University of Amsterdam

Masters Thesis

Forecasting, Uncertainty, and Excitation: A Unified Framework for Cryptocurrency Trading

Author:
Xiaoxuan Zhang

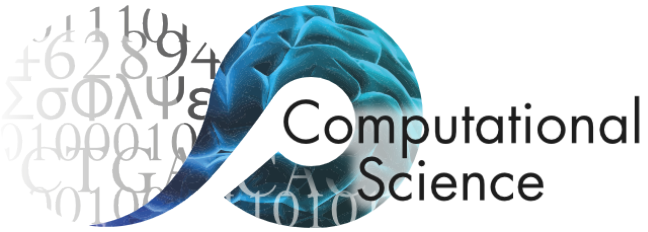
Examiner:
A.S.Z. (Adam) Belloum
Supervisor:
Zhiheng Yang
Assessor:
B.D. (Drona) Kandhai

A thesis submitted in partial fulfilment of the requirements for the degree of Master of Science in Computational Science

in the

Computational Science Lab
Informatics Institute

April 2026

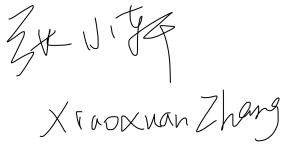

Computational
Science

Declaration of Authorship

I, Xiaoxuan Zhang, declare that this thesis, entitled 'Forecasting, Uncertainty, and Excitation: A Unified Framework for Cryptocurrency Trading' and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at the University of Amsterdam.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed:



Date: 09 April 2026

"What I cannot create, I do not understand."

Richard P. Feynman

UNIVERSITY OF AMSTERDAM

Abstract

Faculty of Science
Informatics Institute

Master of Science in Computational Science

Forecasting, Uncertainty, and Excitation: A Unified Framework for Cryptocurrency Trading

by Xiaoxuan ZHANG

Cryptocurrency quantitative trading strategies commonly rely on model-based predictions as alpha proxies, where forecasting outputs are directly translated into trading decisions. Early approaches primarily employ classical econometric models such as ARIMA and GARCH, while more recent developments have introduced deep learning architectures including LSTM-based models. With the emergence of time-series foundation models (TSFMs), such as Chronos2, forecasting has shifted toward large-scale probabilistic modeling capable of producing both return predictions and uncertainty estimates. However, applying these general-purpose models to cryptocurrency markets remains challenging due to heavy-tailed return distributions, extreme volatility, and rapidly changing regimes, which often lead to miscalibrated uncertainty and suboptimal trading signals.

In this work, we focus on Chronos2 as the core forecasting backbone and propose an improved fine-tuning approach, termed Chronos2-QuExTime, which incorporates a volatility-aware quantile loss to better capture predictive uncertainty and tail behavior in cryptocurrency markets. This modification enhances the quality of probabilistic forecasts, resulting in more informative alpha proxy signals derived from model outputs.

We then construct trading signals by scaling predicted returns with model-implied uncertainty, forming a unified risk-adjusted signal framework.

While this uncertainty-based signal already leads to improved trading performance, it primarily reflects statistical predictive risk. In highly endogenous markets such as cryptocurrencies, additional risk arises from self-exciting dynamics that are not explicitly captured by the forecasting model.

To account for this, we further introduce a Hawkes-process-based excitation layer to capture endogenous market dynamics that are not explicitly modeled by deep learning predictors. The estimated excitation intensity is integrated as an additional risk component, enabling the trading signal to adapt to self-exciting market conditions and improving its robustness under regime changes.

Empirical results on representative cryptocurrency assets demonstrate that the proposed TSFM adaptation and the Hawkes-based enhancement consistently improve risk-adjusted trading performance, while also revealing regime-dependent limitations in highly volatile environments. All data and code for the full pipeline, including data collection, model training, and trading evaluation, are publicly available to ensure reproducibility.

Acknowledgements

I would like to sincerely thank my supervisor, Zhiheng Yang, for his guidance, encouragement, and insightful feedback throughout this research. His support has been crucial in developing the ideas presented in this thesis.

I am also grateful to my examiner, Dr. A.S.Z. (Adam) Belloum, for his careful evaluation, and to the faculty and lecturers of the Computational Science programme for providing a solid academic foundation.

I would also like to thank my second reader, Prof. dr. B.D. (Drona) Kandhai, for his valuable perspectives on financial risk and practical aspects of the work.

I would especially like to thank my wife, Menghan Zhang, for her constant support, patience, and understanding. Her encouragement has been invaluable during both the challenging and rewarding moments of this journey.

Finally, I would like to acknowledge all those who have contributed, directly or indirectly, to this work.

Contents

List of Figures

List of Tables

Abbreviations

CSL	Computational Science Lab
UvA	Universiteit van Amsterdam
TSFM	Time-Series Foundation Model
DL	Deep Learning
RNN	Recurrent Neural Network
LSTM	Long Short-Term Memory
ARIMA	AutoRegressive Integrated Moving Average
GARCH	Generalized AutoRegressive Conditional Heteroskedasticity
CRPS	Continuous Ranked Probability Score
MAE	Mean Absolute Error
MSE	Mean Squared Error
RMSE	Root Mean Squared Error
PICP	Prediction Interval Coverage Probability
IC	Information Coefficient
Rank IC	Rank Information Coefficient
PnL	Profit and Loss
CumRet	Cumulative Return
MaxDD	Maximum Drawdown
FT	Fine-Tuning
LoRA	Low-Rank Adaptation
BTC	Bitcoin
DOGE	Dogecoin
API	Application Programming Interface
MLE	Maximum Likelihood Estimation
QuExTime	Quantile Extreme Time-aware Loss (proposed)

Chapter 1

Introduction

Cryptocurrency markets have become an important testbed for modern quantitative trading systems due to their extreme volatility, rapid regime shifts, and strong speculative dynamics. Reliable trading decisions in such environments require not only accurate return forecasting but also robust uncertainty estimation and risk-aware signal construction.

In quantitative trading practice, model-based forecasts are often used as alpha proxies, in the sense that predictive estimates of future returns serve as signals for portfolio allocation and trading decisions. This paradigm is consistent with classical portfolio theory, where investment decisions are driven by expected return and risk trade-offs [1, 2].

To operationalize such decision-making frameworks in practice, a wide range of time-series models have been developed to estimate future returns and associated risks from historical data.

Classical econometric approaches such as ARIMA and GARCH provide structured modeling of return dynamics and volatility, forming a standard baseline in financial time-series analysis [3–5]. More recent developments have introduced machine learning and deep learning models to improve predictive performance in financial markets [6, 7]. Despite these advances, a common challenge remains: how to convert model outputs into effective risk-aware trading signals that can generalize across different market regimes.

Probabilistic forecasting methods based on quantile regression [8] and proper scoring rules [9] have gained increasing attention because they provide uncertainty estimates that are essential for financial decision making. Such probabilistic outputs offer a natural mechanism for constructing risk-adjusted trading signals, where expected returns are scaled by predictive uncertainty.

Recent advances in *Time-Series Foundation Models* (TSFMs) extend this paradigm by learning general temporal representations from large-scale datasets. Transformer-based models such as Chronos treat time-series forecasting as a sequence modeling problem and produce quantile-based probabilistic forecasts [10].

However, applying such models to cryptocurrency markets introduces new challenges. Crypto price series exhibit rapidly changing market regimes that differ substantially from many datasets used during large-scale pretraining. Moreover, trading decisions depend not only on predictive uncertainty derived from forecasting models but also on structural market dynamics that may amplify risk.

This work investigates two key challenges that arise when applying modern probabilistic forecasting models to cryptocurrency trading systems.

1.1 Motivation and Problem Formulation

This section presents the key challenges that arise when applying probabilistic forecasting models to cryptocurrency trading systems, and formulates them as research questions guiding this work.

1.1.1 Limitations of Quantile Forecasting in Cryptocurrency Markets

Probabilistic forecasting models naturally provide prediction intervals that can be interpreted as measures of predictive uncertainty and are therefore highly relevant for financial trading.

Despite recent progress in time-series foundation models (TSFMs), applying such models to cryptocurrency markets remains challenging. To illustrate this issue, we present representative forecasting results from two model classes.

Figures 1.1 and 1.2 show the predicted median trajectories and quantile intervals over a representative test period.

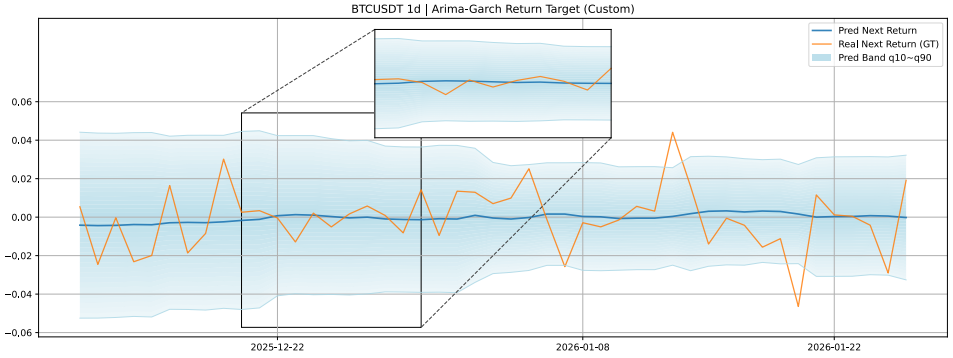


FIGURE 1.1: Quantile forecasts for BTC using ARIMA–GARCH.

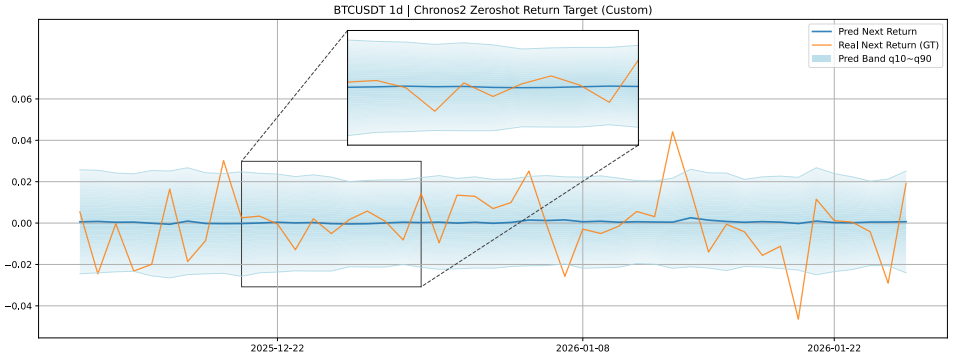


FIGURE 1.2: Zero-shot quantile forecasts for BTC using Chronos2.

From these figures, several limitations in uncertainty estimation can be directly observed. In relatively stable market regimes, the predicted quantile bands (e.g., 10%–90%) tend to be overly conservative, resulting in excessively wide intervals. Such wide prediction bands indicate that the models fail to provide sharp uncertainty estimates even when price dynamics are relatively smooth.

Moreover, the quantile bands often remain broad across entire trend segments, suggesting that the models do not adapt their uncertainty estimates to changing market conditions. This leads to overly diffuse predictive distributions, where uncertainty is overestimated and becomes less informative for downstream decision-making.

TABLE 1.1: Forecasting performance comparison on BTC (test period).

Model	RMSE	CRPS	Pinball$_{q90}$	Sign Acc.	Rank IC
ARIMA–GARCH	0.0162	0.00915	0.00317	0.340	-0.167
Chronos2 (Zero-shot)	0.0160	0.00832	0.00305	0.426	-0.175

While the overall error metrics (e.g., RMSE and CRPS) appear comparable, both models exhibit weak performance in direction-aware metrics such as sign accuracy and rank correlation, which are critical for trading decisions.

This leads to our first research question:

RQ1: *How can quantile-based time-series foundation models be adapted to better capture predictive uncertainty and volatility dynamics in cryptocurrency markets?*

1.1.2 Missing Endogenous Risk Modeling in Trading Signals

Forecasting models typically produce two quantities relevant for trading decisions: expected returns and predictive uncertainty. These outputs can be combined to construct risk-adjusted trading signals in which expected returns are normalized by model-derived uncertainty.

However, predictive uncertainty alone may not fully characterize market risk in cryptocurrency markets. A significant component of risk arises from endogenous feedback mechanisms such as liquidation cascades, leverage-driven trading, and algorithmic responses that amplify price movements.

Such dynamics are often described as *self-exciting processes*, where the occurrence of events increases the likelihood of future events. These mechanisms are not explicitly modeled by most forecasting architectures, whether econometric models or deep learning forecasting systems.

This motivates our second research question:

RQ2: *How can endogenous market excitation be incorporated into model-agnostic trading signals to improve risk-aware decision making in cryptocurrency markets?*

1.2 Contributions

This paper proposes a unified forecasting and trading framework for cryptocurrency markets, with a focus on uncertainty-aware modeling and risk-adjusted signal construction. Our main contributions are summarized as follows:

- **Volatility-aware loss design for quantile-based forecasting under different market regimes.** We propose a tail-sensitive loss formulation for quantile-based time-series foundation models, aimed at improving uncertainty characterization in cryptocurrency markets. Empirical results show that the proposed loss

consistently improves tail-risk calibration and directional signal quality in relatively stable markets (e.g., BTC), while preserving comparable performance on traditional error metrics such as MAE and RMSE. At the same time, experiments on highly volatile assets (e.g., DOGE) reveal a regime-dependent limitation, where excessive emphasis on tail penalties may degrade uncertainty estimation. This highlights the importance of adapting loss design to market characteristics.

- **A unified framework for transforming probabilistic forecasts into trading signals.** We introduce a model-agnostic signal construction pipeline that maps heterogeneous forecasting outputs—ranging from classical econometric models to deep learning foundation models—into a consistent risk-adjusted decision variable. By explicitly separating predicted return and model-native uncertainty (e.g., volatility or quantile interval), the framework provides a general interface for comparing and deploying different forecasting models within a unified trading system.

- **Endogenous market excitation as a complementary risk signal for trading.** We incorporate a Hawkes-process-based excitation layer to model self-exciting dynamics in cryptocurrency markets. The estimated Hawkes intensity is used as an additional risk scaling factor that augments model-native uncertainty. Empirical trading results show that this endogenous risk component consistently improves performance across different forecasting backbones, demonstrating that market excitation provides complementary information beyond standard predictive uncertainty.

Chapter 2

Foundations of Forecasting and Risk Modeling

This section provides background on forecasting models for financial time series and their role in quantitative trading systems. We briefly review the evolution of forecasting approaches from classical econometric models to modern deep learning and foundation models, and discuss how predictive outputs are typically transformed into risk-aware trading signals.

2.1 Evolution of Forecasting Models in Financial Time Series

Classical econometric models provide a long-standing foundation for financial time-series forecasting. ARIMA models capture temporal dependencies in asset returns through autoregressive and moving-average structures [3, 11], while ARCH and GARCH models describe time-varying volatility through autoregressive dynamics of conditional variance [4, 5]. Such ARIMA–GARCH frameworks have been widely applied to model cryptocurrency returns and volatility dynamics.

However, the strong assumptions underlying parametric econometric models often limit their effectiveness in highly nonlinear markets. Empirical studies have shown that cryptocurrency price series exhibit heavy-tailed distributions, volatility clustering, and nonlinear dependencies that challenge traditional model assumptions.

To address these limitations, machine learning and deep learning models have increasingly been adopted for financial forecasting. Recurrent neural networks and LSTM

architectures have demonstrated improved predictive performance in several financial prediction tasks [6]. Nevertheless, most early deep learning approaches focused on point prediction and optimized models using mean squared error.

Because point forecasts do not capture predictive uncertainty, recent research has shifted toward probabilistic forecasting methods that estimate conditional distributions rather than single-point predictions. In this context, quantile regression and related scoring rules have become widely used tools for uncertainty-aware forecasting.

More recently, time-series foundation models have emerged as a new paradigm for probabilistic forecasting. Transformer-based models such as Chronos treat time-series forecasting as a sequence modeling problem and generate quantile-based forecasts that approximate the conditional distribution of future observations.

2.2 From Forecasting to Risk-Aware Trading Signals

While forecasting models estimate future returns or return distributions, trading systems must convert these predictions into actionable decisions. In financial theory, investment decisions are typically formulated as a trade-off between expected return and risk. Classical frameworks such as mean-variance optimization and the Sharpe ratio explicitly combine these two quantities in portfolio allocation.

Forecasting models naturally provide inputs for such decision rules. Econometric volatility models produce conditional variance estimates, while probabilistic forecasting models generate prediction intervals that can serve as measures of predictive uncertainty. These quantities can be combined with expected return predictions to form normalized trading signals.

However, predictive uncertainty does not necessarily capture all relevant sources of market risk. In many financial markets, trading activity itself generates feedback loops that amplify price movements. These endogenous dynamics can lead to clusters of extreme events and volatility bursts.

Hawkes processes provide a natural framework for modeling such self-exciting event dynamics. By estimating the intensity of event arrivals, Hawkes models quantify the degree of endogenous market activity and have been widely applied in financial microstructure analysis.

In this work, we leverage Hawkes intensity as an additional risk signal that captures endogenous market excitation. This design enables a unified signal construction framework in which heterogeneous forecasting models can be consistently transformed into risk-aware trading decisions.

Chapter 3

Literature review

3.1 Forecasting Models for Financial Time Series

Forecasting is a central task in quantitative finance, since trading decisions are often built upon expectations of future returns and the corresponding assessment of uncertainty. In financial markets, and especially in cryptocurrency markets, forecasting is challenging because price series often exhibit heavy tails, volatility clustering, and regime shifts [12–15]. This has motivated a long line of research ranging from classical econometric models to deep neural networks and, more recently, time-series foundation models (TSFMs). In this section, we review the main forecasting paradigms relevant to this thesis, moving from classical white-box models to modern pretrained black-box models.

3.1.1 Classical Econometric Models for Financial Forecasting

Classical econometric forecasting typically starts from parsimonious parametric models with strong structural assumptions. These models remain important because they are interpretable, computationally efficient, and closely connected to statistical inference in finance [3, 11, 16]. In this thesis, they also serve as natural white-box baselines against which more flexible deep learning models can be compared.

A standard starting point is the autoregressive integrated moving average (ARIMA) model. In compact notation, ARIMA(p, d, q) can be written as

$$\phi(B)(1 - B)^d y_t = \theta(B)\varepsilon_t, \tag{3.1}$$

where B denotes the backshift operator, $\phi(B)$ is the autoregressive polynomial, $\theta(B)$ is the moving-average polynomial, d is the differencing order, and ε_t is an innovation

term [3, 11]. Conceptually, ARIMA captures temporal dependence through linear combinations of past observations and past shocks. Its main strength lies in modeling low-dimensional linear dynamics in a transparent way. At the same time, this linear structure also limits its ability to represent the nonlinear and regime-dependent behavior often observed in financial markets.

Since return prediction alone is insufficient in finance, volatility modeling is usually introduced alongside mean modeling. The most influential framework is ARCH/GARCH. In particular, the GARCH$(1,1)$ model specifies the conditional variance as

$$\sigma_t^2 = \omega + \alpha\varepsilon_{t-1}^2 + \beta\sigma_{t-1}^2, \tag{3.2}$$

where $\omega > 0$, $\alpha \geq 0$, and $\beta \geq 0$ [4, 5]. This formulation captures volatility clustering, namely the empirical regularity that large shocks tend to be followed by large shocks and calm periods by calm periods. In financial applications, σ_t^2 is often interpreted as a conditional risk proxy.

For practical forecasting, ARIMA and GARCH are often combined: ARIMA models the conditional mean, while GARCH models the conditional variance [16]. Such a decomposition is especially natural in finance, where expected return and risk are conceptually distinct objects. In empirical cryptocurrency studies, GARCH-type models have been widely used to model the pronounced volatility of Bitcoin and related assets [17, 18]. Nevertheless, the strong assumptions of linearity, stationarity, and parametric variance dynamics can become restrictive in crypto markets, whose behavior is often driven by abrupt nonlinear shifts, speculative episodes, and heterogeneous cross-market interactions [13–15]. This limitation motivates the move toward more flexible deep learning approaches.

3.1.2 Deep Learning Models for Financial Forecasting

Deep learning methods were introduced into financial forecasting to better capture nonlinear dependencies and richer temporal representations than those available in classical parametric models. Compared with ARIMA-type approaches, neural networks are less dependent on explicit linear assumptions and can, in principle, learn complex mappings directly from historical data [6, 7, 19].

Among recurrent models, long short-term memory (LSTM) networks are especially representative. LSTMs were designed to mitigate the vanishing-gradient problem of standard recurrent neural networks by introducing gated memory cells [20]. In financial forecasting, LSTM-based models have often been reported to outperform traditional

statistical baselines when nonlinear structure is strong [6, 7]. However, these models still process information recurrently, which limits parallelization and may hinder efficient scaling.

Transformer models provide a different solution by replacing recurrence with self-attention [21]. Through self-attention, the model can directly relate each time step to all others in the input sequence, which improves parallelization and facilitates the learning of long-range dependencies. For time-series forecasting, this shift was highly influential and led to a broad family of transformer-based forecasting models [22–24].

From the viewpoint of this thesis, the main conceptual advantage of deep learning is not merely higher predictive power, but the transition from handcrafted linear structure to flexible nonlinear representation learning. Still, many early deep forecasting models are optimized mainly for point prediction, typically under losses such as mean squared error. For trading applications, this is an important limitation: a point forecast of future return provides only a central estimate, but not a direct description of predictive dispersion or tail uncertainty. This gap is one of the main reasons why recent literature has increasingly moved toward probabilistic forecasting and TSFMs.

3.1.3 Time-Series Foundation Models

Inspired by the success of large pretrained models in natural language processing and computer vision, recent work has extended the foundation model paradigm to time series [23–25]. The key idea is to pretrain a model on large and heterogeneous collections of time series so that it can learn transferable temporal representations and then generalize to downstream forecasting tasks with little or no task-specific retraining. In contrast to traditional local forecasting models, which fit one model per dataset or per series, TSFMs aim to provide a reusable forecasting engine across domains.

The TSFM literature now includes a rapidly growing family of pretrained forecasting models, such as Chronos, TimeGPT, TimesFM, Moirai, and Lag-Llama [10, 26–29]. Although these models differ in architecture and training objectives, they share several broad characteristics. First, they are trained on large multi-domain corpora rather than a single target dataset. Second, they are designed to support zero-shot or few-shot generalization. Third, many of them move beyond deterministic point forecasting and instead output probabilistic summaries, such as quantiles or parametric predictive distributions. These features make TSFMs particularly relevant for financial applications, where distributional uncertainty is often as important as point accuracy.

From a methodological perspective, TSFMs can be organized according to their underlying model architectures. Prior survey works [22, 23] broadly categorize time-series models into Transformer-based, non-Transformer-based (e.g., MLP/CNN/RNN), and emerging sequence modeling paradigms such as state-space models. Figure 3.1 summarizes this taxonomy together with representative models.

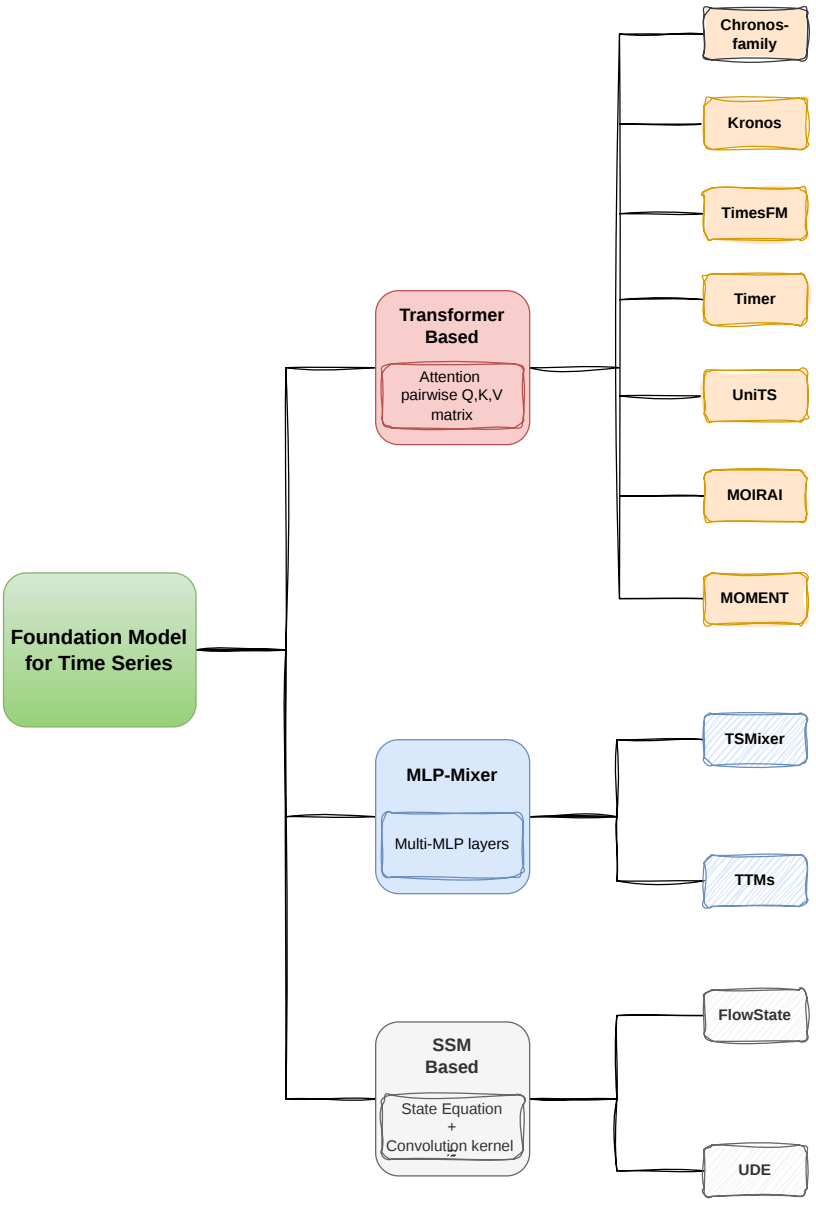


FIGURE 3.1: A taxonomy of time-series foundation models from an architectural perspective, including Transformer-based, MLP-based, and state-space-model-based approaches, together with representative models.

From this perspective, Transformer-based models dominate current TSFM research due to their ability to capture long-range dependencies via attention mechanisms [21]. At the

same time, lightweight MLP-based models such as TSMixer and TTMs offer improved efficiency, while state-space-model-based approaches provide an alternative paradigm for long-sequence modeling with favorable computational scaling. This architectural diversity reflects the ongoing exploration of scalable and generalizable sequence modeling techniques in time-series foundation models.

Chronos2. Chronos2 is a pretrained transformer-based time-series foundation model that supports zero-shot forecasting across a wide range of tasks, including univariate, multivariate, and covariate-informed settings [30–32].

A key strength of Chronos2 lies in its ability to produce probabilistic forecasts through a multi-quantile output head, enabling direct modeling of predictive uncertainty. In addition, its architecture incorporates mechanisms for sharing information across related series and covariates, which improves generalization in realistic forecasting scenarios.

These properties make Chronos2 particularly suitable for financial applications, where both cross-series dependencies and uncertainty estimation are essential. For this reason, Chronos2 is adopted as the primary TSFM backbone in this thesis.

3.2 Uncertainty Quantification and Risk Measurement

In quantitative finance, trading decisions are rarely based on return forecasts alone. Instead, financial theory emphasizes the importance of balancing expected return with risk. Classical portfolio theory and modern risk management frameworks therefore evaluate investment performance through risk-adjusted metrics rather than raw profitability [1, 2, 33]. In practice, risk is often interpreted as the uncertainty associated with future returns, particularly the possibility of large losses or unstable market behavior.

In time-series modeling of financial markets, this uncertainty is frequently operationalized through volatility. A widely adopted approach is to model the conditional variance of returns as a time-varying risk proxy. In ARCH-type models introduced by Engle [4], the conditional variance depends on past squared shocks, capturing the empirical phenomenon of volatility clustering.

While econometric volatility models provide one approach to risk estimation, modern forecasting research has increasingly shifted toward probabilistic prediction. Instead of producing a single point estimate for a future observation, probabilistic forecasting aims to estimate the entire conditional distribution of future outcomes. This perspective has been formalized in the statistical forecasting literature through the concept of proper

scoring rules, which evaluate predictive distributions rather than single-point forecasts [9]. By modeling predictive distributions explicitly, probabilistic forecasting allows the uncertainty of future outcomes to be quantified directly.

A widely used representation of probabilistic forecasts is quantile prediction. Quantile regression, originally introduced by Koenker and Bassett [8], estimates conditional quantiles of a response variable rather than its conditional mean. Given a quantile level τ, the quantile loss (also known as the pinball loss) is defined as

$$L_\tau(y, \hat{y}) = \max\{\tau(y - \hat{y}), (\tau - 1)(y - \hat{y})\}. \tag{3.3}$$

Minimizing this loss produces estimates of conditional quantiles of the predictive distribution. By predicting multiple quantiles simultaneously, a forecasting model can approximate the full distribution of possible future outcomes.

An important consequence of quantile forecasting is that the width of the predicted quantile interval naturally reflects predictive uncertainty. For example, if a model predicts the lower and upper quantiles $Q_{0.1}$ and $Q_{0.9}$, the interval width

$$W_t = Q_{0.9}(t) - Q_{0.1}(t) \tag{3.4}$$

provides a direct measure of the dispersion of the predictive distribution. A wider quantile band indicates greater uncertainty about future outcomes, while a narrower band suggests more confident predictions. In this sense, quantile intervals can be interpreted as a model-implied measure of predictive uncertainty. Chronos2 outputs a dense grid of quantiles for each forecasting horizon, enabling a detailed representation of the predictive distribution.

From the perspective of financial modeling, this probabilistic output is particularly attractive. While classical models such as GARCH quantify risk through the stochastic innovation term ε_t and the resulting conditional variance, quantile-based models describe uncertainty through the spread of predicted quantiles. In other words, both approaches capture the unpredictable component of market dynamics, but they do so through different representations.

This observation provides a natural bridge between econometric risk modeling and modern probabilistic forecasting. In the context of this thesis, the width of the predicted quantile band produced by Chronos2 can therefore be interpreted as a model-based proxy for predictive uncertainty. Analogous to the role played by conditional variance in GARCH models, the dispersion of quantile forecasts provides a quantitative indicator of market instability and potential risk. This interpretation enables a unified view

in which both classical econometric models and modern time-series foundation models contribute uncertainty measures that can be incorporated into risk-aware trading decisions.

3.3 Hawkes Processes and Endogenous Market Risk

While predictive uncertainty captures risk arising from forecasting models, financial markets themselves may generate additional instability through endogenous dynamics. Empirical studies have long documented that financial market events often exhibit strong temporal clustering, where the occurrence of one event increases the likelihood of subsequent events. Such behavior is commonly referred to as self-excitation or endogenous market activity.

A natural framework for modeling such dynamics is the Hawkes process, a class of self-exciting point processes originally introduced by Hawkes [34]. In a Hawkes process, the conditional intensity of events evolves as

$$\lambda(t) = \mu + \sum_{t_i < t} \phi(t - t_i), \tag{3.5}$$

where μ represents the baseline event intensity and the kernel $\phi(\cdot)$ models the excitation effect of past events. This formulation reflects the idea that market activity can propagate through feedback mechanisms, where previous trades or price movements increase the probability of future events.

Hawkes processes have been widely adopted in financial econometrics to study market microstructure, volatility clustering, and price jumps. A comprehensive survey by Bacry et al. [35] shows that a substantial fraction of financial market activity can be explained by endogenous feedback mechanisms captured by Hawkes models. Similarly, Filimonov and Sornette [36] demonstrate that market instability and flash-crash dynamics can be quantified through the degree of endogeneity measured by Hawkes processes.

These endogenous dynamics are particularly relevant in cryptocurrency markets. Due to high leverage, rapid information diffusion, and strong behavioral feedback among traders, cryptocurrency markets often exhibit extreme volatility and panic-driven cascades. Empirical studies have shown that cryptocurrency price movements and trading activity display significant self-excitation patterns consistent with Hawkes-type dynamics [13, 37]. These findings suggest that a significant portion of market risk arises not only from external shocks but also from internally amplified trading activity.

3.4 Risk-adjusted Return in Financial Decision Making

In financial decision making, trading strategies are typically evaluated using risk-adjusted performance metrics rather than raw returns. Classical portfolio theory introduced the fundamental idea that investment decisions should balance expected return against risk [1]. This framework forms the basis of modern quantitative trading, where forecasts of asset returns must be interpreted relative to the associated uncertainty.

One of the most widely used measures of risk-adjusted performance is the Sharpe ratio [2], defined as

$$SR = \frac{E[R - R_f]}{\sigma}, \tag{3.6}$$

where R denotes portfolio return, R_f is the risk-free rate, and σ represents the volatility of returns. The Sharpe ratio measures excess return per unit of risk and is commonly used to compare the performance of trading strategies.

More generally, modern financial risk management often relies on distribution-based measures such as Value-at-Risk (VaR), which quantifies the maximum expected loss at a given confidence level [33, 38]. These approaches highlight that trading decisions fundamentally depend on both expected returns and the uncertainty associated with future price movements.

Consequently, quantitative trading frameworks typically combine predictive signals with risk estimates when constructing trading strategies. Forecasting models provide estimates of expected returns, while risk measures such as volatility or distributional uncertainty determine the appropriate scaling of positions. This principle forms the foundation of many practical trading systems, where predicted returns are normalized or adjusted by estimated risk to produce stable and robust decision signals.

Chapter 4

Methods

4.1 System Overview

Figure 4.1 presents the overall pipeline of the proposed trading framework. The system starts from cryptocurrency market data collection, followed by log-return transformation to obtain the target time series for both forecasting and excitation modeling. Based on the transformed return series, the framework contains two parallel modeling components: a forecasting branch and a market excitation branch.

In the forecasting branch, both white-box and black-box models are used to estimate future return dynamics and the corresponding native risk measures. The white-box baseline is represented by the ARIMA–GARCH framework, where predicted return and conditional volatility are obtained jointly. The black-box forecasting model is built on Chronos2, which produces probabilistic forecasts and quantile-based uncertainty intervals. These model outputs provide two key inputs for trading decisions: predicted return and model-native risk.

In parallel, the market excitation branch models endogenous market risk through a Hawkes process. The return series is first transformed into an event sequence through threshold-based eventization, and the Hawkes model is then fitted on historical data to estimate the time-varying excitation intensity $\lambda(t)$. This excitation intensity serves as an additional market risk descriptor that captures self-exciting dynamics not explicitly represented by the forecasting models.

The final trading signal is constructed through a unified risk scaling layer. Specifically, predicted return is combined with native model risk and Hawkes excitation risk to produce a risk-adjusted signal for trading decisions. This design allows heterogeneous

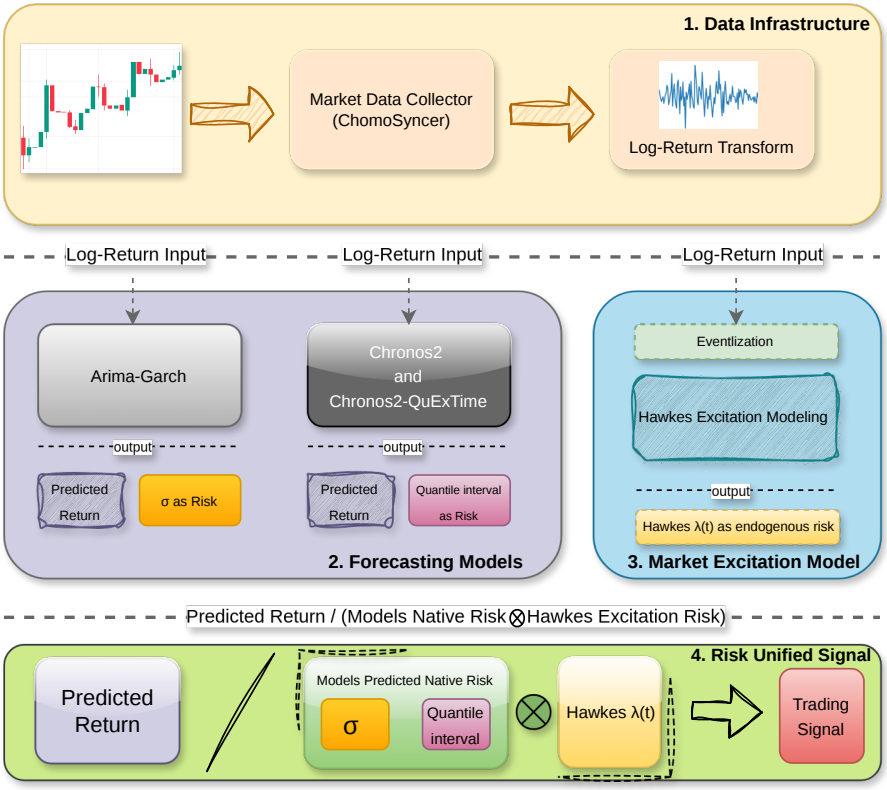
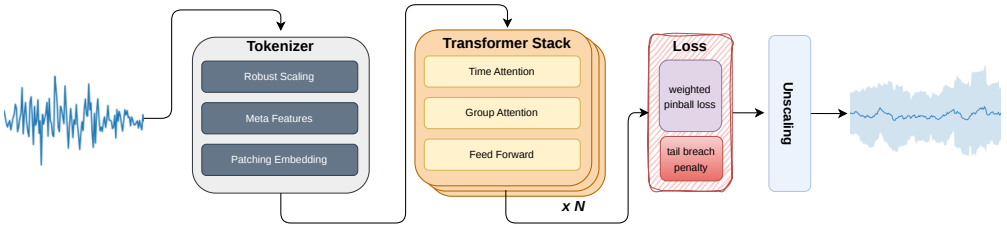


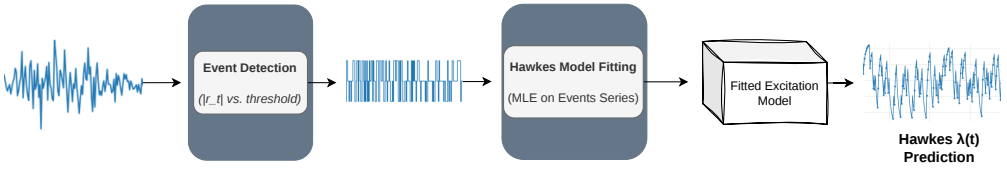
FIGURE 4.1: System overview of the proposed trading framework. The pipeline starts from market data collection and log-return transformation, then proceeds through forecasting models and Hawkes-based excitation modeling, and finally constructs a unified risk-adjusted trading signal by combining predicted return, native model risk, and endogenous market excitation risk.

forecasting models to be evaluated under a common signal construction principle while explicitly incorporating endogenous market excitation into the decision process.

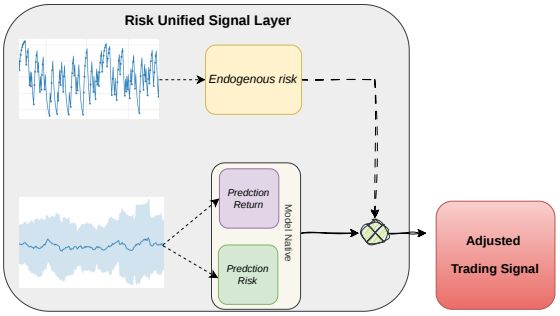
Figure 4.2 further illustrates the three key modules of the framework in detail. Figure 4.2a shows the Chronos2-based forecasting module, including tokenization, transformer-based representation learning, and the proposed horizon- and tail-aware loss, which is QuExTime Loss. Figure 4.2b presents the Hawkes excitation modeling pipeline, where historical returns are converted into event sequences and used to estimate the endogenous market excitation intensity. Finally, Figure 4.2c shows the risk-unified signal layer, where predicted return, prediction risk, and endogenous market risk are integrated to generate the adjusted trading signal.



(A) Chronos2 forecasting module with the QuExTime loss design.



(B) Hawkes excitation modeling for endogenous risk estimation.



(C) Risk-unified signal layer for trading signal generation.

FIGURE 4.2: Detailed modules of the proposed framework. (a) illustrates the Chronos2-based forecasting architecture with the proposed QuExTime loss. (b) shows the Hawkes modeling process used to estimate endogenous market excitation intensity. (c) presents the risk-unified signal layer, where predicted return, model-native risk, and Hawkes-based endogenous risk are combined to generate the adjusted trading signal.

4.2 Data Infrastructure and Market Data Pipeline

Reliable financial modeling requires a consistent and reproducible data pipeline. To support both forecasting and market excitation modeling, we construct a unified data infrastructure that collects, processes, and stores cryptocurrency market data in a temporally consistent manner.

4.2.1 Market Data Collection and Storage

Cryptocurrency market data are obtained from the Binance exchange using a custom-built synchronization system called *ChomoSyncer*. The system retrieves candlestick

(Kline) data through both WebSocket streams and REST interfaces, enabling continuous acquisition of real-time market data while also allowing historical backfilling when necessary.

The collected data are distributed through a Redis streaming layer and simultaneously stored in MongoDB for long-term persistence. Redis enables low-latency access for downstream modeling components, while MongoDB provides a persistent storage layer for offline experiments, backtesting, and reproducibility.

Although historical market data can be downloaded directly through exchange APIs, a streaming-based synchronization system provides several advantages for quantitative research. First, real-time data acquisition avoids potential hindsight bias introduced by retrospective data reconstruction. Second, the same infrastructure can support both offline research and potential online deployment, allowing the forecasting and trading modules to operate on identical data pipelines.

Implementation details of the synchronization system, including system architecture and data flow design, are provided in Appendix 9

4.2.2 Return Series Construction and Unified Preprocessing

After data collection, raw price series are transformed into log-return sequences, which serve as the unified modeling target across the framework. For a price series P_t, the log-return is defined as

$$r_t = \log(P_t) - \log(P_{t-1}). \tag{4.1}$$

Using log-returns rather than raw prices provides several advantages for financial modeling. Return series are typically more stationary and better suited for econometric modeling. Moreover, return-based representations naturally align with both forecasting models and event-based excitation modeling. In the forecasting branch, the objective is to predict the future return distribution. In the Hawkes excitation branch, return magnitudes are used to generate event sequences through threshold-based eventization.

All subsequent modeling components therefore operate on the same return representation, ensuring consistency between the forecasting models and the excitation modeling layer.

4.2.3 Data Splitting and Leakage Control

To ensure rigorous evaluation, all experiments follow a strictly chronological data split. The dataset is divided into training, validation, and test segments according to their temporal order.

Model parameters, normalization statistics, and event thresholds are estimated exclusively on the training set. These parameters are then fixed and applied to the validation and test periods without accessing future information.

This design prevents information leakage and ensures that all forecasting and trading decisions are made using only the information available up to the prediction time. For the Hawkes excitation model, parameters are estimated on the training period, while excitation intensity is updated sequentially during the test period using observed historical events.

By enforcing strict temporal consistency, the proposed data pipeline ensures that the evaluation results accurately reflect realistic trading conditions.

4.3 Forecasting Models

Based on the return series constructed in Section 4.2, the forecasting branch estimates future return dynamics together with model-native uncertainty measures. Two types of forecasting models are considered in the framework: a classical white-box econometric model and a black-box probabilistic forecasting model. Although their internal structures differ, both models ultimately produce two quantities required by the trading framework: a predicted return $\hat{r}_{t+1}$ and a model-implied prediction risk $\hat{\rho}_t$. These quantities are later combined with endogenous market risk in the unified signal layer described in Section 4.5.

4.3.1 White-box Forecasting Models

The white-box baseline adopts an ARIMA–GARCH framework applied to the log-return series r_t. At each decision time step, a rolling historical window is used to estimate model parameters and generate one-step-ahead forecasts.

The return process is decomposed into a conditional mean component and a conditional variance component. The ARIMA model captures the predictable structure of the return series and produces the conditional mean forecast, while the residual process is modeled

using a GARCH process to estimate time-varying volatility. Model parameters are estimated on the training window and subsequently updated in a rolling manner during evaluation.

At each prediction step, the model produces two quantities:

$$\hat{r}_{t+1} = \hat{\mu}_{t+1}, \qquad \hat{\rho}_t = \hat{\sigma}_{t+1}.$$

Here $\hat{r}_{t+1}$ represents the expected future return and $\hat{\rho}_t$ represents the volatility-based risk estimate produced by the white-box model. Within the trading framework, the expected return provides the directional trading signal, while the volatility estimate serves as the native risk measure.

These two outputs are passed to the unified signal layer, where the predicted return is adjusted by the associated risk estimates before generating the final trading signal.

Detailed formulations of the ARIMA mean model and the GARCH volatility dynamics are provided in Appendix 10.1 and 10.2.

4.3.2 Black-box Probabilistic Forecasting Model

The black-box forecasting component is built upon a time-series foundation model (TSFM), specifically the Chronos2 architecture, which produces quantile-based probabilistic forecasts. Given a historical return sequence as input, the model outputs a set of conditional quantiles:

$$\hat{q}_{\tau,t+h}, \qquad \tau \in \mathcal{T},$$

which characterize the predictive distribution of future returns over the forecasting horizon.

Adaptation settings. To systematically evaluate the effectiveness of model adaptation in cryptocurrency markets, we consider three configurations:

- **Zero-shot**: directly applying the pretrained Chronos2 model without any domain adaptation;
- **Native fine-tuning**: fine-tuning Chronos2 using its original training objective;

- **Proposed QuExTime loss fine-tuning**: fine-tuning Chronos2 using the loss function introduced in this work.

This design allows us to isolate the contribution of the QuExTime loss from both pre-training and standard fine-tuning.

QuExTime loss function. The key methodological contribution of this work is a volatility-aware quantile loss, referred to as the *Quantile Extreme Time Decay (QuExTime) Loss*. The objective is to explicitly incorporate two inductive biases that are particularly important for cryptocurrency markets: (i) emphasis on extreme quantiles and (ii) prioritization of short-term forecasting horizons.

Formally, let y_{t+h} denote the realized return and $\hat{q}_{\tau,t+h}$ the predicted quantile at level $\tau \in \mathcal{Q}$. The QuExTime loss is defined as a normalized weighted aggregation of pinball losses:

$$\begin{cases} \mathcal{L} = \dfrac{\sum_{h=1}^{H} \sum_{\tau \in \mathcal{Q}} m_{\tau,h}\, w^{(q)}(\tau)\, w^{(t)}(h)\, \rho_\tau(y_{t+h} - \hat{q}_{\tau,t+h})}{\sum_{h=1}^{H} \sum_{\tau \in \mathcal{Q}} m_{\tau,h}\, w^{(q)}(\tau)\, w^{(t)}(h)}, \\ \rho_\tau(u) = u\,(\tau - \mathbf{1}_{u<0})\,, \\ w^{(q)}(\tau) = 1 + \gamma\,(2|\tau - 0.5|)^{\alpha}\,, \\ w^{(t)}(h) = \delta^{\,h-1}, \quad \delta \in (0, 1]. \end{cases}$$

Here, $\rho_\tau(\cdot)$ denotes the standard pinball loss [8], $m_{\tau,h}$ is a validity mask for forecast entries, $w^{(q)}(\tau)$ increases the contribution of tail quantiles, and $w^{(t)}(h)$ applies an exponential decay over the prediction horizon.

The quantile weighting function assigns larger importance to extreme quantiles, thereby encouraging the model to better capture heavy-tailed return behavior, which is a well-documented stylized fact in cryptocurrency markets [12, 13]. The time-decay weighting prioritizes short-term prediction accuracy, reflecting the fast-changing and high-frequency nature of crypto market dynamics.

The normalization term ensures that the overall scale of the loss remains stable across different weighting configurations, preventing numerical instability during training.

In summary, the proposed QuExTime loss can be interpreted as a reweighted pinball objective that explicitly biases the model toward tail-sensitive and short-horizon probabilistic forecasting. The implementation details are provided in the form of pseudocode in Appendix 10.3.

Forecast-to-signal interface. The probabilistic outputs are converted into return and risk estimates:

$$\hat{r}_{t+1} = \hat{q}_{0.5,t+1}, \qquad \hat{\rho}_t = \hat{q}_{0.9,t+1} - \hat{q}_{0.1,t+1}.$$

Here $\hat{r}_{t+1}$ represents the expected return, while $\hat{\rho}_t$ measures the model-implied uncertainty (prediction interval width), which serves as the *native risk* component in the downstream trading layer.

These quantities are subsequently integrated with the Hawkes-based excitation model for unified risk-adjusted signal construction.

4.4 Hawkes-based Market Excitation Layer

In addition to model-implied prediction risk, the framework explicitly accounts for endogenous market dynamics through a Hawkes excitation model. This component captures self-exciting behavior in financial markets, where large price movements tend to cluster in time.

Illustrated in Figure 4.2b, the Hawkes modeling pipeline operates on the same return series r_t used by the forecasting models. First, the return sequence is converted into an event series by applying a threshold-based event detection rule. Specifically, an event is triggered when the absolute return exceeds a predefined threshold τ,

$$|r_t| > \tau.$$

The resulting event timestamps form an event sequence $\{t_i\}$ used to estimate a Hawkes point process.

A Hawkes process models the conditional event intensity as

$$\lambda(t) = \mu + \sum_{t_i < t} \alpha e^{-\beta(t-t_i)},$$

where μ represents the background intensity, while α and β control the magnitude and decay speed of the excitation effect. Model parameters are estimated using maximum likelihood on the training event sequence.

During evaluation, the fitted parameters remain fixed while the excitation intensity λ_t is updated sequentially as new events occur. This produces a time-varying measure of endogenous market activity.

Within the trading framework, the excitation intensity

$$\hat{\rho}_t^{\text{exo}} = \lambda_t$$

is interpreted as an estimate of endogenous market risk. High excitation intensity indicates increased clustering of large market moves and therefore elevated market instability.

The endogenous risk estimate $\hat{\rho}_t^{\text{exo}}$ is subsequently combined with the forecasting model outputs $\hat{r}_{t+1}$ and $\hat{\rho}_t$ in the unified signal layer described in Section 4.5.

4.5 Unified Risk-adjusted Signal Construction

The forecasting models produce two quantities at each decision time step: the predicted return $\hat{r}_{t+1}$ and the model-implied prediction risk $\hat{\rho}_t$. In parallel, the Hawkes excitation layer produces an endogenous risk measure $\hat{\rho}_t^{\text{exo}} = \lambda_t$.

To combine these components, we construct a unified risk-adjusted signal:

$$S_t = \frac{\hat{r}_{t+1}}{\hat{\rho}_t + \gamma \hat{\rho}_t^{\text{exo}}},$$

where γ controls the contribution of the excitation-based risk.

Signal execution. The trading position is determined by comparing S_t with predefined thresholds:

$$\text{position}_t = \begin{cases} +1, & S_t > \theta, \\ -1, & S_t < -\theta, \\ 0, & \text{otherwise.} \end{cases}$$

This threshold-based rule directly maps the risk-adjusted signal into long, short, or neutral positions.

This unified formulation enables a consistent comparison across forecasting models while incorporating both model-based uncertainty and endogenous market excitation into the trading decision.

Chapter 5

Experiments and Results

This chapter evaluates the proposed framework from two complementary perspectives. First, we examine whether the proposed forecasting models improve return prediction and uncertainty characterization in cryptocurrency markets. Second, we evaluate whether these forecasting improvements can be translated into better trading performance under the unified risk-adjusted signal framework.

Following the overall thesis design, the experiments are organized into two layers: a *forecasting layer* and a *trading layer*. In this chapter, we first present the forecasting results and then move to the trading evaluation. All experiments are conducted under strictly time-ordered data splits to avoid information leakage.

5.1 Experimental Setup

5.1.1 Data Source

The dataset consists of historical cryptocurrency market data collected from the Binance exchange via the data infrastructure described in Chapter 4. Following the forecasting pipeline introduced earlier, the raw close-price series are transformed into log returns before model training and evaluation. This design ensures that both white-box econometric models and black-box foundation models are trained on a unified return-based representation.

5.1.2 Market Stratification

Rather than evaluating a large basket of assets, this study focuses on two representative cryptocurrencies with clearly different volatility characteristics:

- **Bitcoin (BTC):** a large-cap benchmark cryptocurrency with relatively lower volatility, stronger market efficiency, and comparatively cleaner return dynamics. BTC is used to represent a more mature and structurally stable crypto market regime.
- **Dogecoin (DOGE):** a highly volatile altcoin with stronger speculative behavior, more frequent abrupt jumps, and heavier tail noise in return series. DOGE is used to represent a noisier and more unstable market regime.

This two-asset design allows us to test whether the proposed forecasting framework is robust across different volatility regimes. In particular, it enables us to examine whether the model behaves differently in a relatively stable benchmark market (BTC) and in a high-noise speculative market (DOGE).

5.1.3 Training, Validation, and Testing Protocol

All experiments follow a strictly time-ordered split to ensure a realistic forecasting and trading evaluation pipeline. The data are divided into three non-overlapping periods:

- **Training set:** from 2019-10-31 to 2025-10-31
- **Validation set:** from 2025-11-01 to 2025-11-30
- **Test set:** from 2025-12-10 to 2026-01-25

The training set is used to fit model parameters. The validation set is used to select the best model configuration or checkpoint, especially for fine-tuned deep models. The final evaluation is reported on the held-out test set only.

To provide a clearer understanding of the underlying data distribution, we visualize representative time series for BTC and DOGE, including both the original K-line prices and the corresponding log-return transformations.

These visualizations highlight a fundamental difference in data characteristics across assets. BTC exhibits relatively stable volatility clustering and smoother return dynamics, while DOGE displays pronounced heavy tails and irregular bursts of extreme movements. Such differences are consistent with the well-documented stylized facts of cryptocurrency markets, including heavy-tailed return distributions and volatility clustering.

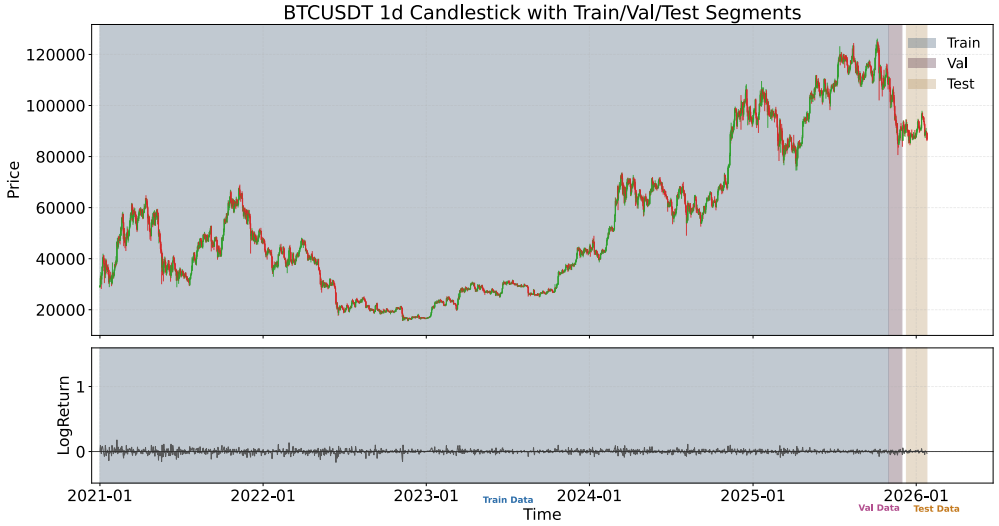


FIGURE 5.1: Log-return time series of BTC. The series exhibits moderate volatility clustering and relatively smoother dynamics, reflecting a more mature and stable market structure.

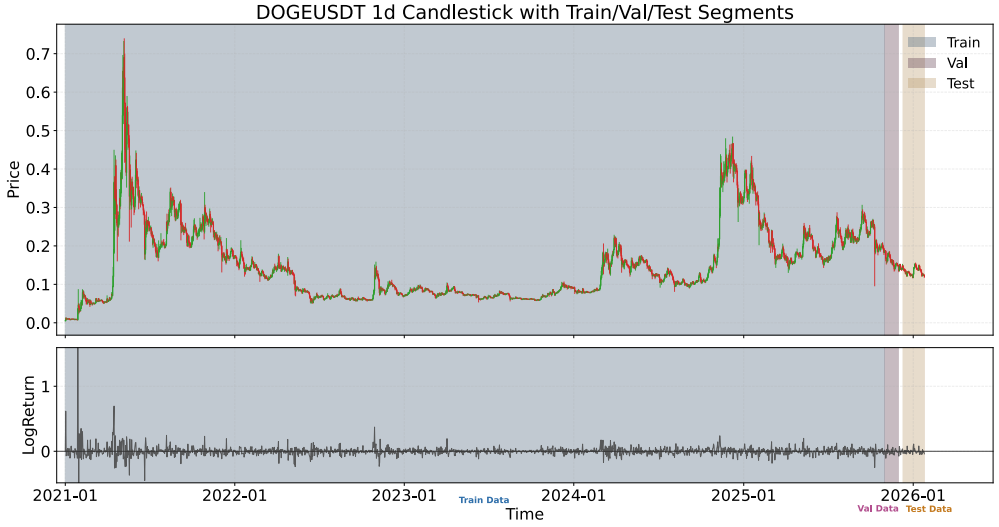


FIGURE 5.2: Log-return time series of DOGE. The series demonstrates extreme volatility, heavy-tailed fluctuations, and frequent abrupt jumps, indicating a highly speculative and unstable market regime.

This contrast plays a crucial role in our experimental design. It allows us to evaluate whether forecasting models and risk-aware trading signals remain robust under different market regimes, ranging from relatively stable large-cap assets to highly volatile speculative assets.

The same data split is used for all forecasting models. This unified protocol prevents future information leakage and ensures a fair comparison between classical econometric baselines and foundation-model-based approaches. Furthermore, it guarantees that any performance difference observed across models is attributable to modeling capability rather than inconsistencies in data partitioning.

5.2 Forecasting Performance Evaluation

We first evaluate the forecasting layer of the proposed framework. The objective of this part is to examine whether the QuExTime fine-tuning strategy improves both return prediction and uncertainty characterization in cryptocurrency markets.

5.2.1 Compared Forecasting Models

The forecasting experiments compare the following four models:

- **ARIMA + GARCH**: white-box econometric baseline
- **Chronos2 Zero-shot**: pretrained TSFM directly applied to crypto data without domain adaptation
- **Chronos2 Fine-tuned**: Chronos2 fine-tuned on cryptocurrency data
- **Chronos2-QuExTime**: Chronos2-QuExTime fine-tuned on cryptocurrency data.

These four models represent progressively stronger forecasting pipelines, ranging from classical econometric modeling to generic TSFM forecasting and finally to the proposed crypto-adapted probabilistic forecasting approach.

5.2.2 Evaluation Metrics

To comprehensively assess forecasting quality, we group the evaluation metrics into three categories: traditional error metrics, risk characterization metrics, and signal-oriented (trend-following) metrics.

Let y_t denote the realized return, $\hat{y}_t$ the predicted return, and $\hat{q}_{\tau,t}$ the predicted τ-quantile.

Traditional error metrics We evaluate point-forecast accuracy using standard regression-based measures.

- **MAE (Mean Absolute Error):** measures the average absolute deviation between predicted and realized returns:

$$\mathrm{MAE} = \frac{1}{N} \sum_{t=1}^{N} |\hat{y}_t - y_t| \,. \tag{5.1}$$

- **RMSE (Root Mean Squared Error):** measures the square root of the average squared error:

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{t=1}^{N}(\hat{y}_t - y_t)^2}. \tag{5.2}$$

These metrics evaluate the accuracy of the predicted central tendency of the return distribution. MAE provides a robust measure under heavy-tailed distributions, while RMSE remains sensitive to large deviations.

Risk characterization metrics To evaluate the quality of probabilistic forecasts, we adopt quantile-based pinball loss.

- **Pinball loss:** for quantile $\tau \in (0, 1)$, the pinball loss is defined as

$$\rho_\tau(u) = u\left(\tau - \mathbf{1}_{u<0}\right), \quad u = y_t - \hat{q}_{\tau,t}, \tag{5.3}$$

  with empirical form

$$\text{Pinball}_\tau = \frac{1}{N}\sum_{t=1}^{N}\rho_\tau(y_t - \hat{q}_{\tau,t}). \tag{5.4}$$

- **Pinball$_{q10}$ and Pinball$_{q90}$:** evaluate lower-tail and upper-tail prediction accuracy respectively.

- **Tail pinball average:** summarizes tail accuracy on both sides:

$$\text{TailPinball} = \frac{\text{Pinball}_{0.1} + \text{Pinball}_{0.9}}{2}. \tag{5.5}$$

Lower pinball loss indicates better quantile estimation. These metrics are particularly important in our framework, as the quantile interval is later used as a model-implied risk proxy in the trading layer.

Signal-oriented (trend-following) metrics Beyond statistical accuracy, trading applications require forecasts to produce meaningful decision signals. In our framework, the predicted return and uncertainty are transformed into risk-adjusted signals, which can be interpreted as an *alpha proxy.*

- **Sign accuracy:** measures directional correctness:

$$\text{SignAcc} = \frac{1}{N}\sum_{t=1}^{N}\mathbf{1}\left(\text{sign}(\hat{y}_t) = \text{sign}(y_t)\right). \tag{5.6}$$

- **Rank IC (Spearman):** measures the rank correlation between predicted and realized returns:

$$\text{RankIC} = \text{corr}_{\text{Spearman}}(\hat{y}_t, y_t). \tag{5.7}$$

Compared with traditional forecasting errors, these signal-oriented metrics are more closely aligned with downstream trading performance. Sign accuracy reflects directional correctness, while Rank IC evaluates the quality of signal ordering, which is essential for generating effective trading decisions.

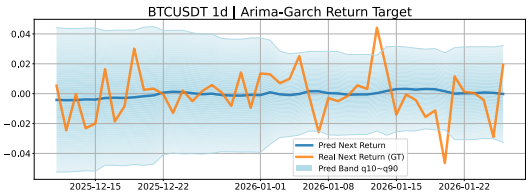
5.2.3 Forecast Visualization

After establishing the above evaluation protocol, we next present representative forecasting results on the test set. To make the comparison clearer, the visualization is organized by asset. For each cryptocurrency, we first show the forecast trajectories under the four competing models, and then summarize the corresponding results using grouped bar charts for the three metric categories: traditional error metrics, risk characterization metrics, and trend-following metrics.

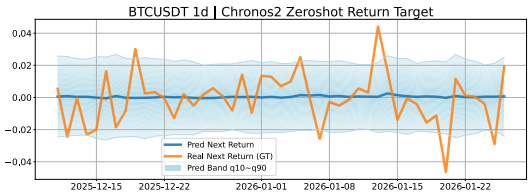
5.2.3.1 BTC Results

We first present the forecasting results for BTC, which serves as the large-cap benchmark asset with relatively lower volatility and cleaner return dynamics.

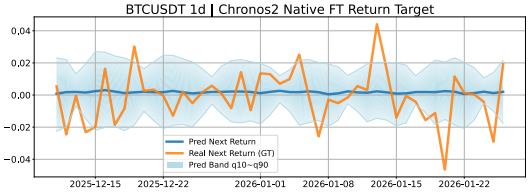
Figure 5.3 provides a visual comparison of the predicted return trajectories and the corresponding quantile intervals under the four forecasting models. This figure allows a direct inspection of how each model captures short-term dynamics and uncertainty bands.



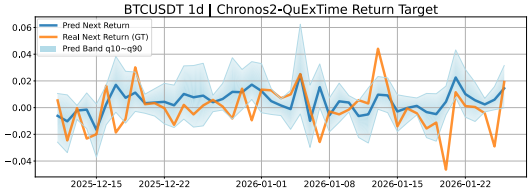
(A) ARIMA-GARCH



(B) Chronos2 Zeroshot

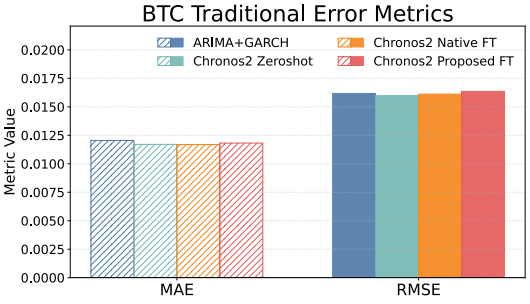


(C) Chronos2 Native Loss Finetuned

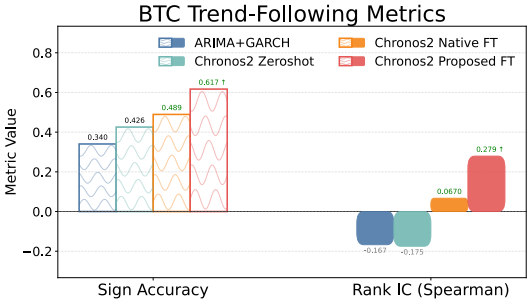


(D) Chronos2-QuExTime Finetuned

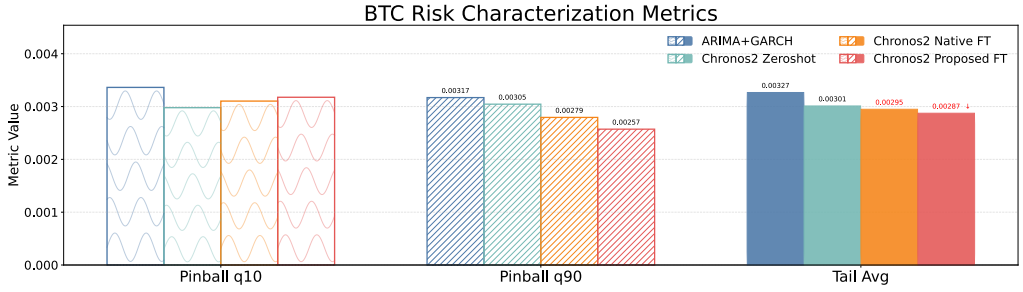
FIGURE 5.3: Forecast visualization on BTC test data under four forecasting models. In each subplot, the blue line denotes the predicted next return, the orange line denotes the realized next return, and the shaded region corresponds to the predicted quantile interval from q_{10} to q_{90}.



(A) Traditional error metrics



(B) Trend-following metrics



(C) Risk characterization metrics

FIGURE 5.4: Forecasting performance on BTC test data from three complementary perspectives. The top row compares traditional point forecasting errors (MAE, RMSE) and trend-following metrics (sign accuracy and rank IC). The bottom panel reports risk characterization metrics, including pinball loss at q_{10}, q_{90}, and their average, reflecting tail-risk calibration quality.

Figure 5.4 further summarize the quantitative performance from three complementary perspectives. Specifically, Figure 5.4a reports traditional point forecasting errors, Figure 5.4b focuses on directional and ranking properties that are directly relevant for

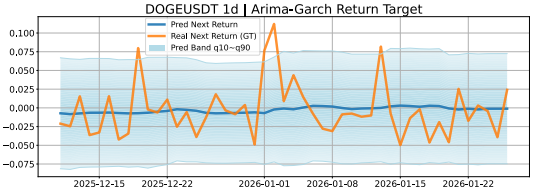
trading signal construction, and Figure 5.4c evaluates the quality of tail-risk characterization.

5.2.3.2 DOGE Results

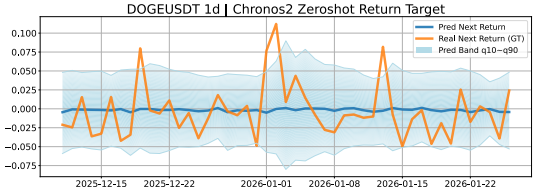
We then present the forecasting results for DOGE, which represents a more speculative and high-noise market regime with stronger tail fluctuations.

Figure 5.5 provides a visual comparison of the predicted return trajectories and the corresponding quantile intervals under the four forecasting models. This figure allows a direct inspection of how each model behaves under a high-volatility and noisy market environment.

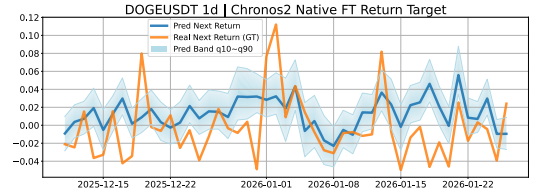
Figure 5.6 further summarizes the quantitative performance from three complementary perspectives. Specifically, Figure 5.6a reports traditional point forecasting errors, Figure 5.6b focuses on directional and ranking properties that are directly relevant for trading signal construction, and Figure 5.6c evaluates the quality of tail-risk characterization.



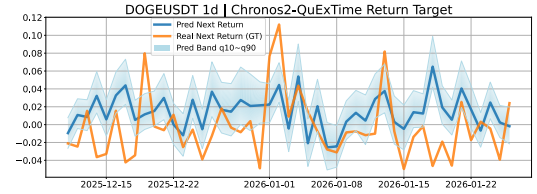
(A) ARIMA-GARCH



(B) Chronos2 Zeroshot

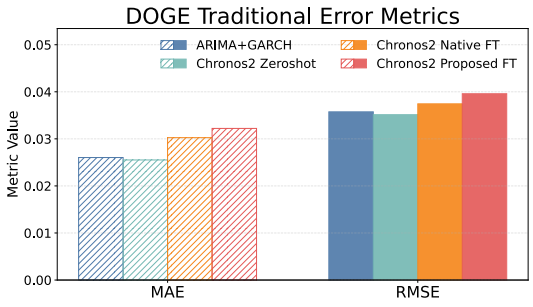


(C) Chronos2 Native Loss Finetuned

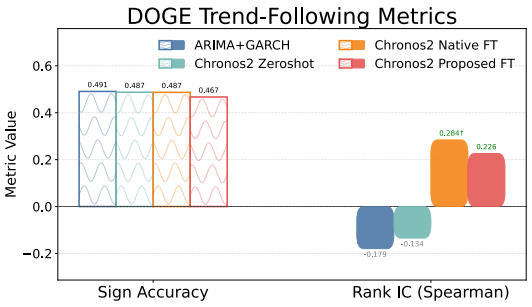


(D) Chronos2-QuExTime Finetuned

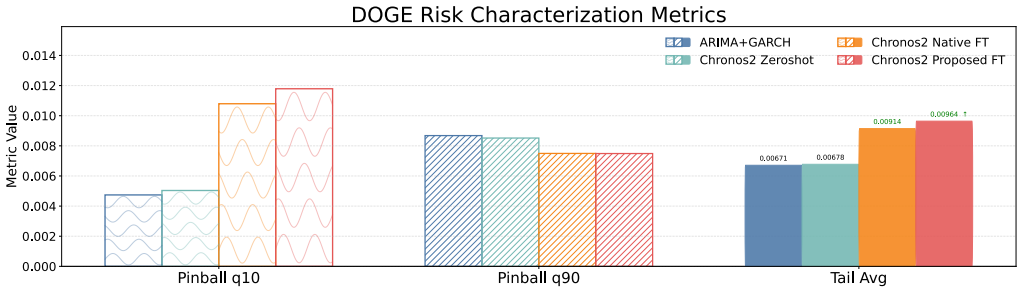
FIGURE 5.5: Forecast visualization on DOGE test data under four forecasting models.



(A) Traditional error metrics



(B) Trend-following metrics



(C) Risk characterization metrics

FIGURE 5.6: Forecasting performance on DOGE test data from three complementary perspectives. The top row compares traditional point forecasting errors (MAE, RMSE) and trend-following metrics (sign accuracy and rank IC). The bottom panel reports risk characterization metrics, including pinball loss at q_{10}, q_{90}, and their average, reflecting tail-risk calibration quality.

5.2.3.3 Unified Metric Summary Across BTC and DOGE

To provide a direct comparison across both market regimes, we finally summarize all forecasting metrics for BTC and DOGE in a unified table. This table allows the reader to compare model behavior across assets from the perspectives of point accuracy, uncertainty quality, and signal usefulness for trading.

Table 5.1 reports the forecasting results on the test period for both BTC and DOGE. Lower values are better for MAE, RMSE, and pinball-based metrics, while higher values are better for sign accuracy and rank IC.

TABLE 5.1: Unified forecasting and signal evaluation metrics across models and assets.

		Traditional Error Metrics		**Risk Characterization Metrics**			**Signal-Oriented Metrics**	
Asset	**Model**	**MAE**	**RMSE**	**Pinball$_{q10}$**	**Pinball$_{q90}$**	**Tail Avg**	**Sign Acc.**	**Rank IC**
BTC	ARIMA–GARCH	0.01205	0.01617	0.00336	0.00317	0.00327	0.340	-0.167
	Chronos2 (Zero-shot)	0.01170	**0.01600**	0.00298	0.00305	0.00301	0.426	-0.175
	Chronos2 (Native FT)	**0.01169**	0.01611	0.00310	0.00279	0.00295	0.489	0.067
	Chronos2-QuExTime FT	0.01182	0.01636	0.00318	0.00257	**0.00287**↓	**0.617**	**0.279**↑
DOGE	ARIMA–GARCH	0.02603	0.03574	0.00474	0.00868	**0.00671**	**0.491**	-0.179
	Chronos2 (Zero-shot)	**0.02551**	**0.03514**	0.00504	0.00852	0.00678	0.487	-0.134
	Chronos2 (Native FT)	0.03025	0.03744	0.01079	0.00750	0.00914↑	0.487	**0.284**↑
	Chronos2-QuExTime FT	0.03222	0.03959	0.01179	0.00749	0.00964↑	0.467	0.226

From Table 5.1, we observe a clear contrast between the two market regimes.

For BTC, the proposed QuExTime loss design consistently improves tail-risk characterization, as reflected by the monotonic decrease in Tail Avg across model variants. At the same time, both sign accuracy and rank IC exhibit substantial improvements with fine-tuning and further gains under the proposed loss. Importantly, these improvements are achieved without significant degradation in traditional error metrics such as MAE and RMSE. This indicates that the proposed loss effectively enhances uncertainty calibration and directional signal quality in relatively stable and liquid cryptocurrency markets.

In contrast, the behavior on DOGE is notably different. While fine-tuning improves rank IC, suggesting better trend extraction, both tail-risk metrics and traditional error metrics deteriorate compared to the zero-shot baseline. In particular, the proposed loss leads to a significant increase in Tail Avg and pinball losses, indicating degraded tail calibration. This suggests that, in highly volatile and noise-dominated markets, emphasizing tail penalties may cause the model to overfit extreme fluctuations, thereby reducing the reliability of uncertainty estimates.

These results highlight a regime-dependent trade-off. The proposed loss is effective in structured markets such as BTC, where tail behavior is more stable, but may introduce instability in highly noisy environments such as DOGE. This observation suggests that future work may benefit from adaptive loss designs that account for market-specific characteristics.

5.3 Trading Performance Evaluation

After evaluating the forecasting layer, we next examine whether these predictive outputs can be translated into effective trading performance. The key question in this part is not only whether a forecasting model produces accurate return estimates, but also whether its signals remain useful after being converted into executable trading decisions under risk control.

Following the framework proposed in Chapter 4, we construct trading signals from the return forecasts and their associated risk measures. For each forecasting model, we further compare two signal-generation settings: a baseline *No-Hawkes* setting and a *Hawkes-scaled* setting, where the original signal is additionally adjusted by the self-excitation intensity estimated from the Hawkes process. In this way, the trading layer directly tests whether market-endogenous risk scaling can improve the practical value of the forecasting models.

5.3.1 Compared Trading Configurations

The trading evaluation is built on top of the four forecasting models introduced in Section 5.2. For each forecasting model, we compare two trading configurations:

- **No-Hawkes:** the trading signal is generated directly from the forecasting output and its native risk estimate, without additional market-excitation adjustment.
- **Hawkes-scaled:** the trading signal is further adjusted by the Hawkes intensity, which serves as an external market risk scaler to reflect endogenous excitation in cryptocurrency markets.

As a result, each forecasting model gives rise to two corresponding trading variants, allowing a direct comparison between signals with and without Hawkes-based risk scaling. Detailed visualizations of the eventization process and the corresponding Hawkes intensity are provided in Appendix 11.1 and Appendix 11.2.

The evaluated forecasting backbones are:

- **ARIMA + GARCH**
- **Chronos2 Zero-shot**
- **Chronos2 Native Fine-tuned**
- **Chronos2-QuExTime Fine-tuned**

In addition, a **Buy-and-Hold** benchmark is reported as a market baseline. This benchmark provides a reference for whether the model-driven trading strategies are able to outperform passive exposure over the same test period.

5.3.2 Trading Evaluation Metrics

To evaluate the practical usefulness of the generated trading signals, we use five standard portfolio-level performance metrics: cumulative return, Sharpe ratio, maximum drawdown, Calmar ratio, and win rate. The formal definitions are given below.

Let $r_t^{(p)}$ denote the portfolio return at time t, and let V_t denote the portfolio value (equity curve).

- **CumRet (Cumulative Return):** measures the total return achieved by the strategy over the full test period:

$$\text{CumRet} = \frac{V_T}{V_0} - 1 = \prod_{t=1}^{T}(1 + r_t^{(p)}) - 1. \tag{5.8}$$

- **Sharpe Ratio:** measures risk-adjusted return as the ratio between the mean portfolio return and its standard deviation:

$$\text{Sharpe} = \frac{\mathbb{E}[r_t^{(p)}]}{\sqrt{\text{Var}(r_t^{(p)})}}. \tag{5.9}$$

- **MaxDD (Maximum Drawdown):** measures the largest peak-to-trough decline in portfolio value:

$$\text{MaxDD} = \max_{t\in[1,T]} \left(\frac{\max_{s\le t} V_s - V_t}{\max_{s\le t} V_s} \right). \tag{5.10}$$

- **Calmar Ratio:** measures return relative to maximum drawdown:

$$\text{Calmar} = \frac{\text{CumRet}}{\text{MaxDD}}. \tag{5.11}$$

- **WinRate:** measures the proportion of profitable periods:

$$\text{WinRate} = \frac{1}{T}\sum_{t=1}^{T} \mathbf{1}\left(r_t^{(p)} > 0\right). \tag{5.12}$$

Taken together, these metrics provide a balanced view of trading quality from the perspectives of profitability, risk-adjusted performance, drawdown control, and decision consistency.

5.3.3 Trading Result Visualization

We next visualize the trading performance on the test set for both BTC and DOGE. For each asset, we compare the equity curves generated by the four forecasting backbones under two signal-generation settings, namely *No-Hawkes* and *Hawkes-scaled*. A Buy-and-Hold benchmark is included in each comparison to provide a direct market reference.

Alongside the equity curves, we report a corresponding summary table of trading performance metrics for each configuration. To maintain a clear presentation in the main text, detailed visualizations of trading signal execution—such as buy/sell decisions and position changes under different configurations—are provided in Appendix 11.3 and Appendix 11.4.

5.3.3.1 BTC Trading Results

We first present the trading results for BTC, which serves as the benchmark large-cap cryptocurrency with relatively lower volatility and stronger market efficiency.

Figure 5.7 visualizes the BTC equity curves under the four forecasting backbones. For each model, the dashed curve corresponds to the native-risk signal without Hawkes scaling, while the solid curve corresponds to the Hawkes-scaled trading signal under the $q = 0.7$ event threshold. A Buy-and-Hold benchmark is also included for reference.

Table 5.2 further summarizes the BTC trading performance under three signal-generation settings: No-Hawkes, Hawkes-scaled with the $q = 0.7$ event threshold, and Hawkes-scaled with the $q = 0.9$ event threshold. The table reports cumulative return, Sharpe ratio, maximum drawdown, Calmar ratio, and win rate for all four forecasting models, together with the Buy-and-Hold baseline.

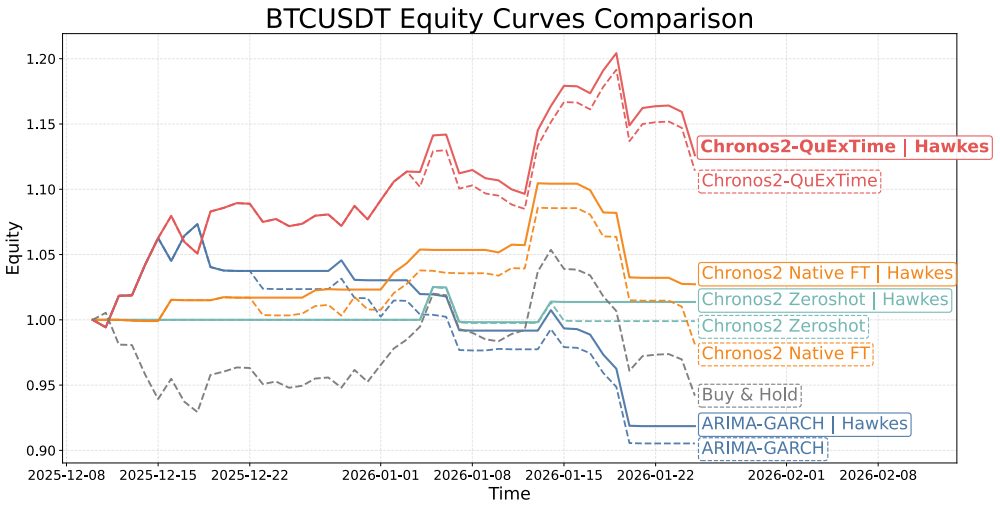


FIGURE 5.7: BTC trading equity curves across forecasting backbones. Dashed lines denote the native-risk trading signals without Hawkes scaling, while solid lines denote the Hawkes-scaled signals under the $q = 0.7$ event threshold. The Buy-and-Hold benchmark is shown for reference.

TABLE 5.2: BTC trading performance comparison under No-Hawkes and Hawkes-scaled settings. For the first four forecasting backbones, three signal-generation cases are reported: No-Hawkes, Hawkes with $q = 0.7$, and Hawkes with $q = 0.9$. The Buy-and-Hold benchmark is reported as a single reference column. Higher is better for CumRet, Sharpe, Calmar, and WinRate, while lower drawdown magnitude is preferred for MaxDD.

Metric	ARIMA–GARCH			Chronos2 (Zero-shot)			Chronos2 (Native FT)			Chronos2-QuExTime FT			Buy&Hold
	No-H	Q70	Q90	No-H	Q70	Q90	No-H	Q70	Q90	No-H	Q70	Q90	–
CumRet	-0.0947	**-0.0814**	-0.0817	-0.0010	**0.0137**	-0.0010	-0.0188	**0.0272**	0.0178	0.1142	**0.1261**↑	0.1142	-0.0584
Sharpe	-3.0987	-2.7234	**-2.6626**	-0.0093	**1.0182**	-0.0093	-0.5503	**1.1578**	0.7683	2.9931	**3.2925**↑	2.9931	-1.4136
MaxDD	-0.1565	**-0.1442**	-0.1444	-0.0269	**-0.0263**↑	-0.0269	-0.0964	**-0.0700**	-0.0700	-0.0649	**-0.0649**	-0.0649	-0.1060
Calmar	-3.4877	**-3.4013**	-3.4033	-0.2943	**4.3217**	-0.2943	-1.4501	**3.3950**	2.1449	20.9207	**24.1072**↑	20.9207	-3.5785
WinRate	0.2941	0.2963	**0.3030**	0.2857	**0.3333**	0.2857	0.3333	**0.3600**	0.3226	0.5870	**0.5870**↑	0.5870	0.4783

5.3.3.2 DOGE Trading Results

We then present the trading results for DOGE, which represents a more speculative and high-noise market regime with stronger tail fluctuations and endogenous excitation.

Figure 5.8 visualizes the DOGE equity curves under the four forecasting backbones. As in the BTC case, dashed lines correspond to native-risk signals without Hawkes scaling, while solid lines correspond to Hawkes-scaled signals under the $q = 0.7$ event threshold. A Buy-and-Hold benchmark is included for reference.

Table 5.3 summarizes the trading performance across three signal-generation settings: No-Hawkes, Hawkes with $q = 0.7$, and Hawkes with $q = 0.9$. The same set of performance metrics is reported to allow direct comparison with the BTC case.

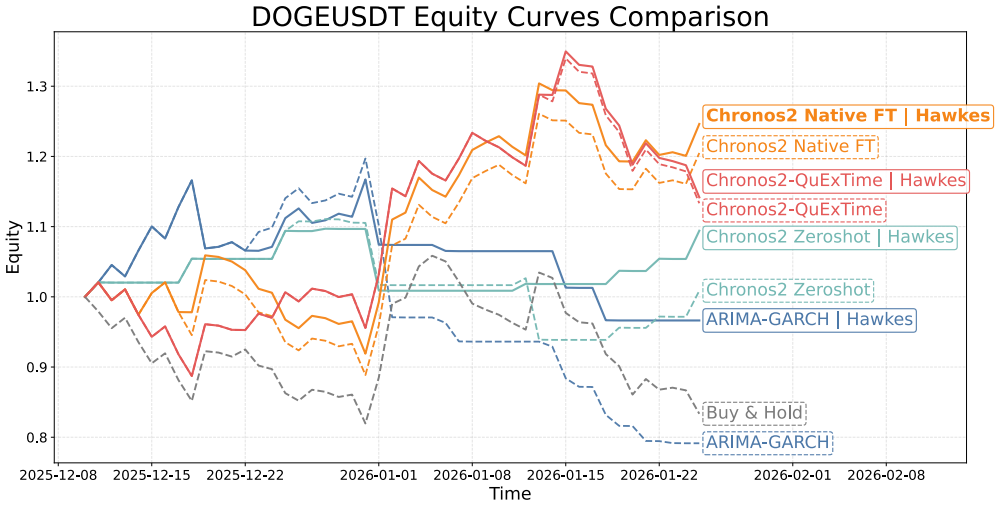


FIGURE 5.8: DOGE trading equity curves across forecasting backbones. Dashed lines denote native-risk trading signals, while solid lines denote Hawkes-scaled signals under the $q = 0.7$ event threshold. The Buy-and-Hold benchmark is shown for comparison.

TABLE 5.3: DOGE trading performance comparison under No-Hawkes and Hawkes-scaled settings. For each forecasting backbone, the best result within each model block is highlighted in bold.

Metric	ARIMA–GARCH			Chronos2 (Zero-shot)			Chronos2 (Native FT)			Chronos2-QuExTime FT			Buy&Hold
	No-H	Q70	Q90	No-H	Q70	Q90	No-H	Q70	Q90	No-H	Q70	Q90	–
CumRet	-0.2086	**-0.0337**	-0.2225	0.0087	**0.0942**	-0.0037	0.2049	**0.2462**↑	0.2462	0.1333	**0.1417**	0.1333	-0.1668
Sharpe	-2.7674	**-0.3177**	-3.0335	0.3741	**2.5067**	0.1229	2.5364	**2.9714**↑	2.9714	1.7599	**1.8457**	1.7599	-1.7437
MaxDD	-0.3388	**-0.1721**	-0.3339	-0.1549	**-0.0805**↑	-0.1549	-0.1322	**-0.1322**	-0.1322	-0.1540	**-0.1540**	-0.1540	-0.2126
Calmar	-2.4901	**-1.3824**	-2.5884	0.4590	**12.9444**	-0.1880	25.6366	**35.8240**↑	35.8240	11.0293	**12.0937**	11.0293	-3.5974
WinRate	0.3947	**0.4828**↑	0.3784	0.4286	**0.4706**	0.4000	0.4130	**0.4130**	0.4130	0.3478	**0.3478**	0.3478	0.3043

Chapter 6

Discussion

The Chapter 5 forecasting and trading results reveal a clear relationship between forecasting quality, risk modeling, and realized profitability, while also highlighting important regime-dependent effects.

For BTC, the results provide strong evidence supporting the proposed framework. As shown in both Table 5.1 and Table 5.2, the proposed loss consistently improves tail-risk calibration and directional metrics (Rank IC and Sign Accuracy) at the forecasting level. These improvements directly translate into superior trading performance, where the *Proposed FT* model achieves the highest cumulative return, Sharpe ratio, and Calmar ratio. Furthermore, the introduction of Hawkes-based risk scaling leads to additional performance gains within each model, with the best overall result achieved by the combination of *Proposed FT + Hawkes*. This demonstrates that improving uncertainty modeling at the forecasting stage can effectively enhance downstream trading decisions in relatively stable and liquid cryptocurrency markets.

In contrast, the results on DOGE exhibit a different behavior. As observed in Table 5.1, both fine-tuned variants show degraded tail-risk metrics compared to the zero-shot model, and the proposed loss does not improve directional metrics relative to the native fine-tuning baseline. This is reflected in the trading results (Table 5.3), where the *Native FT* model achieves the highest returns, while the proposed loss does not lead to further improvement. These observations suggest that, in highly volatile and noise-dominated cryptocurrency markets, emphasizing tail penalties may be less effective and can lead to unstable uncertainty estimates.

Importantly, across both BTC and DOGE, Hawkes-based risk scaling consistently improves trading performance within the same forecasting backbone. This indicates that

modeling market endogenous excitation provides an additional and complementary source of risk information that is not captured by the forecasting model alone.

These results highlight a regime-dependent effect: while the proposed loss enhances forecasting and trading performance in structured markets such as BTC, its effectiveness is reduced in noisier environments such as DOGE. At the same time, the Hawkes module demonstrates robust benefits across both regimes, suggesting that it serves as a general-purpose risk adjustment layer for trading signal construction.

Chapter 7

Conclusion and future work

7.1 Conclusion

This thesis investigates the relationship between probabilistic forecasting, risk modeling, and trading performance in cryptocurrency markets through a unified two-layer framework.

At the forecasting layer, we study the effectiveness of quantile-based time-series foundation models under cryptocurrency-specific market conditions. The proposed tail-sensitive loss demonstrates clear benefits in relatively stable markets such as BTC, where it improves uncertainty calibration and enhances directional signal quality without degrading traditional error metrics. However, experiments on highly volatile assets such as DOGE reveal a notable performance deterioration, indicating that uncertainty modeling in cryptocurrency markets is strongly regime-dependent. This result suggests that improving probabilistic forecasting is not only a modeling problem but also a data-regime adaptation problem.

At the trading layer, we propose a unified risk-adjusted signal construction framework that bridges forecasting outputs and executable trading decisions. By explicitly separating predicted return, model-native uncertainty, and endogenous market excitation, the framework enables consistent comparison across heterogeneous forecasting models. Empirical results show that improvements in uncertainty modeling at the forecasting stage can translate into better trading performance in structured markets.

In addition, we introduce a Hawkes-based excitation layer to capture endogenous market dynamics. The results demonstrate that incorporating market excitation as an external risk signal consistently improves trading performance across different models and market regimes. This finding highlights that predictive uncertainty alone is insufficient to fully

characterize risk in cryptocurrency markets, and that endogenous dynamics provide an important complementary source of information.

Overall, this work shows that effective cryptocurrency trading systems require not only accurate return prediction, but also robust uncertainty modeling and explicit modeling of market structure. The proposed framework provides a unified perspective for integrating these components into a coherent pipeline.

7.2 Future Work

The experimental results of this thesis reveal a fundamental limitation of the current pipeline:

$$\textit{forecast model} \rightarrow \textit{uncertainty} \rightarrow \textit{Hawkes} \rightarrow \textit{trading}$$

While this pipeline performs well in relatively stable market regimes such as BTC, its effectiveness degrades in highly volatile and noise-dominated environments such as DOGE. This discrepancy suggests that the underlying challenge is not merely model performance, but rather the presence of *distribution shift* across different market regimes. Under such non-stationary conditions, both predictive models and uncertainty estimates may become misaligned with the true data-generating process.

To address this issue, future research can be broadly categorized into two complementary directions, depending on whether adaptation is performed *inside* the forecasting model or *outside* through uncertainty correction.

Adaptive models: learning to adjust the forecasting process The first direction is to introduce *adaptive learning mechanisms* that allow the model itself to adjust its behavior according to the input data. The key idea is to learn a mapping from the current market state to model configurations, such as selecting different experts, adjusting loss functions, or modifying training dynamics.

This perspective is closely related to several existing research lines. Mixture-of-experts and ensemble-based approaches dynamically select or weight models based on input features. In the context of non-stationary time series, concept-drift adaptation methods explicitly model distribution changes and update models accordingly [39]. More recent work such as OneNet demonstrates that online ensembling with adaptive weighting can effectively handle regime shifts in time-series forecasting [40]. In addition, meta-learning

approaches enable models to rapidly adapt to new data regimes by learning how to update themselves [41].

From a control perspective, reinforcement learning has also been explored as a mechanism for dynamically selecting model combinations or updating strategies in non-stationary environments [42]. Related work on learning-to-teach and automatic loss scheduling further suggests that training objectives themselves can be adaptively optimized based on model performance [43, 44].

In the context of this thesis, these approaches suggest a promising extension: instead of using a fixed loss or fixed forecasting model, one could design a *policy network* that dynamically adjusts model behavior based on market conditions, such as volatility regime, forecast reliability, or recent prediction errors. This would enable the system to automatically adapt to different assets and market states, addressing the regime-dependent performance observed between BTC and DOGE.

External calibration: generalizable uncertainty correction layers The second direction is to treat the forecasting model as a fixed component and introduce a *model-agnostic uncertainty correction layer* that improves predictive reliability without modifying the underlying model.

This perspective is supported by a growing body of work on probabilistic calibration. Conformalized Quantile Regression (CQR) shows that any model producing quantile forecasts can be augmented with distribution-free calibration guarantees [45]. More generally, calibrated regression methods demonstrate that uncertainty estimates from arbitrary regression models can be corrected through post-hoc procedures [46].

Recent work further extends this idea by integrating calibration directly into the training process. For example, Quantile Recalibration Training (QRT) introduces a unified framework that incorporates calibration objectives into model optimization [47], while large-scale empirical studies show that neural regression models often suffer from systematic miscalibration if such corrections are not applied [48].

Beyond static calibration, an important extension is to handle *distribution shift* explicitly. Adaptive conformal inference methods provide mechanisms to maintain coverage guarantees under changing data distributions [49]. For time-series settings, adaptive conformal prediction [50] and sequential predictive conformal inference [51] allow uncertainty estimates to be updated online as new data arrive. More recent work also explores separating different sources of uncertainty, such as aleatoric and epistemic components, to improve calibration quality [52].

These developments suggest that uncertainty modeling can be treated as a *modular layer* that operates independently of the forecasting architecture. In the context of this thesis, this implies that the proposed loss-based approach could be extended into a more general uncertainty correction module, combining train-time objectives, post-hoc calibration, and online adaptation to improve robustness across different market regimes.

Toward unified adaptive trading systems The two directions above represent complementary strategies for handling distribution shift. Adaptive models modify the internal behavior of the forecasting system, while external calibration layers adjust the reliability of its outputs. A promising future direction is to combine these approaches into a unified framework in which both model behavior and uncertainty estimation are dynamically adapted.

Such a system would enable end-to-end adaptation across the entire pipeline, allowing forecasting, uncertainty modeling, and trading decisions to co-evolve under changing market conditions. This represents an important step toward building robust and self-adaptive quantitative trading systems for highly non-stationary environments such as cryptocurrency markets.

Chapter 8

Ethics and Data Management

8.1 Ethical Considerations

I acknowledge that this thesis adheres to the ethical code[1] and research data management policies[2] of the University of Amsterdam (UvA) and the Informatics Institute (IvI).

The research focuses on financial time-series forecasting and trading strategy construction using cryptocurrency market data. All data used are publicly available and do not contain personal, sensitive, or identifiable information. Therefore, no ethical risks related to privacy, consent, or human subjects are involved.

However, several broader ethical considerations are relevant:

- **Financial risk and responsible use.** The proposed models are capable of generating trading signals. Misuse of such models in real financial markets may lead to financial losses. The results of this thesis are intended for research purposes only and should not be interpreted as financial advice.
- **Market impact and systemic effects.** The use of Hawkes processes highlights endogenous market dynamics, where trading activity can amplify volatility. Large-scale deployment of automated trading systems may contribute to feedback loops and increased market instability, which raises broader systemic considerations.
- **Reproducibility and transparency.** All experiments are conducted under strictly time-ordered splits to avoid data leakage. The full pipeline, including data collection, forecasting, and trading components, is documented and publicly available to ensure transparency and reproducibility.

[1] https://student.uva.nl/en/topics/ethics-in-research
[2] https://rdm.uva.nl/en

8.2 Research Data Management

The data used in this thesis consist of historical cryptocurrency market data collected from the Binance exchange and processed into log-return time series.

The raw data are collected using a custom-built data infrastructure system (ChomoSyncer), which streams real-time tick-level data from Binance. The collected data are archived externally and transformed into structured datasets for forecasting and trading experiments.

The experimental pipeline consists of four main components: (i) data collection and storage, (ii) forecasting models (including Chronos2 and the proposed loss-enhanced variant), (iii) forecasting experiment pipelines, and (iv) trading and backtesting with Hawkes-based risk scaling.

The following table lists all data sources and code repositories used in this thesis. I confirm that the list is complete and that these resources are sufficient to reproduce the results presented in this work.

Short description	**Availability**	**License**
Cryptocurrency data and collection system (ChomoSyncer, including BTC/DOGE Kline dataset)	https://github.com/GeekChomolungma/ChomoSyncer	MIT
Chronos2 (fork) + proposed loss implementation	https://github.com/HarvestStars/chronos-forecasting/tree/dev	Apache 2.0
Crypto forecasting pipeline	https://github.com/GeekChomolungma/Crypto_Forecast	MIT
Trading and backtesting framework (Hawkes Bench)	https://github.com/GeekChomolungma/Hawkes_Bench	MIT

TABLE 8.1: Data and code used in this thesis.

All datasets used in this study are either publicly available or derived from publicly accessible sources. No proprietary or restricted data are used.

The provided data and code collectively implement the full pipeline—from raw data acquisition to forecasting, signal construction, and trading evaluation—ensuring that all experimental results in Chapter 5 can be reproduced.

Chapter 9

Appendix A: Data Infrastructure Implementation

This appendix provides implementation details of the market data infrastructure used in this thesis. The system is built around a custom data synchronization tool called *ChomoSyncer*, which continuously collects cryptocurrency market data from Binance and distributes it to downstream modeling components.

9.1 System Overview

The market data infrastructure is designed as a streaming data pipeline connecting the Binance exchange with downstream machine learning and econometric models.

The system performs three primary tasks:

- Continuous acquisition of real-time candlestick (Kline) market data
- Streaming distribution of data for low-latency model consumption
- Persistent storage of historical data for offline experiments and backtesting

Figure 9.1 illustrates the overall data flow architecture.

The infrastructure enables a unified data pipeline that supports both offline research and potential real-time deployment scenarios.

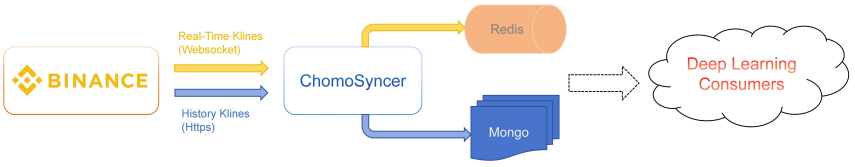


FIGURE 9.1: Market data infrastructure based on ChomoSyncer.

9.2 Data Acquisition from Binance

Market data are obtained from the Binance exchange using both WebSocket and REST APIs.

WebSocket streams provide real-time candlestick updates, allowing the system to continuously track market activity with minimal latency. In parallel, REST interfaces are used for historical backfilling and gap recovery to ensure completeness of the dataset.

The synchronization component supports multiple cryptocurrency symbols and time resolutions simultaneously, enabling large-scale data collection across different markets.

9.3 Redis Streaming Layer

After acquisition, market data are published into a Redis-based streaming layer. Redis acts as a lightweight message broker that allows downstream components to consume market data in real time.

This design decouples data producers from model consumers, enabling multiple models to access the same data stream simultaneously. For example, forecasting models, event-detection modules, and trading signal generators can operate independently while sharing the same market data source.

9.4 MongoDB Persistent Storage

In addition to real-time streaming, all market data are stored in MongoDB for long-term persistence. This persistent storage layer allows the reconstruction of historical datasets for training, validation, and backtesting.

MongoDB was selected due to its flexible document schema and efficient support for time-series data storage. Each candlestick entry contains the timestamp, open, high, low, close, and volume information, enabling complete reconstruction of historical market states.

Chapter 10

Appendix B: Mathematical Details

This appendix provides the mathematical formulations underlying the forecasting models, loss design, Hawkes excitation model, and signal construction used in the proposed framework. These derivations complement the high-level methodological descriptions provided in Sections 4.3–4.5.

10.1 ARIMA Mean Model Formulation

The conditional mean of the return process is modeled using an ARIMA model. In lag-polynomial form, the model can be written as

$$\Phi(L)(1-L)^d r_t = \Theta(L)\varepsilon_t, \tag{10.1}$$

where L denotes the lag operator, d represents the differencing order, and $\Phi(L)$ and $\Theta(L)$ are the autoregressive and moving-average polynomials defined as

$$\Phi(L) = 1 - \phi_1 L - \cdots - \phi_p L^p, \tag{10.2}$$

$$\Theta(L) = 1 + \theta_1 L + \cdots + \theta_q L^q. \tag{10.3}$$

The resulting model captures linear dependencies in the return series and produces the conditional mean forecast

$$\hat{\mu}_{t+1} = \mathbb{E}[r_{t+1} \mid \mathcal{F}_t], \tag{10.4}$$

10.2 GARCH Volatility Dynamics

The volatility dynamics of the residual process are modeled using a GARCH process. For the commonly used GARCH(1,1) specification, the conditional variance evolves according to

$$\sigma_t^2 = \omega + \alpha\varepsilon_{t-1}^2 + \beta\sigma_{t-1}^2, \tag{10.5}$$

The parameters ω, α, and β determine the baseline volatility, the impact of recent shocks, and the persistence of volatility respectively.

The resulting conditional variance forecast

$$\hat{\sigma}_{t+1}^2 = \mathbb{E}[\sigma_{t+1}^2 \mid \mathcal{F}_t], \tag{10.6}$$

serves as the volatility-based risk estimate used in the white-box forecasting model.

10.3 QuExTime Loss Implementation Sketch

For completeness, we provide a simplified sketch of the implementation:

```
quantile_weights = 1 + gamma * (2 * abs(quantiles - 0.5)) ** alpha
time_weights = delta ** steps
time_weights = time_weights / time_weights.mean()

weighted_mask = loss_mask * quantile_weights * time_weights
weighted_loss = quantile_loss * weighted_mask

normalizer = weighted_mask.sum(...).clamp_min(1e-8)
loss = (weighted_loss.sum(...) / normalizer).mean()
```

This implementation directly matches the mathematical formulation above: extreme quantiles receive larger weights through `quantile_weights`, near-term horizons receive

larger weights through `time_weights`, and the final loss is obtained by normalized weighted aggregation.

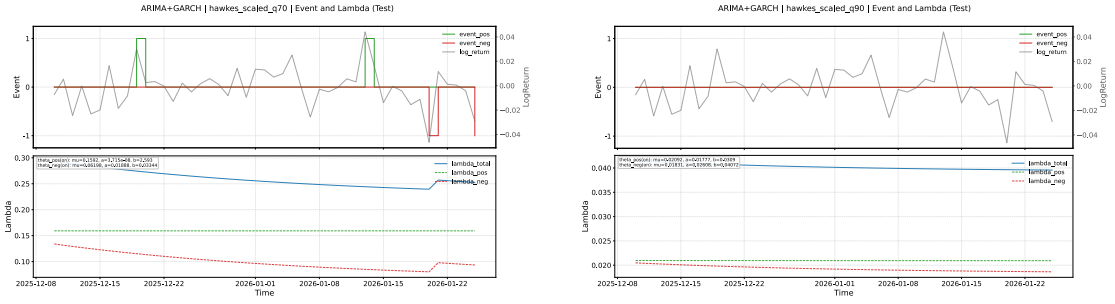
Chapter 11

Appendix C: Trading Visualization

This appendix provides additional visualizations to illustrate intermediate steps of the proposed trading framework, including eventization, Hawkes intensity estimation, and trading signal execution. These figures complement the main results presented in Section 5.3 and are intended to provide further insights into the mechanisms of the proposed approach.

11.1 Eventization and Hawkes Intensity (BTC)

11.1.1 ARIMA–GARCH

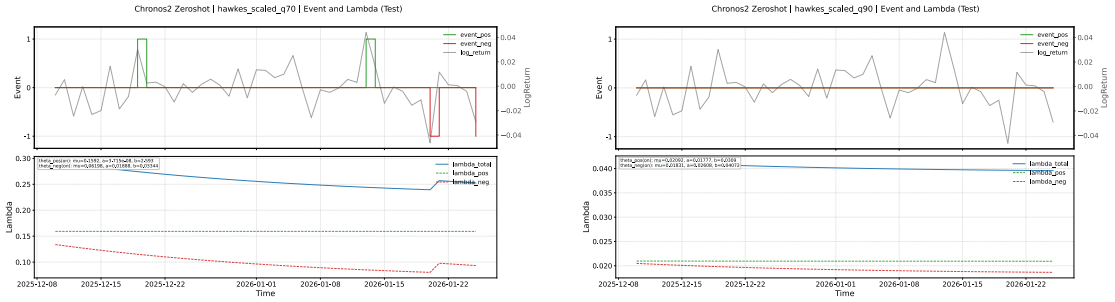


(A) $q = 0.7$ threshold

(B) $q = 0.9$ threshold

FIGURE 11.1: Eventization and Hawkes intensity for BTC under ARIMA–GARCH. The upper panel shows eventized log returns, while the lower panel shows the corresponding Hawkes intensity $\lambda(t)$.

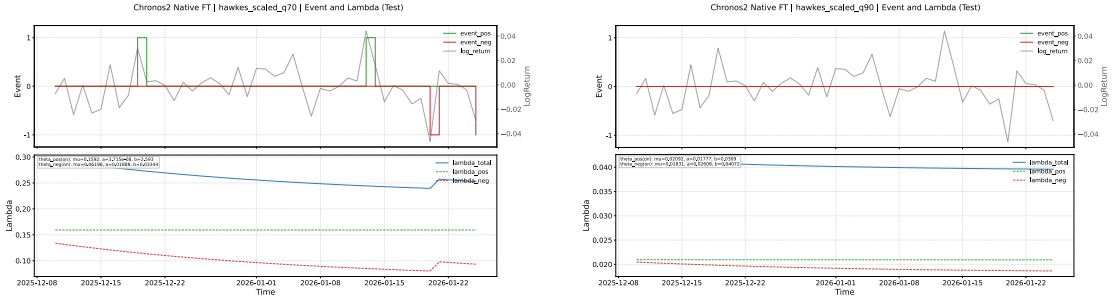
11.1.2 Chronos2 (Zero-shot)



(A) $q = 0.7$ threshold (B) $q = 0.9$ threshold

FIGURE 11.2: Eventization and Hawkes intensity for BTC under Chronos2 (Zero-shot).

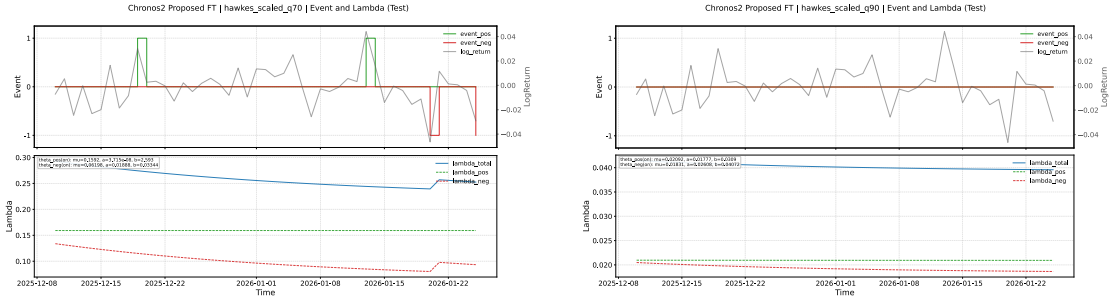
11.1.3 Chronos2 (Native FT)



(A) $q = 0.7$ threshold (B) $q = 0.9$ threshold

FIGURE 11.3: Eventization and Hawkes intensity for BTC under Chronos2 (Native FT).

11.1.4 Chronos2 (Proposed FT)

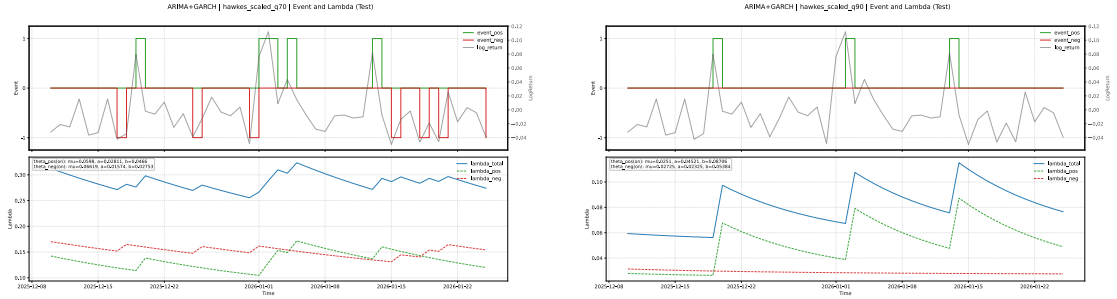


(A) $q = 0.7$ threshold (B) $q = 0.9$ threshold

FIGURE 11.4: Eventization and Hawkes intensity for BTC under Chronos2 (Proposed FT).

11.2 Eventization and Hawkes Intensity (DOGE)

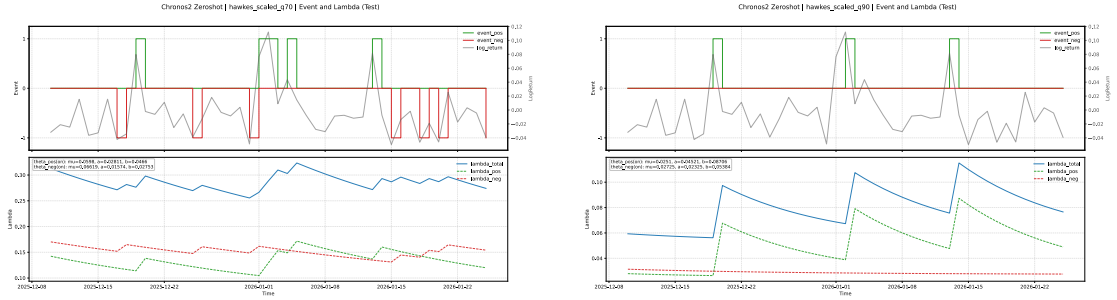
11.2.1 ARIMA–GARCH



(A) $q = 0.7$ threshold (B) $q = 0.9$ threshold

FIGURE 11.5: Eventization and Hawkes intensity for DOGE under ARIMA–GARCH. The upper panel shows eventized log returns, while the lower panel shows the corresponding Hawkes intensity $\lambda(t)$.

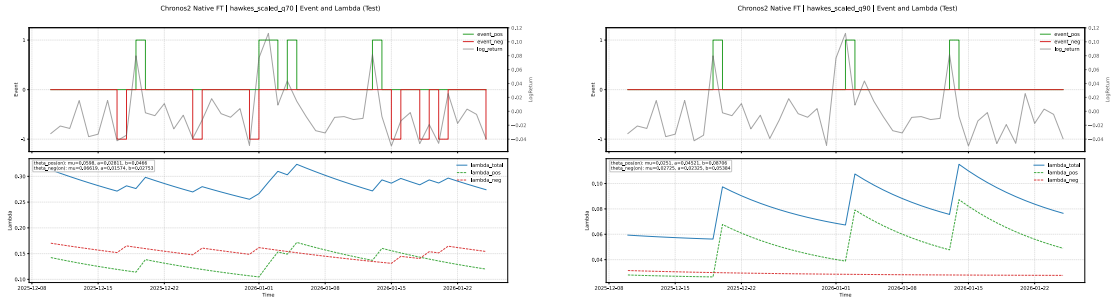
11.2.2 Chronos2 (Zero-shot)



(A) $q = 0.7$ threshold (B) $q = 0.9$ threshold

FIGURE 11.6: Eventization and Hawkes intensity for DOGE under Chronos2 (Zero-shot).

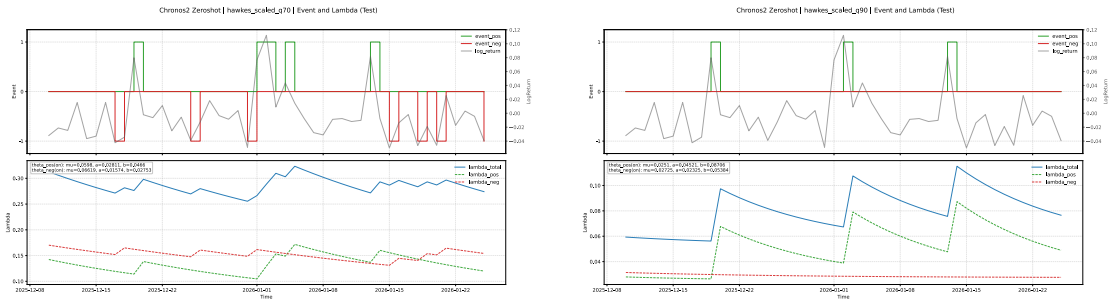
11.2.3 Chronos2 (Native FT)



(A) $q = 0.7$ threshold (B) $q = 0.9$ threshold

FIGURE 11.7: Eventization and Hawkes intensity for DOGE under Chronos2 (Native FT).

11.2.4 Chronos2 (Proposed FT)



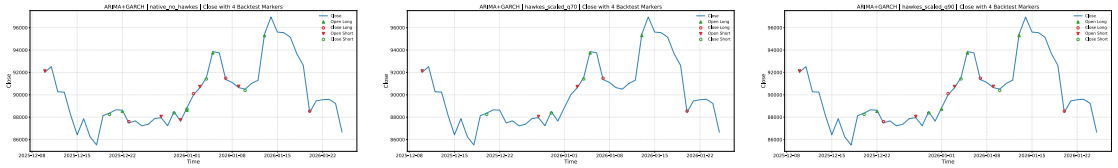
(A) $q = 0.7$ threshold

(B) $q = 0.9$ threshold

FIGURE 11.8: Eventization and Hawkes intensity for DOGE under Chronos2 (Proposed FT).

11.3 Trading Signal Execution (BTC)

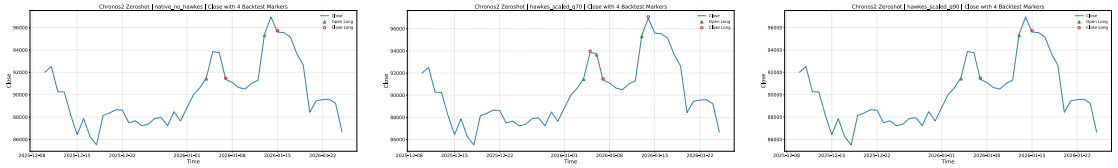
11.3.1 ARIMA–GARCH



(A) No-Hawkes (B) $q = 0.7$ (C) $q = 0.9$

FIGURE 11.9: Trading signal execution for BTC under ARIMA–GARCH. Markers indicate long/short entry and exit points under different signal-generation settings.

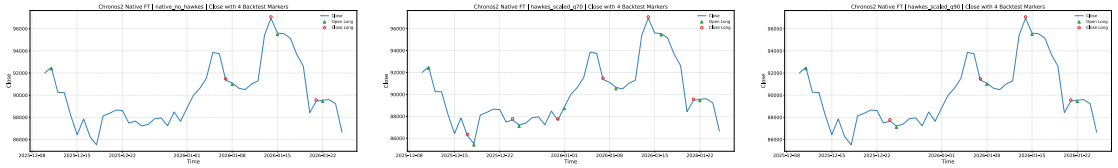
11.3.2 Chronos2 (Zero-shot)



(A) No-Hawkes (B) $q = 0.7$ (C) $q = 0.9$

FIGURE 11.10: Trading signal execution for BTC under Chronos2 (Zero-shot).

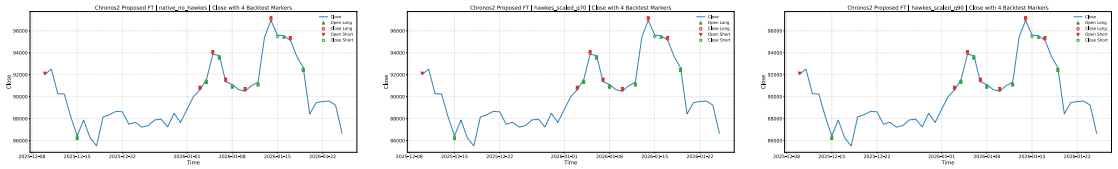
11.3.3 Chronos2 (Native FT)



(A) No-Hawkes (B) $q = 0.7$ (C) $q = 0.9$

FIGURE 11.11: Trading signal execution for BTC under Chronos2 (Native FT).

11.3.4 Chronos2 (Proposed FT)

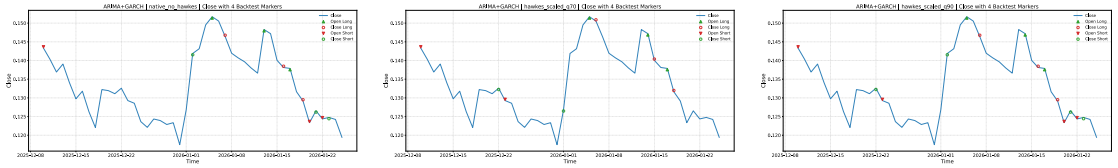


(A) No-Hawkes (B) $q = 0.7$ (C) $q = 0.9$

FIGURE 11.12: Trading signal execution for BTC under Chronos2 (Proposed FT).

11.4 Trading Signal Execution (DOGE)

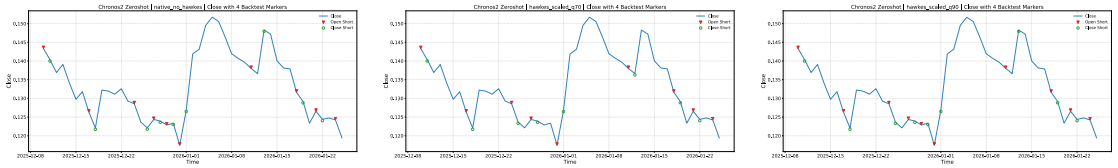
11.4.1 ARIMA–GARCH



(A) No-Hawkes (B) $q = 0.7$ (C) $q = 0.9$

FIGURE 11.13: Trading signal execution for DOGE under ARIMA–GARCH. Markers indicate long/short entry and exit points under different signal-generation settings.

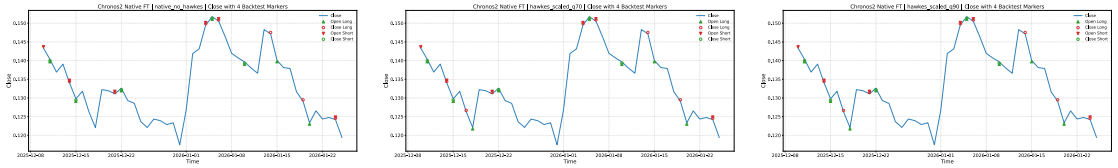
11.4.2 Chronos2 (Zero-shot)



(A) No-Hawkes (B) $q = 0.7$ (C) $q = 0.9$

FIGURE 11.14: Trading signal execution for DOGE under Chronos2 (Zero-shot).

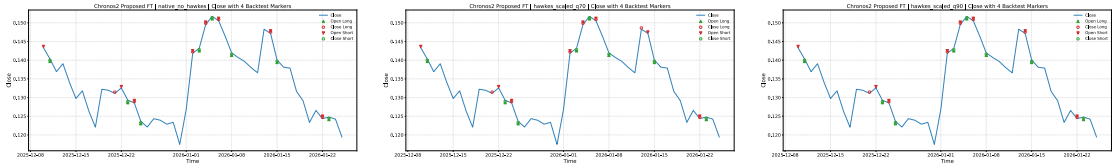
11.4.3 Chronos2 (Native FT)



(A) No-Hawkes (B) $q = 0.7$ (C) $q = 0.9$

FIGURE 11.15: Trading signal execution for DOGE under Chronos2 (Native FT).

11.4.4 Chronos2 (Proposed FT)



(A) No-Hawkes (B) $q = 0.7$ (C) $q = 0.9$

FIGURE 11.16: Trading signal execution for DOGE under Chronos2 (Proposed FT).

Bibliography

[1] Harry Markowitz. Portfolio selection. *Journal of Finance*, 1952.

[2] William F. Sharpe. Mutual fund performance. *Journal of Business*, 1966.

[3] George E. P. Box and Gwilym M. Jenkins. *Time Series Analysis: Forecasting and Control*. Holden-Day, San Francisco, 1970.

[4] Robert F. Engle. Autoregressive conditional heteroscedasticity with estimates of the variance of united kingdom inflation. *Econometrica*, 50(4):987–1007, 1982.

[5] Tim Bollerslev. Generalized autoregressive conditional heteroskedasticity. *Journal of Econometrics*, 31(3):307–327, 1986.

[6] Thomas Fischer and Christopher Krauss. Deep learning with long short-term memory networks for financial market predictions. *European Journal of Operational Research*, 270:654–669, 2018.

[7] Ozan B. Sezer, M. Ugur Gudelek, and A. Murat Ozbayoglu. Financial time series forecasting with deep learning: A systematic literature review: 2005-2019. *Applied Soft Computing*, 90, 2020.

[8] Roger Koenker and Gilbert Bassett. Regression quantiles. *Econometrica*, 46(1): 33–50, 1978.

[9] Tilmann Gneiting and Adrian Raftery. Strictly proper scoring rules, prediction, and estimation. *Journal of the American Statistical Association*, 102(477):359–378, 2007.

[10] Abdul Fatir Ansari, Lorenzo Stella, Caner Turkmen, Xiyuan Zhang, Pedro Mercado, Huibin Shen, Oleksandr Shchur, Syama Sundar Rangapuram, Sebastian Pineda Arango, Shubham Kapoor, et al. Chronos: Learning the language of time series. *arXiv preprint arXiv:2403.07815*, 2024.

[11] James D. Hamilton. *Time Series Analysis*. Princeton University Press, Princeton, NJ, 1994.

[12] Rama Cont. Empirical properties of asset returns: stylized facts and statistical issues. *Quantitative Finance*, 1(2):223–236, 2001.

[13] Aurelio F. Bariviera. Some stylized facts of the bitcoin market. *Physica A: Statistical Mechanics and its Applications*, 484:82–90, 2017.

[14] Andrew Phillip, Jennifer Chan, and Shelton Peiris. A new look at cryptocurrencies. *Economics Letters*, 163:6–9, 2018.

[15] Dirk Baur, KiHoon Hong, and Adrian Lee. Bitcoin: Medium of exchange or speculative assets? *Journal of International Financial Markets, Institutions and Money*, 54:177–189, 2018.

[16] Ruey S. Tsay. *Analysis of Financial Time Series*. John Wiley & Sons, Hoboken, NJ, 3 edition, 2010.

[17] Paraskevi Katsiampa. Volatility estimation for bitcoin: A comparison of garch models. *Economics Letters*, 158:3–6, 2017.

[18] Jeffrey Chu, Stephen Chan, Saralees Nadarajah, and Joerg Osterrieder. Garch modelling of cryptocurrencies. *Journal of Risk and Financial Management*, 10(4): 17, 2017.

[19] Bryan Lim and Stefan Zohren. Time-series forecasting with deep learning: A survey. *Philosophical Transactions of the Royal Society A*, 379(2194):20200209, 2021.

[20] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997.

[21] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

[22] Qingsong Wen, Tian Zhou, Chaoli Zhang, Weiqi Chen, Ziqing Ma, Junchi Yan, and Liang Sun. Transformers in time series: A survey. *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence*, pages 6778–6786, 2023.

[23] Yuxuan Liang, Haomin Wen, Yuqi Nie, Yushan Jiang, Ming Jin, Dongjin Song, Shirui Pan, and Qingsong Wen. Foundation models for time series analysis: A tutorial and survey. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 6555–6565, 2024.

[24] Siva Rama Krishna Kottapalli, Karthik Hubli, Sandeep Chandrashekhara, Garima Jain, Sunayana Hubli, Gayathri Botla, and Ramesh Doddaiah. Foundation models for time series: A survey. *arXiv preprint arXiv:2504.04011*, 2025.

[25] John A. Miller, Mohammed Aldosari, Farah Saeed, Nasid Habib Barna, Subas Rana, I. Budak Arpinar, and Ninghao Liu. A survey of deep learning and foundation models for time series forecasting. *arXiv preprint arXiv:2401.13912*, 2024.

[26] Azul Garza and Max Mergenthaler-Canseco. Timegpt-1. *arXiv preprint arXiv:2310.03589*, 2023.

[27] Abhimanyu Das, Weihao Kong, Rajat Sen, and Yichen Zhou. A decoder-only foundation model for time-series forecasting. *arXiv preprint arXiv:2310.10688*, 2024.

[28] Gerald Woo, Chenghao Liu, Akshat Kumar, Caiming Xiong, Silvio Savarese, and Doyen Sahoo. Unified training of universal time series forecasting transformers. *arXiv preprint arXiv:2402.02592*, 2024.

[29] Kashif Rasul, Arjun Ashok, Andrew Robert Williams, Arian Khorasani, George Adamopoulos, Rishika Bhagwatkar, Marin Biloš, Hena Ghonia, Nadhir Vincent Hassen, Anderson Schneider, et al. Lag-llama: Towards foundation models for time series forecasting. *arXiv preprint arXiv:2310.08278*, 2023.

[30] Abdul Fatir Ansari, Oleksandr Shchur, Jaris Küken, Andreas Auer, Boran Han, Pedro Mercado, Syama Sundar Rangapuram, Huibin Shen, Lorenzo Stella, Xiyuan Zhang, Mononito Goswami, Shubham Kapoor, Danielle C. Maddix, Pablo Guerron, Tony Hu, Junming Yin, Nick Erickson, Prateek Mutalik Desai, Hao Wang, Huzefa Rangwala, George Karypis, Yuyang Wang, and Michael Bohlke-Schneider. Chronos-2: From univariate to universal forecasting. *arXiv preprint arXiv:2510.15821*, 2025.

[31] Amazon. amazon/chronos-2. Hugging Face model card, 2025. Accessed March 2026.

[32] Amazon Science. amazon-science/chronos-forecasting. GitHub repository, 2025. Accessed March 2026.

[33] Alexander J. McNeil, Rüdiger Frey, and Paul Embrechts. *Quantitative Risk Management: Concepts, Techniques and Tools*. Princeton University Press, 2015.

[34] Alan G. Hawkes. Spectra of some self-exciting and mutually exciting point processes. *Biometrika*, 1971.

[35] Emmanuel Bacry, Iacopo Mastromatteo, and Jean-Francois Muzy. Hawkes processes in finance. *Market Microstructure and Liquidity*, 2015.

[36] Vladimir Filimonov and Didier Sornette. Quantifying reflexivity in financial markets: Toward a prediction of flash crashes. *Physical Review E*, 2012.

[37] Spencer Wheatley, Didier Sornette, and Tobias Huber. Bitcoin market dynamics: The role of self-excitation. *Scientific Reports*, 9:17287, 2019.

[38] Philippe Jorion. *Value at Risk: The New Benchmark for Managing Financial Risk*. McGraw-Hill, 2007.

[39] João Gama, Indrė Žliobaitė, Albert Bifet, Mykola Pechenizkiy, and Abdelhamid Bouchachia. A survey on concept drift adaptation. *ACM Computing Surveys*, 46 (4):44:1–44:37, 2014. doi: 10.1145/2523813.

[40] Yifan Zhang, Qingsong Wen, Xue Wang, Weiqi Chen, Liang Sun, Zhang Zhang, Liang Wang, Rong Jin, and Tieniu Tan. Onenet: Enhancing time series forecasting models under concept drift by online ensembling. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

[41] Boris N. Oreshkin, Dmitri Carpov, Nicolas Chapados, and Yoshua Bengio. Meta-learning framework with applications to zero-shot time-series forecasting. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(10):8852–8860, 2021.

[42] Yuwei Fu, Di Wu, and Benoit Boulet. Reinforcement learning based dynamic model combination for time series forecasting. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(6):6639–6647, 2022. doi: 10.1609/aaai.v36i6.20618.

[43] Lijun Wu, Fei Tian, Yingce Xia, Yang Fan, Tao Qin, Jianhuang Lai, and Tie-Yan Liu. Learning to teach with dynamic loss functions. In *Advances in Neural Information Processing Systems*, volume 31, 2018.

[44] Haowen Xu, Hao Zhang, Zhiting Hu, Xiaodan Liang, Ruslan Salakhutdinov, and Eric P. Xing. Autoloss: Learning discrete schedules for alternate optimization. In *International Conference on Learning Representations*, 2019.

[45] Yaniv Romano, Evan Patterson, and Emmanuel J. Candès. Conformalized quantile regression. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

[46] Volodymyr Kuleshov, Nathan Fenner, and Stefano Ermon. Accurate uncertainties for deep learning using calibrated regression. In *International Conference on Machine Learning*, pages 2796–2804. PMLR, 2018.

[47] Victor Dheur and Souhaib Ben Taieb. Probabilistic calibration by design for neural network regression. In *Proceedings of The 27th International Conference on Artificial Intelligence and Statistics*, volume 238 of *Proceedings of Machine Learning Research*, pages 1426–1434. PMLR, 2024.

[48] Victor Dheur and Souhaib Ben Taieb. A large-scale study of probabilistic calibration in neural network regression. In *International Conference on Machine Learning*, pages 7635–7663. PMLR, 2023.

[49] Isaac Gibbs and Emmanuel J. Candès. Adaptive conformal inference under distribution shift. In *Advances in Neural Information Processing Systems*, volume 34, pages 1660–1672, 2021.

[50] Margaux Zaffran, Olivier Féron, Yannig Goude, Julie Josse, and Aymeric Dieuleveut. Adaptive conformal predictions for time series. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 25834–25866. PMLR, 2022.

[51] Chen Xu and Yao Xie. Sequential predictive conformal inference for time series. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 38707–38727. PMLR, 2023.

[52] Raphael Rossellini, Rina Foygel Barber, and Rebecca Willett. Integrating uncertainty awareness into conformalized quantile regression. In *Proceedings of The 27th International Conference on Artificial Intelligence and Statistics*, volume 238 of *Proceedings of Machine Learning Research*, pages 1540–1548. PMLR, 2024.