40	2.05/(1.21)
-1.0	14.21	13.07	1.14/(0.74)
-0.8	12.22	11.21	1.01/(0.70)
-0.6	10.94	8.70	2.23*/(1.67)
-0.4	7.48	6.60	0.88/(0.77)
-0.2	6.74	4.97	1.77*/(1.67)
0.0	5.38	3.54	1.83**/(2.00)
0.2	6.49	5.67	0.81/(0.77)
0.4	6.39	6.03	0.36/(0.33)
0.6	9.41	9.31	0.10/(0.08)
0.8	12.36	11.17	1.19/0.83)
1.0	15.54	14.51	1.03/0.64)
1.2	21.12	17.18	3.94**/(2.24)
1.4	31.69	21.74	9.95***/(5.06)
1.6	43.41	28.42	14.99***/(7.07)
1.8	54.44	39.60	14.84***/(6.72)
2.0	65.69	49.38	16.31***/7.48)
2.2	76.45	61.41	15.05***/(7.36)
2.4	84.10	72.54	11.56***/(6.34)
2.6	94.35	87.22	7.13***/(5.55)
2.8	100.00	97.08	2.92***/5.48
3.0	100.00	100.00	0.00

horizontal axis), considering both DL_LSTM_B and Bessembinder methodologies, observed over a three-year timeframe. The outcomes consistently align with preceding findings, illustrating that the performance of DL_LSTM_B surpasses that of the Bessembinder benchmark. Notably, when the induced AAR values fall within the absolute interval of 1.4 to 2.8, the rejection ratios based on the DL_LSTM_B approach exhibit significantly greater magnitudes compared to those derived from the Bessembinder.

Table 3.12 Specification (Size) of Test Statistics in Random and Non-Random Samples for the model of DL_LSTM_B

The numbers presented in this table represent the percentage of 1,000 random samples with pseudo-event firms selected randomly, which reject the null hypothesis of no one-, three-year, and five-year abnormal return (AR) at the theoretical significant levels of 5 percent in favor of the alternative hypothesis of a significantly negative AR (i.e., the calculated p-value is less than 2.5 percent at the 5 percent significance level) or a significantly positive AR (calculated p-value is greater than 97.5 percent at the 5 percent significance level). DL_LSTM_B is the model that can outcome the stock expected return when C14 is fed into the model. We conduct a two-stage method for random and non-random samples. The non-random smples are the same as those in previous tables, which include sized-based, book-to-market-based, pre-event-return-based, industry clustering, calendar clustering, and overlapping returns samples. In the first stage, we calculate the monthly log abnormal return for all stock months. In the second stage, we conduct the following regression: $AR_{it} = a + b^* D_{it} + u_{it}$, Where D_{it} is an indicator that equals one if firm i experienced a corporate event, and zero otherwise. AR_{it} is the monthly log abnormal return for stock i, at time t. The selecting methods are the same as those in previous tables.The indicator variable (D_{it}) equals one if the firm was selected as a pseudo-event firm during any of the prior 12, 36, or 60 months, respectively for one-, three-, and five-year horizons, and zero otherwise. The Number marked * suggests that the empirical size is significantly different from the theoretical significance level at 1 the 1 percent level, one-sided binomial test statistic.

	Horizon					
	1 Year		3 Year		5 Year	
	Theoretical Cumulative Density Function (%)					
Sample Characteristics	2.5	97.5	2.5	97.5	2.5	97.5
Random samples	2.1	2.2	2.6	2.8	2.4	2.9
Small firms	2.5	2.9	2.4	3.1	2.4	2.7
Large firms	2.6	2.1	2.9	2.2	2.2	2.6
Low book-to-market ratio	4.1*	1.8	3.5	3.5	2.2	3.0
High book-to-market ratio	2.6	1.8	3.1	2.4	1.9	2.7
Poor pre-event returns	2.5	2.6	2.1	3.2	1.8	2.5
Good pre-event returns	4.3*	1.8	2.9	3.8*	3.4	3.1
Industry clustering	2.1	2.2	3.2	2.3	4.4*	3.1
Calendar clustering	2.9	3.4	4.6*	2.1	2.5	3.2
Overlapping returns	2.4	2.6	2.9	2.1	2.8	3.4

Table 3.12 exhibits the result of the specification of test statistics in random and non-random samples for DL_LSTM_B. The data contained in Table 3.12 mirrors analogous patterns as shown in Section 3.5.2. Initially, DL_LSTM_B demonstrates the well specification of t-statistics in random samples. Furthermore, it effectively conducts correct t-tests within some non-random samples. To illustrate, t-tests performed on two non-random samples, classified by the firm size and overlapping return, exhibit precise specifications. Moreover, DL_LSTM_B displays enhancement in its performance across other non-random samples compared to the data presented in section 3.5.2, as gauged against three distinct benchmarks.

The findings presented in the preceding tables in sections 3.5.1, 3.5.2, and 3.5.3 reaffirm the notion that DL techniques have the potential to delve deeper into the intricate relationships between predicted returns and C14, surpassing the capabilities of conventional linear regression. This enhanced capability to uncover intricate associations not only showcases itself in the improved power of tests for long-run abnormal returns but also acts as a catalyst for further exploration and expansion of DL applications within the realm of finance. The observed outcomes thus provide a motive force to harness the capabilities of DL in addressing intricate financial phenomena and contributing to the advancement of this field.

3.5.4 Prediction Error Discussion

Our approach involves a comparative analysis of two methodologies, DL_LSTM and Bessembinder, grounded in the context of the prediction errors of abnormal returns.

Subsequently, we investigate the influence of various characteristics on the prediction error within both methods. To facilitate this examination, we undertake the subsequent simulation procedure.

Firstly, a systematic selection process involves choosing a specific stock at random from the entire set of available securities. Subsequently, from the period spanning January 1980 to December 2020, a month is selected randomly. Secondly, we execute the regression: $Pre(AR_{it}) = a + b^* D_{it} + u_{it}$. Here, $Pre(AR_{it})$ is denoted as the predicted abnormal return for stock i in month t. For Bessembinder, $Pre(AR_{it})$ equals that the realized return on stock i in month t deducts the predicted return obtained from the Bessembinder method, while for the DL_LSTM model, $Pre(AR_{it})$ is just the outcome from the model for stock i in month t. The indicator variable D_{it} equals one if the stock was selected during any of the prior 36 months and zero otherwise. Thirdly, we proceed to compute the prediction error, denoted as "Error," by subtracting the estimated coefficient ($\hat{b}$) from the actual abnormal return. In this context, the actual abnormal return is set to zero, grounded in the assumption that among the randomly chosen stocks from the complete stock market, no abnormal return exists. Lastly, we replicated a substantial number of times – precisely 100,000 iterations (N). Then, we finish the following calculation. $RMSE = \sqrt{\Sigma(Error^2)/N}$ and $MAE = \Sigma|Error|/N$.

Presented in Table 3.13 are the outcomes derived from the simulation described earlier, focusing on RMSE and MAE. Notably, the RMSE values generated by the DL_LSTM and the Bessembinder method are 0.0137 and 0.0214, respectively. This indicates a decline of approximately 27% in RMSE when comparing the DL_LSTM model to the Bessembinder method. Similarly, a corresponding trend is observed in the measure of MAE. These findings

underscore the notable distinctions between the two methodologies and their consequential impact on the observed ratios.

Table 3.13 Prediction Error

RMSE and MAE are presented in Table 3.13. RMSE = $\sqrt{\Sigma(\text{Error}^2)/N}$ and MAE = $\Sigma|\text{Error}|/N$, where Error equals the $\hat{b}$ deducts the actual abnormal return (AR) which is from the regression of $\text{Pre}(AR_{it}) = a + b^* D_{it} + u_{it}$. In the above regression, for the bessembinder, $\text{Pre}(AR_{it})$ equals that the realized return on stock i in month t deducts the predicted return from the bessembinder method, while $\text{Pre}(AR_{it})$ is just the outcome from the DL(LSTM) model for stock-month it. The indicator variable D_{it} equals one if the stock was selected during the month t+1 to t+36, and zero otherwise.

AR	RMSE		MAE	
	Bessembinder	DL	Bessembinder	DL
0	0.0214	0.0137	0.0158	0.00931

Next, we use the list of errors ($\hat{b}_n - AR$) calculated in the previous process mentioned above, and then match the randomly selected stock-month (it) with its corresponding characteristics of firms and markets. We attempt to explore the relationship between the absolute values of errors and characteristics for the methods of Bessembinder and DL, using the regression shown in the equation of 3.5.4.1.

$$|\text{Error}|_{it} = a_{0,i/(t-1)} + a_{n,\,i(t-1)} * \text{Characteristics}_{n,\,i(t-1)} + e_{it} \qquad (3.5.4.1),$$

where $|\text{Error}|_{it}$ is the absolute error of the estimated abnormal return for stock/month it, and $\text{Characteristics}_{n,\,i(t-1)}$ include the firm characteristics, such as LogSize, Illiquidity, Leverage, Idiosyncratic risk, beta, and Turnover for firm/month i(t-1). These firm characteristics are explained in Appendix A.2.2. $\text{Characteristics}_{n,\,i(t-1)}$ also cover market sentiment variables and period dummies. There are two sentiment variables related to the Baker and Wurgler (2006) sentiment index (denoted as SentimentWB) and the hot market dummy variable (denoted as

HotMonth). If SentimentWB is above its 1980 to 2020 median, HotMonth is one, and zero otherwise. We divide the whole period into four sub-periods, such as the 1980s, 1990s, 2000s, and 2010s, and take the 2010s as the reference sub-period, so we just include the other three sub-periods as dummies. Here, AR is zero because of the randomly selected stocks from the whole market.

The outcomes of the fixed effect regression are presented in Table 3.14. Specifically, Columns 2 and 3 provide the estimated coefficients and their corresponding t-values associated with the independent variables for both the Bessembinder and DL methodologies. The respective dependent variables for these methods are represented by the absolute errors, denoted as $|Error|_{Be}$ and $|Error|_{DL}$, respectively. In general, the estimated coefficients exhibit consistent signs across the two methodologies. For the Bessembinder method, 9 out of the 11 estimated coefficients demonstrate significance in varying degrees of significance levels of 1%， 5%， and 10%, whereas only 4 out of 11 estimated coefficients achieve significance in the DL method.

Of particular interest are three distinct changes among these results. Firstly, within the Bessembinder method, the variable LogSize significantly impacts the absolute error at a 1% significance level (t-statistic: -6.76). Remarkably, the DL method reveals an improvement whereby the LogSize variable ceases to exert a significant effect on the absolute error (t-statistic: -1.589). This sheds light on the reason why DL displays better-specified test statistics within the nonrandom group of firm size, compared to Bessembinder. Secondly, estimated coefficients for the two dummies ("D1990s" and "D2000s") cease to attain significance in the DL method, indicating that the absolute error from DL remains more

stable across various periods. Lastly, the DL method's error demonstrates reduced

Table 3.14 The Regression of the Error of Prediction on Characteristics

Table 3.14 reports the coefficients estimated and their corresponding t-statistics (in parentheses) from the regression: $|Error|_{it} = a_{0,it} + a_{n,i(t-1)} * Characteristics_{n,\ i(t-1)} + e_{it}$. We conduct the fixed effect model with the cluster of times. We estimate the regressions using the data lists of the errors from the simulation mentioned in 5.4. The dependent variables in columns 1 and 2 are $|Error|_{Be}$ and $|Error|_{DL}$ which represent the absolute errors for Bessembinder and DL methods, respectively. Column 3 of Table 3.14 reports the results of the regression with the dependent variable of $|Error|_{Be}-|Error|_{DL}$, which is the difference between the absolute error for the method of Bessembinder and the absolute error for the DL method. Independent variables, such as LogSize, Illiquidity, Leverage, Idiosyncratic risk, beta, and Turnover are explained in Appendix Table 1. SentimentWB is the Baker and Wurgler (2006) sentiment index, while HotMonth is a dummy variable that equals one if SentimentWB is above its 1980-2020 median, and zero otherwise. D1980s, D1990S, and D2000s refer to time period dummies, and 2010s is the reference time period. ***, **, and * correspond to statistical significance at the 1%, 5%, and 10% levels, respectively. The associated t-statistics are reported in parentheses.

	Dep. Var		
	$\|Error\|_{Be}$	$\|Error\|_{DL}$	$\|Error\|_{Be}-\|Error\|_{DL}$
Indep. Var			
LogSize	-0.0008***	-0.0001	-0.0007***
	(-6.76)	(-1.58)	(-12.91)
Illiquidity	-0.0067**	-0.0014	-0.0053***
	(-2.18)	(-0.89)	(-3.55)
Leverage	0.0004**	0.0002***	0.0001*
	(2.41)	(2.78)	(1.96)
Idiosyncratic risk	0.0012***	0.0004***	0.0008***
	(12.88)	(8.03)	(17.71)
beta	0.0071***	0.0041***	0.003**
	(2.70)	(2.86)	(2.36)
Turnover	0.007***	0.0014	0.0056***
	(4.24)	(1.54)	(7.06)
SentimentWB	6.91E-05	5.28E-05	1.63E-05
	(0.37)	(0.44)	(0.22)
HotMonth	0.0014***	0.0007*	0.0008***
	(2.80)	(1.73)	(3.25)
D1980s	0.0005	0.0005	-6.54E-05
	(0.75)	(1.42)	(-0.23)
D1990s	0.0023***	0.0002	0.0021***
	(4.53)	(0.72)	(9.25)
D2000s	0.0021***	0.0002	0.0019***
	(3.76)	(0.56)	(7.73)
N		100,000	
F-statistic	62.67	19.33	130.5
R2	0.119	0.040	0.240

susceptibility to the influence of market sentiment indices, as reflected in the estimated coefficients of the HotMonth dummy variable. For the Bessembinder method, the HotMonth variable's estimated coefficient yields a t-statistics of 2.801, whereas its counterpart in the DL method registers a mere 1.727. This signifies that the HotMonth variable exercises a diminished influence on the absolute error within the DL method.

Moving on, Column 4 of Table 3.14 presents the outcomes of a fixed effect regression. Here, the dependent variable is |Error|Be-|Error|DL, signifying the disparity between the absolute error generated by the Bessembinder method and that produced by the DL method. Among the 11 estimated coefficients, 9 exhibit statistical significance at levels of 1%, 5%, or 10%. Notably, the coefficients corresponding to the LogSize, HotMonth, "D1990s," and

"D2000s" variables – discussed earlier in the previous paragraph – are significant at the 1% level. This underscores the statistically significant reduction in absolute error attributed to these four characteristics.

3.6 Conclusion

In the research, we employ sophisticated techniques, deep learning (DL), to enhance the capability of detecting the abnormal returns (AR) of stocks after the disclosure of significant events, with the replacement of conventional linear regression. We introduce an innovative research approach aimed at directly estimating the abnormal return (AR) based on

the trained DL models. Because DL techniques have the potential to explore more deeply the complex interconnections between the input and output, we rely on them to build the relationships between the output of the abnormal return (AR) and inputs of exploration variables (In the study, they are C14 and the realized return of a stock). With the intention of improving the performance of DL models during training, we supply more informative contexts for them by adding extra returns to realized returns to create phenomena where stocks hold arbitrary abnormal returns, ranging from -3% to 3% in the interval of 0.2% and on an annual scale. Our hypothesis posits market efficiency, suggesting that stocks, on the whole, do not exhibit abnormal returns. Thus, in our context, the abnormal return of a stock is aligned with the extra return arbitrarily added to it.

We compare the power of testing the abnormal return between DL models and the other three benchmarks, such as the methods of Bessembinder, Lyno, and FF5F. DL models achieve a substantial improvement over all benchmarks in detecting the abnormal return within the scenarios with a range of -3% to 3% of induced ARs when using the random data set. Further, the performance of DL models is consistent in the non-random datasets categorized by the firm size and market condition. This consistency emphasizes the likelihood that method DL_LSTM exhibits a more uniform power in these two non-random samples, compared to Bessembinder.

DL models can yield well-specified test statistics in random samples akin to benchmark methods, and they also demonstrate enhancement in non-random datasets. More specifically, the model of DL_LSTM delivers well-defined test statistics in non-random sets, including the firm size, calendar clustering, and overlapping return samples. In the case of other

non-random samples such as Book-to-Market Ratio, Industry Clustering, and Pre-Event Return, DL models contribute to a reduction in the degree of misspecifications. This emphasizes the strength and efficacy of DL models across various data scenarios.

We addressed the reasons behind the potential enhancement power of testing the abnormal return by conducting the regression of the estimation error of the abnormal return on firms' characteristics, market conditions, and period dummies. We observe that the DD model displays lower sensitivity to factors such as firm size, market conditions, and periods, in contrast to the Bessembinder benchmark.

In summary, our study contributes to event studies and stock return analysis in several ways. First, we apply deep learning, specifically ANNs and LSTMs, to abnormal return estimation, offering a data-driven alternative to linear regression. Second, we model the relationship between abnormal returns, firm characteristics, and realized returns, broadening understanding. Additionally, our deep learning models outperform benchmarks in power tests. Finally, we propose a practical AR estimator framework, providing researchers with a valuable tool for event study analyses.

Our current research is preliminary. We aspire to enhance the DL model's performance by incorporating a range of additional factors and advanced architectures of artificial intelligence in our future work. Additionally, we anticipate the development of a robust business model that can serve as a valuable resource for fellow researchers who require accurate abnormal return calculations in the case study field.

Chapter Four

Turning Points Using Deep Learning

This chapter presents the third essay, which examines the role of deep learning (DL) in enhancing the time series momentum (TSMOM) rules, demonstrating its adaptability to market dynamics and its capacity to achieve superior risk-adjusted returns. Section 4.1 presents a brief overview of the essay and explains the motivation to conduct this study. Section 4.2 summarizes the literature review. Section 4.3 develops our methodology. Section 4.4 analyzes the empirical results. Section 4.5 concludes the essay.

Abstract

We evaluate the performance of our deep learning (DL) predictors in forecasting momentum turning points by applying the confusion matrix and comparing them to the benchmark model (GHM2) proposed by Goulding, Harvey, and Mazzoleni (2023). When tested on U.S. stocks from January 1990 to December 2023, our DL predictors show greater accuracy in identifying turning points than the benchmark. Additionally, Time series momentum (TSMOM) trading strategies incorporating our DL predictors yield significantly higher returns and Sharpe ratios compared to those using GHM2. Regression results further indicate that our models respond more effectively to fluctuations in stock and market return volatility than the benchmark.

Keywords: Time series momentum, Turning points, Deep Learning.

4.1 Introduction

Moskowitz, Ooi, and Pedersen (2012) first document the evidence of time-series momentum (denoted as TSMOM), which is present in diversified markets including equity indices, bonds, currencies, and commodities. They find that the past 12-month excess returns of an instrument could be an indicator to predict its future returns, which receives wide recognition and attention from academic literature[1].

The classic TSMOM trading rule states that if the 12-month (called the look-back window) excess return of an instrument is positive, a long position should be taken; if negative, a short position should be initiated. The trend-following strategies, such as TSMOM rules, based on the direction of trailing returns over a specific look-back horizon, generally perform well during clear uptrends or downtrends. However, at momentum turning points—where trends shift from upward to downward or vice versa—such strategies are vulnerable to poor performance. This is because they depend on historical returns, which often represent a mix of different trend regimes and market noise. Goulding, Harvey, and Mazzoleni (2023) and Goulding, Harvey, and Mazzoleni, (2024) emphasize the importance of predicting momentum turning points in financial markets, noting that traditional momentum strategies face challenges during these phases. Slow signals tend to respond too late, while fast signals often produce false alarms, making accurate prediction essential for effective investment decisions. They identify or predict a turning point for an asset as a month of t+1 in which its longer look-back horizon (e.g., from the month of t-11 to t) and shorter

[1] Such as Moreover and Chakrabarti, 2015; Marshall, Nguyen, and Visaltanachoti, 2017; Shi and Zhou, 2017; Guo, Han, Li, and Zhou, 2018; Qin, Pan, and Bai, 2020; Li and Liu, 2022; Zhang, 2022; Hanauer and Windmüller, 2023, etc.

look-back horizon (e.g., from the month of t-1 to t) momentum signals differ in their indications to buy or sell. Based on their turning points, they characterize four stock market cycles: Bull, Correction, Bear, and Rebound. Specifically, if both the 12-month and 2-month excess returns are positive (or negative), it indicates that the trend remains upward-Bull (or downward-Bear), and if the 12-month and 2-month excess returns show different signs, it reflects a Correction or Rebound. Their findings suggest that during sustained bull or bear markets, a long or short position should be maintained. However, during periods of correction or rebound, it may be advisable to shift positions into risk-free assets or adjust the balance of long and short positions, following strategies similar to those proposed in their research. They present trend-following strategies that adapt dynamically to asset turning points across multiple assets and asset classes. Their results demonstrate that turning point signals provide predictive information that enhances the performance of multi-asset trend-following strategies, particularly in the months following asset turning points, when compared to static trend strategies.

This research aims to accurately predict the occurrence of the TSMOM turning points of stocks by leveraging deep learning (DL) techniques and then achieve better performance than the method (called GHM2) presented in the paper of Goulding, Harvey, and Mazzoleni (2023) (hereafter, GHM). In recent times, DL has not only been widely applied in fields such as military, science, medicine, and linguistics, but it is also increasingly favored by researchers in the domains of economics and finance. DL algorithms present considerable advantages over conventional methods. By utilizing the strengths of artificial intelligence, these algorithms enable more effective forecasting of stock prices and returns[2].

In our study, we apply the classical DL models, such as Deep Neural Network (DNN) and the Long Short-Term Memory (LSTM) to forecast if a turning point will occur in the month of t+1 for a given stock. In order to make a fair comparison with GHM2, our DL models only collect excess returns in a rolling window of 12 months as models' inputs rather than add additional information, such as trade volumes, volatility, market conditions, etc. Our data includes all stocks in NYSE, NASDAQ, and AMEX, and training data for DL models ranges from Jan 1968 to Dec 1985, and the test period starts from Jan 1990 to Dec 2023. The testing period coincides exactly with the time frame used in GHM.

A confusion matrix is used to compare the prediction accuracy of turning points on all stocks traded in the test time period between our DL models and GHM2, and further, four relative measurements, such as Accuracy, Precision, Recall, and F1-score are calculated based on the results of the Confusion matrix. To clearly present our results from the confusion matrix on stock data, we report the counts of true positives, true negatives, false positives, and false negatives for the prediction of turning points in relevant stocks for each year from 1990 to 2023. Additionally, we provide four performance metrics—Accuracy, Precision, Recall, and F1-score—calculated in the same manner.

We also examine the profitability improvements of our models compared to GHM2. To this end, we apply both the DL model and the benchmark (GHM2) to all stocks during the test years, utilizing monthly returns. Specifically, we place a long (short) position in the month t+1 if the excess return in the 12-month look-back window (from the month of t-11 to

[2] Such as work Zhang, Zhai, Wang, 2021; Gu, Kelly, and Xiu, 2020; Messmer and Audrino, 2022; Sonkavde, Dharrao, Bongale, Deokate, Doreswamy. and Bhat, 2023: Chen, Pelger, and Zhu, 2024; etc.

t) is positive (negative) and the models predict no turning point for the month t+1. Otherwise, we shift the position out of the stock market into risk-free instruments. For each year from 1990 to 2023, we compile monthly mean returns, standard deviation of returns, and the Sharpe ratio produced by the models for stock trading.

Another interesting question is why our DL models can achieve high profitability. We attempt to explore the reasons behind this. Thus, we organize two linear regressions: a cross-sectional and a time series regression. For the cross-sectional one, we consider the characteristics of stocks as independent variables, such as capital value, liquidity in the market, and the volatility of stock returns. In the time series one, the market excess return, the volatility of market returns, credit spread, and the interest rate are included as independent variables. We then discuss the differences in regression results between our models and GHM2.

This paper contributes to the field of time series momentum (TSMOM) research in several ways. First, we introduce deep learning (DL) techniques to enhance the performance of TSMOM strategies, building upon the approach proposed by GHM. By predicting or identifying turning points using appropriate tools, researchers and traders can make informed decisions to close or adjust long and short positions when sustained momentum is likely to come to an end. Given the effective predictive capabilities of DL techniques, it is reasonable to expect them to outperform the method based on the interaction between shorter and longer look-back windows, as presented in benchmark research by GHM. The DL predictor is trained on time-series data. Second, we use the confusion matrix to compare the predictability of the turning point among our DL methods and the benchmark for each year from 1990 to

2023 on all relative stocks. The results from the confusion matrix indicate a substantial increase in the number of true positives and a significant decrease in false negatives from our models, compared to the results from GHM2 for each year. Four performance metrics—Accuracy, Precision, Recall, and F1-score—from DL models significantly outperform the results achieved by the benchmark. Third, DL models and GHM2 are applied to implement TSMOM trading rules. The statistical analysis of mean returns on stocks traded during the relevant years shows that the DL models achieve higher monthly mean returns with lower return volatility, resulting in a higher Sharpe ratio compared to the GHM2. Lastly, the results from the cross-sectional and time series regressions indicate that our DL models respond more effectively to changes in stock and market return volatility compared to the benchmark model, GHM2. This responsiveness supports the superior performance of the DL models relative to the benchmark. By designing ensemble methods that integrate multiple DL models and incorporating additional non-price and return information into the DL model—such as trade volumes, return volatility, market conditions, and others—we anticipate further performance improvements.

The remainder of this paper proceeds as follows: Section 4.2 summarizes the literature review. Section 4.3 develops our methodology relative to DNN and LSTM turning point prediction models. Section 4.4 analyzes empirical results. Section 4.5 concludes the paper.

4.2 Literature Review

Time-series momentum (TSMOM) is an asset pricing anomaly characterized by a persistent trend effect over specific time horizons. Moskowitz, Ooi, and Pedersen (2012)

were the first to identify TSMOM across a wide range of asset classes and markets, noting its low sensitivity to traditional risk factors and its challenge to the 'random walk' hypothesis. They demonstrated that the past 12-month excess return of an asset—whether from diverse asset classes such as 9 equity index futures, 13 bond index futures in developed markets, 10 currency forwards, or 24 different commodity futures—serves as a strong positive predictor of its future return. They point out that TSMOM is associated with the behavior (i.e., under-reaction and over-reaction) of traders.

The profitability of time-series momentum (TSMOM) strategies has garnered significant attention from researchers. D'Souza, Srichanachaichok, Wang, and Yao (2016) document nearly 100 years of TSMOM in individual stocks on the NYSE, AMEX, and NASDAQ, extending their analysis to 11 European countries from 1975. Similar to Moskowitz, Ooi, and Pedersen (2012) and He and Li (2015), they find positive predictors in the lagged returns' coefficients up to the 12th month. To explore the source of profitability, Marshall, Nguyen, and Visaltanachoti (2017) compare moving average rules with TSMOM. They document correlations, raw and risk-adjusted returns across stocks, size-quintile portfolios, and indices from 10 international markets. They conclude that both strategies yield higher Sharpe ratios than the buy-and-hold (BH) rule, produce positive Jensen alphas, and confirm the existence of TSMOM in stock markets. Shi and Zhou (2017) examine TSMOM in Chinese stock indices and portfolios. They find TSMOM in short-term look-back and holding periods, while contrarian effects emerge over longer periods. They also note that TSMOM is weaker in Chinese indices compared to the U.S. market, attributing this to data frequency rather than higher market efficiency in China. Guo, Han, Li, and Zhou (2018)

apply TSMOM to high-frequency ETF data, showing that the first half-hour return predicts the last half-hour return with significant results. They link profitability to higher volatility, trading volume, and key economic news. Chakrabarti (2015) presents evidence of TSMOM profitability during the 2004-2015 period, which can be characterized as a global economic cycle, across stock markets in the Asian, European, and U.S. regions. Other researchers have explored TSMOM across a broader range of asset classes and extended sample periods, reporting similar findings. (Georgopoulou and Wang, 2017; Hurst, Ooi, Pedersen, 2017; Babu, Levine, Ooi, Pedersen, and Stamelos, 2018; Su, 2021; Zakamulin and Giner, 2022; Sakkas and Urquhart, 2022; Karki and Bikram, 2024.).

The profitability of TSMOM in financial markets remains debated. While Moskowitz and Pedersen (2013) argue for 'value and momentum everywhere,' Huang, Li, Wang, and Zhou (2020) challenge this, providing evidence to refute Moskowitz et al. (2012). Using similar data and running pooled and asset-by-asset regressions, they find weak support for TSMOM, suggesting its performance in Moskowitz et al.'s work may stem from strategies relying on the historical mean. Kim, Tse, and Wald (2016) find that TSMOM profits are primarily driven by volatility scaling, with unscaled TSMOM alphas similar to unscaled long passive strategies. Analyzing 55 futures markets from 1984 to 2013, they show that unscaled TSMOM underperforms unscaled buy-and-hold strategies, while scaled TSMOM does not significantly outperform scaled buy-and-hold strategies. Zakamulin (2014) also questions the strong performance of TSMOM and moving average rules, calling them 'too good to be true.' He highlights potential data-mining biases, where researchers use full historical data to test multiple look-back and holding periods, then select the best results for back-testing.

Throughout out-of-sample tests with varying split points, Zakamulin finds that performance heavily depends on these choices, concluding that TSMOM's effectiveness is likely overstated.

Other researchers have sought to improve TSMOM strategies for enhanced performance. He and Li (2015) develop a continuous-time asset pricing model using heterogeneous agent models (HAMs) to explain the TSMOM phenomenon. The model involves three types of traders: fundamental, momentum, and contrarian. Fundamental traders rely on economic and financial information, while momentum and contrarian traders' base decisions on past price trends. They show that TSMOM's success depends on look-back periods (k) and market conditions. In momentum-dominated markets, TSMOM with short-term k outperforms, but when fundamental and contrarian traders dominate, TSMOM strategies can underperform. In 2018, He and Li (2015) extended this model to create an optimal strategy combining fundamentals, market timing, and TSMOM signals, showing it significantly outperforms the S&P 500 and pure TSMOM or reversal strategies. Qin, Pan, and Bai (2020) find that the optimal look-back period (k*) varies across assets and shifts over time. To address this, they propose a model that predicts asset-specific, time-varying k*, and evaluate TSMOM strategies using these predictions. Their findings reveal that incorporating time-varying k* significantly improves the predictability of TSMOM strategies, outperforming those based on historically optimal k*. Liu, Lu, and Wang (2021) examine time series momentum on 31 Chinese commodity futures from 2007 to 2019, noting several drawdowns. By creating a decision function based on upper and lower partial momentum, termed "Managed Time Series Momentum," they successfully improve the

reward-to-volatility ratio and systematically reduce drawdowns. Building on advances in Multi-Task Learning (MTL), Ong and Herremans (2023) introduce a DNN technique that jointly learns portfolio construction and auxiliary tasks, such as forecasting realized volatility using various estimators. Back testing on a diversified futures portfolio from 2000 to 2020, their model consistently outperforms existing TSMOM strategies, even with transaction costs up to 3 basis points.

Goulding, Harvey, and Mazzoleni (2023) and Goulding, Harvey, and Mazzoleni, (2024) emphasize a concept of the turning point mentioned in the cross-sectional momentum literature (Cooper, Gutierrez, and Hameed 2004; Daniel, Jagannathan, and Kim,2012; Daniel and Moskowitz, 2016), which can break the time series momentum of the instrument price movement. At and after momentum turning points, STMOM strategies risk poor performance as trailing returns reflect outdated trends. Shorter signals (e.g., a few months) exacerbate this by capturing noise rather than true trend shifts. They use the intersection of 12-month and 2-month trailing excess returns to identify a potential turning point for the current month, with a turning point indicated when signals from these two look-back windows differ to buy or sell. Their dynamic trend-following approach (Noted as GHM2) outperforms the classical TSMOM strategy.

Goulding, Harvey, and Mazzoleni (2023, 2024) improve upon earlier time-series momentum models by combining fast and slow trailing windows to predict momentum turning points, unlike traditional methods such as those by Moskowitz, Ooi, and Pedersen (2012) or He and Li (2015), which rely on a single fixed lookback window. Their model dynamically adapts to evolving market conditions, striking a balance between responsiveness

and reliability while reducing the drawbacks of slow signals reacting too late and fast signals generating false alarms. Furthermore, their approach enhances predictive accuracy and adaptability, enabling traders to anticipate trend reversals more effectively, ultimately leading to improved returns and better risk management compared to static momentum models.

Our research aims to enhance the predictability of the turning point by employing deep learning (DL) techniques, which have been validated as effective predictors based on big data resource, ultimately achieving improved performance compared to the approach presented in Goulding, Harvey, and Mazzoleni (2023).

The analysis and prediction of financial time series, including future stock prices, returns, and their movements, have long been a focal area of research. Traditional methods such as linear regression often struggle to capture the complex, non-linear relationships inherent in stock market data, while deep learning (DL) techniques are capable of learning from large datasets to identify intricate patterns. Patel, Shah, Thakkar, and Kotecha, (2015) highlighted the potential of Artificial Neural Networks (ANNs) in forecasting stock prices, emphasizing their capacity to capture non-linear relationships. Xia, Liu, Yu, Deng, Zhang, Su, and Zheng (2024) explore the application of neural networks in financial predictions, which informed our application of a Deep Neural Network (DNN) to model complex patterns in stock price movements. Recurrent Neural Networks (RNNs), especially Long Short-Term Memory (LSTM) networks, are widely recognized for their capacity to model time-dependent data. LSTMs are specifically designed to capture temporal dependencies, making them ideal for time series forecasting. Brownlee (2017) emphasized the effectiveness of LSTMs across diverse applications, such as predicting stock prices. Fischer and Krauss (2018) successfully

applied LSTMs to stock price prediction, demonstrating their capability to capture sequential patterns. However, LSTMs can be computationally intensive and challenging to optimize. Recently, ensemble methods combining multiple models have gained attention for harnessing the strengths of different algorithms. Yan, Zhao, and Zhou (2023) emphasized that ensembles can enhance predictive accuracy by reducing variance and bias. A study by Fernández-Delgado, Cernadas, Barro, Amorim (2014) further evaluated various ensemble techniques, confirming their effectiveness across diverse predictive tasks.

In this study, we will concentrate on classic DL models, specifically DNN and LSTM, to forecast the occurrence of the turning point. We will also follow the principles of the strategy in Goulding, Harvey, and Mazzoleni (2023) (denoted GHM2) to assess whether our method offers improvements over GHM2.

4.3 Methodology

This research builds on the methodology proposed by Goulding, Harvey, and Mazzoleni (2023). First, we replace their prediction method (denoted as GHM2) with DL techniques, specifically Deep Neural Network (DNN) and Long Short-Term Memory (LSTM) models, to forecast the turning point occurrences. We then apply both our models (denoted as DNNp and LSTMp) and their GHM2 to investment in all stocks traded on the three major U.S. stock markets (NYSE, NASDAQ, and AMEX) from January 1990 to December 2023.

4.3.1 The Prediction of the Turning Point

A turning point is defined as follows: if a stock's return in month t+1 is non-negative

(non-positive), while its excess return over the prior 12-month period (from t−11 to t) is negative (positive), then month t+1 is considered a turning point for that stock.

4.3.1.1 The Prediction Method of GHM2

The turning point prediction method in Goulding, Harvey, and Mazzoleni (2023), denoted as GHM2, is as follows: if a stock's excess return over a 2-month look-back period (from month t-1 to t) has a zero return or an opposite sign to its excess return over the prior 12 months (from month t-11 to t), then month t+1 is predicted as a turning point for that stock.

4.3.1.2 The Prediction Method of DL Techniques

We employ classic deep learning (DL) techniques, including Deep Neural Network (DNN) and Long Short-Term Memory (LSTM), to develop our prediction models—denoted as DNNp and LSTMp, respectively—aimed at forecasting the occurrence of the turning point. These models, in month t, directly predict whether month t+1 will be a turning point for a given stock based on relevant historical data fed into the models.

DNNp Model

The DNN component of our model includes Dense and Activation layers. Dense layers are used to project input training data into a high-dimensional feature space, supporting the effective learning of complex stock market patterns. These layers apply a linear transformation to the input by multiplying it with a weight and adding a bias, as shown in

equation (4.3.1), where w denotes the weight, x is the input data, and b is the bias (Balas, Roy, Sharma, and Samui, 2019).

$$y = x_n * w_n + b_n \qquad (4.3.1)$$

In the output layer, we apply the SoftMax activation function (see equation 4.3.2), a logistic function that outputs values between 0 and 1. The prediction corresponds to the value closest to 1, where z represents neuron values from the output layer.

$$Softmax(z_i) = \frac{e^{z_i}}{\sum_j e^{z_j}} \qquad (4.3.2)$$

The Flatten layer is mainly used to prepare data for subsequent Dense layers by converting multidimensional outputs into a one-dimensional format. We apply a linear activation to keep outputs unscaled, optimize learning using the Adam optimizer, and assess performance with Mean Squared Error (MSE) and Mean Absolute Error (MAE). Adam, introduced by Kingma (2014), is widely used in deep learning due to its combination of momentum and adaptive learning rates, which facilitates faster convergence during training.

Input:

To ensure a fair comparison with our benchmark GHM2, we use only the stock's excess returns as variables, specifically 12 variables representing the excess returns during the month t-11 to the month t.

Output:

DNNp model output either one or zero, representing the occurrence or non-occurrence of the turning point in month t+1, respectively.

Specifically, our Deep Neural Network (DNN) designed for turning point prediction consists of an input layer with 12 neurons, corresponding to 12 variables representing excess returns from the month t-11 to the month t. It includes three hidden layers with 128, 64, and 32 neurons, respectively, each using the ReLU (Rectified Linear Unit) activation function to introduce non-linearity. To prevent overfitting, dropout layers with a rate of 0.2 are applied after each hidden layer. The model is trained using the Adam optimizer with a learning rate of 0.001 and employs binary cross-entropy as the loss function. Batch normalization is incorporated to stabilize learning, and L2 regularization (lambda = 0.01) is used to control model complexity. The output layer consists of a single neuron with a linear activation function, SoftMax, predicting the turning point. The model is trained over 100 epochs with a batch size of 32 to ensure optimal generalization and performance.

Training Data:

We collect monthly data from the NYSE, NASDAQ, and AMEX, covering all stocks traded from January 1868 to December 1985. Missing stock returns are filled with the stock's average return, and untraded dates are excluded. The dataset includes 9,747 stocks with a minimum of 25 months of data, totaling 1,082,174 records. To ensure consistency, we use the same 25-month minimum required for LSTMp model.

LSTMp Model

The Long Short-Term Memory (LSTM) network enhances traditional Recurrent Neural Networks (RNNs) for time series prediction. LSTMs introduce three gates—forget, input, and output—to effectively mitigate the vanishing gradient issue, leading to improved performance and making them highly popular for time series modeling.

An LSTM consists of multiple neurons, each processing data sequentially through specific gates. First, the forget gate discards irrelevant information from the input. Next, the input gate selects new information to retain, combining the output of the previous neuron with the current input, processed through sigmoid and tanh functions to generate updated values. Finally, the output gate decides what information from the input gate to pass forward, feeding the output of each neuron into the next.

Input:

The input data for an LSTM is a three-dimensional array, organized into time, sample, and feature dimensions. The time dimension corresponds to the sliding time window, the sample dimension represents the number of samples in training and testing, and the feature dimension holds the input features. We use a 12-month time window to forecast the occurrence of the turning point for the 13th month, with 12 features corresponding to the excess returns of stocks from months 1 through 12.

Output:

LSTMp model output either one or zero, representing the occurrence or non-occurrence of the turning point in month t+1, respectively.

In our specific Long Short-Term Memory (LSTM) model configuration, we use 12 features as input variables and a time step of 12, meaning the model processes sequential data

across 12 time periods to capture temporal dependencies effectively. The architecture consists of an input layer that feeds data into two stacked LSTM layers, each with 64 hidden units, enabling the model to learn long-term dependencies in financial time series. A dropout layer with a rate of 0.2 follows each LSTM layer to prevent overfitting. The output from the LSTM layers is passed through three fully connected dense layers, with 128, 64, and 32 neurons, respectively, followed by a final output layer with a single neuron for the prediction of the occurrence of turning points. The model is optimized using the Adam optimizer with a learning rate of 0.001, and the loss function is set to binary cross-entropy to measure prediction accuracy.

Training Data:

Similar to the training of DNNp, the training data for LSTMp also includes 9,747 stocks from the three main U.S. markets, each with a minimum of 25 months of data, totaling 1,082,174 samples from January 1868 to December 1985.

4.3.2 The Assessment Method on Prediction Performance of the Turning Point

We assess the predictive performance of DNNp, LSTMp, and the benchmark GHM2 on the test dataset (January 1990 to December 2023) using a confusion matrix, which includes four metrics: true positive (TP), true negative (TN), false positive (FP), and false negative (FN). Here, a turning point is labeled as positive (1) if it occurs, and negative (0) otherwise. Each model provides monthly predictions on the likelihood of a turning point. Four metrics are defined as follows.

True Positive (TP): Prediction is positive (1) when the actual outcome is also positive (1).

False Positive (FP): Prediction is positive (1), but the actual outcome is negative (0).

True Negative (TN): Prediction is negative (0) when the actual outcome is also negative (0).

False Negative (FN): Prediction is negative (0), but the actual outcome is positive (1).

Based on the four metrics above, we calculate Accuracy, precision, recall, and F1 score—to further assess the predictive performance of our DL models and the benchmark. For our DL model evaluation, which predicts the occurrence of turning points, selecting Accuracy, Precision, Recall, and F1-score ensures a well-rounded performance assessment. Accuracy provides an overall measure of correctness but may be misleading if turning points are rare events. Precision is critical because false positives—incorrectly predicting a turning point—can lead to unnecessary trades and increased transaction costs. Recall, on the other hand, ensures that actual turning points are detected, reducing the risk of missing significant market shifts. The F1-score, balancing Precision and Recall, is particularly useful in this context, as it accounts for both missed opportunities and false alarms. Given the high-impact nature of market turning points, relying on these metrics helps refine the model's predictive ability, enhancing its practical utility for traders and analysts.

Accuracy: This metric reflects the proportion of correct predictions (both true positives and true negatives) over all predictions made. It is calculated as the equation of 4.3.3:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \qquad (4.3.3)$$

Precision: Also known as the Positive Predictive Value (PPV), precision measures how many of the predicted positive instances are actually positive. It is defined as the equation of 4.3.4:

$$Precision = \frac{TP}{TP + FP} \qquad (4.3.4)$$

Recall: Also called sensitivity, recall quantifies how many actual positive instances are correctly identified by the model. It is calculated as the equation of 4.3.5:

$$Recall = \frac{TP}{TP + FN} \qquad (4.3.5)$$

F1 Score: This metric, also known as F-score or F-measure, is the harmonic mean of precision and recall, providing a balanced measure of the model's performance. It is especially useful when there is an uneven class distribution. The F1 score is computed as:

$$F1\ Score = \frac{2 * Precision * Recall}{Precision + Recall} \qquad (4.3.6)$$

Each of these metrics offers insights into different aspects of model performance, with the F1 score providing a single measure that balances precision and recall.

4.3.3 Trading Strategies

We apply DNNp, LSTMp, and the benchmark model, GHM2, on U.S stocks, including 24767 stocks traded during January 1990 to December 2023. For each stock in each month t+1, the return is as follows in the equation of 4.3.7:

$$r_{t+1} = \begin{cases} r_{t+1}, & if\ \ LB12 > 0\ and\ non-turning\ point\ in\ month\ t+1. \\ rf_{t+1}, & if\ turning\ point\ in\ month\ t+1. \\ -r_{t+1}, & if\ LB12 < 0\ and\ non-turning\ point\ in\ month\ t+1. \end{cases} \qquad (4.3.7)$$

In the equation above, r_{t+1} represents the return for the given stock in month t+1, rf_{t+1} denotes the risk-free return in month t+1, and LB12 refers to the 12-month (from month t-11 to month t) excess return preceding month t+1.

4.4 Results

4.4.1 Data

The data for our research is sourced from the Wharton Research Data Services (WRDS) database and includes monthly stock data from the three primary U.S. stock markets: NYSE, NASDAQ, and AMEX. To match our model's data structure, we retain only stocks with a minimum of 25 successive trading months. The training dataset spans January 1968 to December 1985, while the testing dataset covers January 1990 to December 2023, aligning with the benchmark GHM2's test period. Data from January 1988 to December 1989 fills the 12-month look-back window for initial testing in LSTMp, DNNp, and GHM2. The 1986–1987 period serves as a buffer to prevent data leakage. Missing stock return data is imputed with the average return for the given stock, while data from untraded periods is removed. Data on market returns and risk-free rates are sourced from Fama-French Portfolios and Factors in WRDS, while credit spread data is obtained from Moody's website.

4.4.2 Performance of the Turning Point Prediction

Time Series Momentum (TSMOM) strategies seek to capture asset momentum by taking long positions during "uptrend" phases and short positions during "downtrend" phases. Trade signals are typically derived from excess returns over a specified look-back period, often 12 months. A positive excess return over this period signals a long position, while a negative excess return signals a short position. At momentum turning points, where trends shift from uptrend to downtrend or vice versa, TS momentum strategies often make poor bets due to their reliance on observed returns, which reflect a mixture of different trend regimes

and noise. A longer look-back window (e.g., 12 months) in momentum signals can reduce noise but is slow to detect turning points (Type II error). Conversely, a shorter window (e.g., 1 month) responds quickly to turning points but is more affected by noise (Type I error). Building on the idea generated by Goulding, Harvey, and Mazzoleni (2023), we define a momentum turning point as follows: if a stock's return in month t+1 is non-negative (non-positive), while its excess return over the prior 12-month period (from t−11 to t) is negative (positive), then month t+1 is considered a turning point for that stock.

Goulding, Harvey, and Mazzoleni (2023) developed a method, GHM2, to predict turning points, which enhances the profitability of the classic TSMOM strategy that relies on a 12-month look-back period. In this research, we use our models, DNNp and LSTMp, to predict turning points to enhance predictability compared to GHM2. We then seek to further improve the TSMOM strategy based on these predictions.

4.4.2.1 Prediction Performance Over the Entire Test Period (1990-2023)

Figure 4.1 displays the confusion matrix results for three turning point prediction models applied to all stocks over the period from 1990 to 2023, with a total sample size of 2,979,106. TP, FP, TN, and FN are explained in Section 4.3.2. Figure 4.1 results indicate that the DNNp and LSTMp models significantly increase the number of true positives (TP) from 477,373 in GHM2 to 729,788 and 786,662, respectively, with only a slight decrease in true negatives (TN) from 1,065,013 to 946,312 and 975,213. This indicates that the DNNp and LSTMp models are more accurate in predicting turning points than GHM2, though they are slightly less accurate in identifying non-turning point months.

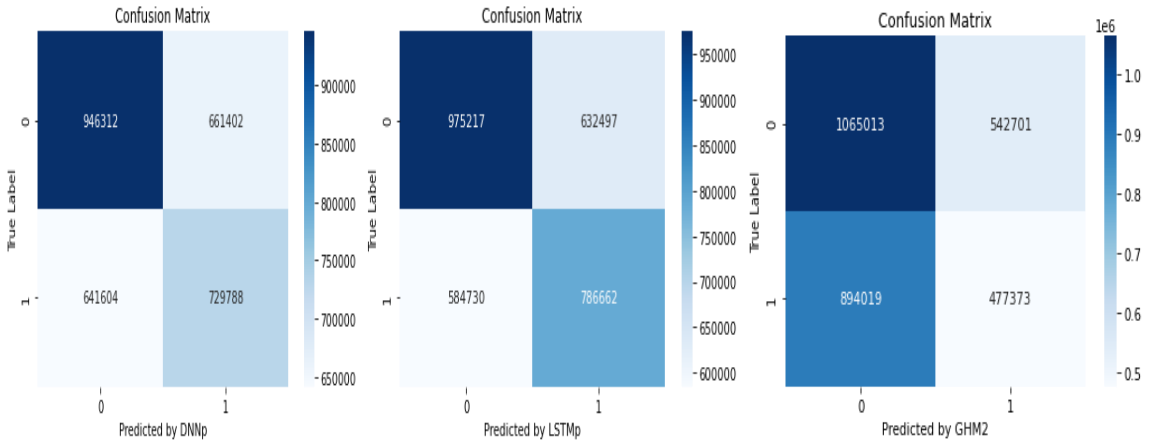

Figure 4.1 Confusion Matrix for Models of DNNp, LSTMp, and GHM2

This figure displays the confusion matrix results for three turning point prediction models (DNNp, LSTMp, and GHM2) applied to all stocks over the period from 1990 to 2023, with a total sample size of 2,979,106. The vertical axis represents the actual outcome, where 1 indicates a turning point and 0 a non-turning point. The horizontal axis shows the predicted outcome from each of the three models, where 1 and 0 similarly indicate a turning point and a non-turning point, respectively. True Positive (TP): Prediction is positive (1) when the actual outcome is also positive (1). False Positive (FP): Prediction is positive (1), but the actual outcome is negative (0). True Negative (TN): Prediction is negative (0) when the actual outcome is also negative (0). False Negative (FN): Prediction is negative (0), but the actual outcome is positive (1).

Table 4.1 displays four metrics—accuracy, precision, recall, and F1 score—derived from the confusion matrix results for the DNNp, LSTMp, and benchmark GHM2 models. The definition of these four metrics is expressed in Section 4.3.2. Both DNNp and LSTMp demonstrate substantial improvements over GHM2 in predictive performance. Accuracy for DNNp and LSTMp stands at 0.5626 and 0.5914, respectively, notably higher than GHM2's 0.5177. Precision for DNNp and LSTMp reaches 0.5246 and 0.5542, respectively, surpassing GHM2's 0.4679. Notably, recall for DNNp and LSTMp rises substantially compared to GHM2's 0.3481, reaching 0.5315 and 0.5736, respectively, indicating that our models predict turning points more accurately than the benchmark. The F1 score follows a similar trend to recall. Overall, LSTMp achieves the highest predictive performance among the three models.

Table 4.1 Four Metrics from Confusion Matrix

This table presents four metrics—accuracy, precision, recall, and F1 score—to evaluate the predictive performance of our DNNp and LSTMp models compared to the benchmark GHM2 across the full test period from 1990 to 2023.

	Accuracy	Precision	Recall	F1 Score
GHM2	0.5177	0.4679	0.3481	0.3992
DNNp	0.5626	0.5246	0.5315	0.5280
LSTMp	0.5914	0.5543	0.5736	0.5638

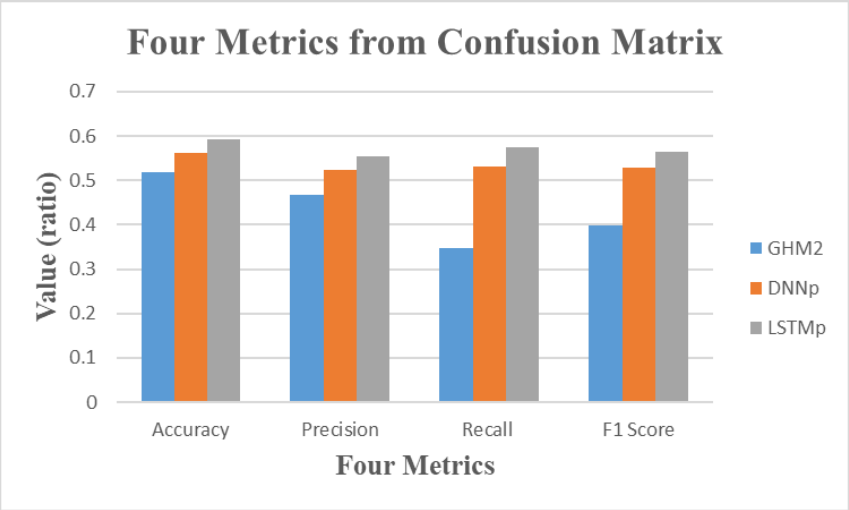

Figure 4.2 Four Metrics from Confusion Matrix

This figure displays four metrics—accuracy, precision, recall, and F1 score—to evaluate the predictive performance of our DNNp and LSTMp models compared to the benchmark GHM2 across the full test period from 1990 to 2023.

We visualize the results from Table 4.1 in Figure 4.2 to clearly illustrate the differences in four key metrics—accuracy, precision, recall, and F1-score—across three models: DNNp, LSTMp, and GHM2.

Although our DL models outperform the benchmark GHM2 in predicting turning points, the four-evaluation metrics remain relatively low. For example, the accuracy values for DNNp and LSTMp are 0.5626 and 0.5914, respectively, indicating the need for further

improvement before applying this strategy in practical investments.

4.4.2.2 Annual Prediction Performance (1990–2023)

To explore the predictability of these models in detail, we further complete the confusion matrix for each model annually from 1990 to 2023. Figure 4.3 shows the annual counts of TP, TN, FP, and FN for the DNNp, LSTMp, and GHM2 models from 1990 to 2023. The annual prediction performance results for these models show a similar pattern to those

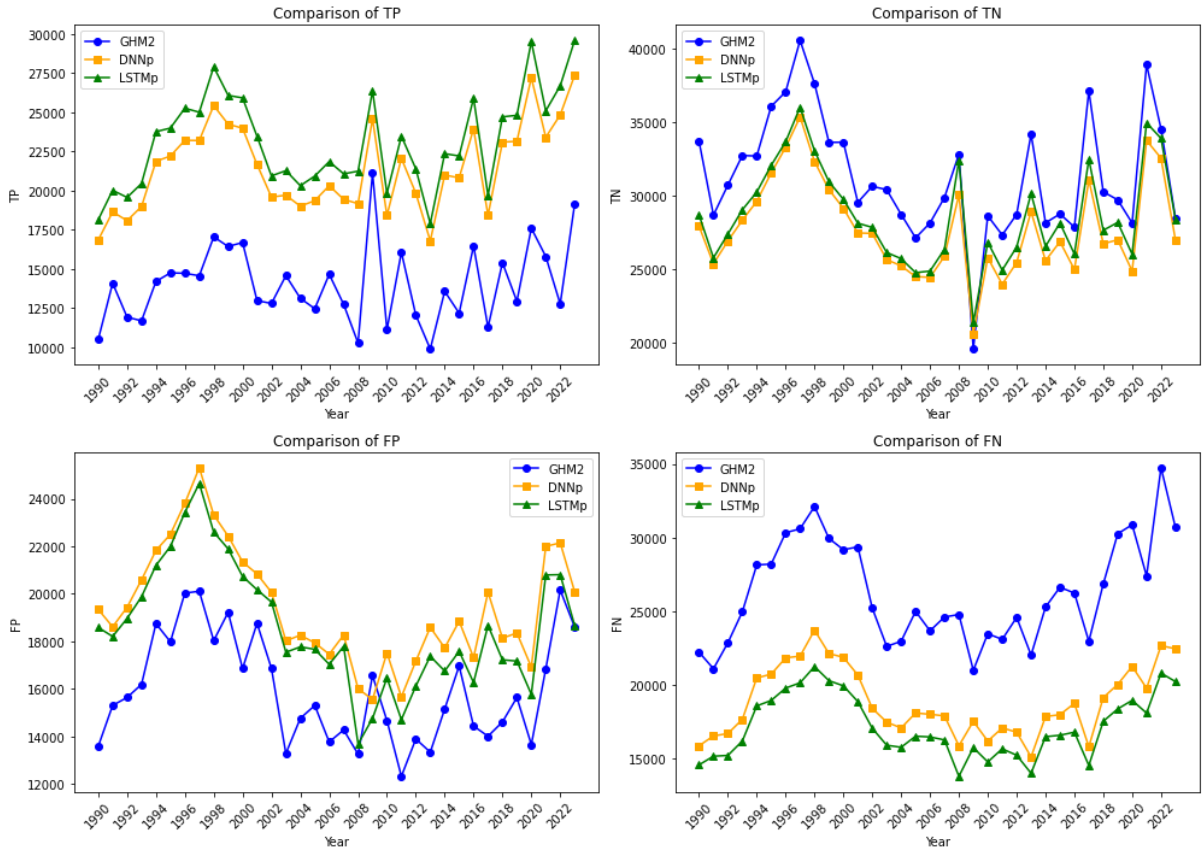

Figure 4.3 Annual TP, TN, FP, and FN

This figure displays the confusion matrix results for three turning point prediction models (DNNp, LSTMp, and GHM2) applied to stocks over each year from 1990 to 2023. TP, TN, FP, and FN stand for true positive (turning point), true negative (non-turning point), false positive, and false negative, respectively. True Positive (TP): Prediction is positive (1) when the actual outcome is also positive (1). False Positive (FP): Prediction is positive (1), but the actual outcome is negative (0). True Negative (TN): Prediction is negative (0) when the actual outcome is also negative (0). False Negative (FN): Prediction is negative (0), but the actual outcome is positive (1).

observed over the entire test period. For each year from 1990 to 2023, the TP and FN counts for DNNp and LSTMMp are considerably higher and lower, respectively, compared to GHM2, whereas the TN and FP counts for DNNp and LSTMp are only slightly lower and higher, respectively, than those for GHM2. DL models appear to outperform the benchmark in the prediction of TP (the occurrences of the turning point), with LSTMp remaining the most effective predictor among the three models.

Figure 4.4 shows the annual counts of prediction tests conducted from 1990 to 2023. Results indicate that the minimum number of tests is 77,902 in 1990, with a maximum of 105,820 in 2010.

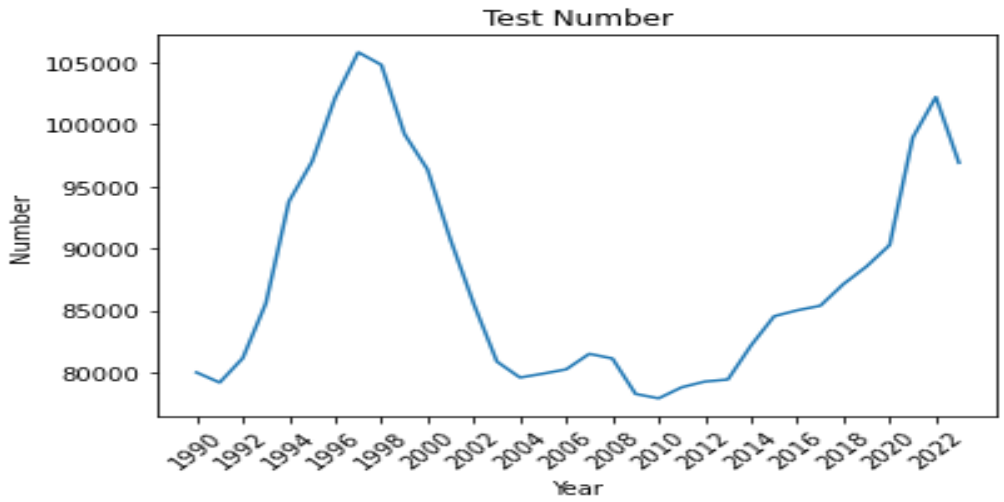

Figure 4.4 Annual Test Number for the Turning Point Prediction During 1990 to 2023
This figure shows the annual counts of prediction tests conducted from 1990 to 2023.

Figure 4.5 exhibits four annual metrics -Accuracy, Precision, Recall, and F1 score, calculated from the confusion matrix results for three turning point prediction models (DNNp, LSTMp, and GHM2) from 1990 to 2023.

Figure 4.5 demonstrates that our models, DNNp and LSTMp, consistently outperform the benchmark across all four metrics each year from 1990 to 2023. LSTMp performs as the best predictor. In 2008, both DNNp and LSTMp achieve their highest prediction accuracy at 0.6069 and 0.6612, respectively, showing a significant improvement over GHM2's 0.5311. This suggests that using DL predictors could reach better performance in this major crash year, compared to the benchmark.

Figure 4.5 Annual Four Metrics-Accuracy, Precision, Recall, and F1 score

This figure displays annual four metrics -Accuracy, Precision, Recall, and F1 score, calculated from the confusion matrix results for three turning point prediction models (DNNp, LSTMp, and GHM2) from 1990 to 2023.

4.4.3 Profitability of TSMOM Strategies with DNNp, LSTMp, and GHM2

We respectively integrate turning point prediction models, such as DNNp, LSTMp, and GHM2, into the classic time series momentum (TSMOM) strategy, using a 12-month look-back window to invest in U.S. stock markets, encompassing 24,767 stocks traded from January 1990 to December 2023. If a stock's excess return over a 12-month look-back window (the month t-11 to t) is positive (negative) and the prediction model forecasts no turning point for the month t+1, we take a long (short) position in month t+1 on the stock. Otherwise, we exit the stock market and allocate to a risk-free asset. For each stock in each month t+1, the return is calculated according to the equation provided in Section 4.3.3.

Table 4.2 presents the statistics of mean log returns of stocks, the standard deviation of returns, and the Sharpe ratio for trading strategies based on the DNNp, LSTMp, and GHM2 prediction models, covering the entire test period from 1990 to 2023. TS_GHM2, TS_DNNp, and TS_LSTMp represent the 12-month TSMOM strategy integrated with different turning point predictors: GHM2, DNNp, and LSTMp, respectively. Table 4.2 results show that TS_DNNp and TS_LSTMp achieve significantly higher monthly mean log returns and Sharpe ratios, along with lower return standard deviations, compared to the TS_GHM2 strategy. These improvements are both economically and statistically significant.

Specifically, we compile statistics on monthly mean log returns, monthly return standard deviations, and annualized Sharpe ratios for all 24,767 stocks. The TS_DNNp and TS_LSTMp rules achieve average monthly mean log returns of 0.0148 and 0.0177, respectively, compared to 0.0108 from the TS_GHM2 rule. They also show lower average return standard deviations of 0.1168 and 0.01151, versus 0.338 from TS_GHM2, and higher

Table 4.2 Profitability of TSMOM Strategies with DNNp, LSTMp, and GHM2

This table presents the statistics of monthly mean log returns of stocks, the standard deviation of returns, and the annualized Sharpe ratio for trading strategies based on the DNNp, LSTMp, and GHM2 prediction models, covering the entire test period from 1990 to 2023. Mean_logreturn refers the average log return of the stock derived from TSMOM strategies with different turning point prediction models. TS_GHM2, TS_DNNp, and TS_LSTMp represent the 12-month TSMOM strategy integrated with different turning point predictors: GHM2, DNNp, and LSTMp, respectively.

	Mean_logreturn (Monthly)							
	Mean	DiffMean	Std	1%	25%	50%	75%	99%
TS_GHM2	0.0108		0.0376	-0.0436	-0.0016	0.0039	0.0149	0.1323
TS_DNNp	0.0148	0.0040***	0.0299	-0.0315	0.0024	0.0083	0.0193	0.1191
TS_LSTMp	0.0177	0.0069***	0.033	-0.0263	0.0044	0.0109	0.0231	0.1235
	Standard deviation of monthly stock returns							
	Mean	DiffMean	Std	1%	25%	50%	75%	99%
TS_GHM2	0.1338		0.0923	0.0073	0.0064	0.1149	0.1006	0.4256
TS_DNNp	0.1168	-0.0170***	0.0816	0.0061	0.0565	0.1009	0.1579	0.3667
TS_LSTMp	0.1151	-0.0187***	0.08	0.0057	0.0557	0.0991	0.1552	0.3761
	Annualized Sharpe Ratio							
	Mean	DiffMean	Std	1%	25%	50%	75%	99%
TS_GHM2	0.3468		2.3268	-4.5144	-0.5928	0.2604	1.2072	5.544
TS_DNNp	0.9828	0.6360***	1.9104	-0.4248	0.0732	0.9312	1.8432	5.9652
TS_LSTMp	1.3512	1.0044***	2.1108	-3.4392	0.4584	1.2972	2.2128	6.1356
The number of stocks traded in Jan 1990-Dec 2023: 24767								

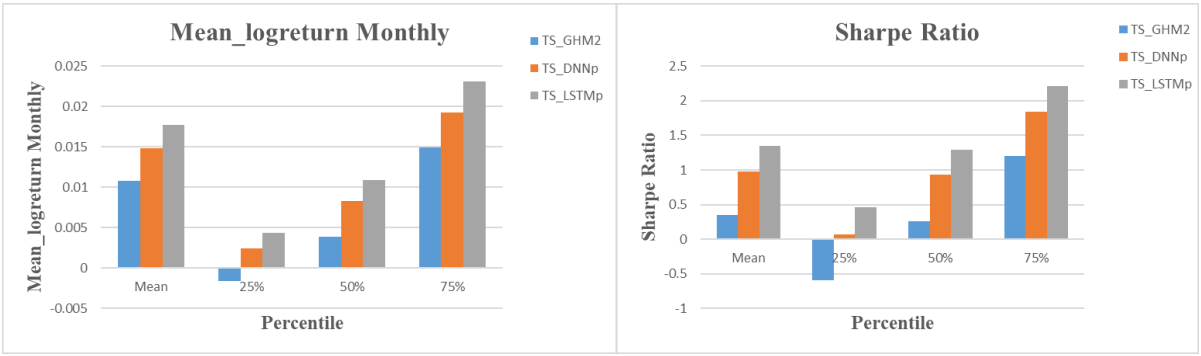

Figure 4.6 Profitability of TSMOM Strategies with DNNp, LSTMp, and GHM2

This figure displays the statistics of monthly mean log returns of stocks and the annualized Sharpe ratio for trading strategies based on the DNNp, LSTMp, and GHM2 prediction models, covering the entire test period from 1990 to 2023. Mean_logreturn refers to the average log return of the stock derived from TSMOM strategies with different turning point prediction models. TS_GHM2, TS_DNNp, and TS_LSTMp represent the 12-month TSMOM strategy integrated with different turning point predictors: GHM2, DNNp, and LSTMp, respectively.

average Sharpe ratios of 0.9828 and 1.3512, compared to 0.3468 from the TS_GHM2 model. Once again, LSTMp demonstrates the best performance, yielding the highest profits among the three predictors. The median (50th percentile) results align with the pattern observed in the mean results.

We also present a visual representation of the results from Table 4.2 in Figure 4.6, illustrating the Mean Log Return (monthly) and Sharpe Ratio for three trading strategies—TS_GHM2, TS_DNNp, and TS_LSTMp. This visualization aims to provide a clearer comparison of their performance differences.

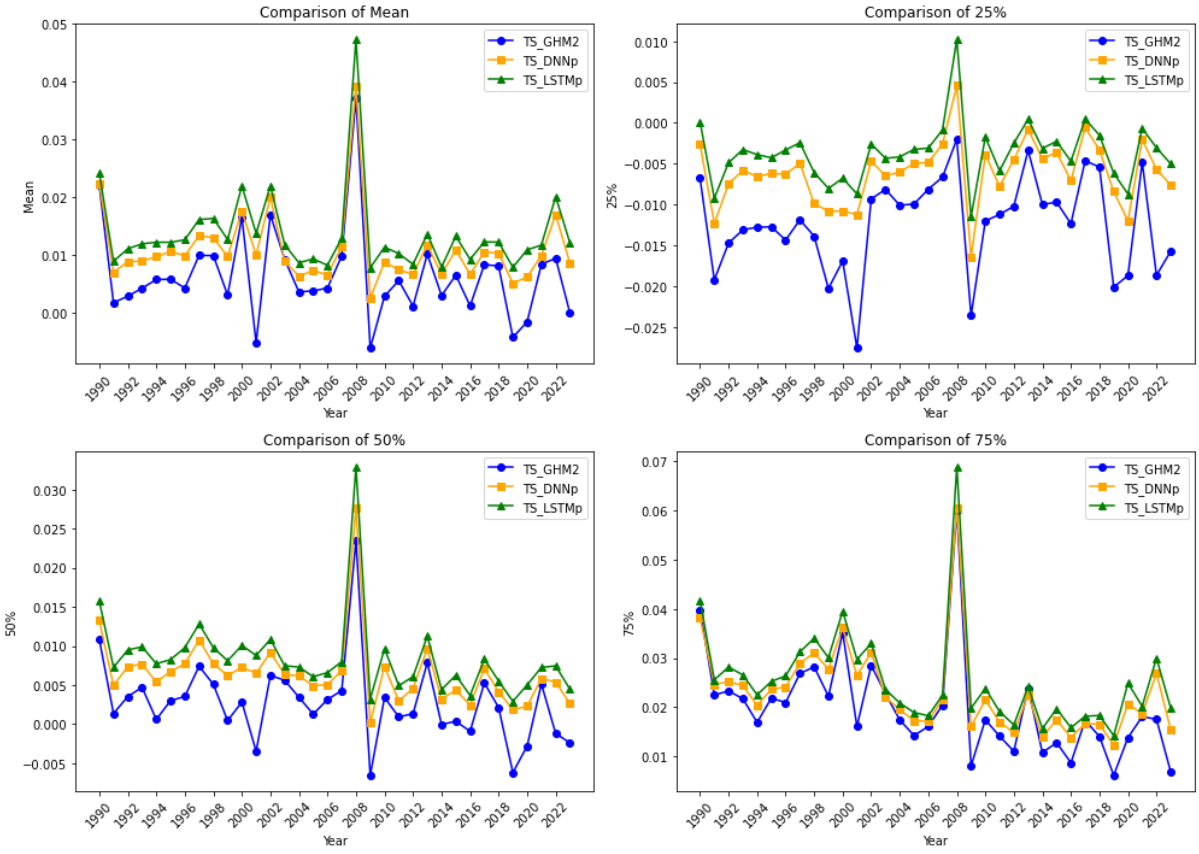

Figure 4.7 Annual statistics of monthly mean log returns from 1990 to 2023

This figure presents the annual statistics for the mean, 25th, 50th (median), and 75th percentiles of monthly mean log return on stocks from 1990 to 2023. TS_GHM2, TS_DNNp, and TS_LSTMp represent the 12-month TSMOM strategy combined with different turning point predictors: GHM2, DNNp, and LSTMp, respectively. The vertical axis shows monthly mean log returns for each statistical measure: mean, 25th, 50th, and 75th percentiles.

Figure 4.7 presents the annual statistics for the mean, 25th, 50th (median), and 75th percentiles of monthly mean log return on stocks from 1990 to 2023. TS_GHM2, TS_DNNp, and TS_LSTMp represent the 12-month TSMOM strategy combined with different turning point predictors: GHM2, DNNp, and LSTMp, respectively. Annual results in Figure 4.7 mirror the patterns observed in Table 4.2, with TS_DNNp and TS_LSTMp outperforming the benchmark rule, TS_GHM2. Notably, in the top left graph of Figure 4.7, the yearly averages of monthly mean log return from TS_DNNp and TS_LSTMp consistently exceed those from TS_GHM2. Additionally, while TS_DNNp and TS_LSTMp maintain positive average monthly mean log returns across all test years, TS_GHM2 shows four years (2001, 2009, 2019, and 2020) with negative averages.

4.4.4 Robust Check of Profits for Trading Rules

Transaction costs commonly impact profit assessments for various TSMOM strategies. In this section, we account for these costs to further evaluate the profitability of the TS_DNNp, TS_LSTMp, and TS_GHM2 trading rules, as well as to compare them with the buy-and-hold (BH) strategy. Here, TS_GHM2, TS_DNNp, and TS_LSTMp represent the 12-month TSMOM strategy combined with different turning point predictors: GHM2, DNNp, and LSTMp, respectively.

We focus on the one-way transaction cost. Lynch and Balduzzi (2000) recommend a value of 25 basis points for stocks, while Glabadanidis (2015) utilizes a figure of 50 basis points. Considering the actual transaction costs in U.S. stock markets, we assume that the

one-way trading costs for stocks are 50 basis points (0.5%) and 0.5 basis points (0.005%) for risk-free investments.

Table 4.3 presents the statistics for monthly mean log returns of stocks and the annualized Sharpe ratio for trading strategies that include one-way transaction costs over the full test period from 1990 to 2023. Trading rules TS_DNNp, TS_LSTMp, and TS_GHM2 are applied based on the DNNp, LSTMp, and GHM2 prediction models, respectively.

Table 4.3 Profitability of TSMOM Strategies Accounting for the One-Way Transaction Cost

This table presents the statistics for monthly mean log returns of stocks and the annualized Sharpe ratio for trading strategies that include one-way transaction costs over the full test period from 1990 to 2023. Trading rules TS_DNNp, TS_LSTMp, and TS_GHM2 are applied based on the DNNp, LSTMp, and GHM2 prediction models, respectively. Mean_logreturn refers the average log return of the stock derived from TSMOM strategies with different turning point prediction models. DMean1 denotes the mean difference between the BH and TSMOM strategies, while DMean2 represents the mean difference between TS_GHM2 and our models, TS_DNNp and TS_LSTMp. TS_GHM2, TS_DNNp, and TS_LSTMp represent the 12-month TSMOM strategy integrated with different turning point predictors: GHM2, DNNp, and LSTMp, respectively.

	Mean_logreturn (monthly)								
	Mean	DMean1	DMean2	Std	1%	25%	50%	75%	99%
BH	-0.0140			0.0464	-0.4035	-0.0211	0.0013	0.0079	0.0388
TS_GHM2	0.0099	0.0239***		0.0378	-0.0456	0.0004	0.0018	0.0107	0.1312
TS_DNNp	0.0135	0.0275***	0.0036***	0.0299	-0.0339	0.0000	0.0091	0.0171	0.1170
TS_LSTMp	0.0163	0.0303***	0.0064***	0.0335	-0.0286	0.0020	0.0115	0.0207	0.1215
	Annualized Sharpe Ratio								
	Mean	DMean1	DMean2	Std	1%	25%	50%	75%	99%
BH	-0.4192			2.6344	-6.8750	-1.4843	-0.0781	0.8748	4.2535
TS_GHM2	0.2069	0.6261***		2.4366	-5.4111	-0.9041	0.0145	1.0037	5.4852
TS_DNNp	0.7092	1.1284***	0.5023***	2.1846	-5.5031	0.0408	0.7216	1.5066	5.7420
TS_LSTMp	1.0442	1.4634***	0.8373***	2.3718	-0.0328	0.0852	1.0070	1.8456	5.8791
The number of stocks traded in Jan 1990 to Dec 2023: 24767									

Overall, when accounting for transaction costs in the trading rules, the results in Table 4.3 reflect the same pattern observed in Table 4.2. Table 4.3 shows that our models,

TS_DNNp and TS_LSTMp, significantly outperform the benchmark model TS_GHM2 in both average mean log returns of stocks and the risk-adjusted measure, as indicated by average Sharpe ratios. Additionally, all three TSMOM rules outperform the BH rule, with the TS_LSTMp model showing the highest performance among all models. Specifically, the ST_DNNp and TS_LSTMp models achieve average mean log returns of 0.0135 and 0.0163, respectively, both economically and statistically higher than the 0.0099 from GHM2 and -0.0140 from BH. The 50th percentile mean log returns follow a similar pattern. Likewise, TS_DNNp and TS_LSTMp attain Sharpe ratios of 0.7092 and 1.0442, respectively, compared to 0.2069 for the benchmark TS_GHM2.

4.4.5 Discussion on Regressions

In this section, we aim to explore why our trading models achieve superior profitability, using cross-sectional and time series linear regression models. These two regression analyses are organized as follows.

The cross-sectional linear regression model (4.4.5.1):

$$logreturn_i = b0_i + b1_i * Spread_i + b2_i * LogSize_i + b3_i * Turnover_i + b4_i * Std_i + e_i. \quad (4.4.5.1)$$

Here, the dependent variable, $logreturn_i$, represents the mean log return of stock i, while the independent variables—$Spread_i$, $LogSize_i$, $Turnover_i$, and Std_i—correspond to the average transaction spread, log of market capitalization, turnover, and monthly standard deviation of returns for stock i, respectively.

The time series linear regression model (4.4.5.2):

$$logreturn_t = a0_t + a1_t * rf_t + a2_t * mktrf_t + a3_t * std\_mktrf_t + a4_t * creditSpread_t + q_t. \quad (4.4.5.2)$$

In this context, the dependent variable, $logreturn_t$, represents the average log return for all stocks traded in month t. The independent variables—rf_t, $mktrf_t$, std_mktrf_t, and $creditSpread_t$—represent the monthly risk-free rate, market excess return, standard deviation of daily market returns, and monthly credit spread for month t across all traded stocks, respectively.

Table 4.4 The Cross-Sectional Linear Regression Model Result

This table presents the results from cross-sectional liner regression models of 4.4.5.1 for trading rules of TS_DNNp, TS_LSTMp, and TS_GHM2.Trading rules TS_DNNp, TS_LSTMp, and TS_GHM2 refer to trading strategies based on the DNNp, LSTMp, and GHM2 prediction models, respectively. Here, the dependent variable, $logreturn_i$, represents the mean log return of stock i, while the independent variables—$Spread_i$, $LogSize_i$, $Turnover_i$, and Std_i—correspond to the transaction spread, log of market capitalization, turnover, and monthly standard deviation of returns for stock i, respectively. The number of stock observations covers trades from January 1990 to December 2023.

	TS_GHM2	TS_DNNp	TS_LSTMp
Dep. Variable	Logret	Logret	Logret
Indep. Variables:			
Spread	0.007***	0.0046***	0.0051***
	5.44	4.66	4.66
LogSize	-0.0006***	-0.0003**	-0.0002*
	-4.71	-2.14	-1.93
Turnover	0.1981***	0.14***	0.1759***
	8.83	8.216	9.14
Std	0.103***	0.1161***	0.146***
	39.64	58.57	66.45
R-squared	0.094	0.168	0.195
Number of observations: 24767			

Table 4.4 presents the results of the cross-sectional linear regression model outlined in equation 4.4.5.1 for the TS_DNNp, TS_DNNp, and TS_GHM2 trading rules. In the regression for TS_GHM2, all independent variable coefficients are significant at the 1% level. However, in the regressions for TS_DNNp and TS_LSTMp, the LogSize coefficient is only significant at the 5% and 10% levels, suggesting that our models, particularly TS_LSTMp, are less affected by stock capitalization. Additionally, the t-statistics for the Std variable coefficients increase to 58.57 and 66.45 for TS_DNNp and TS_LSTMp, respectively, compared to TS_GHM2, indicating that our models are more sensitive to the standard deviation of stock returns. The R-squared values for TS_DNNp and TS_LSTMp rise to 0.168 and 0.195, respectively, while TS_GHM2's R-squared is 0.094.

Table 4.5 presents the results from time series linear regression models explained in the equation of 4.4.5.2 for trading rules of TS_DNNp, TS_LSTMp, and TS_GHM2. Our models differ from TS_GHM2 in two key ways. First, they are more sensitive to the risk-free rate: the coefficients on the risk-free rate are significant at the 10% and 5% levels for TS_DNNp and TS_LSTMp, respectively, while the coefficient for the benchmark model TS_GHM2 is not significant. Second, our models are more responsive to market return volatility; the coefficients on the std_mktrf variable are significant at the 1% level for both TS_DNNp and TS_LSTMp, whereas the coefficient for TS_GHM2 is only significant at the 5% level.

Overall, the discussion above suggests that our models are more responsive to changes in stock and market return volatility, enabling them to achieve stronger performance compared to the benchmark model.

Table 4.5 The Time Series Linear Regression Model Result

This table presents the results from time series linear regression models of 4.4.5.2 for trading rules of TS_DNNp, TS_LSTMp, and TS_GHM2. Trading rules TS_DNNp, TS_LSTMp, and TS_GHM2 refer to trading strategies based on the DNNp, LSTMp, and GHM2 prediction models, respectively. In the table, the dependent variable, $logreturn_t$, represents the average log return for all stocks traded in month t. The independent variables—rf_t, $mktrf_t$, std_mktrf_t, and $creditSpread_t$—represent the monthly risk-free rate, market excess return, standard deviation of daily market returns, and monthly credit spread for month t across all traded stocks, respectively. The period varies from January 1990 to December 2023, with a total of 408 months.

	TS_GHM2	TS_DNNp	TS_LSTMp
Dep. Variable	Logret	Logret	Logret
Indep. Variables:			
rf	0.4206	0.7095*	0.8659**
	0.683	1.695	2.043
mktrf	-0.2771***	-0.1797***	-0.1805***
	-10.22	-9.75	-9.671
std_mktrf	0.5189**	0.4805***	0.6307***
	2.168	3.216	5.827
creditSpread	-0.0029	0.0006	0.0018
	-0.749	0.223	0.667
R-squared	0.278	0.291	0.318
Number of observations: 408			

4.5 Conclusions

In this research, we build on the methodology of Goulding, Harvey, and Mazzoleni (2023) (abbreviated as GHM) to enhance time series momentum (TSMOM) performance with deep learning predictors. GHM's approach uses a combination of long (12-month) and short (2-month) look-back period excess returns to identify potential turning points in stock prices. When TSMOM signals differ between these periods for a stock in month t, it indicates a turning point in month t+1, potentially interrupting or reversing the current trend. GHM

advises that investors rebalance their positions or shift to risk-free assets at these turning points. This turning point prediction method by GHM, referred to as GHM2, is replaced in our study with deep learning (denoted as DL) techniques, aiming to improve the accuracy of turning point predictions and, consequently, further enhance TSMOM trading rule performance. Specifically, we choose the traditional DL models, such as Deep Neural Network (DNN) and the Long Short-Term Memory (LSTM), to forecast the turning point and denote these predictors as DNNp and LSTMp, respectively. The DL predictor is trained on time-series data.

We use a confusion matrix to compare the predictive performance of our turning point predictors, DNNp and LSTMp, against the benchmark GHM2 over the full test period from January 1990 to December 2023. Both DNNp and LSTMp significantly increase the number of true positives (TP: actual turning points), from 477,373 in GHM2 to 729,788 and 786,662, respectively, while only slightly reducing true negatives (TN: non-turning points), from 1,065,013 to 946,312 for DNNp and 975,213 for LSTM. Across all four metrics—accuracy, precision, recall, and F1 score—DNNp and LSTMp outperform GHM2. For instance, DNNp and LSTMp achieve accuracy levels of 0.5626 and 0.5914, respectively, considerably higher than GHM2's 0.5177, with LSTMp demonstrating the highest predictive performance among the models.

The annual prediction performance for these models aligns with the overall results. From 1990 to 2023, DNNp and LSTMp consistently record higher TP (true positive) and lower FN (false negative) counts compared to GHM2, while their TN (true negative) and FP (false positive) counts are only slightly lower and higher, respectively. Each year, DNNp and

LSTMp outperform the benchmark across all four metrics, with LSTMp consistently leading. Notably, in 2008, DNNp and LSTMp reach their peak accuracy, at 0.6069 and 0.6612, respectively, showing a marked improvement over GHM2's 0.5311.

We integrate turning point prediction models—DNNp, LSTMp, and GHM2—into the classic TSMOM strategy, using a 12-month look-back window to invest in the U.S. stock market, covering 24,767 stocks from January 1990 to December 2023. The TS_DNNp and TS_LSTMp trading rules achieve significantly higher monthly mean log returns and Sharpe ratios, along with lower return standard deviations, than the benchmark strategy TS_GHM2. These gains are both economically and statistically significant. For instance, TS_DNNp and TS_LSTMp yield average monthly mean log returns of 0.0148 and 0.0177, respectively, compared to 0.0108 for TS_GHM2. Annual results show similar patterns over the test period, with TS_DNNp and TS_LSTMp consistently outperforming TS_GHM2.

When accounting for transaction costs, the profitability patterns remain consistent with those observed without costs. TS_DNNp and TS_LSTMp continue to outperform the benchmark TS_GHM2 in terms of both average mean log returns and average Sharpe ratios. All three TSMOM rules also surpass the buy-and-hold (BH) strategy, with TS_LSTMp showing the highest performance overall. Specifically, TS_DNNp and TS_LSTMp achieve average mean log returns of 0.0135 and 0.0163, respectively, significantly above TS_GHM2's 0.0099 and BH’s -0.0140.

We also examine the factors contributing to the superior profitability of our trading models by employing cross-sectional and time series linear regression models. The regression results reveal that TS_DNNp and TS_LSTMp are more responsive to changes in stock and

market return volatility than TS_GHM2, which supports their stronger performance relative to the benchmark model.

In summary, this study contributes to the time series momentum (TSMOM) research by integrating deep learning (DL) techniques to enhance strategy performance, building on the approach in Goulding, Harvey, and Mazzoleni (2023) (abbreviated as GHM). By accurately predicting turning points, traders can optimize long and short positions more effectively than relying on GHM's look-back window interactions. Our confusion matrix analysis from 1990 to 2023 demonstrates a notable increase in true positives and reduction in false negatives, with DL models significantly outperforming GHM2 (the prediction of the turning point in GHM) across Accuracy, Precision, Recall, and F1-score. Applying DL models to TSMOM trading rules results in higher monthly mean returns with lower volatility, yielding a superior Sharpe ratio. Additionally, cross-sectional and time series regressions show that DL models respond better to stock and market return volatility than GHM2, reinforcing their effectiveness. Our enhanced TSMOM trading rules offer valuable insights for investors in developing practical trading strategies for real-world investment applications.

This study has certain limitations. Our best DL model (LSTMp) achieves only 0.5914 accuracy in predicting turning points, which may pose challenges for ensuring reliable profitability in practical investments. Future research should focus on two key areas to enhance the model's predictive performance: improving input quality and refining DL model architectures. Stock price movements are influenced by multiple factors beyond historical prices, including non-price trading data (e.g., trading volume), company fundamentals, industry conditions, macroeconomic indicators, and investor behavior. Identifying and

incorporating more relevant variables could improve prediction accuracy. Additionally, advancing DL models and optimizing their architectures remain essential. By significantly increasing model accuracy, DL-based TSMOM trading strategies could become more effective and applicable in real-world investments.

Chapter Five

Conclusion

In this chapter, the thesis concludes the application of deep learning (DL) techniques in three research topics by outlining the key discoveries and their significance within each of the three essays and discussing the potential economic implications for investors and professionals.

5.1 Essay One

In Essay One, we incorporate deep learning (DL) techniques into the Hub Strategy (HS), a chart analysis approach, to create a novel trading strategy, HSDL. Our DL prediction models utilize features directly derived from HS chart patterns, along with additional data such as trading volume, price volatility, and sentiment. We test these trading rules on S&P 500 stocks, comparing their performance against benchmarks, including the pure HS strategy and buy-and-hold (BH). The empirical results demonstrate that the HSDL strategy consistently outperforms these benchmarks.

The HSDL rules deliver substantial annualized average returns on S&P 500 stocks, ranging from 23.26% to 25.75%, significantly outperforming the buy-and-hold (BH) rule (9.64%). Moreover, HSDL demonstrates improvements over the corresponding pure HS strategies. When transaction costs are considered, the HSDL strategies remain robust, showing lower sensitivity to such costs. An analysis of the 10 worst market days reveals that the HSDL strategies are not particularly vulnerable to market crashes.

To address data mining concerns, we employ a false discovery rate (FDR) controlling approach, which confirms that the HSDL strategy is superior to the pure HS strategy and outperforms the BH rule on 80.7% of the sample stocks. Additionally, we utilize a recent asset pricing model with 153 risk factors and a two-stage regression, applying Lasso in the first stage and OLS in the second. Results show the HSDL strategy outperforms HS and BH, with significantly positive alphas on most stocks.

The main contribution of this study is the introduction of a novel chart analysis method, HSDS, which integrates the Hub Strategy (HS) with a deep learning (DL) predictor. Our

findings demonstrate that the structural elements of HS chart patterns can be effectively utilized by DL models to achieve high-accuracy predictions of price direction. Additionally, our HSDS trading strategies exhibit strong and consistent profitability. Furthermore, our direct chart pattern recognition approach can be applied to identify other traditional chart patterns, offering valuable insights for chart pattern analysis. This integration of DL techniques with traditional chart patterns enhances their predictive power and profitability, advancing both academic research and practical trading applications.

We recognize several limitations in our study. First, the limited availability of HS chart patterns restricts the learning efficiency of deep learning (DL) models. Second, our analysis is confined to a single timeframe—daily data—potentially overlooking valuable insights from other time horizons. Third, we employ relatively basic DL models, leaving room for further optimization. To enhance future research, we suggest expanding the dataset by incorporating more stocks and markets, which would improve model training and predictive performance. Additionally, integrating multiple timeframes, such as daily, weekly, and 30-minute HS chart patterns, could refine trade entry and exit strategies. Lastly, developing more advanced DL architectures may further boost prediction accuracy and enhance the effectiveness of HSDS trading strategies.

5.2 Essay Two

Essay Two builds on the research framework proposed by Bessembinder, Cooper, and Zhang (2019) (hereafter Bessembinder) and utilize deep learning (DL) techniques to improve the detection of abnormal stock returns following significant event disclosures, replacing

conventional linear regression. This study introduces an innovative approach to directly estimate the abnormal return (AR) using trained DL models.

We compare the ability of DL models to test abnormal returns against three benchmarks, including Bessembinder, Lyno (Lyon, Barber, and Tsai (1999)), and FF5F (Fama and French (2015)). DL models significantly outperform in detecting ARs within -3% to 3% using random datasets and maintain consistent performance across non-random datasets categorized by firm size and market conditions. This consistency highlights DL's superior uniformity compared to Bessembinder. Additionally, DL models produce well-specified test statistics in both random and non-random samples, reducing misspecifications in cases like Book-to-Market Ratio and Industry Clustering. Regression analysis reveals DL's lower sensitivity to firm size, market conditions, and periods, enhancing its robustness.

This study contributes to the literature by introducing a deep learning (DL)-based approach for directly estimating abnormal returns (AR), demonstrating improved accuracy and adaptability over previous advanced methods. Additionally, we highlight the potential benefits of developing more sophisticated AR estimators, which would offer researchers a more precise and efficient tool for abnormal return calculations in case studies.

Our research is still in its early stages. Future work should incorporate additional factors influencing stock returns, including industry conditions, macroeconomic indicators, and financial market dynamics. Expanding and organizing large datasets across different financial markets could further enhance the performance of the deep learning-based abnormal return (AR) estimator. Additionally, exploring more advanced DL architectures remains essential for improving the accuracy and robustness of AR estimation.

5.3 Essay Three

In this research, we build on the methodology of Goulding, Harvey, and Mazzoleni (2023) (abbreviated as GHM) and replace their look-back period predictor with deep learning (DL) predictor, aiming to improve the accuracy of turning point predictions and, consequently, further enhance the time series momentum (TSMOM) trading rule performance.

We employ a confusion matrix to evaluate the predictive performance of our turning point predictors, DNNp and LSTMp, in comparison to the benchmark GHM2 from GHM across all stocks in the three major U.S. stock markets during the test period from January 1990 to December 2023. Both DNNp and LSTMp significantly increase the number of true positives (TP: actual turning points), while only slightly reducing true negatives (TN: non-turning points), Across all four metrics—accuracy, precision, recall, and F1 score—DNNp and LSTMp outperform GHM2. The annual prediction performance for these models aligns with the overall results.

We incorporate turning point prediction models—DNNp, LSTMp, and GHM2—into the classic TSMOM strategy, utilizing a 12-month look-back window to invest in the U.S. stock market. The TS_DNNp and TS_LSTMp trading rules deliver significantly higher monthly mean log returns and Sharpe ratios, coupled with lower return standard deviations, compared to the benchmark strategy TS_GHM2. These improvements are both economically and statistically meaningful.

In addition, we examine the factors contributing to the superior profitability of our trading models by employing cross-sectional and time series linear regression models. The regression results reveal that TS_DNNp and TS_LSTMp are more responsive to changes in

stock and market return volatility than TS_GHM2, which supports their stronger performance relative to the benchmark model.

This study contributes to the literature by leveraging deep learning (DL) techniques to predict turning points, enhancing both prediction accuracy and profitability compared to the advanced time series momentum strategy proposed by Goulding, Harvey, and Mazzoleni (2023). Our improved time series momentum (TSMOM) trading rules provide valuable guidance for investors in formulating practical trading strategies for real-world investment applications.

This study has limitations, as our best DL model (LSTMp) achieves only 0.5914 accuracy in predicting turning points, posing challenges for reliable profitability. Future research should focus on enhancing input quality and refining DL architectures. Stock price movements depend on various factors beyond historical prices, such as trading volume, company fundamentals, industry trends, macroeconomics, and investor behavior. Incorporating more relevant variables could improve accuracy. Additionally, advancing DL models and optimizing architectures are crucial. Increasing model accuracy would make DL-based TSMOM trading strategies more effective for real-world investments.

5.4 Overall Discussion

This thesis expands the application of Deep Learning (DL) in finance by exploring three key areas: novel chart analysis (Hub Strategy, HS), abnormal return (AR) estimation, and time series momentum strategy (TSMOM). Our findings demonstrate that DL-based trading rules significantly outperform benchmarks in profitability, while the DL-based AR

estimator achieves greater accuracy than advanced alternatives, highlighting DL's potential in these domains. This research holds value for both investors and researchers. With further advancements, DL-based strategies related to HS and TSMOM could offer investors reliable, substantial profits and be adapted into automated trading systems. Additionally, a refined and publicly available DL-based AR estimator would enhance efficiency for researchers conducting case studies.

The primary limitation of this thesis is the need for continued improvement in the prediction accuracy of deep learning (DL) models. All three research areas—Hub Strategy (HS), abnormal return (AR) estimation, and time series momentum (TSMOM)—involve forecasting stock price direction or returns. While our DL-based models outperform benchmarks, enhancing their predictive accuracy remains essential. Future research should focus on two key approaches: model-oriented and feature-oriented improvements. The model-oriented approach involves optimizing model selection and integrating multiple models for better accuracy, while the feature-oriented approach emphasizes identifying effective input variables and constructing high-quality datasets. For DL-based trading models, incorporating additional variables beyond past prices—such as company fundamentals, industry conditions, macroeconomic indicators, market trends, and trader behaviour—could enhance performance. Similarly, integrating these factors into the DL-based AR estimator would improve its precision. By refining both model architectures and input features, we can further strengthen our DL predictors, enhancing their practical value in trading strategies and AR estimation.

References

Abe, M., & Nakayama, H. (2018). Deep learning for forecasting stock returns in the cross-section. In *Pacific-Asia conference on knowledge discovery and data mining*. 273-284. Springer, Cham.

Andriyanto, A. (2020). Sectoral stock prediction using Convolutional Neural Networks with candlestick patterns as input images. *International Journal of Emerging Trends in Engineering Research, 8*(6), 2249–2252.

Arévalo, R., García, J., Guijarro, F., & Peris, A. (2017). A dynamic trading rule based on filtered flag pattern recognition for stock market price forecasting. *Expert Systems with Applications*, *81*, 177–192.

Asness, C. S., Moskowitz, T. J., & Pedersen, L. H. (2013). Value and momentum everywhere. *The journal of finance, 68*(3), 929-985.

Athey, S., & Imbens, G. W. (2019). Machine learning methods that economists should know about. *Annual Review of Economics*, *11*, 685-725.

Azevedo, V., Hoegner, C., & Velikov, M. (2023). The Expected Returns on Machine-Learning Strategies. Available at SSRN.

Babu, A., Levine, A., Ooi, Y. H., Pedersen, L. H., & Stamelos, E. (2018). Trends everywhere. *Journal of Investment Management,* Forthcoming, NYU Stern School of Business.

Balas, V. E., Roy, S. S., Sharma, D., & Samui, P. (Eds.). (2019). *Handbook of deep learning applications (Vol. 136).* New York: Springer.

Barber, B. M., & Lyon, J. D. (1997). Detecting long-run abnormal stock returns: The empirical power and specification of test statistics. *Journal of financial economics, 43*(3), 341-372.

Benjamini, Y., & Hochberg, Y. (1995). Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the Royal statistical society*: series B (Methodological), *57*(1), 289-300.

Bessembinder, H., & Zhang, F. (2013). Firm characteristics and long-run stock returns after corporate events. *Journal of Financial Economics*, *109*(1), 83-102.

Bessembinder, H., Cooper, M. J., & Zhang, F. (2019). Characteristic-based benchmark returns and corporate events. *The Review of Financial Studies*, *32*(1), 75-125.

Biao Li, 2008, the website of publications relevant Hub Strategy. https://www.google.com/search?sxsrf=AOaemvJJfTCvadwTPmNdHZVMjka8h-o_CQ:1641266246255&q=%E7%BC%A0%E4%B8%AD%E8%AF%B4%E7%A6%85&tbm=isch&chips=q:%E7%BC%A0+%E4%B8%AD+%E8%AF%B4+%E7%A6%85,online_chips:%E5%9B%BE%E4%B9%A6:VsxR8C1V6tw%3D&usg=AI4_-kRDg-4IMhjiQBDzMbS7ss7aXMxgAA&sa=X&ved=2ahUKEwixwNuckZf1AhVaSGwGHV40BF0QgIoDKA

F6BAgnEBQ&biw=1463&bih=759&dpr=2.63

Billah, M. M., Sultana, A., Bhuiyan, F., & Kaosar, M. G. (2024). Stock price prediction: comparison of different moving average techniques using deep learning model. *Neural Computing and Applications, 36*(11), 5861-5871.

Boainain PG, Pereira PLV (2009) Head and shoulder: testing the profitability of graphic pattern of technical analysis for the brazilian stock exchange. Munich Personal RePEc Archive Paper (in Portuguese) (15653).

Boehme, D. R. and S. M. Sorescu (2002). The Long-Run Performance Following Dividend Initiations and Resumptions: Underreaction or Product of Chance? *Journal of Finance*, *57*, 871-900.

Botezatu, A. P., Burlacu, A., & Orhei, C. (2024). A Review of Deep Learning Advancements in Road Analysis for Autonomous Driving. *Applied Sciences, 14*(11), 4705.

Brav, A. (2000). Inference in long‐horizon event studies: a Bayesian approach with application to initial public offerings. *The Journal of Finance*, *55*(5), 1979-2016.

Brock, W., Lakonishok, J., LeBaron, B. (1992). Simple Technical Trading Rules and the Stochastic Properties of Stock Returns. *Journal of Finance 47*, 1731– 64.

Brown, S. and J. Warner (1980). Measuring Security Price Performance, *Journal of Financial Economics, 8*, 205-258.

Brownlee, J. (2017). *Long short-term memory networks with python: develop sequence prediction models with deep learning*. Machine Learning Mastery.

Bulkowski TN (2011) Encyclopedia of Chart Patterns, vol 225. *John Wiley & Sons*

Campbell, C. and C. Wasley (1993). Measuring Security Price Performance Using Daily NASDAQ Returns, *Journal of Financial Economics, 33*, 73-92.

Carhart, M. M. (1997). On persistence in mutual fund performance. *The Journal of Finance*, *52*(1), 57–82.

Cervelló-Royo, R., Guijarro, F., & Michniuk, K. (2015). Stock market trading rule based on pattern recognition and technical analysis: Forecasting the DJIA index with intraday data. *Expert systems with Applications*, *42*(14), 5963-5975.

Chai, J., & Li, A. (2019, July). Deep learning in natural language processing: A state-of-the-art survey. In *2019 International Conference on Machine Learning and Cybernetics (ICMLC)* (pp. 1-6). IEEE.

Chai, J., & Ngai, E. W. (2020). Decision-making techniques in supplier selection: Recent accomplishments and what lies ahead. *Expert Systems with Applications*, 140, 112903.

Chakrabarti, G. (2015). Time-series momentum trading strategies in the global stock market. *Business Economics, 50*, 80-90.

Chang, P. K., & Osler, C. L. (1999). Methodical madness: Technical analysis and the irrationality of exchange‐rate forecasts. The Economic Journal, *109*(458), 636-661.

Chava, S., & Reguly, A. (2024). The Long-Run Stock Market Performance of Mergers and Acquisitions. Available at SSRN 4742401.

Chen, K., Zhou, Y., & Dai, F. (2015, October). A LSTM-based method for stock returns prediction: A case study of China stock market. In *2015 IEEE international conference on big data (big data)* (pp. 2823-2824). IEEE.

Chen, J., Wu, W., & Tindall, M. L. (2016). *Hedge fund return prediction and fund selection: A machine-learning approach* (No. 16-4). Federal Reserve Bank of Dallas.

Chen, L., Pelger, M., & Zhu, J. (2024). Deep learning in asset pricing. *Management Science*, *70*(2), 714-750.

Chen, Y., He, K., & Tso, G. K. (2017). Forecasting crude oil prices: a deep learning-based model. *Procedia computer science*, 122, 300-307.

Choi, S., & Renelle, T. (2019, July). Deep learning price momentum in US equities. In *2019 International Joint Conference on Neural Networks (IJCNN)* (pp. 1-8). IEEE.

Chowdhury, M. S., Nabi, N., Rana, M. N. U., Shaima, M., Esa, H., Mitra, A., ... & Naznin, R. (2024). Deep Learning Models for Stock Market Forecasting: A Comprehensive Comparative Analysis. *Journal of Business and Management Studies, 6*(2), 95-99.

Cochrane, J. H. (2011). Presidential address: Discount rates. *The Journal of finance*, *66*(4), 1047-1108.

Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., & Kuksa, P. (2011). Natural language processing (almost) from scratch. *Journal of Machine Learning Research*, 12, 2493–2537.

Cooper, M. J., Gutierrez Jr, R. C., & Hameed, A. (2004). Market states and momentum. *The journal of Finance, 59*(3), 1345-1365.

Goulet Coulombe, P., Leroux, M., Stevanovic, D., & Surprenant, S. (2022). How is machine learning useful for macroeconomic forecasting? *Journal of Applied Econometrics*, *37*(5), 920-964.

Daniel, K., & Titman, S. (1997). Evidence on the characteristics of cross-sectional variation in stock returns. *the Journal of Finance*, *52*(1), 1-33.

Daniel, K., Jagannathan, R., & Kim, S. (2012). *Tail risk in momentum strategy returns* (No. w18169). National Bureau of Economic Research.

Daniel, K., & Moskowitz, T. J. (2016). Momentum crashes. *Journal of Financial economics*, *122*(2), 221-247.

Das, S. R., Mokashi, K., & Culkin, R. (2018). Are markets truly efficient? Experiments using deep learning algorithms for market movement prediction. *Algorithms*, *11*(9), 138.

Deng, Y., Bao, F., Kong, Y., Ren, Z., & Dai, Q. (2016). Deep direct reinforcement learning for financial signal representation and trading. *IEEE transactions on neural networks and learning systems, 28*(3), 653-664.

Dezsi, E., & Nistor, I. A. (2016). Can deep machine learning outsmart the market? a

comparison between econometric modelling and long-short term memory. *Romanian Economic and Business Review.*

Ding, X., Zhang, Y., Liu, T., & Duan, J. (2015, June). Deep learning for event-driven stock prediction. In *Twenty-fourth international joint conference on artificial intelligence*.

D'Souza, I., Srichanachaichok, V., Wang, G. J., & Yao, C. Y. (2016). The enduring effect of time-series momentum on stock returns over nearly 100-years. In *Asian Finance Association (AsianFA) 2016 Conference*.

Dutta, A. (2015). Improved calendar time approach for measuring long-run anomalies. *Cogent Economics & Finance, 3*(1), 1065948.

Edwards, R. D., Magee, J., & Bassetti, W. C. (2018). *Technical analysis of stock trends*. CRC press.

Eckbo, B. E., R. W. Masulis and Ø. Norli (2000). Seasoned Public Offerings: Resolution of the 'New Issues Puzzle'*, Journal of Financial Economics, 56*, 251-291.

Fama, E. F., & French, K. R. (2015). A five-factor asset pricing model. *Journal of financial economics*, *116*(1), 1-22.

Fama, E. (1998). Market Efficiency, Long-term Returns, and Behavioural Finance, *Journal of Financial Economics*, *49*, 283-306.

Fama, E. F., & French, K. R. (1996). Multifactor explanations of asset pricing anomalies. *The journal of finance*, 51(1), 55-84.

Fama, E. (1991). Efficient Capital Markets: II, *Journal of Finance*, *46*, 1575-1617.

Fama, E. F., & French, K. R. (1993). Common risk factors in the returns on stocks and bonds. *Journal of Financial Economics*, *33*(1), 3–56

Fan, J., Xue, L., & Yao, J., 2017. Sufficient forecasting using factor models. *Journal of econometrics*, *201*(2), 292-306.

Fama, E. F., & French, K. R. (2015). A five-factor asset pricing model. *Journal of financial economics*, *116*(1), 1-22.

Fernández-Delgado, M., Cernadas, E., Barro, S., & Amorim, D. (2014). Do we need hundreds of classifiers to solve real world classification problems? *The journal of machine learning research*, *15*(1), 3133-3181.

Feng, G., Giglio, S., & Xiu, D. (2020). Taming the factor zoo: A test of new factors. *The Journal of Finance*, *75*(3), 1327-1370.

Feng, G., He, J., Polson, N. G., & Xu, J. (2024). Deep learning in characteristics-sorted factor models. *Journal of Financial and Quantitative Analysis, 59*(7), 3001-3036.

Fischer, T. and Krauss, C., 2018. Deep learning with long short-term memory networks for financial market predictions. *European journal of operational research*, *270*(2), pp.654-669.

Ghoshal, S., & Roberts, S. (2020). Thresholded ConvNet ensembles: neural networks for technical forecasting. *Neural Computing and Applications*, *32*(18), 15249-15262.

Geertsema, P., & Lu, H. (2023). Relative Valuation with Machine Learning. *Journal of Accounting Research*, *61*(1), 329-376.

Georgopoulou, A., & Wang, J. (2017). The trend is your friend: Time-series momentum strategies across equity and commodity markets. *Review of Finance, 21*(4), 1557-1592.

Giglio, S., Liao, Y., & Xiu, D. (2021). Thousands of alpha tests. *The Review of Financial Studies*, *34*(7), 3456-3496.

Glabadanidis, P. (2014). The market timing power of moving averages: evidence from US REITs and REIT indexes. *International Review of Finance, 14*(2), 161-202.

Glabadanidis, P. (2015). Market timing with moving averages. *International Review of Finance, 15*(3), 387-425.

Glorot, X., Bordes, A., & Bengio, Y. (2011, June). Deep sparse rectifier neural networks. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics* (pp. 315-323). JMLR Workshop and Conference Proceedings.

Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep learning. MIT press.

Goulding, C. L., Harvey, C. R., & Mazzoleni, M. G. (2023). Momentum turning points. *Journal of Financial Economics, 149*(3), 378-406.

Goulding, C. L., Harvey, C. R., & Mazzoleni, M. G. (2024). Breaking Bad Trends. *Financial Analysts Journal, 80*(1), 84-98.

Goulet Coulombe, P., Leroux, M., Stevanovic, D. and Surprenant, S. (2022). How is machine learning useful for macroeconomic forecasting? *Journal of Applied Econometrics, 37*(5), pp.920-964.

Gu, S., Kelly, B., & Xiu, D. (2020). Empirical asset pricing via machine learning. *The Review of Financial Studies, 33*(5), 2223-2273.

Gao, L., Han, Y., Li, S. Z., & Zhou, G. (2018). Market intraday momentum. *Journal of Financial Economics, 129*(2), 394-414.

Guo, Y., Liu, Y., Oerlemans, A., Lao, S., Wu, S., & Lew, M. S. (2016). Deep learning for visual understanding: *A review. Neurocomputing, 187*, 27–48.

Gudelek, M. U., Boluk, S. A., & Ozbayoglu, A. M. (2017, November). A deep learning-based stock trading model with 2-D CNN trend detection. In *2017 IEEE symposium series on computational intelligence (SSCI)* (pp. 1-8). IEEE.

Hambly, B., Xu, R., & Yang, H. (2023). Recent advances in reinforcement learning in finance. *Mathematical Finance, 33*(3), 437-503.

Hanauer, M. X., & Windmüller, S. (2023). Enhanced momentum strategies. *Journal of Banking & Finance, 148,* 106712.

Harvey, C. R., Liu, Y., & Zhu, H. (2016). … and the cross-section of expected returns. *The Review of Financial Studies, 29*(1), 5-68.

Haugen, R. A., & Baker, N. L. (1996). Commonality in the determinants of expected stock returns. *Journal of financial economics, 41*(3), 401-439.

He, X. Z., & Li, K. (2015). Profitability of time series momentum. *Journal of Banking & Finance, 53*, 140-157.

He, X. Z., Li, K., & Li, Y. (2018). Asset allocation with time series momentum and reversal. *Journal of Economic Dynamics and Control, 91,* 441-457.

He, Z., Kelly, B., & Manela, A. (2017). Intermediary asset pricing: New evidence from many asset classes. *Journal of Financial Economics, 126*(1), 1-35.

Heaton, J. B., Polson, N. G., & Witte, J. H. (2016). Deep learning in finance. arXiv preprint arXiv:1602.06561.

Hiransha, M. E. A. G., Gopalakrishnan, E. A., Menon, V. K., & Soman, K. P. (2018). NSE stock market prediction using deep-learning models. *Procedia computer science, 132,* 1351-1362.

Hochreiter, S. (1997). Long Short-term Memory. *Neural Computation MIT-Press*.

Hou, K., Xue, C., & Zhang, L. (2015). Digesting anomalies: An investment approach. *The Review of Financial Studies, 28*(3), 650-705.

Hou, K., Xue, C., & Zhang, L. (2020). Replicating anomalies. *The Review of Financial Studies, 33*(5), 2019-2133.

Huang, D., Li, J., Wang, L., & Zhou, G. (2020). Time series momentum: Is it there*? Journal of financial economics, 135*(3), 774-794.

Hunt, J. O., Myers, J. N., & Myers, L. A. (2022). Improving earnings predictions and abnormal returns with machine learning. Accounting Horizons, 36(1), 131-149.

Hurst, B., Ooi, Y., Pedersen, L., (2017). A century of evidence on trend-following investing. *Journal of Portfolio Management 44*, 15–29.

Huynh, H. D., Dang, L. M., & Duong, D. (2017, December). A new model for stock price movements prediction using deep neural network. In *Proceedings of the 8th International Symposium on Information and Communication Technology* (pp. 57-62).

Jacobson, R., & Mizik, N. (2009). The financial markets and customer satisfaction: re-examining the value implications of customer satisfaction from the efficient market's perspective. *Marketing Science, 28*(5), 810–819.

Jearanaitanakij, K., & Passaya, B. (2019). Predicting Short Trend of Stocks by Using Convolutional Neural Network and Candlestick Patterns. *2019 4th International Conference on Information Technology* (InCIT2019).

Jegadeesh, N., & Titman, S. (1993). Returns to buying winners and selling losers: Implications for stock market efficiency. *The Journal of finance, 48*(1), 65-91.

Jensen, T. I., Kelly, B., & Pedersen, L. H. (2023). Is there a replication crisis in finance? *The Journal of Finance, 78*(5), 2465-2518.

Jiang, J., Kelly, B., & Xiu, D. (2023). (Re‐) Imag (in) ing price trends. *The Journal of Finance,* 78(6), 3193-3249.

Jones, C. M. (2002). A century of stock market liquidity and trading costs. *Available at SSRN*

313681.

Jurgovsky, J., Granitzer, M., Ziegler, K., Calabretto, S., Portier, P. E., Guelton, L. H., & Caelen, O. (2018). Sequence classification for credit-card fraud detection. *Expert Systems with Applications*, *100*, 234–245.

Karki, D., & Bikram Khadka, P. (2024). Momentum Investment Strategies across Time and Trends: A Review and Preview. *Nepal Journal of Multidisciplinary Research, 7*(1), 62-83.

Kim, A. Y., Tse, Y., & Wald, J. K. (2016). Time series momentum and volatility scaling. *Journal of Financial Markets, 30,* 103-124.

Kim, M., & McAlister, L. M. (2011). Stock market reaction to unexpected growth in marketing expenditure: Negative for sales force, contingent on spending level for advertising. *Journal of Marketing, 75*(4), 68-85.

Kingma, D. P. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.*6980.

Kolari, J. W. and S. Pynnönen (2011). Nonparametric Rank Tests for Event Studies. *Journal of Empirical Finance, 18,* 953-971.

Kothari, S. P., & Warner, J. B. (2007). Econometrics of event studies. In *Handbook of empirical corporate finance* (pp. 3-36). Elsevier.

Kothari, S. P., & Warner, J. B. (1997). Measuring long-horizon security price performance. *Journal of financial economics, 43*(3), 301-339.

Kuan, C. M., & Liu, T. (1995). Forecasting exchange rates using feedforward and recurrent neural networks. *Journal of applied econometrics, 10*(4), 347-364.

Kusuma, R. M. I., Ho, T.-T., Kao, W.-C., Ou, Y.-Y., & Hua, K.-L. (2019). Using Deep Learning Neural Networks and Candlestick Chart Representation to Predict Stock Market. http://arxiv.org/abs/1903.12258

Lasheras, F. S., de Cos Juez, F. J., Sánchez, A. S., Krzemień, A., & Fernández, P. R. (2015). Forecasting the COMEX copper spot price by means of neural networks and ARIMA models. *Resources Policy, 45,* 37-43.

Leigh, W., Paz, N., & Purvis, R. (2002). Market timing: a test of a charting heuristic. *Economics Letters, 77*(1), 55-63.

Liang, Q., Rong, W., Zhang, J., Liu, J., & Xiong, Z. (2017, May). Restricted Boltzmann machine-based stock market trend prediction. In *2017 International Joint Conference on Neural Networks (IJCNN)* (pp. 1380-1387). IEEE.

Li, B. (2023). Spillover Effects of Abnormal Stock Returns and Fluctuations in NASDAQ-Listed Industry Leaders: A Machine Learning Approach. Available at SSRN 4618676.

Li, K., & Liu, J. (2022). Optimal dynamic momentum strategies. *Operations Research, 70*(4), 2054-2068.

Li, Z., Sakkas, A., & Urquhart, A. (2022). Intraday time series momentum: Global evidence and links to market characteristics. *Journal of Financial Markets, 57,* 100619.

Liu, G., & Wang, X. (2019). A new metric for individual stock trend prediction. *Engineering Applications of Artificial Intelligence, 82*, 1-12.

Liu, Z., Lu, S., & Wang, S. (2021). Asymmetry, tail risk and time series momentum. *International Review of Financial Analysis, 78,* 101938.

Lo, A. W., & Hasanhodzic, J. (2010). The heretics of finance: *Conversations with leading practitioners of technical analysis (Vol. 16).* John Wiley and Sons.

Lo, A. W., Mamaysky, H., & Wang, J. (2000). Foundations of technical analysis: Computational algorithms, statistical inference, and empirical implementation. *The journal of finance, 55*(4), 1705-1765.

Loughran, T., & Ritter, J. R. (2000). Uniformly least powerful tests of market efficiency. *Journal of Financial Economics, 55*(3), 361–389.

Lynch, A. W., & Balduzzi, P. (2000). Predictability and transaction costs: The impact on rebalancing rules and behavior. *The Journal of Finance, 55*(5), 2285-2309.

Lyon, J., B. Barber and C. Tsai (1999). Improved Methods of Tests of Long horizon Abnormal Stock Returns, *Journal of Finance, 54*, 165-201.

Maqsood, H., Mehmood, I., Maqsood, M., Yasir, M., Afzal, S., Aadil, F., ... & Muhammad, K. (2020). A local and global event sentiment based efficient stock exchange forecasting using deep learning. *International Journal of Information Management, 50,* 432-451.

Marshall, B. R., Young, M. R., & Rose, L. C. (2006). Candlestick technical trading strategies: Can they create value for investors? *Journal of Banking & Finance, 30*(8), 2303-2323.

Marshall, B. R., Nguyen, N. H., & Visaltanachoti, N. (2017). Time series momentum and moving average trading rules. *Quantitative Finance, 17*(3), 405-421.

Matsubara, T., Akita, R., & Uehara, K. (2018). Stock price prediction by deep neural generative model of news articles. *IEICE TRANSACTIONS on Information and Systems*, *101*(4), 901-908.

Messmer, M., & Audrino, F. (2022). The lasso and the factor zoo-predicting expected returns in the cross-section. *Forecasting, 4*(4), 969-1003.

McLean, R. D., & Pontiff, J. (2016). Does academic research destroy stock return predictability? *The Journal of Finance, 71*(1), 5-32.

Mitchell, M. and E. Stafford (2000). Managerial Decisions and Long-term Stock price Performance, *Journal of Business, 73,* 287-329.

Moskowitz, T. J., Ooi, Y. H., & Pedersen, L. H. (2012). Time series momentum. *Journal of financial economics, 104*(2), 228-250.

Nazareth, N., & Reddy, Y. V. R. (2023). Financial applications of machine learning: A literature review. *Expert Systems with Applications, 219*, 119640.

Nazário, R. T. F., e Silva, J. L., Sobreiro, V. A., & Kimura, H. (2017). A literature review of

technical analysis on stock markets. *The Quarterly Review of Economics and Finance, 66*, 115-126.

Nelson, D. M., Pereira, A. C., & De Oliveira, R. A. (2017, May). Stock market's price movement prediction with LSTM neural networks. In *2017 International joint conference on neural networks (IJCNN)* (pp. 1419-1426). IEEE.

Norasaed, W., & Siriborvornratanakul, T. (2024). Market movement prediction using chart patterns and attention mechanism. *Discover Analytics, 2*(1), 1.

Olorunnimbe, K., & Viktor, H. (2023). Deep learning in the stock market—a systematic survey of practice, backtesting, and applications. *Artificial Intelligence Review, 56*(3), 2057-2109.

Osler CL, Chang PH (1995) Head and shoulders: not just a flaky pattern. *FRB of New York Staff Report* (4).

Ong, J., & Herremans, D. (2023). Constructing time-series momentum portfolios with deep multi-task learning. *Expert Systems with Applications, 230,* 120587.

Pang, X., Zhou, Y., Wang, P., Lin, W., & Chang, V. (2020). An innovative neural network approach for stock market prediction. *The Journal of Supercomputing, 76*, 2098-2118.

Partheepan, S., Sanati, F., & Hassan, J. (2023). Autonomous unmanned aerial vehicles in bushfire management: Challenges and opportunities. *Drones, 7*(1), 47.

Patel, J., Shah, S., Thakkar, P., & Kotecha, K. (2015). Predicting stock and stock price index movement using trend deterministic data preparation and machine learning techniques. *Expert systems with applications, 42*(1), 259-268.

Patil, P., Wu, C. S. M., Potika, K., & Orang, M. (2020, January). Stock market prediction using ensemble of graph theory, machine learning and deep learning models. In *Proceedings of the 3rd international conference on software engineering and information management* (pp. 85-92).

Di Persio, L., & Honchar, O. (2017). Recurrent neural networks approach to the financial forecast of Google assets. *International journal of Mathematics and Computers in simulation, 11,* 7-13.

Pring, M. J. (2002). Study guide for technical analysis explained (Vol. 5). New York: McGraw-Hill.

Qin, Y., Pan, G., & Bai, M. (2020). Improving market timing of time series momentum in the Chinese stock market. *Applied Economics, 52*(43), 4711-4725.

Ritter, J. R. (1991). The long‐run performance of initial public offerings. *The journal of finance, 46*(1), 3-27.

Rumelhart, D.E., Hinton, G.E. and Williams, R.J. (1988), Learning Internal Representations by Error Propagation, Institute for Cognitive Science, University of California, San Diego.

Saad, E. W., Prokhorov, D. V., & Wunsch, D. C. (1998). Comparative study of stock trend

prediction using time delay, recurrent and probabilistic neural networks. *IEEE Transactions on neural networks, 9*(6), 1456-1470.

Sant'Anna, L. R., & Vancin, D. F., (2015). Can Chart Patterns Predict Price Movements? A Monte Carlo Analysis in the Brazilian Stock Market. In *XV Encontro Brasileiro de Finanças*.

Savin, G., Weller, P., & Zvingelis, J. (2007). The predictive power of "head-and-shoulders" price patterns in the US stock market. *Journal of Financial Econometrics, 5*(2), 243-265.

Schwager, J. D. (2012). *Market wizards, updated: Interviews with top traders*. John Wiley & Sons.

Selvin, S., Vinayakumar, R., Gopalakrishnan, E. A., Menon, V. K., & Soman, K. P. (2017, September). Stock price prediction using LSTM, RNN and CNN-sliding window model. In *2017 international conference on advances in computing, communications and informatics (icacci)* (pp. 1643-1647). IEEE.

Sezer, O. B., Ozbayoglu, M., & Dogdu, E. (2017). A deep neural-network based stock trading system based on evolutionary optimized technical analysis parameters. *Procedia computer science, 114,* 473-480.

Sezer, O. B., & Ozbayoglu, A. M. (2018). Algorithmic financial trading with deep convolutional neural networks: Time series to image conversion approach. *Applied Soft Computing, 70,* 525-538.

Sezer, O. B., & Ozbayoglu, A. M. (2019). Financial trading model with stock bar chart image time series with deep convolutional neural networks. arXiv preprint arXiv:1903.04610.

Sezer, O. B., Gudelek, M. U., & Ozbayoglu, A. M. (2020). Financial time series forecasting with deep learning: A systematic literature review: 2005–2019. *Applied soft computing, 90*, 106181.

Sloan, R. G. (1996). Do stock prices fully reflect information in accruals and cash flows about future earnings? *Accounting review,* 289-315.

Shen, F., Chao, J., & Zhao, J. (2015). Forecasting exchange rate using deep belief networks and conjugate gradient method. *Neurocomputing, 167*, 243-253.

Shen, J., & Shafiq, M. O. (2020). Short-term stock market price trend prediction using a comprehensive deep learning system. *Journal of big Data, 7*, 1-33.

Shi, H. L., & Zhou, W. X. (2017). Time series momentum and contrarian effects in the Chinese stock market. *Physica A: Statistical Mechanics and its Applications, 483,* 309-318.

Smidt, S. (1965). A test of the serial independence price changes in soybean futures. *Food Research Institute Studies,* 5(2), 117-136.

Sonkavde, G., Dharrao, D. S., Bongale, A. M., Deokate, S. T., Doreswamy, D., & Bhat, S. K. (2023). Forecasting stock market prices using machine learning and deep learning models: A systematic review, performance analysis and discussion of implications. *International Journal of Financial Studies, 11*(3), 94.

Su, C. (2021). A comprehensive investigation into style momentum strategies in China. *Financial Markets and Portfolio Management, 35*(1), 101-144.

Thammakesorn, S., & Sornil, O. (2019). Generating Trading Strategies Based on Candlestick Chart Pattern Characteristics. *Journal of Physics: Conference Series, 1195*(1).

Tsinaslanidis, P., & Guijarro, F. (2021). What makes trading strategies based on chart pattern recognition profitable? *Expert Systems, 38*(5), e12596.

Wang, Q., & Sun, X. (1996). Enhanced artificial neural network model for Chinese economic forecasting. In *Proceedings of the international conference on management science and the economic development of China* (Vol. 1, pp. 30-6).

Xia, Y., Liu, S., Yu, Q., Deng, L., Zhang, Y., Su, H., & Zheng, K. (2024, May). Parameterized Decision-Making with Multi-Modality Perception for Autonomous Driving. In *2024 IEEE 40th International Conference on Data Engineering (ICDE)* (pp. 4463-4476). IEEE.

Yang, B., Gong, Z. J., & Yang, W. (2017, July). Stock market index prediction using deep neural network ensemble. In *2017 36th Chinese control conference (ccc)* (pp. 3882-3887). IEEE.

Yan, Y. H., Zhao, P., & Zhou, Z. H. (2023). Universal online learning with gradient variations: A multi-layer online ensemble approach. *Advances in Neural Information Processing Systems, 36*, 37682-37715.

Zakamulin, V. (2014). The real-life performance of market timing with moving average and time-series momentum rules. *Journal of Asset Management, 15*, 261-278.

Zakamulin, V., & Giner, J. (2022). Time series momentum in the US stock market: Empirical evidence and theoretical analysis. *International Review of Financial Analysis, 82*, 102173.

Zhang, J., Zhai, J., & Wang, H. (2021). A survey on deep learning in financial markets. In *Proceedings of the First International Forum on Financial Mathematics and Financial Technology* (pp. 35-57). Springer Singapore.

Zhang, R., Shen, F., & Zhao, J. (2014, July). A model with fuzzy granulation and deep belief networks for exchange rate forecasting. In *2014 international joint conference on neural networks (IJCNN)* (pp. 366-373). IEEE.

Zhang, S. (2022). Dissecting currency momentum. *Journal of Financial Economics,* 144(1), 154-173.

Zhang, S., Yang, Y., Chen, C., Zhang, X., Leng, Q., & Zhao, X. (2024). Deep learning-based multimodal emotion recognition from audio, visual, and text modalities: A systematic review of recent advancements and future prospects. *Expert Systems with Applications, 237*, 121692.

Zhang, X., & Tan, Y. (2018). Deep stock ranker: A LSTM neural network model for stock selection. In *Data Mining and Big Data: Third International Conference, DMBD 2018, Shanghai, China, June 17–22, 2018, Proceedings 3* (pp. 614-623). Springer International

Publishing.

Zheng, H., Wu, J., Song, R., Guo, L., & Xu, Z. (2024). Predicting financial enterprise stocks and economic data trends using machine learning time series analysis. *Applied and Computational Engineering, 87*, 26-32.

Zhou, B. (2019). Deep learning and the cross-section of stock returns: Neural networks combining price and fundamental information. *Available at SSRN 3179281*.

Zhou, F., Zhou, H. M., Yang, Z., & Yang, L. (2019). EMD2FNN: A strategy combining empirical mode decomposition and factorization machine based neural network for stock market trend prediction. *Expert Systems with Applications, 115*, 136-151.

Appendices

A.1 Appendix to "Chart Analysis with Deep Learning"

We attempt to utilize the deep learning (DL) techniques to supplementally predict the future price directions, locating Hub Strategy (HS) trading points, and expect that this way can enhance the profitability of pure HS.

A.1.1 Deep Learning Model Design and Training

We totally construct 6 deep learning models, which correspond to three buy rules and three sell rules, respectively. Specifically, we denote three buy DL models as DL_Re_B, DL_FF_B, and DL_SF_B, and likewise, three sell models are DL_Re_S, DL_FF_S, and DL_SF_S. We extract input features (see Table in 3.2.2 Inputs and outputs of DMLPs) relative to HS chart patterns and the output from the in-sample data. The label for output is either one or zero, which means that the price of the 3-step ahead is above or below the current price. The observation numbers for these models are displayed as follows.

- DL_Re_B: 5380; DL_FF_B: 3704; DL_SF_B:5491.
- DL_Re_S: 7743; DL_FF_S: 4221; DL_SF_S:3099.

Observations for each model are split into the train and validation groups, and the train size and validation test size are 80% and 20% respectively. We feed features into DL models and conduct the supervised training by adjusting hyperparameters multiple. The prediction accuracy for these models is exhibited as follows.

- DL_Re_B: 70.80%; DL_FF_B: 69.13%; DL_SF_B:68.65%.
- DL_Re_S: 62.45%; DL_FF_S: 67.29%; DL_SF_S:65.84%.

For brevity, we only depict the training processes of the loss function and accuracy of prediction for the second following buy (DL_SF_B) and sell (DL_SF_S) rule models, as the representative of others (see Figure A.1.1).

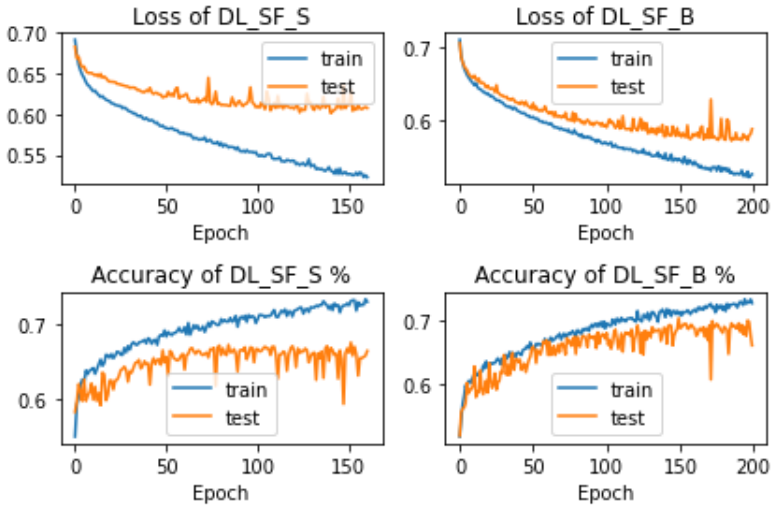

Figure A.1.1 Training Process of DL_SF_B and DL_SF_S models

A.1.2 Performance of DL Models on the Out-Sample Data

Evaluating the effectiveness of deep learning (DL) models involves conducting an out-of-sample test. We collect the Hub Strategy (HS) chart patterns from out-of-sample data involving all 505 individual stocks to examine the accuracy and precision of our DL models.

Our prediction models target a binary outcome equal to one for a positive return over the line segment horizon of 3-step ahead and zero otherwise. Two scenarios arise in our prediction models. Buy prediction models (DL_Re_B, DL_FF_B, and DL_SF_B) focus on precisely forecasting positive returns, while sell predictionaries (DL_Re_S, DL_FF_S, and

DL_SF_S) aim to select negative returns. Therefore, the relevant concepts are explained as follows.

True positive (TP): For buy models, prediction is a positive return and the reality is the 'upward trend'; and for sell models, prediction is a negative return and the reality is the 'downward trend'. We desire that.

True negative (TN): For buy models, prediction is a negative return, and the reality is the 'downward trend'; and for sell models, prediction is a positive return and the reality is the 'upward trend'. We want that too.

False positive (FP): For buy models, prediction is a positive return, and the reality is the 'downward trend'; and for sell models, prediction is a negative return, and the reality is the 'upward trend'. We will lose money if we trust these predictions, which is the worst.

False negative (FN): For buy models, prediction is a negative return, and the reality is the 'upward trend'; and for sell models, prediction is a positive return, and the reality is the 'downward trend'. We will just lose the opportunity rather than money if we trust these predictions. False alarm.

Accuracy = (TP+TN)/(TP+FP+FN+TN), which displays how many down or upward trends models correctly label out of all trends.

Precision = TP/(TP+FP), and it answers how many of those that buy models labeled as upward trends are actually "up" and how many of those that sell models labeled as downward trends are actually 'down'. Precision measure dominates the profitability of our DL models.

Table A.1.1 displays the accuracy and precision of DL models. The buy models,

DL_Re_B, DL_FF_B, and DL_SF_B, can achieve accuracy of 66.42%, 66.10%, and 68.51%, while the prediction accuracy for the sell models, DL_Re_S, DL_FF_S, and DL_SF_S, ranges from 58.05% to 65.54%. A more promising thing is that our models can hit higher precision. All buy models have a precision of above 70%, and the highest precision, 77.5%, appears in the model of DL_Re_B. The measures of precision for sales models are higher than the models' accuracy outcomes. These models with high precision ensure our trading strategies can reach high profits.

Table A.1.1 Performance of DL Models

Table A.1.1 presents the measure of performance, accuracy, and precision, for DL models over 505 individual stocks in out-of-sample. The sample period ranges from 03/01/2007 to 31/12/2020.

DL model	Observations	Accuracy %	Precision %
DL_Re_B	5218	66.42	77.54
DL_FF_B	3985	66.1	71.11
DL_SF_B	5991	68.51	75.89
DL_Re_S	4527	58.05	59.32
DL_FF_S	2933	65.54	69.77
DL_SF_S	4104	64.57	68.75

A.2 Appendix to "Long Run Event Study with Deep Learning"

A.2.1 Definition of the C14 Firm Characteristics

We measure these characteristics following Bessembinder, Cooper, and Zhang (2019). All variables are created using data from the CRSP stock price files and the Compustat annual data on the platform of Wharton Research Data Services (WRDS). Accounting data are assumed to be available 4 months after the fiscal year-end. Table A.2.1 presents the

definitions for the C14 firm characteristics. We gather stock data from the CRSP monthly file and the Compustat annual file, accessible through the Wharton Research Data Services (WRDS) platform. Our dataset spans from January 1970 to December 2021. Summary statistics for firm characteristics, measured monthly, are provided in Table A.2.1.

A.2.2 Methods of Benchmarks

A.2.2.1 Bessembinder

Bessembinder refers to the method proposed by Bessembinder, Cooper, and Zhang (2019). We follow this method to conduct two-stage method. In the first stage, we estimate the following cross-sectional regression for each month t:

$$R_{it} = \alpha_t + \beta_t * X_{i,t-1} + \varepsilon_{it}, \qquad \text{(A.2.1)}$$

where R_{it} is the stock i's realized log (or simple) return in month t, and $X_{i,t-1}$ is a vector of firm i's characteristics measured at the end of month t-1. Then we estimate predicted returns using rolling averages of the past 12-month slope coefficients, which are shown as the equation of A.2.2.

$$E[R_{it}|I_{t-1}] = \frac{1}{12}\sum_{s=t-12}^{t-1}\hat{\alpha}_s + \left(\frac{1}{12}\sum_{s=t-12}^{t-1}\hat{\beta}_s\right)X_{i,t-1}, \qquad \text{(A.2.2)}$$

where $\hat{\alpha}_s$ and $\hat{\beta}_s$ are the coefficients estimated from Equation (A.2.1) in month s. In the second stage, they run the regression shown in equation A.2.3, using all stocks.

Table A.2.1 Definition of the C14 Firm Characteristics

log size	Natural log of market capitalization, which is stock price (prc in CRSP monthly stock file) times number of shares outstanding (shrout), at the end of the prior month
log book-to-market ratio	Natural log of the book-to-market ratio at the end of the prior month. Book value is the firm's common equity (Compustat item ceq) in the latest annual report. Market value is the firm's market capitalization (prc times shrout) at the end of the prior month reported in CRSP
Momentum	Buy-and-hold stock returns over months (-12,-2) before the month of interest
ROA	Income before extraordinary items (ib) divided by average total assets (at) in the year
Asset growth	Natural log of the ratio of total assets (at) at the end of the year to total assets at the beginning of the year, following Cooper, Gulen, and Schill (2008)
Beta	Market beta estimated using monthly excess stock returns and market risk premiums over the preceding 60 months. We require a minimum of six data points for the accuracy of the estimation
Accrual	Change in working capital from the last year minus depreciation and amortization (dp), divided by average total assets (at) in the year, following Sloan (1996). Working capital equals current assets (act) minus cash and short-term investment (che) minus current liabilities (lct) plus debt in current liabilities (dlc) plus income taxes payable (txp). Missing act, che, lct, dlc, txp, and dp are replaced with zero
Dividend	Dividends per share over the prior 12 months divided by the price at the end of the prior month
log LR return	Natural log of buy-and-hold stock returns over months (-13,-36) before the month of interest
Idiosyncratic risk (Idio)	In each month, we compute the standard deviation of the residual daily stock returns in the Fama and French (1993) three factor regression, following Ang et al. (2006). Idiosyncratic risk is the average standard deviation over the prior 12 months
Illiquidity	The average daily ratio of absolute stock return to dollar trading volume during the prior 12 months, as defined by Amihud (2002)
Turnover	Average monthly turnover (shares traded divided by shares outstanding) during the prior 12 months
Leverage	Debt in current liabilities (dlc) plus long-term debt (dltt), divided by market capitalization (prc times shrout in CRSP) at the end of the last month. Missing dlc and dltt are replaced with zero
Sales/price	Sales (sale) divided by market capitalization (prc times shrout in CRSP) at the end of the last month

Table A.2.2 Summary Statistics of Firm Characteristics

This table reports summary statistics regarding the firm characteristics that we employ to predict stock returns. It includes all firm months from January 1970 to December 2021 in each month, firm characteristics are winsorized at the upper and lower 1% levels.

Variable	N	Mean	Std.dev.	5th pctl	25th pctl	Median	75th pctl	95th pctl
log Return	2748617	0.0141	0.1911	-0.2397	-0.0708	0.0000	0.0788	0.2827
log size	2748617	7.5678	2.1988	1.5536	3.5934	6.5032	8.1161	12.3557
log BM	2748617	-0.3995	1.0099	-1.8291	-0.9479	-0.4454	-0.0117	1.0907
Momentum	2748617	0.1322	0.4817	-0.8215	-0.1789	0.0689	0.3687	1.2961
ROA	2748617	0.0001	0.1985	-0.5764	0.0027	0.0674	0.941	0.1935
Asset growth	2748617	0.3228	0.3281	-0.1941	-0.0013	0.1017	0.2728	0.9635
Beta	2746658	1.0674	0.7993	-0.0463	0.5199	0.9782	1.4909	2.5703
Accrual	2748617	0.0044	0.1829	-0.2907	-0.0892	-0.0075	0.1157	0.3118
Dividend	2748595	0.0177	0.0267	0.0000	0.0000	0.0023	0.0276	0.0737
log LR return	2432115	0.08913	0.6733	-1.1647	-0.2377	0.1607	0.4858	1.0941
Idio. Risk	2748454	0.0334	0.0293	0.0198	0.0311	0.0409	0.0585	0.08134
Illiquidity	2536942	5.6485	19.9471	0.0003	0.0148	0.1525	2.5152	22.7366
Turnover	2537061	0.1178	0.1456	0.0063	0.0273	0.0667	0.1458	0.4098
Leverage	2748617	1.111	2.401	0.0000	0.0356	0.2532	0.78423.	3.9822
Sales/price	2741412	3.0781	5.6184	0.2211	1.0437	2.3466	5.3011	13.7639

$$AR_{it} = R_{it} - E\,[R_{it}|I_{t-1}] = a + b^{*}\, D_{it} + u_{it}\,, \qquad \text{(A.2.3)}$$

where AR_{it} is the abnormal return that equals R_{it} minus $E\,[R_{it}|I_{t-1}]$, and D_{it} is an indicator that equals one if firm i experienced a corporate event during a specified horizon before month t, and zero otherwise.

A.2.2.2 Lyon

Lyon refers to the method of buy-and-hold size/book-to-market portfolio, which is proposed by Lyon, Barber, and Tsai (1999). The key feature of their method is the careful construction of reference portfolios. We follow this method to construct 70

size/book-to-market reference portfolios as follows: i) We multiply the price per share by the shares outstanding as the firm size, in June of each year, for all firms. ii) We rank all NYSE firms based on firm size and then generate size decile portfolios. iii) We place AMEX and Nasdaq firms in the appropriate NYSE size decile based on the June market value of these firms. iv) We further divide the smallest size decile into quintiles based on the firm value in June of each year. v) Each size portfolio is further partitioned into five book-to-market quintiles in June of year t. The book-to-market ratio is the ratio of common equity (Compustat item ceq) at the end of the fiscal year that ends in the calendar year prior to the formation date to the market capitalization at the end of December of year t-1 reported in CRSP. vi) We compute the monthly buy-and-hold returns on each of these seventy portfolios.

The method of computing the buy-and hold return on a reference portfolio involves first compounding the returns on stocks constituting the portfolio and then summing across stocks:

$$R_{ps\tau}^{bh} = \sum_{i=1}^{n_s} \frac{[\prod_{t=s}^{s+\tau}(1+R_{it})]-1}{n_s}, \qquad (A.2.4)$$

where n_s is the number of stocks traded in month s, the beginning period for the return calculation, τ is the period of investment (in months), and R_{it} is the return on stock i in month t.

We calculate monthly or long-horizon buy-and-hold abnormal returns as:

$$AR_{i\tau} = R_{i\tau} - E(R_{i\tau}), \qquad \text{(A.2.5)}$$

where $AR_{i\tau}$ is the τ period buy-and-hold abnormal return for stock i, $R_{i\tau}$ is the τ period buy-and-hold return, and $E(R_{i\tau})$ is the buy-and-hold return of the corresponding size/book-to-market reference portfolio. Lastly, we then transfer these returns to log returns for experiments and annualize the returns over 3- and 5-year horizons.

A.2.2.3 FF5F

FF5F refers to the method using the Fama and French (2015) five-factor model to estimate the predicted return of the stock. For each stock in our population, we estimate its loadings of risk factors using stock returns in the trailing 60 months before month t, which is shown in the following regression (A.2.6).

$$R_{is} - R_{Fs} = a_i + b_i(R_{Ms} - R_{Fs}) + s_i SMB_s + h_i HML_s + r_i RMW_s + c_i CMA_s + e_{is}, \text{ s = t-1 to t-60,} \quad \text{(A.2.6)}$$

where R_{is} is the return on stock i for period s, R_{Fs} is the risk-free return, R_{Ms} is the return on the value-weight market portfolio, SMB_s is the return on a diversified portfolio of small stocks minus the return on a diversified portfolio of big stocks, HML_s is the difference between the returns on diversified portfolios of high and low B/M stocks, RMW_s is the difference between the returns on diversified portfolios of stocks with robust and weak profitability, CMA_s is the difference between the returns on diversified portfolios of the stocks of low and high investment firms, and e_{is} is a zero-mean residual. Here, b_i, s_i, h_i, r_i, and c_i are loadings. We use the 1-month Treasury-bill yield as the proxy for the risk-free interest rate. Our estimation procedure of risk factor loadings is similar to Bessembinder, Cooper, and

Zhang (2019) and Daniel and Titman (1997). To mitigate the influence of outliers, we implement a method known as winsorization every month. Specifically, within each month, we perform winsorization on the factor loadings obtained from data spanning the preceding 60 months, targeting the top and bottom 2.5 percentiles of the distribution across firms. It is important to emphasize that this winsorization process is conducted without incorporating any information from subsequent periods. We download the FF5F factors from Kenneth French's website. We then estimate the predicted stock return in month t as $R_{Fs}+.\widehat{b_i}(R_{Mt}-R_{Ft})+ \widehat{s_i}SMBt+\widehat{h_i}HML_t +.\widehat{r_i}RMW_t+\widehat{c_i}CMA_t.$