Lakehead University

Department of Electrical and Computer Engineering

Multimodal Deep Learning for Multi-Horizon Corporate Revenue Forecasting

A Thesis

submitted in partial fulfillment of the requirements for the degree of Master of Science in Electrical and Computer Engineering at Lakehead University

by

Qiping Wu

Thunder Bay, Ontario, Canada

April 2026

Abstract

Corporate revenue forecasting matters for valuation, portfolio management, and capital allocation. However, it is difficult because financial statements mainly reflect the past, while investors and firms often need forecasts from the next quarter to a rolling one-year horizon. This challenge becomes even greater over longer horizons, especially in fast-changing industries. This thesis addresses the problem by building a forecasting framework that starts with a broad quantitative baseline and then extends to a multimodal approach.

First, this thesis develops a Temporal Fusion Transformer (TFT) baseline for next-quarter revenue forecasting across 155 continuously listed S&P 500 firms. Under a strict chronological evaluation protocol, the TFT model achieves a test Mean Absolute Percentage Error (MAPE) of 9.31%, a Root Mean Squared Error (RMSE) of 1,973 million USD, and a Mean Absolute Error (MAE) of 1,790 million USD. Controlled ablation analysis further shows that accurate short-horizon forecasting depends not only on autoregressive revenue history, but also on structured firm context, including sector identity, year-over-year growth, and firm scale variables such as total assets and equity.

Second, the framework is extended from one-quarter-ahead to four-quarter-ahead forecasting. The results show that forecast accuracy deteriorates as the horizon expands, with MAPE rising from 9.31% at one quarter ahead ($t + 1$) to 12.07% at four quarters ahead ($t + 4$). A comparison with an LSTM baseline under the same chronological setting further suggests that this deterioration is not specific to a single model, but reflects a broader limitation of purely financial forecasting approaches. The effect is especially pronounced in technology-oriented firms, highlighting the limits of relying only on lagged financial data in non-linear growth environments.

Third, the work proposes a multimodal TFT framework that integrates earnings-call-derived textual signals into the forecasting pipeline. Focusing on the Mega-Cap 5 companies, the framework uses both Financial Bidirectional Encoder Representations from Transformers (FinBERT) and a locally deployed Llama-3 8B model to extract finance-domain sentiment and richer generative narrative features from quarterly earnings call transcripts. These results show that transcript-based narrative features improve long-horizon forecasting. Among the models, the Llama-3 representation delivers the biggest improvement. For example, the pure TFT has a MAPE of 53.85%, while the FinBERT+TFT and Llama-3+TFT hybrids reduce it to 48.70% and 43.01%, respectively.

Overall, this thesis presents a practically deployable multimodal forecasting framework that bridges the gap between backward-looking financial fundamentals and forward-looking managerial narratives in corporate revenue forecasting.

Acknowledgments

Reaching the end of this academic journey in 2026 fills me with deep gratitude. My path to a MSc in ECE at Lakehead University was anything but linear. It began with an admission to an Economics Qualifying Year, then shifted into the MBA program in September 2024. Yet, after only one week in the MBA, a transformative thirty-minute conversation with Dr. Yassine changed my trajectory, leading me into Software Engineering and reigniting the academic dreams of my youth. I am deeply grateful to Dr. Yassine for that pivotal discussion, his guidance, and his unwavering support throughout this new academic journey.

I sincerely thank Dr. Akilan for his support and mentorship, as the early chapters of this thesis grew from the course project. I also thank Dr. Zhou for his careful coordination, and Dr. Deng, whose Advanced Optimization course strongly shaped my technical approach to problem-solving.

Life is full of beautiful surprises. Here in Thunder Bay, I was fortunate to cross paths with Dr. Liu and Meili Wang. In a wonderful coincidence, Meili shares the exact same first name as my elder sister, gifting me a lovely "local sister" and brother right here in Canada. I am deeply thankful for the warm guidance and personal support they both provided during my time in the North.

Also, I dedicate this achievement to my family. Three years ago, I brought my elder son, Xiuyuan, to Canada to accompany him through his high school years. Accompanying him allowed me to return to the ivory tower myself. As he now prepare to embark on his own undergraduate journey, I could not be prouder. To my younger son, Zhiyuan—my little Tidan—his eagerness to learn, pursuit of excellence, and brilliant communication skills remind me so much of my own youth. Seeing both of them pursue knowledge with such passion inspires me every single day.

My wife is the absolute anchor of our family. Behind every line of code written and every chapter drafted, there was her unseen labor—the daily symphony of pots, pans, and everyday life. She provided the strong backing and the safe haven that made this entire endeavor possible. She courageously overcame language barriers to build connections in a new country, ultimately bringing wonderful friends into our lives and turning this city into a true home.

Finally, I would like to sincerely thank all my friends who supported and encouraged me, including my lab-mates in ATAC 4013. Their advice, help, inspiration, and even small moments of distraction brought comfort and joy to this journey.

Contents

List of Figures

List of Tables

List of Abbreviations and Symbols

Table 1: List of Abbreviations and Definitions

Abbreviation	Definition
AI	Artificial Intelligence
AdamW	Adaptive Moment Estimation with decoupled Weight Decay
ARIMA	Autoregressive Integrated Moving Average
BERT	Bidirectional Encoder Representations from Transformers
CNN	Convolutional Neural Network
CQ	Calendar Quarter
ECE	Electrical and Computer Engineering
EPS	Earnings Per Share
FinBERT	Financial Bidirectional Encoder Representations from Transformers
FMP	Financial Modeling Prep
GAAP	Generally Accepted Accounting Principles
GICS	Global Industry Classification Standard
GPU	Graphics Processing Unit
GRN	Gated Residual Network
GRU	Gated Recurrent Unit
LLM	Large Language Model
Llama-3	Meta's third-generation Large Language Model Meta AI
LSTM	Long Short-Term Memory
MAE	Mean Absolute Error
MAPE	Mean Absolute Percentage Error
MD&A	Management's Discussion and Analysis
Mega-Cap 5	Apple, Microsoft, Amazon, Alphabet, and Nvidia
ML	Machine Learning
NF4	NormalFloat 4-bit quantization format
NLP	Natural Language Processing

Continued on next page

Abbreviation	Definition
Q&A	Question and Answer
QoQ	Quarter-over-Quarter
R&D	Research and Development
RMSE	Root Mean Squared Error
RNN	Recurrent Neural Network
S&P 500	Standard & Poor's 500 Index
SARIMA	Seasonal Autoregressive Integrated Moving Average
SG&A	Selling, General, and Administrative Expenses
SVM	Support Vector Machine
TFT	Temporal Fusion Transformer
VAR	Vector Autoregression
VRAM	Video Random Access Memory
VSN	Variable Selection Network
YoY	Year-over-Year

Table 2: List of Symbols and Descriptions

Symbol	Description
i	Firm or entity identifier index
t	Current time step (calendar or fiscal quarter)
τ / h	Forecast horizon in quarters
L	Encoder length / historical look-back window
H	Prediction length / maximum forecast horizon
$y_{i,t}$	Actual target variable for firm i at quarter t (log-transformed quarterly revenue)
$\hat{y}_{i,t+\tau}$	Point forecast of the target variable at future horizon τ
$\hat{y}^{(q)}_{i,t+\tau}$	Predicted quantile forecast at level q for firm i at horizon τ
$\mathbf{x}_{i,t}$	General input feature vector for firm i at time t
$\mathbf{s}_i$	Vector of static, time-invariant covariates
$\mathbf{o}_{i,t}$	Vector of observed-past, time-varying covariates
$\mathbf{k}_{i,t}$	Vector of known-future, time-varying covariates
$\mathbf{z}_{i,t}$	Combined multimodal feature vector at time t after structured and textual fusion
$\mathbf{c}^{(\text{sel})}$	Static context vector conditioning variable selection networks
$\mathbf{c}^{(\text{enr})}$	Static context vector conditioning enrichment layers / GRNs
$\mathbf{c}^{(\text{ctx})}$	Static context vector used to initialize temporal states
$\alpha_{t,\tau}$	Temporal attention weight assigned to historical time step t for horizon τ
$\mathbf{h}_t$	Hidden state at time step t in the temporal backbone
θ	Learnable parameters of the forecasting model
q	Target quantile in Quantile Loss (e.g., 0.10, 0.50, 0.90)
$\mathcal{D}_{i,t}$	Earnings call transcript text or token sequence for firm i at quarter t
$d_{i,t,j}$	The jth sentence or text segment in transcript $\mathcal{D}_{i,t}$
$N_{i,t}$	Number of valid transcript segments or sentences for firm i at quarter t
$P_{\text{pos},j}$	FinBERT probability that transcript segment j is positive
$P_{\text{neg},j}$	FinBERT probability that transcript segment j is negative
$P_{\text{neu},j}$	FinBERT probability that transcript segment j is neutral
$Sen_{i,t}$	Quarter-level FinBERT net sentiment score for firm i at quarter t
$\bar{P}^{\text{pos}}_{i,t}$	Mean positive sentiment probability aggregated at the firm-quarter level
$\bar{P}^{\text{neg}}_{i,t}$	Mean negative sentiment probability aggregated at the firm-quarter level
$\bar{P}^{\text{neu}}_{i,t}$	Mean neutral sentiment probability aggregated at the firm-quarter level
$G_{i,t,j}$	Llama-3 generated sentiment score for transcript segment j
$\bar{G}_{i,t}$	Mean Llama-3 sentiment score at the firm-quarter level (`sent_net_mean`)
$F_{i,t}$	Forward-looking intensity feature at the firm-quarter level (`sent_fli_count`)
$\mathcal{L}_q$	Quantile loss at quantile level q
$\mathcal{L}$	Total training loss
η	Learning rate
$\eta_{\max}$	Peak learning rate after warm-up
e	Training epoch index
E	Total number of training epochs

Chapter 1

Introduction

1.1 Introduction and Context

Accurate forecasting of quarterly corporate revenue is important for both investors and firms. In financial markets, revenue expectations affect views on growth, value, and risk. Inside firms, revenue forecasts support budgeting, hiring, inventory planning, and major investment decisions. Despite its importance, quarterly revenue remains difficult to predict well because it is low-frequency, shaped by seasonality and reporting conventions, and sensitive to changes in macroeconomic conditions, competitive dynamics, and firm strategy.

Traditional statistical approaches provide a natural starting point for this problem. Classical time-series models such as Autoregressive Integrated Moving Average (ARIMA) and Seasonal Autoregressive Integrated Moving Average (SARIMA) remain strong benchmarks when the process is relatively linear and stable over time [1]. Vector autoregression (VAR) further allow multiple variables to be modeled jointly [2]. However, these methods often become less effective when relationships change, growth is nonlinear, or variables interact in complex ways. DL models offer a more flexible alternative because they can learn nonlinear temporal patterns directly from data. Recurrent neural networks (RNNs), especially Long Short-Term Memory (LSTM) networks, have therefore been widely used for sequential prediction tasks [3]. But in financial forecasting, flexibility alone is not enough. Models must also support multi-horizon forecasts and produce results that can be understood and trusted.

Thus, this research proposes the TFT as a structured DL architecture for multi-horizon quarterly revenue forecasting [4]. TFT is designed for mixed-input settings and can jointly incorporate static firm attributes, observed historical variables, and known future inputs. It also provides built-in variable selection and interpretable attention mechanisms, making it well suited to a forecasting problem where both predictive accuracy and interpretability matter.

This thesis further extends structured financial forecasting by incorporating narrative information from earnings call transcripts. Earnings calls contain forward-looking managerial discussion that may complement backward-looking financial statements [5]. To capture this information, the paper integrates text-derived signals generated by both FinBERT and Llama 3 into the TFT framework, with the goal of improving multi-horizon revenue forecasting in a unified and auditable pipeline.

1.2 Research Motivation

The main motivation of this study is that revenue forecasting is still very important in practice, but it becomes much harder over the time horizon that investors and analysts care about. In many real-world financial settings, the focus is not limited to the next quarter. Instead, market participants often assess firms based on expected revenue and other key financial indicators over the next four quarters, or on a rolling one-year basis. This makes multi-horizon forecasting important for valuation, portfolio construction, capital allocation, and expectation management.

At the same time, the forecasting problem is far from solved. Although analyst consensus is often treated as the practical benchmark, prior research shows that simpler models can sometimes match or outperform professional forecasts. For example, Pagach and Warr report that ARIMA can equal or exceed analyst consensus accuracy in a meaningful share of quarterly earnings forecasting cases [6]. Choi et al. further show that some machine learning (ML) methods can outperform professional forecasts in selected settings, while standard deep networks do not always deliver robust gains under low-frequency and noisy financial data [7]. These findings suggest that the key issue is not model complexity alone, but whether the forecasting system is designed, evaluated, and deployed in a disciplined way.

A further motivation comes from a structural weakness in purely financial forecasting. Quarterly financial statements are inherently backward-looking: they summarize realized performance, but they do not fully capture management expectations, strategic adjustments, demand inflections, or emerging innovation cycles at the forecast origin. This limitation becomes more serious as the horizon extends from one quarter to four quarters, where the predictive value of lagging fundamentals may decay substantially.

These ideas shape the research strategy of this thesis. First, a strong quantitative baseline shows what structured financial data alone can do. Second, we add forward-looking narrative information to improve results. This study combines financial fundamentals with earnings call text, using FinBERT and Llama-3 to enhance multi-horizon forecasting.

1.3 Technical Challenges

Designing a reliable multi-horizon forecasting system for corporate quarterly revenue involves several connected technical challenges. In this study, these challenges are treated as key constraints that shape the method design.

1.3.1 Challenge I: Low-frequency and Highly Heterogeneous Quarterly Data

Corporate revenue forecasting is difficult at the quarterly level because the data is limited and highly heterogeneous across firms. Each firm contributes only four observations per year, which limits the amount of usable time-series information even in relatively long historical panels. At the same time, firms in the S&P 500 differ substantially in business model, scale, growth trajectory, cyclicality, and sector structure. A forecasting system must therefore pool information across firms to gain statistical power while still preserving meaningful firm-level differences.

This problem is made harder by the need for strict timing. All input data must be available at the forecast point, and even small mistakes can lead to look-ahead bias. Leakage can arise not only through direct use of future values, but also through premature preprocessing, feature selection, or scaling before the chronological split [8], [9]. Constructing a reliable broad-panel baseline therefore requires both a temporal pipeline and an input design that can represent cross-sectional heterogeneity without sacrificing chronological validity.

1.3.2 Challenge II: Predictive Power Decaying Over Time

Even with a strong next-quarter baseline, forecasting over longer horizons brings another challenge: the predictive power of structured financial data tends to weaken over time. This matters in practice because investors and analysts often care about the next four quarters, not just the next one. Quarterly financial statements are mainly backward-looking. They show past performance, but they do not fully capture management expectations, strategy changes, demand shifts, or new innovation trends. As the forecast horizon extends, this gap becomes more serious.

This leads to the horizon degradation problem: a model that works well at $h = 1$ may perform much worse at $h = 4$. The reason is not only higher uncertainty, but also that the input data becomes less informative over time. This problem is especially serious in fast-changing industries, where past fundamentals are less able to capture future turning points.

1.3.3 Challenge III: Earnings-Call Narratives Are Hard to Turn into Forecasting Signals

To improve long-horizon forecasting, it is natural to use forward-looking qualitative information from earnings call transcripts. However, this creates new challenges. First, the text must be aligned carefully with the forecast timeline. If not, the model may accidentally use future information. Earnings calls often occur after quarter-end but before the next reporting cycle, so an incorrectly aligned text-derived feature can easily leak future information into the model [5].

Second, earnings calls are long, noisy, and full of complex meaning. Turning them into useful forecasting signals requires NLP methods that can capture important managerial information. While FinBERT provides a strong financial sentiment baseline [10], richer generative LLMs can potentially capture more continuous and nuanced narrative gradients. However, these models also require much more computation and memory. Quantization methods such as 4-bit NF4 therefore become important not only for engineering efficiency, but also for making multimodal financial forecasting practically feasible [11], [12]. Therefore, the main challenge is to build a multimodal pipeline that is time-valid, informative, and computationally practical.

1.4 Methodology Overview

This study uses a step-by-step approach to address a key problem: structured financial data works well for short-term forecasts but weakens over longer horizons. It has three stages, and all experiments use strict time-based splits with ablation, interpretability, and reproducibility analysis.

1.4.1 Stage I: Quantitative Baseline Construction

The first methodological stage constructs a rigorous quantitative forecasting baseline for next-quarter corporate revenue prediction. Revenue forecasting is formulated as a supervised panel time-series problem, where the target variable is the log-transformed quarterly revenue of firm i at time t. A global TFT model is trained on a broad panel of 155 continuously listed S&P 500 firms under a strict chronological evaluation framework. To ensure proper timing, all variables are grouped by when they are available, including static firm features, past financial data, and known future calendar inputs.

Within this baseline framework, particular emphasis is placed on structured input design. In addition to autoregressive revenue history, the model incorporates static sector identity, year-over-year growth rates, and scale-related lagged financial variables such as total assets and total equity. This design allows the architecture to capture both temporal dependence and cross-sectional heterogeneity across firms. Controlled ablation analysis is then used to validate the predictive contribution of these feature groups, thereby identifying an empirically effective input configuration for TFT-based corporate revenue forecasting.

1.4.2 Stage II: Expanding to Four-quater Forecasting Horizon

The second methodological stage extends the validated quantitative baseline from one-quarter-ahead forecasting to a four-quarter forecasting horizon. To isolate the effect of horizon length itself, the underlying dataset, chronological split, and core model configuration are held as consistent as possible, while the prediction window is expanded from $h = 1$ to $h = 4$. This controlled extension enables a direct empirical examination of how predictive accuracy changes as the forecasting horizon increases.

Horizon-specific evaluation metrics are computed independently for each forecast step at Horizon ($t + 1, t + 2, t + 3$, and $t + 4$) so that aggregate averages do not conceal long-horizon deterioration. This design makes it possible to quantify the horizon degradation problem in purely financial forecasting and to examine whether the severity of degradation differs across sectors. In this thesis, the sector-level analysis is especially important because it reveals that technology-oriented firms exhibit the strongest long-horizon vulnerability, thereby exposing the structural limitation of relying exclusively on lagging financial statements.

1.4.3 Stage III: Multimodal Narrative Augmentation with FinBERT and Llama-3

The third methodological stage addresses the long-horizon limitations identified in Stage II by introducing a leakage-safe multimodal forecasting framework. This framework augments the structured financial baseline with forward-looking textual signals extracted from quarterly earnings call transcripts. To ensure a focused and computationally feasible multimodal testbed, this stage is conducted on the Mega-Cap 5 technology group, where long-horizon forecasting is particularly challenging due to rapid innovation cycles, structural breaks, and non-linear revenue dynamics.

Two complementary NLP pipelines are employed. First, FinBERT is used to generate domain-specific sentiment features that provide a strong textual baseline. Second, a 4-bit NF4 quantized Llama-3 8B model is deployed locally to extract richer and more continuous narrative features from long-form transcripts. These text-derived signals are integrated into the TFT architecture through a dual-role temporal design,

in which sentiment features function both as observed historical inputs and as forward-filled known-future contextual covariates. This design enables the multimodal model to inject qualitative managerial guidance into multi-horizon forecasting without introducing look-ahead bias.

1.5 Research Contributions

This thesis follows a progressive methodology-driven design: it first establishes a strong TFT-based quantitative baseline for next-quarter forecasting, then expands to four-quarter prediction, and finally introduces multimodal TFT architectures augmented with earnings-call-derived textual signals to stabilize long-horizon revenue forecasts. The main contributions are:

1. To the best of our knowledge, this work presents the first applications of the TFT for quarterly corporate revenue forecasting across S&P 500 firms. Under a strict leakage-free chronological evaluation framework, the proposed model establishes a strong next-quarter quantitative baseline, achieving a test MAPE of 9.31% with RMSE/MAE of 1,973/1,790 million USD, while consistently outperforming LSTM and Box–Jenkins benchmarks.

2. This thesis successfully engineered the optimal feature set for the TFT architecture. By incorporating static sector identity, year-over-year growth rates, and key scale-related financial metrics (such as lagged total assets and equity), the proposed architecture effectively captures cross-sectional heterogeneity. Furthermore, controlled ablation testing explicitly confirms the validity of these selected variables, proving they act as strong predictive drivers that materially reduce aggregate forecast errors across the panel. Academically, this contribution helps clarify which structured inputs are most effective for TFT-based revenue forecasting; practically, it provides a validated input design for real-world corporate forecasting applications.

3. It proposes a multimodal TFT framework that integrates earnings-call-derived textual sentiment through a dual-role temporal design. Building upon a strong domain-specific textual baseline established by FinBERT, we further demonstrate the Llama-3-derived generative Large Language Models. Using a 4-bit NF4 quantized Llama-3 8B model deployed on consumer-grade hardware, the proposed framework extracted continuous, high-fidelity narrative gradients. For example, the pure TFT records a MAPE of 53.85%, while FinBERT+TFT and Llama-3+TFT hybrid reduce the errors to 48.70% and 43.01%, respectively. These results indicate that transcript-derived narrative features materially improve long-horizon forecasting, with the richer Llama-3 representation providing the largest gains.

1.6 Thesis Organization

The remainder of this thesis is organized in a progressive and methodology-driven manner.

- **Chapter 2 (Theoretical Background)** presents the mathematical and conceptual foundations of the thesis. It introduces the principles of multi-horizon time-series forecasting, the TFT architecture, and quantile regression for probabilistic prediction. It also provides the NLP background required for the multimodal extensions, including FinBERT and quantized Llama-3 models.

- **Chapter 3 (Related Work)** situates this thesis within the broader literature on financial forecasting and machine learning. It reviews classical econometric forecasting approaches, deep learning models for financial time series, and recent advances in financial NLP and large language models. This chapter also identifies the research gaps in broad-panel, long-horizon, and multimodal corporate revenue forecasting that motivate the present study.

- **Chapter 4 (Quantitative TFT Forecasting for S&P 500 Firms)** develops the pure quantitative forecasting framework using a broad panel of 155 continuously listed S&P 500 firms. It first establishes a leakage-free TFT baseline for next-quarter revenue prediction and validates the structured input design through ablation analysis. It then extends the same framework to four-quarter-ahead forecasting in order to systematically quantify horizon degradation and examine sector-level differences in long-horizon predictive stability.

- **Chapter 5 (Multimodal TFT Forecasting with Earnings-Call Narratives)** presents the proposed multimodal extension designed to address the long-horizon limitations identified in Chapter 4. It describes the extraction of textual features from earnings-call transcripts using both FinBERT and Llama-3 pipelines, and explains how these narrative signals are integrated into TFT through a leakage-safe dual-role temporal design. The chapter then compares the multimodal models against the pure financial baseline and evaluates their ability to stabilize long-horizon forecasting performance, particularly for the Mega-Cap technology cohort.

- **Chapter 6 (Conclusions and Future Work)** summarizes the main findings of the thesis and evaluates the overall contribution of the proposed forecasting framework. It discusses the implications of the results for applied financial forecasting and institutional decision-making, and concludes by outlining promising directions for future research in multimodal financial machine learning.

Chapter 2

Theoretical Background

This chapter establishes the theoretical and methodological foundations of the thesis, and defines the formal forecasting setting, the principal modeling tools, and the evaluation rules that govern the empirical analysis in later chapters. In particular, this chapter clarifies how quarterly corporate revenue is formulated as a multi-horizon panel forecasting problem, how structured financial covariates are engineered under strict point-in-time constraints, why the TFT is well suited to this setting, and how earnings-call transcripts can be transformed into usable numerical features for multimodal forecasting.

Figure 2.1 provides an overall roadmap of the thesis and situates the methodological components within the broader research design. As shown in the figure, the thesis is organized as a two-stage progression. Chapter 4 develops a purely structured forecasting framework based on historical financial fundamentals and the TFT architecture. Chapter 5 then extends this baseline into a multimodal setting by incorporating earnings-call transcripts and extracting text-derived signals through FinBERT and Llama-3. The role of the present chapter is therefore to supply the conceptual and technical bridge between these later empirical chapters by introducing the shared foundations on which both stages rely.

Guided by this overall framework, the chapter proceeds from structured forecasting foundations to multimodal extensions. It first introduces the target variable, notation, and forecasting objectives for both one-quarter-ahead and four-quarter-ahead prediction. It then presents the feature-engineering principles used to construct stable, leakage-aware structured covariates from quarterly fundamentals. After that, it reviews the benchmark model families used in this thesis, including classical statistical baselines and recurrent neural networks. The chapter then turns to the TFT architecture, emphasizing its treatment of mixed covariates, variable selection mechanism, and interpretable multi-horizon forecasting design [4].

The second half of the chapter introduces the natural language processing (NLP) foundations required for the multimodal extension. It reviews Transformer-based language modeling, FinBERT for financial sentiment extraction, and prompt-based feature extraction with generative large language models. Because the multimodal component of this thesis relies on transcript-derived features that must remain both temporally valid and practically deployable, the chapter also discusses transcript alignment rules, local deployment constraints, and the role of low-bit quantization in making large models usable under limited hardware resources [11]. Finally, it defines the evaluation framework, including point-forecast metrics, probabilistic metrics, and the leakage-control principles applied throughout all experiments.

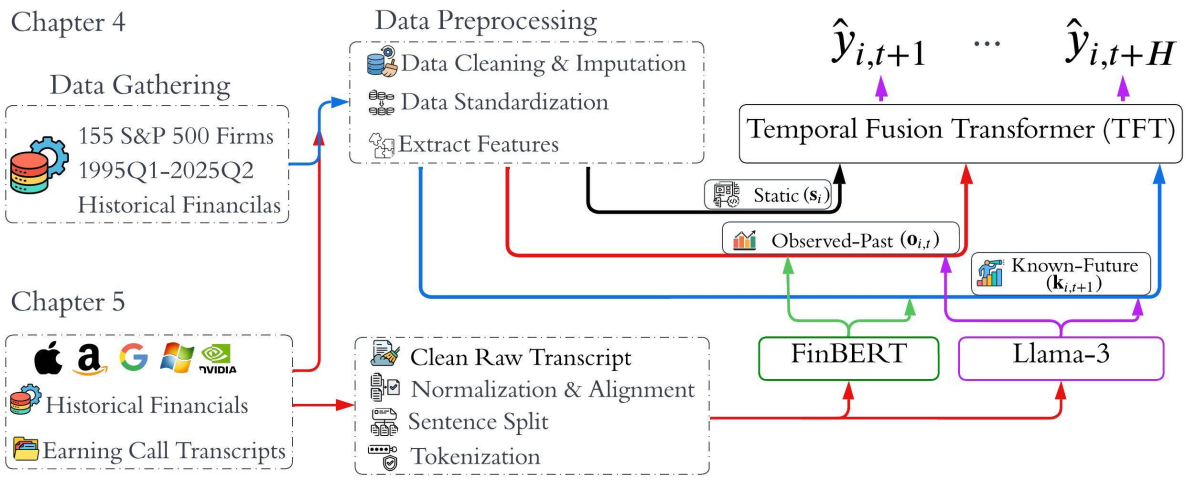


Figure 2.1: Overall thesis forecasting framework

2.1 Problem Formulation and Forecasting Objective

This section formalizes the forecasting problem studied in this thesis and introduces the notation used in later chapters. The target variable is the natural logarithm of quarterly corporate revenue. For firm i in calendar quarter t,

$$y_{i,t} \triangleq \log(\text{Revenue}_{i,t}). \tag{2.1}$$

The log transformation is used to reduce scale disparities across firms and to make growth patterns easier to model, since corporate revenue can differ by orders of magnitude and often evolves in a multiplicative rather than purely additive manner [1], [13].

Revenue forecasting is treated as a supervised panel time-series problem. At forecast origin t, the model observes a historical window of past targets together with a set of structured covariates that are available at that time. Following the TFT formulation, inputs are separated according to their temporal availability:

- **Observed-past features** $\mathbf{o}_{i,t}$: time-varying variables known only up to time t.
- **Known-future features** $\mathbf{k}_{i,t}$: time-varying variables whose future values are known in advance.
- **Static attributes** $\mathbf{s}_i$: time-invariant firm descriptors.

Let L denote the encoder length, or historical look-back window. The target history is written as $y_{i,t-L+1:t} \triangleq (y_{i,t-L+1}, \ldots, y_{i,t})$, and the aligned observed-past sequence is $\mathbf{o}_{i,t-L+1:t} \triangleq (\mathbf{o}_{i,t-L+1}, \ldots, \mathbf{o}_{i,t})$. This notation allows the forecasting task to be defined consistently across single-horizon, multi-horizon, and multimodal settings.

2.1.1 Single-Horizon Forecasting

The first forecasting objective considered in this thesis is one-step-ahead prediction. For quarter-ahead forecasting, the model predicts the next revenue value at $t + 1$ using only information available at time t. Formally, the model learns a mapping f_θ:

$$\hat{y}_{i,t+1} = f_\theta\Big(\underbrace{y_{i,t-L+1:t}}_{\text{target history}}, \underbrace{\mathbf{o}_{i,t-L+1:t}}_{\text{observed–past}}, \underbrace{\mathbf{k}_{i,t+1}}_{\text{known–future}}, \underbrace{\mathbf{s}_i}_{\text{static}} \Big), \tag{2.2}$$

where θ denotes the learnable parameters. This single-horizon formulation provides the initial quantitative benchmark later developed in Chapter 4.

2.1.2 Multi-Horizon Forecasting

Because the practical forecasting demand in finance is rarely limited to the next quarter alone, this thesis also considers multi-horizon prediction. Let H denote the maximum forecasting horizon, with $H = 4$ in this thesis. The model then predicts a full sequence of future values:

$$\hat{\mathbf{y}}_{i,t+1:t+H} = f_\theta\Big(\underbrace{y_{i,t-L+1:t}}_{\text{target history}}, \underbrace{\mathbf{o}_{i,t-L+1:t}}_{\text{observed–past}}, \underbrace{\mathbf{k}_{i,t+1:t+H}}_{\text{known–future}}, \underbrace{\mathbf{s}_i}_{\text{static}} \Big), \tag{2.3}$$

where $\hat{\mathbf{y}}_{i,t+1:t+H} \triangleq (\hat{y}_{i,t+1}, \ldots, \hat{y}_{i,t+H})$ and $\mathbf{k}_{i,t+1:t+H}$ contains all known-future inputs across the prediction window. This is the standard multi-horizon forecasting setup used by TFT and forms the basis for the horizon-degradation analysis later conducted in Chapter 4.

2.1.3 Multimodal Sentiment-Augmented Forecasting

Structured financial statements are inherently backward-looking, whereas earnings calls often contain forward-looking managerial language and narrative guidance [5]. Recent work also shows that local large language models can be prompted to extract continuous structured signals from earnings-call transcripts, such as sentiment and forward-looking orientation, which can then be used as quantitative covariates [14]. Motivated by this, the forecasting objective is extended to include text-derived features.

Let $\mathbf{o}^{\text{num}}$ denote the structured numerical covariates, and let $\mathbf{u}^{\text{sent}}$ denote transcript-derived sentiment or narrative features observed up to time t. Calendar covariates are denoted by $\mathbf{k}^{\text{cal}}$. In addition, a forward-filled sentiment context term $\mathbf{k}^{\text{sent}}$ can be constructed at the forecast origin and treated as available across the future horizon by design. The resulting multimodal forecasting objective is:

$$\hat{\mathbf{y}}_{i,t+1:t+H} = f_\theta\Bigg(\underbrace{y_{i,t-L+1:t}}_{\text{target history}}, \underbrace{\big[\mathbf{o}^{\text{num}}_{i,t-L+1:t} \oplus \mathbf{u}^{\text{sent}}_{i,t-L+1:t}\big]}_{\text{observed inputs}}, \underbrace{\big[\mathbf{k}^{\text{cal}}_{i,t+1:t+H} \oplus \mathbf{k}^{\text{sent}}_{i,t+1:t+H}\big]}_{\text{known future}}, \underbrace{\mathbf{s}_i}_{\text{static}} \Bigg) \tag{2.4}$$

where:

- $y_{i,t-L+1:t}$ denotes the historical revenue target sequence;

- $\mathbf{o}^{\text{num}}$ denotes historically observed numerical covariates;
- $\mathbf{u}^{\text{sent}}$ denotes historically observed text-derived features;
- $\mathbf{k}^{\text{cal}}$ denotes deterministic calendar covariates;
- $\mathbf{k}^{\text{sent}}$ denotes the forward-filled sentiment context used in the decoder; and
- $\mathbf{s}_i$ denotes static firm identity and related categorical attributes.

This unified formulation allows structured fundamentals and narrative signals to be modeled jointly under a single leakage-aware multi-horizon framework. The operational details of this hybrid design are developed later in Chapter 5.

2.2 Financial Feature Engineering

This section explains how raw quarterly fundamentals and calendar information are transformed into model-ready structured covariates. The main design goals are threefold: first, to create features whose scale is numerically stable across firms of very different size; second, to preserve economically meaningful temporal patterns such as seasonality, momentum, and medium-run growth; and third, to ensure that all features satisfy strict point-in-time validity.

2.2.1 Log Transformations, Lags, and Year-Over-Year Features

Quarterly financial variables often vary dramatically across firms. Large-cap firms may have revenue and balance-sheet values that are several orders of magnitude larger than those of smaller firms, and many accounting variables evolve multiplicatively rather than additively. To reduce scale-related instability and heteroskedasticity, logarithmic transformations are applied to major level variables such as revenue, total assets, total equity, and operating expenses. Log transformations are standard in forecasting when the variance of a series grows with its level, and they also allow changes to be interpreted approximately in relative terms.

Lagged variables are also included to represent recent temporal dependence. Examples include one-quarter and two-quarter lags of the target and selected fundamentals. Such lagged predictors are a standard way to capture delayed effects, momentum, and short-run persistence in time series models [13].

Because the target is quarterly, seasonal effects are important. To capture medium-run growth while filtering out quarter-specific seasonality, year-over-year (YoY) features are computed by comparing the current quarter with the same quarter one year earlier. In log space, this corresponds to a seasonal difference with lag 4:

$$\text{YoY}_{i,t}^{(\log)} = y_{i,t} - y_{i,t-4}. \tag{2.5}$$

Seasonal differencing is widely used to reduce recurring seasonal structure and to highlight changes relative to the same quarter in the previous year. In this thesis, lagged values and YoY features play complementary

roles: lags capture short-run dynamics, while YoY features encode medium-run growth after seasonal adjustment.

2.2.2 Covariate Taxonomy for Multi-Horizon Forecasting

A central principle of this thesis is that covariates must be organized according to what is known at the forecast origin. Following [4], the structured inputs are grouped into three categories as Table 2.1:

Table 2.1: Illustrative covariate taxonomy for quarterly revenue forecasting.

Type	**Examples**
Static $\mathbf{s}_i$	Sector/industry label, firm identifier embedding
Observed past $\mathbf{o}_{i,t}$	Lagged revenue, lagged margins, lagged YoY growth, trailing ratios
Known future $\mathbf{k}_{i,t}$	Quarter-of-year dummies, fiscal quarter index, deterministic calendar features

This taxonomy is not merely a modeling convenience; it is also a leakage-control mechanism. Known-future variables can legitimately be supplied across all forecast horizons, whereas observed-past variables must stop at the forecast origin. This strict separation ensures that the learning problem matches the information set that would be available in real deployment.

2.2.3 Time Indexing and Seasonality Encoding

Calendar variables provide simple but important structure in quarterly forecasting. This thesis uses quarter-of-year indicators together with a running time index to encode recurring seasonal patterns and slow-moving temporal drift. In regression-style forecasting, seasonal dummy variables are a standard way to represent regular seasonal effects; for quarterly data, three dummy variables are usually sufficient. Within the TFT framework, these variables naturally belong to the known-future covariate group because their future values are deterministic.

2.2.4 Global vs. Local Modeling in Panel Forecasting

With panel data, two broad training strategies are common. A *local* strategy fits a separate model for each firm, whereas a *global* strategy fits one shared model across all firms and conditions on firm-level descriptors. Local models can capture idiosyncratic firm behavior, but they are often data-hungry and unstable when each series is short, as is typical with quarterly data. Global models, by contrast, share statistical strength across firms and can generalize better in large panels composed of many short series [15], [16].

Because each firm contributes only a limited number of quarterly observations, this thesis adopts a global modeling setup. Firm-level heterogeneity is preserved through static descriptors such as sector identity and firm embeddings, while the shared model benefits from the larger pooled dataset. This choice is especially important for the broad-panel structured baseline in Chapter 4.

2.3 Benchmark Models: Statistical and Recurrent Baselines

To evaluate whether the proposed TFT-based framework offers a meaningful advance, this thesis compares it against two families of widely used baselines: classical univariate time-series models and recurrent neural networks. These baselines provide complementary reference points. The statistical models offer transparency and strong performance in relatively stable settings, while recurrent neural networks serve as standard deep learning baselines for sequential data.

2.3.1 ARIMA and SARIMA

ARIMA is a standard univariate forecasting model. An $ARIMA(p, d, q)$ specification combines: (i) autoregressive terms based on past observations, (ii) differencing to reduce non-stationarity, and (iii) moving-average terms based on past forecast errors [1], [13]. Using the backshift operator B (where $By_t = y_{t-1}$), the model can be written as

$$\phi(B)(1 - B)^d y_t = c + \theta(B)\varepsilon_t, \tag{2.6}$$

where y_t is the target, c is a constant, ε_t is white noise, $\phi(B) = 1 - \phi_1 B - \cdots - \phi_p B^p$ is the AR polynomial, and $\theta(B) = 1 + \theta_1 B + \cdots + \theta_q B^q$ is the MA polynomial.

Quarterly revenue often exhibits seasonal structure, so Seasonal ARIMA is also considered. A SARIMA(p,d,q) $(P, D, Q)_s$ model adds seasonal autoregressive and moving-average components together with seasonal differencing. For quarterly data, the seasonal period is $s = 4$, and the model takes the form

$$\Phi(B^s)\phi(B)(1 - B^s)^D(1 - B)^d y_t = c + \Theta(B^s)\theta(B)\varepsilon_t, \tag{2.7}$$

where $\Phi(B^s)$ and $\Theta(B^s)$ denote the seasonal AR and MA polynomials.

In this thesis, ARIMA and SARIMA are used as univariate benchmarks without exogenous regressors. Their orders are selected using the validation block, and final performance is reported on the held-out test block under the same chronological evaluation protocol applied to the deep learning models.

2.3.2 Recurrent Architectures: RNN and LSTM

Recurrent neural networks (RNNs) model sequential data by updating a hidden state over time. However, standard RNNs often struggle with long-range dependencies because gradients can vanish or explode during training [17]. Gated recurrent models were proposed to mitigate this problem. As illustrated in Figure 2.2, the LSTM architecture introduces an internal cell state together with input, forget, and output gates that regulate the flow of information through time [3]. Similarly, the Gated Recurrent Unit (GRU) provides a more compact gating structure and often serves as a practical alternative [18], [19].

Let $\mathbf{x}_t \in \mathbb{R}^{d_x}$ denote the input vector at quarter t, and let L denote the look-back length. A standard sequence-to-one recurrent baseline produces the next-quarter forecast using the input sequence over the historical window:

$$\hat{y}_{t+1} = g_\theta(\mathbf{x}_{t-L+1}, \ldots, \mathbf{x}_t), \tag{2.8}$$

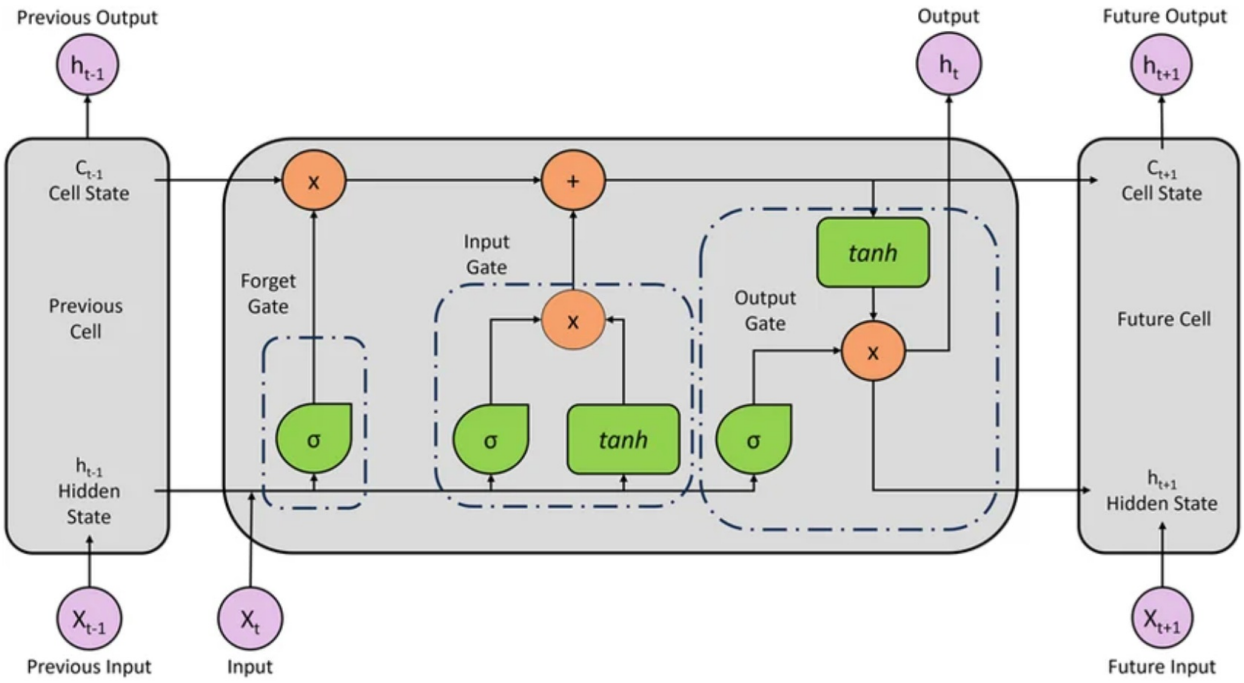


Figure 2.2: Architecture of a LSTM neural network cell.

where g_θ represents the recurrent model parameterized by θ.

LSTM update equations. At each time step t, an LSTM updates a hidden state $\mathbf{h}_t$ and a cell state $\mathbf{c}_t$:

$$\mathbf{i}_t = \sigma(W_i \mathbf{x}_t + U_i \mathbf{h}_{t-1} + \mathbf{b}_i)\,, \tag{2.9}$$

$$\mathbf{f}_t = \sigma\big(W_f \mathbf{x}_t + U_f \mathbf{h}_{t-1} + \mathbf{b}_f\big)\,, \tag{2.10}$$

$$\mathbf{o}_t = \sigma(W_o \mathbf{x}_t + U_o \mathbf{h}_{t-1} + \mathbf{b}_o)\,, \tag{2.11}$$

$$\tilde{\mathbf{c}}_t = \tanh(W_c \mathbf{x}_t + U_c \mathbf{h}_{t-1} + \mathbf{b}_c)\,, \tag{2.12}$$

$$\mathbf{c}_t = \mathbf{f}_t \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \tilde{\mathbf{c}}_t, \tag{2.13}$$

$$\mathbf{h}_t = \mathbf{o}_t \odot \tanh(\mathbf{c}_t), \tag{2.14}$$

where $\sigma(\cdot)$ is the sigmoid function and $\odot$ denotes element-wise multiplication [3].

Output layer and training objective. A common output layer maps the final hidden state to the next prediction:

$$\hat{y}_{t+1} = W_y \mathbf{h}_t + b_y. \tag{2.15}$$

Training can use mean squared error or a more robust alternative such as the Huber loss [20]:

$$\mathcal{L}_\delta(y, \hat{y}) = \begin{cases} \frac{1}{2}(y - \hat{y})^2, & |y - \hat{y}| \le \delta, \\ \delta |y - \hat{y}| - \frac{1}{2}\delta^2, & |y - \hat{y}| > \delta. \end{cases} \tag{2.16}$$

Limitations of recurrent baselines. Recurrent models remain useful baselines, but they have several limitations in quarterly corporate forecasting:

1. **Long-range dependence remains difficult:** even with gating, very long dependencies can be challenging to learn in low-frequency and noisy financial series [17].
2. **Multi-horizon prediction is not especially natural:** many recurrent baselines rely on recursive forecasting or sequence-to-sequence wrappers, which can accumulate errors as the horizon grows.
3. **Interpretability is limited:** predictions are mediated through hidden states that are difficult to inspect, which is a disadvantage in governance-constrained financial applications.

2.4 The Temporal Fusion Transformer

As illustrated in Figure 2.3, the TFT is a DL architecture explicitly designed for multi-horizon forecasting with mixed covariates [4]. This makes it well aligned with the forecasting problem studied in this thesis, where the model must combine firm-level static attributes, historical financial variables, and deterministic future calendar inputs under a unified temporal framework.

At a high level, TFT combines three important design ideas. First, it separates inputs based on their temporal availability and processes them through dedicated variable-selection networks. Second, it uses a lightweight recurrent layer to model local temporal dynamics. Third, it applies interpretable attention to capture longer-range dependencies and produce horizon-specific representations. This combination allows TFT to move beyond purely black-box sequence modeling while retaining the flexibility of modern DL.

2.4.1 Input Structure and Covariate Roles

A defining feature of TFT is that it does not treat all inputs symmetrically. Instead, it separates covariates according to their time availability at forecast origin as shown in Table 2.1.

This separation is important both theoretically and operationally. It aligns the architecture with deployable information sets and helps prevent leakage by construction: future financial variables cannot be passed into the known-future branch simply because they are not available at time t.

2.4.2 Gating Mechanisms: GLU and GRN

Financial covariates are noisy, and their usefulness can vary across firms and over time. TFT controls this complexity through gating mechanisms. The basic building block is the Gated Linear Unit (GLU) [4]:

$$\mathrm{GLU}_{\omega}(\gamma) = \sigma(\mathbf{W}_{4,\omega}\gamma + \mathbf{b}_{4,\omega}) \odot (\mathbf{W}_{5,\omega}\gamma + \mathbf{b}_{5,\omega}), \tag{2.17}$$

where $\sigma(\cdot)$ denotes the sigmoid function and $\odot$ denotes element-wise multiplication, ω define as layer-specific parameters.

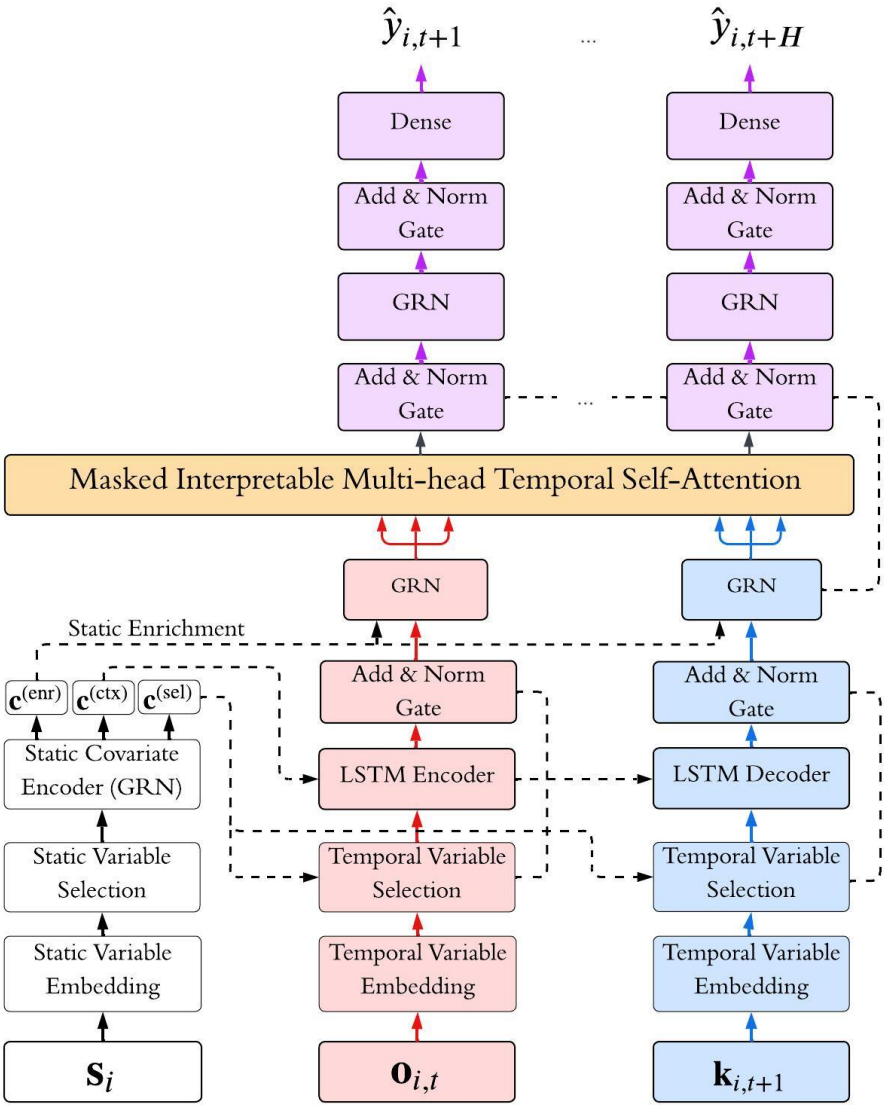


Figure 2.3: High-level architecture of the TFT.

The GLU is used within a Gated Residual Network (GRN), which augments the nonlinear transformation with a residual connection and layer normalization [4]:

$$\mathrm{GRN}_\omega(\mathbf{a}, \mathbf{c}) = \mathrm{LayerNorm}\Big(\mathbf{a} + \mathrm{GLU}_\omega(\boldsymbol{\eta}_1)\Big), \tag{2.18}$$

$$\boldsymbol{\eta}_1 = \mathbf{W}_{1,\omega}\boldsymbol{\eta}_2 + \mathbf{b}_{1,\omega}, \tag{2.19}$$

$$\boldsymbol{\eta}_2 = \mathrm{ELU}\Big(\mathbf{W}_{2,\omega}\mathbf{a} + \mathbf{W}_{3,\omega}\mathbf{c} + \mathbf{b}_{2,\omega}\Big), \tag{2.20}$$

where $\mathbf{a}$ is the main input and $\mathbf{c}$ is an optional context vector. The residual path allows the network to fall back toward a simpler transformation when a more complex nonlinear mapping is unnecessary, which can improve training stability and reduce overfitting.

2.4.3 Variable Selection Networks

Not all covariates are equally informative in every firm-quarter. TFT addresses this using Variable Selection Networks (VSNs), which assign context-dependent weights to the available input representations [4]. Let $\boldsymbol{\Xi}_t$ denote the concatenated variable representations at time t, and let $\mathbf{c}^{(\mathrm{sel})}$ denote the selection context. The variable-selection weights are defined as

$$\mathbf{v}_{x,t} = \mathrm{Softmax}\Big(\mathrm{GRN}_{v_x}(\boldsymbol{\Xi}_t, \mathbf{c}^{(\mathrm{sel})})\Big). \tag{2.21}$$

Each variable representation $\boldsymbol{\xi}_t^{(j)}$ is transformed by its own GRN:

$$\tilde{\boldsymbol{\xi}}_t^{(j)} = \mathrm{GRN}_{\tilde{\xi}(j)}\big(\boldsymbol{\xi}_t^{(j)}\big), \tag{2.22}$$

and the final VSN output is a weighted sum:

$$\tilde{\boldsymbol{\xi}}_t = \sum_{j=1}^{m_x} v_{x,t}^{(j)}\, \tilde{\boldsymbol{\xi}}_t^{(j)}. \tag{2.23}$$

This mechanism is especially useful in financial forecasting because the predictive value of features can shift across sectors, time periods, and forecasting horizons.

2.4.4 Sequence Processing and Static Enrichment

After variable selection, TFT uses an LSTM encoder–decoder to capture local temporal dynamics [3], [4]. A gated skip connection regulates how strongly the recurrent layer modifies the representation:

$$\tilde{\boldsymbol{\phi}}(t,n) = \mathrm{LayerNorm}\Big(\tilde{\boldsymbol{\xi}}_{t+n} + \mathrm{GLU}_{\phi}(\boldsymbol{\phi}(t,n))\Big). \tag{2.24}$$

TFT then applies static enrichment so that temporal representations are conditioned on firm-level context. Given a static context vector $\mathbf{c}^{(\mathrm{enr})}$,

$$\boldsymbol{\theta}(t,n) = \mathrm{GRN}_{\theta}\big(\tilde{\boldsymbol{\phi}}(t,n), \mathbf{c}^{(\mathrm{enr})}\big). \tag{2.25}$$

This allows the model to process temporal information in a way that depends on stable firm attributes such as sector or latent firm identity.

2.4.5 Interpretable Multi-Head Attention

To capture longer-range dependencies, TFT applies an interpretable form of multi-head attention over the encoded sequence [4]. The core attention computation follows the scaled dot-product form introduced in the Transformer literature [21]:

$$\mathrm{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \mathrm{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^{\top}}{\sqrt{d_k}}\right)\mathbf{V}, \tag{2.26}$$

TFT modifies this attention mechanism to improve interpretability and then refines the attended representation through additional GRN blocks [4]:

$$\boldsymbol{\psi}(t,n) = \mathrm{GRN}_{\psi}\big(\boldsymbol{\delta}(t,n)\big), \tag{2.27}$$

$$\tilde{\boldsymbol{\psi}}(t,n) = \mathrm{LayerNorm}\Big(\tilde{\boldsymbol{\phi}}(t,n) + \mathrm{GLU}_{\tilde{\psi}}\big(\boldsymbol{\psi}(t,n)\big)\Big). \tag{2.28}$$

These horizon-specific representations are then used to generate the final forecasts.

2.4.6 Quantile Regression and Uncertainty Estimation

A practical advantage of TFT is that it supports probabilistic forecasting through quantile prediction. Instead of outputting only a point estimate, the model can predict conditional quantiles and optimize the quantile, or pinball, loss [4], [22]:

$$\mathcal{L}_q(y,\hat{y}) = \max\big(q(y-\hat{y}),\ (1-q)(\hat{y}-y)\big), \tag{2.29}$$

where $q \in (0,1)$. By predicting multiple quantiles, such as 0.1, 0.5, and 0.9, the model produces prediction intervals that are useful for uncertainty-aware financial decision-making.

2.5 Financial Natural Language Processing

Structured accounting variables are useful, but they do not capture many of the forward-looking signals embedded in managerial language. Earnings calls, annual reports, and financial news frequently contain guidance, risk discussion, and qualitative framing that may affect expectations before those effects become visible in reported financial statements [5]. For this reason, modern forecasting systems increasingly combine numerical fundamentals with unstructured text.

Early approaches to financial text analysis often relied on dictionary-based methods, which counted words from predefined positive and negative lists. These methods are simple and interpretable, but they are also brittle. They ignore context, negation, and modality, and they can mis-handle domain-specific language. In finance, words such as "liability," "capital," or "tax" are not inherently negative, which makes generic sentiment dictionaries unreliable in many settings [23].

Representation learning methods address these limitations by learning contextual text representations rather than relying on isolated word counts. This shift is closely tied to the rise of Transformer architectures, which model long-range dependencies through self-attention [21].

2.5.1 Transformer Models and BERT

Transformers replace sequential recurrence with self-attention, allowing each token to attend directly to other tokens in the sequence [21]. Given an input token sequence $\mathbf{x} = (x_1, \ldots, x_T)$, a Transformer encoder produces contextualized embeddings $\mathbf{H} = (\mathbf{h}_1, \ldots, \mathbf{h}_T)$ through stacked attention and feed-forward layers. The core attention computation follows the same scaled dot-product mechanism defined earlier in Eq. (2.26), where $\mathbf{Q}$, $\mathbf{K}$, and $\mathbf{V}$ are query, key, and value matrices, and d_k is the key dimension. Self-attention is particularly

useful for financial text because the meaning of a sentence may depend on qualifiers, risk statements, or hedging expressions appearing many tokens away.

BERT is a bidirectional Transformer encoder trained using masked language modeling (MLM) [24]. Let $\mathcal{M}$ denote the masked positions in a sequence. The model objective is:

$$\max_{\theta} \sum_{t \in \mathcal{M}} \log p_\theta(x_t \mid \mathbf{x}_{\setminus \mathcal{M}}), \tag{2.30}$$

where $\mathbf{x}_{\setminus \mathcal{M}}$ denotes the corrupted sequence and θ denotes the model parameters. Because the model uses both left and right context, it can capture patterns such as negation, modality, and forward-looking phrasing that are common in corporate disclosure language.

2.5.2 FinBERT for Financial Sentiment

Generic language models are often suboptimal for financial text because the financial domain uses a specialized vocabulary and distinctive contextual patterns. FinBERT adapts the BERT architecture to finance by domain-specific pretraining and subsequent fine-tuning for sentiment classification [10].

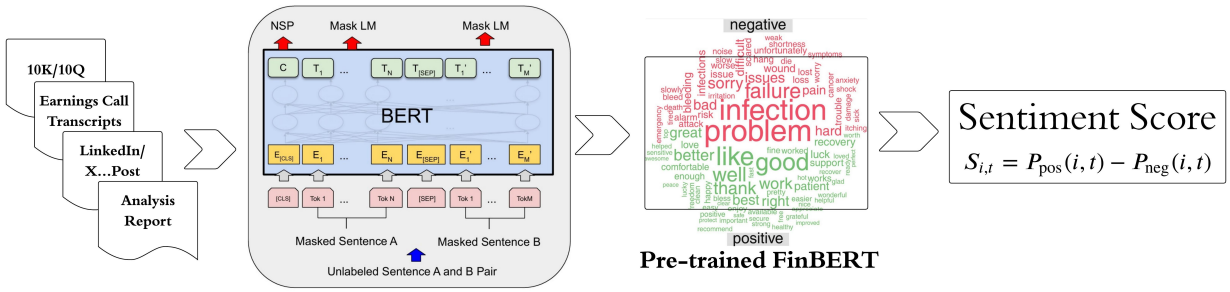


Figure 2.4: The FinBERT processing pipeline.

As illustrated in Figure 2.4, the development and deployment of this model follow a rigorous two-stage pipeline. First, the generic BERT architecture undergoes domain-specific pre-training on a massive corpus of financial texts (such as financial news, analyst reports, and corporate filings) to internalize the contextual intricacies of market language. Following this, the model is fine-tuned explicitly for the downstream task of financial sentiment classification.

When deployed in our multimodal framework, this fine-tuned pipeline processes the raw text of corporate earnings calls. Rather than simply returning a binary label, the final classification layer outputs a normalized probability distribution across three distinct sentiment classes: Positive (P_{pos}), Negative (P_{neg}), and Neutral. As defined in our problem formulation, these output probabilities are subsequently aggregated to compute the net sentiment score ($Sen_{i,t}$), transforming unstructured executive narratives into a structured, quantitative covariate for the predictive model.

Given an input text $\mathbf{x}$, FinBERT uses the pooled representation of the `[CLS]` token:

$$\mathbf{h}_{\texttt{[CLS]}} = \text{Enc}_\theta(\mathbf{x})_{\texttt{[CLS]}}. \tag{2.31}$$

A linear classifier then outputs class probabilities over the sentiment classes {positive, neutral, negative}:

$$\mathbf{p} = \text{softmax}\left(\mathbf{W}\mathbf{h}_{\texttt{[CLS]}} + \mathbf{b}\right), \qquad \mathbf{p} \in \mathbb{R}^3. \tag{2.32}$$

With labeled training data $\{(\mathbf{x}^{(n)}, y^{(n)})\}_{n=1}^{N}$, the model is typically optimized using cross-entropy:

$$\min_{\theta,\mathbf{W},\mathbf{b}} -\sum_{n=1}^{N} \log p_\theta\left(y^{(n)} \mid \mathbf{x}^{(n)}\right). \tag{2.33}$$

In this thesis, the class probabilities are converted into a continuous sentiment covariate. Specifically, the net sentiment score is defined as

$$Sen_{i,t} = P_{\text{pos}}(i,t) - P_{\text{neg}}(i,t), \tag{2.34}$$

and segment-level scores are aggregated across the transcript, for example by averaging. This yields a bounded continuous feature in $[-1, 1]$ that can be incorporated into the forecasting model as a structured time-varying covariate.

2.5.3 Generative Large Language Models and Llama-3

While FinBERT is well suited to discrete sentiment classification, longer and more nuanced corporate narratives often contain information that cannot be reduced to a simple three-class label. Recent surveys of financial NLP emphasize that generative large language models can be used not only for summarization, but also for structured feature extraction from complex financial text [25]. Through prompting, such models can produce continuous, task-specific outputs related to sentiment, forward-looking intensity, uncertainty, vagueness, or narrative emphasis.

This thesis uses Llama-3 as the generative backbone for richer transcript-based feature extraction. As detailed in Figure 2.5, Llama-3 is a decoder-only Transformer model designed for autoregressive language generation [26]. In a decoder-only architecture, each token is generated sequentially based on the preceding context, which makes the model especially well suited to instruction-following and structured output generation. This makes decoder-only models particularly useful when the task is not merely to classify a transcript, but to translate a long narrative into multiple prompt-defined numerical attributes. Unlike encoder-only models such as BERT, which are primarily optimized for contextual representation learning and classification, Llama-3 can be prompted to extract multiple continuous, task-specific features from long-form earnings-call transcripts. In this thesis, this capability is used to derive richer narrative covariates, such as sentiment intensity and forward-looking orientation, that complement structured financial fundamentals in the multi-horizon forecasting pipeline.

A practical reason for using an open-weights model such as Llama-3 is that it supports local deployment. In financial settings, local inference can be attractive because it improves data privacy, avoids dependence on external APIs, and makes the extraction pipeline more reproducible [14]. These considerations are especially relevant when the pipeline must process large numbers of firm-quarter transcripts under stable model settings.

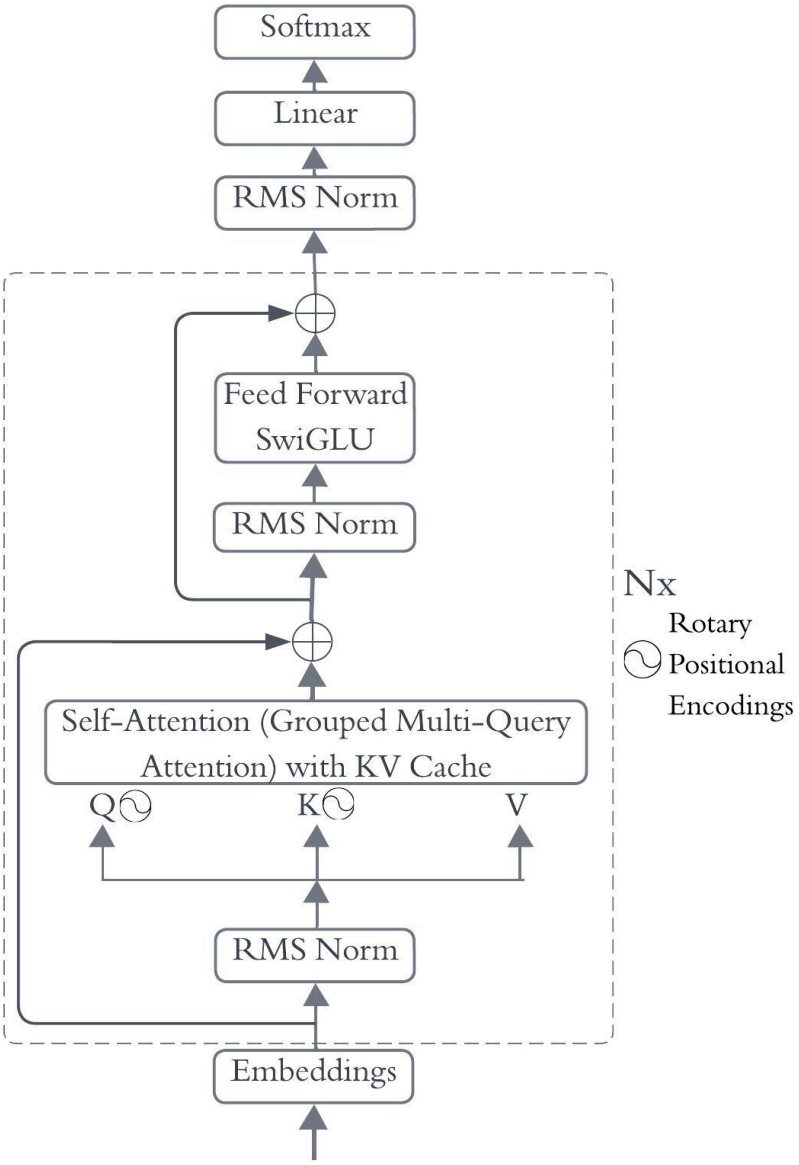


Figure 2.5: High-level architecture of the Llama-3 generative model

Hardware Constraints and Low-Bit Quantization

Earnings-call transcripts are often long enough to stress the context limits and memory budget of local models, so real systems frequently rely on chunking or multi-stage processing [5], [14]. Hardware limitations are an additional concern. Low-bit quantization is a widely used solution for reducing memory requirements, and QLoRA introduces 4-bit NormalFloat (NF4) quantization as a practical way to preserve much of a model's useful capacity while lowering its computational footprint [11]. Finance-oriented studies also report practical benefits of QLoRA-style configurations in prediction tasks [12]. These findings motivate the quantized local

LLM setup adopted in this thesis.

Generative Feature Extraction from Earnings Calls

Generative models can be used to extract multiple continuous dimensions from text, not only sentiment. For example, [14] prompts large language models to score earnings-call narratives along axes such as overall sentiment, forward-looking orientation, and vagueness, and then aggregates those scores into continuous measures. This creates a bridge from unstructured transcript language to structured numeric covariates.

This thesis follows the same general idea. LLM-based outputs are treated as structured features derived from text and aligned to the corresponding firm-quarter. These features are then used to complement structured financial fundamentals in the multi-horizon forecasting framework developed in Chapter 5.

2.5.4 Temporal Alignment of Text with Quarterly Fundamentals

Combining text with quarterly structured data introduces a serious leakage risk if timing is handled incorrectly. Earnings calls typically occur after quarter-end and often contain both discussion of realized results and guidance about the future [5]. As a result, transcript-derived features must be aligned using actual call dates and strict availability rules.

More generally, all evaluation must follow the arrow of time. Chronological splits and point-in-time preprocessing are necessary because random resampling can leak future context into the training process [8], [9], [13]. This thesis enforces these timing rules so that reported multimodal results correspond to information that would actually have been available at the forecast origin.

2.6 Evaluation Framework and Metrics

This section defines the evaluation framework used throughout the thesis. The goal is to ensure that reported performance is both statistically meaningful and operationally realistic. Although some models are trained on transformed targets, such as log revenue, final results must be reported on an interpretable scale. Therefore, predictions are transformed back to the revenue scale before evaluation [13]. This allows errors to be interpreted directly in dollar terms and supports fair comparison across competing models.

Because forecasting is inherently directional in time, all evaluation follows chronological backtesting rules. No random shuffling is used, since such procedures can introduce look-ahead bias and overstate model performance [8], [9].

2.6.1 Point Forecast Metrics

Three standard point-forecast metrics are reported: mean absolute error (MAE), root mean squared error (RMSE), and mean absolute percentage error (MAPE). Let $\mathcal{T}_{\text{test}}$ denote the set of test instances indexed by (i, t, h), where h is the forecast horizon. Let $y_{i,t+h}$ denote the realized revenue on the original scale and $\hat{y}_{i,t+h}$ the corresponding forecast.

MAE measures the average absolute deviation and is relatively robust to occasional large misses:

$$\mathrm{MAE} = \frac{1}{|\mathcal{T}_{\text{test}}|} \sum_{(i,t,h)\in\mathcal{T}_{\text{test}}} \left| y_{i,t+h} - \hat{y}_{i,t+h} \right|. \tag{2.35}$$

RMSE penalizes larger forecast errors more heavily, making it sensitive to tail misses:

$$\mathrm{RMSE} = \sqrt{\frac{1}{|\mathcal{T}_{\text{test}}|} \sum_{(i,t,h)\in\mathcal{T}_{\text{test}}} \left(y_{i,t+h} - \hat{y}_{i,t+h} \right)^2}. \tag{2.36}$$

MAPE expresses forecast error in percentage terms and is useful when comparing firms of different scale:

$$\mathrm{MAPE}(\varepsilon) = \frac{100}{|\mathcal{T}_{\text{test}}|} \sum_{(i,t,h)\in\mathcal{T}_{\text{test}}} \left| \frac{y_{i,t+h} - \hat{y}_{i,t+h}}{\max(\varepsilon,\ |y_{i,t+h}|)} \right|, \tag{2.37}$$

with $\varepsilon = 10^{-6}$ for numerical stability.

Because MAPE can be unstable when the denominator is close to zero, it is always interpreted together with MAE and RMSE [13], providing a more balanced view of percentage error and absolute dollar error. *Note:* Unless otherwise stated, all RMSE and MAE values in this thesis are expressed in million USD.

2.6.2 Probabilistic Forecast Metrics

TFT can output predictive quantiles, such as $q \in \{0.1, 0.5, 0.9\}$. In that case, point metrics alone are insufficient, and uncertainty quality must also be assessed.

Quantile loss (pinball loss). For target quantile level q, the quantile loss is [22]:

$$\mathcal{L}_q(y, \hat{y}^{(q)}) = \max\big(q(y - \hat{y}^{(q)}),\ (1-q)(\hat{y}^{(q)} - y)\big), \tag{2.38}$$

where $\hat{y}^{(q)}$ is the predicted q-th conditional quantile. This loss is averaged across horizons and test instances and is consistent with the probabilistic forecasting objective used by TFT [4].

Coverage and interval width. Prediction intervals should also be calibrated. For example, a 10–90% interval should contain roughly 80% of the realized outcomes if the model is well calibrated. This thesis therefore reports: (i) empirical coverage, defined as the fraction of realized values falling within $[\hat{y}^{(0.1)}_{i,t+h}, \hat{y}^{(0.9)}_{i,t+h}]$, and (ii) average interval width. Together, these metrics assess whether the model's uncertainty estimates are both reliable and practically useful.

2.6.3 Temporal Validation and Leakage Control

All experiments in this thesis follow a time-respecting backtesting design. Two rules are central **(i) Rolling or expanding-window evaluation:** models are trained only on data available up to time t and tested on later periods. This matches the true forecasting setting and avoids leakage from random resampling [8], [13].

(ii) Strict feature availability at time t: every covariate used at the forecast origin must be available at that time. This requires clear separation of static, observed-past, and known-future covariates [4], [9]. In multimodal forecasting, this rule is especially important because earnings-call transcripts are released after quarter end and can easily leak future information if aligned incorrectly [5]. These controls are essential for making the reported results operationally credible rather than artificially optimistic.

2.7 Chapter Summary

This chapter established the theoretical and methodological foundations for the remainder of the thesis. It first formulated quarterly corporate revenue forecasting as a supervised panel time-series problem and defined the corresponding single-horizon, multi-horizon, and multimodal forecasting objectives. It then introduced the structured feature-engineering principles used throughout the thesis, including log transformations, lagged variables, year-over-year growth measures, deterministic calendar features, and a point-in-time covariate taxonomy.

The chapter also reviewed the benchmark model families used for comparison. Classical models such as ARIMA and SARIMA provide transparent univariate references, while recurrent neural networks such as LSTM offer a standard deep learning baseline for sequential data. Against these baselines, the chapter presented TFT as the central forecasting architecture of the thesis, emphasizing its treatment of mixed covariates, its multi-horizon design, and its built-in interpretability mechanisms.

Finally, the chapter introduced the NLP foundations required for the multimodal extension. It explained why generic dictionary-based sentiment methods are limited in finance, reviewed Transformer-based contextual language modeling, and described how FinBERT and Llama-3 can be used to convert earnings-call text into structured narrative features. Particular attention was given to temporal alignment and leakage control, since transcript-derived features are only useful if they are integrated under strict point-in-time rules.

These foundations support the empirical chapters that follow. Chapter 3 situates the thesis within the broader literature on corporate forecasting and financial NLP. Chapter 4 builds the pure quantitative TFT framework and extends it to multi-horizon forecasting in order to diagnose horizon degradation. Chapter 5 then introduces the multimodal extension based on earnings-call narratives to address the long-horizon limitations identified in the structured baseline.

Chapter 3

Related Work

Corporate revenue forecasting sits at the intersection of accounting, time-series analysis, machine learning, and natural language processing. At a basic level, the task is to infer future firm performance from historical observations. In practice, however, the problem is more complex. Revenue is affected not only by past fundamentals, but also by seasonality, macroeconomic conditions, firm-specific strategy shifts, and qualitative managerial guidance. As a result, the literature has evolved from simple statistical baselines to more flexible machine learning and multimodal forecasting systems [1], [13].

This chapter reviews the literature in a sequence that mirrors the methodological logic of this thesis. It begins with classical econometric approaches, which remain important benchmarks because they are transparent, stable, and easy to validate. It then reviews standard machine learning and deep sequence models, including tree ensembles and recurrent neural networks, which improve nonlinear modeling capacity but still face limitations in multi-horizon forecasting and interpretability. The chapter next turns to the TFT, which is especially relevant because it is explicitly designed for multi-horizon forecasting with mixed covariates and interpretable feature selection [4]. Finally, the review moves to financial NLP and multimodal forecasting, tracing the progression from dictionary-based sentiment to FinBERT and more recent large language model (LLM) pipelines for earnings-call analysis.

The literature suggests two broad conclusions. First, purely quantitative forecasting models can provide strong short-horizon baselines, but their performance often deteriorates as the forecasting horizon extends, especially when the information set is limited to backward-looking fundamentals. Second, narrative information from corporate disclosures can contain incremental predictive value, but its usefulness depends critically on temporal alignment, feature construction, and leakage-aware evaluation.

3.1 Econometric and Classical Approaches

The earliest and most established approaches to corporate forecasting come from econometrics and classical time-series analysis. These methods remain important because they provide strong transparent baselines and make relatively few modeling assumptions beyond stationarity, linear dependence, and the structure imposed by the chosen model family.

3.1.1 Classical Time-Series Models

ARIMA and its seasonal counterpart SARIMA remain standard benchmarks in business and financial forecasting [1], [13]. Their appeal lies in their interpretability, well-understood estimation procedures, and reproducible forecasting workflows. In relatively stable environments, these models can perform surprisingly well, especially when the target series exhibits persistent autoregressive and seasonal structure.

At the same time, ARIMA-type models also have well-known limitations. They are fundamentally linear models of the conditional mean and typically rely on differencing and stable temporal relationships. In practice, firm revenue can experience structural breaks, nonlinear growth phases, demand shocks, and strategic inflection points that are difficult to capture within a purely linear framework [13], [27]. As the underlying data-generating process becomes more heterogeneous or regime-dependent, the performance of purely classical models may deteriorate.

Even so, classical baselines remain highly relevant. Using IBES and Compustat quarterly EPS data, Pagach and Warr show that ARIMA forecasts can match or outperform analyst consensus in approximately 40% of firm-quarters [6]. This result is important because it shows that transparent statistical models can still be competitive in real financial forecasting settings. For this reason, ARIMA and SARIMA remain useful baseline comparators for any more complex modeling framework.

3.1.2 Structural Econometric Models

Beyond univariate forecasting, structural econometric approaches such as vector autoregression (VAR) attempt to model the joint dynamics of multiple related variables [2]. VAR is particularly useful when the target variable is strongly linked to macroeconomic indicators or other system-wide drivers, and when dynamic interactions among variables are themselves of analytical interest.

However, quarterly firm-level revenue forecasting presents several difficulties for VAR-style models. First, firm performance is often shaped by idiosyncratic events such as product cycles, competitive positioning, and management decisions, which may not be captured well by a low-dimensional macro system. Second, firm-level financial panels are usually short in time but broad in cross-section, which makes high-parameter multivariate systems difficult to estimate reliably. Related evidence from asset pricing also suggests that idiosyncratic variation can be large relative to market-wide variation [28].

For these reasons, the literature increasingly shifts away from purely structural econometric systems toward models that can incorporate firm-specific covariates, handle cross-sectional heterogeneity, and scale better in broad-panel settings. This shift motivates the move from classical econometrics toward machine learning and global forecasting architectures.

3.2 Standard Machine Learning for Revenue Prediction

Machine learning models extend the forecasting toolkit by allowing more flexible nonlinear mappings from structured inputs to future outcomes. In fundamentals-based prediction, this flexibility is especially attractive because firms differ widely in size, sector, cyclicality, and growth dynamics. However, not all

machine learning models are equally well suited to time-respecting multi-horizon forecasting.

3.2.1 Tree-Based Ensembles

Tree-based models such as Random Forests [29] and gradient boosting systems such as XGBoost [30] are widely used for structured tabular prediction. They can model nonlinear relationships, interactions among variables, and threshold effects without requiring heavy feature scaling.

In financial statement research, Amel-Zadeh et al. [31] show that machine learning models can extract useful signals from quarterly Compustat fundamentals and forecast market reactions around earnings announcements. This line of work supports the broader idea that flexible tabular models can learn economically meaningful structure from high-dimensional accounting inputs.

Yet for corporate revenue forecasting as a time-series task, tree ensembles have an important limitation: they do not model temporal dependence natively. In practice, they rely on manually engineered lags, growth rates, and rolling-window summaries, and they typically require either separate models or carefully designed features for each forecasting horizon. As a result, they are often useful as strong tabular benchmarks, but less natural as unified multi-horizon forecasting architectures.

This limitation is consistent with empirical evidence. Thieren [32], for example, uses histogram-based gradient boosting trees to predict one-year net income growth from trailing fundamentals and finds that the fundamentals-only setup does not outperform analyst consensus; combining fundamentals with analyst expectations yields only modest improvement. These results suggest that model flexibility alone is not sufficient if the temporal structure and information set are not handled appropriately.

3.2.2 Support Vector Machines

Support Vector Machines (SVMs) offer another path for nonlinear prediction through the use of kernel functions [33]. In finance, Kim [34] reports competitive performance of SVMs in stock index forecasting relative to alternative methods, illustrating that margin-based methods can capture nonlinear decision boundaries in financial data.

However, SVMs are less commonly used in modern large-scale forecasting systems. Their performance depends heavily on kernel specification and hyperparameter tuning, and training can become computationally expensive as the dataset size grows [33]. In firm-level broad-panel forecasting, these practical limitations make SVMs less attractive than tree ensembles or more recent deep forecasting models.

Overall, the machine learning literature shows that structured fundamentals do contain useful predictive information, but it also highlights an unresolved issue: flexible tabular models do not automatically provide a clean solution for sequential, multi-horizon, leakage-aware forecasting.

3.3 Deep Sequence Models for Time-Series

Deep learning models extend machine learning further by learning hierarchical nonlinear representations directly from sequential data. In financial prediction, they are especially attractive when the relationship

between past observations and future outcomes is complex, nonlinear, and difficult to encode manually. The most widely used deep sequence families include recurrent networks and convolution-based sequence models.

3.3.1 LSTM Networks

LSTM networks were introduced to address the vanishing-gradient problem in recurrent neural networks by using gated memory updates [3]. In financial applications, LSTMs have shown promising performance in several sequential prediction settings. Fischer and Krauss, for example, use daily S&P 500 constituent data and report that LSTM models outperform a range of memory-free baselines on directional stock-movement prediction [35]. Other studies combine LSTM with wavelet decomposition, denoising, or hybrid feature-learning components and report gains on financial time-series tasks [36].

Despite these successes, LSTMs also face important limitations. Their performance can be highly sensitive to architecture choices, initialization, and training procedures, and comparative studies show that different LSTM variants can behave quite differently across tasks [37]. More importantly, standard recurrent baselines are not especially natural for direct multi-horizon forecasting. They often require recursive prediction or sequence-to-sequence wrappers, which can accumulate error as the forecast horizon grows.

Interpretability is another concern. In multivariate forecasting settings, it is often difficult to explain which variables or which historical periods drove a given prediction, and post hoc explanation tools for time series can be unstable [38]. This is a major drawback in financial applications where model governance, auditability, and horizon-specific diagnostics matter.

3.3.2 Emerging Deep Learning Architectures

Beyond recurrence, convolution-based deep architectures have also been applied to financial forecasting. Convolutional Neural Networks (CNNs) can learn local temporal or cross-feature patterns within fixed windows and have shown competitive performance in some high-frequency market tasks. Hoseinzadeh and Haratizadeh, for instance, propose CNN-based stock-market prediction methods and report improvements over several baselines for next-day forecasting across major indices [39].

The main limitation of standard CNNs is that they primarily capture local dependencies. Without deeper stacks, dilation, or hierarchical design, they may fail to represent longer-range temporal structure. Recent work addresses this with dilated and hierarchical convolutional architectures. Reisenhofer et al. propose HARNet for realized volatility forecasting and use dilated convolutions to enlarge the effective receptive field while maintaining computational efficiency [40].

Taken together, the deep-sequence literature shows that richer nonlinear temporal modeling is possible, but it also reveals a persistent gap between expressive sequence learning and the practical needs of low-frequency, multi-horizon, and interpretable fundamentals forecasting.

3.4 Transformer-Based Multi-Horizon Forecasting

Transformer-based forecasting models have attracted increasing attention because they combine nonlinear sequence modeling with attention-based mechanisms for learning long-range dependencies [21]. In finance, this is particularly valuable when the forecasting task must accommodate heterogeneous covariates, nonlinear interactions, and changing temporal relevance across horizons.

Among these models, the TFT is especially relevant to this thesis. TFT was proposed specifically for multi-horizon forecasting with mixed covariates [4]. Its architecture separates static attributes, observed-past variables, and known-future inputs; applies variable-selection networks to reduce noise and highlight relevant features; and uses interpretable attention to provide horizon-specific temporal attribution [4]. In addition, TFT supports probabilistic forecasting through quantile outputs, making it appealing for uncertainty-aware decision environments.

Recent research has applied TFT to a growing range of financial and macroeconomic problems. Frank (2023) evaluates TFT for realized volatility forecasting during turbulent market periods and reports strong performance relative to conventional machine learning baselines [41]. Petrosino et al. combine GARCH with TFT for ETF volatility prediction, using econometric structure to complement nonlinear feature learning [42]. In exchange-rate forecasting, Zhang proposes TFT-ICENet and reports improvements in multi-step foreign-exchange prediction [43]. Beyond market prices, Laborda et al. apply TFT to multi-country GDP forecasting and emphasize both multi-horizon accuracy and interpretability [44]. Related work also adapts TFT to highly volatile crypto markets [45].

Although this literature is expanding rapidly, most finance-oriented TFT applications focus on prices, volatility, or macroeconomic aggregates. Comparatively less attention has been given to firm-level quarterly fundamentals, especially revenue forecasting across a broad corporate panel. This is an important gap because quarterly revenue combines low-frequency structure, cross-sectional heterogeneity, and strong practical relevance to valuation and planning. It also creates a natural setting in which horizon degradation can be studied explicitly, since the predictive value of historical fundamentals may weaken as the forecast window extends.

Taken together, the studies reviewed above show that purely quantitative forecasting in finance has evolved from classical linear time-series models to more flexible machine learning and deep learning architectures. Although these approaches differ substantially in modeling capacity, they share a common goal: to extract predictive structure from historical numerical signals alone. Table 3.1 summarizes representative purely quantitative forecasting studies in finance and highlights their datasets, methodological choices, and main empirical findings.

As shown in Table 3.1, purely quantitative models can provide strong baselines, especially in short-horizon or well-structured forecasting settings. However, the literature also suggests that performance often depends on regime stability, feature engineering, and the ability to handle longer-range temporal dependence. These limitations motivate the next strand of research reviewed in this chapter: financial NLP, which seeks to extract additional predictive signals from corporate language rather than relying on historical numerical inputs alone.

While the previous review focused on representative studies, it is also helpful to compare the main

Table 3.1: Comparative Summary of Purely Quantitative Forecasting Studies in Finance

Author–year	Methodology	Dataset	Key finding (very brief)
Pagach & Warr (2020) [6]	Analysts vs. ARIMA	IBES/Compustat, quarterly	ARIMA matches or exceeds analyst consensus in ∼40% of EPS firm-quarters.
Dong (2024) [46]	Deep learning vs. analyst price targets	Compustat, CRSP, IBES (2010–2023)	DL rivals 12-month analyst targets, especially for high-volatility firms.
Thieren (2023) [32]	HGBR net income growth vs. consensus	Compustat & IBES, U.S. firms	Consensus remains stronger overall; combining analyst and ML forecasts yields only marginal gains.
Fischer & Krauss (2018) [35]	LSTM daily return vs. RF/DNN	S&P 500, 1992–2015	LSTM outperforms RF and DNN baselines in daily return prediction.
Jencová et al. (2024) [47]	ARIMA / time-series models	Monthly retail/wholesale indicators	ARIMA performs well in short-horizon settings; seasonal variants help periodic sectors.
Joshi et al. (2023) [48]	Hybrid LSTM–CNN	BSE/NSE daily prices (India)	Hybrid architecture improves prediction accuracy by combining temporal and local pattern learning.
Amel-Zadeh (2022) [31]	Random Forest on financial ratios	Compustat North America (1990–2017)	RF identifies risk-relevant accounting ratios and supports financial statement screening.
Jegadeesh & Livnat (2006) [49]	Revenue/earnings surprises	U.S. firm-quarters (1987–2003)	Revenue surprises move prices and show more persistent return effects than earnings surprises.
Peik et al. (2024) [50]	Subseries + TFT + Markov/LSTM	Binance 1-min crypto, 18 months	Improves short-horizon accuracy and profitability; gains depend on selector accuracy.
Petrosino et al. (2025) [42]	Hybrid GARCH–TFT (volatility)	U.S. ETF volatility (multi-years)	Outperforms ARIMA, SVR, and DL baselines for ETF volatility forecasting.
Zhang (2025) [51]	TFT vs. ARIMA/SVR/LSTM/GRU	FX USD rates (1979–2024)	TFT is competitive or best, especially in volatile regimes with mixed covariates.
Peik et al. (2025) [45]	Adaptive TFT	ETH (Binance, 10-min)	Improves accuracy and trading performance relative to TFT baselines.
Ho & Hung (2024) [52]	CEEMD-based TFT	S&P 500 (2019–2024)	Multi-component TFT substantially improves forecasting performance.
Laborda et al. (2023) [44]	TFT (multi-horizon GDP)	OECD macro (multi-nation)	TFT generalizes well across countries and achieves strong multi-horizon accuracy.
Hu (2021) [53]	Temporal Fusion Transformer	S&P 500 (Yahoo, 2010–2021)	TFT outperforms classical baselines on broad-panel daily stock prediction.

quantitative model families directly from the perspective of quarterly revenue forecasting. In particular, the relevant distinctions concern not only predictive flexibility, but also horizon design, interpretability, and sensitivity to low-frequency financial data. Table 3.2 therefore compares the principal quantitative models used or discussed in the literature for quarterly revenue forecasting.

Table 3.2 shows that no single quantitative model dominates along every dimension. Classical models remain attractive for transparency and simplicity, whereas modern deep forecasting models such as TFT are better suited to mixed covariates and multi-horizon prediction. This comparison clarifies why TFT is adopted as the structured forecasting backbone in this thesis, and it also creates a natural transition to the next question: whether text-derived narrative signals can further improve forecasting performance beyond what structured numerical models can achieve on their own.

3.5 The Evolution of Financial NLP

Purely structured financial models are limited by the fact that many forward-looking signals appear first in language rather than in reported numbers. Earnings calls, annual reports, and other corporate disclosures often contain guidance, risk discussion, managerial framing, and forward-looking expectations that shape market beliefs before those effects fully appear in subsequent accounting statements [5]. This motivates the

Table 3.2: Comparison of Quantitative Models for Quarterly Revenue Forecasting

Model	Application scope	Modeling framework	Performance	Advantages	Limitations
ARIMA [54]	Univariate revenue; short horizon	Linear AR + I + MA	Strong baseline for mature, low-volatility firms	Simple, interpretable; fast; low data needs	Assumes stationarity/linearity; weak under regime shifts
SARIMA [54]	Univariate revenue with strong seasonality	ARIMA + seasonal AR/MA/D	Beats ARIMA when Q1–Q4 seasonality is pronounced	Captures quarterly cycles clearly; transparent diagnostics	Same linear limits; manual seasonal order search
SVR [33]	Next-quarter regression with engineered lags	Kernel regression / margin maximization	Competitive with careful features and scaling	Works with small/medium samples; robust to outliers	Sensitive to feature design, kernel choice, and tuning
RF [29]	Tabular regression with rich covariates	Bagged trees	Strong baseline when exogenous features matter	Handles nonlinearities/interactions; resilient to noise; modest tuning	Weak long-range extrapolation; jagged forecast paths
GBM [30]	Same as RF; strong with many engineered features	Gradient-boosted trees	Often top classical performer with rich features	High accuracy; handles heterogeneity/missingness; efficient training	High leakage risk without strict lagging/splits; can overfit
LSTM [3]	Multivariate sequence forecasting	Gated RNN	Strong on nonlinear temporal dependencies	Learns sequential structure directly; flexible setups	Needs more data and tuning; slower; limited interpretability; error drift on long horizons
TFT [4]	Multi-horizon quarterly revenue with static + time-varying covariates	LSTM encoder + attention, gating, quantile loss	State-of-the-art for multi-horizon business forecasting	Uses mixed covariates well; interpretable variable selection and attention	More hyperparameters; overfit risk; requires disciplined data pipeline

growing role of financial NLP in forecasting and decision systems.

3.5.1 Dictionary-Based Sentiment Methods

Early financial text methods often relied on dictionary-based sentiment measures. Common examples include VADER [55] and the Loughran–McDonald finance dictionary [23]. These approaches are attractive because they are simple, computationally cheap, and easy to explain.

However, dictionary methods also have substantial limitations. They mostly treat words as independent units and therefore handle context poorly. As a result, they often miss negation, modality, and hedging, such as the difference between "may weaken" and "will weaken." They can also misinterpret domain-specific financial language. Loughran and McDonald show that general-purpose dictionaries frequently misclassify common financial terms, which is precisely why finance-specific lexicons are needed [23]. For forecasting tasks that require stable and nuanced narrative signals, dictionary methods are often too coarse.

3.5.2 Transformer Models and FinBERT

The transition from dictionary methods to contextual language models marked a major change in financial NLP. Transformer architectures model words in context rather than in isolation, allowing them to capture

sentence-level structure, negation, and long-range semantic dependencies [21], [24]. BERT, in particular, became a foundational encoder architecture for contextual representation learning [24].

In the financial domain, domain adaptation is crucial because general-purpose language models may misread specialized terminology and disclosure conventions. FinBERT addresses this issue by adapting BERT to financial text and is now widely used for finance-domain sentiment classification [10]. In most implementations, FinBERT outputs probabilities over the classes {positive, neutral, negative}. These outputs can then be converted into continuous sentiment features, such as net sentiment, and integrated into downstream forecasting systems.

This makes FinBERT especially useful as an intermediate step between raw narrative text and structured time-series models. However, its output space is still relatively constrained: it is designed primarily for sentiment classification rather than for richer prompt-defined feature extraction.

3.5.3 Generative Large Language Models in Finance

More recent research moves beyond classification into generative and instruction-following language models. Large language models can be prompted not only to summarize or label financial text, but also to produce structured outputs along multiple semantic dimensions [25]. This is particularly valuable for earnings calls, where a single passage may simultaneously contain reported results, future guidance, uncertainty, and strategic framing.

Cook et al. argue that local deployment of open LLMs can improve privacy, reproducibility, and operational control relative to closed commercial APIs [14]. In their earnings-call study, they prompt LLMs to extract continuous scores for sentiment, temporality, and vagueness, demonstrating that transcript language can be translated into structured numerical features suitable for downstream prediction [14]. Related work also explores target-aware stance extraction [56], few-shot prompting and fine-tuning for earnings-call generation [57], and counterfactual narrative manipulation to study analyst reactions [58].

At the same time, LLM-based financial inference introduces practical constraints. Model size, latency, hardware cost, and reproducibility all matter, especially in local deployment settings. Ni et al. show that efficient adaptation using QLoRA and 4-bit Llama-style models can make this kind of pipeline much more practical [12]. These developments suggest that generative LLMs are increasingly viable as feature extractors in structured financial forecasting pipelines, provided that compute, timing, and evaluation are handled carefully.

The financial NLP literature has progressed from generic language representation models to increasingly specialized domain-adapted systems. In forecasting-related settings, the key issue is not only whether a model performs well on sentiment classification, but also whether its outputs can be translated into usable numerical signals for downstream prediction tasks. Table 3.3 summarizes representative financial NLP studies relevant to forecasting and highlights how the literature has evolved from general-purpose Transformer foundations to finance-specific and multilingual sentiment models.

As Table 3.3 indicates, financial NLP has moved steadily toward stronger domain adaptation and more context-aware modeling. These developments make text-derived features increasingly useful for forecasting, but they do not by themselves solve the downstream integration problem. The next step, therefore, is to

Table 3.3: Comparative Summary of Financial NLP Studies Relevant to Forecasting

Author–year	Methodology	Dataset	Key finding (very brief)
Devlin et al. (2019) [24]	BERT pre-training	BooksCorpus + English Wikipedia	Introduces strong bidirectional contextual representations and sets the foundation for Transformer NLP.
Huang et al. (2023) [59]	Finance-domain BERT (FinBERT)	10-K/10-Q filings, analyst reports, earnings calls	FinBERT outperforms lexicon-based and classic ML/DL baselines on financial sentiment tasks.
Chen et al. (2023) [60]	FinBERT	FOMC Minutes sentences	Improves sentiment classification accuracy relative to the original FinBERT benchmark.
Bansal et al. (2025) [61]	FinBERT, DistilBERT, VADER	SEntFiN, FiQA, PhraseBank, headlines, microblogs	Transformer-based models outperform lexicon baselines across financial sentiment datasets.
Nasiopoulos et al. (2025) [62]	Fine-tuned GPT-4o vs. BERT/FinBERT	FiQA + Financial PhraseBank	Fine-tuned GPT-style models achieve higher sentiment classification accuracy.
Huang et al. (2024) [63]	FinBERT-based sentiment framework	Chinese financial news headlines	Domain-specific pretraining improves classification performance in Chinese finance text.
Ayush (2024) [64]	Hindi FinBERT	Hindi financial corpora	Additional finance-domain pretraining improves sentiment performance in low-resource settings.

examine multimodal and hybrid systems that combine narrative signals with structured financial variables inside an explicit forecasting framework.

Pure LLM-based financial forecasting is possible and is becoming an active research direction. However, in this study, a hybrid framework is used instead, because it gives more stable numerical outputs, clearer time-series modeling, and better control of evaluation. For this reason, the thesis combines LLM-based text features with a structured forecasting model, rather than relying on the LLM alone.

3.6 Multimodal Integration in Financial Forecasting

A growing literature combines narrative text with structured financial variables in order to improve forecasting and risk analysis. In many early hybrid systems, text-derived sentiment is extracted first and then used as an additional covariate inside a quantitative model. This design reflects a common intuition: textual narratives contain incremental information not visible in fundamentals alone, but the final prediction task still benefits from a structured forecasting backbone.

A representative example is the use of FinBERT-style sentiment with downstream sequence or tabular models. Halder proposes a FinBERT–LSTM framework that combines news sentiment with NASDAQ-100 data and reports improvements over price-only baselines [65]. Hossain et al. extend a similar design to cryptocurrency markets using FinBERT and BiLSTM [66]. Ruan and Jiang combine FinBERT-derived news sentiment with technical and statistical indicators inside an XGBoost model and add SHAP-based attribution to improve interpretability [67]. Outside forecasting, Taheripour et al. incorporate FinBERT sentiment from quarterly reports into a portfolio optimization framework to improve risk control [68].

Recent studies have begun to explore the use of LLMs in financial forecasting. For example, Jin et al. investigates explainable financial time-series forecasting by combining temporal data with LLM-based reasoning in a stock forecasting setting [69]. More recent work such as Anonymous extends this direction by proposing a unified multimodal large language model for both micro-level stock prediction and macro-level systemic risk assessment [70]. These studies show that LLMs can play a useful role in multimodal financial forecasting, but the framework used in this thesis is different in several important ways. First, this thesis

Table 3.4: Comparison of Financial NLP and Multimodal Methods Relevant to Revenue Forecasting

Model / Method	Primary task	Modeling framework	Typical performance	Advantages	Limitations
FinBERT [59]	Financial sentiment analysis	BERT adapted via finance-domain pretraining	88.2% accuracy	Strong finance vocabulary and contextual understanding	More computationally expensive than lexicon methods
FinBERT-FOMC [60]	Central-bank communications sentiment	FinBERT fine-tuning	+5% overall accuracy; +17.4% on complex sentences	More robust on complex financial language	Requires careful preprocessing and domain alignment
FinBERT-Sentiment (Chinese) [63]	Chinese financial news sentiment	FinBERT-based classifier	94.52% accuracy; beats BERT/CNN/SVM baselines	Strong domain and language adaptation	Performance depends on dataset/topic coverage
DistilBERT [61]	Financial sentiment analysis	Compressed Transformer encoder + classifier	Can achieve high accuracy	Faster and cheaper than larger Transformers	Less domain-specific; may miss subtle financial semantics
GPT-4o [62]	Financial sentiment classification	Fine-tuned LLM on finance sentiment labels	Accuracy 0.8779 after fine-tuning	Strong few-shot / zero-shot baseline	Higher cost and latency; harder explainability and governance
FinBERT + TFT [71]	Market forecasting with text + prices	FinBERT sentiment fused into TFT	Lower errors reported than several neural baselines	Multimodal fusion with horizon-aware forecasting backbone	Engineering complexity; strict anti-leakage alignment required
FinBERT + CCVaR [68]	Portfolio optimization with textual signals	FinBERT sentiment-adjusted return/risk modeling	Empirical DJIA study	Integrates qualitative reporting into risk-aware optimization	More moving parts; assumptions require stress testing
Temporal Data Meets LLM [69]	Financial time-series forecasting	LLM + temporal data integration	Reported improvements over classical baselines	Explainable multimodal reasoning for financial prediction	LLM used as primary predictor; less focus on structured forecasting backbones
Uni-FinLLM [70]	Multimodal financial prediction and risk assessment	Unified LLM-based multimodal architecture	Strong performance across tasks	Integrates multiple modalities in a unified framework	High model complexity; less focus on firm-level structured forecasting tasks

focuses on quarterly corporate revenue, not stock returns, price direction, or systemic risk. Second, Llama-3 is not used as the final predictor. It is used to turn forward-looking earnings-call text into numerical sentiment features, which are then fed into TFT. Third, the framework puts strong emphasis on firm-quarter alignment, correct transcript timing, and chronological evaluation. This positioning bridges the gap between recent LLM-based financial prediction models and structured time-series forecasting approaches, and motivates the hybrid design adopted in this thesis.

The studies reviewed above suggest that text can contribute incremental predictive value, but the practical usefulness of financial NLP depends heavily on how narrative signals are encoded and how they are integrated with downstream forecasting models. For the specific problem of quarterly revenue forecasting, the most relevant methods are those that either extract structured financial text features or combine such features with a forecasting backbone. Table 3.4 compares these financial NLP and multimodal methods from the perspective of their relevance to revenue forecasting.

Table 3.4 highlights two important patterns. First, finance-domain language models such as FinBERT provide strong and interpretable sentiment representations. Second, hybrid systems become more valuable

when they are paired with a forecasting backbone that is capable of handling horizon structure and mixed covariates. These observations directly support the methodological direction of this thesis, which combines finance-specific text extraction with a TFT-based multi-horizon forecasting architecture.

These studies support the broader view that text can provide incremental predictive value. However, many hybrid systems still face two limitations. First, the forecasting backbone is often not designed explicitly for multi-horizon prediction, so longer-horizon forecasting may rely on recursive schemes that accumulate error. Second, interpretability is often either limited or only loosely connected to how text signals affect the forecast across time.

This motivates the use of forecasting models designed for mixed covariates and horizon-aware prediction. TFT is one such candidate. Because it separates static attributes, observed-past variables, and known-future inputs, it offers a natural structure for integrating structured and text-derived covariates in a single framework [4]. Jin and Lin provide an example of this direction by combining FinBERT sentiment with TFT in a systemic-risk early-warning setting and reporting performance gains relative to several baselines [72].

At the same time, recent multimodal benchmarks caution against overly optimistic assumptions. FinCall-Surprise provides a large-scale benchmark with synchronized transcripts, audio, and slides for earnings-related prediction and shows that model performance remains sensitive to dataset imbalance and modality utilization [5]. Related work explores discourse-aware structures [73], acoustic-text integration [74], and speaker- and emotion-aware annotation frameworks [75]. Together, these studies show that multimodal integration is promising, but not trivial: the value of additional modalities depends on correct timing, meaningful feature design, and realistic evaluation.

Beyond text-only sentiment models, a broader multimodal literature now combines financial narratives with other information channels, including prices, macro variables, images, and alternative data sources. This literature is especially relevant because it shows both the promise and the practical difficulty of integrating heterogeneous signals into a unified predictive system. Table 3.5 summarizes representative multimodal and hybrid systems in finance and highlights how different modalities are fused in downstream applications. As

Table 3.5: Comparative Summary of Multimodal and Hybrid Systems in Finance

Author–year	**Methodology**	**Dataset**	**Key finding (very brief)**
Friday (2025) [76]	ViT (candlesticks) + TFT	Global equity indices	Spatial and temporal fusion improves movement classification performance.
Shahsafi & Naderkhani (2025) [77]	TFT + GAF/ResNet18 (image inputs)	AAPL daily (2015–2025)	Image-enhanced TFT outperforms LSTM/GRU and raw-data TFT on forecast errors.
Lodoen & Myklebust (2024) [78]	TFT + macro/sentiment	Bitcoin + Twitter + Trends	Hybrid macro/sentiment features materially improve forecasting performance.
Thota (2025) [71]	Hybrid FinBERT+TFT	Daily stock data + Twitter/Reddit/news	FinBERT+TFT achieves lower forecast errors than several neural baselines.
Taheripour et al. (2025) [68]	Portfolio optimization using FinBERT	DJIA stock universe	Quarterly-report sentiment improves risk-aware portfolio optimization under uncertainty.

Table 3.5 shows, multimodal integration is promising but far from automatic. Performance gains depend on the quality of feature extraction, the suitability of the downstream forecasting backbone, and, above all, the correctness of temporal alignment and evaluation design. These observations lead directly to the final section of this chapter, which identifies the remaining research gaps and positions the present thesis within this evolving literature.

3.7 Identified Research Gaps and Thesis Positioning

The reviewed literature shows substantial progress in structured forecasting, deep sequence modeling, and financial NLP. However, their intersection remains incomplete, especially for firm-level quarterly revenue forecasting. Existing work leaves several gaps that directly motivate this thesis.

Gap 1: High-Frequency Bias and Limited Focus on Firm Fundamentals

Much of the multimodal finance literature focuses on high-frequency targets such as daily price movement, volatility, or short-term market direction [67], [79]. By contrast, there is far less work on forecasting quarterly firm fundamentals, especially revenue, even though revenue is a direct measure of business performance and is central to valuation and planning.

Gap 2: Limited Evidence on Long-Horizon Revenue Forecasting

Multi-step forecasting becomes substantially harder as the horizon expands, and error accumulation is a well-known problem in time-series forecasting [13]. Although TFT was explicitly designed for multi-horizon forecasting, its use in firm-level quarterly revenue prediction remains limited relative to applications in prices, volatility, and macroeconomic aggregates [4]. As a result, there is still limited evidence on how well structured models perform when revenue forecasting is extended from one quarter to a full year.

Gap 3: Incomplete Treatment of Timing Rules and Leakage Control for Text

Earnings calls are released after quarter-end and often mix retrospective discussion with forward-looking guidance. This makes them valuable, but also risky, as forecasting inputs. If text-derived features are aligned incorrectly, the resulting model can easily leak future information. Prior work, including earnings-call benchmarks, recognizes this issue [5], but a standardized leakage-aware pipeline for mapping transcript-derived features into firm-quarter forecasting systems remains underdeveloped.

Gap 4: Limited Use of Generative LLM Outputs as Structured Time-Varying Covariates

Generative LLMs can extract richer signals than three-class sentiment, including forward-looking emphasis, vagueness, and narrative tone [14], [25]. However, there is still limited work that converts these outputs into stable, repeatable, time-aligned covariates inside an interpretable multi-horizon forecasting model. Practical constraints, including memory usage and inference cost, make this gap even more relevant [11], [12].

Thesis Positioning

This thesis addresses these gaps by shifting the forecasting target from noisy high-frequency market variables to firm-level quarterly revenue, and by treating earnings-call transcripts as a structured source of forward-looking narrative information. It first establishes a leakage-free TFT-based quantitative baseline on a broad panel of S&P 500 firms, then extends that baseline to four-quarter forecasting in order to diagnose

horizon degradation, and finally introduces multimodal extensions based on FinBERT and Llama-3-derived narrative features. Methodologically, the thesis combines three elements that are rarely integrated in a single framework: (i) a broad-panel, multi-horizon forecasting design, (ii) strict point-in-time alignment and leakage control for transcript-derived variables, and (iii) the use of both domain-specific sentiment classifiers and generative LLM-based narrative features as structured covariates inside an interpretable TFT backbone. In this way, the thesis positions itself at the intersection of corporate fundamentals forecasting, financial NLP, and practical multimodal machine learning.

Chapter 4

Quantative TFT Forecasting for S&P 500 Firms

4.1 Introduction and Chapter Roadmap

This chapter develops the purely quantitative forecasting framework of the thesis and evaluates its capabilities under both short-horizon and long-horizon settings. Its purpose is twofold. First, it establishes a leakage-free TFT baseline for next-quarter corporate revenue forecasting across a broad panel of 155 continuously listed S&P 500 firms. This stage determines how accurately structured financial fundamentals alone can support firm-level revenue prediction under a strict chronological evaluation protocol. Second, the chapter extends the same framework from one-quarter-ahead forecasting to four-quarter-ahead forecasting in order to examine whether that predictive strength can be maintained over the longer horizon that is often relevant in practical financial analysis.

As illustrated in Figure 4.1, the chapter follows a two-stage quantitative design. It begins with the construction of a structured forecasting framework based on S&P 500 panel data and engineered financial covariates. Stage I then establishes and validates the leakage-free next-quarter TFT baseline through benchmark comparison, ablation testing, and interpretability analysis. Building on this baseline, Stage II extends the same structured setting to four-quarter forecasting in order to isolate and diagnose horizon degradation under controlled conditions, both at the aggregate level and across sectors.

Taken together, these two stages establish both the strength and the boundary of purely fundamentals-driven forecasting. The results show that structured financial inputs can support a strong next-quarter benchmark, but that their predictive value weakens materially as the horizon expands, especially in more dynamic sectors. This empirical boundary provides the motivation for the next chapter, which introduces a multimodal extension based on earnings-call-derived narrative features.

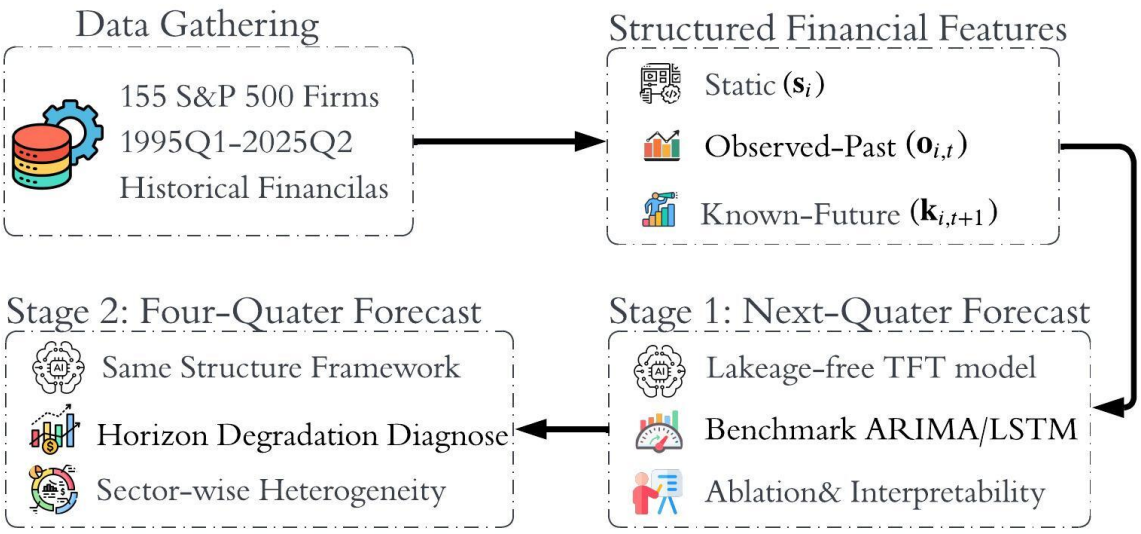


Figure 4.1: Two-stage quantitative design of Chapter 4.

4.2 Data, Target Variable, and Covariate Design

This section defines the structured forecasting setting used throughout the chapter. It first formalizes the target variable and the one-step forecasting objective, then describes the construction of the S&P 500 panel, the preprocessing pipeline, and the covariate taxonomy used by the baseline and comparison models. Because the empirical purpose of this chapter is to isolate the predictive value of structured financial information alone, particular care is taken to enforce point-in-time validity and leakage-free preprocessing.

4.2.1 Forecasting Target and Problem Formulation

The quantitative framework developed in this chapter focuses on firm-level quarterly revenue forecasting using structured financial fundamentals only. Let firm i denote a continuously listed S&P 500 constituent and let t denote a calendar quarter. We formulate next-quarter forecasting as a supervised learning problem under a strict point-in-time information set. Following the TFT framework of Lim et al. [4], the available inputs are partitioned into three groups as shown in Table 2.1.

The baseline objective in Stage I of this chapter is one-quarter-ahead forecasting. Let L denote the encoder length, or look-back window. The model observes the target history $y_{i,t-L+1:t}$ together with aligned structured covariates and learns a mapping as Equation 2.2. In Stage II, this same structured setting is extended to four-quarter forecasting in order to examine horizon degradation under controlled conditions.

4.2.2 The S&P 500 Panel Dataset

The main panel consists of continuously listed S&P 500 constituents over the period

$$T = \{1995\text{Q1}, \ldots, 2025\text{Q2}\},$$

which spans $|T| = 122$ fiscal quarters. Let $m_{i,t} \in \{0, 1\}$ denote an index-membership indicator. The continuously listed cohort is defined as

$$\mathcal{F} = \{\, i : \; m_{i,t} = 1 \;\; \forall t \in T \,\}.$$

This yields $|\mathcal{F}| = 156$ firms and $156 \times 122 = 19{,}032$ firm-quarter observations. For each firm-quarter, raw structured fundamentals are retrieved from standardized U.S. GAAP financial statements through the Financial Modeling Prep (FMP) API [80]. The initial download includes $p_{\text{raw}} = 43$ accounting items drawn primarily from the income statement and balance sheet. This yields 818,376 raw feature values.

$$43 \times 156 \times 122 = 818{,}376$$

Because the Real Estate sector contains only one continuously listed firm in the sample and is therefore not representative, that firm is excluded from the subsequent analysis. The final modeling sample contains 155 firms and $155 \times 122 = 18{,}910$ firm-quarter observations. After cleaning, harmonization, and coverage screening, the final modeling set retains a core group of $p = 23$ structured covariates, corresponding to

$$23 \times 155 \times 122 = 434{,}930$$

covariate values in the final panel. To support reproducibility, API versioning, request parameters, and access timestamps are logged throughout the data collection process [80]. FMP sector labels are harmonized to GICS sectors through a fixed mapping. Table 4.1 reports the distribution of continuously listed firms by sector prior to excluding Real Estate.

4.2.3 Preprocessing Pipeline and Leakage Control

All preprocessing is conducted under a strict information-set assumption: every feature used at forecast origin t must be computable using information available at or before t, except for legitimately known-future calendar variables. This principle is essential for preventing look-ahead bias and for ensuring that the resulting forecasts correspond to a deployable setting rather than an artificially optimistic backtest.

To maintain comparability across models, the chapter uses a single reproducible *per-ticker* preprocessing pipeline. For each firm, the cleaned and engineered features are written to a standardized firm-level file (e.g., `AAPL_feature.csv`), from which model-specific datasets are later constructed. This design avoids a monolithic master panel, makes each transformation auditable at the firm level, and ensures that all models consume features generated under the same causal rules.

For each ticker, the preprocessing pipeline applies the following steps:

Table 4.1: Mapping of FMP sectors to GICS sectors (continuously listed sample, 1995Q1–2025Q2).

fmp_sector	gics_sector	No. Firms
Industrials	Industrials	32
Financial Services	Financials	22
Consumer Defensive	Consumer Staples	19
Healthcare	Health Care	18
Utilities	Utilities	16
Technology	Information Technology	14
Consumer Cyclical	Consumer Discretionary	12
Energy	Energy	9
Basic Materials	Materials	9
Communication Services	Communication Services	4
Real Estate	Real Estate	1

Note: Real Estate is excluded from subsequent analyses because the continuously listed cohort contains only one firm in that sector.

1. **Quarter boundary correction (5-day anchor).** Fiscal period-end dates are shifted back by five days before extracting year-quarter labels in order to reduce end-of-quarter mislabeling, without altering the underlying accounting magnitudes.

2. **Units and scaling.** Monetary items, including revenue, gross profit, SG&A, R&D, operating income, net income, total assets, and total equity, are standardized to USD millions.

3. **Conservative row filtering.** Observations are removed only when essential keys or targets are missing, such as `Year_Quarter` or revenue. Non-critical missing covariates are preserved whenever possible to maintain panel continuity.

4. **Outliers and impossible values.** Continuous covariates are winsorized at the 1st and 99th percentiles, and impossible values such as negative revenue are removed.

5. **Missingness handling.** Structural absence, such as firms that do not report R&D, is encoded through an `is_missing` indicator, with zero-imputation applied only after relevant transformations. Short gaps may be forward-filled when appropriate; longer unresolved gaps are excluded where required by model input constraints.

6. **Target transform and inverse mapping.** The target is modeled as $\log(1 + \text{revenue})$, and predictions are mapped back to the original scale during evaluation using $\widehat{\text{rev}} = \exp(\hat{y}) - 1$.

7. **Lagging and warm-up trim.** After creating lags, growth rates, and rolling statistics, the initial warm-up periods without sufficient history are dropped according to the sequence length required by each model.

8. **Sector harmonization.** FMP sector labels are mapped to GICS and retained as categorical descriptors for downstream modeling.

Chronological train/validation/test splits are defined later in this chapter, but the preprocessing pipeline itself is constructed so that all derived variables respect time order before model fitting begins.

4.2.4 Covariate Taxonomy and Model Inputs

Following the TFT taxonomy, the structured covariates are divided into static features $\mathbf{s}_i$, observed-past features $\mathbf{o}_{i,t}$, and known-future features $\mathbf{k}_{i,t+h}$. This taxonomy is useful not only for TFT itself, but also for clarifying the information sets available to the benchmark models. Table 4.2 summarizes which feature groups are consumed by ARIMA, SARIMA, LSTM, and TFT.

Table 4.2: Features used by each model (ARIMA, SARIMA, LSTM, TFT).

Feature Group (sub-item)	**ARIMA**	**SARIMA**	**LSTM**	**TFT**
Static $\mathbf{s}_i$ (identity/sector)			ticker/CIK (series id); GICS sector (embed/one-hot)	ticker/CIK (group id); GICS sector (embedded)
Observed–past $\mathbf{o}_{i,t}$: Revenue	Univariate revenue $y_{i,\tau}$, $\tau \le t$	Univariate revenue with seasonal operators (s=4)	Lags $y_{i,t-\ell}$ (ℓ=1:16); log-transform; QoQ/YoY; rolling mean/std	Lags (encoder window, 12q); log(1+y); QoQ/YoY; rolling mean/std
Observed–past $\mathbf{o}_{i,t}$: Ratios			R&D intensity (lags k=1:4 / up to 12); SG&A intensity; cost-of-revenue; gross margin	Same as LSTM (R&D lags; SG&A; cost-of-revenue; gross margin)
Observed–past $\mathbf{o}_{i,t}$: Calendar		Seasonality via seasonal differencing/orders (s=4)	Quarter dummies or sinusoidal encodings (Q1–Q4)	Usually treated as known-future; see below
Known–future $\mathbf{k}_{i,t+h}$: Calendar				Quarter-of-year indicators (Q1–Q4); year/quarter counters
Known–future $\mathbf{k}_{i,t+h}$: Scale proxy				`totalAssets_lag1` carried to t+1; in later ablations, alternative carry-forward conventions are tested

Static features $\mathbf{s}_i$. Static attributes represent firm-level information that does not vary over time within the forecasting window. In this chapter, the main static descriptors are firm identity (used as `group_id`) and GICS sector.

Observed-past features $\mathbf{o}_{i,t}$. Observed-past variables are time-varying inputs available only up to the forecast origin t.

1. **Revenue dynamics.** ARIMA and SARIMA consume only the univariate revenue history $y_{i,\tau}$ for $\tau \le t$, with SARIMA modeling annual seasonality through $s = 4$. LSTM and TFT use richer sequential

inputs, including autoregressive revenue lags, the $\log(1+\text{revenue})$ transform, quarter-over-quarter and year-over-year growth features, and rolling summary statistics.

2. **Fundamentals and ratios.** For LSTM and TFT, the observed-past feature set also includes accounting ratios available at time t, such as R&D intensity with distributed lags, SG&A intensity, cost-of-revenue ratio, and gross margin.

3. **Seasonality proxies.** SARIMA handles seasonality through seasonal operators. LSTM may encode quarter-of-year seasonality using dummy variables or sinusoidal encodings. In TFT, deterministic calendar information is assigned to the known-future branch rather than the observed-past branch.

Known-future features $\mathbf{k}_{i,t+h}$. Known-future inputs are used only by TFT in this chapter.

1. **Calendar features.** Quarter-of-year indicators and time-index variables are treated as deterministic known-future covariates.

2. **Scale proxies under a controlled information-set convention.** Variables such as `totalAssets_lag1` and `totalEquity_lag1` are carried forward to Horizon $(t+1)$ under the assumption that the most recent reported scale measures remain the best available firm-size context at the forecast origin. Their treatment is revisited in later ablation and horizon-extension analyses.

Univariate baselines. ARIMA and SARIMA are implemented as strictly univariate models, without ARIMAX or SARIMAX regressors. Orders are selected using validation-set MAPE, with ties broken first by lower MAE and then by model parsimony.

4.3 Models and Experimental Design

Having defined the panel and the covariate structure, we next specify the forecasting models and the common evaluation protocol used throughout this chapter. The TFT serves as the main structured forecasting architecture, while ARIMA, SARIMA, and LSTM provide statistical and recurrent baselines. To ensure comparability, all models are trained and evaluated under the same chronological split and leakage-aware information set, with model-specific hyperparameters selected using the validation block only.

4.3.1 TFT Instantiation for Structured Revenue Forecasting

The main forecasting model in this chapter is the TFT, which is well suited to structured quarterly revenue forecasting because it can jointly process static attributes, observed-past covariates, and known-future inputs within a single multi-step forecasting architecture [4]. In the present setting, TFT is configured as a global panel model operating over the 155-firm S&P 500 cohort.

As illustrated in Figure 4.2, the model combines four principal components:

1. **Static covariate encoding**, which transforms firm-level attributes such as sector identity into learned contextual signals;

2. **Variable selection networks**, which dynamically weight the relevance of covariates at each time step;
3. **An LSTM encoder–decoder**, which captures local temporal structure in the observed history and the known-future path; and
4. **Masked interpretable multi-head attention**, which identifies the historical quarters most relevant to each forecast.

Following [4], the TFT instantiation used in this chapter combines static-context construction, temporal variable selection, sequence processing, and attention-based refinement in a unified architecture. The resulting design is especially useful in the present application because it accommodates mixed covariates while preserving horizon-aware interpretability.

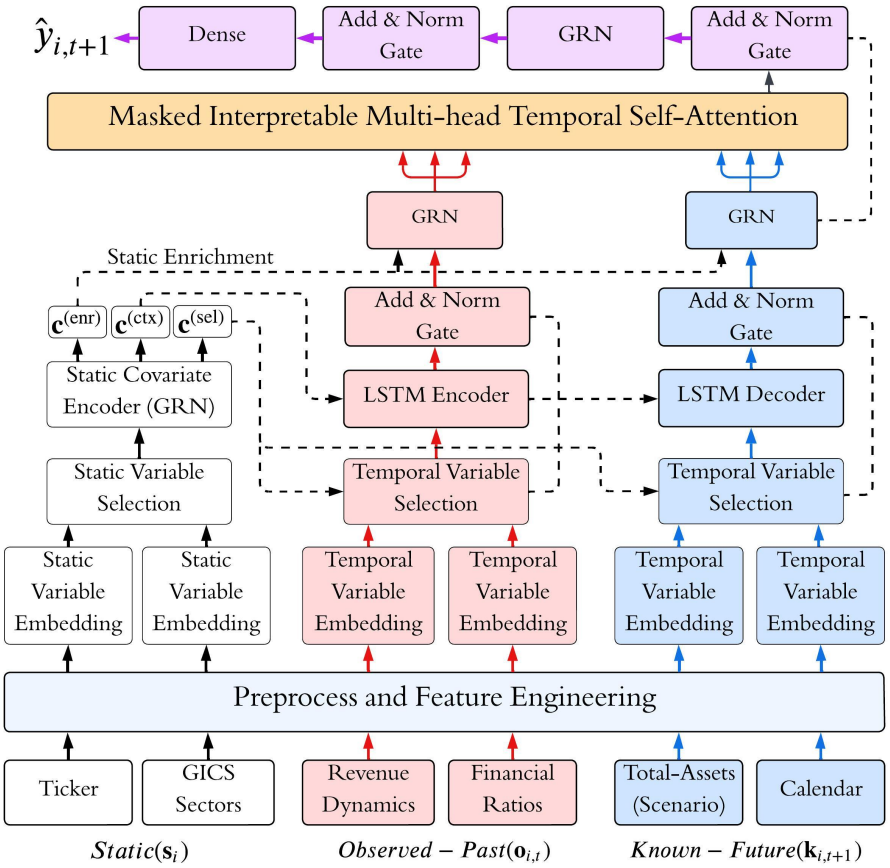


Figure 4.2: Illustration of the TFT pipeline used for firm-level revenue forecasting.

4.3.2 Key Processing Stages in TFT

Figure 4.2 summarizes the end-to-end TFT pipeline. For the purposes of this chapter, the most important processing stages are as follows.

1. Static context construction. Static firm descriptors, such as GICS sector, are first transformed into context vectors through static variable selection and gated residual processing. These context vectors guide downstream variable selection, enrich temporal representations with firm-level context, and initialize the recurrent sequence blocks. In this way, the temporal pipeline does not interpret all firms identically, but conditions its processing on relatively stable corporate characteristics.

2. Temporal variable encoding and selection. At each quarter, the time-varying covariates are encoded and passed through temporal variable selection networks. This allows the model to assign higher weight to the most relevant features at a given time step and to suppress noisy or redundant inputs. In the present setting, this mechanism is especially useful because the structured financial panel contains variables of very different scale and predictive relevance.

3. Encoder–decoder modeling with static enrichment. The selected historical signals are processed through an LSTM encoder, while known-future inputs such as calendar variables enter through the decoder branch. Static enrichment layers condition these temporal representations on firm-level context. Gated residual connections and normalization stabilize training and allow simpler transformations to pass through when additional nonlinear complexity is unnecessary.

4. Interpretable attention and output generation. The decoder states query the encoded history through masked interpretable multi-head attention. The masking rule preserves causality by ensuring that future information cannot influence the forecast. The resulting attention weights provide a direct diagnostic signal regarding which historical quarters are most influential. Finally, the model outputs predictive quantiles, from which the median forecast is used as the main point prediction in the empirical analysis.

5. Training objective and inverse mapping. TFT is trained as a probabilistic forecaster using quantile loss. Throughout this chapter, the target is modeled as $\log(1 + \text{revenue})$, and point forecasts are taken from the median quantile ($q = 0.50$).

4.3.3 Benchmark Models: ARIMA, SARIMA, and LSTM

Before describing the individual baselines, it is useful to distinguish their input constraints. As shown in Figure 4.3, the models differ fundamentally in the information they are permitted to use:

1. **TFT** consumes all three feature families: static attributes ($\mathbf{s}_i$), observed-past covariates ($\mathbf{o}_{i,t}$), and known-future covariates ($\mathbf{k}_{i,t+1}$).
2. **LSTM** uses observed-past multivariate inputs only.
3. **ARIMA and SARIMA** are strictly univariate baselines operating only on the historical target series.

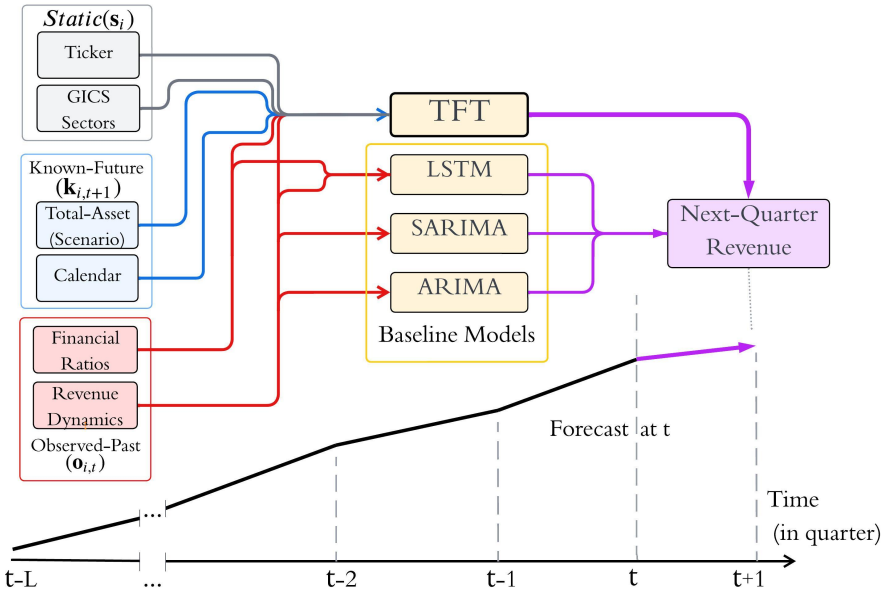


Figure 4.3: Input feature flow for quarterly revenue forecasting models.

ARIMA. ARIMA(p, d, q) serves as the primary classical statistical baseline. For each firm, a grid of non-seasonal order combinations is evaluated, and the specification with the lowest validation-set MAPE is selected. Because ARIMA is strictly univariate here, it does not use any external structured covariates. Estimation and diagnostics are implemented using the `statsmodels` library [81].

SARIMA. SARIMA$(p, d, q) \times (P, D, Q)_s$ extends ARIMA by modeling seasonal dynamics explicitly. Because the data are quarterly, the seasonal period is fixed at $s = 4$. This makes SARIMA particularly relevant for firms whose revenue exhibits stable recurring quarter-of-year patterns. As with ARIMA, model orders are selected by validation performance using a controlled grid search [81].

LSTM. The LSTM network serves as the main recurrent deep learning baseline. Unlike the classical univariate models, the LSTM consumes multivariate observed-past inputs, including revenue lags and selected accounting ratios, and maps them to the next-quarter target. In this chapter, the LSTM baseline uses a single recurrent layer with 64 hidden units and a 10% dropout rate. It is trained using the Adam optimizer and Huber loss, with early stopping based on validation performance. Final predictions are inverse-transformed to the original revenue scale for comparison with the other models [3], [37], [82].

4.3.4 Chronological Data Splitting

All models are evaluated under a strict chronological split in order to prevent look-ahead bias. Rather than using random resampling, the panel is divided into a continuous 70–15–15 sequence based on the forecast target quarter. As illustrated in Figure 4.4, the training block spans the earliest part of the sample, the validation block occupies the subsequent period used for model tuning, and the final test block is fully held

out for out-of-sample evaluation.

Figure 4.4: Chronological train–validation–test split used throughout the forecasting experiments.

This design ensures that all model tuning, early stopping, and order selection are performed without access to the held-out test period. Firms contribute samples to a given split only when sufficient historical observations exist to form the required look-back window and forecast target. Table 4.3 reports the exact calendar ranges, number of quarters, firms, and resulting samples in each block. No artificial temporal gap is introduced between adjacent splits. This choice reflects a realistic forecasting setting, where future quarters naturally follow recent historical observations. Although temporal adjacency may increase similarity between neighboring periods, this does not constitute data leakage. Look-ahead bias is avoided because all model training, validation, and testing are conducted in strict chronological order, without access to future observations from the validation or test periods.

Table 4.3: Chronological data splits shared by all models.

Sample Set	Calendar Range	# Quarters	# Firms	# Samples
Train	1995Q1–2016Q1	85	155	13,175
Validation	2016Q2–2020Q3	18	155	2,790
Test	2020Q4–2025Q2	19	155	2,945

4.3.5 Hyperparameter Selection, Training Configuration and Evaluation Metrics

Model hyperparameters are selected using the validation block only. This ensures that the test set remains untouched until all modeling decisions are finalized.

Implementation environment. Deep learning experiments are implemented using `PyTorch` and `PyTorch Lightning`, with `pytorch-forecasting` supporting the TFT training pipeline. ARIMA and SARIMA are estimated using `statsmodels`, while preprocessing and evaluation utilities rely on `scikit-learn`, `pandas`, and `numpy`. Hyperparameter search is conducted with `Optuna`, and experiment logging is managed through `Weights&Biases` with CSV-based fallbacks when needed.

Hardware and reproducibility. Experiments are run on a single NVIDIA GeForce RTX 4060 GPU with a multi-core Intel i5-14400F CPU. Package versions are pinned, random seeds are fixed across NumPy, PyTorch, and Lightning, and the best-performing validation checkpoint is retained for final test evaluation. Data snapshots, scaler and imputer parameters, configuration files, and code version identifiers are logged to support reproducibility.

Classical baselines. For ARIMA and SARIMA, model orders are selected using validation-set MAPE. When multiple configurations yield similar performance, the simpler specification is preferred in order to preserve parsimony.

Deep learning models. For LSTM and TFT, hyperparameter tuning is performed over architecture and optimization settings using the validation block. TFT is trained with quantile loss over the set $Q = \{0.1, 0.5, 0.9\}$, while the median quantile is used for point forecasting. Table 4.4 summarizes the final configurations used in this chapter.

Table 4.4: Hyperparameter settings for the LSTM baseline and the proposed TFT model.

Setting	LSTM baseline	TFT model
Windows	Look-back L=16, horizon h=1.	max_encoder_length = 12, max_prediction_length = 1.
Architecture	Single LSTM layer, hidden size 64, dropout 0.10, batch size 32.	Hidden size 64, 4 attention heads, dropout 0.15, batch size 64, mixed precision (bf16).
Objective	Huber loss (δ=1.0).	Quantile loss with Q=$\{0.1, 0.5, 0.9\}$; median quantile used for point forecasts.
Optimizer	Adam (LR 5×10^{-3}, weight decay 1×10^{-5}), gradient clipping = 1.0.	AdamW (LR 1×10^{-3}, weight decay 1×10^{-5}), cosine LR schedule with 5% warm-up, gradient clipping = 1.0.
Early stopping	Patience = 5; restore checkpoint with lowest validation loss.	Monitor validation P50 loss; restore the lowest-loss checkpoint.

Evaluation Metrics The empirical analysis in Stage I focuses on one-quarter-ahead forecasting. Model performance is therefore evaluated at horizon $h = 1$ using the standard error metrics defined earlier in Section 2.6: MAE, RMSE, MAPE.

For all models trained on transformed targets, predictions are inverse-mapped to the original revenue scale before metric calculation. This ensures that reported errors remain economically interpretable and directly comparable across models.

4.4 Stage I: Next-Quarter Forecasting Results

We first evaluate all models on the one-quarter-ahead forecasting task, which serves as the quantitative baseline of the thesis. This stage addresses two core questions. First, does TFT improve predictive accuracy relative to classical and recurrent baselines under a strict chronological evaluation design? Second, if so, which structured feature groups appear to drive that improvement? The results show that TFT establishes a strong broad-panel benchmark at the short horizon, while the ablation and interpretability analyses clarify why the model performs well in this setting.

4.4.1 Training Dynamics

Before comparing forecasting accuracy across models, it is useful to verify that the TFT baseline trains stably under the selected optimization settings. TFT is trained for up to 50 epochs using the chronological split defined earlier, with the validation median quantile loss used for checkpoint selection. In practice, the best checkpoint typically occurs around epochs 20–25.

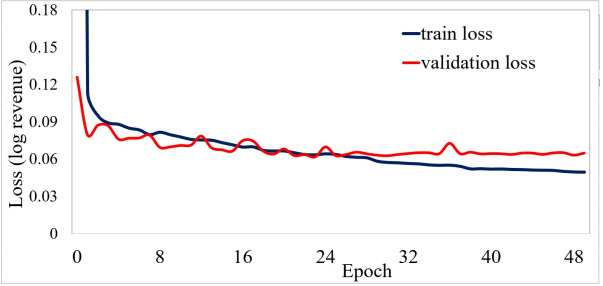


Figure 4.5: Training and validation loss curves of the TFT baseline across epochs.

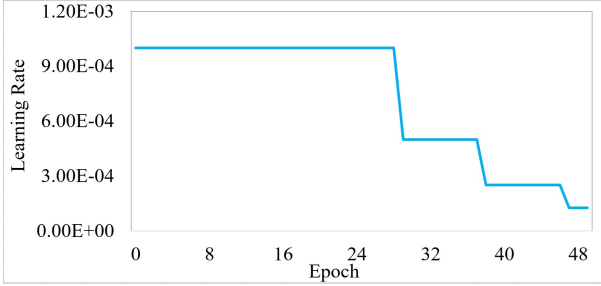


Figure 4.6: Learning-rate schedule used in TFT training.

As shown in Figure 4.5 and Figure 4.6, the optimization process is stable throughout training. The training and validation loss curves exhibit a rapid initial decline, indicating efficient early learning, while the subsequent flattening of the validation curve suggests stable convergence rather than unstable oscillation. At the same time, the learning-rate schedule follows a stepwise decay pattern, allowing the optimizer to make larger updates in the earlier epochs and finer adjustments in the later stages of training. Importantly, the validation loss does not diverge materially from the training loss, which provides evidence against severe overfitting under the selected dropout, early-stopping, and optimization settings.

4.4.2 Aggregate Panel Performance

Table 4.5 reports aggregate test-set performance across the 155-firm panel. TFT achieves the strongest overall results, with a mean test MAPE of 9.31%, substantially lower than the LSTM baseline (28.75%) and the classical Box–Jenkins benchmarks. Among the univariate statistical models, the best SARIMA specification $(3, 1, 2)(0, 0, 1)_4$ attains a mean MAPE of 23.01%, while the best ARIMA model $(1, 1, 1)$ yields 25.26%.

Absolute error metrics also favor TFT. On the held-out test set, TFT achieves an RMSE of 1,973 and an MAE of 1,790, compared with RMSE values above 3,300 for ARIMA/SARIMA and above 5,000 for LSTM. These results indicate a substantial performance advantage for TFT over both the recurrent and classical baselines in the next-quarter forecasting task, consistent with TFT's ability to combine heterogeneous covariates within a single structured forecasting framework.

The aggregate comparison in Table 4.5 is visualized in Figure 4.7 and Figure 4.8. Figure 4.7 emphasizes the substantial reduction in percentage forecasting error achieved by TFT relative to the benchmark models, while Figure 4.8 shows that the same ranking also holds for the absolute error metrics RMSE and MAE.

A firm-level view is provided in Figure 4.9 and Figure 4.10. Figure 4.9 plots per-ticker MAPE against realized revenue and shows that most firms cluster at relatively low percentage error, although dispersion

Table 4.5: Overall test-set performance for next-quarter revenue forecasting.

Model	MAPE (%)		RMSE		MAE	
	Mean	Std. Dev.	Mean	Std. Dev.	Mean	Std. Dev.
ARIMA $(1,1,1)$	25.26	43.77	3,578	6,163	3,109	5,475
SARIMA $(3,1,2)(0,0,1)_4$	23.01	14.00	3,312	5,507	2,801	4,791
LSTM	28.75	20.80	5,186	10,715	5,079	10,639
TFT	9.31	5.90	1,973	6,983	1,790	6,753

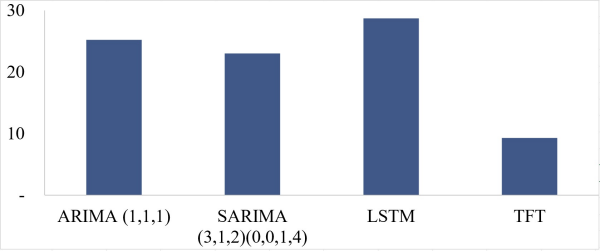


Figure 4.7: Comparison of mean test-set MAPE (%) for next-quarter revenue forecasting.

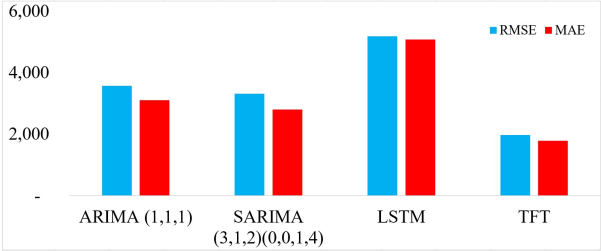


Figure 4.8: Comparison of mean test-set RMSE and MAE in million USD

increases among the largest firms. Figure 4.10 complements this view by plotting absolute error against realized revenue. As expected, absolute forecasting error tends to increase with firm size, but the broader pattern still indicates that the TFT maintains useful predictive performance across a wide range of revenue scales.

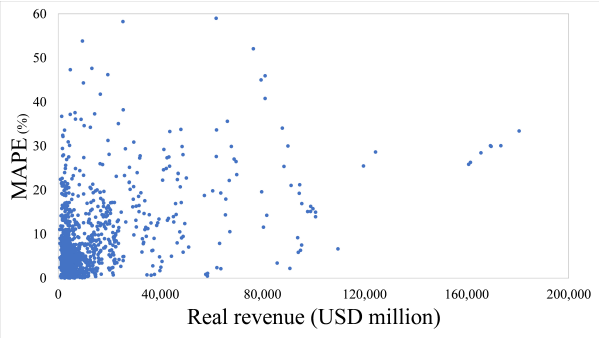


Figure 4.9: Per-ticker MAPE versus realized revenue for the TFT baseline on the test set.

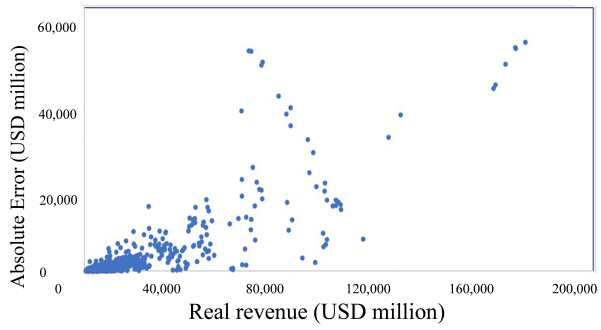


Figure 4.10: Per-ticker absolute error versus realized revenue for the TFT baseline on the test set

4.4.3 Sector-Wise Performance

To examine whether the aggregate advantage of TFT is broadly distributed across the panel or concentrated in a few sectors, Table 4.6 reports sector-wise mean MAPE for the next-quarter forecasting task. TFT achieves the lowest mean MAPE in every reported sector, although the size of the improvement varies materially across industries.

The gains are especially large in Information Technology, Materials, Health Care, and Industrials, where TFT substantially outperforms both ARIMA/SARIMA and the LSTM baseline. This pattern suggests that TFT benefits from its ability to combine autoregressive revenue history with broader structured firm context in settings where revenue dynamics are more heterogeneous. By contrast, the margins are somewhat smaller in comparatively stable sectors such as Consumer Staples and Utilities, where seasonality is more regular and classical models remain relatively more competitive.

The sector-level results therefore reinforce the aggregate findings: the broad-panel quantitative TFT baseline is not merely performing well on average, but delivers systematic gains across most industry groups.

Table 4.6: Sector-wise mean MAPE (%) for next-quarter revenue forecasting.

GICS Sector	ARIMA (1,1,1)	SARIMA $(3,1,2)(0,0,1)_4$	**LSTM**	**TFT**
Industrials	20.24	18.69	26.09	7.00
Communication Services	15.70	10.87	19.76	6.84
Materials	22.17	19.13	34.64	6.96
Consumer Discretionary	26.28	23.35	20.94	9.85
Consumer Staples	14.94	14.93	17.33	10.66
Health Care	16.87	19.51	33.93	10.01
Utilities	20.17	18.51	20.12	8.97
Information Technology	24.49	28.22	45.68	11.14
Energy	43.99	29.93	19.13	5.89
Financials	47.84	26.70	35.90	17.23

Note: (i) Lower values indicate better performance. (ii) Real Estate is omitted because the continuously listed cohort contains only one firm in that sector and is therefore not representative.

4.4.4 Ablation Study and Feature Importance

To identify which structured feature groups are most responsible for TFT's short-horizon forecasting gains, we conduct a controlled ablation study. Starting from the full TFT specification, one feature block is removed at a time while holding all other settings fixed, including the data split, preprocessing pipeline, model architecture, optimization procedure, and early-stopping rule. Performance is evaluated on the held-out test set so that all variants remain directly comparable.

Table 4.7 reports the full ablation results. First, **autoregressive revenue dynamics are indispensable**. Removing revenue lags causes a severe performance collapse, with MAPE rising from 9.31% to 42.47%, while RMSE and MAE also increase dramatically. This indicates that short-run revenue persistence and momentum are the dominant drivers of one-quarter-ahead predictability in this setting.

Second, **relative growth and firm context provide meaningful incremental signal**. Removing YoY features, sector embeddings, or balance-sheet scale proxies increases MAPE into the 11–13% range. These inputs therefore appear to add economically relevant context beyond pure lag structure, likely capturing seasonally adjusted growth, industry-specific dynamics, and firm-scale effects.

Third, **financial ratios contribute modest but non-zero value**. Excluding ratio-based covariates

increases MAPE only slightly, from 9.31% to 9.71%. This suggests that ratio features act more as incremental refinements than as primary drivers of short-horizon revenue prediction.

Table 4.7: Ablation results for the TFT baseline on next-quarter revenue forecasting.

Configuration	RMSE		MAE		MAPE (%)	
	Mean	Std. Dev.	Mean	Std. Dev.	Mean	Std. Dev.
Full features	1,973	4,985	1,791	4,865	9.31	5.90
No financial ratios	2,694	6,987	2,515	6,883	9.71	8.14
No YoY features	2,089	5,282	1,905	5,157	12.69	6.23
No sector features	2,958	7,490	2,764	7,332	11.52	8.33
No total asset features	2,458	5,580	2,227	5,353	11.33	7.43
No revenue lags	8,466	17,039	8,368	16,977	42.47	18.96

The performance shifts reported in Table 4.7 are visualized in Figure 4.11 and Figure 4.12. Figure 4.11 highlights the strong sensitivity of percentage forecasting accuracy to the removal of key feature groups, especially revenue lags. Figure 4.12 shows that the same pattern is also visible in the absolute error metrics, where the degradation associated with removing revenue lags is especially pronounced.

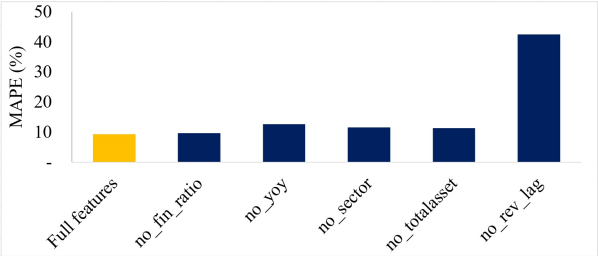


Figure 4.11: Mean test-set MAPE (%)across the TFT ablation variants.

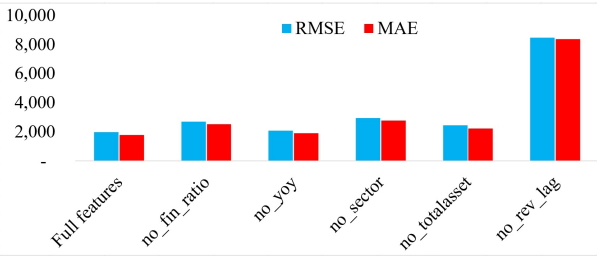


Figure 4.12: Mean test-set RMSE and MAE (in million USD)across the TFT ablation variants.

4.4.5 Interpretability of the One-Step Baseline

In institutional finance, predictive accuracy alone is not sufficient; forecasts must also be interpretable enough to support review, validation, and governance. For this reason, interpretability is treated as a first-class objective in the one-step baseline analysis. Because TFT provides built-in attribution mechanisms, the chapter emphasizes intrinsic interpretability, while also using targeted perturbation-style diagnostics to assess whether the learned attributions are economically plausible.

Interpretability toolkit. The interpretation framework combines several complementary views of the trained model:

1. **Variable Selection Networks (VSNs):** aggregated importance weights for static, observed-past, and known-future covariates;

2. **Interpretable multi-head attention:** horizon-specific attention over the encoder window, revealing which historical quarters the decoder relies on most strongly;
3. **Gating diagnostics:** inspection of GRN/GLU activity as a proxy for whether the model is using more linear or more nonlinear pathways; and
4. **Perturbation tests:** controlled shifts to selected inputs, such as calendar indicators, to verify that attribution patterns correspond to measurable sensitivity in forecast error.

Global attribution patterns. Across the full S&P 500 panel, the one-step TFT baseline consistently assigns the greatest importance to recent revenue lags and year-over-year growth features. The attention mechanism also tends to focus most strongly on the most recent one to four quarters of history. These patterns align closely with the ablation results in Table 4.7, where removing lagged revenue or YoY growth produces the largest error increases among the structured feature groups.

Scale-related features such as total assets also contribute, but their importance appears more heterogeneous across firms and sectors. This is consistent with their role as contextual scale proxies rather than dominant drivers. Sector embeddings provide useful static context as well, although their contribution is more incremental, particularly in a global model where part of the firm-level structure is already absorbed by the grouped panel design.

Sector-wise interpretation. When attribution patterns are summarized by sector, more differentiated behavior emerges. In Information Technology and Financials, the model places particularly strong weight on recent lags and relative growth features, reflecting the faster-changing and more idiosyncratic dynamics of these sectors. In Energy, Utilities, and Materials, by contrast, the model relies somewhat more on seasonality-related signals, which helps explain why SARIMA remains relatively more competitive there than in technology-oriented firms.

These attribution results are consistent with the sector-wise performance differences reported in Table 4.6. In more volatile sectors, TFT's flexible structured architecture appears to extract greater value from mixed covariates; in more stable seasonal sectors, the advantage over classical baselines narrows.

Interpretability takeaways. Overall, the one-step interpretability analysis supports three conclusions. First, autoregressive revenue structure is the primary driver of short-horizon forecasting performance. Second, growth and contextual firm characteristics provide meaningful secondary signal. Third, TFT's built-in attribution mechanisms produce explanations that are broadly consistent with the controlled ablation results, which strengthens confidence that the model is learning economically plausible patterns rather than relying on spurious correlations.

4.5 Stage II: Four-Quarter Forecasting under a Controlled Extension

The strong one-quarter baseline established in Stage I does not by itself guarantee practical usefulness in longer-horizon decision settings. In institutional finance, firms are often evaluated on a rolling annual

basis, which makes next-four-quarter forecasting especially relevant for valuation, capital allocation, and longer-term planning.

We therefore extend the same structured TFT framework from $h = 1$ to $h = 4$ under a controlled design, holding the dataset, preprocessing rules, and core model configuration as constant as possible. This allows forecast horizon itself to be isolated as the main experimental variable. To provide a clearer evaluation context, a recurrent LSTM model is also implemented under the same chronological setting, enabling a consistent comparison of multi-horizon forecasting performance across model architectures.

4.5.1 The Temporal Challenge and Multi-Horizon Reformulation

The short-horizon results in Stage I showed that TFT can produce a strong next-quarter benchmark when trained on structured financial fundamentals alone. However, the practical forecasting problem is often broader than immediate next-quarter prediction. Investors, analysts, and portfolio managers frequently need visibility over a full rolling year rather than a single quarter. This raises a central empirical question: can a structured forecasting architecture trained only on historical financial statements maintain its predictive strength when the horizon is extended from one quarter to four?

This question is especially important because quarterly financial statements are inherently backward-looking. They summarize realized business performance, but they do not fully capture future strategy shifts, demand inflections, or emerging innovation cycles at the forecast origin. As the forecasting window lengthens, the informational lag embedded in structured accounting data may therefore become more consequential.

To examine this issue, the one-step forecasting framework is extended to a multi-horizon setting. As illustrated in Figure 4.13, the maximum prediction length of the TFT decoder is expanded from $H = 1$ to $H = 4$. Instead of producing only the next-quarter forecast, the model now outputs a four-quarter path of revenue predictions. The multi-horizon mapping at forecast origin t is defined as Equation 2.3.

This reformulation preserves the same structured information taxonomy used in Stage I while extending the forecast target from a single next quarter to an entire annual horizon.

4.5.2 Controlled Extension from $h = 1$ to $h = 4$

To isolate the effect of forecast horizon as cleanly as possible, the multi-horizon experiments retain the same structured panel, preprocessing rules, and chronological split used in Stage I. The static covariates, observed-past fundamentals, and known-future calendar features are therefore unchanged. This design ensures that the empirical comparison between one-quarter and four-quarter forecasting is driven primarily by horizon extension rather than by a different data construction process.

The model configuration is also kept as consistent as possible. The optimization framework remains the same, using AdamW, the same learning-rate schedule, and the same leakage-free 70–15–15 chronological split. Within this controlled design, two implementation changes are necessary.

First, the decoder output is extended from a single forecast step to a four-step sequence. At each forecast origin t, the model now produces the joint forecast path $(\hat{y}_{i,t+1}, \hat{y}_{i,t+2}, \hat{y}_{i,t+3}, \hat{y}_{i,t+4})$ for each firm. Second, the evaluation procedure is modified so that error metrics are computed separately for each horizon. This

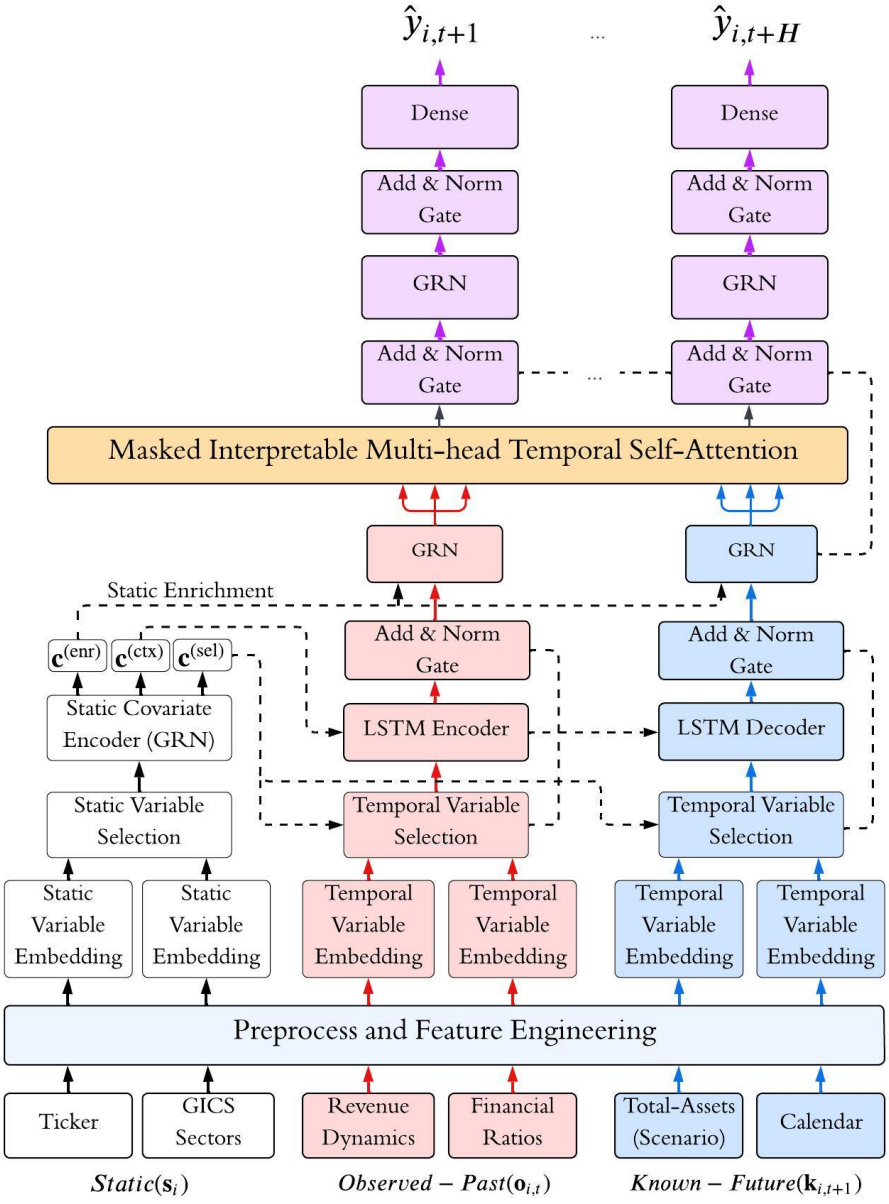


Figure 4.13: Architectural adaptation of the Temporal Fusion Transformer for multi-horizon forecasting.

is important because standard multi-step forecasting pipelines often report aggregated error over all output steps, which can obscure horizon-specific deterioration. In this chapter, MAE, RMSE, and MAPE are therefore calculated independently for each of the four forecast horizons.

4.5.3 Aggregate Horizon Degradation

Evaluating the multi-horizon forecasting performance on the held-out test set reveals a clear deterioration in accuracy as the horizon lengthens. Table 4.8 reports the horizon-specific RMSE, MAE, and MAPE values for both TFT and LSTM under the four-quarter forecasting setting.

Table 4.8: Forecast horizon degradation comparison between the pure financial TFT and LSTM.

	TFT			LSTM		
Horizon	**RMSE**	**MAE**	**MAPE (%)**	**RMSE**	**MAE**	**MAPE (%)**
Horizon ($t+1$)	1,973	1,790	9.31	4,983	4,870	28.36
Horizon ($t+2$)	2,207	2,086	11.23	4,949	4,839	28.97
Horizon ($t+3$)	2,268	2,138	11.21	5,079	4,969	29.92
Horizon ($t+4$)	2,381	2,244	12.07	5,323	5,221	30.09

The comparison shows two consistent patterns. First, TFT outperforms LSTM across all horizons in terms of RMSE, MAE, and MAPE, indicating that the structured temporal design of TFT is more effective than the recurrent baseline for this forecasting task. Second, both models exhibit horizon degradation, with errors increasing as the prediction horizon extends.

For TFT, MAPE rises from 9.31% at Horizon ($t+1$) to 12.07% at Horizon ($t+4$), an absolute increase of 2.76 percentage points, or approximately 29.6% relative to the one-quarter-ahead error. Over the same range, RMSE increases from 1,973 to 2,381 million USD and MAE rises from 1,790 to 2,244 million USD. A similar pattern is observed for LSTM, where MAPE remains substantially higher and increases from 28.36% to 30.09% as the horizon extends. Although the degradation is not perfectly monotonic at every intermediate step, the overall trend is clear: predictive accuracy weakens as the forecast horizon expands for both models.

These results provide explicit empirical evidence of horizon degradation in purely financial forecasting, and show that this effect is not specific to TFT but also appears in alternative sequence-based models such as LSTM. In the short horizon, models can rely on recent revenue persistence, near-term growth dynamics, and structured firm context. As the forecast extends to a full year, however, the predictive value of these lagging fundamentals declines. The widening error spread therefore suggests that structured historical financial information has a limited predictive half-life when used on its own.

4.5.4 Sector-Wise Heterogeneity and Technology Vulnerability

The aggregate results show that horizon degradation is real, but they do not reveal whether it affects all sectors equally. To examine the cross-sectional pattern of this deterioration, Table 4.9 reports sector-wise mean MAPE across all four forecast horizons based on the TFT baseline model. The sector analysis focuses on TFT to maintain a consistent structured forecasting framework across the full S&P 500 panel.

The sector-level results reveal that the horizon penalty is strongly heterogeneous. Some sectors, such as Energy and Communication Services, remain comparatively stable across the forecasting window. Their error profiles fluctuate only modestly from $h = 1$ to $h = 4$, suggesting that their revenue dynamics are more persistent and that historical financial structure retains predictive usefulness over longer horizons.

Other sectors exhibit much stronger deterioration. Materials, for example, rises from 6.96% MAPE at $h = 1$ to 12.26% at $h = 4$, while Utilities and Healthcare also show clear long-horizon deterioration. These sectors appear more vulnerable to delayed recognition of macro-sensitive or operational shifts that are only imperfectly reflected in trailing accounting data.

Table 4.9: Sector-wise mean MAPE (%) across forecast horizons (h=1 to h=4).

Sector	Horizon (t+1)	Horizon (t+2)	Horizon (t+3)	Horizon (t+4)
Energy	5.89	5.90	5.22	5.59
Communication Services	6.84	6.50	6.35	6.69
Materials	6.96	10.94	11.30	12.26
Industrials	7.00	8.11	8.53	9.33
Utilities	8.97	9.27	10.56	12.13
Healthcare	10.01	12.17	11.81	12.86
Consumer Staples	10.83	10.52	9.50	10.00
Information Technology	11.14	12.87	13.65	14.90
Consumer Discretionary	9.85	10.77	9.39	11.62
Financial Services	17.94	19.61	19.38	19.23

The most important pattern for the broader thesis, however, emerges in the Information Technology sector. Although the one-quarter-ahead error is still moderate at 11.14%, the error increases steadily as the horizon extends, reaching 14.90% at $h = 4$. Excluding the inherently difficult Financial Services sector, this is one of the most pronounced long-horizon deterioration patterns in the panel. This result suggests that purely financial statements are especially limited in technology-oriented settings, where revenue dynamics are more strongly influenced by rapid innovation cycles, product transitions, and structural regime shifts.

Taken together, the sector-wise evidence strengthens the main conclusion of Stage II. Horizon degradation is not merely an aggregate averaging effect; it is concentrated most strongly in precisely those sectors where forward-looking information is likely to matter most. This provides the empirical bridge to the next chapter, where multimodal forecasting is introduced as a possible remedy for the long-horizon weakness of purely quantitative models.

4.5.5 Interpretability Analysis of Horizon-4 Feature Weights

To complement the predictive results reported in Stage II, this subsection opens the black box of the four-quarter TFT model by examining its variable-selection weights. While the $h = 1$ baseline used ablation experiments to identify which inputs were most influential for next-quarter forecasting, the $h = 4$ setting provides an additional interpretability perspective through the internal weighting mechanism of the TFT. In particular, the variable-selection networks assign different relative importance to static features, observed-past inputs in the encoder, and known-future inputs in the decoder. Although these weights should not be interpreted as causal effects, they provide a useful indication of which types of information the model relies on most when forecasting revenue four quarters ahead.

Figure 4.14 reports the variable-selection weights for the static inputs. The most dominant static factor is `revenue_log_scale`, which accounts for approximately 60% of the total static importance. This result suggests that long-horizon forecasting remains strongly conditioned on the firm-level scale of revenue. In contrast, `gics_sectors`, `ticker`, and `encoder_length` each receive moderate weights of roughly 12%, while `revenue_log_center` contributes only a small share. Overall, the static-weight profile indicates that

the four-quarter TFT relies primarily on persistent cross-sectional structure, especially firm size, while sector identity and firm-specific embeddings provide secondary contextual support.

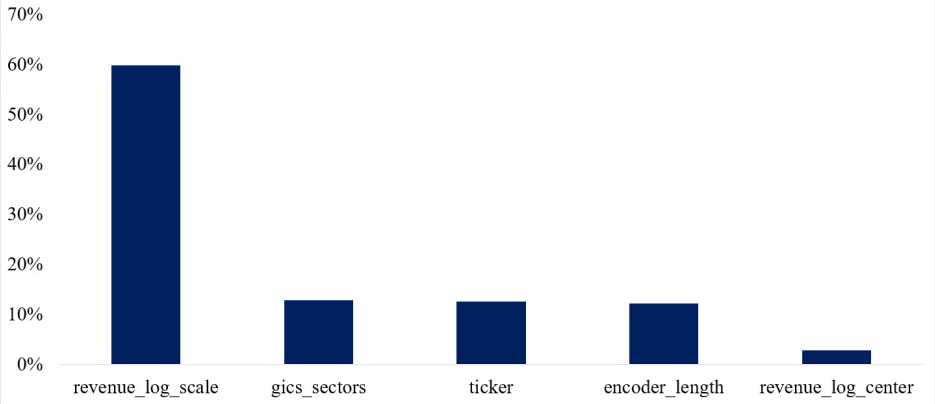


Figure 4.14: Static feature weights in the Horizon-4 TFT model

Figure 4.15 presents the encoder-side weights for observed-past covariates. The lagged target `revenue_log_lag1` receives the highest weight, at roughly 35%, which confirms that historical revenue remains the single most important dynamic signal even at a longer forecasting horizon. The second most important variable is `totalAssets_lag1_log`, at around 15%, followed by `grossProfit`, `costOfRevenue`, and `totalEquity_lag1_log`. By comparison, variables such as `ebitda`, `revenue_log`, `OIncomeRatio_yoy`, `rnd_to_rev_ratio_yoy`, and `grossProfitRatio` receive relatively small weights. This pattern suggests that, under the Horizon-4 setting, the model places greater emphasis on revenue persistence and broad balance-sheet scale than on short-term ratio-based indicators.

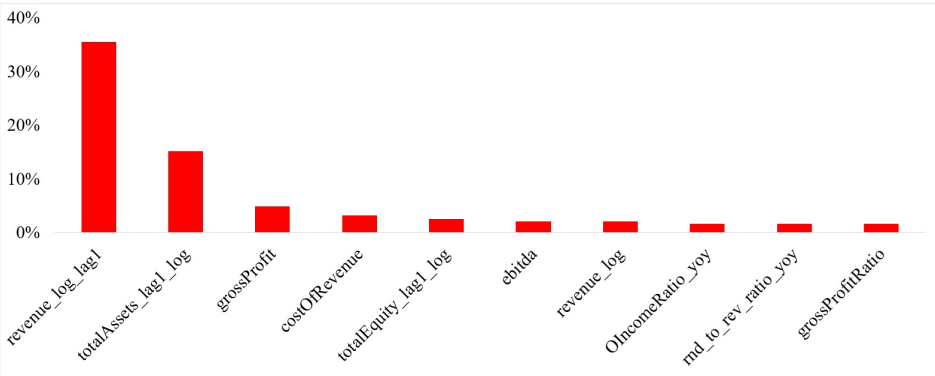


Figure 4.15: Encoder weights for observed-past features in the Horizon-4 TFT model

Figure 4.16 shows the decoder-side weights for known-future inputs. Here, `totalAssets_lag1_log` is the most influential decoder variable by a large margin, with a weight of 57.6%. The next most important inputs are `year` (17.2%) and `totalEquity_lag1_log` (13.7%), followed by `quarter_int` (8.1%) and `relative_time_idx` (3.5%). This distribution indicates that, when predicting four quarters ahead, the decoder depends much more on persistent firm-scale information and broad temporal position than on a simple seasonal quarter indicator alone. In other words, long-horizon forecasting appears to be driven less

by short-run fluctuations and more by stable structural characteristics and macro-temporal context.

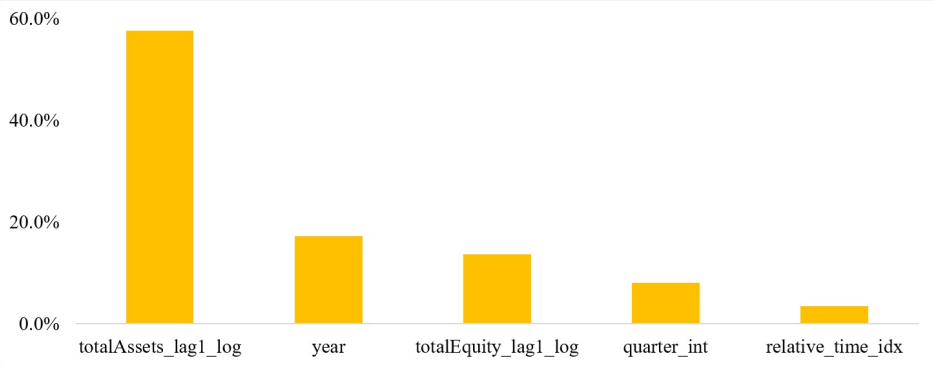


Figure 4.16: Decoder weights for known-future features in the Horizon-4 TFT model

So, these three groups of weights reveal a consistent pattern. Compared with the $h = 1$ setting, where ablation analysis highlighted the marginal contribution of individual inputs to near-term prediction accuracy, the Horizon-4 interpretability results suggest that the model shifts toward more persistent and structural signals. Firm scale, lagged revenue, total assets, and total equity dominate the weighting structure, whereas many fine-grained profitability and ratio variables receive much smaller attention. This finding helps explain why long-horizon corporate revenue forecasting is substantially more difficult: as the forecast window expands, the model relies increasingly on slow-moving structural anchors rather than short-term operational details. Therefore, the black-box analysis in Stage II not only improves model transparency, but also provides an economic interpretation of how the TFT adapts its information usage under longer forecast horizons.

4.6 Robustness Checks and Practical Implications

To assess whether the empirical findings reported in this chapter are stable rather than incidental, this section examines the robustness of the quantitative TFT framework under alternative architectural and preprocessing choices. It also considers the practical implications of the results for institutional forecasting. The objective is twofold: first, to determine whether the short-horizon advantages of the TFT baseline remain stable under reasonable specification changes; and second, to clarify the circumstances under which purely quantitative forecasting is operationally useful, as well as the point at which its long-horizon limitations become materially important.

4.6.1 Sensitivity to Encoder Length, Capacity, and Preprocessing

The main results of this chapter are supported by several robustness checks. At a broad level, the TFT baseline exhibits three reassuring properties: stable training dynamics, persistent outperformance under a strict leakage-free protocol, and sensitivity patterns that are economically interpretable rather than erratic.

First, the training behavior is stable. As shown earlier in Figure 4.5, the training and validation losses decline smoothly and flatten without a pronounced widening generalization gap. This indicates that the selected regularization and early-stopping settings are sufficient to control overfitting in the one-step forecasting task.

Second, the baseline ranking remains consistent under the common chronological evaluation design. As reported in Table 4.5, TFT achieves the lowest MAPE, RMSE, and MAE among all evaluated models. Because all models are trained and selected under the same non-overlapping train, validation, and test windows, these gains cannot be attributed to random reshuffling or to leakage from future periods into model selection. Stateful preprocessing objects are fit only on the training block, validation is used exclusively for tuning and checkpoint selection, and the test block remains untouched until final scoring.

Third, the structured TFT framework remains stable under targeted sensitivity checks. In additional experiments, the encoder history length is varied over $L \in \{12, 14, 16\}$ while holding the remaining configuration fixed. This tests whether the next-quarter results depend excessively on a single look-back choice. The same logic is applied to capacity and regularization settings by varying dropout rates, hidden dimensions, and attention-head counts over reasonable ranges. Across these variations, the model rankings remain stable, indicating that the performance is not an artifact of a narrowly tuned hyperparameter configuration.

Preprocessing sensitivity is also examined by re-running the pipeline under modified but still reasonable data-handling choices, including sector-demeaned standardization and alternative winsorization thresholds. The main comparative conclusion remains unchanged: TFT continues to outperform the classical and recurrent baselines. This consistency suggests that the reported gains reflect genuine modeling advantages rather than dependence on a fragile preprocessing recipe.

4.6.2 Scale Invariance and Error Behavior Across Firm Size

An important concern in broad-panel corporate forecasting is whether the model performs reliably across firms of very different scale. Figure 4.9 addresses this issue by plotting per-ticker MAPE against realized revenue. The figure shows that relative forecasting error remains moderate for the majority of firms across the panel, even though absolute dollar volatility naturally rises with firm size. Error dispersion does increase among very large firms, which is expected because revenue fluctuations for mega-cap companies can span billions of dollars from quarter to quarter. However, many of the largest firms still remain within a relatively low MAPE range. This indicates that the TFT baseline preserves useful relative accuracy even in high-scale settings, rather than performing well only on smaller and less consequential firms.

From a practical standpoint, this result is important because it suggests that the short-horizon structured baseline is not restricted to a narrow subset of low-scale firms. Instead, it appears capable of supporting broad-panel screening and comparison, even when the panel includes firms with highly heterogeneous revenue levels.

4.6.3 Model Risk Management Considerations

Deploying a structured forecasting model in practice requires explicit model-risk controls. The results of this chapter point to three broad requirements.

Monitoring. Forecast quality should be tracked after each earnings release using rolling MAPE or SMAPE, the frequency of extreme forecast misses, and calibration diagnostics where probabilistic outputs are used. Persistent deterioration in these indicators may signal drift, regime change, or data-quality issues.

Re-validation under structural change. The sensitivity analysis suggests that some scale-related features can become less stable under strong cross-sectional heterogeneity or major economic disruption. This implies that the model should be re-validated after large macro shocks, sectoral dislocations, or accounting-regime changes rather than assumed to remain reliable indefinitely.

Control through model comparison. For high-impact cases, forecast review can be strengthened through structured comparison against simpler benchmarks such as SARIMA or against internal analyst expectations. Large disagreements should trigger a review checklist covering data freshness, corporate actions, one-off events, restatements, and unusual reporting behavior.

Taken together, these considerations suggest that the quantitative TFT baseline is operationally valuable, but only within an appropriately governed forecasting process. Its short-horizon performance is strong enough to justify practical use, yet the long-horizon evidence in Stage II also makes clear that structured historical fundamentals alone are not sufficient for stable annual-horizon forecasting in all sectors.

4.7 Chapter Summary

This chapter developed and evaluated the purely quantitative forecasting framework of the thesis through two sequential stages. In Stage I, it established a leakage-free TFT baseline for next-quarter revenue forecasting across a broad panel of 155 continuously listed S&P 500 firms. Under a strict chronological evaluation design, the TFT achieved a mean test MAPE of 9.31%, substantially outperforming the ARIMA, SARIMA, and LSTM baselines. The ablation and interpretability analyses further showed that short-horizon forecasting is driven primarily by autoregressive revenue dynamics, while sector identity, year-over-year growth, and scale-related financial variables provide meaningful incremental predictive value.

In Stage II, the same structured framework was extended from one-quarter-ahead forecasting to four-quarter-ahead forecasting in order to quantify horizon degradation under controlled conditions. A comparative evaluation with an LSTM baseline under the same chronological setting further confirmed that forecast accuracy deteriorates as the prediction horizon expands across different model architectures. This indicates that the predictive value of lagging financial fundamentals weakens materially over time. The deterioration was not uniform across sectors, and was especially pronounced in technology-oriented firms, where revenue dynamics are more strongly shaped by rapid innovation, non-linear growth, and structural regime shifts.

Taken together, the evidence in this chapter establishes both the strength and the boundary of purely structured quantitative forecasting. Historical financial statements can support a strong and interpretable short-horizon benchmark, but they are less sufficient for stable long-horizon forecasting in dynamic sectors. The results show that structured-only models become less accurate as the forecast horizon extends, suggesting that additional information, particularly forward-looking signals, is needed. This observation motivates the next chapter, which introduces a multimodal approach that incorporates earnings-call text to improve long-horizon forecasting.

Chapter 5

Multimodal TFT Forecasting with Earning-Call Narratives

5.1 Introduction and Chapter Roadmap

Chapter 4.7 established two findings that directly motivate the present chapter. First, a purely quantitative TFT can provide a strong and interpretable next-quarter revenue forecasting baseline when trained on structured financial fundamentals alone. Second, that same framework exhibits clear horizon degradation when extended from one-quarter-ahead prediction to four-quarter-ahead prediction, with the deterioration being especially pronounced in technology-oriented firms. These results suggest that the limitation is not simply architectural, but informational: lagging financial statements become less sufficient as the forecasting horizon expands and the revenue-generating environment becomes more nonlinear.

This chapter addresses that limitation by introducing a multimodal extension of the TFT framework for a focused Mega-Cap 5 technology companies comprising Apple, Microsoft, Amazon, Alphabet, and Nvidia. The central idea is to augment structured financial fundamentals with forward-looking narrative signals extracted from quarterly earnings call transcripts. Two NLP pipelines are evaluated: a finance-domain sentiment baseline based on FinBERT, and a richer generative feature-extraction pipeline based on a locally deployed, quantized Llama-3 8B model. These text-derived features are then integrated into the TFT through a leakage-safe dual-role temporal design.

Figure 5.1 summarizes the overall workflow of Chapter 5. As shown in the figure, the chapter begins with two synchronized data streams: structured financial fundamentals and quarterly earnings call transcripts. The structured variables are preprocessed into static, observed-past, and known-future covariates for the TFT, while the transcript data are cleaned, aligned, segmented, and tokenized before being passed through the FinBERT and Llama-3 pipelines. The resulting sentiment and semantic features are then incorporated into the multimodal TFT to generate revenue forecasts across multiple horizons. This roadmap therefore provides the conceptual bridge from raw multimodal inputs to leakage-safe long-horizon forecasting outputs.

The chapter proceeds in five steps. It first explains why purely numerical fundamentals exhibit structural blind spots in long-horizon forecasting and motivates the Mega-Cap 5 cohort as a specialized multimodal

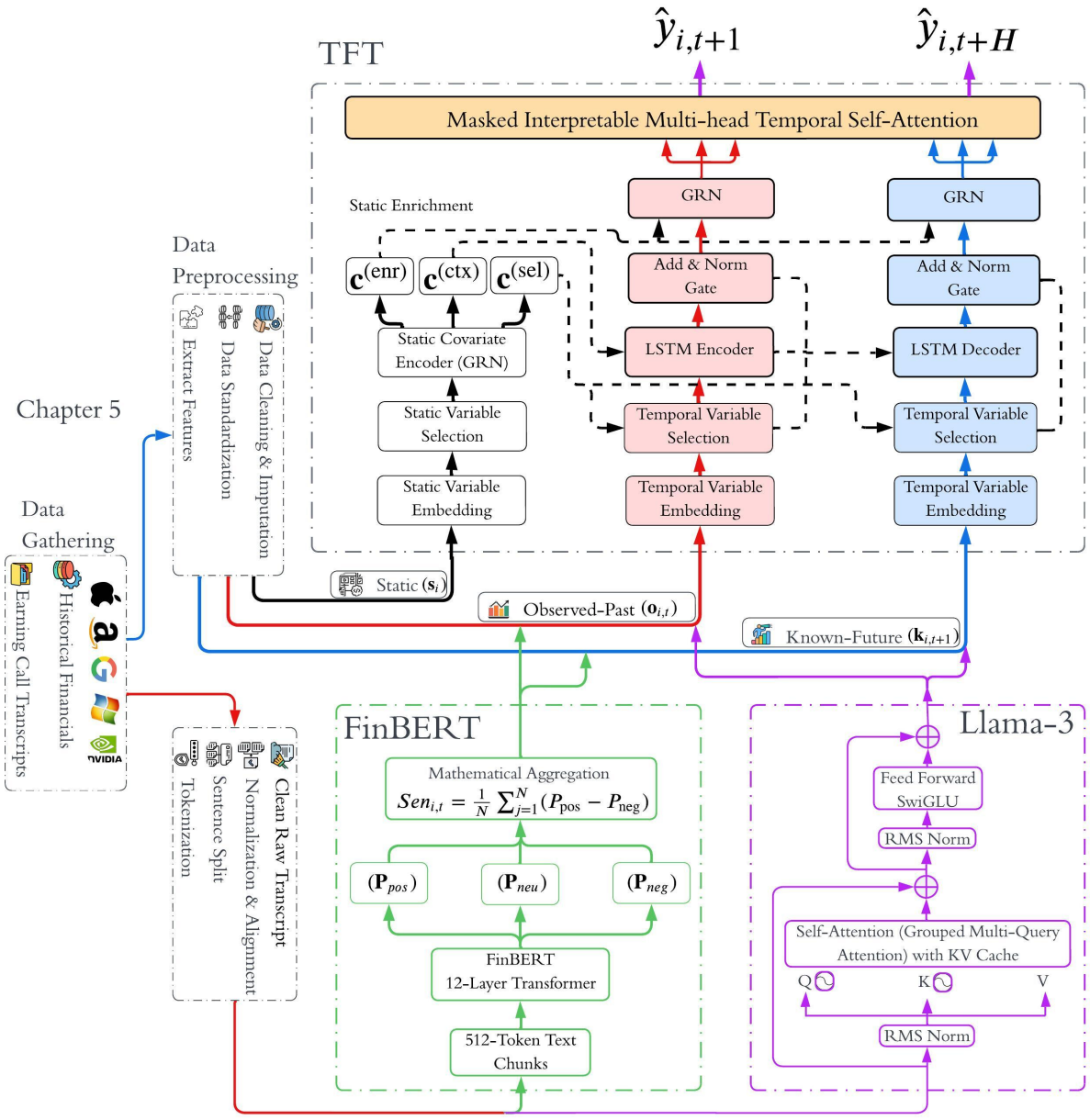


Figure 5.1: Roadmap of the multimodal forecasting framework in Chapter 5

testbed. It then formalizes the text-augmented forecasting problem, describes the structured and textual data pipelines, and presents the FinBERT and Llama-3 feature-extraction procedures. Next, it defines the controlled evaluation protocol used to compare the pure TFT, FinBERT+TFT, and Llama-3+TFT variants. The chapter then reports the comparative multi-horizon results and examines whether the multimodal architectures materially stabilize long-horizon forecasting. Finally, it analyzes the internal attribution patterns of the hybrid model and discusses the practical implications of the multimodal framework for institutional forecasting.

5.2 Why Purely Quantitative Forecasting Needs Narrative Augmentation

5.2.1 Structural Blind Spots of Quantitative Fundamentals

The horizon degradation documented in Chapter 4 does not imply that the TFT architecture is intrinsically unsuitable for long-horizon forecasting. Rather, it reveals a limitation of the information set supplied to the model. Structured financial statements contain economically meaningful data, but they remain backward-looking summaries of realized business performance. For immediate next-quarter forecasting, these lagging signals may still retain substantial predictive value. For longer horizons, however, two limitations become increasingly important.

Lagging measurement of business change. Core accounting variables such as revenue, operating income, total assets, and equity describe realized outcomes that have already passed through operational and reporting filters. They do not directly encode management expectations, strategy revisions, supply-chain adjustments, or demand inflections at the time they begin to matter economically. As the horizon extends from one quarter to a full year, this temporal lag becomes more costly because the model is increasingly asked to predict future business states using signals that mainly describe the recent past.

Exposure to structural breaks and nonlinear regimes. Purely numerical models are also vulnerable when the future departs materially from the historical pattern on which the model was trained. This issue is especially acute in large technology firms, where revenue can be affected by platform transitions, product super-cycles, cloud adoption waves, and sudden infrastructure booms. Such shifts are often discussed in managerial language before they are fully reflected in trailing accounting statements. As a result, a forecasting system that ignores narrative information may systematically underreact during structural transitions.

Taken together, these blind spots suggest that a structured-only forecasting framework may remain useful in stable short-horizon settings, but become increasingly incomplete in long-horizon, high-volatility, and innovation-driven regimes.

5.2.2 Deployment Risks and Institutional Governance

A second motivation for the multimodal design concerns deployment discipline. In real forecasting systems, both structured financial data and unstructured transcripts must satisfy strict point-in-time rules. Earnings calls occur after quarter end and often mix retrospective review with prospective guidance. If transcript-derived features are aligned carelessly, the resulting backtest may leak future information into the model and produce inflated performance.

For this reason, the multimodal architecture in this chapter is designed with institutional governance in mind. It uses only publicly available financial statements and earnings call transcripts, applies strict chronological alignment, and treats the resulting forecasts as decision-support outputs rather than stand-alone investment instructions. This emphasis on auditability is especially important once text-derived signals begin to influence long-horizon capital allocation.

5.3 Multimodal Problem Formulation and Framework Design

5.3.1 Mathematical Formulation of the Hybrid Forecasting Problem

To evaluate whether narrative information can mitigate horizon degradation, we focus on the Mega-Cap 5 technology cohort and formulate the multimodal problem as a multi-horizon panel forecasting task. At each forecast origin t, the model predicts future log-transformed revenues for firm i over the next $H = 4$ quarters, mathematical formulation is shown as Equation 2.4. This formulation preserves the same structured forecasting logic used in Chapter 4 while augmenting it with narrative features extracted from earnings call transcripts.

5.3.2 The Mega-Cap 5 Cohort as a Multimodal Testbed

The multimodal experiments are conducted on a focused Mega-Cap 5 family: Apple (AAPL), Microsoft (MSFT), Amazon (AMZN), Alphabet (GOOGL), and Nvidia (NVDA). This restriction is motivated by both substantive and operational considerations.

Table 5.1: Mega-Cap 5 Cohort Summary

Firm	Ticker	Business profile	Structured data span	Transcript span	Number of quarters / transcripts	Market Cap / Trillion
Nvidia	NVDA	Semiconductor design, accelerated computing, AI hardware, and data-center platform solutions.	2007Q1-2025Q2	2007Q1-2025Q2	73	4.42
Alphabet	GOOGL	Internet search, digital advertising, cloud services, and platform-based technology ecosystems.	2007Q1-2025Q2	2007Q1-2025Q2	73	3.76
Apple	AAPL	Consumer electronics, digital services, and integrated hardware–software ecosystem.	1995Q1-2025Q2	2005Q3-2025Q2	80	3.73
Microsoft	MSFT	Enterprise software, cloud computing, productivity platforms, and AI-enabled digital infrastructure.	1995Q1-2025Q2	2006Q1-2025Q2	77	2.97
Amazon	AMZN	E-commerce, cloud services, digital advertising, and logistics platform operations.	2007Q1-2025Q2	2007Q1-2025Q2	73	2.31

Substantively, these firms operate at the frontier of the modern digital economy and collectively span several distinct revenue-generating models: consumer hardware cycles, enterprise software and cloud migration, digital advertising ecosystems, e-commerce logistics, and AI infrastructure demand. Their revenue trajectories therefore provide a demanding testbed for forecasting models intended to cope with nonlinear

strategic and macroeconomic regimes.

Operationally, the cohort offers long, relatively standardized histories of both structured financial data and earnings call transcripts over the period 2007–2025. This time span includes several economically distinct regimes, including the Global Financial Crisis, the post-crisis cloud transition, the 2020 pandemic shock, and the generative AI investment cycle. These episodes make the cohort especially suitable for testing whether narrative features can stabilize long-horizon forecasts under structural change.

5.3.3 A Multimodal Framework for Narrative-Augmented Forecasting

The proposed framework augments the structured TFT pipeline with a parallel narrative stream extracted from earnings call transcripts. As illustrated in Figure 5.2, the model processes two broad categories of information. The first consists of the quantitative branch already established in Chapter 4: historical fundamentals, static firm attributes, and deterministic calendar covariates. The second consists of text-derived narrative features generated by the FinBERT and Llama-3 pipelines.

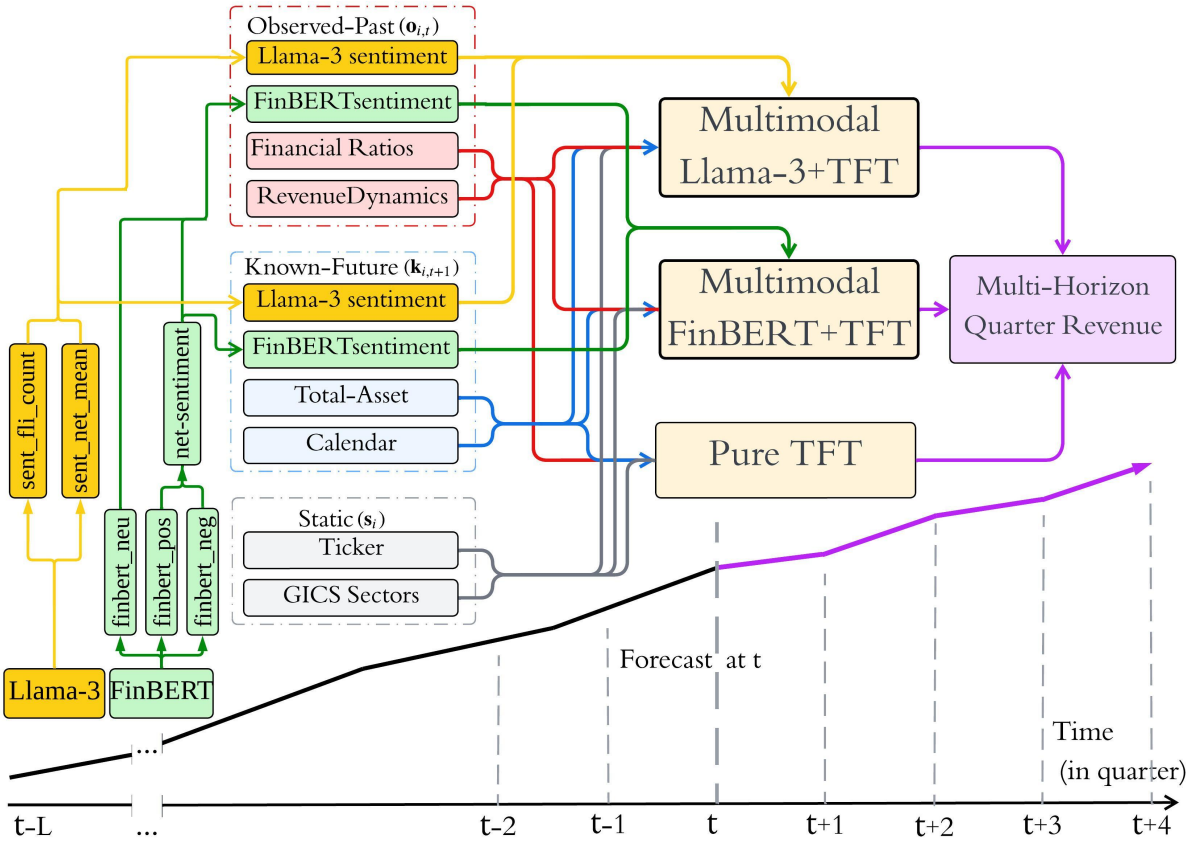


Figure 5.2: Architectural flowchart of the proposed multimodal forecasting framework.

The aim is not to replace structured fundamentals, but to complement them. Historical financial variables continue to anchor the revenue trajectory, while narrative features provide additional forward-looking context

that may be especially useful when the future diverges from the recent financial pattern.

5.3.4 The Dual-Role Sentiment Strategy

A central design challenge is how to integrate transcript-derived signals into a multi-horizon forecasting framework without introducing look-ahead bias. To address this, the multimodal experiments adopt a dual-role sentiment strategy. As shown in Figure 5.3, the same sentiment signal is allowed to enter the model in two distinct capacities.

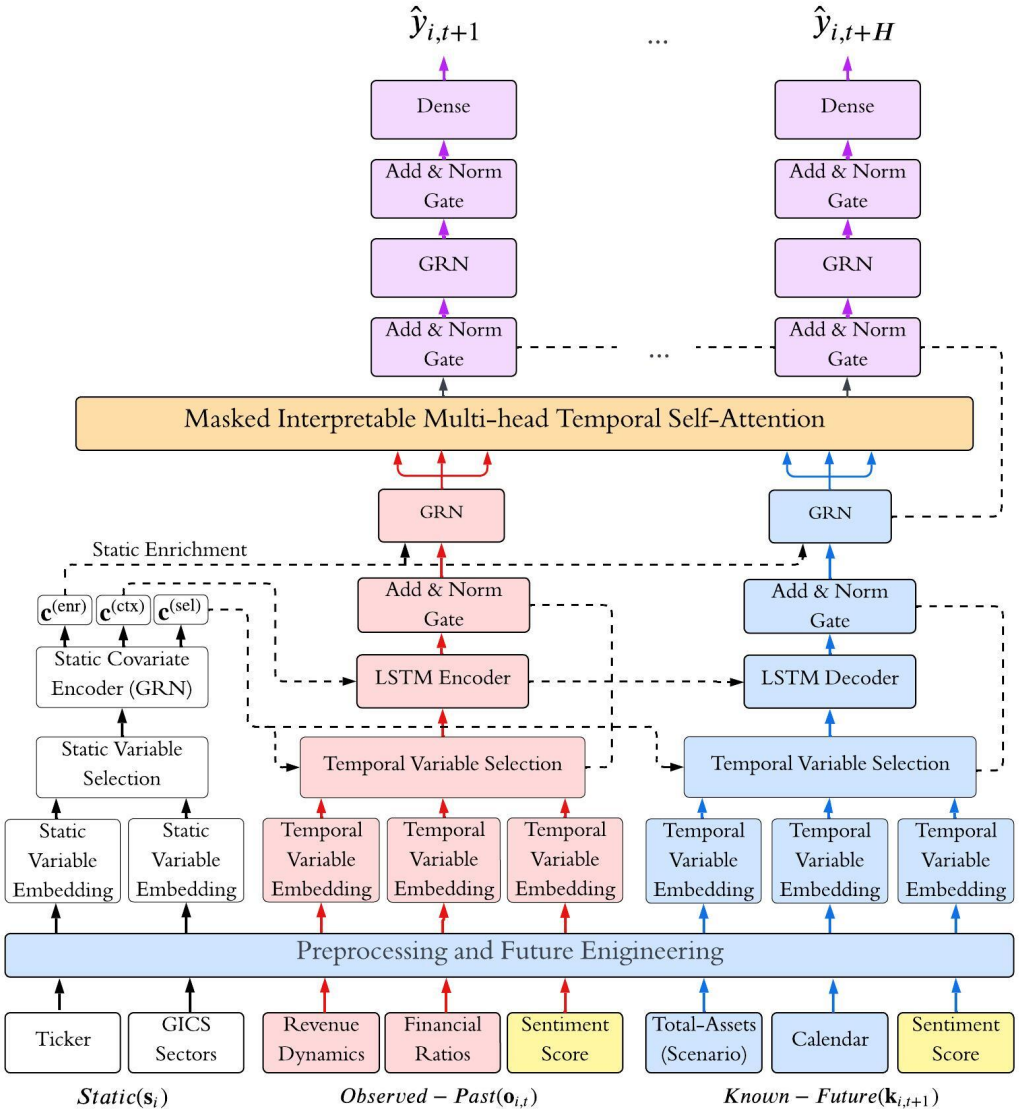


Figure 5.3: Dual-role sentiment integration in the multimodal TFT architecture.

1. Observed-history role. Within the encoder window, the sentiment signal is treated as a realized time-varying input. In this role, the model learns how historical managerial tone relates to subsequent realized

revenue.

$$o_{i,t}^{\text{sent}} = \text{Sen}_{i,t}. \tag{5.1}$$

2. Known-future contextual role. For the decoder, the most recently observed sentiment signal at origin t is forward-filled across the prediction horizon. This forward-filled tensor acts as a baseline narrative trajectory. Crucially, because it uses only information already available at time t, it does not violate point-in-time validity. The strategy therefore allows the decoder to retain the latest managerial tone as future context without using future transcripts.

$$k_{i,t+h}^{\text{sent}} = \text{Sen}_{i,t}, \qquad \forall h \in \{1, 2, 3, 4\}. \tag{5.2}$$

5.4 Structured and Textual Data Construction

5.4.1 Financial Data Acquisition and Calendar-Quarter Alignment

Structured numerical data are obtained from the Financial Modeling Prep (FMP) API, using the same broad modeling logic established in Chapter 4. Because the Mega-Cap 5 firms operate under different internal fiscal calendars, all reports are standardized to a calendar-quarter basis. This choice is important because it allows common macroeconomic and seasonal conditions to be aligned across firms even when their internal accounting calendars differ.

The primary target remains the log-transformed quarterly revenue, and the structured covariates include quarter-over-quarter growth, year-over-year growth, and key balance-sheet and margin-based features. Missing values are handled using only historically available information, and extreme outliers are winsorized to improve numerical stability during optimization.

5.4.2 Earnings Call Transcripts as Narrative Data

The textual branch of the framework is built from quarterly earnings call transcripts covering the Mega-Cap 5 firms over 2007–2025, which were download manually from the website Seeking Alpha (www.seekingalpha.com). Earnings calls are used because they provide direct managerial commentary on performance, strategic priorities, risk, and future expectations. Relative to third-party news or social media, they offer a more direct and firm-specific source of narrative information. Before feature extraction, the raw transcripts undergo a cleaning and segmentation pipeline. Boilerplate content such as operator instructions and legal disclaimers is removed where possible, and the remaining transcript is segmented into manageable text units for model input. This segmentation step is especially important because both encoder-based and decoder-based language models are constrained by finite context windows.

5.4.3 Temporal Alignment and Leakage Control for Text Features

The most important design rule in the textual pipeline is temporal alignment. Earnings calls are not mapped back to the quarter they discuss as if they were available at quarter close. Instead, transcript-derived

Table 5.2: Summary of NLP feature-extraction pipelines used in Chapter 5.

Pipeline	Model / checkpoint	Input unit	Output type	Aggregation level	Final features
FinBERT	`ProsusAI/finbert`	Pre-cleaned forward-looking sentences, tokenized with a maximum length of 512	Positive, neutral, and negative probabilities	Sentence → firm-quarter	`net_sentiment`, `finbert_pos`, `finbert_neg`, `finbert_neu`, `num_sentences`
Llama-3	Local `Llama3-Local` deployment of Meta-Llama-3-8B-Instruct	Prompted forward-looking sentences	Structured generative sentiment scores	Sentence → firm-quarter	`sent_net_mean`, `sent_fli_count`

features are aligned according to their actual availability at the forecast origin. This prevents the model from inadvertently using future narrative information while predicting earlier periods. In this way, the multimodal system retains the same leakage-safe philosophy used for the structured panel in Chapter 4.

5.5 Natural Language Processing Pipelines

This section introduces the two text-processing pipelines used to convert raw earnings call transcripts into structured numerical features for multimodal forecasting. Although both pipelines operate on the same cleaned transcript corpus, they differ substantially in modeling philosophy and feature granularity. As Table 5.2 summarize, FinBERT serves as a domain-specific sentiment baseline that maps forward-looking transcript sentences into discrete sentiment probabilities, whereas Llama-3 is used as a prompt-based generative extractor designed to capture richer narrative and forward-looking signals. In both cases, the final outputs are aggregated at the firm-quarter level so that they can be aligned with the structured financial panel and incorporated into the TFT under leakage-safe forecasting rules.

5.5.1 FinBERT as the Domain-Specific Sentiment Baseline

The first NLP pipeline uses FinBERT as a finance-domain sentiment baseline. Figure 5.4 shows that FinBERT builds on the bidirectional Transformer architecture of BERT, but is adapted to financial language through domain-specific pretraining and fine-tuning [10]. This makes it more suitable than generic language models for interpreting specialized financial vocabulary, earnings-call tone, and management language that may not be well represented in general-purpose corpora.

In the present study, the FinBERT pipeline is implemented using the checkpoint `ProsusAI/finbert`. Rather than processing an entire transcript as a single input, the script operates on a pre-cleaned TSV of forward-looking sentences with columns `ticker`, `Year_Quarter`, and `sentence`. Each sentence is tokenized independently under the standard BERT sequence constraint with truncation enabled and a maximum input length of 512 tokens. Inference is performed in batches of 32 sentences on GPU when available.

For each sentence, the model outputs a probability distribution over the classes positive, negative, and

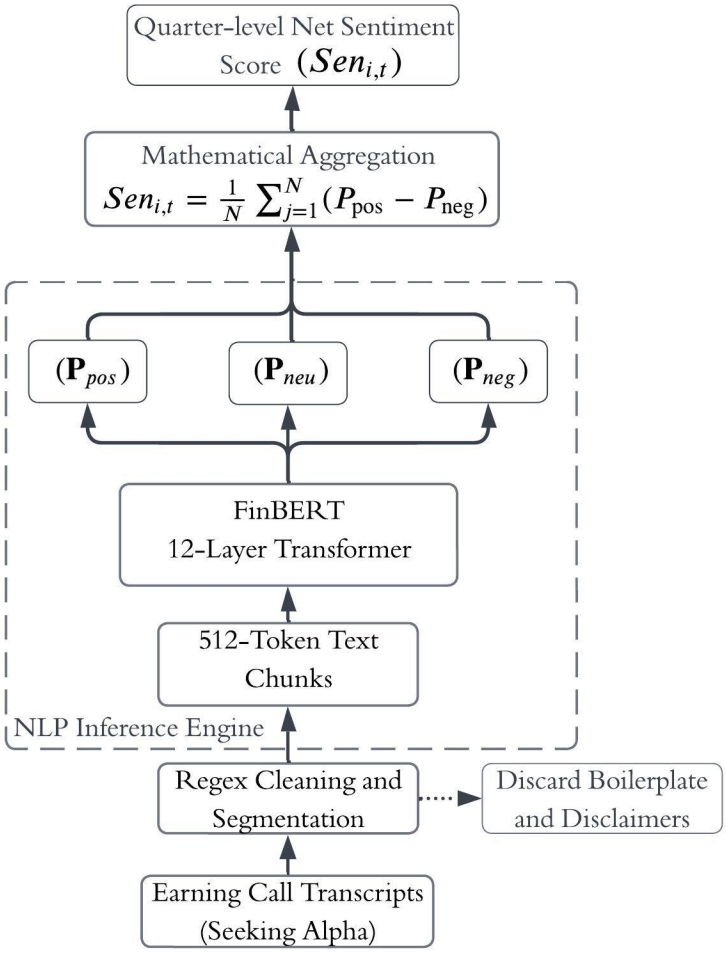


Figure 5.4: FinBERT sentence-level processing pipeline.

neutral:

$$\text{Softmax}(\mathbf{z})_c = \frac{e^{z_c}}{\sum_{j\in\{\text{pos,neg,neu}\}} e^{z_j}}, \tag{5.3}$$

where $c \in \{\text{pos}, \text{neg}, \text{neu}\}$.

These sentence-level probabilities are then aggregated into quarter-level sentiment variables. Let N denote the number of valid forward-looking sentences for firm i in quarter t. The main net sentiment score is defined as:

$$Sen_{i,t} = \frac{1}{N}\sum_{j=1}^{N}\left(P_{\text{pos},j} - P_{\text{neg},j}\right). \tag{5.4}$$

In implementation, the same averaging rule is also used to compute the quarter-level mean positive, negative, and neutral probabilities, stored respectively as `finbert_pos`, `finbert_neg`, and `finbert_neu`. The script also records `num_sentences`, the number of valid forward-looking sentences contributing to the quarterly aggregate.

The final quarter-level FinBERT feature definition is therefore sentence-based rather than transcript-block-based: each forward-looking sentence is scored independently, and the sentence-level outputs are averaged within each firm-quarter to form the final sentiment covariates. In downstream panel construction, these quarterly sentiment features are merged into the structured dataset under strict chronological ordering, lagged by one quarter, forward-filled when necessary, and initialized to zero at the beginning of the series to prevent look-ahead bias.

5.5.2 Llama-3 as a Generative Feature-Extraction Pipeline

While FinBERT provides a strong and interpretable finance-domain baseline, its sentiment output remains relatively coarse. To capture richer narrative structure, the second NLP pipeline uses a locally deployed Llama-3 8B model. Unlike FinBERT, which is optimized primarily for classification, Llama-3 is a decoder-only generative Transformer capable of instruction-following, prompt-conditioned inference, and structured feature extraction from forward-looking text.

In the present study, the Llama-3 pipeline is implemented through a local deployment path, `Llama3-Local`, corresponding to `Meta-Llama-3-8B-Instruct`. Because deploying an 8-billion-parameter model on a consumer-grade GPU is computationally demanding, inference is performed with 4-bit NormalFloat (NF4) quantization using `bitsandbytes`. More specifically, the script sets `load_in_4bit=True`. Model loading further uses `torch_dtype=torch.float16` and `device_map={"": 0}`, which confines inference to the local RTX 4060 GPU. The script notes an estimated VRAM footprint of approximately 6GB, which makes the pipeline feasible under the available 8GB hardware budget.

The Llama-3 pipeline uses prompt-based extraction rather than direct sentiment classification. Each forward-looking sentence is wrapped in a structured instruction prompt that asks the model to act as a financial analyst and rate management sentiment specifically with respect to future revenue and growth outlook on a strict scale from -1.0 to $+1.0$. The prompt requires the model to output only one numerical score without any explanation. This keeps the model in a focused scoring role and helps convert unstructured text into consistent numerical features that can be aggregated at the firm-quarter level. In this framework, the LLM is used as a feature extractor rather than the final forecasting model. The following prompt template is used to guide the LLM to produce structured numerical sentiment scores:

> **System instruction:** "You are a senior financial quant analyst. Read the forward-looking sentence from an earnings call transcript. Rate the management's sentiment specifically regarding the company's future revenue and growth outlook. Rate on a strict scale from -1.0 (extremely negative/pessimistic) to +1.0 (extremely positive/optimistic). Do not explain your reasoning. Output ONLY a floating-point number between -1.0 and +1.0."
>
> **User Input:** "Sentence: <forward-looking sentence>"

This design is not limited to Llama-3. We used Llama-3 because it follows instructions well and gives stable numerical scores. But it is not the only choice. Other large language models could also work in the same framework if they can follow the prompt and produce consistent outputs.

To improve reproducibility and control output length, decoding is performed with fixed settings: `max_new_tokens=10`, `temperature=0.01`, `top_p=0.9`, and `do_sample=True`. The tokenizer uses the end-of-sequence token as the padding token and applies left padding, which is standard for causal language-model generation. Inference is executed with a small batch size of 4, reflecting the memory constraints of the local 8GB GPU. If the generated text does not contain a valid numeric value, the parser falls back to a neutral score of 0.0.

Using this prompt-based generative inference, the Llama-3 pipeline produces two quarter-level features for each firm-quarter:

- `sent_net_mean`: the mean generated sentiment score across all valid forward-looking sentences in that quarter; and
- `sent_fli_count`: the count of forward-looking sentences, used as a proxy for forward-looking discussion intensity.

The quarter-level aggregation rule is therefore sentence-based and transparent. First, each forward-looking sentence receives a scalar sentiment score in $[-1, 1]$. Second, the sentence-level scores are averaged within each firm-quarter to form `sent_net_mean`. Third, the number of processed forward-looking sentences is recorded as `sent_fli_count`. These aggregated narrative variables are then aligned with the structured financial panel and passed to the hybrid TFT as additional temporal covariates.

Taken together, the two pipelines provide complementary forms of narrative information. FinBERT offers a transparent finance-domain sentiment baseline that is computationally efficient and easy to interpret. Llama-3, by contrast, is computationally heavier but capable of extracting a richer continuous representation of forward-looking managerial tone. The experimental comparison in later sections evaluates whether this added representational richness translates into stronger multi-horizon forecasting performance.

5.6 Experimental Setup and Evaluation Protocol

5.6.1 Fair-Comparison Design

To isolate the predictive contribution of text-derived features, the Pure TFT, FinBERT+TFT, and Llama-3+TFT models are compared under a strict fair-comparison protocol. All three variants use the same chronological split, the same aligned firm-quarter panel, and the same core TFT architecture. The only intentional architectural difference is the inclusion or exclusion of the textual feature tensors. This design is important because it prevents the multimodal models from benefiting from unrelated tuning advantages. As a result, the observed performance differences can be interpreted as the incremental value of narrative augmentation rather than the by-product of a different optimization regime or a larger forecasting backbone.

The comparison is also fair in a second sense: all structured inputs are held fixed across the three model families. Static variables, observed-past covariates, and known-future calendar features are constructed once and then reused in the same manner for all experiments. The hybrid models extend this common structured foundation by adding transcript-derived features, but they do not alter the basic forecasting task, the target

definition, or the horizon setting. In this way, the chapter compares three forecasting systems under a common experimental scaffold rather than three independently tuned pipelines.

5.6.2 Chronological Split and Leakage Control

Because the objective is to evaluate realistic forward-looking forecasting performance, the Mega-Cap 5 sample is partitioned chronologically rather than randomly. This is especially important in financial panel forecasting, where random shuffling can allow future information to contaminate earlier observations and thereby generate overly optimistic results. The present chapter therefore follows the same leakage-free evaluation philosophy established in Chapter 4, but applies it to the shorter multimodal sample formed by the intersection of structured fundamentals and earnings call transcript availability.

Table 5.3 summarizes the split design. The earliest portion of the sample is used for model fitting, the middle portion for validation and checkpoint selection, and the most recent portion for final out-of-sample testing. No random reshuffling is applied at either the firm level or the time level. This ensures that every reported test forecast is generated only from information that would have been available at the relevant forecast origin. No artificial temporal gap is introduced between the validation and test sets, as this choice reflects a realistic forecasting setting. Although temporal adjacency may increase similarity between consecutive periods, look-ahead leakage is still avoided because all model estimation and evaluation are conducted strictly in chronological order, without access to future observations.

Table 5.3: Chronological split protocol for the Mega-Cap 5 multimodal sample.

Split	**Relative share**	**Time period**
Training	Earliest 70%	2007Q1–2019Q4
Validation	Next 15%	2020Q1–2022Q3
Test	Final 15%	2022Q4–2025Q2

Leakage control is applied not only to the split itself but also to the treatment of text. Transcript-derived features are aligned at the firm-quarter level and are only allowed to enter the model once the corresponding earnings call would have been observable in real time. In other words, future transcripts are never permitted to inform earlier forecast origins. This rule is particularly important in the decoder portion of the TFT, where known-future variables must remain genuinely available in advance. The multimodal design therefore preserves temporal validity at both the dataset level and the feature-engineering level.

5.6.3 Hyperparameter Selection for the Shorter Mega-Cap 5 Sample

Although Chapter 5 inherits the same broad TFT design from Chapter 4, the multimodal sample is materially shorter because it is restricted to the 2007–2025 period and further constrained by transcript availability. This has an immediate consequence for sequence-length selection. In Chapter 4, a 12-quarter encoder was still reasonable because the structured panel spans three decades. In Chapter 5, however, the effective time range is much shorter, and every additional encoder quarter reduces the number of usable sliding windows

after requiring a four-quarter prediction horizon. For this reason, the encoder look-back window is reduced from 12 quarters to 8 quarters.

This adjustment is not merely a computational convenience. It reflects a deliberate trade-off between historical context and sample efficiency. A longer encoder may provide more distant history, but in a short panel it also discards more potential training examples and increases the chance that the model overfits a relatively small number of long sequences. By shortening the encoder to 8 quarters while preserving a 4-quarter prediction window, the model still observes two full years of recent operating history while retaining a more adequate number of effective training instances. This compromise is particularly appropriate for the Mega-Cap 5 cohort, where the objective is to test the incremental value of narrative features under realistic sample constraints rather than to maximize sequence length at all costs.

Table 5.4 reports the principal hyperparameters used in the multimodal experiments. Unless otherwise noted, the same settings are applied across the Pure TFT, FinBERT+TFT, and Llama-3+TFT variants so that model capacity remains broadly comparable.

Two aspects of Table 5.4 deserve emphasis. First, the encoder reduction from 12 to 8 quarters is the main horizon-specific adaptation introduced in Chapter 5. Second, the optimization and hidden-layer settings are intentionally kept close to the Chapter 4 baseline. This strengthens the identification logic of the chapter: the empirical gains should come primarily from narrative augmentation rather than from a substantially different network scale or training recipe.

5.6.4 Computational Constraints and Hardware Setup

The experiments are executed on a local workstation equipped with an NVIDIA GeForce RTX 4060 GPU. Under this hardware budget, FinBERT-based inference is manageable using standard batched processing, whereas Llama-3 inference must be quantized to remain feasible within the available 8GB VRAM constraint. In particular, the Llama-3 pipeline is deployed using 4-bit NF4 quantization, which substantially reduces memory usage while preserving enough semantic fidelity for quarterly feature extraction.

For this reason, transcript processing is treated as an offline feature-generation stage rather than an end-to-end jointly trained module. The raw transcripts are first cleaned, normalized, segmented, and tokenized; the resulting text representations are then converted into quarter-level features through either FinBERT sentiment scoring or Llama-3 prompt-based extraction. Once these features are aligned with the structured panel, downstream TFT training proceeds in the same way as in the structured baseline. This staged design is not only computationally necessary but also methodologically helpful, because it keeps the comparison transparent: the forecasting backbone remains the TFT, while the text models act as upstream feature generators.

5.6.5 Optimization Protocol

The multimodal variants are trained using the same AdamW-centered optimization logic as the structured baseline. The learning-rate schedule follows a cosine-decay design with a short warm-up phase, which is especially useful for Transformer-style architectures. The warm-up period prevents unstable large updates

Table 5.4: Core TFT hyperparameters used in the Chapter 5 multimodal experiments.

Hyperparameter	**Value**	**Purpose / rationale**
Target variable	`revenue_log`	Log-transformed quarterly revenue target used throughout the thesis.
Group identifier	`ticker`	Panel grouping variable for firm-specific sequence construction.
Maximum encoder length	8 quarters	Reduced from 12 in Chapter 4 to preserve sample efficiency in the shorter 2007–2025 multimodal panel.
Prediction length	4 quarters	Fixed four-quarter forecasting horizon for annual-lookahead evaluation.
Hidden size	64	Main latent dimension of the TFT backbone.
Hidden continuous size	8	Compression size for continuous-variable representations.
LSTM layers	1	Single recurrent layer balances expressiveness and overfitting control in a short sample.
Attention head size	4	Multi-head temporal attention with four heads.
Head dimension	16	Implied projection size per attention head.
Dropout	0.15	Moderate regularization to improve generalization under limited sample size.
Optimizer	AdamW	Same optimizer family as the structured baseline.
Initial learning rate	1×10^{-3}	Peak learning rate used together with warm-up and cosine decay.
Weight decay	1×10^{-5}	Mild weight regularization for stable optimization.
Loss function	QuantileLoss	Supports multi-quantile forecasting and uncertainty-aware training.
Output quantiles	0.02, 0.10, 0.25, 0.50, 0.75, 0.90, 0.98	Seven quantile outputs; the median forecast is used for point-metric evaluation.
Causal attention	True	Preserves temporal directionality in multi-horizon forecasting.
Best checkpoint epoch	82	Representative best-performing checkpoint in the reported multimodal run.

at the start of training, while the later cosine decay gradually reduces the step size and supports finer convergence in the later epochs. This scheduling choice is discussed in greater detail in Section 5.6.7, where the joint behavior of the loss curve and learning-rate trajectory is analyzed directly.

The main point for the experimental protocol is that the same optimization principles are applied across all three model families. This keeps the training conditions symmetric and further supports the claim that the final performance differences are attributable to the information content of the text-derived features rather than to privileged optimization settings.

5.6.6 Horizon-Specific Evaluation Metrics

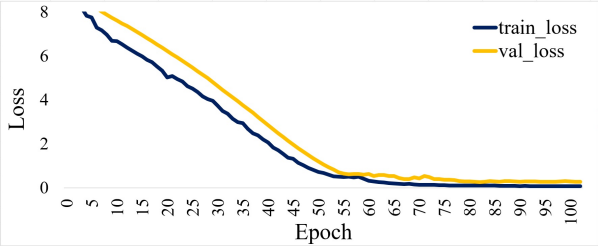
To evaluate whether multimodal augmentation stabilizes long-horizon forecasting, errors are computed separately for each horizon $\tau \in \{1, 2, 3, 4\}$. This is necessary because an aggregate multi-horizon metric can conceal deterioration at the far end of the forecast window. A model may perform reasonably well at $h = 1$ while failing badly at $h = 4$, and such behavior would be partially masked if only an averaged score were reported. The horizon-specific design therefore makes the stabilization question empirically visible.

The main evaluation metrics remain RMSE, MAE, and MAPE. RMSE and MAE are computed on the original revenue scale, while MAPE is reported in percentage terms to facilitate relative-error comparison across firms of different size. Because the TFT is trained with QuantileLoss and produces multiple forecast quantiles, the point-metric evaluation is based on the median forecast (the 0.50 quantile). In addition to reporting aggregate averages across horizons, the chapter therefore reports separate horizon-wise metrics so that the comparison between the Pure TFT baseline and the two hybrid variants remains transparent at both short and long forecast distances.

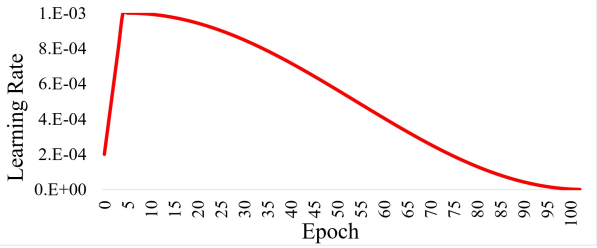
Taken together, these protocol choices define a conservative evaluation framework. The data split is chronological, text alignment is leakage-safe, model capacity is held broadly constant, optimization settings are shared, and performance is examined both in aggregate and by forecast horizon. This makes the Chapter 5 comparison intentionally strict. Any improvement obtained by FinBERT+TFT or Llama-3+TFT must therefore survive a design that is deliberately structured to minimize spurious experimental advantage.

5.6.7 Hybrid Training Dynamics

The multimodal TFT variants are trained under the same AdamW-based optimization framework as the structured baseline, using a cosine learning-rate schedule with a 5% warm-up phase. This scheduling strategy is especially important for Transformer-style architectures, because the optimization process is highly sensitive to the learning rate during the earliest training iterations. Figure 5.5 summarizes the optimization behavior in Chapter 5, with panel (a) showing the training and validation loss trajectories and panel (b) showing the corresponding learning-rate schedule.



(a) Training and validation loss trajectories during multimodal TFT optimization.



(b) Learning-rate schedule with 5% warm-up followed by cosine decay.

Figure 5.5: Training dynamics of the multimodal TFT variants in Chapter 5.

The learning-rate curve in Figure 5.5b follows a deliberate two-stage design rather than an arbitrary fluctuation. In the initial warm-up phase, the learning rate increases gradually from a near-zero value to the

preset peak. This stage allows the randomly initialized network to enter a stable optimization regime before large parameter updates are applied. At the beginning of training, both the loss magnitude and the associated gradients can be highly unstable. If the optimizer were to start immediately at the maximum learning rate, the model could overshoot useful descent directions, produce sharp oscillations, or even fail to converge. The warm-up stage therefore acts as a controlled stabilization period, allowing the multimodal TFT to form an initial representation of the structured and text-derived inputs before full-rate optimization begins.

After the warm-up phase, the schedule enters the cosine decay period, during which the learning rate decreases smoothly over the remaining training epochs. This second stage serves two complementary purposes. In the earlier part of the decay window, the learning rate remains sufficiently large to support fast progress and to help the model move across poor local regions of the loss surface. In the later part of training, however, the gradually shrinking step size becomes more important than speed. Once the model has approached a low-loss region, smaller updates allow it to refine the parameter values more carefully and reduce the risk of repeated overshooting around a good solution. Compared with a simple linear decay, cosine decay typically preserves a relatively strong learning signal for longer and then transitions more gently into the fine-tuning regime near the end of training.

The loss behavior in Figure 5.5a is consistent with this optimization strategy. Both the training and validation losses decline smoothly, with no evidence of severe instability or late-stage divergence. The early epochs show a rapid decrease in loss, indicating that the model quickly captures useful predictive structure once the warm-up phase has stabilized the optimization path. This is followed by a longer period of more gradual improvement, which aligns with the cosine decay stage in Figure 5.5b. In other words, the loss curve and the learning-rate curve should be interpreted jointly: the initial rise in learning rate supports safe acceleration, while the subsequent cosine decay supports controlled refinement.

Equally important, the gap between the training and validation losses remains well behaved throughout the optimization process. This suggests that the addition of transcript-derived features does not introduce destructive optimization noise or immediate overfitting pressure. Instead, the multimodal variants remain trainable under the same general optimization framework used in the structured baseline, which strengthens the practical case for extending TFT from purely numerical forecasting to a text-augmented setting.

Overall, these training dynamics provide an additional piece of evidence that the Chapter 5 multimodal design is not only empirically more accurate, but also optimization-feasible. The combination of warm-up and cosine decay produces a stable training trajectory, supports efficient convergence, and enables the hybrid architectures to absorb additional narrative features without sacrificing numerical training stability.

5.7 Empirical Results and Comparative Analysis

5.7.1 Aggregate Performance Comparison

To assess the value of the proposed multimodal extensions, the hybrid models are evaluated under the same chronological and leakage-aware protocol used for the structured baseline. Performance is explicitly compared across the structured-only TFT, FinBERT+TFT, and Llama-3+TFT models using standard forecasting metrics, including MAPE, RMSE, and MAE. This design enables a consistent and fair evaluation of the

incremental contribution of text-derived features.

Table 5.5 reports the aggregate error comparison across four forecasting architectures: LSTM, Pure TFT, FinBERT+TFT, and Llama-3+TFT. Among these models, LSTM achieves the lowest overall error on the Mega-Cap 5 sample. This suggests that sequence-based models are effective at capturing revenue dynamics in this setting. However, within the TFT-based family, the results show a clear and consistent improvement when textual information is added. Specifically, MAPE decreases from 53.85% in Pure TFT to 48.70% in FinBERT+TFT, and further to 43.01% in Llama-3+TFT. This indicates that earnings-call-derived narrative signals provide meaningful incremental value for improving multi-horizon forecasting within a structured temporal framework.

Table 5.5: Aggregate error comparison for the Mega-Cap 5 under four forecasting architectures.

Metric	**LSTM**	**Pure TFT**	**FinBERT+TFT**	**Llama-3+TFT**
RMSE	29,628	46,297	43,895	37,769
MAE	28,047	44,688	42,099	35,884
MAPE (%)	29.01	53.85	48.70	43.01

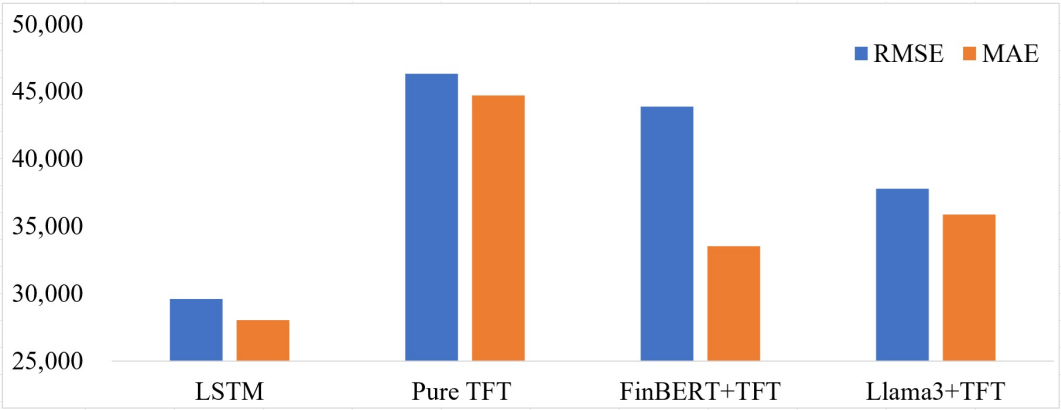


Figure 5.6: Aggregate RMSE and MAE comparison under the three forecasting architectures.

Figure 5.6 provides a visual comparison of RMSE and MAE across the four forecasting architectures. LSTM achieves the lowest overall error on the Mega-Cap 5 sample. However, within the TFT-based models, the inclusion of textual features consistently improves forecasting performance. Compared with Pure TFT, FinBERT+TFT reduces RMSE, MAE, and MAPE, and Llama-3+TFT delivers further gains. The RMSE reduction from 46,297 in Pure TFT to 37,769 in Llama-3+TFT is especially important because RMSE places greater weight on large forecast misses. This suggests that Llama-3-derived text features help the TFT framework better handle periods of high volatility or structural change.

5.7.2 Horizon-Wise Error Dynamics and Forecast Stabilization

The aggregate comparison becomes more informative when examined horizon by horizon. While the pure TFT baseline already performs weakly on the Mega-Cap 5 cohort at $h = 1$, its error remains elevated and

continues to drift upward as the forecast horizon extends. By contrast, both multimodal hybrids reduce MAPE at every step of the four-quarter prediction window. Table 5.6 reports the detailed horizon-wise comparison, and Figure 5.7 visualizes the same pattern.

Table 5.6: Horizon-wise MAPE comparison for the Mega-Cap 5 Group.

Metric	**LSTM**	**Pure TFT**	**FinBERT+TFT**	**Llama-3+TFT**
MAPE $h = 1$ (%)	28.37	52.97	46.93	40.29
MAPE $h = 2$ (%)	27.25	53.51	47.12	41.92
MAPE $h = 3$ (%)	28.36	54.05	49.33	43.71
MAPE $h = 4$ (%)	32.07	54.87	51.41	46.11

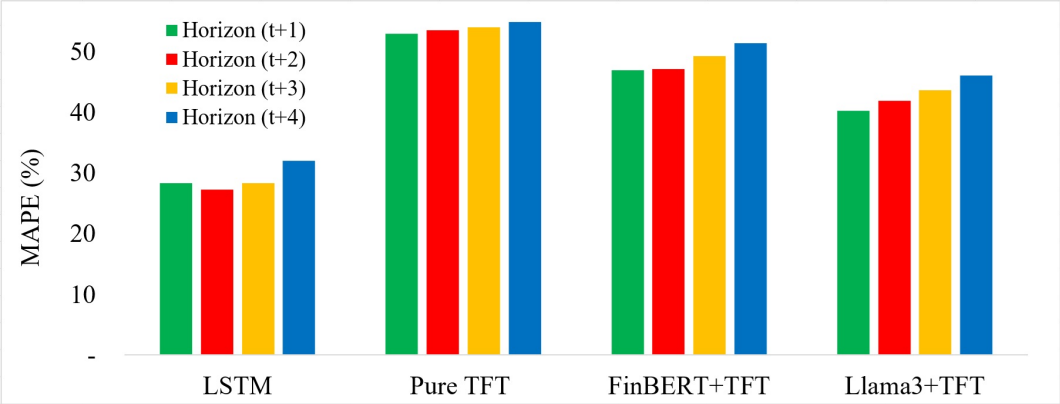


Figure 5.7: Horizon-wise MAPE comparison from $h = 1$ to $h = 4$.

The horizon-wise comparison shows a clear pattern across the four-quarter forecasting window. LSTM achieves the lowest MAPE at each horizon on the Mega-Cap 5 sample. However, within the TFT-based models, narrative augmentation consistently improves forecasting performance at every step. Compared with Pure TFT, FinBERT+TFT reduces MAPE from 52.97% to 46.93% at $h = 1$ and from 54.87% to 51.41% at $h = 4$. Llama-3+TFT delivers larger gains, further reducing MAPE to 40.29% at $h = 1$ and 46.11% at $h = 4$. This suggests that text-derived features help mitigate horizon degradation by adding forward-looking narrative information to the structured financial inputs.

5.7.3 Computational Trade-Offs and Deployment Feasibility

The predictive gains of the multimodal framework come with non-trivial computational cost. Querying structured tabular financial data is relatively light, while extracting features from thousands of long earnings call transcripts through large language models is considerably heavier. In particular, the Llama-3 pipeline requires quantized local inference and batch-style preprocessing.

However, this cost is not incurred during every forecast query. Earnings calls are quarterly events, so the transcript-processing step can be handled asynchronously as an offline preprocessing stage. Once the text-derived features are generated, appended, and aligned, TFT inference remains fast. From a deployment

perspective, this architecture therefore shifts complexity upstream into quarterly feature generation rather than into live forecasting latency.

5.8 Explainability of the Multimodal TFT

One advantage of the TFT backbone is that the multimodal extension remains internally interpretable. Because the model still uses variable selection networks (VSNs) and attention-based temporal attribution, it is possible to observe how the architecture balances structured fundamentals against narrative inputs.

5.8.1 Static and Temporal Attribution Patterns

Figure 5.8 and Figure 5.9 summarize the model's temporal attention and static variable importance patterns. The attention structure indicates that the model does not rely uniformly on all historical quarters; rather, it places concentrated weight on specific lag windows. The static attribution view further shows that the model continues to use persistent firm-level context, rather than discarding structured information after text is introduced.

Figure 5.8: Multi-head attention weights across historical time steps.

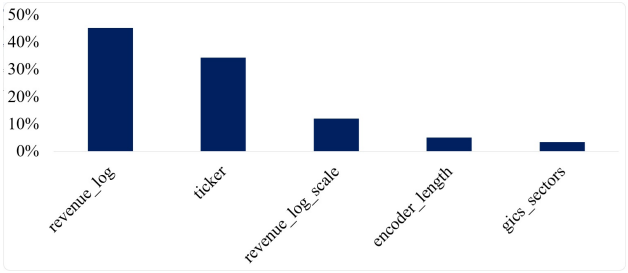


Figure 5.9: Static variable importance.

5.8.2 Variable Selection Network Weights

The most direct evidence of narrative usefulness comes from the VSN weights. Figure 5.10 and Figure 5.11 report the encoder-side and decoder-side variable importance patterns for the Llama-3 hybrid.

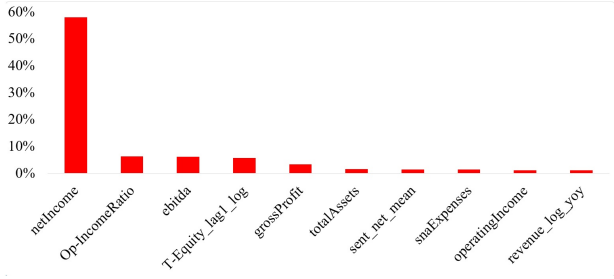


Figure 5.10: Encoder variable importance.

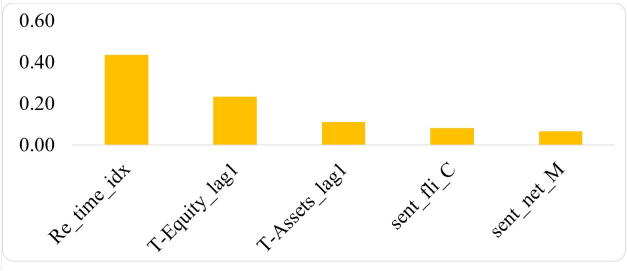


Figure 5.11: Decoder variable importance.

Historical revenue remains the single most important input, which is consistent with standard forecasting intuition. However, the narrative features do not appear as marginal add-ons. The Llama-3-derived net sentiment score and forward-looking statement count receive substantial weights within the encoder and decoder branches, indicating that the model is actively using them in combination with the structured history. This supports the central multimodal claim of the chapter: narrative features are not merely correlated with performance ex post, but are directly used by the model while constructing the forecast.

5.8.3 Temporal Attention and Delayed Narrative Realization

The temporal attention structure also suggests that the hybrid model learns a delayed realization mechanism. Rather than placing all weight on the immediately preceding quarter, attention often concentrates on specific earlier quarters where narrative tone appears to have conveyed durable information about future business change. This is consistent with the idea that managerial language may lead realized accounting performance by more than one quarter, especially when large strategic pivots require time to materialize operationally.

5.9 Discussion and Practical Implications

5.9.1 Structural-Break Evidence: Nvidia During the AI Cycle

The practical value of the multimodal framework becomes especially visible during structural breaks, when trailing financial statements and purely sequence-based patterns may fail to reflect the speed and magnitude of an emerging regime shift. Nvidia provides the clearest example in the present chapter. During the 2023–2024 AI cycle, the pure quantitative TFT remains anchored to the historical pattern and under-predicts the scale of the revenue breakout. The FinBERT hybrid responds in the correct direction, but still underestimates the magnitude of the shift. By contrast, the Llama-3 hybrid shows the strongest response, suggesting that its narrative features capture forward-looking information about infrastructure demand, product momentum, and management conviction that is not yet visible in the lagged structured covariates. While LSTM achieves strong aggregate performance in the previous section, the following analysis focuses on scenarios where purely sequential patterns may be insufficient, and where forward-looking narrative information becomes particularly valuable.

Figure 5.12 reinforces this interpretation by comparing the extracted NLP signals with Nvidia's realized year-over-year revenue growth. The qualitative alignment is economically meaningful: as the AI cycle intensifies, the Llama-3 feature appears to track the acceleration of the narrative regime more flexibly than the FinBERT sentiment score.

One explanation is that FinBERT relies on a relatively discrete sentiment-classification structure, which can saturate once transcript language becomes uniformly optimistic. In contrast, the Llama-3 pipeline produces richer generative features that continue to scale with the intensity, emphasis, and thematic concentration of forward-looking discussion. Therefore, the Nvidia example provides an interpretable illustration of the broader chapter result: the advantage of the Llama-3 hybrid is greatest when the forecasting environment is changing rapidly and when purely historical accounting variables are least sufficient on their own. Even

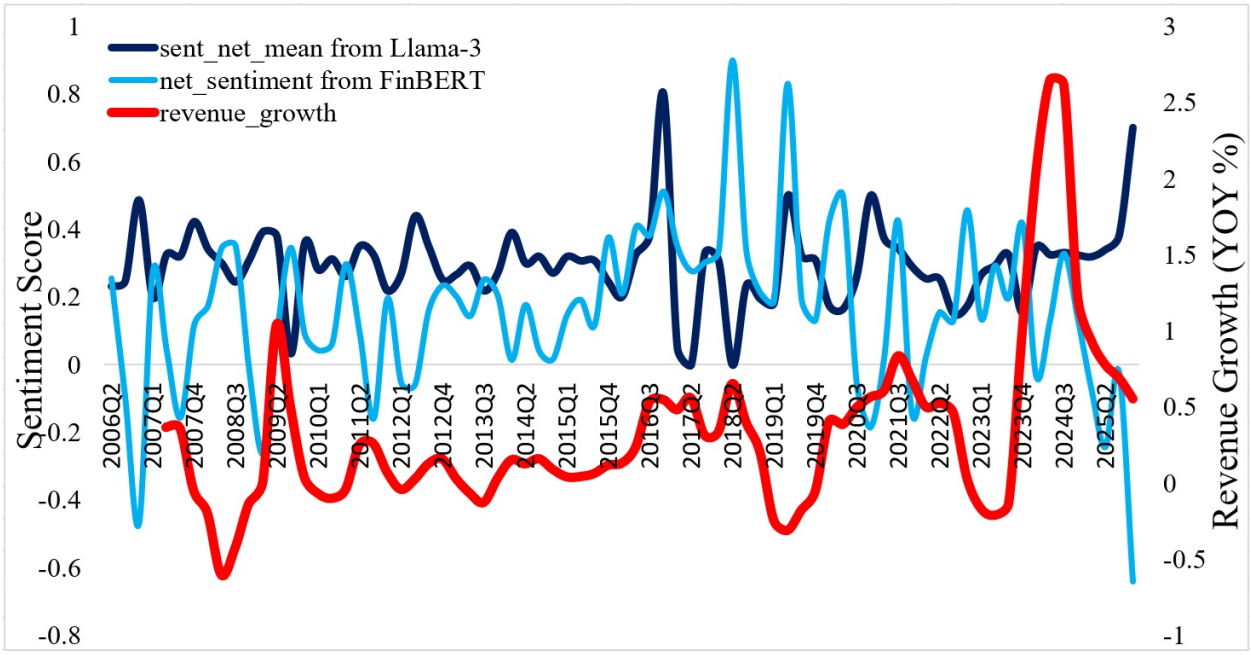


Figure 5.12: Alignment of extracted NLP signals with Nvidia's realized year-over-year revenue growth.

though LSTM captures short-term dynamics effectively, it still relies on historical patterns and does not explicitly incorporate forward-looking information from managerial communication.

5.9.2 Computational Cost Versus Predictive Value

The multimodal framework is more computationally expensive than the structured-only baseline, particularly when the Llama-3 pipeline is used. However, this additional cost is primarily concentrated in the quarterly transcript processing stage rather than in every downstream forecast query.

While LSTM provides a computationally efficient alternative with strong baseline performance, it relies mainly on historical patterns and does not explicitly incorporate forward-looking information. By contrast, the multimodal framework introduces narrative signals derived from earnings calls, which can enhance the model's responsiveness to emerging changes.

In return, the chapter documents a substantial reduction in large long-horizon forecast errors within the TFT-based framework. In institutional settings where avoiding large forecast misses is economically meaningful, this trade-off between computational cost and predictive robustness can be justified.

5.9.3 Applications in Institutional Forecasting

The hybrid framework has several practical uses. First, it can improve long-horizon scenario construction for large technology firms whose revenue paths are sensitive to strategic transitions. Second, the probabilistic outputs of TFT can still be used to produce forecast intervals, allowing the narrative-enhanced model to contribute not only point forecasts but also uncertainty-aware decision support. Third, the model's built-in attribution mechanisms make it more suitable for governance-constrained use than a purely opaque

black-box system. More broadly, the multimodal framework can be interpreted as a systematic way of translating managerial language into structured forecasting inputs. In that sense, it does not replace traditional fundamentals analysis, but extends it by giving the model access to information that human investors have always considered economically important: tone, guidance, conviction, and forward-looking emphasis.

5.10 Chapter Summary

This chapter introduced a multimodal extension of the TFT forecasting framework in order to address the long-horizon limitations identified in Chapter 4. Using the Mega-Cap 5 technology cohort as a focused testbed, it combined structured financial fundamentals with text-derived narrative features extracted from quarterly earnings call transcripts. Two NLP pipelines were evaluated: a finance-domain sentiment baseline based on FinBERT and a richer generative feature-extraction pipeline based on a locally deployed, quantized Llama-3 8B model.

To provide a stronger benchmark, an LSTM model was also evaluated under the same chronological setting. The results show that LSTM achieves strong overall performance, indicating that sequence-based models can effectively capture revenue dynamics in this setting. However, within the TFT-based framework, both multimodal variants consistently improve upon the pure quantitative baseline, with the Llama-3+TFT architecture delivering the largest gains.

These improvements are visible across the full forecasting horizon, suggesting that narrative augmentation can materially stabilize multi-quarter revenue forecasting in high-volatility technology settings. The interpretability analysis further shows that the hybrid model actively uses both structured fundamentals and text-derived features, rather than treating narrative information as a superficial add-on.

Overall, the evidence in this chapter suggests that forward-looking managerial language can provide useful incremental signal precisely where purely historical accounting data become less sufficient. This finding completes the empirical core of the thesis and sets up the final chapter, which synthesizes the main conclusions, limitations, and directions for future research.

Chapter 6

Conclusions and Future Work

6.1 Conclusion

Corporate revenue forecasting is central to valuation, portfolio management, and capital allocation. However, it remains challenging because financial statements are inherently backward-looking, while decision-making often requires forward-looking estimates from the next quarter to a rolling one-year horizon. This challenge becomes more pronounced as the forecast horizon extends, particularly in fast-changing industries.

To address this problem, this thesis first developed a TFT baseline for next-quarter revenue forecasting using a broad panel of 155 continuously listed S&P 500 firms from 1995Q1 to 2025Q2. Under a strict chronological evaluation framework, the TFT achieved strong out-of-sample performance, with a test MAPE of 9.31%, RMSE of 1,973 million USD, and MAE of 1,790 million USD. Further analysis showed that accurate short-horizon forecasting depends not only on past revenue, but also on structured firm characteristics, including sector identity, year-over-year growth, and firm scale variables such as total assets and equity.

The framework was then extended from one-quarter-ahead to four-quarter-ahead forecasting. The results showed clear horizon degradation, with MAPE increasing from 9.31% at Horizon ($t + 1$) to 12.07% at Horizon ($t + 4$). A comparative evaluation with an LSTM baseline under the same chronological setting further confirmed that this deterioration is not specific to a single model, but reflects a broader limitation of purely financial forecasting approaches. The effect was especially pronounced in technology-oriented firms, highlighting the limits of relying solely on lagged financial data in dynamic, non-linear growth environments.

To improve long-horizon performance, this thesis proposed a multimodal TFT framework that integrates structured financial data with earnings-call-derived textual features. Focusing on the Mega-Cap 5 technology firms (Apple, Microsoft, Amazon, Alphabet, and Nvidia), textual signals were extracted from quarterly earnings call transcripts over the period 2007–2025 using both FinBERT and a locally deployed Llama-3 8B model. These features were incorporated into the forecasting pipeline in a leakage-safe manner.

To provide a stronger benchmark, an LSTM model was also evaluated under the same chronological setting. The results show that LSTM achieves strong overall performance on this sample, indicating that sequence-based models can effectively capture revenue dynamics. However, within the TFT-based framework, multimodal integration consistently improves forecasting performance. Across the four-quarter horizon, the pure TFT baseline recorded an RMSE of 46,297, MAE of 44,688, and MAPE of 53.85%. The

FinBERT+TFT hybrid reduced these to 43,895, 42,099, and 48.70%, respectively, while the Llama-3+TFT model further improved performance to 37,769, 35,884, and 43.01%. Horizon-wise results confirmed consistent gains, with the Llama-3 hybrid reducing MAPE from 52.97% to 40.29% at $h = 1$ and from 54.87% to 46.11% at $h = 4$.

In conclusion, this thesis presents a practically deployable multimodal forecasting framework that bridges backward-looking financial fundamentals and forward-looking managerial narratives. The results show that while structured financial data provides a strong and interpretable short-horizon baseline, its predictive power weakens over longer horizons across different model classes. The integration of narrative information can meaningfully improve forecasting performance within a structured temporal framework, particularly in high-volatility and rapidly evolving sectors where forward-looking signals are most valuable.

6.2 Limitations of the Present Study

While the findings of this thesis provide useful evidence for both short-horizon and long-horizon corporate revenue forecasting, several limitations should still be recognized. The results show the value of structured financial data in near-term prediction and the potential of narrative information in improving longer-horizon forecasts. At the same time, the scope of the sample, the design of the text features, and the remaining level of forecasting uncertainty all suggest that the present study is not a complete solution. These limitations are discussed in the following points.

1. The multimodal analysis was limited to the Mega-Cap 5 technology firms: Apple, Microsoft, Amazon, Alphabet, and Nvidia. This focus was appropriate for examining forecasting performance in a high-volatility, narrative-sensitive setting, especially where earnings-call language may contain valuable forward-looking information. However, it also limits the generalizability of the text-based findings. The predictive value of narrative features may differ across sectors with more stable business models, lower disclosure intensity, or less frequent structural change.

2. The textual representation remains relatively simplified. Although FinBERT and Llama-3 provide two useful and contrasting NLP approaches, both ultimately compress rich earnings-call content into a small set of structured features for downstream forecasting. As a result, some important dimensions of managerial communication, such as uncertainty, competitive positioning, strategic redirection, pricing power, and regulatory exposure, may not be fully captured. This means that the text pipeline still reflects only part of the information embedded in corporate narratives.

3. Although the multimodal framework improves long-horizon forecasting performance, substantial uncertainty remains. Even the strongest model, Llama-3+TFT, still records a MAPE of 43.01% on the Mega-Cap 5 cohort over the four-quarter horizon. This indicates that long-horizon revenue forecasting remains inherently difficult in highly dynamic environments. While the hybrid models improve information quality and forecasting stability, they cannot fully overcome the effects of structural breaks, macroeconomic shocks, firm-specific events, or rapid changes in competitive conditions.

4. The use of large language models also introduces concerns such as bias, hallucination, and data leakage. In this study, the LLM is used only for feature extraction, and several steps are taken to reduce

these risks, including numerical-only outputs, fixed prompts, controlled decoding, and strict chronological alignment. However, a full treatment of LLM-related risks is beyond the scope of this thesis and should be explored in future work.

5. The better forecasting performance of the multimodal framework also comes with greater model complexity. Compared with the structured-only baseline, the hybrid design adds text preprocessing, LLM-based feature extraction, and multimodal integration, which increase computational cost and implementation effort. In particular, the Llama-3 pipeline is more resource-intensive than the FinBERT-based alternative. A more systematic analysis of efficiency trade-offs, such as runtime and inference cost, is left for future work.

These limitations suggest that the findings of this thesis should be interpreted as evidence of meaningful progress rather than a complete solution. The proposed framework improves long-horizon forecasting by integrating structured financial data with narrative information, but there remains considerable room for broader validation, richer text modeling, and further methodological refinement.

6.3 Future Work and Research Directions

The findings of this thesis highlight both the value of structured financial forecasting and the added benefit of incorporating forward-looking narrative information. At the same time, they also point to several areas where the present framework can be extended, tested more broadly, and refined further. Building on these results, several directions for future research follow naturally.

1. The multimodal forecasting framework can be extended to a broader set of firms, such as the full S&P 500 or smaller-cap companies. This would help evaluate whether earnings-call-derived textual features provide similar forecasting value beyond the Mega-Cap 5 technology firms and across sectors with different business characteristics.

2. Future research can expand the textual information set by incorporating additional corporate disclosure sources, such as Management's Discussion and Analysis (MD&A) sections, shareholder letters, and investor presentations. These materials may contain complementary forward-looking signals that are not fully reflected in earnings call transcripts alone.

3. Future work can develop richer narrative feature representations. Beyond general sentiment, it would be useful to model more specific dimensions of managerial communication, such as uncertainty, competitive pressure, supply-chain conditions, pricing power, strategic confidence, and capital allocation priorities. These features may provide a more nuanced view of the information embedded in corporate language.

4. Future forecasting models can incorporate macroeconomic and market-level variables, such as interest rates, inflation, industry conditions, and broader business-cycle indicators. These factors may help improve predictive performance, especially for longer horizons where firm-level historical data becomes less informative on its own.

In summary, this thesis should be viewed as a starting point rather than a final solution. It shows that integrating structured financial data with managerial narratives can improve long-horizon corporate revenue forecasting. At the same time, it opens several paths for further development, including broader validation, richer text representations, and more comprehensive forecasting frameworks.

References

[1] G. E. P. Box, G. M. Jenkins, G. C. Reinsel, and G. M. Ljung, *Time Series Analysis: Forecasting and Control*, 5th ed. Hoboken, NJ: John Wiley & Sons, 2015.

[2] C. A. Sims, "Macroeconomics and reality," *Econometrica*, vol. 48, no. 1, pp. 1–48, 1980. DOI: 10.2307/1912017.

[3] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997. DOI: 10.1162/neco.1997.9.8.1735.

[4] B. Lim, S. O. Arik, N. Loeff, and T. Pfister, "Temporal fusion transformers for interpretable multi-horizon time series forecasting," *International Journal of Forecasting*, vol. 37, no. 4, pp. 1748–1764, 2021. DOI: 10.1016/j.ijforecast.2021.03.012.

[5] D. Shu, Y. Liu, H. Zhang, and M. Du, *Fincall-surprise: A large scale multi-modal benchmark for earning surprise prediction*, 2025. arXiv: 2510.03965 [cs.MM].

[6] D. P. Pagach and R. S. Warr, "Analysts versus time-series forecasts of quarterly earnings: A maintained hypothesis revisited," *Advances in Accounting*, vol. 51, p. 100 497, 2020. DOI: 10.1016/j.adiac.2020.100497.

[7] S. J. Choi, X. Cui, and J. Zhao, "Boosting over deep learning for earnings," in *AAAI Workshop on Knowledge Discovery from Data (KDF)*, Available online: aaai-kdf.github.io, 2021.

[8] C. Bergmeir, R. J. Hyndman, and B. Koo, "A note on the validity of cross-validation for evaluating autoregressive time series prediction," *Computational Statistics & Data Analysis*, vol. 120, pp. 70–83, 2018. DOI: 10.1016/j.csda.2017.11.003.

[9] M. A. Lones, "Tutorial: Avoiding common machine learning pitfalls," *Patterns*, 2024. DOI: 10.1016/j.patter.2024.100188.

[10] D. Araci, *Finbert: Financial sentiment analysis with pre-trained language models*, 2019. arXiv: 1908.10063 [cs.CL].

[11] T. Dettmers, A. Pagnoni, A. Holtzman, and L. Zettlemoyer, *Qlora: Efficient finetuning of quantized llms*, 2023. arXiv: 2305.14314 [cs.LG].

[12] H. Ni et al., "Harnessing earnings reports for stock predictions: A qlora-enhanced llm approach," in *2024 6th International Conference on Data-driven Optimization of Complex Systems (DOCS)*, IEEE, 2024. DOI: 10.1109/DOCS63458.2024.10704454.

[13] R. J. Hyndman and G. Athanasopoulos, *Forecasting: Principles and Practice*, 3rd ed. OTexts, 2021, Online textbook.

[14] T. R. Cook, S. Kazinnik, A. L. Hansen, and P. McAdam, "Evaluating local language models: An application to financial earnings calls," Federal Reserve Bank of Kansas City, Research Working Paper RWP 23-12, Nov. 2023, Electronic copy available at SSRN: 4627143. DOI: 10.18651/RWP2023-12.

[15] P. Montero-Manso and R. J. Hyndman, "Principles and algorithms for forecasting groups of time series: Locality and globality," *International Journal of Forecasting*, vol. 37, no. 4, pp. 1632–1653, 2021. DOI: 10.1016/j.ijforecast.2021.03.004.

[16] H. Hewamalage, C. Bergmeir, and K. Bandara, "Global models for time series forecasting: A simulation study," *Pattern Recognition*, vol. 124, p. 108 441, 2022. DOI: 10.1016/j.patcog.2021.108441.

[17] Y. Bengio, P. Simard, and P. Frasconi, "Learning long-term dependencies with gradient descent is difficult," *IEEE Transactions on Neural Networks*, vol. 5, no. 2, pp. 157–166, 1994. DOI: 10.1109/72.279181.

[18] K. Cho et al., "Learning phrase representations using RNN encoder–decoder for statistical machine translation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Association for Computational Linguistics, 2014, pp. 1724–1734. DOI: 10.3115/v1/D14-1179.

[19] J. Chung, Ç. Gülçehre, K. Cho, and Y. Bengio, *Empirical evaluation of gated recurrent neural networks on sequence modeling*, 2014. arXiv: 1412.3555 [cs.NE].

[20] P. J. Huber, "Robust estimation of a location parameter," *Annals of Mathematical Statistics*, vol. 35, no. 1, pp. 73–101, 1964. DOI: 10.1214/aoms/1177703732.

[21] A. Vaswani et al., "Attention is all you need," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

[22] R. Koenker and G. Bassett, "Regression quantiles," *Econometrica*, vol. 46, no. 1, pp. 33–50, 1978. DOI: 10.2307/1913643.

[23] T. Loughran and B. McDonald, "When is a liability not a liability? textual analysis, dictionaries, and 10-ks," *The Journal of Finance*, vol. 66, no. 1, pp. 35–65, 2011. DOI: 10.1111/j.1540-6261.2010.01625.x.

[24] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of NAACL-HLT*, Association for Computational Linguistics, 2019, pp. 4171–4186. DOI: 10.18653/v1/N19-1423.

[25] Y. Nie et al., *A survey of large language models for financial applications: Progress, prospects and challenges*, 2024. arXiv: 2406.11903 [q-fin.GN].

[26] A. Grattafiori, A. Dubey, A. Jauhri, et al., *The llama 3 herd of models*, 2024. arXiv: 2407.21783 [cs.AI].

[27] R. S. Tsay, *Analysis of Financial Time Series*, 2nd ed. Hoboken, NJ: John Wiley & Sons, 2005, ISBN: 978-0471690740.

[28] J. Y. Campbell, M. Lettau, B. G. Malkiel, and Y. Xu, "Have individual stocks become more volatile? an empirical exploration of idiosyncratic risk," *The Journal of Finance*, vol. 56, no. 1, pp. 1–43, 2001. DOI: 10.1111/0022-1082.00318.

[29] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001. DOI: 10.1023/A:1010933404324.

[30] T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '16)*, San Francisco, CA, USA, Aug. 2016, pp. 785–794. DOI: 10.1145/2939672.2939785.

[31] A. Amel-Zadeh, J.-P. Calliess, D. Kaiser, and S. Roberts, *Machine learning-based financial statement analysis*, SSRN 3520684, Working paper (SSRN), Jan. 2020.

[32] G. Thieren, "Comparing earnings prediction models to analysts' consensus forecasts: A machine learning approach," Academic year 2022–2023, M.S. thesis, Louvain School of Management, UCLouvain, 2023.

[33] C. M. Bishop, *Pattern Recognition and Machine Learning*. New York, NY, USA: Springer, 2006, ISBN: 978-0-387-31073-2.

[34] K.-j. Kim, "Financial time series forecasting using support vector machines," *Neurocomputing*, vol. 55, pp. 307–319, 2003. DOI: 10.1016/S0925-2312(03)00372-2.

[35] T. Fischer and C. Krauss, "Deep learning with long short-term memory networks for financial market predictions," *European Journal of Operational Research*, vol. 270, no. 2, pp. 654–669, 2018. DOI: 10.1016/j.ejor.2017.11.054.

[36] W. Bao, J. Yue, and Y. Rao, "A deep learning framework for financial time series using stacked autoencoders and long-short term memory," *PLOS ONE*, vol. 12, no. 7, e0180944, 2017. DOI: 10.1371/journal.pone.0180944.

[37] K. Greff, R. K. Srivastava, J. Koutník, B. R. Steunebrink, and J. Schmidhuber, "Lstm: A search space odyssey," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 10, pp. 2222–2232, 2017. DOI: 10.1109/TNNLS.2016.2582924.

[38] A. A. Ismail, M. Gunady, H. Corrada Bravo, and S. Feizi, "Benchmarking deep learning interpretability in time series predictions," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2020. arXiv: 2010.13924 [cs.LG].

[39] E. Hoseinzadeh and S. Haratizadeh, "Cnnpred: Cnn-based stock market prediction using a diverse set of variables," *Expert Systems with Applications*, vol. 129, pp. 273–285, 2019. DOI: 10.1016/j.eswa.2019.03.029.

[40] R. Reisenhofer, X. Bayer, and N. Hautsch, *Harnet: A convolutional neural network for realized volatility forecasting*, 2022. DOI: 10.48550/arXiv.2205.07719. arXiv: 2205.07719 [econ.EM].

[41] J. Frank, "Forecasting realized volatility in turbulent times using temporal fusion transformers," Friedrich-Alexander University Erlangen-Nürnberg (FAU), Institute for Economics, FAU Discussion Papers in Economics 03/2023, 2023. [Online]. Available: https://www.iwf.rw.fau.de/files/2023/02/03_2023.pdf.

[42] L. Petrosino, L. Bacco, G. Salvati, M. Merone, and M. Papi, "A GARCH-temporal fusion transformer model for the volatility prediction of exchange traded funds," *Neural Computing and Applications*, vol. 37, no. 26, pp. 21 435–21 458, 2025. DOI: 10.1007/s00521-025-11468-z.

[43] D. Zhang, "A temporal fusion transformer network for enhanced international currency exchange prediction," *The Computer Journal*, vol. 69, no. 2, pp. 259–271, 2026, Published online 27 Sep 2025. DOI: 10.1093/comjnl/bxaf113.

[44] J. Laborda et al., "Multi-country and multi-horizon GDP forecasting using temporal fusion transformers," *Mathematics*, vol. 11, no. 12, p. 2625, 2023. DOI: 10.3390/math11122625.

[45] A. Peik, M. A. Zare Chahooki, A. Milani Fard, and M. Agha Sarram, *Adaptive temporal fusion transformers for cryptocurrency price prediction*, 2025. DOI: 10.48550/arXiv.2509.10542. arXiv: 2509.10542 [q-fin.ST].

[46] G. N. Dong, "Can AI replace stock analysts? evidence from deep learning financial statements," Working paper, Carroll School of Management, Boston College, 2024.

[47] S. Jenčová, P. Vašaničová, M. Košíková, and M. Miškufová, "A time series approach to forecasting financial indicators in the wholesale and retail trade," *World*, vol. 6, no. 1, Jan. 2025. DOI: 10.3390/world6010005.

[48] S. Joshi, B. L. Mahanthi, P. G, K. S. Pokkuluri, S. S. Ninawe, and R. Sahu, "Integrating LSTM and CNN for stock market prediction: A dynamic machine learning approach," *Journal of Artificial Intelligence and Technology*, vol. 5, pp. 168–179, 2025. DOI: 10.37965/jait.2025.0652.

[49] N. Jegadeesh and J. Livnat, "Revenue surprises and stock returns," *Journal of Accounting and Economics*, vol. 41, no. 1–2, pp. 147–171, 2006. DOI: 10.1016/j.jacceco.2005.09.003.

[50] A. Peik, M. A. Z. Chahooki, A. M. Fard, and M. A. Sarram, "Leveraging time series categorization and temporal fusion transformers to improve cryptocurrency price forecasting," *arXiv preprint arXiv:2412.14529*, 2024.

[51] D. Zhang, "A temporal fusion transformer network for enhanced international currency exchange prediction," *The Computer Journal*, pp. 1–13, 2025. DOI: 10.1093/comjnl/bxaf113.

[52] R. Ho and K. Hung, "Ceemd-based multivariate financial time series forecasting using a temporal fusion transformer," in *Proc. 2024 IEEE 14th Symposium on Computer Applications & Industrial Electronics (ISCAIE)*, 2024. DOI: 10.1109/ISCAIE61308.2024.10576340.

[53] X. Hu, "Stock price prediction based on temporal fusion transformer," in *Proc. 3rd Int. Conf. on Machine Learning, Big Data and Business Intelligence (MLBDBI)*, 2021, pp. 60–66. DOI: 10.1109/MLBDBI54094.2021.00019.

[54] G. E. P. Box, G. M. Jenkins, G. C. Reinsel, and G. M. Ljung, *Time Series Analysis: Forecasting and Control*, 5th ed. Hoboken, NJ, USA: John Wiley & Sons, 2016, First ed. 1970, ISBN: 978-1-118-67502-1.

[55] C. J. Hutto and E. Gilbert, "Vader: A parsimonious rule-based model for sentiment analysis of social media text," in *Proceedings of the International AAAI Conference on Web and Social Media (ICWSM)*, vol. 8, 2014, pp. 216–225. DOI: 10.1609/icwsm.v8i1.14550.

[56] N. Gyawali, D. Caragea, A. Vasenkov, and C. Caragea, *Evaluating large language models for stance detection on financial targets from sec filing reports and earnings call transcripts*, arXiv:2510.23464v1, 2025. DOI: 10.48550/arXiv.2510.23464. arXiv: 2510.23464 [cs.CL].

[57] S. K. Nath, Y. Zhang, and J. V. Li, "Earnings call scripts generation with large language models using few-shot learning: Prompt engineering and fine-tuning methods," *Applied AI Letters*, vol. 6, e110, 2025. DOI: 10.1002/ail2.110.

[58] G. Matera, *Corporate earnings calls and analyst beliefs*, arXiv:2511.15214v2, 2025. DOI: 10.48550/arXiv.2511.15214. arXiv: 2511.15214.

[59] A. H. Huang, H. Wang, and Y. Yang, "FinBERT: A large language model for extracting information from financial text," *Contemporary Accounting Research*, vol. 40, no. 2, pp. 806–841, 2023. DOI: 10.1111/1911-3846.12832.

[60] Z. Chen, S. Gössi, and coauthors, "FinBERT-FOMC: Fine-tuned FinBERT model with sentiment," in *Proceedings of the 4th ACM International Conference on AI in Finance (ICAIF '23)*, Specialized FinBERT for FOMC minutes, Brooklyn, NY, USA: ACM, 2023, pp. 359–366.

[61] S. Bansal et al., *An analysis of different sentiment analysis models on financial text*, Comparative study of FinBERT, DistilBERT, VADER, and Logistic Regression on multiple financial sentiment datasets, 2025.

[62] D. K. Nasiopoulos, K. I. Roumeliotis, D. P. Sakas, K. Toudas, and P. Reklitis, "Financial sentiment analysis and classification: A comparative study of fine-tuned deep learning models," *International Journal of Financial Studies*, vol. 13, no. 2, p. 75, 2025. DOI: 10.3390/ijfs13020075.

[63] Y. Huang et al., "A FinBERT framework for sentiment analysis of chinese financial news," in *2024 4th International Symposium on Computer Technology and Information Science (ISCTIS)*, Xi'an, China: IEEE, 2024, pp. 796–799. DOI: 10.1109/ISCTIS63324.2024.10699096.

[64] Ayush, "Hindi finbert: A pre-trained language model for financial text classification," Supervisor: Dr. Rejwanul Haque, MSc Research Project, National College of Ireland, School of Computing, 2024.

[65] S. Halder, "FinBERT-LSTM: Deep learning based stock price prediction using news sentiment analysis," *arXiv preprint*, vol. arXiv:2211.07392, 2022. DOI: 10.48550/arXiv.2211.07392. [Online]. Available: https://arxiv.org/abs/2211.07392.

[66] M. F. B. Hossain, L. Z. Lamia, M. M. Rahman, and M. M. Khan, "FinBERT-BiLSTM: A deep learning model for predicting volatile cryptocurrency market prices using market sentiment dynamics," *arXiv preprint arXiv:2411.12748*, 2024.

[67] L. Ruan and Y. Jiang, "Stock price prediction using FinBERT-enhanced sentiment analysis and multimodal features with differential privacy," *Mathematics*, vol. 13, no. 17, p. 2747, 2025. DOI: 10.3390/math13172747.

[68] H. Taheripour, M. Rezaee, M. M. Parhizgar, and A. Pirouzmand, "A novel approach to portfolio construction: An application of FinBERT sentiment analysis and credibilistic CVaR criterion," *IEEE Access*, vol. 13, pp. 76 775–76 795, 2025, Please verify exact page range/DOI against the IEEE Access version.

[69] B. Jin, Y. Yang, et al., "Temporal data meets large language models: Explainable financial time series forecasting," *arXiv preprint*, 2023. arXiv: 2306.11025 [cs.LG].

[70] Anonymous, "Uni-finllm: A unified multi-modal large language model for financial forecasting and risk assessment," *arXiv preprint arXiv:2601.02677*, 2026. [Online]. Available: https://arxiv.org/abs/2601.02677.

[71] R. Thota, S. M. Potluri, B. Kaki, H. M. Abbas, and G. Raghu, "Financial bidirectional encoder representations from transformers with temporal fusion transformer for predicting financial market trends," in *2025 International Conference on Intelligent Computing and Knowledge Extraction (ICICKE)*, IEEE, 2025. DOI: 10.1109/ICICKE65317.2025.11136233.

[72] X. Jin and S.-L. Lin, "An early prediction model on systemic risk under global risk: Using finbert and temporal fusion transformer to multimodal data fusion framework," *The North American Journal of Economics and Finance*, vol. 76, p. 102 361, 2025. DOI: 10.1016/j.najef.2025.102361.

[73] A. Álvarez Castro and J. Ordieres-Meré, *Multimodal proposal for an ai-based tool to increase cross-assessment of messages*, arXiv:2509.03529v1, 2025. DOI: 10.48550/arXiv.2509.03529. arXiv: 2509.03529 [cs.CL].

[74] X. Chen et al., *The sound of risk: A multimodal physics-informed acoustic model for forecasting market volatility and enhancing market interpretability*, arXiv:2508.18653v1, 2025. DOI: 10.48550/arXiv.2508.18653. arXiv: 2508.18653 [cs.LG].

[75] W. Kaikaus, "Multimodal emotion recognition and speaker identification in financial conversations," Ph.D. dissertation, University of Illinois Urbana-Champaign, 2025.

[76] I. K. Friday, S. P. Pati, and D. Mishra, "A multi-modal approach using a hybrid vision transformer and temporal fusion transformer model for stock price movement classification," *IEEE Access*, vol. 13, 2025. DOI: 10.1109/ACCESS.2025.3589063.

[77] S. Shahsafi and F. Naderkhani, *Integrating image-based time series features with the temporal fusion transformer for stock price prediction*, Preprint, Concordia University, 2025.

[78] S. Lødøen and M. Myklebust, "Predicting bitcoin volatility using temporal fusion transformer," Master's thesis, Norwegian University of Science and Technology (NTNU), Trondheim, Norway, 2024.

[79] J. Jang, T. Kim, and S.-H. Park, "Stock index forecasting using an explainable TAFT model with online data-driven social sentiment index," in *Proceedings of the 5th ACM International Conference on AI in Finance (ICAIF '24)*, Association for Computing Machinery, 2024, pp. 787–794. DOI: 10.1145/3677052.3698618.

[80] Financial Modeling Prep, *Financial modeling prep api*, https://site.financialmodelingprep.com/, REST API for financial statements and market data; quarterly fundamentals used in this study. Accessed: 2025-09-25., 2025.

[81] S. Seabold and J. Perktold, "Statsmodels: Econometric and statistical modeling with python," in *Proceedings of the 9th Python in Science Conference (SciPy 2010)*, Software: statsmodels v0.14.5, 2010, pp. 92–96. [Online]. Available: https://www.statsmodels.org/.

[82] F. A. Gers, J. Schmidhuber, and F. Cummins, "Learning to forget: Continual prediction with LSTM," *Neural Computation*, vol. 12, no. 10, pp. 2451–2471, 2000. DOI: 10.1162/089976600300015015.