Master's thesis

NTNU
Norwegian University of Science and Technology
Faculty of Economics and Management
Dept. of Industrial Economics and Technology Management

Markus Ramstad Høge
Egil Tanem

Deciphering Momentum and Reversal Effects

An Interpretative Approach Using Temporal Fusion Transformers

Master's thesis in Industrial Economics and Technology Management
Supervisor: Einar Belsom
June 2023

NTNU
Norwegian University of
Science and Technology

Markus Ramstad Høge
Egil Tanem

Deciphering Momentum and Reversal Effects

An Interpretative Approach Using Temporal Fusion Transformers

Master's thesis in Industrial Economics and Technology Management
Supervisor: Einar Belsom
June 2023

Norwegian University of Science and Technology
Faculty of Economics and Management
Dept. of Industrial Economics and Technology Management

NTNU
Norwegian University of
Science and Technology

Preface

This Master of Science thesis, completed in the spring of 2023, marks the end of our academic journey in Industrial Economics and Technology Management at the Norwegian University of Science and Technology (NTNU). Throughout our five years of intensive studies, we have cultivated a unique interdisciplinary expertise, combining the realms of computer science and finance. We have harnessed this potent blend of knowledge to delve into the complex dynamics of the stock market. This work stands as a testament to the practical application of our academic learning, where technology and economics intersect, and we hope it provides valuable insights and contributes to the broader understanding of financial market behaviors.

We would like to thank our supervisor, Einar Belsom, for his invaluable input and mentorship during the course of this research project. His insightful feedback and support have been instrumental in shaping the development and quality of this thesis.

Markus Høge	Egil Tanem
Markus Ramstad Høge	Egil Tanem

Trondheim, June 2023

Abstract

We employ the Temporal Fusion Transformer (TFT), a complex deep learning model, to predict stock returns of the S&P 500 constituents from January 2000 through December 2022. The aim of our study is two-fold: Firstly, to demonstrate the potency of emerging machine learning models — specifically the TFT — in generating trading signals that lead to desirable financial outcomes; Secondly, to highlight the inherent interpretability of the TFT, which we showcase as a tool for deciphering financial market dynamics. By leveraging the TFT's interpretable components and conducting post-model analyses, we uncover influential explanatory variables, important timesteps, persistent seasonal patterns, and the market state's influence on dependencies between past and future returns. Informed by these findings, we devise a simple yet effective rule-based strategy, Adaptive Momentum (AMOM), that harnesses the insights derived from the TFT. This strategy dynamically adjusts its formation and holding periods, integrating both momentum and reversal approaches. Through our extensive analysis, the TFT proves its efficacy, achieving a statistically significant average monthly return of 2.66%. Moreover, AMOM demonstrates a remarkable performance with a statistically significant average monthly return of 1.62%, surpassing comparable traditional strategies like cross-sectional momentum and time series momentum, which only manage to generate insignificant average monthly returns fluctuating around zero.

Sammendrag

Vi anvender Temporal Fusion Transformer (TFT), en kompleks dyp læringsmodell, til å predikere avkastningen til aksjer på S&P 500-indeksen i perioden fra januar 2000 til desember 2022. Studiens formål er todelt: For det første ønsker vi å demonstrere effektiviteten til nye maskinlæringsmodeller, spesifikt TFT, i å generere handelssignaler som resulterer i gunstige økonomiske utfall. For det andre ønsker vi å framheve den iboende tolkbarheten til TFT som et verktøy for å forstå finansielle markedsdynamikker. Ved å utnytte de tolkbare komponentene i TFT og foreta analyser av modellens resultater, avdekker vi innflytelsesrike forklaringsvariabler, viktige tidssteg, vedvarende sesongmønstre og hvordan markedsforholdene påvirker avhengighetene mellom tidligere og fremtidige avkastninger. I lys av disse funnene konstruerer vi en enkel regelbasert strategi kalt Adaptive Momentum (AMOM), som utnytter innsikt utledet fra TFT. Denne strategien tilpasser hvor langt tilbake den ser for å velge posisjoner, og hvor lenge den holder disse. I tillegg innlemmer AMOM metodikk fra både momentum- og reverseringsstrategier. Gjennom vår omfattende analyse viser TFT seg å være svært effektiv, med en statistisk signifikant gjennomsnittlig månedlig avkastning på 2,66%. Videre demonstrerer AMOM imponerende resultater med en statistisk signifikant gjennomsnittlig månedlig avkastning på 1,62%, noe som overgår sammenlignbare tradisjonelle strategier som tversnitt-momentum og tidsserie-momentum, som kun klarer å generere ikke-signifikante gjennomsnittlige månedlige avkastninger som svinger rundt nullpunktet.

Contents

List of Figures

List of Tables

List of Abbreviations

ML	Machine Learning	1
AI	Artificial Intelligence	1
XAI	Explainable Artificial Intelligence	1
TFT	Temporal Fusion Transformer	2
DNN	Deep Neural Network	2
EMH	Efficient Market Hypothesis	2
CAPM	Capital Asset Pricing Model	5
DL	Deep Learning	6
FNN	Feed-forward Neural Network	6
CNN	Convolutional Neural Network	6
RNN	Recurrent Neural Network	6
LSTM	Long Short-Term Memory Network	6
GRU	Gated Residual Unit	6
NLP	Natural Language Processing	7
DSSM	Deep Space-State Model	7
MQRNN	Multi-horizon Quantile Recurrent Forecaster	7
DeepAR	Deep Autoregressive Model	7
LIME	Local Interpretable Model-Agnostic Explanations	7
SHAP	Shapley Additive exPlanations	8
MLP	Multi-Layer Perceptron	8
Seq2Seq	Sequence-to-Sequence	8
GRN	Gated Residual Network	12
VSN	Variable Selection Network	12
VIX	Volatility Index	17
TRBC	Thomson Reuters Business Classification	17
CSMOM	Cross-sectional Momentum	25
TSMOM	Time Series Momentum	25
CSREV	Cross-sectional Reversal	40
AMH	Adaptive Market Hypothesis	40
AMOM	Adaptive Momentum	40

Chapter 1

Introduction

Do rising stocks continue to rise over specific investment horizons, or is there a tendency for price reversal? If these patterns exist, past price performance's predictive power over future returns could be used to generate excess returns by anticipating future winner and loser stocks. Over the past decades, these phenomena, commonly referred to as momentum and reversal effects, have captivated the interest of academics and investors alike.

There exists a plethora of research highlighting how past performance can be utilized to generate excess returns, ranging from the traditional, straightforward approaches of Jegadeesh and Titman (1993) and Moskowitz, Ooi, et al. (2012) to the intricate, Machine Learning (ML)-driven approaches of Fischer and Krauss (2018) and Lim, Zohren, et al. (2019). However, the existing literature leaves certain gaps that merit further investigation. While traditional momentum strategies may offer inherent explainability due to their reliance on straightforward decision rules, modern momentum research faces a notable challenge in the use of Artificial Intelligence (AI) models, which, despite their advanced capabilities, often lack interpretability and explainability. The need for Explainable Artificial Intelligence (XAI) has become increasingly important, not only in a financial context, but for AI systems in general. The European Commission underscored the importance of this issue by introducing the Ethics Guidelines for Trustworthy AI in 2019. These guidelines outlined seven key requirements that AI must meet to be considered trustworthy, of which several can be linked to the topic of XAI (European Commission, 2019). Additionally, four federal agencies in the US have recently expressed concerns about potentially harmful uses of automated AI systems and vowed to vigorously enforce their collective authorities and to oversee the development and use of these systems (Consumer Financial Protection Bureau, 2023).

For institutional investors, the integration of interpretable models can bolster confidence in ML predictors within the financial sector. Such transparency also fosters trust with clients who seek to understand the rationale behind decisions made on their behalf. Furthermore, it is essential to not only emphasize regulatory demands and the value of trust in predictions for end users but also the confidence interpretability instills in the model builders. Interpretability may allow researchers and practitioners to evaluate the validity of the relationships the model captures and the factors it considers significant. This, in turn, provides the assurance that the model is functioning as intended and is able to identify meaningful patterns in the data. Adding to the aforementioned benefits, there are several other purposes of explainability for time series models, including reproducibility, robustness, stability, and interactivity (Rojat et al., 2021).

Modern momentum research has several caveats beyond the lack of interpretable approaches. Firstly, ML based momentum strategies typically rely on frequent (daily or intra-daily) rebalancing, which may lead to higher per-share transaction costs for retail investors, who may not be able to execute trades in larger volumes (see e.g. Fischer and Krauss, 2018; Ghosh et al., 2022; Lim, Zohren, et al., 2019). Secondly, there is still a paucity of research exploring momentum within a multi-horizon forecasting framework, despite various efforts to incorporate multi-horizon forecasting models in stock movement predictions (such as Hu, 2021; Zhang et al., 2022). Future predictions across the entire forecast path can help investors identify the most opportune times for entering, holding, and exiting individual positions, and have the potential to significantly enhance the effectiveness of momentum strategies. Finally, it should be noted that several models employed in time series and momentum research are inadequate for handling non-stationary data. As per Schmitt et al. (2013) and Wood, Roberts, et al. (2022), financial markets are prominent examples of noisy, highly non-stationary systems that can experience sudden changes in volatility, correlation length, mean-reversion length, or a combination thereof. Lim, Zohren, et al. (2019) suggest that future pathways of enhancing momentum strategies should incorporate methods to better deal with this non-stationarity.

To bridge the gaps in the existing momentum literature, we employ a Temporal Fusion Transformer (TFT) model — an attention-based Deep Neural Network (DNN) architecture designed for high-performance multi-horizon forecasting while enabling new forms of interpretability and enhanced robustness towards non-stationarity — using the past 12 months of data to forecast the returns of S&P 500 constituents over the next 6 months. To explore momentum in conjunction with company-specific features, we include exogenous time series of liquidity and size, along with static metadata such as industries and ticker symbols, as inputs in addition to past return data. Moreover, time-related inputs such as the day of the week, month, quarter, and time indices are incorporated to capture potential seasonal effects. We then leverage both the model's interpretable components and conduct post-model analyses of the predictions. This allows us to shed light on variable importance, complex temporal patterns, and significant events to provide a more refined understanding of the underlying drivers of momentum and reversal effects. Finally, a simple, rule-based trading strategy is formulated based on the derived insights and backtested to assess whether our inferred understanding can lay the foundation for profitable trading.

While previous research has already demonstrated the effectiveness of the TFT in stock price prediction tasks, our contribution stands out with several key enhancements. First, we incorporate specific company-related data, including liquidity, size, and industry, into the model. These features have been intentionally selected as they have been identified as significant factors in the momentum literature. This integration could potentially enhance our model's predictive capabilities and may offer a broader perspective on the potential predictive power these features possess. Second, we purposefully harness the interpretability capabilities of the TFT to compare how derived insights align with the Efficient Market Hypothesis (EMH) and the notion of momentum and reversals. Last, we go beyond generating trading signals directly from the model and use the derived insights to shape a new trading strategy, emphasizing the practical application and implementation of our findings.

The rest of this paper is organized as follows: Chapter 2 provides a comprehensive review of relevant literature. Then, an extensive description of the selected methodology and dataset is outlined in Chapter 3, before the results are presented and discussed in Chapter 4. At last, Chapter 5 concludes the study and proposes interesting pathways for further research.

Chapter 2

Related Work

This chapter aims to navigate the reader through the labyrinth of research related to stock return predictions, starting from the foundational EMH, which posits that it is impossible to predict future stock movements, to the classical momentum and reversal strategies that challenge the EMH. As we delve deeper, we explore the observed limitations of momentum and reversals, specifically their susceptibility to market crashes and their inherent riskiness. This prompts a pivot towards studies that integrate momentum with factor models in an attempt to understand how momentum correlates with various factors. Then, we trace the evolution of momentum research as it begins to incorporate additional features, leading us to the beginning of ML applications in this field. We explore how these advanced models have elevated our capabilities, enabling multi-horizon forecasting that was previously beyond reach. However, the black-box nature of these ML models presents a significant obstacle, sparking a need for explainable methods that provide insight into their decision-making processes. Finally, the chapter culminates by introducing a state-of-the-art model that marks the next milestone in momentum research. This model aims to overcome the hurdles outlined throughout the chapter, promising a comprehensive approach to predicting stock returns with a new level of interpretability.

2.1 Efficient Market Hypothesis

The Efficient Market Hypothesis (EMH) states that financial markets are efficient in processing and reflecting all available information about an asset, such as a stock (Fama, 1970). The weak form suggests today's stock prices reflect all the data of past prices and that no form of technical analysis can aid investors. The semi-strong form asserts that because public information is part of a stock's current price, investors cannot utilize either technical or fundamental analysis, though non-public information may still provide an advantage to investors. In its strong form, the EMH posits that it is impossible to consistently achieve above-average returns by predicting stock price movements using technical analysis, fundamental analysis, or any other method. According to the EMH, stock prices always fully reflect all publicly available information and any new information is rapidly incorporated into the stock price (Fama, 1991). Consequently, stock prices follow a random walk, and any future price changes are unpredictable (Malkiel, 1973). This implies that it is futile to attempt to outperform the market by exploiting perceived inefficiencies or price patterns.

2.2 Classical Momentum Strategies

Contrary to the EMH, an extensive body of financial literature documents momentum strategies generating abnormal returns across various asset classes, markets, and investment horizons. In *"The returns to buying Winners and Selling Losers: Implications for Stock Market Efficiency"* by Jegadeesh and Titman (1993), significant positive returns are generated over holding periods of 3 to 12 months by constructing zero-cost portfolios, going long past winners and short selling past losers in the U.S. stock market from 1965 through 1989. However, the momentum effect does not seem to persist, as a portion of the abnormal returns begins to dissipate when the portfolio is held for more than a year, due to long-term reversals. In their approach, they employ cross-sectional or relative strength momentum, a strategy where the stock selection is based on past relative performance. Rouwenhorst (1998) finds support for the presence of a cross-sectional momentum effect, applying a similar methodology in his study of 12 European stock markets and different asset classes. The momentum anomaly appears not to be limited to U.S. or European markets, as Rouwenhorst (1999) reports similar, though slightly weaker, results in a subsequent study of 20 emerging markets. Furthermore, Chan et al. (2000) provide statistically significant evidence of momentum profits in 23 international stock indices, of which 11 are European, nine Asian, two North American, and one South African.

While focusing on shorter horizons and contrarian strategies, the short-term reversal anomaly remains a crucial element in the momentum literature. Short-term reversal occurs when stocks with relatively low (high) returns over the past week or month experience positive (negative) abnormal returns in the subsequent week or month. In his study of the U.S. stock market, Jegadeesh (1990) observes highly significant and negative first-order serial correlation in monthly stock returns. Leveraging this systematic behavior of the returns, he estimates one-month-ahead forecasts and constructs 10 decile portfolios. The difference in returns between the top and bottom portfolios amounts to 2.49% per month. Lehmann (1990) further corroborates the short-term reversal effect by presenting evidence that past one-week winners and losers undergo significant return reversals in the following week. He demonstrates how these reversals yield persistent excess returns even after accounting for bid-ask spreads and plausible transaction costs.

The aforementioned momentum literature focuses on the relative performance of securities in the cross-section, finding that securities that recently outperformed their peers continue to do so on average, at least over the subsequent months. In *"Time series momentum"* by Moskowitz, Ooi, et al. (2012), an alternative strategy to cross-sectional momentum is introduced. Time series momentum, often referred to as trend-following or absolute momentum, focuses solely on a security's own absolute past performance rather than its relative performance to determine its future direction. The strategy essentially involves taking long (short) positions in assets with positive (negative) past returns over a specific lookback period. In their comprehensive study of futures and forwards contracts from Central European, Asian, and American markets, the authors find strong positive predictability from a security's own past returns and show that a diversified portfolio of time series momentum strategies across all asset classes delivers substantial abnormal returns with little exposure to standard asset pricing factors. Consistent with the cross-sectional momentum effect found by Jegadeesh and Titman (1993), the time series momentum effect persists for approximately a year before partially reversing over longer investment horizons.

Time series momentum is less dependent on the broader market context and can potentially generate profits in both rising and falling markets, offering a distinct advantage over cross-sectional momentum strategies that primarily rely on relative performance among assets. In their study comparing the performance of cross-sectional and time series momentum strategies across 24 markets, Bird et al. (2017) observe that both strategies generate statistically significant profits in most markets, with the latter strategy being superior. Time series momentum strategies can vary the number of stocks included in the winner and loser portfolios depending on the state of the market. For instance, in a strongly rising market, more assets are likely to have had positive past returns, leading to a larger long portfolio and possibly no short positions at all. Conversely, cross-sectional strategies select a fixed number of stocks in the winner and loser portfolios regardless of the market state. According to the authors, this flexibility allows time series momentum strategies to capture the momentum effect more efficiently and potentially generate higher returns than cross-sectional variants. However, time series strategies cannot be considered "zero-cost" strategies due to the dynamic sizing of the long and short portfolios, as opposed to cross-sectional strategies that generate equally sized long and short portfolios.

2.3 Risk and Market Conditions

Although cross-sectional and time series momentum strategies have been found to generate abnormal returns across various markets, the underlying causes of the momentum anomaly remain unclear and a topic of ongoing research. Factor models have been widely adopted in this context to assess whether momentum profits represent compensation for risk or an exploitable market anomaly. As an example of this, Carhart (1997) expands the Fama and French (1993) three-factor model, adding an additional factor capturing the one-year momentum anomaly of Jegadeesh and Titman (1993), finding that the momentum factor can explain a substantial amount of the excess return achieved by mutual funds. Furthermore, Bello (2008) compares the single-factor Capital Asset Pricing Model (CAPM), the Fama-French three-factor model, and the Carhart four-factor model, finding that the Carhart model is better at stock price prediction. Conversely, some studies show that a part of the momentum effect can be attributed to other factors such as trading volume, post-earnings announcement drift, and book-to-market ratio rather than past price movement in isolation (see e.g. Asness et al., 2013; C. Lee and Swaminathan, 2000; Sadka, 2006). Additionally, B. T. Kelly et al. (2021) find that the predictive content of momentum is mostly subsumed by conditional expected returns derived from time-varying factor exposures that depend on observable firm characteristics.

Several studies have been conducted to understand how momentum relates to market conditions. Cooper et al. (2004) find that the profitability of momentum strategies is significantly influenced by market conditions. This finding was later reaffirmed by Bird et al. (2017), who show that both cross-sectional and time-series strategies perform best in bullish markets. In contrast, these strategies, particularly the cross-sectional variation, are prone to severe return deterioration in bearish markets. Jegadeesh and Titman (1993) note that their cross-sectional strategy tends to select high- (low-) beta stocks following a market increase (decrease) and hence underperforms during market reversals. This underperformance of momentum strategies during market reversals was later corroborated by Asem and Tian (2010). Similar evidence of momentum strategies occasionally generating severe and persistently negative returns is shown by Daniel and Moskowitz (2016). They claim these crashes are somewhat predictable because they frequently arise from extremely volatile ”panic states” in which the market declines abruptly and then recovers with rising stock prices. Barroso and Santa-Clara (2015) note that although classical cross-sectional momentum investing occasionally results in severe crashes, as shown by the returns’ large negative skewness and excess kurtosis, the risk may be mitigated by scaling the portfolio to have constant volatility. Ruenzi and Weigert (2018) introduces crash sensitivity as an additional explanatory variable in the Fama and French (1993) three-factor model. Controlling for exposure to systematic crash risk renders the momentum strategy’s annualized returns insignificant. Thus, they conclude that momentum returns are not an exploitable market anomaly, but rather a risk premium for the strategy’s exposure to systematic crash risk.

2.4 Momentum and Company-specific Features

Momentum research has moved beyond historical price information as the sole measure of past performance and has studied momentum in conjunction with company-specific features. Pástor and Stambaugh (2003) investigate the relationship between liquidity and momentum profits, introducing the former as an explanatory variable in a multi-factor model. They show that stocks with high sensitivities to liquidity produce higher average returns and that half of the profits associated with a momentum strategy can be attributed to the liquidity risk factor. Furthermore, Connolly and Stivers (2003) find that substantial momentum effects arise from weeks with unexpectedly high turnover, whereas reversals follow in weeks with abnormally low turnover. In contrast to arbitrage intuition, Avramov et al. (2016) discover a negative momentum-illiquidity relation where momentum profits are higher (lower) in liquid (illiquid) market states. Their time series regression, which considers variables such as market illiquidity, the direction of market return, aggregated market volatility, and the three factors proposed by Fama and French (1993), show that market illiquidity emerges as the sole significant factor in explaining momentum profits.

Company market capitalization has also been found to explain excess returns, including those observed in momentum trading strategies. In a study examining the relationship between momentum and company size, Hong et al. (2000) show that the profitability of momentum strategies declines significantly with increasing firm size. The authors suggest that momentum arises from gradual information flow, and thus, smaller stocks for which information dissemination is slower should exhibit greater momentum effects.

Momentum has also been extensively researched in the context of industries. Moskowitz and Grinblatt (1999) find that traditional price momentum can be captured by industrial factors and show that an industrial momentum strategy going long securities from past winning industries and short securities from past losing industries is able to outperform traditional momentum strategies. Swinkels (2002), Su (2011), and Grobys and Kolari (2020) corroborate these findings and report significant industrial momentum effects in the European, Chinese, and American markets, respectively. In addition, numerous academic papers highlight the efficacy of industrial momentum strategies across several different asset classes, markets, and timeframes (see e.g Andreu et al., 2013; O'Neal, 2000; Szakmary and Zhou, 2015; Tan and Cheng, 2019).

2.5 Machine Learning

Machine Learning (ML) and Deep Learning (DL) models have become increasingly relevant in the context of financial time series forecasting and momentum trading as they have demonstrated several advantages over traditional statistical models. In their extensive comparison of various statistical, ML, and DL forecasting models, Makridakis et al. (2023) highlight several merits of deploying DL models in time series forecasting. First, ML and DL models can handle complex, high-dimensional data with improved accuracy. Second, in line with the research of Takeuchi and Y.-Y. A. Lee (2013) and Dixon et al. (2015), they are better at capturing nonlinear relationships and patterns in data, which are often present in financial time series. Third, ML and DL models can be trained on past data to adapt to changing market conditions. Finally, these models offer greater flexibility and can be used for a wide range of forecasting and trading tasks, including predicting asset prices, identifying trends, and constructing optimal portfolios. As an example, Lim, Zohren, et al. (2019) use an ML based approach that simultaneously learns both trend estimation and position sizing while optimizing the Sharpe ratio.

As ML and DL models undoubtedly exhibit clear benefits there is no surprise that a plethora of research on the financial applications of these methodologies has been conducted. Initially, ML for time series forecasting was primarily focused on one-step-ahead predictions. In the early days, simple Feed-forward Neural Networks (FNNs) were tested in order to predict future values (see e.g. De Oliveira et al., 2013; Niaki and Hoseinzade, 2013). Although initially designed to handle grid data, Convolutional Neural Networks (CNNs) have also been examined in a time series context. Gunduz et al. (2017) and Sezer and Ozbayoglu (2018) show that the kernel-based architecture of the CNN can be exploited in the modeling of sequential data. However, the fixed input size of CNNs limits their flexibility in modeling time series of varying lengths. Recurrent Neural Networks (RNNs) architectures such as the Long Short-Term Memory Network (LSTM) and Gated Residual Unit (GRU) are often preferred because they are specifically designed to handle sequential data. Their memory units enable them to process input sequences of variable lengths and to capture long-term dependencies. To exemplify, both Fischer and Krauss (2018) and Lim, Zohren, et al. (2019) display impressive results in terms of prediction accuracy and financial performance when deploying the LSTM in a momentum framework.

Although recurrent architectures such as LSTM and GRU have been established as ”state-of-the-art” in the modeling of sequential data, numerous efforts have since continued to push the boundaries of recurrent models. Due to their sequential processing, recurrent models may struggle to capture long-term dependencies when the input sequence length becomes too large. To overcome this, researchers have augmented these models with attention mechanisms that assign weights corresponding to the importance of the different parts of the time series according to the model. For instance, Chen and Ge (2019) and Kim and Kang (2019) show how their proposed attention-based LSTM models significantly enhance prediction performance in the Chinese and Korean stock markets, respectively.

Despite the success of utilizing attention mechanisms in combination with recurrent architectures, more recent advancements have focused on utilizing attention mechanisms directly to capture long-term dependencies. In *"Attention Is All You Need"* by Vaswani et al. (2017), the encoder-decoder-based Transformer architecture is unveiled for the first time. In contrast to RNNs, the Transformer avoids recurrence and instead relies entirely on attention mechanisms to draw a direct connection to all previous timesteps, enhancing its ability to model long-term dependencies and contextual information. Furthermore, it processes input sequences in parallel, reducing training time compared to sequential RNNs. Due to its inherent capabilities, this architecture has become the standard for many Natural Language Processing (NLP) models such as BERT and GPT, as explored in the work of Topal et al. (2021). Moreover, both Zhang et al. (2022) and Y. Li et al. (2022) have effectively used Transformer-based attention networks to predict stock movements, thereby showcasing the versatility and applicability of this architecture across diverse domains.

2.6 Multi-horizon Forecasting

The development of ML models has opened new avenues for multi-horizon forecasting within time series analysis. Rather than providing one-step-ahead predictions, multi-horizon forecasting provides future predictions at multiple future steps, allowing users to optimize their actions across the entire path. Similar to one-step-ahead predictors, most multi-horizon models such as Rangapuram et al.'s (2018) Deep Space-State Model (DSSM), Wen et al.'s (2017) Multi-horizon Quantile Recurrent Forecaster (MQRNN) and the Deep Autoregressive Model (DeepAR) by Salinas et al. (2020) primarily concentrate on incorporating different RNN structures, whereas recent advancements have incorporated attention-based methods (see Fan et al., 2019) and Transformers (see S. Li et al., 2019) to improve the selection of relevant past timesteps. Despite the increasing popularity of multi-horizon forecasting, there is a common oversight in these models that fail to account for the temporal difference in various types of inputs commonly involved in multi-horizon forecasting. Most existing models either assume knowledge of all exogenous inputs into the future (see e.g. S. Li et al., 2019; Rangapuram et al., 2018; Salinas et al., 2020) — a common problem with autoregressive models — or fail to consider the importance of static covariates, as these are concatenated with other time-dependent features at each step (see Wen et al., 2017). It is also worth noting that the autoregressive models adopt an iterative approach where predictions are fed back into the model recursively. This method introduces a key limitation: for long-range predictions, which are commonplace in classical momentum literature, errors tend to rack up over time. As a result, there is a need for direct methods that are capable of forecasting over multiple horizons without such compounding of errors.

2.7 Explainable AI

Most of the current multi-horizon forecasting architectures not only neglect the heterogeneity of common inputs but are also often perceived as black boxes because they rely on complex nonlinear interactions between multiple parameters, reducing end users' comprehension of the predictions. Siddiqui et al. (2019) claim that the interpretability of time series models is an especially challenging task due to the sequential and unintuitive nature of time series data: Computers and humans represent temporal data differently, mainly due to differences in their processing and storage mechanisms. Humans form an intuitive understanding of time, grounded in their personal experiences and contexts. In contrast, computers operate on mathematical and logical principles, manipulating data in a structured, algorithmic manner. This often leads to a gap between the computer's numerical representation of temporal data and the human's experiential, context-based understanding. To bridge this gap, researchers have been developing several pre-model, post-model, and intrinsic XAI techniques. Unfortunately, pre-model explainability methods such as correlation analysis may not be able to capture the complex temporal dependencies that are crucial for time series forecasting. Furthermore, popular post-model explainability methods such as Ribeiro et al.'s (2016) Local Interpretable Model-Agnostic Explanations (LIME) and the Shapley Additive exPlanations (SHAP) of Lundberg and S.-I. Lee (2017) do not consider the time ordering of input features. For instance, surrogate models are independently constructed for each data point in LIME, and SHAP inherently assumes independence between neighboring timesteps. As dependencies between timesteps are typically significant in time series, these approaches would lead to poor explanation quality. On the other hand, attention-based architectures with intrinsic interpretability can offer insights into relevant timesteps, but they are unable to differentiate the significance of different features at a given timestep. Therefore, novel methodologies are needed.

2.8 State-of-the-art

In Lim, Arik, et al. (2021), *"Temporal Fusion Transformers for interpretable multi-horizon time series forecasting"*, the TFT is introduced. The TFT is a novel attention-based architecture that combines high-performance direct multi-horizon forecasting with interpretable insights into temporal dynamics. It uses recurrent layers for local processing and interpretable self-attention layers for long-term dependencies, allowing it to capture temporal relationships at different scales. To enable high performance across various scenarios, the TFT incorporates specialized components to select relevant features and gating mechanisms to suppress unnecessary components. In their study, the authors are able to demonstrate considerable performance improvements over existing benchmarks on a variety of real-world datasets. They begin by evaluating the performance of the model on electricity and traffic datasets used in S. Li et al. (2019), Rangapuram et al. (2018) and Salinas et al. (2020). These datasets focus on simpler univariate time series containing only known inputs alongside the target variable. Next, they test the model on a retail dataset to benchmark its performance using a full range of complex inputs typically observed in multi-horizon forecasting. Finally, to assess the model's robustness against overfitting on smaller, noisy datasets, the authors apply the TFT to the financial task of volatility forecasting, using data from 31 international indices. When benchmarked against competing methods such as Ridge regression, Multi-Layer Perceptron (MLP), Sequence-to-Sequence (Seq2Seq), MQRNN, DeepAR, DSSM, and the Convolutional Transformer, the TFT shows a 3% to 26% performance improvement over the next best alternative across all experiments. Unsurprisingly, the model's demonstrated capabilities have led to its use in real-world applications, particularly in retail and logistics industries where it is employed to enhance forecasting accuracy and provide much-needed interpretability. Having these promising results in mind, it would be highly interesting to deploy the TFT in a multi-horizon momentum framework.

Chapter 3

Methodology

In this chapter, we present the methodology employed in our study, which aims to produce interpretable and accurate multi-horizon forecasts of stock returns using a TFT model. Our approach comprises several key steps that we will elaborate upon in the subsequent sections. First, we will introduce the architecture of the TFT, describing its layers and configurations to provide a comprehensive understanding of the model's foundation. Second, we will shed light on the techniques used to make our model more transparent and explainable. Third, we delve into the process of selecting, collecting, and pre-processing our data, ensuring that it is suitable for input into our TFT. Following the dataset discussion, we cover how the parameter optimization process is conducted. At last, we describe how the output of our model will be used in designing a trading strategy, and the metrics used to evaluate the strategy's performance.

3.1 Temporal Fusion Transformer

Multi-horizon forecasting is crucial in stock movement prediction as it caters to diverse investment objectives, enhances risk management, enables adaptability to changing market conditions, promotes portfolio diversification, and provides comprehensive market insights. However, generating accurate multi-horizon forecasts is a challenging task due to the inherent complexity and uncertainty associated with predicting dependent movements over various timeframes. To maximize the accuracy and reliability of these forecasts, it is vital to use all available information. In addition to temporal data on the variable of interest, a multi-horizon model should be able to incorporate information that may be known or unknown in the future. Moreover, it is crucial to account for static variables, which are time-invariant, as these can provide valuable context and contribute to a more robust and comprehensive forecast for each individual entity.

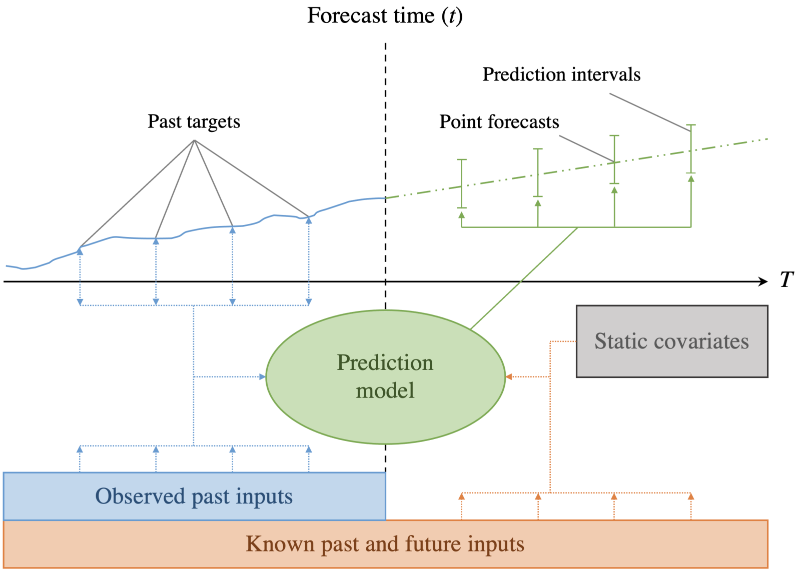

Figure 3.1 Multi-horizon forecasting with static covariates and various time-dependent inputs.

Mathematically we have I unique entities in a given time series dataset — for example, individual stocks. Each entity i is associated with a set of static covariates $\mathbf{s}_i \in \mathbb{R}^{m_\mathbf{s}}$ (e.g., the stock ticker symbol), as well as features $\boldsymbol{\chi}_{i,t} \in \mathbb{R}^{m_\boldsymbol{\chi}}$ and univariate targets $\mathbf{y}_{i,t} \in \mathbb{R}$ at each timestep $t \in [0, T_{\max}]$. Time-dependent input features are subdivided into two categories $\boldsymbol{\chi} = \left[\mathbf{z}_{i,t}^T, \mathbf{x}_{i,t}^T\right]^T$ — observed inputs $\mathbf{z}_{i,t} \in \mathbb{R}^{m_\mathbf{z}}$ which can only be measured at each step and are unknown beforehand, and known inputs $\mathbf{x}_{i,t} \in \mathbb{R}^{m_\mathbf{x}}$ which are predetermined (e.g., day-of-week at time t). The problem is illustrated in Figure 3.1.

The Temporal Fusion Transformer (TFT) uses components to efficiently build feature representations for each input type (i.e., static, known, and observed inputs) and attention mechanisms. Figure 3.2 shows the high-level architecture of TFT. The overall process of the model is as follows: For a given timestep t, a lookback window k, and a $\tau_{\max}$ step ahead window, the model takes as input past features $\boldsymbol{\chi}$ in the time period $[t-k, t]$, future known features $\mathbf{x}$ in the time period $[t+1, t+\tau_{\max}]$, and the static variables $\mathbf{s}$. The input is passed through the network and compared to the target variables $\mathbf{y}$ that spans the time window $[t+1, t+\tau_{\max}]$, updating the weights to learn the correct transformation from input to the output variables $\hat{\mathbf{y}}$. As the technical details of the TFT are rather involved, we provide only a brief description of the individual components in the subsequent subsections. For further details on the architecture and workings of Transformers and the TFT, see Lim, Arik, et al. (2021), Vaswani et al. (2017) and S. Li et al. (2019).

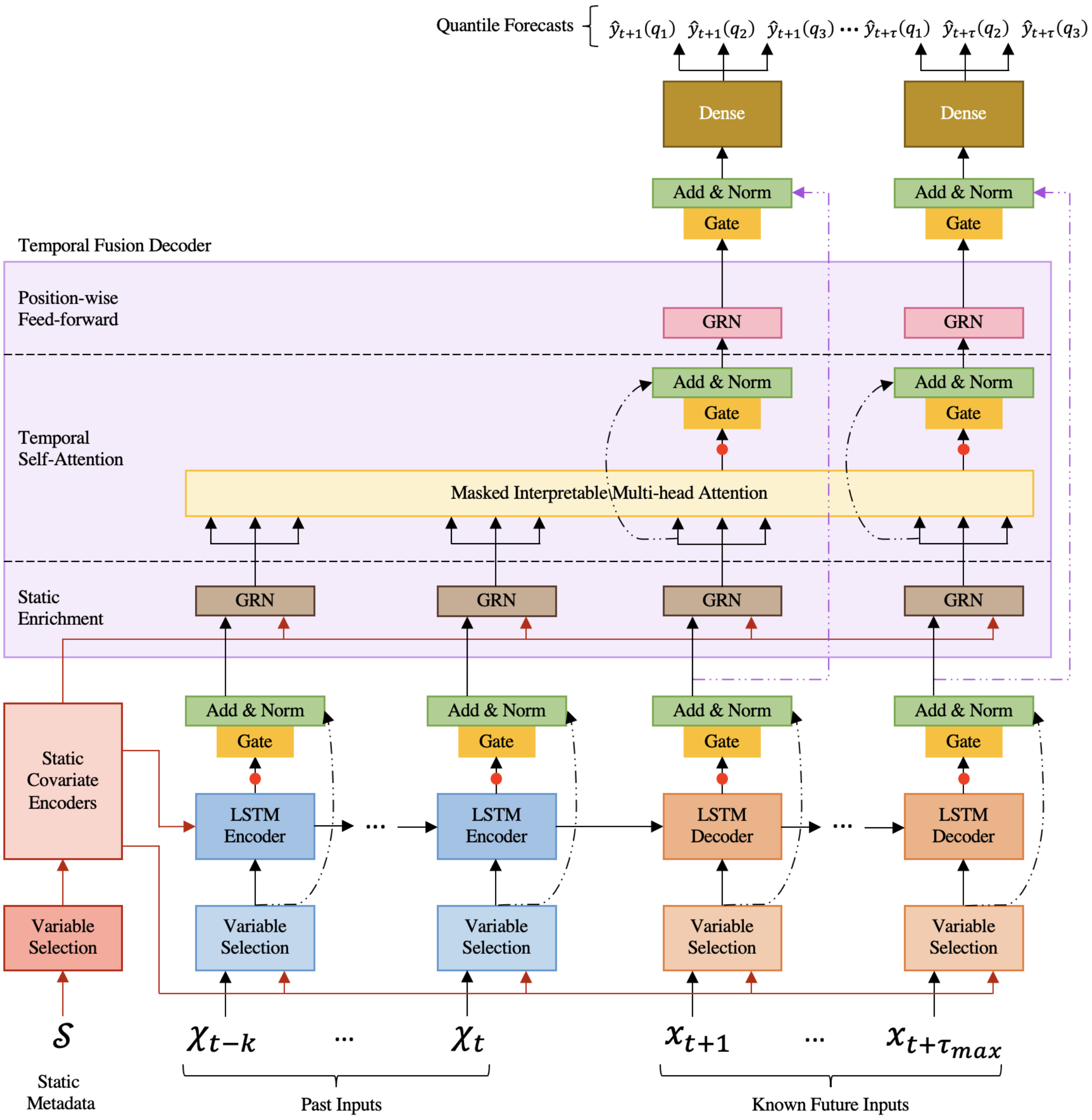

Figure 3.2 The Temporal Fusion Transformer (TFT) takes in three types of input data: static metadata, past time-varying inputs, and future time-varying inputs that are known beforehand. Variable Selection Networks (VSN) blocks are used to focus on the most important features from these inputs. The Gated Residual Network (GRN) blocks enhance the flow of information by incorporating skip connections and gating layers. Long Short-Term Memory Networks (LSTMs) process local patterns, while multi-head attention mechanisms integrate information across different timesteps. Matching colors denote weight sharing.

3.1.1 Gating Mechanisms

The exact relationship between inputs and target variables is often unknown in advance, making it challenging to identify relevant features. Additionally, determining the necessary degree of nonlinear processing can be difficult, and there could be situations where employing simpler models is advantageous, for example, when dealing with small or noisy data.

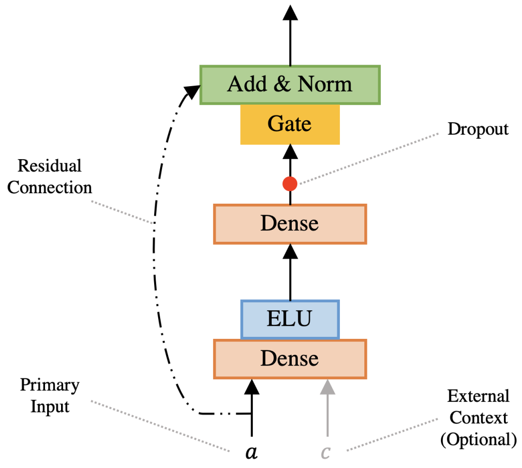

Figure 3.3 Gated Residual Network (GRN).

The Gated Residual Network (GRN) illustrated in Figure 3.3 is a component used throughout Figure 3.2. These gating mechanisms skip over any unused components of the model (learned from the data), providing adaptive depth and flexibility to apply nonlinear processing only where needed. The GRN takes in a primary input **a** and, depending on where the GRN is situated, makes use of static variables with a context vector **c** and yields:

$$\mathrm{GRN}_w(\mathbf{a},\mathbf{c}) = \mathrm{Norm}(\mathbf{a} + \mathrm{GLU}_w(\boldsymbol{\eta}_1)), \tag{3.1}$$

$$\boldsymbol{\eta}_1 = \mathbf{W}_{1,w}\boldsymbol{\eta}_2 + \mathbf{b}_{1,w}, \tag{3.2}$$

$$\boldsymbol{\eta}_2 = \mathrm{ELU}(\mathbf{W}_{2,w}\mathbf{a} + \mathbf{W}_{3,w}\mathbf{c} + \mathbf{b}_{2,w}), \tag{3.3}$$

$$\mathrm{GLU}_w(\boldsymbol{\eta}_1) = \sigma(\mathbf{W}_{4,w}\boldsymbol{\eta}_1 + \mathbf{b}_{4,w}) \odot (\mathbf{W}_{5,w}\boldsymbol{\eta}_1 + \mathbf{b}_{5,w}), \tag{3.4}$$

where ELU is the Exponential Linear Unit activation function (see Clevert et al., 2016), $\boldsymbol{\eta}_1$ and $\boldsymbol{\eta}_2$ are intermediate layers, Norm is layer normalization (see Lei Ba et al., 2016), w is an index to denote weight sharing, $\sigma(\cdot)$ is the sigmoid activation function, **W** and **b** are weights and biases, $\odot$ is the element-wise Hadamard product, and GLU are Gated Linear Units (see Dauphin et al., 2017). Both ELU and GLU help the network understand which input transformations are simple and which require more complex modeling by suppressing the nonlinear contribution.

3.1.2 Variable Selection Networks

Some features hold less predictive power over the output than others. The Variable Selection Network (VSN), depicted in Figure 3.2 and Figure 3.4, not only offers insights into which variables are most significant for the prediction task but also enables the TFT to eliminate unnecessary and noisy inputs that could adversely affect performance. Given that there are three types of inputs, the TFT uses three instances of the VSN. Consequently, each instance has distinct weights (signified by the different colors of each VSN unit in Figure 3.2), but the functional form remains consistent across all instances.

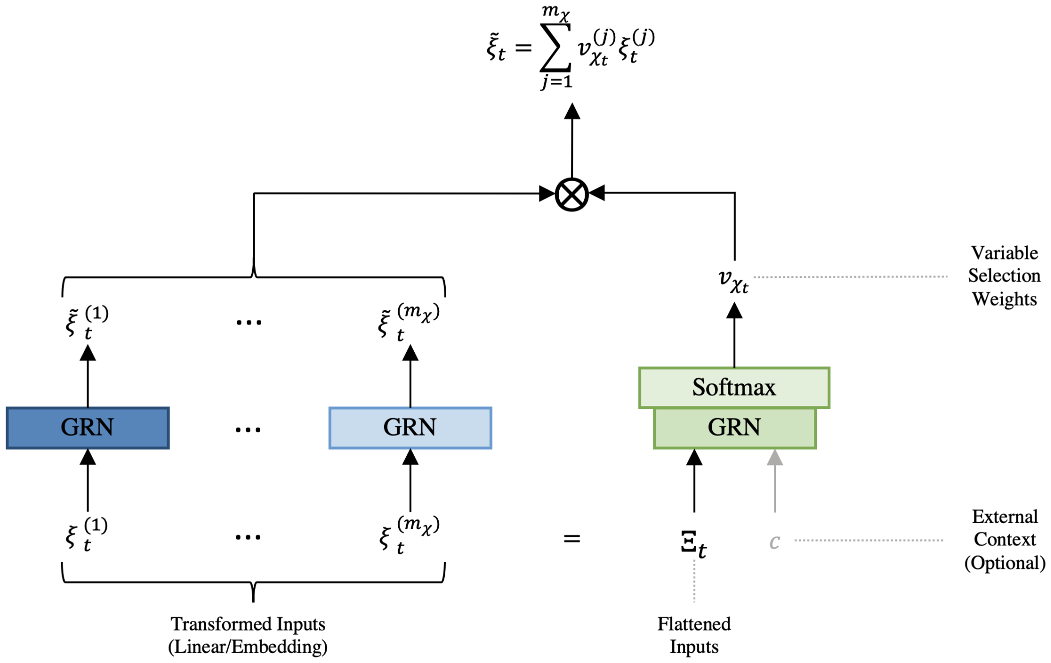
$\tilde{\xi}_t = \sum_{j=1}^{m_\chi} v_{\chi_t}^{(j)} \tilde{\xi}_t^{(j)}$
$\tilde{\xi}_t^{(1)}$
...
$\tilde{\xi}_t^{(m_\chi)}$
v_{χ_t}
Variable Selection Weights
Softmax
GRN
GRN
GRN
$\xi_t^{(1)}$
$\xi_t^{(m_\chi)}$
=
Ξ_t
c
External Context (Optional)
Transformed Inputs (Linear/Embedding)
Flattened Inputs

Figure 3.4 Variable Selection Network (VSN).

VSNs encode the variables based on their data types — categorical or numerical. Entity embeddings (see Gal and Ghahramani, 2016) are used for categorical variables, while numerical variables are put through linear transformations. Let $\boldsymbol{\xi}_t^{(j)}$ denote the transformed input of the jth variable at time t, and $\Xi_t = \left[\boldsymbol{\xi}_t^{(1)^T}, ..., \boldsymbol{\xi}_t^{(m_\chi)^T}\right]^T$ the flattened vector of all variable's past inputs. Ξ_t and a context vector $\mathbf{c_s}$ (see subsection 3.1.3) are fed through a filtering GRN unit and then a softmax function, producing a normalized vector of weights $\mathbf{v}_{\chi_t}$. Note that the VSN for static variables does not take into account the context vector. Moreover, each transformed variable passes through its own GRN with weights shared across all timesteps t, which adds an additional layer of filtering. Lastly, each processed variable is then weighted by their variable selection weights $v_{\chi_t}^{(j)}$ and combined, where $v_{\chi_t}^{(j)}$ is the jth element of $\mathbf{v}_{\chi_t}$.

3.1.3 Static Covariate Encoders

The TFT integrates information from static data by running the output of the static data VSN through four separate GRNs, resulting in four distinct context vectors. These context vectors are wired into various locations in Figure 3.2 where static variables play an important role in processing. More specifically, these contexts are for (1) temporal variable selection ($\boldsymbol{c_s}$), (2) local processing of temporal features ($\boldsymbol{c_c}, \boldsymbol{c_h}$), and (3) enriching of temporal features with static information ($\boldsymbol{c_e}$).

3.1.4 Temporal Fusion Decoder

Up to this point, the input has passed through VSN and has properly transformed and weighted the features. However, since our input is time series data, points of significance are often identified in relation to their surrounding values so the model should also make sense of the sequential ordering.

The original Transformer by Vaswani et al. (2017) addresses the problem by using self-attention mechanisms and stacked layers of encoders and decoders and the connections between them. To account for all types of inputs and the differing number of past and future inputs, the TFT builds upon the idea of the original Transformer with a sequence-to-sequence layer to naturally handle these differences — feeding $\tilde{\boldsymbol{\xi}}_{t-k:t}$ into the encoder and $\tilde{\boldsymbol{\xi}}_{t+1:t+\tau_{\max}}$ into the decoder. Inspired by its success in canonical sequential encoding problems, the TFT considers the use of stacked LSTMs for the encoder-decoder. LSTM is described in detail in Appendix A, but it suffices to know that it provides an appropriate inductive bias for the time ordering of the inputs. The LSTM encoder-decoder module produces context-aware embeddings which serve as inputs into the temporal fusion decoder itself.

Directly combining the embedding with the context vectors $\boldsymbol{c}$ from static data would result in an information mix that does not accurately represent the separate influences of static and temporal variables on the target output. To allow static data to influence local processing, the initial hidden state h_0 and the cell state c_0 of the first LSTM are initialized with the $\boldsymbol{c_h}$ and $\boldsymbol{c_c}$ vectors respectively. As a consequence, the final context-aware embeddings will be properly conditioned on the exogenous information, without altering the temporal dynamics. After temporal processing, the TFT enhances the temporal features with static data using the vector $\boldsymbol{c_e}$ in an enrichment layer.

Following static enrichment, the TFT employs a self-attention mechanism. All Transformer-based architectures leverage attention to learn long-range dependencies across different timesteps. In general, attention mechanisms scale values based on relationships between keys and queries. These terms are inspired by the retrieval process in information systems, where you use a query to search a database (which contains keys) to retrieve the corresponding values. Each input vector is linearly transformed into query ($\boldsymbol{Q}$), key ($\boldsymbol{K}$), and value ($\boldsymbol{V}$) matrices using the weight matrices $\boldsymbol{W}_Q$, $\boldsymbol{W}_K$, and $\boldsymbol{W}_V$ which are formed during the training process. These transformed matrices are used to calculate the attention scores, which determine the importance of each timestep in the context of the current timestep:

$$\text{Score}(\boldsymbol{Q}, \boldsymbol{K}) = \text{Softmax}(\boldsymbol{Q}\boldsymbol{K}^T / \sqrt{d_{\text{attn}}}), \tag{3.5}$$

where the scores are divided by the square root of the key vector dimension to get more stable gradients and the result is passed through a softmax function which normalizes the scores so they are positive and add up to 1. This softmax score determines how much each timestep will be expressed at this position. The score is then multiplied by the value matrix:

$$\text{Attention}(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}) = \text{Score}(\boldsymbol{Q}, \boldsymbol{K})\boldsymbol{V} \tag{3.6}$$

The intuition here is to keep intact the values of the focal timestep(s), and drown-out irrelevant timesteps.

To improve the learning capacity of the standard attention mechanism, Vaswani et al. (2017) proposes multi-head attention, employing multiple heads to project the input embeddings into different representation subspaces:

$$\text{Multi-head}(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}) = [\boldsymbol{H}_1, ..., \boldsymbol{H}_{m_H}]\, \boldsymbol{W}_H, \tag{3.7}$$

$$\boldsymbol{H}_h = \text{Attention}\left(\boldsymbol{Q}\boldsymbol{W}_Q^{(h)}, \boldsymbol{K}\boldsymbol{W}_K^{(h)}, \boldsymbol{V}\boldsymbol{W}_V^{(h)}\right), \tag{3.8}$$

where $\boldsymbol{W}_Q^{(h)}$, $\boldsymbol{W}_K^{(h)}$, $\boldsymbol{W}_V^{(h)}$ are head-specific weights for keys, queries and values, and $\boldsymbol{W}_H$ is a matrix that condenses the concatenated matrices $\boldsymbol{H}_h$ into a single matrix.

The drawback of this approach is that the weight matrices have no common ground and thus cannot be easily interpreted. TFT's multi-head attention adds a new grouping such that the different heads share some weights, and employ additive aggregation of all heads:

$$\text{Interpretable Multi-head}(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}) = \widetilde{\boldsymbol{H}}\boldsymbol{W}_H, \tag{3.9}$$

$$\begin{aligned}
\widetilde{\boldsymbol{H}} &= \tilde{A}(\boldsymbol{Q}, \boldsymbol{K})\boldsymbol{V}\boldsymbol{W}_V, && (3.10)\\
&= \left\{\frac{1}{m_H}\sum_{h=1}^{m_H} \text{Score}\left(\boldsymbol{Q}\boldsymbol{W}_Q^{(h)}, \boldsymbol{K}\boldsymbol{W}_K^{(h)}\right)\right\}\boldsymbol{V}\boldsymbol{W}_V, && (3.11)\\
&= \frac{1}{m_H}\sum_{h=1}^{m_H} \text{Attention}\left(\boldsymbol{Q}\boldsymbol{W}_Q^{(h)}, \boldsymbol{K}\boldsymbol{W}_K^{(h)}, \boldsymbol{V}\boldsymbol{W}_V\right), && (3.12)
\end{aligned}$$

where $\boldsymbol{W}_V$ are value weights shared across all heads. Comparing equation (3.10) to (3.6), we see that the final output of interpretable multi-head attention bears a strong resemblance to a single attention layer — the key difference being an increased representation capacity over multiple heads, while still allowing simple interpretability studies to be performed by analyzing a single set of attention weights.

Note that decoder masking (see S. Li et al., 2019; Vaswani et al., 2017) is applied to the multi-head attention layer to ensure that each temporal dimension can only attend to features preceding it. Thus the layer preserves the causal information flow.

The final part of the temporal fusion decoder is a position-wise feed-forward layer that applies additional nonlinear processing to the outputs of the self-attention layer using GRNs. As seen in Figure 3.2, this layer is also directly connected to the LSTM layer through a gate that skips over the entire transformer block — yielding a simpler model if additional complexity is not required.

3.1.5 Prediction Intervals

In financial time series forecasting, the prediction of the target variable is not sufficient, and it is equally important to estimate the uncertainty of the prediction. TFT addresses this by generating prediction intervals alongside point forecasts. This is achieved by the simultaneous prediction of various percentiles at each timestep. Quantile forecasts are generated using a linear transformation of the output $\tilde{\psi}$ from the temporal fusion decoder:

$$\hat{y}(q) = \boldsymbol{W}_q \tilde{\psi}(t, \tau) + b_q \tag{3.13}$$

where $\boldsymbol{W}_q$ and b_q are linear coefficients for the specified quantile q.

3.2 Model Interpretability

The aforementioned components in the model design can be analyzed to interpret the general relationships the model has learned. The TFT architecture enables three valuable interpretability use cases: helping users identify (1) globally-important variables for the prediction problem, (2) persistent temporal patterns, and (3) significant events.

3.2.1 Variable Importance

TFT calculates the impact of each variable by taking the robustness of predictions into account. During training, the weights of the VSNs, denoted by $v_{\chi_t}^{(j)}$ for each variable j, are updated based on the prediction error. Variable importance can then be measured by aggregating the selection weights, recording the percentiles of each sampling distribution.

3.2.2 Temporal Patterns

The analysis of persistent temporal patterns is often key to understanding the time-dependent relationships present in the dataset. In contrast to traditional and ML time series methods, which rely on model-based specifications for seasonality and lag analysis, the TFT can learn such patterns from raw training data through self-attention.

The self-attention weights from the interpretable multi-head layer reveal which timesteps during the lookback and step ahead periods are the most important by measuring the contributions of features at fixed lags in the past on forecasts at various horizons. From equation (3.10), the self-attention layer contains the combined matrix $\tilde{A}$ of scores at each forecast time t. Multi-head attention outputs at each forecast horizon τ can then be described as a score-weighted sum of the preceding features at each position i:

$$\beta(t,\tau) = \sum_{i=-k}^{\tau_{\max}} \alpha(t,i,\tau)\tilde{\boldsymbol{\theta}}(t,i), \tag{3.14}$$

where $\alpha(t,i,\tau)$ is the (τ,i)-th element of $\tilde{A}$, and $\tilde{\boldsymbol{\theta}}(t,i)$ is a sequence being processed in the layer. Due to decoder masking, we also note that $\alpha(t,i,j) = 0, \forall i > j$. For each forecast horizon τ, the average importance of a previous time point $i < \tau$ can hence be determined by analyzing the weights $\alpha(t,i,\tau)$ across all timesteps and entities. As a consequence, visualization of those weights reveals the most prominent seasonalities.

3.2.3 Regime Identification

Financial time series are notorious for being susceptible to sudden changes in their properties during rare events (Ang and Timmermann, 2012). Even worse, those events are typically very elusive. Identifying such latent regime changes and altering behavior accordingly provides strong insights into the underlying problem of predicting stock movements.

TFT allows us to measure significant shifts in temporal dynamics for a given entity using the distance between attention vectors at each point in time with the entity's average pattern from subsection 3.2.2, aggregated over all forecast horizons:

$$\text{dist}(t) = \sum_{\tau=1}^{\tau_{\max}} \kappa\left(\overline{\boldsymbol{\alpha}}(t), \boldsymbol{\alpha}(t,\tau)\right)/\tau_{\max} \tag{3.15}$$

where $\overline{\boldsymbol{\alpha}}(t)$ is the average attention weight vector and $\boldsymbol{\alpha}(t,\tau)$ is the attention weight vector, $\kappa(\boldsymbol{p},\boldsymbol{q}) = \sqrt{1-\rho(\boldsymbol{p},\boldsymbol{q})}$ is a distance metric using using the Bhattacharyya coefficient $\rho(\boldsymbol{p},\boldsymbol{q}) = \sum_j \sqrt{p_j q_j}$ measuring the overlap between discrete distributions (Kailath, 1967).

3.3 Dataset

In this paper, we choose to study temporal stock performance effects, such as momentum and reversals, in the U.S. stock market due to its benefits. Firstly, the U.S. market is one of the largest and most liquid markets in the world, making it highly interesting to international investors and relevant from a practical and real-world perspective. Secondly, it encompasses a wide range of industries and sectors, providing a diverse set of stocks for nuanced analysis. Thirdly, the U.S. market has a rich history and availability of historical data, allowing for comprehensive backtesting and evaluation of trading strategies. Finally, many studies and benchmarks in momentum research focus on this particular market, which enhances comparability and benchmarking against existing literature.

As a proxy for the U.S. stock market, we use the S&P 500 index. The S&P 500 is a stock market index that represents a selection of about 500 large-cap U.S. stocks chosen based on various criteria, such as market capitalization, liquidity, and industry representation. These stocks are chosen by Standard & Poor's, the index provider, which periodically reviews and updates the composition of the index. The index is widely considered a benchmark for the overall performance of the U.S. stock market as it covers a significant portion of the market capitalization of publicly traded companies in the U.S. and represents a diverse range of industries. As a result, changes in the S&P 500 are often seen as a reflection of the health and direction of the U.S. economy. The general development of the S&P 500 index is plotted in Figure 3.5.

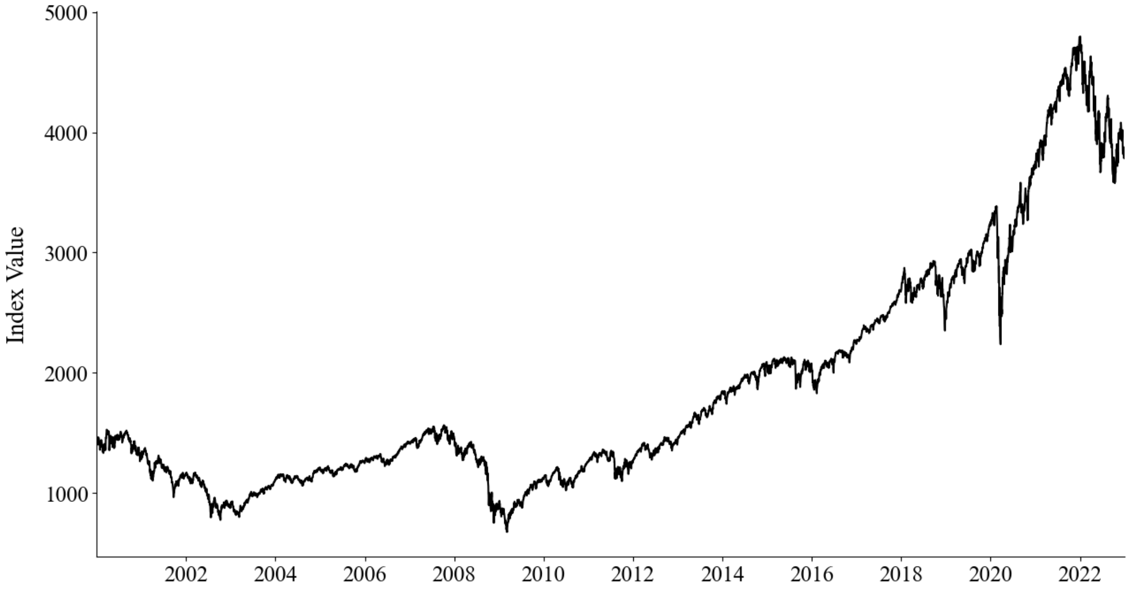

Figure 3.5 S&P 500 index value from Jan. 2000 to Dec. 2022.

3.3.1 Data Sources and Pre-processing

The datasets collected for this study encompass S&P 500 constituents data, U.S. one-month treasury bill rates, and index values of the S&P 500 and the Volatility Index (VIX). After data processing and feature generation, the S&P 500 constituents dataset is ultimately used as input to the TFT model to generate predictions and unveil intricate temporal relationships. The treasury bill rates and index values of the S&P 500 and the VIX are used as proxies for the risk-free rate, market return, and market sentiment, respectively. The treasury bill data is retrieved on a monthly basis from the Kenneth R. French database, whereas all other datasets are collected from the Refinitiv Eikon database at daily frequencies. The datasets range from January 2000 up to and including December 2022, which defines our study period.

As outlined in Chapter 2, future performance may be influenced by company-specific features such as liquidity, size, and industry apart from past price performance in isolation. Thus, we retrieve daily adjusted closing prices, bid/ask prices, market capitalizations, industries (as given by the Thomson Reuters Business Classification (TRBC) economic sector name), and ticker symbols for the 1048 different stocks with constituency on the S&P 500 from January 2000 up to and including December 2022. As a complete S&P 500 dataset is not directly attainable from Refinitiv Eikon, several queries and operations need to be conducted in order to assemble a dataset that resembles the S&P 500. We first obtain daily constituent lists for the S&P 500 throughout the study period. Then, complete time series of daily closing prices, bid/ask prices, market capitalizations and industries across the entire study period are fetched for the 1048 stocks with index constituency. Thereafter, the constituents lists are utilized for filtering to avoid survivorship bias: Delisted and transferred stocks are included up to the date of the delisting or transfer. As such, we are able to approximately reproduce the S&P 500 at any given point in time between January 2000 and December 2022. Additionally, we note that the ”adjusted” closing prices available in Refinitiv Eikon are solely adjusted for corporate events, stock splits and reverse stock splits, but not for dividends. Thus, to attain a more accurate representation of price development, net dividend payouts are collected separately to adjust the closing prices.

To transform the S&P 500 dataset into a format that can be processed by the TFT model, data pre-processing is required. We assume that stocks with gaps in the price data were not traded during this period and thus have the same price until the price is updated. Missing values are therefore handled by forward filling, as the TFT requires continuous input data.

3.3.2 Feature Generation and Target Selection

We create and incorporate various input features to fully leverage the TFT capabilities in handling different types of inputs (i.e. static, known, and observed) for accurate predictions. By generating features that cater to these input categories, we aim to maximize the TFT's potential in capturing the underlying patterns and relationships within the data. In turn, various input types allow the model to adapt its behavior, focus on relevant features, and identify significant temporal dependencies.

Firstly, we enrich the dataset with date-related features by using the timestamps of the retrieved variables. For each row, we add the corresponding day-of-month, month, and quarter. Then, we incorporate proxies for liquidity and company size. We use the bid-ask spread percentage to gauge liquidity, and the market capitalization as a proxy for company size. However, due to the trillions of dollars difference in market capitalization of the large and small companies in the dataset, we apply a log transformation to mitigate the impact of extreme values. Then, we use the closing price to calculate simple daily returns. Finally, a time index is added to denote the number of days from the first day of the study period, i.e. $t \in \{0, ..., T_{\max}\}$, thereby providing the model with a contextual understanding of the chronological sequence of events. We treat all date-related variables (i.e. day-of-month, month, and quarter), the stock ticker symbol, and industry as categorical inputs. We keep the real values of the return, time index, liquidity, and size. The final data definitions are summarized in Table 3.1, and represent our 9 input features $F_t^{(j,s)}$ for $j \in \{1, ..., 9\}$ and for each stock $s \in S$ (ignoring the time subscript when dealing with static variables) which comprise our set of features F.

Table 3.1
Data definitions.

Feature j	Data type	Input type
Return	Real-valued	Target
Time index	Real-valued	Known
Liquidity	Real-valued	Observed
Size	Real-valued	Observed
Day of month	Categorical	Known
Month	Categorical	Known
Quarter	Categorical	Known
Ticker symbol	Categorical	Static
Industry	Categorical	Static

TFT networks require sequences of input features for training, i.e. the values of the features at consecutive points in time. We generate sequences by looking back 252 business days ($\approx$ 1 year) and looking ahead 126 business days ($\approx$ 6 months), equivalent to one and a half years of financial data. While our model is trained using daily past data, we recognize that forecasting daily returns over the monthly horizons commonly described in the momentum literature may not yield the most informative results. In order to align our forecasts with the temporal granularity that is most appropriate to our analysis, we resample the future values to monthly data. Specifically, the daily return values, which serve as our targets $R_t^{(s)}$, are accumulated into monthly returns. Consequently, we input the network with sequences that comprise $k = 252$ daily observed timesteps and $\tau_{\max} = 6$ monthly known timesteps. Moreover, we generate overlapping sequences to avoid a bias introduced by the starting date as noted by Jegadeesh and Titman (1993). Thereby, each sequence is offset to the other by one trading day (see Figure 3.6). By including sequences starting on all days during the study period, we reduce the noise that comes from, among other things, earnings announcements, holiday/weekend effects, and investors seeking tax benefits at the turn of the year.

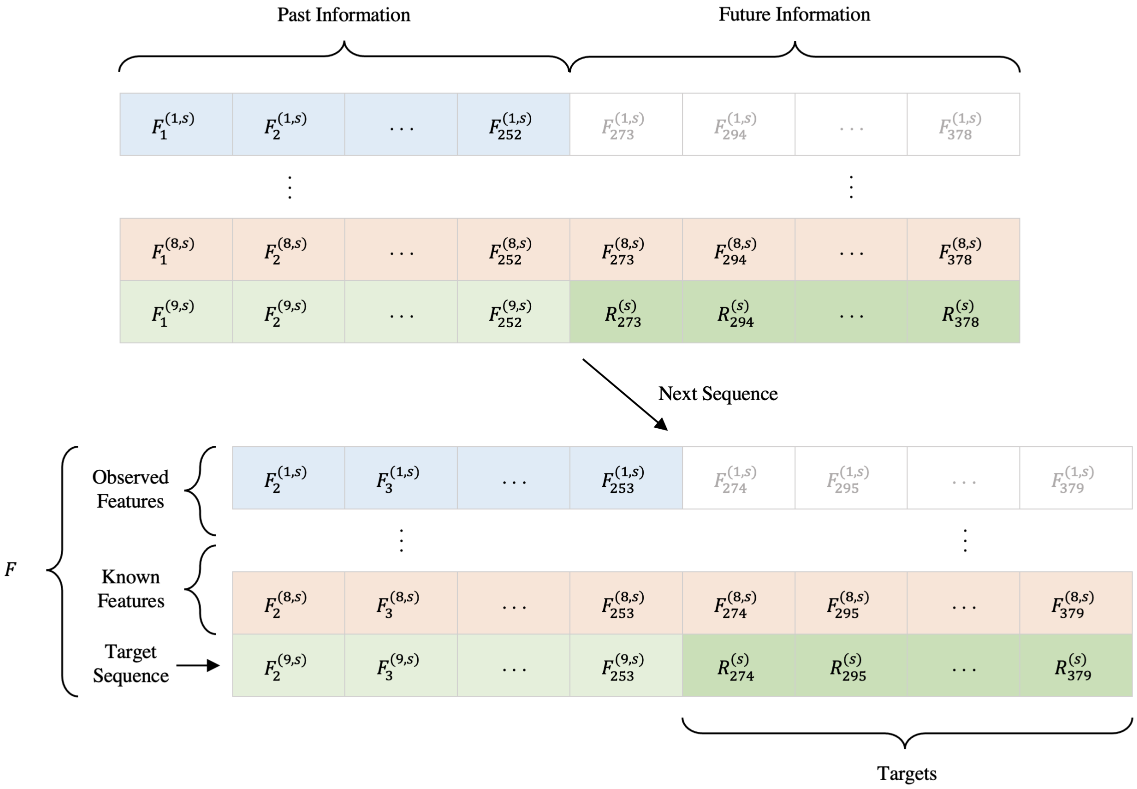
Past Information
Future Information
$F_1^{(1,s)}$ $F_2^{(1,s)}$ … $F_{252}^{(1,s)}$ $F_{273}^{(1,s)}$ $F_{294}^{(1,s)}$ … $F_{378}^{(1,s)}$
$F_1^{(8,s)}$ $F_2^{(8,s)}$ … $F_{252}^{(8,s)}$ $F_{273}^{(8,s)}$ $F_{294}^{(8,s)}$ … $F_{378}^{(8,s)}$
$F_1^{(9,s)}$ $F_2^{(9,s)}$ … $F_{252}^{(9,s)}$ $R_{273}^{(s)}$ $R_{294}^{(s)}$ … $R_{378}^{(s)}$
Next Sequence
F
Observed Features
Known Features
Target Sequence
$F_2^{(1,s)}$ $F_3^{(1,s)}$ … $F_{253}^{(1,s)}$ $F_{274}^{(1,s)}$ $F_{295}^{(1,s)}$ … $F_{379}^{(1,s)}$
$F_2^{(8,s)}$ $F_3^{(8,s)}$ … $F_{253}^{(8,s)}$ $F_{274}^{(8,s)}$ $F_{295}^{(8,s)}$ … $F_{379}^{(8,s)}$
$F_2^{(9,s)}$ $F_3^{(9,s)}$ … $F_{253}^{(9,s)}$ $R_{274}^{(s)}$ $R_{295}^{(s)}$ … $R_{379}^{(s)}$
Targets

Figure 3.6 Data sequences of observed and known features. The observed features, which would not be available in real-time, are masked in the future timesteps to maintain the accuracy of the simulation. However, the observed target sequence remains unmasked throughout, allowing the network to continuously compare its output with these values for weight adjustments. Static variables, which remain constant, are not depicted here as their values do not change across timesteps.

For each stock $s \in S$, we partition the dataset into $|S|$ 2D matrices of $|F|$ columns and a maximum of $T_{\max}$ rows, where $|F|$ is the number of features, i.e., the number of variables and their lags hence equal to $(252 + 6, 9)$. We fill the matrices with the respective $|F|$ features as defined above. The final dataset is illustrated in Figure 3.7. Finally, we transform the 2D tabular data into a 3D matrix with shape `(num_samples, time_steps, num_features)` by concatenating the data of all stocks in S to get the collective dataset fit for input to the TFT.

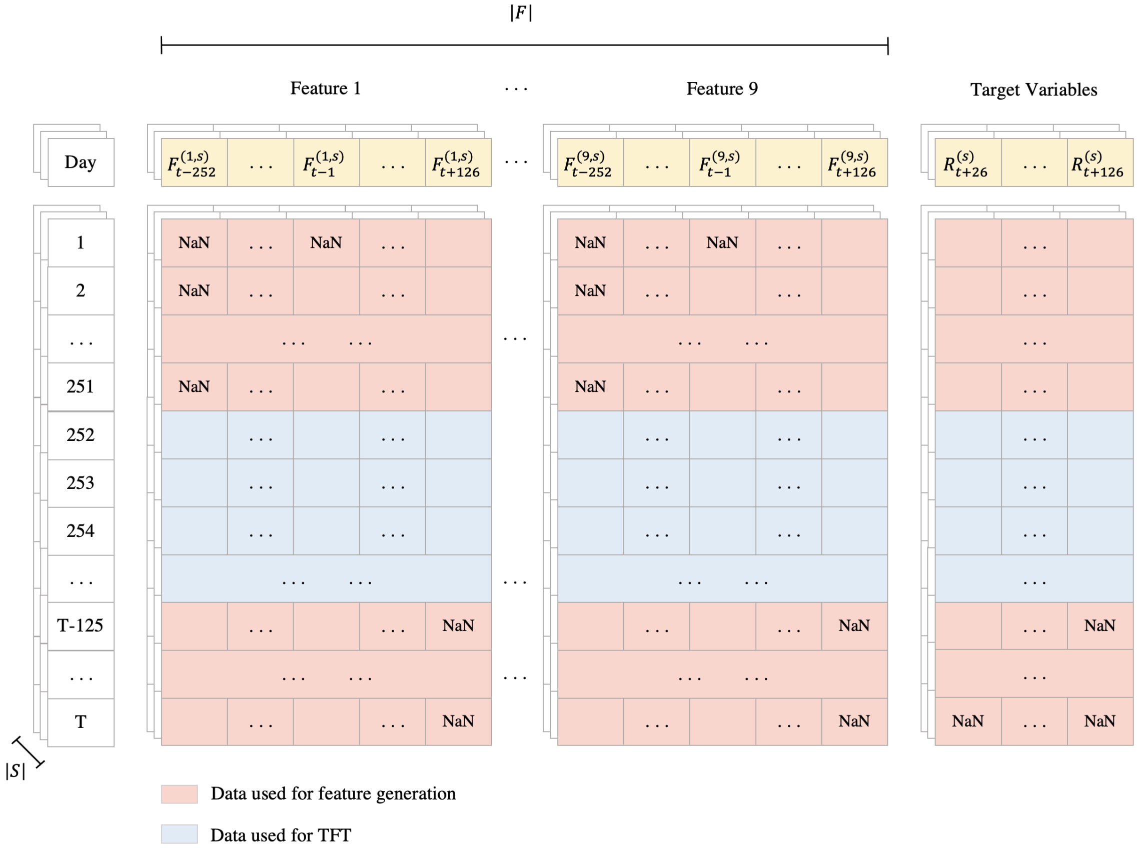

Figure 3.7 Data batching for TFT. One input sample is defined by one row of one matrix s. Since by definition, $F_t^{(j,s)}$, is not defined when t is less (greater) than the stock's first (last) day on the index, columns of the top 252 and bottom 126 rows are partially filled and hence only used for feature generation.

3.3.3 Summary Statistics

Although we initially retrieve daily data for the 1048 constituents, 155 of them are lost due to feature generation, i.e. these are stocks that do not exhibit one and a half years (252 + 126 business days) of continuous time series data as required by the model. After feature generation, we are thus left with 893 different stocks in the dataset. Figure 3.8 displays the number of stocks per timestep in our S&P 500 dataset before and after feature generation. Unsurprisingly, the number of stocks per timestep prior to feature generation remains relatively stable at around 500. As indicated by the difference between the black line and the dark shaded area, the feature generation process causes us to reduce the number of stocks per timestep. More specifically, the average, maximum and minimum number of stocks per timestep is 497 (480), 506 (502) and 477 (387), respectively, before (after) feature generation is conducted. The noticeable dips observed before 2002, in 2018, and in the midst of 2020 are caused by the process of partitioning the dataset into training, validation, and test sets. Right before the split points, some stocks will naturally leave the index, and stocks taking their place will never have one year of past return on the index and are therefore also not included.

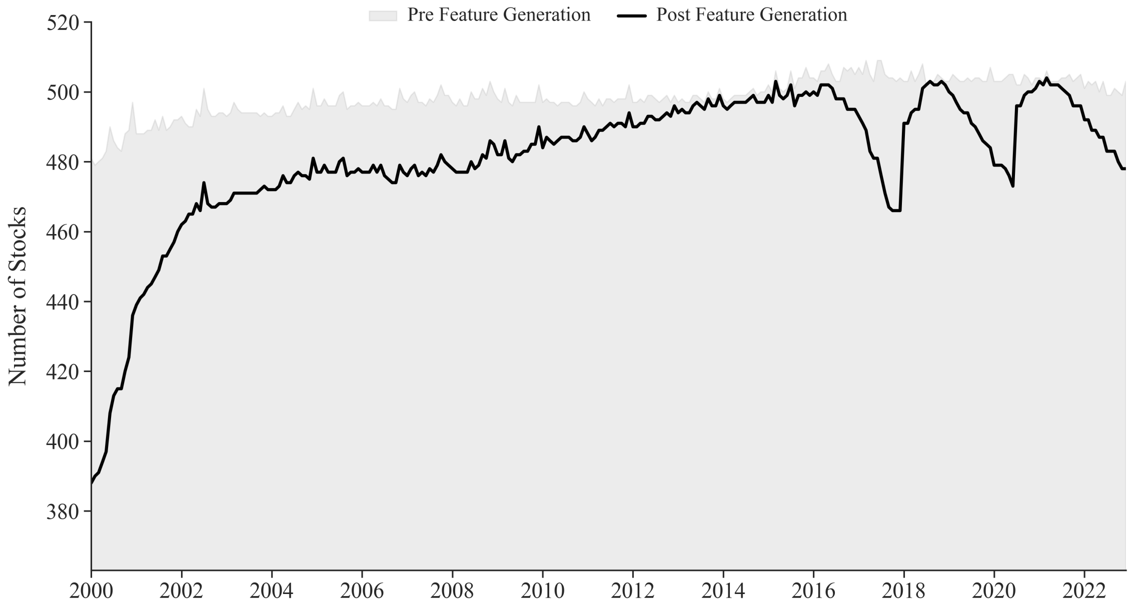

Figure 3.8 Number of stocks per timestep before and after feature generation.

Table 3.2 shows the summary statistics of the past observed input features in the S&P 500 dataset after data processing and feature generation. When looking at the daily returns, the mean daily return of 0.05% is slightly higher than the median of 0.04%, indicating a slight right skewness in the distribution. Although the mean and median returns are close to zero, the effect of compounding over time is evident in Figure 3.5 where small positive returns accumulate and lead to a larger overall gain. The range of returns is quite large, with a maximum (minimum) daily return of 288.01% (-89.63%). Additionally, the standard deviation of the returns is relatively high at 2.43%, indicating that there is a significant degree of variability in the returns. Taken together, these results suggest that the S&P 500 constituents experience periods of both large gains and losses, but with a marginal upward trend.

Table 3.2
Summary statistics of past observed input features. Returns are daily returns, liquidity is approximated using the bid-ask spread percentage, and the size is represented by the logarithm of market capitalization.

	Return	Liquidity	Size
Mean	0.0005	0.0008	23.4917
St. dev.	0.0243	0.0016	1.1696
Minimum	-0.8963	0.0000	15.1574
Quartile 1	-0.0094	0.0002	22.7278
Median	0.0004	0.0004	23.3900
Quartile 3	0.0103	0.0010	24.1739
Maximum	2.8801	0.4688	28.7206

In total, the liquidity statistics suggest that liquidity levels are generally high (narrow bid-ask spreads) and stable. Similar to the return distribution, the bid-ask spread distribution show mean and median values close to zero, which reflects the efficiency of the S&P 500. The mean and median values being marginally positive, suggest ask prices being slightly higher than bid prices — a natural cause of the compensation market makers earn for the risk of providing liquidity to buyers and sellers in the market. Furthermore, there might be occasional instances of lower liquidity, as indicated by the maximum spread of 46.88%, but this extreme value is likely an outlier, as the quartiles and the mean indicate more moderate levels.

When examining summary statistics for size, a higher mean than median value tells that the S&P 500 comprises a minority of very large companies pulling the mean up while the majority exhibit lower market capitalizations. Additionally, the standard deviation is also quite large at 1.17, indicating that there is significant variability in the market capitalization sizes of S&P 500 constituents. This is further supported by the large dispersion between the maximum and minimum log market capitalizations of 28.72 and 15.16, respectively.

In Table 3.3 the 893 different constituents are categorized into 11 different industries as defined by the TRBC economic sector names. The table shows that the industry distribution is relatively evenly spread out, with no single industry dominating the S&P 500 index. Technology is the largest industry, with 159 stocks, making up 17.79% of the dataset. Conversely, only three stocks are categorized under Academic & Educational Services, accounting for less than 1% of the dataset.

Table 3.3
Industry distribution of the S&P 500 dataset after processing and feature generation.

Industry	Number of Stocks	Percentage
Technology	159	17.7852
Consumer Cyclicals	142	15.8837
Financials	112	12.5280
Industrials	110	12.3043
Healthcare	98	10.9620
Energy	69	7.7181
Consumer Non-Cyclicals	67	7.4944
Basic Materials	53	5.9284
Utilities	41	4.5861
Real Estate	39	4.3624
Academic & Educational Services	3	0.3356

3.4 Training Procedure

We partition the dataset into three parts — a training set for learning, a validation set for weight tuning, and a hold-out test set for performance evaluation.

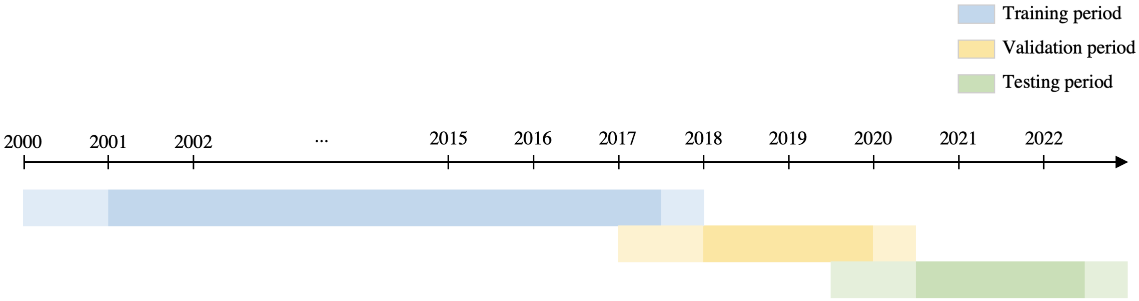

Figure 3.9 Training, validation, and testing periods.

Let n denote the number of stocks with price data in the training period. For the training set, we consider all n stocks with the history they have available. Some stocks exhibit a full 17-year price history, and some only a subset of this timeframe, e.g. when they are listed later.

The testing set is composed of m stocks, possibly different from the n stocks in the training period due to listings and delistings. The difference is negligible as the main goal is to learn the general past-performance effect on stock returns, not stock-specific movements. Further, if a constituent exhibits no price data after a certain day within the testing period, it is considered for trading up until $\tau_{\max}$ months before that day. Note that to avoid survivorship bias, we do not intentionally eliminate stocks during the training or testing period in case they drop out of the S&P 500. The only criterion for being trained on is that they have price information available for feature generation (see subsection 3.3.2). Thus, each of the dataset partitions includes a feature generation year corresponding to the maximum lookback window k which may overlap other partitions, and a feature generation semester corresponding to the maximum step ahead window $\tau_{\max}$, as illustrated by the lighter periods in Figure 3.9. Note that the chosen testing period effectively captures the recent non-stationarity observed in financial markets: Our testing period is characterized by significant market volatility, as evidenced by substantial fluctuations in index value in Figure 3.5.

In our time series input data, the subsequent instance is merely the preceding instance offset by a single day, resulting in a long computational time if we train on all instances and minimal variation between adjacent instances. To address this, we employ a sampling strategy that randomly selects 750,000 samples uniformly distributed over the entire in-sample period. We allocate 90% of these samples from the training period for training purposes and the remaining 10% from the validation period for validation. The test set, on the other hand, remains unaltered and we test on all instances to avoid selection bias and ensure the integrity of the results.

3.4.1 Feature Normalization

Feature normalization is a critical processing step in DNNs to improve model performance and facilitate training. Feature normalization ensures that real-valued features are on a similar scale, preventing the model from being biased towards features with larger magnitudes (Ioffe and Szegedy, 2015). Furthermore, feature normalization is vital in mitigating issues such as exploding gradients and unstable training. When input features have different scales, the gradients can become disproportionately large, leading to an exploding gradient problem (Pascanu et al., 2013). This issue can cause the training to diverge, making it difficult to find the optimal set of weights for the model (Bengio et al., 1994). Therefore, normalization promotes convergence and better generalization during training (Lecun et al., 1998).

We consider $F_t^{(j,s)}$ our jth feature of stock s. We standardize the real-valued features using z-score normalization across all entities as we find it beneficial to assess relative performance, sizing, and liquidity:

$$\tilde{F}_t^{(j,s)} = \frac{F_t^{(j,s)} - \mu_{\text{train}}^j}{\sigma_{\text{train}}^j}$$

where μ_{train}^j is the mean and σ_{train}^j is the standard deviation of the jth feature in the training period. To avoid data leakage, the mean and standard deviation are obtained from the training set only.

For the categorical features, we translate the non-numerical data into a numerical format that can be fed into the network. This transformation enables the model to capture relationships and patterns in the categorical features, ultimately enhancing its predictive capabilities (Goodfellow et al., 2016).

In addition, TFT incorporates multiple normalization gates throughout Figure 3.2 to improve its performance and stability during training. These gates serve as a crucial component for effective gradient flow, ensuring that the model converges more efficiently.

3.4.2 Loss Function

TFT is trained by minimizing the quantile loss summed across all quantile outputs $\hat{y}$ from subsection 3.1.5:

$$\mathcal{L}(\Omega, \boldsymbol{W}) = \sum_{y_t \in \Omega} \sum_{q \in Q} \sum_{\tau=1}^{\tau_{\max}} \frac{QL(y_t, \hat{y}(q, t-\tau, \tau), q)}{N_{\tau_{\max}}} \tag{3.16}$$

$$QL(y, \hat{y}, q) = \max\left[q(y - \hat{y}), (1 - q)(y - \hat{y})\right] \tag{3.17}$$

where Ω is the domain of training data containing N samples, $\boldsymbol{W}$ represents the weights of TFT, and $Q = \{0.1, 0.5, 0.9\}$ are the 10th, 50th and 90th quantiles. In the quantile regression loss in Equation 3.17, the first term will be positive and dominate when under-predicting ($\hat{y} < y$), and the second term will dominate when over-predicting ($\hat{y} > y$). For q equal to 0.5, under-prediction and over-prediction will be penalized by the same factor, and the median is obtained. The larger the value of q, the more under-predictions are penalized compared to over-predictions. For instance, for q equal to 0.9, under-predictions will be penalized by a factor of 0.9, and over-predictions by a factor of 0.1. The model will then try to avoid under-predictions approximately nine times as hard as over-predictions, and the 0.9 quantile will be obtained.

3.4.3 Parameter Optimization

The network weight calibration is undertaken using minibatch stochastic descent with the Adam optimizer and equation (3.16) as the objective function to minimize. We apply dropout regularization after the LSTM layer, the self-attention layer, and within the GRNs. Hereby, a fraction of the input units are randomly dropped at each update during training, resulting in a reduced risk of overfitting and better generalization. To further avoid vanishing or exploding gradients, we also clip the norm of the gradients (see Goodfellow et al., 2016, p. 396 and 409).

The specific hyperparameters that control the learning process are chosen over a pre-defined search space found in Table 3.4. An unfit combination of hyperparameters produces sub-optimal results, as they do not minimize the loss function. Therefore, hyperparameter tuning is an essential part of any ML model, and its importance is highlighted several times in the literature (see, for example, Gu et al., 2020; Takeuchi and Y.-Y. A. Lee, 2013).

Table 3.4
Hyperparameter search space.

Hyperparameter	Search grid
Hidden layer size	10, 20, 40, 80, 160
Dropout rate	0.1, 0.2, 0.3, 0.4, 0.5
Minibatch size	64, 128
Learning rate	0.0001, 0.001, 0.01
Max gradient norm	0.01, 1.0, 100.0
Number of heads	1, 4

Due to the extensive number of hyperparameter values, the search space where we seek to find our optimal model configurations quickly becomes intractable. A grid search, where all 900 possible combinations are tested, is not feasible in terms of computational time. Thus, hyperparameter optimization is conducted via 50 iterations of random search.

Backpropagation is performed up to a maximum of 100 training epochs. Then, we use the validation data to determine convergence — with early stopping triggered when the validation loss has not improved for 5 epochs to reduce the risk of overfitting further. The inherent interpretability is also used continuously during the process of training models to ensure correct model behavior and capture any erroneous coding oversights.

Each TFT model is trained on a single NVIDIA GeForce GTX 1650 GPU. Our optimal model takes slightly over 5 hours to train, with each epoch taking roughly 40 minutes. We note that the number of training samples, hidden layer, minibatch size search ranges, and training times are capped by memory usage, and TFT training can be improved with hardware-specific optimizations. The specified topology of our trained TFT model is summarized in Table 3.5.

Table 3.5
Information on dataset and optimal TFT configuration.

	Parameter	Value
Dataset details	Training set samples	675,000
	Validation set samples	75,000
Network details	Lookback k	252 business days
	Step ahead $\tau_{\max}$	6 months
	Dropout rate	0.5
	Hidden layer size	40
	Number of heads	1
Training details	Epochs	100
	Minibatch size	64
	Learning rate	0.01
	Max gradient norm	1.0
	Early stopping patience	5
Computational cost	Hardware	NVIDIA GeForce GTX 1650
	Minutes per epoch	40 min
	Total training time	320 min

3.5 Forecasting, Ranking and Trading

Our TFT model outputs 10th, 50th, and 90th percentile predictions of monthly returns for each stock at each of our six monthly prediction horizons. The percentile predictions provide a comprehensive and nuanced understanding of the range of likely return values. Specifically, the 10th percentile predictions represent conservative estimates, indicating the lower bounds of potential returns and helping assess downside risk. The 90th percentile predictions represent more optimistic estimates, reflecting the upper bounds and providing insights into potential high returns. The 50th percentile predictions (medians) serve as midpoints, offering estimates of the most likely outcomes. Compared to solely relying on point estimates, percentile predictions provide a solid foundation for creating trading signals that aim to maximize profitability while taking the associated risks into account.

3.5.1 Designing a Strategy

We aim to assess the interpretability of the TFT by developing a step-by-step decision-rule-based strategy that is grounded in the signals generated by the TFT. We leverage both the inherent interpretable components within the TFT architecture and conduct post-model analysis of the percentile predictions, which will aid in formulating the decision rules governing the proposed strategy. Our objective is to design a strategy that is sufficiently straightforward for retail investors to benefit from or apply the insights without possessing the technical expertise required to implement such an intricate model independently.

3.5.2 Benchmarks

To see if our designed strategy yields adequate results under the classic long-term momentum effect assumption, we compare it to the original Cross-sectional Momentum (CSMOM) and Time Series Momentum (TSMOM) strategies of Jegadeesh and Titman (1993) and Moskowitz, Ooi, et al. (2012), respectively.

The CSMOM strategy works as follows: On the first trading day of each month, the stocks are ranked in ascending order based on their returns in the past J months. Based on these rankings, 10 decile portfolios are formed. The zero-cost strategy buys the portfolio with the highest past

J-month return and short-sells the portfolio with the lowest past J-month return, holding this position for K months for a J-month/K-month strategy. Additionally, the strategy closes out the position initiated in month $t - K$. As a result, the weights of $1/K$ of the stocks in the overall portfolio are adjusted every month, while the remaining weights are carried over from the previous month. All stocks in the long and short portfolios are assigned equal monetary weight, and the returns of the portfolios are simply calculated as the average return of all the individual stocks they are comprised of.

The TSMOM strategy functions in the following manner: On the first trading day of each month, the strategy goes long (short) all stocks that exhibit a positive (negative) past J-month return, holding this position for K-months for a J-month/K-month strategy. Following the methodology of CSMOM, TSMOM also revises the weights of $1/K$ of the stocks in the portfolio every month. Contrary to CSMOM, the long and short portfolios are not equal-weighted portfolios, as the position size of each stock is set to be inversely proportional to its ex-ante volatility, $1/\sigma^s_{t-1}$, each month. The ex-ante volatility at each point in time is given by the exponentially weighted lagged squared daily returns. More specifically, the ex-ante annualized variance σ^2_t for each stock is calculated as follows:

$$\sigma^2_t = 261 \sum_{i=0}^{\infty} (1-\delta)\delta^i (r_{t-1-i} - \bar{r}_t)^2$$

where the scalar 261 scales the variance to be annual, the weights $(1-\delta)\delta^i$ add up to one, and r_t is the exponentially weighted average return computed similarly. The parameter δ is chosen so that the center of mass of the weights is $\sum_{i=0}^{\infty}(1-\delta)\delta^i i = \delta/(1-\delta) = 60$ days. The returns of the portfolios are determined by computing the weighted average of the individual stock returns based on their respective position sizes.

In line with Jegadeesh and Titman (1993) and Moskowitz, Ooi, et al. (2012), we explore various combinations of J and K within the boundaries of the TFT model's 12-month lookback window and 6-month step ahead horizon, selecting the optimal configurations as our CSMOM and TSMOM benchmarks. We intentionally permit the benchmarks to 'peek' at the test data during configuration testing, effectively enhancing their performance. The rationale behind this approach is that if our designed strategy is able to outperform these 'inflated' benchmarks, it provides stronger evidence of our strategy's efficacy. However, it is important to note that such an advantage is not usually available in real-world scenarios, hence any comparison should consider this element of potential bias in the benchmarks' favor. For more information regarding the performance of the tested J/K-strategies, the reader is referred to Appendix B.

Additionally, our designed strategy is compared to the market return, represented by the S&P 500 index, to further assess its potential to generate excess returns. Analogous to the CSMOM and TSMOM strategies, we buy the index on the first trading day of each month and close the position at the end of each month.

3.5.3 Performance Evaluation

In order to obtain a more realistic evaluation of performance, it is necessary to consider the impact of transaction costs. We incorporate a transaction fee of 0.05% — a fairly robust value for U.S large-cap companies. For comparison, Jha (2016) adopts a lower transaction cost assumption of 0.02% for the largest 500 stocks in the U.S. stock universe, covering a similar time period. To model transaction costs, we follow the approach of Avellaneda and J.-H. Lee (2010). Specifically, transaction costs are linearly modeled, premised on the assumption of a fixed slippage cost amounting to 0.05% per half turn. This representation indicates that the total transaction value — encompassing both buying and selling activities — is adjusted by this percentage. While this simplified assumption is widely adopted in financial modeling, it is important to acknowledge that it overlooks the nuances of real-world transaction costs. Factors such as traded share volume, brokerage specifications, and tiered pricing structures can potentially influence these costs, underlining that this transaction cost model may not perfectly mirror all intricacies of real-world trading environments. Additionally, it is important to note that our implementation does not account for typical components associated with short-selling, such as margin calls, dividend payments, and borrowing fees or interest costs. This abstraction simplifies the modeling process but also limits the model's real-world applicability and should be considered when interpreting the results.

To provide a straightforward measure of profitability, we consider the average monthly returns for our designed strategy and the benchmarks. This approach allows for a quick comparison of their average profitability and provides insights into their month-to-month performance consistency. However, it is important to recognize that relying on averages does not account for the compounding effect of returns over time, which can lead to a discrepancy between average and cumulative returns. While cumulative return measures may be able to reflect actual trading performance more accurately, they are heavily influenced by the chosen investment period, and therefore, provide a less generalizable measure of overall strategy performance. We opt for generality but emphasize that our measure of profitability reflects the expected monthly performance of each strategy, providing an indication of how well it is anticipated to perform on average and may not reflect actual trading performance.

To enhance our understanding of the strategies' performance, we complement our profitability measure with risk-related metrics and performance ratios. This allows a more nuanced performance evaluation, taking into account both the magnitude of returns and the level of risk involved. Overall, the performance of the strategies is judged based on the following metrics:

1) *Profitability* – Expected returns ($\mathbb{E}\left[R\right]$), distribution of returns (minimum, quartile 1, median quartile 3, maximum, skewness and excess kurtosis), and the percentage of positive returns observed across the testing period.

2) *Risk* – Volatility (standard and downside deviation), and the maximum drawdown of the portfolios (MDD).

3) *Performance ratios* – Risk-adjusted performance measured by the Sharpe and Sortino ratios.

Chapter 4

Results

Our results are presented in three stages. We begin by examining the interpretability use cases of the TFT's model weights in the in-sample training period, dissecting the inner workings of the model to understand the logic behind its decision-making processes. Following this, we shift our focus to post-model analyses of the TFT's outputs. With a thorough understanding of the model's operation, we then construct a novel trading strategy based on the insights learned from our interpretation of the TFT, aiming to provide improved outcomes over existing strategies in the out-of-sample testing period.

4.1 Interpretation of Model Weights

We analyze the model's behavior in the in-sample training period to avoid any look-ahead bias. This analysis focuses on the weights assigned by the TFT model, as these act as a determinant of the influence that each input has on the resulting predictions. The approach offers a comprehensive understanding of how the model processes and assigns importance to inputs. It ultimately reveals the model's internal decision-making process, providing insights into its performance and overall efficacy.

4.1.1 Analyzing Variable Importance

We analyze the distribution of variable selection weights, $v_{\chi_t}^{(j)}$, on the input layer — using this to quantify the relative importance of a given feature for the prediction problem in general. Table 4.1 shows the variable importance of static, past, and future variables.

Table 4.1
Variable importance. The 10th, 50th, and 90th percentiles and the mean of the sampling distributions of the variable selection weights are shown, with mean values larger than 0.1 highlighted in bold.

	Sampling Distribution			
Temporal Characteristic	Mean	P10	P50	P90
Static Covariates				
Ticker Symbol	**0.9819**	0.9819	0.9819	0.9819
Industry	0.0181	0.0181	0.0181	0.0181
Past Temporal Variables				
Liquidity	0.0233	0.0232	0.0233	0.0233
Size	0.0837	0.0834	0.0837	0.0839
Time Index	0.0359	0.0358	0.0359	0.0360
Day of Month	0.0817	0.0816	0.0817	0.0818
Month	**0.1098**	0.1097	0.1098	0.1099
Quarter	0.0151	0.0150	0.0151	0.0151
Returns	**0.6506**	0.6500	0.6506	0.6511
Future Temporal Variables				
Time Index	**0.7175**	0.7174	0.7175	0.7176
Day of Month	0.0006	0.0006	0.0006	0.0006
Month	**0.1444**	0.1443	0.1444	0.1445
Quarter	**0.1375**	0.1374	0.1375	0.1375

For static covariates, the largest weights are attributed to the stock ticker symbol that uniquely identifies different stocks. This finding underlines that individual stock performance supersedes broader categorizations.

For past inputs, values of the target (i.e. returns) are critical as expected, as return forecasts are inferences from past observations. The month variable is also marked, hinting at the existence of cyclical fluctuation patterns throughout the year. Another variable of note is size, which exhibits mean variable importance just shy of our 0.1 threshold. Its significance is nonetheless relatively higher than most other lower-ranking variables. On a related note, the day-of-month variable shows similar significance, albeit its importance appears to diminish in future predictions. Liquidity emerges as a less influential variable. However, it is important to bear in mind that this could potentially be attributed to the brief nature of the bid-ask spread percentage, which might undervalue its significance when considering its importance over a year.

For future inputs, the time index, month, and quarter have the greatest influence on returns forecasts. The month and quarter variables, for instance, can capture potential cyclical patterns or seasonality effects in the stock market. These effects might be attributable to various factors such as fiscal policies, earnings reports, or investor sentiment, which often exhibit periodic fluctuations throughout the year. These yearly seasonalities have been a considerable focus of study in the momentum context (see, for instance, Bird et al., 2017; Jegadeesh and Titman, 1993).

The future time index provides a sense of linear progression, making it a significant tool for the model to comprehend the chronology of events. Specifically, the time index aids the model in understanding the temporal distance between different data points and aids in recognizing the shift from daily to monthly dynamics. The model can thus use this information to appropriately scale or transform the input daily returns, aligning them with the temporal context of the monthly output.

On the whole, the results show that the TFT extracts only a subset of key inputs that intuitively play a significant role in the predictions. We acknowledge that the presented variable importance might not be absolutely indicative of their actual impact. For instance, the transient characteristics inherent in certain features might not be fully captured in this representation, thereby affecting the perceived importance of these variables.

4.1.2 Visualizing Persistent Temporal Patterns

We aggregate the attention scores $\alpha(t, i, \tau)$ from equation (3.14) across all timesteps and stocks, obtaining the average attention score for all positional indices i and forecast horizons τ, and plot the score in Figure 4.1. The plot shows the most prominent seasonality patterns similar to methods such as autocorrelation plots and time signal decompositions. However, studying the attention weights of the TFT has extra advantages: We can confirm that our model captures the apparent seasonal dynamics of our sequences. Our model may also reveal hidden patterns because the attention weights of the input windows consider all past inputs in the dataset. Moreover, an autocorrelation plot refers to one particular sequence, while the attention weights here focus on the impact of each timestep by considering all covariates and time series.

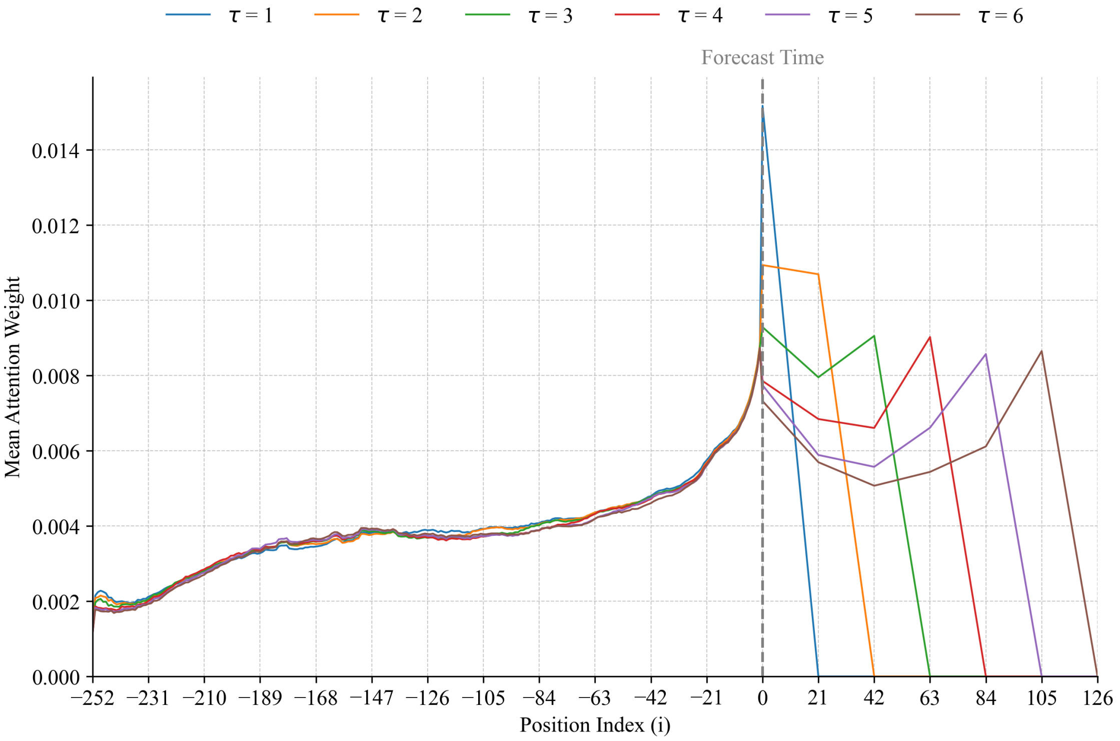

Figure 4.1 Average attention weights $\alpha(t, i, \tau)$, for forecasts at various horizons τ.

As depicted in Figure 4.1, the average attention of past inputs exhibits an inverse S-shaped curve, characterized by rapid increases at both the beginning and end of the sequence. The temporal pattern suggests intriguing implications regarding the model's perceived importance of different time horizons of data for predicting future stock returns.

The persistent temporal pattern of past inputs across different forecasting horizons τ indicates that the model consistently values the same temporal segments, irrespective of how far into the future it is predicting. This consistency might suggest that the model deems certain time-dependent effects, such as momentum or reversals, important across all forecast horizons. However, it is worth noting that although the model might use the same temporal information for different forecast horizons, it does not necessarily mean that it weighs the information in the same way for each forecast. The model could be capturing other time-varying factors, such as changes in market volatility or investor sentiment, that could influence stock returns differently over different forecast horizons. It might also be learning complex interactions and nonlinear relationships between the inputs and outputs that vary with the forecast horizon in the final layer of the model's design. Unfortunately, this is beyond the scope of the model's intrinsic interpretability.

The limited attention given to distant past data aligns with aspects of EMH, particularly the weak form which posits that all past prices are fully reflected in current prices, and thus, historical returns should have little predictive power for future returns.

No strong persistent patterns are observed for the past inputs between $i \approx -147$ and $i \approx -63$ (i.e. seven to three months in the past) — attention weights are equally distributed across all positions on average. This resembles a moving average filter at the feature level, and given the high degree of randomness associated with the stock movement process — is useful in extracting the trend over this period by smoothing out high-frequency noise.

The attention spikes during the last three months of the past temporal segment. The result is consistent with the findings from previous studies such as Fischer and Krauss (2018), Krauss et al. (2017), and Moritz and Zimmermann (2016), which discover that the latest returns are the most important variables for their ML models. The model's heightened attention to recent data may capture the impact of more transient, high-frequency factors such as liquidity, proxied by the bid-ask spread percentage, and recent shifts in market sentiment or risk, which can exert significant influence on stock returns in the future. Moreover, the relatively flat attention weights in the middle past temporal segment, coupled with a distinct spike in the most recent data points, suggest a nuanced interplay of market dynamics. This pattern possibly captures the effects of mean-reversion or reversals, where the trends in past price movements correct themselves, while concurrently emphasizing the crucial role of recent market information in shaping future stock returns.

TFT giving increased attention to the most recent data points also aligns with the weak form of EMH. This suggests that the model is learning that the most recent stock movement, which should theoretically encapsulate all available information up to that point, is the most important in predicting future returns.

The model's overall attention to past data points could be seen as inconsistent with a strict interpretation of EMH. However, it should be noted that the ability of the model to potentially exploit these patterns does not necessarily contradict the EMH, as real-world markets have various frictions that may cause slight inefficiencies, and the model may be capturing risk factors that are rewarded with higher returns or behavioral biases that can lead to predictable patterns in returns. Nonetheless, it does suggest that the model perceives the stock market as exhibiting some degree of predictability, in contrast to a strict interpretation of the EMH.

In the future temporal segment of Figure 4.1 we observe for all forecast horizons τ a decline in the degree of attention given to older future information, contrasted with an increase for more recent future information. This temporal pattern aligns intuitively with our expectations, suggesting that the model is learning to prioritize more recent future data as it is potentially more indicative of the near-future state of the stock returns.

4.1.3 Identifying Significant Regimes

Identifying distinct market states, and understanding how an investor should behave in each regime, forms a cornerstone of developing a profitable trading strategy. Equation (3.15) allows us to analyze significant shifts in temporal dynamics for a single entity, so we select the S&P 500 index, treating it as a stand-in for all individual stocks within our dataset, and input it into our trained TFT. The decision to use the S&P 500 index as a proxy for all individual constituents is a strategic one. The S&P 500 diversity is instrumental in capturing the varying responses of different stocks and industries during different market regimes. Indeed, different sectors may react differently to the same market conditions — some may thrive while others struggle. By treating the entire S&P 500 index as a single stock, we aim to encapsulate these varied reactions, creating a composite representation that, without loss of generality, broadly reflects the behavior of all constituents. This approach gives us a comprehensive view of how the market, as a whole, transitions through different regimes, and how our model responds to these changes.

To illustrate the model's event detection, we compare the attention vector of the S&P 500 index for each forecast date with its average attention vector across all timesteps. In Figure 4.2, significant deviations in attention patterns can be observed around recognizable periods of high historical volatility — corresponding to the peaks observed in $\mathrm{dist}(t)$. Firstly, we observe a significant regime from 2001 to 2003, which covers the tail end of the dot-com bubble burst and the subsequent market recovery, as well as the events of September 11, 2001, causing significant fluctuations in the stock market. Secondly, the deviations peak in late 2008 to 2010, corresponding to the great recession triggered by the 2008 financial crisis. Then, the model highlights late 2011 to late 2012, which was a period marked by the European sovereign debt crisis, which had a global impact on financial markets. Finally, we have a discovered event in the years surrounding 2016. This period witnessed significant political events that caused market volatility, including the Brexit vote and the U.S. Presidential election.

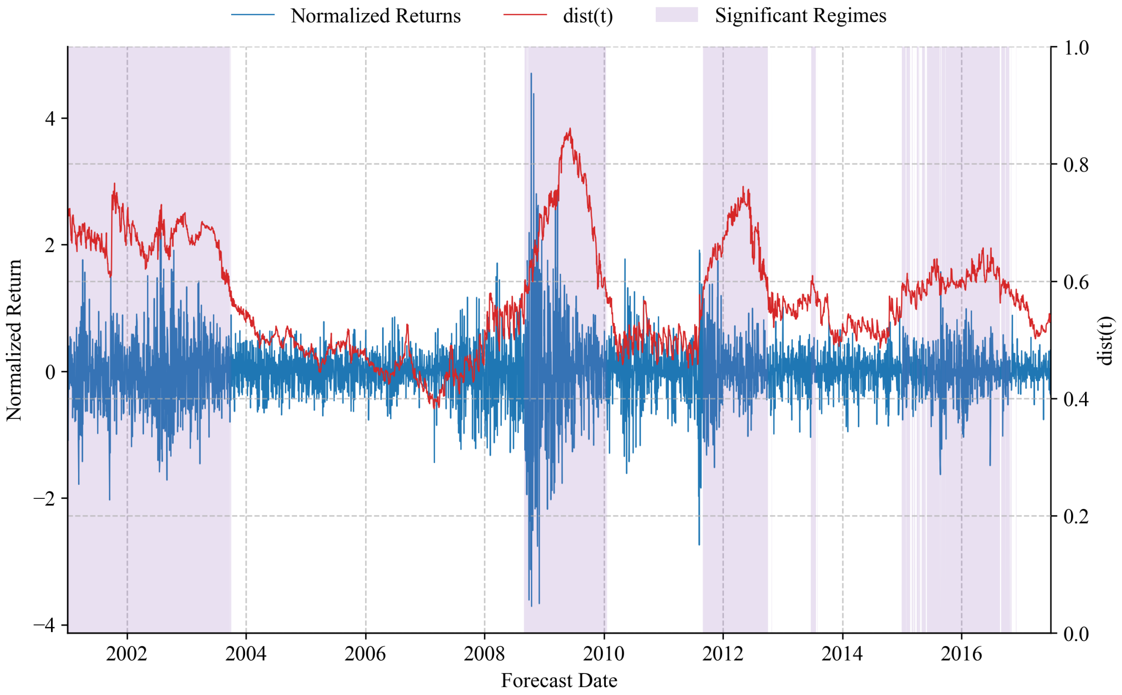

Figure 4.2 Distance between the average attention vector across all timesteps, $\overline{\boldsymbol{\alpha}}(t)$, and the attention vector of a given forecast date, $\boldsymbol{\alpha}(t, \tau)$, aggregated over all forecast horizons τ, plotted against the normalized returns of the S&P 500 index. We use a threshold of $\mathrm{dist}(t) > 0.6$ to denote significant regimes, as highlighted in purple.

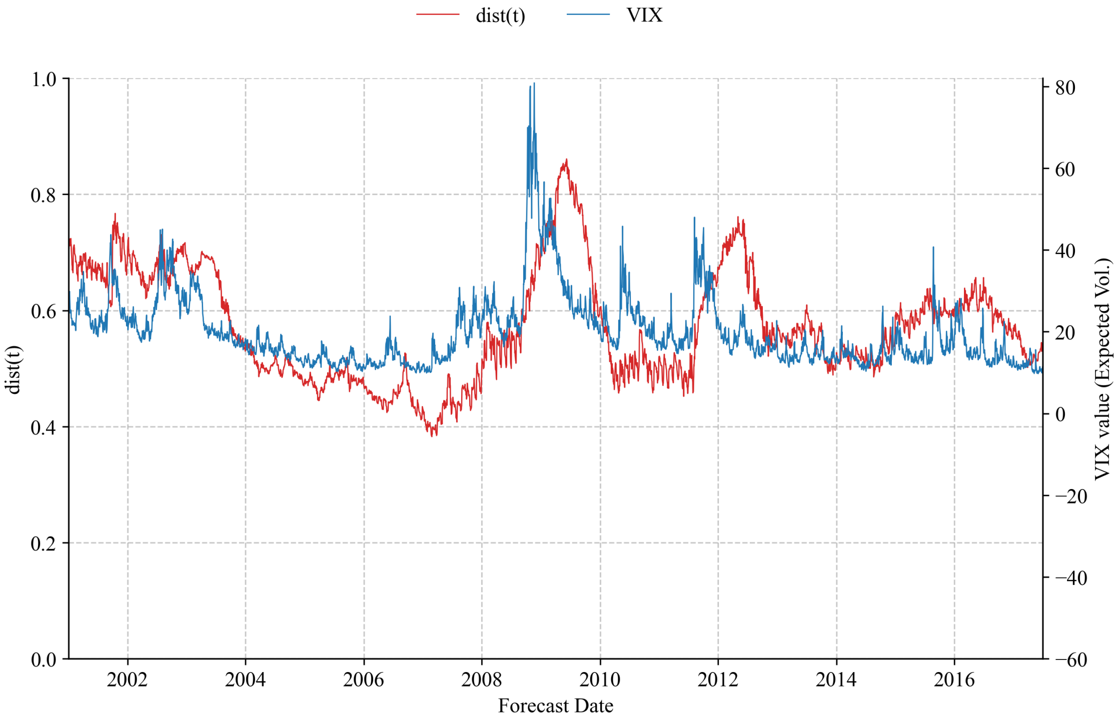

Figure 4.3 Distance between the average attention vector across all timesteps, $\overline{\boldsymbol{\alpha}}(t)$, and the attention vector of a given forecast date, $\boldsymbol{\alpha}(t, \tau)$, aggregated over all forecast horizons τ, plotted against the index value of the VIX.

The dist(t) in Figure 4.2 looks like a volatility plot. Indeed, the model's perception of changing market states bears a remarkable resemblance to the pattern exhibited by the VIX, albeit with a slight rightward shift, as seen in Figure 4.3. This intriguing correlation suggests that the model's interpretation of changing market states is substantially aligned with the VIX, which is a real-time index representing the market's expectations for the 30-day forward-looking volatility. The VIX has been shown to be an effective predictor of future realized volatility (see e.g Blair et al., 2010; Christensen and Prabhala, 1998; Christensen and Hansen, 2002) and is often termed the "fear gauge" of the market, as it tends to rise during periods of financial turbulence or uncertainty when market participants anticipate large swings in the stock market. Conversely, during periods of relative calm and economic stability, the VIX typically decreases as expected volatility is low. Therefore, the rightward shift could be interpreted as the model reflecting the inherent lag between the market's expectations (captured by the VIX) and when these expectations are realized in the stock returns data (captured by dist(t)). Alternatively, the shift might indicate the model's latency in recognizing and adapting to changing market conditions. Nonetheless, this alignment underscores the model's ability to perceive and respond to market volatility effectively, mirroring the market's expected future volatility encapsulated in the VIX.

In order to provide a detailed analysis of our TFT's response to different market states, we examine the attention vector of the S&P 500 index in relation to the normalized returns over the preceding year and the subsequent semester for two distinct dates. By comparing these contrasting periods, we aim to gain insights into the model's adaptability to varying market dynamics. The relationship between the attention vector and the daily returns over these periods can provide valuable information about how the model adjusts its focus to accommodate changes in market volatility and regime shifts. Furthermore, this analysis can help us understand how the model prioritizes certain temporal effects such as momentum and reversals in response to these market changes.

For this analysis, we present two specific dates, June 1st, 2006 and September 1st, 2009, which are located before and within the period of the 2008 financial crisis, as seen in Figure 4.2. From the upper plot of Figure 4.4, we observe that June 1st, 2006, represents a period of relative market stability, with daily returns exhibiting minor fluctuations indicative of a steady market state.

Conversely, September 1st, 2009, depicted in Figure 4.5, is characterized by highly variable daily returns for about eight months, suggestive of a turbulent market regime. This period of high volatility transitions into a phase of reduced fluctuations during the last four months leading up to the selected date.

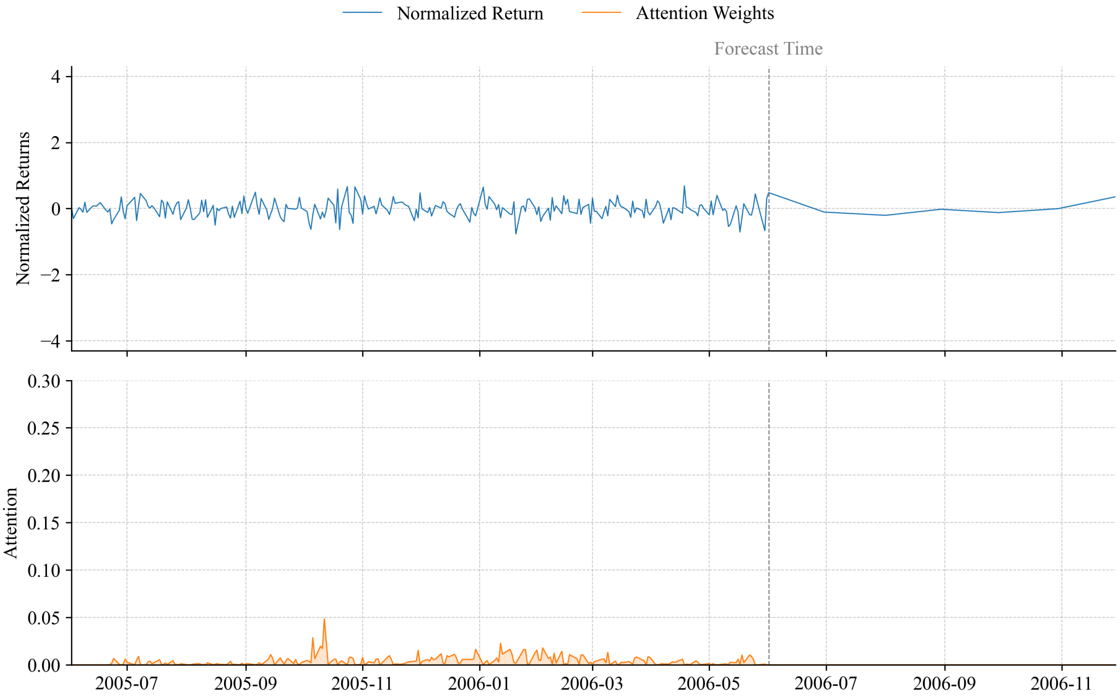

Figure 4.4 The normalized return vector and attention vector $\boldsymbol{\alpha}(t, i, 1)$ of the S&P 500 index at forecast date June 1st, 2006. The returns indicate a period with no significant regime changes.

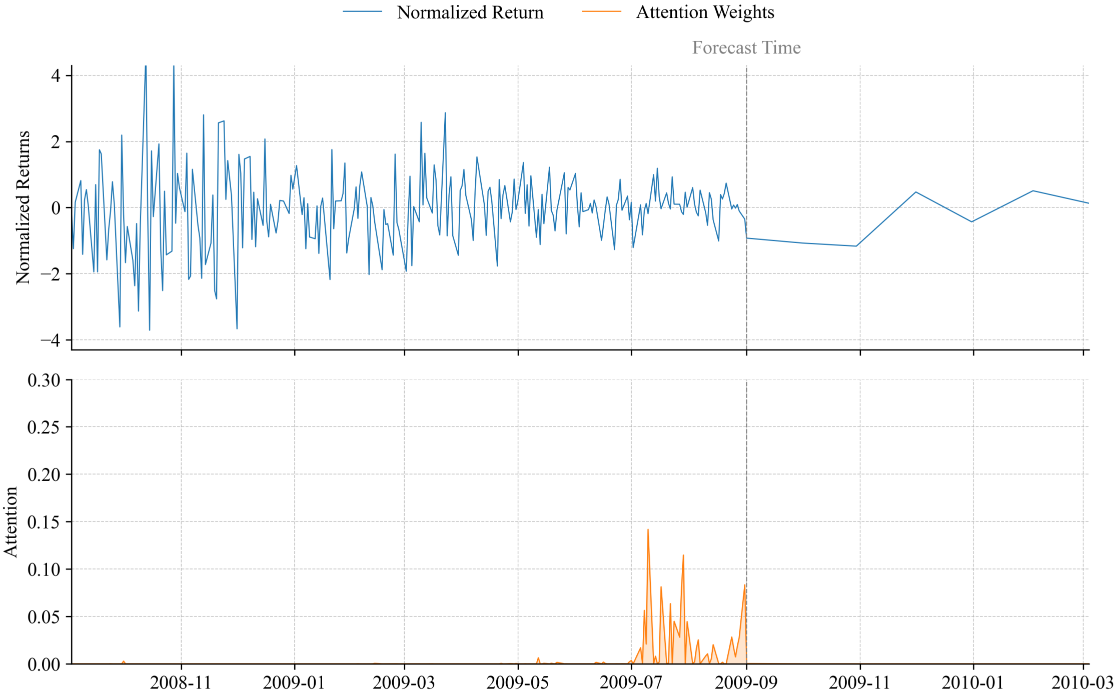

Figure 4.5 The normalized return vector and attention vector $\boldsymbol{\alpha}(t, i, 1)$ of the S&P 500 index at forecast date September 1st, 2009. The returns indicate a period with a significant regime change.

From the bottom plots of Figure 4.4 and Figure 4.5, we can see that TFT appears to alter its behavior between regimes — suggesting differences in temporal dynamics learned in each of these cases. In periods where no significant regime changes are detected, the TFT places its attention rather evenly across all past inputs. This behavior suggests that the model is treating the data as part of a more stable, consistent market environment, where long-term trends and patterns may hold more predictive value. Here, the attention mechanism may be acting as a filter, smoothing out the noise and focusing on the underlying trend across the entire lookback window. This strategy is analogous to a momentum-based approach, whereby the prevailing trend is used as a basis for future predictions. Conversely, during periods of changing regimes, the TFT concentrates its attention predominantly within the current market state, effectively disregarding inputs from past timesteps outside this regime. This indicates that the model is sensitive to shifts in the market, adjusting its focus to prioritize recent and similar past information, which may be more indicative of the near-future state. Such a strategy suggests a belief in the persistence of the current market condition, with older data deemed less relevant in the face of recent changes. The marked difference in the model's attention distribution between stable and volatile periods underscores the TFT's trained preference in handling the complexities and uncertainties inherent in financial markets to create accurate predictions.

The combined insight of Figure 4.2, Figure 4.4, and Figure 4.5 casts a new light on the observations made in Figure 4.1. The skyrocketing average attention at the most recent timesteps may indicate that the market states are on average changing constantly, and thus the model finds it beneficial to revert most of the attention to recent timesteps. As such, these findings further substantiate the nuanced, ever-changing nature of stock market behavior and the importance of adaptive, flexible forecasting tools.

4.2 Post-model Analysis

Seeking to identify potential relationships between past and future performance that may have eluded our initial evaluation of the TFT model's interpretable components, we undertake a thorough post-model analysis of the model's forecasts. First, we probe the model's predictions, with a particular focus on discerning seasonal patterns and their plausible impact on stock returns. Second, we investigate whether contrasting patterns exist between stocks that are predicted to overperform or underperform in relation to the entire stock universe. This post-model analysis enhances our understanding of the dynamic interplay between various elements affecting stock returns, ultimately providing a more granular perspective of the underlying market dynamics.

4.2.1 Seasonal Patterns

To investigate the possible seasonal patterns based on the highlighted importance of monthly and quarterly date-related variables in Table 4.1, we analyze the predictions produced by the TFT. We focus specifically on the $\tau = 1$ predictions, i.e. one-month ahead return forecasts. We proceed to calculate the predicted return on each forecast date, averaged across all stocks. While the entirety of our findings over the in-sample period is seen in the top plot of Figure 4.6, we also focus on a calm sub-period (evidenced by Figure 4.2) in the bottom plot for the sake of clarity and enhanced readability. We observe that our TFT model predicts an intriguing seasonal pattern in stock returns: the predicted returns are on average higher in winter months (Nov. to Apr.) compared to summer months (May. to Oct.).

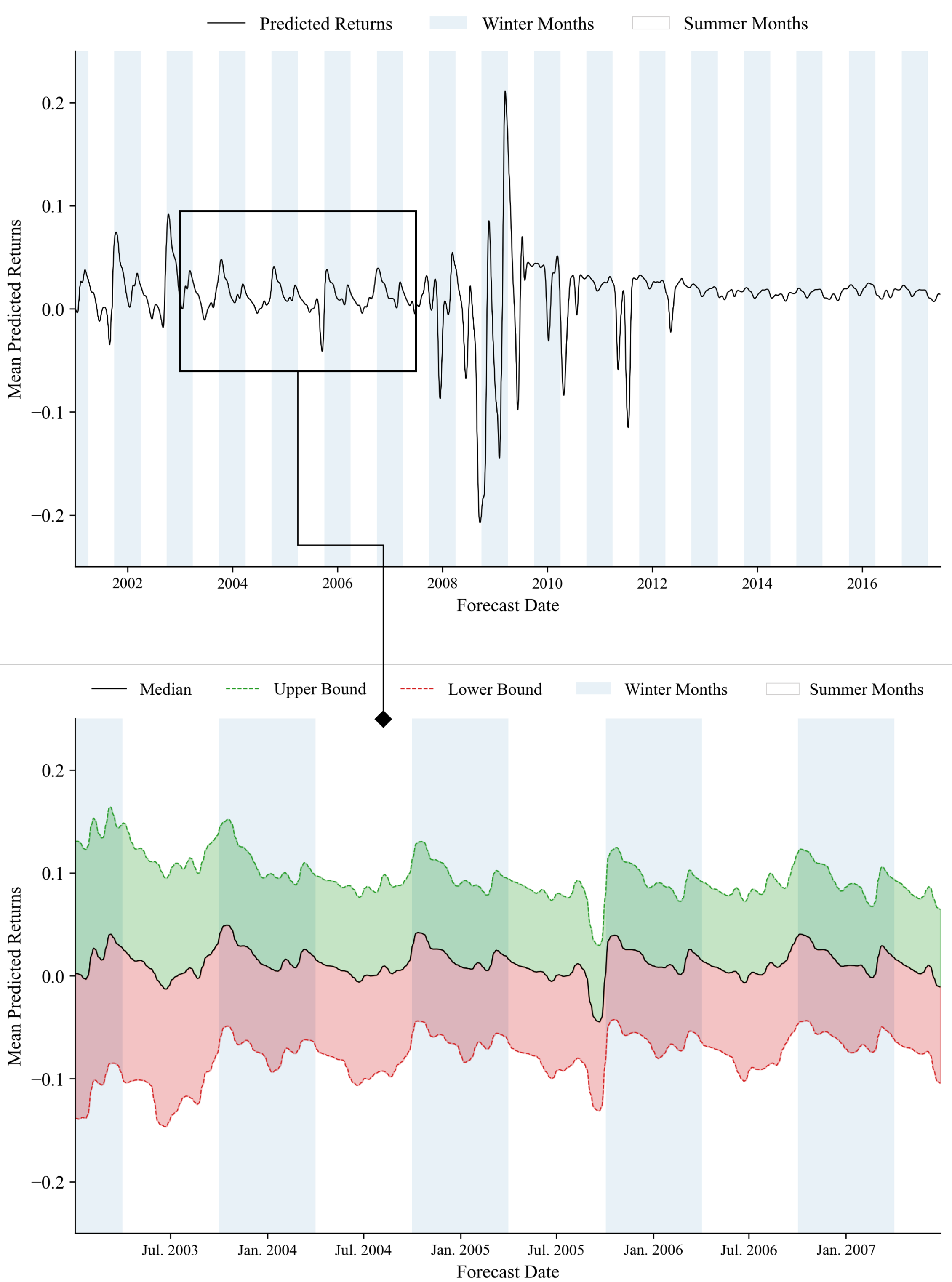

Figure 4.6 The median (P50), upper bound (P90), and lower bound (P10) one-month-ahead predicted returns on each forecast date in the entire in-sample period, averaged across all stocks. We focus on the calm period between 2003 and 2007. The higher highs and lower lows in winter months in most years suggest that the model finds strong evidence of seasonal effects in stock returns. The magnitude of this seasonal anomaly appears to be decreasing in more recent years.

Researchers have extensively scrutinized such seasonal patterns to understand their influence on market trends and to strategize investing decisions. Our finding echoes one of the most recognized calendar anomalies: the "Halloween effect". This phenomenon, first observed[1] by Bouman and Jacobsen (2002), asserts that stock market returns are generally higher from November through April than in other months of the year. Bouman and Jacobsen (2002) note that this difference in returns cannot be explained by risk factors or the January effect (see Rozeff and Kinney, 1976), and is most likely explained by behavioral finance. The Halloween effect has subsequently been analyzed extensively, and most studies find significant Halloween effects in several markets and time periods (see, for instance, Guo et al., 2014; Jacobsen and Visaltanachoti, 2009; Lean, 2011; Plastun et al., 2020). Our model's predictions align with these historical observations, reaffirming the persistence of this seasonal effect in the U.S. stock market.

4.2.2 Selecting Top and Flop Stocks

We leverage the predicted percentile stock returns — P10, P50, and P90 — to identify two groups of stocks: those predicted to perform well (which we term top stocks), and those forecasted to underperform (referred to as flop stocks). This analysis serves two objectives. Firstly, by comparing these subsets with the entire stock universe, we can observe any potential differing patterns, leading to an enhanced understanding of the behavior of well-performing and underperforming stocks. Secondly, this approach introduces a method that allows us to trade based on the outputs of the TFT.

Our stock selection must acknowledge that the future is inherently uncertain and that different stocks come with varying degrees of risk and potential return. Therefore, we find it valuable to introduce a method for determining a risk-adjusted return for each stock's forecasted return path, thus assigning more importance to certain outcomes based on the perceived outlook and uncertainty surrounding a particular stock. This method of weighting serves two primary purposes. Firstly, it provides a systematic approach for balancing risk and reward, which is crucial to successful stock selection. Secondly, it acknowledges and incorporates the inherent uncertainty present in stock return forecasting.

First, we utilize the upper bound as a baseline, as this particular return path registered the lowest validation quantile loss during the model's training phase. Then, we adjust the baseline returns by scaling with the ratio seen in equation (4.1). Specifically, the ratio is the upside potential, given by the difference between the median and upper bound, divided by the downside risk, i.e. the difference between the median and lower bound:

$$\text{Risk-adjusted Return} = \text{P90} \times \left(\frac{|\text{P50} - \text{P90}|}{|\text{P50} - \text{P10}|} \right) \tag{4.1}$$

Next, we accumulate the risk-adjusted returns and calculate the equivalent predicted monthly return over all forecast horizons τ. We identify top (flop) stocks by maximizing (minimizing) risk-adjusted monthly returns and storing the respective optimal holding time, selecting the best decile. Therefore, on each forecast date, top stocks are those that possess large returns and promising upside potential relative to the downside risk, indicating a higher likelihood of outperforming the market. Conversely, flop stocks are those with smaller returns and less upside potential relative to the downside risk, suggesting that they are more likely to underperform in the market.

We repeat the analysis of subsection 4.1.1 and 4.1.2 on the subsets of top and flop stocks in results that are not shown, and find no notable differences in variable importance or temporal patterns, indicating that the presented patterns are applicable to specifically selecting over- and underperformers.

4.3 Building A Novel Momentum Strategy

Our findings underscore the significance of adaptive and flexible forecasting tools in navigating the intricate dynamics of the financial markets. According to C. Lee and Swaminathan (2000), stock

[1] It is worth noting that some variation of this effect has been around for quite a long time. The axiom "Sell in May, go away" so often coined in financial media was also repeated over the last two centuries.

prices typically oscillate around their fair value. In such a world, the success of any momentum strategy hinges on decision rules that are aligned with the periodicity of the pricing cycles. As momentum trading signals are grounded in recent price movements, they will always be late in identifying future winners and losers. However, the more successful momentum strategies will be those based on decision rules that identify winners (losers) early in their up-(down) cycle and reverse these positions with optimal timing. Thus we intend to create a dynamic momentum trading strategy, adeptly adjusting and reorienting itself depending on the observed and future expected market state. As a base to further build upon, we use the CSMOM approach of Jegadeesh and Titman (1993), and go long (short) the stocks that exhibit the highest (lowest) returns over a defined formation period and hold the positions over a specified holding period.

Our approach for defining formation and holding periods is guided by filtration of the important variables in Table 4.1. Some of the notable implications of this are the relative irrelevance of the industry, size, and liquidity as deciding factors for future returns, which have previously been found to affect momentum strategies. In stark contrast, a strong emphasis is placed on past price performance, consistent with the principles of momentum investing. Finally, our proposed strategy incorporates significant adaptations based on the month or quarter of the year as these factors have been shown to have a great influence on the model's return forecasts.

4.3.1 Dynamic Formation Periods

We begin by progressively fine-tuning the formation period. The first aspect that informs our decision is our observation from Figure 4.1. While the plot shows that the average attention given to the past year of timesteps is a non-decreasing function and never hits zero, indicating the relevance of a year's historical data, it also underscores that attention to distant past data is rather limited. The lower attention values seen prior to the seven-month mark suggest that the most influential data is largely confined to the previous seven months. Consequently, we initiate our strategy with a baseline formation period of seven months.

We then turn to Figure 4.2, which reveals that our model adeptly recognizes major shifts in market state and incorporates these observations into its predictions. We find that the VIX echoes this pattern of the model, as depicted in Figure 4.3, presenting an opportunity to leverage the VIX as an indicator of market state changes. To implement this, we draw upon the findings of Banerjee et al. (2007), which suggest that VIX levels provide more reliable insights than innovations. Accordingly, we define a VIX value threshold to denote highly volatile market states. This threshold is set at the third quartile of VIX values during the in-sample period, corresponding to a VIX value of approximately 24. We classify months where the VIX value exceeds the threshold as high volatility states, whereas all other months are classified as normal periods.

Further analysis of the model's behavior unveils more nuanced characteristics. As evidenced in Figure 4.4, the model utilizes attention to filter out high-frequency noise and extracts the prevailing trend over the entire period in less volatile times. However, in periods of market turmoil, as shown in Figure 4.5, the model effectively discards information predating the regime change, hinting at a potential strategy to reduce the formation period during periods of high volatility.

Lastly, the skyrocketing average attention during the three most recent months observed in Figure 4.1 intimates that market states are constantly changing. As these shifts occur, the model progressively prioritizes the trend from the past three months, peaking in the most recent month of data. Consequently, we opt to tighten the formation period to one month when volatility escalates.

4.3.2 Dynamic Holding Periods

Following our formation period modifications, we turn our focus to the holding period and ways to optimize it in accordance with observed patterns and market indicators. Our first guidepost in this direction is the TFT's inference of a general trend of higher stock returns during winter compared to summer, seen in Figure 4.6. This observation points towards a starting point for our holding period dynamics, rooted in the recognized Halloween effect in stock returns. In an attempt to capture the Halloween effect, strategic holding periods are selected as baselines: 1 (6) months for the long (short) portfolio from November to April, and a reverse strategy of 6 (1) months from May to October. A shorter holding period allows for more active trading, facilitating the capture of frequent gains when the market is projected to trend favorably. Conversely, extending the holding period in anticipation of less favorable market conditions provides an opportunity to ride out temporary downturns. This reduces the risk of incurring losses due to premature selling, particularly when the market is forecasted to move counter to the long/short portfolios' orientation.

We note that while seasonality studies provide valuable insights into seasonal effects in stock markets, their findings should be considered with caution. Seasonal effects are highly debated, subject to change, and may not be present in all time periods or market conditions. For example, some studies claim to disprove the Halloween effect (see Maberly and Pierce, 2004), only for other studies to subsequently emerge that challenge and rebut these initial refutations (see Witte, 2010). Moreover, Plastun et al. (2020) reveal that in the U.S. stock market and other developed markets, the Halloween effect first appeared in the middle of the 20th century. However, in most recent years the effect is less prevalent which is in line with our findings in Figure 4.6, where the magnitude of the Halloween effect appears to be declining in more recent years. Furthermore, the period of heightened volatility following 2008 seems to disrupt the conventional seasonal pattern, indicating the potential influence of significant market upheavals on historical trends.

Thus, a strategy solely based on seasonal effects might be overly simplistic and prone to failure in less predictable circumstances. We, therefore, aim to bolster the strategy with an additional dimension — again using the VIX. This market fear gauge can provide us with the required sensitivity to upcoming market movements that could disrupt the anticipated seasonal pattern.

In support of our approach, we draw upon the findings of Banerjee et al. (2007). They demonstrate that VIX has a strong predictive ability for future stock returns, and find a positive correlation between S&P 500 future performance and VIX levels. The results are strongest for two-month returns, probably because they estimate the average time for mean reversion of VIX long-term volatility to be 44.1 trading days, which is very close to two calendar months. The findings are not surprising and consistent with prior discoveries related to VIX and future returns, and the notion of a significant negative volatility risk premium (Ang, Hodrick, et al., 2006; Bakshi and Kapadia, 2003; Bouchaud et al., 2001; Coval and Shumway, 2001). Consequently, we override our seasonal holding periods when the VIX indicates high volatility states, and hold the long and short portfolios for two months. We implicitly assume that any transitory fluctuations in stock returns during volatile states may be independent of the seasonal effect. The specific timeframe is selected to fully capitalize on the predictive influence of the VIX on future returns, harnessing the peak of its effect before it reverts to its mean value.

4.3.3 Reversals

So far, we have outlined how our strategy adapts the formation and holding periods to accommodate for seasons and heightened market volatility. Under volatile conditions, the formation and holding periods are shortened to one and two months, respectively. This alteration lays the groundwork to exploit another notable aspect of market dynamics — short-term reversals in stock returns.

The short-term reversal phenomenon is well-documented in the financial literature. Jegadeesh (1990) and Lehmann (1990) provide evidence that stocks which have performed well or poorly in the recent past are likely to undergo significant return reversals in the near future. While the cause of these reversals remains subject to debate, a possible explanation is market overreactions to news (particularly adverse news, see e.g. Nam et al., 2001). On the other hand, Jegadeesh and Titman (1993) suggest these reversals might stem from short-term price pressures and liquidity constraints. We note that these two explanations are not necessarily mutually exclusive, as investor sentiment and liquidity are inherently intertwined (Baker and Stein, 2004).

Our own observations, as represented in Figure 4.1, also hint at the significance of these short-term reversal effects. As noted, the plot exhibits relatively flat attention weights in the middle past temporal segment, while the most recent data points show a sharp spike. This pattern might be capturing the interplay of mean-reversion or reversals, where past price trends correct themselves using the most recent information.

Drawing on these insights, and considering the shortened formation and holding periods during periods of heightened volatility, we adjust our strategy to take advantage of these short-term reversal effects. Specifically, we reverse our approach by buying past underperformers and short-selling past overperformers. The baseline strategy thus transforms into a Cross-sectional Reversal (CSREV) strategy. This adjustment rests on the expectation of a reversal in the recent trend, which could lead to a turnaround in the relative performance of these stocks.

4.3.4 The Adaptive Momentum Strategy

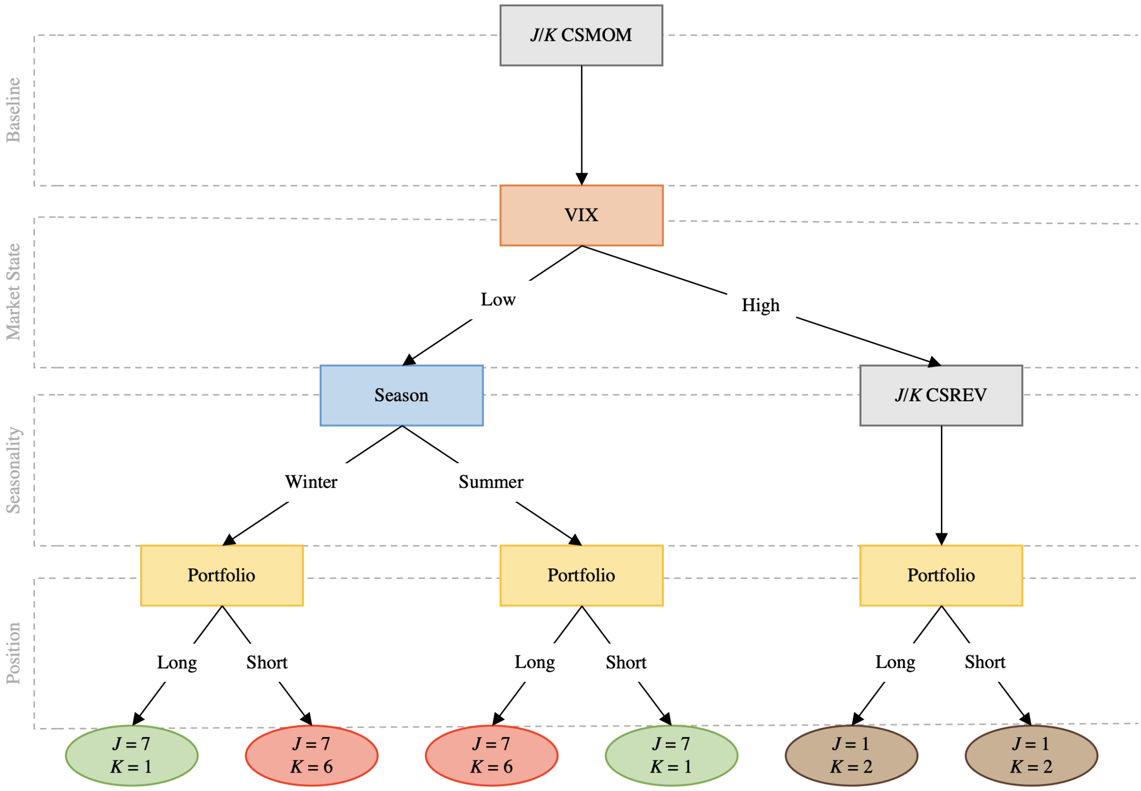

Figure 4.7 A step-by-step decision-rule-based momentum and reversal strategy that dynamically adjusts the formation period (J) and holding period (K), grounded in the signals generated by the TFT. Rectangles and ovals denote decisions and outcomes respectively. Equal colors represent the same decisions/outcomes.

The resulting strategy is summarized in Figure 4.7 and combines principles of the well-known EMH with behavioral finance. The strategy maintains aspects of the EMH through the dynamic reduction of the formation period, aligning with the theory's assertion that markets swiftly integrate all available information into stock prices. Simultaneously, it employs aspects of behavioral finance by adjusting the holding period based on market participants' anticipations and leveraging reversals to acknowledge the behavioral bias towards overreaction. Such a strategy holds true under the Adaptive Market Hypothesis (AMH) assumption (Lo, 2004). Andrew Lo, the hypothesis's founder, believes that people are mainly rational, but can sometimes overreact during periods of heightened market volatility. Hence, our strategy embraces the rationality of the EMH while acknowledging the behavioral nuances of real-world investors. We refer to this strategy as Adaptive Momentum (AMOM).

4.4 Performance Review

We shift focus to out-of-sample testing of four distinct trading strategies: TFT, AMOM, CSMOM and TSMOM. Both the CSMOM and TSMOM strategies are implemented as outlined in subsection 3.5.2. Our TFT and AMOM strategies offer unique approaches based on the Temporal Fusion Transformer model. Specifically, the TFT strategy capitalizes on forecasts made by the TFT model. On the first day of each month, this strategy buys the top stocks and short-sells the flop stocks, and holds these positions for the predicted optimal holding time, as detailed in subsection 4.2.2. Our AMOM strategy, on the other hand, leans on interpretations of the TFT model. In line with CSMOM, on the first day of each month, AMOM buys the top decile of past performers and short-sells the bottom decile. However, the formation and holding times of these positions are governed by the decision rules outlined in Section 4.3.

To assess the performance of each strategy, we employ the evaluation methods described in subsection 3.5.3. The results from this comparative analysis are found in Table 4.2. The upcoming analysis of the results aims to highlight the strengths and weaknesses of each strategy and provide insights into the best practices for implementing momentum trading strategies.

4.4.1 Profitability

Examining the profitability measures of the four strategies, a few key observations emerge. The TFT strategy stands out in particular, generating the highest average returns in the long and short portfolios, resulting in an expected total return of 2.66% (2.37%) per month before (after) transaction costs. Curiously, all the strategies yield negative returns in the short portfolio. This pattern is consistent with findings reported by Bird et al. (2017), who tested various implementations of momentum strategies and found that short portfolios typically yield negative returns across multiple markets. Our joint findings may be attributed to several factors that are inherent to short-selling. Short-selling is a risky strategy that entails potential unlimited losses. Furthermore, stocks have an overall historical tendency to rise over time, which makes profiting from short-selling more challenging. Nevertheless, the TFT strategy emerges superior by virtue of its ability to minimize these losses, making it the most successful strategy in managing short positions.

The superiority of TFT is to be expected, given that the TFT strategy leverages a sophisticated ML based model to capture complex relationships and patterns between past and future performance. Conversely, the other strategies primarily rely on simpler decision rules to draw these dependencies. More interestingly, the results underline that leaning on the interpretations of the TFT to build a straightforward rule-based trading strategy proves successful in generating positive returns. Indeed, the AMOM strategy ranks second in terms of profitability characteristics, with an expected total return of 1.62% (1.51%) per month before (after) transaction costs.

Moreover, when considering the average total returns, only TFT and AMOM distinguish themselves by producing returns significantly greater than zero. Specifically, TFT posts a t-statistic at 2.20, thereby demonstrating its statistical significance at the 2% level. Concurrently, AMOM delivers a t-statistic at 2.11, indicating its statistical significance at the 3% level. This denotes that both TFT and AMOM are capable of generating more reliable positive returns, and they are statistically more likely to outperform the other strategies and the market in terms of profitability.

4.4.2 Risk

Shifting the lens to a more nuanced view of profitability, we analyze the distribution of returns and the risk metrics to understand the deeper layers of risk and return performance across the strategies. TFT is once again a notable standout, with the highest minimum return, lowest downside deviation, impressive quartile and maximum returns, and a healthy median return. Moreover, its return distribution, characterized by a pronounced positive skewness and substantial excess kurtosis, points towards sizable gains overshadowing the rare instances of substantial losses. This reinforces our initial observations of TFT's superior profitability, demonstrating its consistently

Table 4.2
Financial performance characteristics of the portfolios, before and after transaction costs (TC) for the TFT and our proposed strategy (AMOM), compared to the approach of Jegadeesh and Titman (1993) (CSMOM), Moskowitz, Ooi, et al. (2012) (TSMOM), and the general market (MKT) from July 2020 to December 2022. S&P 500 index represents the market. The profitability panel depicts monthly return characteristics. The risk panel shows monthly risk characteristics. The performance ratio panel displays annualized risk-return metrics. Best values are highlighted in bold. t-statistics and p-values are reported below the total expected return and derived from one-tailed t-tests, testing the H_0 that the mean total return is less than or equal to zero.

	Profitability											Risk			Performance ratios	
Strategy	$\mathbb{E}[R]$ (long)	$\mathbb{E}[R]$ (short)	$\mathbb{E}[R]$ (total)	Min	Q1	Median	Q3	Max	Pos. perc.	Skew.	Ex. kurt.	St. dev.	Ds. dev.	MDD	Sharpe ratio	Sortino ratio
w/o TC																
TFT	**0.0336**	**-0.0070**	**0.0266** **(2.1969)	**-0.0443**	-0.0132	0.0154	0.0479	0.1905	0.6000	1.3771	1.5283	0.0605	**0.0141**	0.1002	**1.5164**	**6.4929**
AMOM	0.0271	-0.0108	0.0162 **(2.1110)	-0.0570	**-0.0093**	0.0118	0.0427	0.1121	**0.7200**	0.4599	0.2661	**0.0385**	0.0160	**0.0570**	1.4536	3.4956
CSMOM	0.0257	-0.0193	0.0063 (0.4242)	-0.1387	-0.0161	**0.0286**	0.0433	**0.1974**	0.6400	-0.0386	0.4121	0.0745	0.0468	0.1758	0.2892	0.4606
TSMOM	0.0249	-0.0217	0.0032 (0.2874)	-0.1103	-0.0095	0.0193	0.0273	0.1011	0.6800	-0.4689	-0.2995	0.0551	0.0357	0.1743	0.1923	0.2968
MKT	0.0124	-	0.0124 (1.2411)	-0.0859	-0.0218	0.0198	**0.0556**	0.1064	0.6400	-0.2132	-0.8330	0.0500	0.0223	0.2025	0.8524	1.9111
w/ TC																
TFT	**0.0329**	**-0.0093**	**0.0237** **(1.9669)	**-0.0461**	-0.0157	0.0125	0.0461	0.1877	0.6000	1.3665	1.4758	0.0602	**0.0138**	0.1167	**1.3570**	**5.9243**
AMOM	0.0265	-0.0113	0.0151 **(1.9686)	-0.0580	**-0.0103**	0.0106	0.0416	0.1109	**0.6800**	0.4594	0.2634	**0.0384**	0.0175	**0.0580**	1.3548	2.9694
CSMOM	0.0247	-0.0204	0.0043 (0.2885)	-0.1409	-0.0181	**0.0266**	0.0413	**0.1953**	0.6400	-0.0406	0.4101	0.0745	0.0468	0.1877	0.1952	0.3107
TSMOM	0.0232	-0.0232	0.0000 (-0.0028)	-0.1136	-0.0131	0.0156	0.0253	0.0979	0.6400	-0.4763	-0.3024	0.0552	0.0393	0.1855	-0.0087	-0.0122
MKT	0.0114	-	0.0114 (1.1411)	-0.0869	-0.0228	0.0188	**0.0546**	0.1053	0.6400	-0.2132	-0.8330	0.0500	0.0223	0.2074	0.7831	1.7558

Note: *$p < 10\%$, **$p < 5\%$, ***$p < 1\%$

strong performance across different market conditions.

The return distribution of AMOM is remarkably similar to that of the TFT strategy. AMOM demonstrates resilience with an impressive minimum, quartile, and median return, paralleling the strengths of the TFT strategy. This resemblance indicates the exciting possibility that AMOM can effectively mimic the TFT strategy, implying that we have successfully deciphered and implemented some of the most influential dependencies that the TFT model responds to. Still, there are certain disparities between the AMOM and TFT strategies. AMOM has a lower maximum return and a higher percentage of positive return, standing at an impressive 72% (68%) before (after) transaction costs. This results in lower variability in its returns, as evidenced by a lower standard deviation, similar downside deviation, and a reduced maximum drawdown compared to the TFT strategy. These differences might hint that the AMOM strategy is more risk-averse, which may be advantageous for investors who prefer a more conservative approach towards risk management, valuing stability and predictability over the allure of extreme gains.

CSMOM appears to be an intriguing mix of risk and reward, exhibiting a broad range of returns both below and above the median. This strategy suffers the largest losses, yet it also reaps the largest rewards, resulting in higher variability in its returns. The elevated downside deviation and the notably high maximum drawdown show its susceptibility to significant losses during downturns. These characteristics point to the potential for steep decreases in portfolio value, which could erode gains quickly, as evident in the very low expected total returns. Yet, it is important to highlight the silver lining within this scenario. Despite these risk indicators, CSMOM registers the highest median return, implying that the strategy often yields substantial profits. Hence, investors with a high risk tolerance might still find CSMOM to be an attractive option due to its enticing return potential.

Shifting our focus to TSMOM, we find a return distribution similar to CSMOM, particularly in terms of its relatively heightened risk metrics. However, TSMOM distinguishes itself through more constrained variability in returns, evidenced by a narrower range of minimum and maximum values, along with lower standard and downside deviations compared to CSMOM. This narrower range is similar to AMOM, albeit positioned on a less desirable part of the spectrum, as indicated by the low measures of central tendency and the negative skewness of returns. Such a distribution provides insight into TSMOM's lagging performance in total returns compared to other strategies. Despite these less favorable outcomes, TSMOM demonstrates a certain consistency, as highlighted by the relative stability in returns and a high percentage of positive returns. This stable performance across various market states, although forgoing higher returns, underscores the resilient nature of TSMOM in comparison to CSMOM, a finding that aligns with Bird et al. (2017).

Lastly, MKT, the market-wide strategy of buying the index, records a notably high median. Its percentage of positive returns is also remarkably high at 64%, which indicates the average market direction during our testing period. This observation potentially explains why the strategies struggle with generating positive returns in their short portfolios. However, the skewness and excess kurtosis for MKT stand at -0.21 and -0.83, respectively, suggesting a distribution that is relatively symmetric and normal, characteristic of a mature, efficient market.

4.4.3 Performance Ratios

Given the nuanced profiles of each strategy's returns, it becomes challenging to categorically define the best strategy, as this is highly contingent on the individual investor's preferences, investment objectives, and risk tolerance. However, we can turn to the risk-return performance ratios of Sharpe and Sortino, which provide a more balanced view of performance by accounting for both the return and risk involved.

The TFT strategy shines in this regard, boasting the highest Sharpe ratio of 1.52 (1.36) before (after) transaction costs. Its risk profile, particularly in terms of downside risk, is commendably low, resulting in a highly impressive Sortino ratio of 6.49 (5.92) before (after) transaction costs. The AMOM strategy comes in as a strong contender, registering a Sharpe ratio marginally below the TFT strategy both before and after transaction costs, and with Sortino ratios of 3.50 (2.97) before (after) transaction costs. On the other hand, CSMOM and TSMOM register notably low Sharpe ratios around zero, accompanied by similarly low Sortino ratios. CSMOM presents a high-risk, high-reward profile, while TSMOM shows a contrasting lower-risk, lower-reward profile, illustrating the challenges in achieving a satisfactory risk-return balance for both benchmarks. Concluding our comparative analysis, we find that while the TFT strategy emerges as the overall best performer, the AMOM is noteworthy as the top-performing strategy among the simpler, rule-based approaches.

4.4.4 Sources of Financial Performance

There are likely several factors contributing to the superior performance demonstrated by the AMOM strategy when compared to both CSMOM and TSMOM. What particularly catches our attention is that, despite being structurally related, AMOM and CSMOM display large differences in their return distributions. This dissimilarity is somewhat surprising considering that if we were to remove the dynamic formation, holding periods, and reversals characterizing AMOM, we would have a CSMOM strategy. Consequently, the disparities between these strategies underscore the influential impact of the distinctive features incorporated into AMOM.

Followers of EMH suggest that market anomalies can often be ephemeral. As these anomalies are discovered and market participants try to take advantage of them, the very process of doing so can cause these anomalies to disappear, as their exploitation leads to more efficient pricing in the market. Therefore, it is important to note that CSMOM and TSMOM strategies were backtested during the late 20th century. It is conceivable that markets have grown increasingly efficient since then, thanks to technological advancements, increased regulation, and wider access to information, thereby diminishing the effectiveness of these traditional strategies. In line with this, Hwang and Rubesam (2015) specifically found that the return premium associated with pure momentum strategies disappeared in the late 1990s.

While the transition from the 20th to the 21st century saw stock markets potentially becoming more efficient, it was also the period in which the Halloween effect showed remarkable persistence. Plastun et al. (2020) argue that, unlike other anomalies, the Halloween effect does not suffer from Murphy's law, in that it is not perpetuated by its own discovery. Several reasons for this have been put forward in the literature, with the most creative focusing on psychological and environmental factors underpinning investor behavior, such as weather and investors' mood (see e.g. Cao and Wei, 2005; Kamstra et al., 2003; P. J. Kelly and Meschke, 2010). Nevertheless, this unique feature may allow our trading strategy to continue benefiting from the market anomalies despite increasing market efficiency, thereby outperforming strategies based solely on momentum anomalies.

The superior performance of AMOM compared to traditional benchmark strategies may, in part, also be attributed to its utilization of dynamic formation periods. As Bird et al. (2017) suggest, the formation period's purpose is to strike an optimal balance between trend identification and acting on short-term fluctuations, essentially discerning signal from noise. With an insufficiently long formation period, the strategy may react excessively to transient market fluctuations. Conversely, an overly lengthy formation period may limit the strategy's responsiveness and result in missed opportunities, as it might not fully capitalize on certain trends. By dynamically adapting the formation period using the VIX, the AMOM strategy may potentially be better suited to balance this trade-off in turbulent times, enabling it to harness short-term trends that might be overlooked with a longer, static formation period.

The adaptive use of the VIX by AMOM to set holding periods could be a crucial component of its superior risk-reward trade-off. A key element to this superior performance might be rooted in the research of Banerjee et al. (2007), which suggests that the VIX could be a priced risk factor. The concept of priced risk factors is a cornerstone of financial theory, as per the CAPM and the Fama-French three-factor model, implying that certain systematic risks are associated with a compensatory premium (Fama and French, 1993). If the VIX is indeed a systematic risk factor, then the higher expected profitability of AMOM could be explained as compensation for selectively taking on this additional risk. Unlike CSMOM and TSMOM, AMOM actively manages its exposure to this risk factor, at times assuming higher risk to exploit short-term trends, while in more stable periods leveraging the anticipated market calm to seek out longer-term returns.

Finally, in periods of heightened risk signaled by elevated VIX values, AMOM adjusts from a momentum to a reversal strategy. This tactical shift further aligns with the principles of risk and reward dynamics in financial markets. Given the established negative correlation between asset prices and volatility (see e.g. Black, 1976; Bouchaud et al., 2001), high levels of VIX would indicate a period of lower prices and potentially higher future returns. In these circumstances, AMOM's reversal strategy on the long-side seeks to capitalize on these 'undervalued' assets that may be well-positioned for potential recovery once market volatility subsides. Da et al. (2014) attribute the reversal on the long-side to the fact that recent losers are more likely to be financially distressed, and constrained investors are forced to sell, causing large price concessions. Subsequent price recovery can then be interpreted as compensation for providing liquidity to these distressed assets.

Conversely, the justification for AMOM's short-side reversals is less apparent. While numerous studies have documented significant short-term reversal effects on the short-side, Da et al. (2014) find that these profits are harder to explain using conventional risk factors. Instead, they suggest the influential role of investor sentiment in driving these outcomes. The authors refer to the idea that investors become overly optimistic and may drive the prices up beyond their fundamental value. However, in our context, this explanation seems contradictory as periods of high VIX, which prompts AMOM's shift to a reversal strategy, are generally associated with investor anxiety rather than optimism. An alternative interpretation is that shorting appreciated stocks during high volatility periods could potentially be seen as a bet on the negative correlation of prices and volatility extending to these specific assets, i.e., expecting their prices to revert due to the overall negative sentiment reflected in the high VIX levels. We must underscore, however, that this phenomenon is highly complex, shaped by several factors including market conditions, regulatory constraints on short-selling, and investor behavior, among others. A definitive conclusion would demand more extensive exploration.

Regardless of underlying drivers, the strategic agility in responding to volatility forecasts sets AMOM apart from CSMOM and TSMOM, which consistently overlook the information embedded in the VIX. As a result, AMOM seems to identify more profitable trading opportunities.

Chapter 5

Conclusion

Our results affirm two critical points: First, the sophisticated Temporal Fusion Transformer (TFT) model demonstrates the effectiveness of novel machine learning models in generating trading signals that lead to attractive financial performance. Second, the interpretability inherent in the TFT is not merely a luxury, but a valuable asset that offers deeper insight into financial market dynamics. We effectively quantify the value of this interpretability by harnessing the insights derived from the model to craft a simple, rule-based strategy that outperforms comparable benchmarks. This achievement illustrates the compelling advantage of combining state-of-the-art machine learning techniques with insightful interpretation in the field of financial market analysis.

Upon interpreting the TFT, we observe a range of patterns that are intuitive or align with established financial theory. For example, the model gives attention to recent data during periods of market shifts, recognizes known turbulent times, and also reflects calendar anomalies well-documented in financial literature. The TFT learns these temporal patterns from raw training data without any human hard-coding. The patterns further validate the model's capacity to predict stock returns by learning from past trends and adapting to volatility levels, a key aspect of successful forecasting in the stock market. Such interpretable capability is shown to be very useful in creating a trading strategy and is expected to foster trust with financial experts via sanity-checking.

In addition to its capabilities as a trusted forecasting model, we show that the TFT also serves as a valuable data analysis tool. It could complement traditional statistical methods such as autocorrelation plots and regression analyses, where the latter often rely on assumptions about the data such as linearity, and the existence of certain statistical properties like homoscedasticity or normality. These assumptions are generally necessary to ensure accurate inferences. Deep learning models like the TFT, however, operate under a different paradigm. They focus on fitting and generalizing the patterns inherent in data, with their primary objective revolving around accurate prediction rather than inference. Consequently, such models, in their pursuit of effective prediction, open up new pathways for understanding complex, high-dimensional time series data through creative visualization of the model weights and predictions. Model developers can utilize these findings towards model improvements, e.g. via specific feature engineering or data collection.

While our results are promising, there remains a wealth of opportunities for further research and exploration. Firstly, we have primarily focused on certain interpretations of the model's output, but there may be additional ways to interpret our presented plots or alternative analyses of the output which could yield new insights. Deepening our understanding of the model's output could potentially lead to a more nuanced understanding of persisting market dynamics, which can help shape trading strategies able to improve financial performance even further.

Secondly, while we use the TFT to predict returns and leverage the uncertainty of the predictions to find the most promising over- and underperformers, it might be insightful to train the TFT to directly output position sizes, as in Wood, Giegerich, et al. (2022), and analyze the resulting portfolios. This approach could provide an enriched perspective on market dynamics and optimal portfolio composition, such as examining the distribution of long and short positions throughout the year and the diversification of the portfolios. Such analyses might further our understanding of stock markets, leading to more robust trading strategies.

Lastly, our research has focused on the S&P 500, a representation of a highly efficient market. It is important to note, however, that markets greatly vary in their levels of efficiency. Inefficiencies can stem from various factors, including information asymmetries, lack of liquidity, high transaction costs or delays, market psychology, and human emotion, among others. It could be valuable to explore how the TFT performs in less efficient developing markets where such inefficiencies might offer greater opportunities to exploit profitable market anomalies. In conclusion, there remain ample opportunities for further research.

Bibliography

Andreu, L., Swinkels, L., and Tjong-A-Tjoe, L. (2013). “Can exchange traded funds be used to exploit industry and country momentum?” *Financial Markets and Portfolio Management* 27(2), pp. 127–148.

Ang, A. and Timmermann, A. (2012). “Regime changes and financial markets”. *Annual Review of Finacial Economics* 4(1), pp. 313–337.

Ang, A., Hodrick, R. J., et al. (2006). “The cross-section of volatility and expected returns”. *The journal of finance* 61(1), pp. 259–299.

Asem, E. and Tian, G. Y. (2010). “Market Dynamics and Momentum Profits”. *The Journal of Financial and Quantitative Analysis* 45(6), pp. 1549–1562.

Asness, C. S., Moskowitz, T. J., and Pedersen, L. H. (2013). “Value and Momentum Everywhere”. *The Journal of Finance* 68(3), pp. 929–985.

Avellaneda, M. and Lee, J.-H. (2010). “Statistical arbitrage in the US equities market”. *Quantitative Finance* 10(7), pp. 761–782.

Avramov, D., Cheng, S., and Hameed, A. (2016). “Time-Varying Liquidity and Momentum Profits”. *The Journal of Financial and Quantitative Analysis* 51(6), pp. 1897–1923.

Baker, M. and Stein, J. C. (2004). “Market liquidity as a sentiment indicator”. *Journal of financial Markets* 7(3), pp. 271–299.

Bakshi, G. and Kapadia, N. (2003). “Delta-hedged gains and the negative market volatility risk premium”. *The Review of Financial Studies* 16(2), pp. 527–566.

Banerjee, P. S., Doran, J. S., and Peterson, D. R. (2007). “Implied volatility and future portfolio returns”. *Journal of Banking & Finance* 31(10), pp. 3183–3199.

Barroso, P. and Santa-Clara, P. (2015). “Momentum has its moments”. *Journal of Financial Economics* 116(1), pp. 111–120.

Bello, Z. Y. (2008). “A Statistical Comparison of the CAPM to the Fama-French Three Factor Model and the Carhart’s Model”. *Global Journal of Finance and Banking Issues* 2(2), pp. 14–24.

Bengio, Y., Simard, P., and Frasconi, P. (1994). “Learning long-term dependencies with gradient descent is difficult”. *IEEE Transactions on Neural Networks* 5(2), pp. 157–166.

Bird, R., Gao, X., and Yeung, D. (2017). “Time-series and cross-sectional momentum strategies under alternative implementation strategies”. *Australian Journal of Management* 42(2), pp. 230–251.

Black, F. (1976). “Studies of stock market volatility changes”. *1976 Proceedings of the American statistical association business and economic statistics section.*

Blair, B. J., Poon, S.-H., and Taylor, S. J. (2010). *Forecasting S&P 100 volatility: the incremental information content of implied volatilities and high-frequency index returns.* Springer.

Bouchaud, J.-P., Matacz, A., and Potters, M. (2001). “Leverage effect in financial markets: The retarded volatility model”. *Physical review letters* 87(22), p. 228701.

Bouman, S. and Jacobsen, B. (2002). “The Halloween Indicator, ”Sell in May and Go Away”: Another Puzzle”. *American Economic Review* 92(5), pp. 1618–1635.

Cao, M. and Wei, J. (2005). “Stock market returns: A note on temperature anomaly”. *Journal of Banking & Finance* 29(6), pp. 1559–1573.

Carhart, M. M. (1997). “On Persistence in Mutual Fund Performance”. *The Journal of Finance* 52(1), pp. 57–82.

Chan, K., Hameed, A., and Tong, W. (2000). "Profitability of Momentum Strategies in the International Equity Markets". *The Journal of Financial and Quantitative Analysis* 35(2), pp. 153–172.

Chen, S. and Ge, L. (2019). "Exploring the attention mechanism in LSTM-based Hong Kong stock price movement prediction". *Quantitative Finance* 19(9), pp. 1507–1515.

Christensen, B. J. and Prabhala, N. R. (1998). "The relation between implied and realized volatility". *Journal of financial economics* 50(2), pp. 125–150.

Christensen, B. J. and Hansen, C. S. (2002). "New evidence on the implied-realized volatility relation". *The European Journal of Finance* 8(2), pp. 187–205.

Clevert, D.-A., Unterthiner, T., and Hochreiter, S. (2016). "Fast and Accurate Deep Network Learning by Exponential Linear Units (ELUs)". ICLR (2016).

Connolly, R. and Stivers, C. (2003). "Momentum and Reversals in Equity-Index Returns During Periods of Abnormal Turnover and Return Dispersion". *The Journal of Finance* 58(4), pp. 1521–1556.

Consumer Financial Protection Bureau (2023). *Joint Statement On Enforcement Efforts Against Discrimination And Bias In Automated Systems.* URL: https://www.consumerfinance.gov/about-us/newsroom/cfpb-federal-partners-confirm-automated-systems-advanced-technology-not-an-excuse-for-lawbreaking-behavior/ (visited on Apr. 27, 2023).

Cooper, M. J., Gutierrez Jr., R. C., and Hameed, A. (2004). "Market States and Momentum". *The Journal of Finance* 59(3), pp. 1345–1365.

Coval, J. D. and Shumway, T. (2001). "Expected option returns". *The journal of Finance* 56(3), pp. 983–1009.

Da, Z., Liu, Q., and Schaumburg, E. (2014). "A closer look at the short-term return reversal". *Management science* 60(3), pp. 658–674.

Daniel, K. and Moskowitz, T. J. (2016). "Momentum crashes". *Journal of Financial Economics* 122(2), pp. 221–247.

Dauphin, Y. N. et al. (2017). "Language modeling with gated convolutional networks". *International conference on machine learning.* PMLR, pp. 933–941.

De Oliveira, F. A., Nobre, C. N., and Zárate, L. E. (2013). "Applying Artificial Neural Networks to prediction of stock price and improvement of the directional prediction index–Case study of PETR4, Petrobras, Brazil". *Expert systems with applications* 40(18), pp. 7596–7606.

Dixon, M., Klabjan, D., and Bang, J. H. (2015). "Implementing deep neural networks for financial market prediction on the Intel Xeon Phi". *Proceedings of the 8th Workshop on High Performance Computational Finance*, pp. 1–6.

European Commission (2019). *Requirements of Trustworthy AI.* URL: https://ec.europa.eu/futurium/en/ai-alliance-consultation/guidelines/1 (visited on Apr. 27, 2023).

Fama, E. F. (1970). "Efficient Capital Markets: A Review of Theory and Empirical Work". *Journal of Finance* 25(2), pp. 383–417.

— (1991). "Efficient Capital Markets: II". *Journal of Finace* 46(5), pp. 1575–1617.

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

Fan, C. et al. (2019). "Multi-Horizon Time Series Forecasting with Temporal Attention Learning". *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 2527–2535.

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.

Gal, Y. and Ghahramani, Z. (2016). "A theoretically grounded application of dropout in recurrent neural networks". *Advances in neural information processing systems* 29.

Ghosh, P., Neufeld, A., and Sahoo, J. K. (2022). "Forecasting directional movements of stock prices for intraday trading using LSTM and random forests". *Finance Research Letters* 46, p. 102280.

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

Grobys, K. and Kolari, J. (2020). "On industry momentum strategies". *Journal of Financial Research* 43(1), pp. 95–119.

Gu, S., Kelly, B., and Xiu, D. (2020). "Empirical Asset Pricing via Machine Learning". *The Review of Financial Studies* 33(5), pp. 2223–2273.

Gunduz, H., Yaslan, Y., and Cataltepe, Z. (2017). "Intraday prediction of Borsa Istanbul using convolutional neural networks and feature correlations". *Knowledge-Based Systems* 137, pp. 138–148.

Guo, B., Luo, X., and Zhang, Z. (2014). “Sell in May and go away: Evidence from China”. *Finance Research Letters* 11(4), pp. 362–368.

Hong, H., Lim, T., and Stein, J. C. (2000). “Bad News Travels Slowly: Size, Analyst Coverage, and the Profitability of Momentum Strategies”. *The Journal of Finance* 55(1), pp. 265–295.

Hu, X. (2021). “Stock Price Prediction Based on Temporal Fusion Transformer”. *2021 3rd International Conference on Machine Learning, Big Data and Business Intelligence (MLBDBI)*, pp. 60–66.

Hwang, S. and Rubesam, A. (2015). “The disappearance of momentum”. *The European Journal of Finance* 21(7), pp. 584–607.

Ioffe, S. and Szegedy, C. (2015). “Batch normalization: Accelerating deep network training by reducing internal covariate shift”. *International conference on machine learning.* pmlr, pp. 448–456.

Jacobsen, B. and Visaltanachoti, N. (2009). “The Halloween effect in US sectors”. *Financial Review* 44(3), pp. 437–459.

Jegadeesh, N. (1990). “Evidence of Predictable Behavior of Security Returns”. *The Journal of Finance* 45(3), pp. 881–898.

Jegadeesh, N. and Titman, S. (1993). “Returns to Buying Winners and Selling Losers: Implications for Stock Market Efficiency”. *The Journal of Finance* 48(1), pp. 65–91.

Jha, V. (2016). “Timing Equity Quant Positions with Short-Horizon Alphas”. *The Journal of Trading* 11(3), pp. 53–59.

Kailath, T. (1967). “The divergence and bhattacharyya distance measures in signal selection”. *IEEE Transactions on Communication Technology* 15(1), pp. 52–60.

Kamstra, M. J., Kramer, L. A., and Levi, M. D. (2003). “Winter blues: A SAD stock market cycle”. *American economic review* 93(1), pp. 324–343.

Kelly, B. T., Moskowitz, T. J., and Pruitt, S. (2021). “Understanding momentum and reversal”. *Journal of Financial Economics* 140(3), pp. 726–743.

Kelly, P. J. and Meschke, F. (2010). “Sentiment and stock returns: The SAD anomaly revisited”. *Journal of Banking & Finance* 34(6), pp. 1308–1326.

Kim, S. and Kang, M. (2019). “Financial series prediction using Attention LSTM”. *arXiv:1902.10877.*

Krauss, C., Do, X. A., and Huck, N. (2017). “Deep neural networks, gradient-boosted trees, random forests: Statistical arbitrage on the S&P 500”. *European Journal of Operational Research* 259(2), pp. 199–219.

Lean, H. H. (2011). “The Halloween puzzle in selected Asian stock markets”. *International Journal of Economics and Management* 5(1), pp. 216–225.

Lecun, Y. et al. (1998). “Gradient-based learning applied to document recognition”. *Proceedings of the IEEE* 86(11), pp. 2278–2324.

Lee, C. and Swaminathan, B. (2000). “Price Momentum and Trading Volume”. *The Journal of Finance* 55(5), pp. 2017–2069.

Lehmann, B. N. (1990). “Fads, Martingales, and Market Efficiency”. *The Quarterly Journal of Economics* 105(1), pp. 1–28.

Lei Ba, J., Kiros, J. R., and Hinton, G. E. (2016). “Layer Normalization”. *arXiv:1607.06450.*

Li, S. et al. (2019). “Enhancing the locality and breaking the memory bottleneck of transformer on time series forecasting”. *Advances in neural information processing systems* 32.

Li, Y. et al. (2022). “Incorporating Transformers and Attention Networks for Stock Movement Prediction”. *Complexity* 2022, p. 7739087.

Lim, B., Arik, S. Ö., et al. (2021). “Temporal Fusion Transformers for interpretable multi-horizon time series forecasting”. *International Journal of Forecasting* 37(4), pp. 1748–1764.

Lim, B., Zohren, S., and Roberts, S. (2019). “Enhancing time-series momentum strategies using deep neural networks”. *The Journal of Financial Data Science* 1(4), pp. 19–38.

Lo, A. W. (2004). “The adaptive markets hypothesis: Market efficiency from an evolutionary perspective”. *Journal of Portfolio Management, Forthcoming.*

Lundberg, S. M. and Lee, S.-I. (2017). “A Unified Approach to Interpreting Model Predictions”. *Advances in Neural Information Processing Systems* 30.

Maberly, E. D. and Pierce, R. M. (2004). “Stock market efficiency withstands another challenge: Solving the” sell in May/buy after Halloween” puzzle”. *Econ Journal Watch* 1(1), p. 29.

Makridakis, S. et al. (2023). “Statistical, machine learning and deep learning forecasting methods: Comparisons and ways forward”. *Journal of the Operational Research Society* 74(3), pp. 840–859.

Malkiel, B. G. (1973). *A Random Walk Down Wall Street*. W.W. Norton & Company.

Medsker, L. and Jain, L. (2000). *Recurrent neural networks: Design and applications*. CRC Press.

Moritz, B. and Zimmermann, T. (2016). "Tree-based conditional portfolio sorts: The relation between past and future stock returns". *SSRN 2740751*.

Moskowitz, T. J. and Grinblatt, M. (1999). "Do Industries Explain Momentum?" *The Journal of Finance* 54(4), pp. 1249–1290.

Moskowitz, T. J., Ooi, Y. H., and Pedersen, L. H. (2012). "Time series momentum". *Journal of Financial Economics* 104(2), pp. 228–250.

Nam, K., Pyun, C. S., and Avard, S. L. (2001). "Asymmetric reverting behavior of short-horizon stock returns: An evidence of stock market overreaction". *Journal of Banking & Finance* 25(4), pp. 807–824.

Niaki, S. T. A. and Hoseinzade, S. (2013). "Forecasting S&P 500 index using artificial neural networks and design of experiments". *Journal of Industrial Engineering International* 9(1), p. 1.

O'Neal, E. S. (2000). "Industry Momentum and Sector Mutual Funds". *Financial Analysts Journal* 56(4), pp. 37–49.

Pascanu, R., Mikolov, T., and Bengio, Y. (2013). "On the difficulty of training recurrent neural networks". *Proceedings of the 30th International Conference on Machine Learning* 28(3), pp. 1310–1318.

Pástor, Ľ. and Stambaugh, R. F. (2003). "Liquidity Risk and Expected Stock Returns". *Journal of Political Economy* 111(3), pp. 642–685.

Plastun, A. et al. (2020). "Halloween Effect in developed stock markets: A historical perspective". *International Economics* 161, pp. 130–138.

Rangapuram, S. S. et al. (2018). "Deep State Space Models for Time Series Forecasting". *Advances in Neural Information Processing Systems* 31.

Ribeiro, M. T., Singh, S., and Guestrin, C. (2016). ""Why Should I Trust You?": Explaining the Predictions of Any Classifier". *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1135–1144.

Rojat, T. et al. (2021). "Explainable artificial intelligence (xai) on timeseries data: A survey". *arXiv:2104.00950*.

Rouwenhorst, K. G. (1998). "International Momentum Strategies". *The Journal of Finance* 53(1), pp. 267–284.

— (1999). "Local Return Factors and Turnover in Emerging Stock Markets". *The Journal of Finance* 54(4), pp. 1439–1464.

Rozeff, M. S. and Kinney, J. (1976). "Capital market seasonality: The case of stock returns". *Journal of Financial Economics* 3(4), pp. 379–402.

Ruenzi, S. and Weigert, F. (2018). "Momentum and crash sensitivity". *Economics Letters* 165, pp. 77–81.

Sadka, R. (2006). "Momentum and post-earnings-announcement drift anomalies: The role of liquidity risk". *Journal of Financial Economics* 80(2), pp. 309–349.

Salinas, D. et al. (2020). "DeepAR: Probabilistic forecasting with autoregressive recurrent networks". *International Journal of Forecasting* 36(3), pp. 1181–1191.

Schmitt, T. A. et al. (2013). "Non-stationarity in financial time series: Generic features and tail behavior". *Europhysics Letters* 103(5), p. 58003.

Sezer, O. B. and Ozbayoglu, A. M. (2018). "Algorithmic financial trading with deep convolutional neural networks: Time series to image conversion approach". *Applied Soft Computing* 70, pp. 525–538.

Siddiqui, S. A. et al. (2019). "TSViz: Demystification of Deep Learning Models for Time-Series Analysis". *IEEE Access* 7, pp. 67027–67040.

Su, D. (2011). "An Empirical Analysis of Industry Momentum in Chinese Stock Markets". *Emerging Markets Finance & Trade* 47(4), pp. 4–27.

Swinkels, L. (2002). "International industry momentum". *Journal of Asset Management* 3(2), pp. 124–141.

Szakmary, A. C. and Zhou, X. (2015). "Industry Momentum in an Earlier Time: Evidence from the Cowles Data". *Journal of Financial Research* 38(3), pp. 319–347.

Takeuchi, L. and Lee, Y.-Y. A. (2013). "Applying deep learning to enhance momentum trading strategies in stocks". *Technical Report*. Stanford University Stanford, CA, USA.

Tan, Y. M. and Cheng, F. F. (2019). “Industry- and liquidity-based momentum in Australian equities”. *Financial Innovation* 5(1), p. 43.

Topal, M. O., Bas, A., and Heerden, I. van (2021). “Exploring transformers in natural language generation: Gpt, bert, and xlnet”. *arXiv:2102.08036*.

Vaswani, A. et al. (2017). “Attention is all you need”. *Proceedings of the 31st Conference on Neural Information Processing Systems (NIPS 2017)*.

Wen, R. et al. (2017). “A multi-horizon quantile recurrent forecaster”. *arXiv:1711.11053*.

Witte, H. D. (2010). “Outliers and the Halloween effect: comment on Maberly and Pierce”. *Econ Journal Watch* 7(1), p. 91.

Wood, K., Giegerich, S., et al. (2022). “Trading with the Momentum Transformer: An Intelligent and Interpretable Architecture”. *arXiv:2112.08534*.

Wood, K., Roberts, S., and Zohren, S. (2022). “Slow Momentum with Fast Reversion: A Trading Strategy Using Deep Learning and Changepoint Detection”. *The Journal of Financial Data Science* 4(1), pp. 111–129.

Zhang, Q. et al. (2022). “Transformer-based attention network for stock movement prediction”. *Expert Systems with Applications* 202, p. 117239.

Appendix A

Long Short-Term Memory Networks

Long Short-Term Memory (LSTM) networks belong to the class of recurrent neural networks (RNNs), i.e., neural networks whose ”underlying topology of inter-neuronal connections contains at least one cycle” (Medsker and Jain, 2000, p. 82).

LSTM networks are composed of an input layer, one or more hidden layers, and an output layer. The dimension of the input layer is equal to the number of explanatory variables (feature space). The number of neurons in the output layer reflects the output space. The main characteristic of LSTM networks is contained in the hidden layer.

LSTM networks have hidden “LSTM memory cells” that have an internal recurrence (a self-loop), controlling the flow of information based on its state (c_t). Each of the memory cells has three gates maintaining and adjusting the cell state: a forget gate (f_t), an input gate (i_t), and an output gate (o_t), with their corresponding weight matrices (W) and bias vectors (b). The structure of a memory cell is illustrated in Figure A.1.

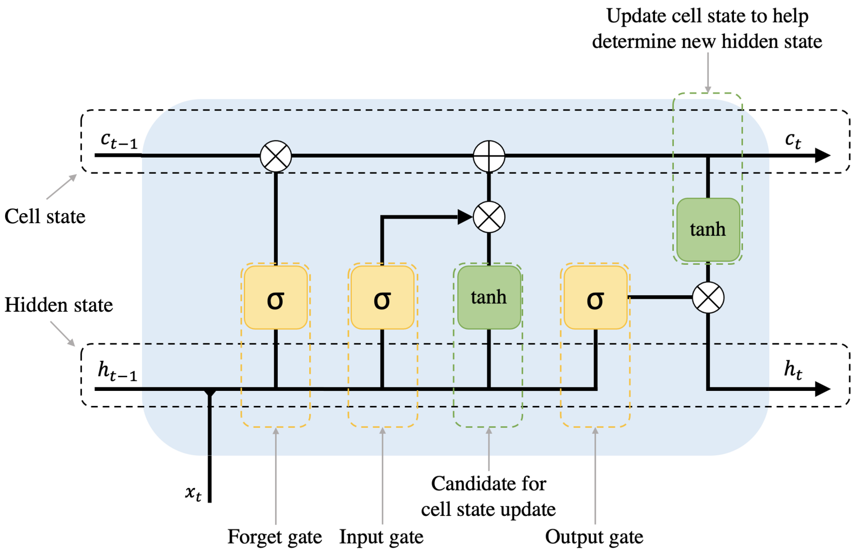

Figure A.1 LSTM memory cell.

At each timestep t, the network is presented with the input x_t and the output h_{t-1} of the memory cells from the previous timestep $t-1$. Firstly, the LSTM layer of cells determines which information should be removed from its previous cell states using the forget gate:

$$f_t = \sigma(W_{f,x}x_t + W_{f,h}h_{t-1} + b_f) \tag{A.1}$$

where $\sigma(\cdot)$ is the sigmoid function that sets this weight to a value between 0 (completely forget) and 1 (completely remember). Then, the LSTM layer determines which information should be added to the network's cell states:

$$i_t = \sigma(W_{i,x}x_t + W_{i,h}h_{t-1} + b_i) \tag{A.2}$$

$$\tilde{c}_t = \tanh(W_{\tilde{c},x}x_t + W_{\tilde{c},h}h_{t-1} + b_{\tilde{c}}) \tag{A.3}$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t \tag{A.4}$$

where $\odot$ is the Hadamard (element-wise) product. In the last step, the output h_t of the memory cells is derived as denoted in the following two equations:

$$o_t = \sigma(W_{o,x}x_t + W_{o,h}h_{t-1} + b_o) \tag{A.5}$$

$$h_t = o_t \odot \tanh(c_t) \tag{A.6}$$

Equations (A.1)–(A.6) show that the LSTM uses the cell state as a compact summary of past information, controlling memory retention with the forget gate, incorporating new information via the input gate, and specifying which information from the cell state to pass on via the output gate. As such, the LSTM can learn representations of long-term relationships relevant to the prediction task — sequentially updating its internal memory states with new observations at each step.

Appendix B

Benchmark Strategies

The results from examining various combinations of formation periods (J) and holding periods (K) for the Cross-Sectional Momentum (CSMOM) (Jegadeesh and Titman, 1993) and Time-Series Momentum (TSMOM) (Moskowitz, Ooi, et al., 2012) benchmark strategies are provided in Table B.1 and Table B.2, respectively. The number of combinations tested is constrained by the boundaries set by the TFT model's 12-month lookback window and 6-month step ahead prediction horizon. Importantly, the various implementations are applied to our test set, which spans a period from July 2020 through to December 2022. We consciously allow the benchmarks to have a 'look ahead' advantage during configuration testing, effectively augmenting their performance. The underlying logic for this method is simple: if our strategy can outperform these 'inflated' benchmarks, it serves as a robust testament to its effectiveness.

The results generally depict total returns that fluctuate around zero, indicating limited effectiveness of the strategies in our test period. The strategies particularly struggle when applied with extended formation and holding periods, which is evidenced by the lower returns located in the bottom right sections of the tables. Notably, it is the 4/1-implementation that emerges as the superior choice for both the CSMOM and TSMOM strategies. Thus, it is chosen as the optimal configuration for our benchmarks.

Table B.1
Expected monthly returns for CSMOM (Jegadeesh and Titman, 1993) implementations using various formation periods (J) and holding periods (K). The implementations are applied to our test set spanning from July 2020 to December 2022. The optimal implementation is highlighted in bold.

		K					
		1	2	3	4	5	6
J	1	0.0045	-0.0108	-0.0045	-0.0034	-0.0089	-0.0100
	2	-0.0117	-0.0106	-0.0033	-0.0064	-0.0111	-0.0117
	3	-0.0043	-0.0036	-0.0019	-0.0043	-0.0076	-0.0078
	4	**0.0063**	-0.0027	-0.0024	-0.0061	-0.0115	-0.0128
	5	-0.0090	-0.0109	-0.0058	-0.0098	-0.0158	-0.0158
	6	-0.0046	-0.0056	-0.0050	-0.0105	-0.0140	-0.0168
	7	0.0023	-0.0065	-0.0090	-0.0106	-0.0188	-0.0226
	8	-0.0060	-0.0134	-0.0116	-0.0180	-0.0263	-0.0308
	9	-0.0116	-0.0141	-0.0152	-0.0216	-0.0293	-0.0345
	10	-0.0068	-0.0152	-0.0166	-0.0228	-0.0303	-0.0363
	11	-0.0186	-0.0216	-0.0210	-0.0272	-0.0343	-0.0381
	12	-0.0175	-0.0213	-0.0204	-0.0258	-0.0324	-0.0348

Table B.2
Expected monthly returns for TSMOM (Moskowitz, Ooi, et al., 2012) implementations using various formation periods (J) and holding periods (K). The implementations are applied to our test set spanning from July 2020 to December 2022. The optimal implementation is highlighted in bold.

		K					
		1	2	3	4	5	6
J	1	-0.0023	-0.0217	-0.0218	-0.0192	-0.0263	-0.0317
	2	-0.0081	-0.0106	-0.0098	-0.0087	-0.0135	-0.0167
	3	-0.0054	-0.0101	-0.0102	-0.0101	-0.0129	-0.0150
	4	**0.0032**	-0.0048	-0.0086	-0.0094	-0.0119	-0.0134
	5	-0.0045	-0.0087	-0.0107	-0.0101	-0.0122	-0.0124
	6	-0.0037	-0.0078	-0.0088	-0.0081	-0.0102	-0.0104
	7	-0.0018	-0.0066	-0.0087	-0.0080	-0.0091	-0.0085
	8	-0.0031	-0.0054	-0.0066	-0.0067	-0.0077	-0.0074
	9	-0.0004	-0.0031	-0.0063	-0.0055	-0.0070	-0.0075
	10	-0.0036	-0.0056	-0.0094	-0.0083	-0.0095	-0.0101
	11	-0.0048	-0.0061	-0.0088	-0.0085	-0.0100	-0.0113
	12	-0.0066	-0.0083	-0.0096	-0.0095	-0.0125	-0.0130

NTNU

Norwegian University of
Science and Technology