Latent Diffusion for Event Driven Asset Pricing

Anonymous ACL submission

Abstract

Event studies have garnered widespread attention in academic research due to their significant impact on asset prices, playing a crucial role in both risk management and the understanding of market dynamics. However, existing methods face notable challenges. One such issue is the lack of effective multimodal alignment schemes, with many approaches relying on discretizing time series data when aligning it with language modalities. This often leads to the loss of valuable information, as continuous frameworks are better suited for capturing the dynamic nature of market behavior and accurately tracking rapid shifts in asset prices, as demonstrated by extensive theoretical and empirical work. Additionally, these methods struggle to model the inherent randomness of financial systems. To address these challenges, we introduce a Multimodal Latent Diffusion model specifically designed for event-driven asset pricing. Our approach integrates textual representations of sudden events with financial time series in a continuous latent space, preserving subtle temporal variations and fully leveraging the rich semantic cues embedded in the event-related text. Through comprehensive experiments and case studies, we demonstrate that our method consistently enhances predictive accuracy for event-driven asset pricing, while also expanding practical applications for risk management.

1 Introduction

Fama's efficient market hypothesis reshaped finance (Fama, 1970). Since then, asset pricing has evolved from modeling stable risk - return relationships to deciphering how markets digest unforeseen shocks. The COVID pandemic offered a stark lesson (Scherf et al., 2022): when lockdown announcements triggered simultaneous selloffs in cruise stocks and surges in cloud computing, classical factor models failed to explain the speed and selectivity of repricing. These episodes expose a fundamental tension in modern finance—while markets exhibit structured behavior during calm periods, their responses to discrete events resemble complex signal-processing systems, decoding narratives from earnings calls, policy statements, and supply chain alerts.

Another recent example, as shown in Figure 1, highlights the importance of event-driven asset pricing is related to Nvidia. The U.S. government released export control regulations for artificial intelligence, introducing uncertainty about the future prospects of related tech companies. This uncertainty led to fluctuations in Nvidia's stock price. Following this, DeepSeek (Guo et al., 2025) launched an open-source model that achieved performance comparable to OpenAI's (Roumeliotis and Tselikas, 2023) model, with significantly lower computational requirements, which caused Nvidia's stock price to decline again. On the other hand, U.S. tech giants like Google and Amazon are poised to continue investing in AI, which is expected to drive up the demand for graphics cards. This shift in market sentiment subsequently fueled an increase in Nvidia's stock price. This series of events underscores the critical role of event-driven asset pricing, as it helps investors navigate the market's reaction to sudden, impactful events that can significantly influence asset values.

However, event-driven asset pricing is a challenging task due to the dual nature of financial information. Consider a pharmaceutical company's drug trial announcement: the text's semantic content (e.g., "statistically significant efficacy") must intertwine with the firm's historical return patterns to determine its pricing impact. Traditional approaches (Zhang and Skiena, 2010; Pagolu et al., 2016; Ma et al., 2023) address this duality through compartmentalization—event studies analyze text disclosures as binary signals, while time series models treat prices as autoregressive processes. This artificial separation creates what we term

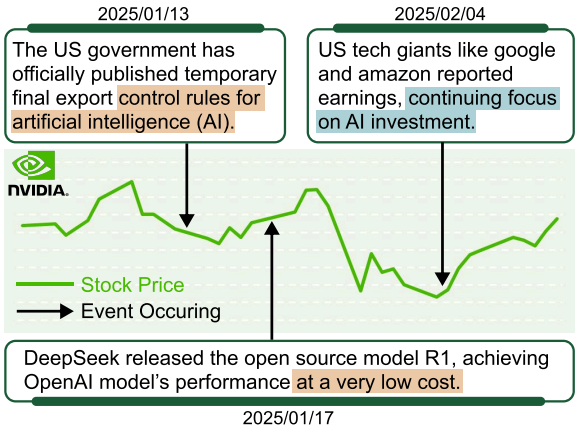

Figure 1: A recent example involving Nvidia illustrates that some sudden events can impact company pricing.

the *interpretation gap*: models either capture the *what* of market reactions (magnitude/direction) or the *why* (causal drivers), but seldom both. Recent advances in multimodal learning (Yuan et al., 2025) initially seemed promising, but empirically underperformed. The root issue traces to a mismatch in uncertainty handling: language models output deterministic embeddings that suppress textual ambiguity, while asset returns inherently embody stochasticity from incomplete information.

To address the aforementioned challenges, in this paper, we propose a latent diffusion based asset pricing framework. Our framework introduces three pivotal advances in bridging modern machine learning with financial theory.

- Our method aligns the temporal and linguistic modalities in a continuous latent space, transforming the asset pricing task into a conditional generation task. Our approach addresses the issue of ineffective alignment in previous works.
- The stochastic diffusion process in our framework explicitly models financial systems' inherent randomness, adapting to both structured market regimes and crisis-period discontinuities.
- Through extensive comparisons with numerous baselines, our model proves to be exceptionally robust. Additionally, we have experimentally verified that our method can be well-compatible with classic asset pricing theories, offering strong interpretability.

2 Related Work

Single-modal methods often fail to encapsulate the full scope of challenges present in financial tasks (Chen et al., 2023; Yu et al., 2023), particularly those involving both time series data and textual information. The recent development of Large Language Models (LLMs) has brought about a transformative moment in time series analysis, facilitating the integration of natural language with numerical data (Yu et al., 2023; Li et al., 2024). Some recent LLM-based methods have incorporated endogenous text derived from numerical data such as linguistic descriptions of statistical information (Gruver et al., 2024; Jin et al., 2023; Cao et al., 2023; Liu et al., 2024b; Sun et al., 2023; Liu et al., 2024c). Beyond relying on the capabilities of LLMs, there have also been efforts to design multimodal forecasting models (Xu et al., 2024; Liu et al., 2024a). Rather than using raw text data, these models typically combine LLM-derived text embeddings with time series using mechanisms such as cross-attention (Xu et al., 2024) or integrating separate forecasting outputs from both modalities (Liu et al., 2024a). Rather than relying on a frozen LLM to generate the textual embeddings, Kim et al. (2024) proposed a hybrid model capable of learning textual embeddings and supported event forecasting. However, discretizing time series data risks losing valuable information. Some works have attempted to mitigate it by plotting time series into charts (Hao et al., 2024; Daswani et al., 2024) to ensure continuity between modalities. Our method integrates textual representations of events with time series data within a shared latent space, preserving subtle temporal variations while fully leveraging the rich semantic cues embedded in the event-related text.

3 Methodology

3.1 General Asset Pricing Framework

Modern asset pricing theory operates within a probabilistic framework that accounts for evolving economic uncertainty. The foundation is a filtered probability space $(\Omega, \mathcal{F}, \{\mathcal{F}_t\}_{t\geq 0}, \mathbb{P})$, where the sample space Ω represents all possible trajectories of economic fundamentals - from routine business cycles to unexpected events like geopolitical shocks or technological disruptions. The filtration $\{\mathcal{F}_t\}$, an increasing sequence of σ-algebras, models the gradual revelation of market information: at each time t, the σ-algebra $\mathcal{F}_t$ encodes all knowable events, such as historical price movements, realized dividends, and central bank policy decisions up to that moment.

The total cash payoff at $t+1$ comprises two components: dividend income D_{t+1} and proceeds from asset resale P_{t+1}. Formally:

$$\text{Total Payoff}_{t+1} = \underbrace{D_{t+1}}_{\text{Dividend}} + \underbrace{P_{t+1}}_{\text{Resale Price}}$$

The fundamental recursive pricing relationship emerges as:

$$P_t = \mathbb{E}_t \left[M_{t+1}(D_{t+1} + P_{t+1}) \right] \tag{1}$$

where M_{t+1} denotes the stochastic discount factor that encodes both time preference and risk adjustments. This equation possesses an inherently recursive structure - the current price P_t depends explicitly on the next period's price P_{t+1}, which itself satisfies an identical pricing equation. Through successive substitutions of future prices $P_{t+k} = \mathbb{E}_{t+k}[M_{t+k+1}(D_{t+k+1} + P_{t+k+1})]$ and application of the law of iterated expectations, we derive the explicit present value formula:

$$P_t = \mathbb{E}_t \left[\sum_{k=1}^{\infty} \left(\prod_{s=1}^{k} M_{t+s} \right) D_{t+k} \right] \tag{2}$$

The equivalence between these representations relies crucially on the transversality condition $\lim_{T\to\infty} \mathbb{E}_t[(\prod_{s=1}^{T} M_{t+s}) P_{t+T}] = 0$, which prohibits self-fulfilling speculative bubbles. Each substitution step embeds deeper future price dependencies into the current valuation, as demonstrated by a three-period expansion:

$$\begin{aligned} P_t = \mathbb{E}_t \Big[& M_{t+1} D_{t+1} + M_{t+1} \mathbb{E}_{t+1} \Big[\\ & M_{t+2} D_{t+2} + M_{t+2} \mathbb{E}_{t+2} \Big[\\ & M_{t+3} \left(D_{t+3} + P_{t+3} \right) \Big] \Big] \Big] \end{aligned}$$

Continuing this process infinitely collapses the recursive structure into the discounted dividend series of Equation (2). The product operator $\prod_{s=1}^{k} M_{t+s}$ captures the compounding of stochastic discounting over multiple periods, weighting each dividend D_{t+k} by the marginal utility of consumption along each economic path $\omega \in \Omega$.

3.2 Event-Driven Asset Pricing Framework

The above framework exhibits discontinuous repricings due to stochastic events that alter cash flow dynamics and risk perceptions. These events, whose timing and magnitude are *a priori* uncertain, can be formalized through an *event impact operator* $\zeta_t : \Omega \to \mathbb{R}$ adapted to the filtration $\{\mathcal{F}_t\}$, capturing instantaneous changes in fundamentals or preferences. Let $\mathbb{I}_t \in \{0, 1\}$ denote an $\mathcal{F}_t$-measurable indicator marking event occurrence at time t, with $\mathbb{E}_{t-1}[\mathbb{I}_t] = \pi_t$ representing the conditional event probability.

The total payoff structure generalizes to incorporate event-driven discontinuities:

$$\text{Total Payoff}_{t+1} = D_{t+1} + P_{t+1} + \mathbb{I}_{t+1}\zeta_{t+1}, \tag{3}$$

where ζ_{t+1} quantifies the event's financial impact. Substituting into the fundamental pricing equation yields:

$$P_t = \mathbb{E}_t \left[M_{t+1} \left(D_{t+1} + P_{t+1} + \mathbb{I}_{t+1}\zeta_{t+1} \right) \right]. \tag{4}$$

Recursive expansion generates two distinct present value components:

$$\begin{aligned} P_t = & \underbrace{\mathbb{E}_t \left[\sum_{k=1}^{\infty} \left(\prod_{s=1}^{k} M_{t+s} \right) D_{t+k} \right]}_{\text{Fundamental Value}} \\ & + \underbrace{\mathbb{E}_t \left[\sum_{k=1}^{\infty} \left(\prod_{s=1}^{k} M_{t+s} \right) \mathbb{I}_{t+k}\zeta_{t+k} \right]}_{\text{Event Risk Premium}}. \end{aligned} \tag{5}$$

The transversality condition now constrains both terms:

$$\begin{aligned} \lim_{T\to\infty} \mathbb{E}_t \Bigg[& \prod_{s=1}^{T} M_{t+s} \\ & \times \left(P_{t+T} + \sum_{k=T+1}^{\infty} \mathbb{I}_{t+k}\zeta_{t+k} \right) \Bigg] = 0 \end{aligned}$$

prohibiting explosive paths in both dividend expectations and event impact projections.

Event risk permeates the stochastic discount factor through two channels:

$$M_{t+1} = \beta \frac{u'\left(C_{t+1} + \mathbb{I}_{t+1}\zeta_{t+1}\right)}{u'(C_t)} \left(1 + \theta_t \mathbb{I}_{t+1}\right),$$

where θ_t encodes compensation for event timing uncertainty. The multiplicative adjustment $(1 + \theta_t \mathbb{I}_{t+1})$ generates event-specific risk premia, even when ζ_t and $\mathbb{I}_t$ are orthogonal to consumption shocks.

Event impacts decompose into systematic and idiosyncratic components via:

$$\zeta_t = \underbrace{\mathbb{E}_t \left[\zeta_t | \mathbb{I}_t = 1\right]}_{\text{Priced Impact}} + \underbrace{\zeta_t - \mathbb{E}_t \left[\zeta_t | \mathbb{I}_t = 1\right]}_{\text{Unpriced Residual}}.$$

Only the conditional expectation $\mathbb{E}_t[\zeta_t | \mathbb{I}_t = 1]$ affects equilibrium prices, reflecting investors' compensation for predictable event consequences.

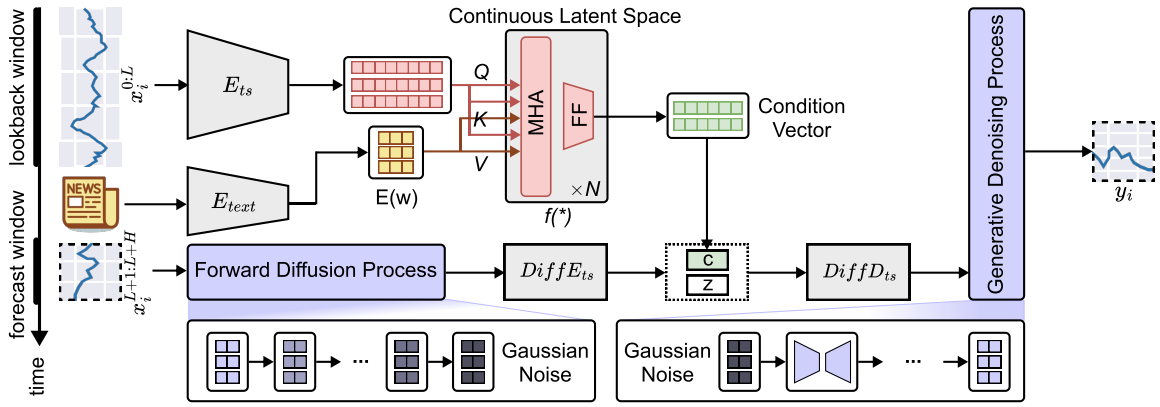

Figure 2: The framework integrates diffusion processes for generating time-series data conditioned on real-world data. Time-series data (x_{ts}) and news data ($news$) are encoded into latent representations using layers E_{ts} and E_{text}, respectively. The model utilizes a forward diffusion process during training, where DenseNet-based encoders $DiffE_{ts}$ and decoders $DiffD_{ts}$ handle the noisy transformations. After training, the model generates a condition vector, which guides the generative denoising process to produce the final prediction (y_i). Once trained, the forward diffusion process is no longer required during inference.

3.3 Bridging Prediction and Asset Pricing

Conventional price forecasting models operate under fundamentally different assumptions than the asset pricing framework in Equations (2–5). Where structural models decompose prices into discounted cash flows and risk premia, predictive approaches typically estimate reduced-form mappings $P_{t+1} = f(P_t, \mathbf{x}_t) + \epsilon_t$ that ignore the recursive equilibrium structure. This creates three critical disconnects:

First, the martingale structure in Equation (1) requires future prices P_{t+1} to be equilibrium objects rather than exogenous targets. Second, event impacts $\mathbb{I}_t \zeta_t$ in Equation (3) manifest through both cash flow shocks and risk premium adjustments, a dual channel absent in standard prediction tasks. Third, the transversality condition imposes nonlinear constraints on long-horizon forecasts that conventional models violate.

Our latent diffusion framework resolves these tensions by reformulating prediction as structured expectation estimation. The model targets the decomposition:

$$\mathbb{E}_t\left[\frac{P_{t+k}}{P_t}\right] = \underbrace{\mathbb{E}_t\left[\prod_{s=1}^{k} M_{t+s}\right]^{-1}}_{\text{Discounting}} + \underbrace{\text{Cov}_t\left(\prod_{s=1}^{k} M_{t+s}, \frac{P_{t+k}}{P_t}\right)}_{\text{Risk Adjustment}}.$$

The diffusion process learns to estimate both components through its denoising mechanics. Each reverse step $t \to t-1$ implicitly computes:

$$\mathbb{E}_t\left[M_{t+1}(\cdot)\right] \approx \epsilon_\theta(\mathbf{y}_t, t, \mathbf{x}, \mathbf{s})$$

where the noise predictor ϵ_θ encodes time-varying risk premia through its attention patterns. Textual inputs $\mathbf{s}$ modulate event probabilities π_t and impact distributions ζ_t via the cross-attention mechanism in Equation (11), preserving the structure of Equation (5).

Our approach fundamentally differs from conventional forecasting by maintaining the present value identity throughout the diffusion process. The model doesn't merely predict prices - it estimates the equilibrium value process consistent with Equation (1) and Equation (4), filtered through market data and textual disclosures. Further theoretical analysis can be found in Appendix A.

3.4 Latent Diffusion for Asset Pricing

Consider an asset pricing problem with triplet observational data $\mathcal{D} = \{(\mathbf{s}_i, \mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^N$, where $\mathbf{s}_i$ represents textual metadata (e.g., earnings call transcripts), $\mathbf{x}_i \in \mathbb{R}^L$ denotes L-dimensional historical market feature vectors (containing time series data such as returns, trading volume, and macroeconomic indicators), and $\mathbf{y}_i \in \mathbb{R}^H$ represents H-period forward-looking asset returns. We model the conditional distribution $p(\mathbf{y}|\mathbf{x}, \mathbf{s})$ through a latent space stochastic differential equation, enabling unified processing of continuous market signals and discrete event impacts via a coordinated diffusion mechanism.

3.4.1 Diffusion Mechanism

The forward diffusion process perturbs asset return trajectories $\mathbf{y}_0$ according to a volatility-aware noise schedule $\{\beta_t\}_{t=1}^T$:

$$q(\mathbf{y}_t|\mathbf{y}_{t-1}) = \mathcal{N}\left(\mathbf{y}_t; \sqrt{1-\beta_t}\mathbf{y}_{t-1}, \beta_t\mathbf{I}\right) \quad (6)$$

$$q(\mathbf{y}_t|\mathbf{y}_0) = \mathcal{N}\left(\mathbf{y}_t; \sqrt{\bar{\alpha}_t}\mathbf{y}_0, (1-\bar{\alpha}_t)\mathbf{I}\right) \quad (7)$$

where the cumulative noise scaling factor $\bar{\alpha}_t = \prod_{s=1}^{t}(1-\beta_s)$ is derived through multiplicative noise scheduling. The reverse process employs conditional transition probability:

$$p_\theta(\mathbf{y}_{t-1}|\mathbf{y}_t, \mathbf{x}, \mathbf{s}) = \mathcal{N}\left(\mathbf{y}_{t-1}; \mu_\theta(\mathbf{y}_t, t, \mathbf{x}, \mathbf{s}), \sigma_t^2\mathbf{I}\right) \quad (8)$$

explicitly incorporating market state information, with variance term $\sigma_t^2 = \beta_t(1-\bar{\alpha}_{t-1})/(1-\bar{\alpha}_t)$ maintaining diffusion process variance conservation. The mean function:

$$\mu_\theta = \frac{1}{\sqrt{1-\beta_t}}\left(\mathbf{y}_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(\mathbf{y}_t, t, \mathbf{x}, \mathbf{s})\right), \quad (9)$$

separates fundamental values from speculative noise, where the noise predictor $\epsilon_\theta : \mathbb{R}^H \times \mathbb{N} \times \mathbb{R}^L \times \mathcal{S} \to \mathbb{R}^H$ learns to identify non-fundamental price components.

3.4.2 Market-Adapted Architecture

The noise prediction network ϵ_θ implements asset pricing through tripartite information fusion:

Textual Encoder: A pretrained language model E_{text} extracts semantic features from text $\mathbf{s}$, with domain-adapted projection matrix $W_p \in \mathbb{R}^{d \times d_{\text{enc}}}$ mapping embeddings to financial semantic space:

$$\mathbf{H}_s = W_p \cdot E_{\text{text}}(\mathbf{s})$$

Time Series Encoder: Past sequence $\mathbf{x}_t \in \mathbb{R}^L$ encodes historical information through position-aware embeddings:

$$\mathbf{E}_t = W_e\mathbf{z}_t + \mathbf{P},$$

where learnable matrix $W_e \in \mathbb{R}^{d \times L}$ enables feature lifting, and positional encoding $\mathbf{P} \in \mathbb{R}^{d \times L}$ captures time-dependent market momentum effects. $\mathbf{P}$ is a trainable matrix. While we also explored using recurrent neural networks such as LSTM (Lindemann et al., 2021) for encoding the time series, the performance was not as effective as the simpler encoder described here.

Cross-Modal Attention: Cascaded attention mechanisms simulate fundamental analysis workflows:

$$\mathbf{E}' = \text{Softmax}\left(\frac{(\mathbf{E}W_Q)(\mathbf{E}W_K)^\top}{\sqrt{d}}\right)\mathbf{E}W_V \quad (10)$$

$$\mathbf{E}'' = \text{Softmax}\left(\frac{(\mathbf{E}'W'_Q)(\mathbf{H}_sW'_K)^\top}{\sqrt{d}}\right)\mathbf{H}_sW'_V \quad (11)$$

Projection matrices $W_Q, W_K, W_V \in \mathbb{R}^{d \times d}$ map inputs to query-key-value spaces. The first stage $\mathbf{E}'$ extracts intrinsic value patterns from market data, while the second stage $\mathbf{E}''$ adjusts valuations based on textual information. c is the matrix obtained by applying a trainable matrix projection to $\mathbf{E}''$.

3.4.3 Learning Objective

The model minimizes spectral norms to separate market noise:

$$\mathcal{L}(\theta) = \mathbb{E}_{t,\epsilon,\mathbf{x},\mathbf{s}}\Bigg[\|\epsilon - \epsilon_\theta\Big(\underbrace{\sqrt{\bar{\alpha}_t}\mathbf{y}_0 + \sqrt{1-\bar{\alpha}_t}\epsilon}_{\text{noise-perturbed returns}}, t, \mathbf{x}, \mathbf{s}\Big)\|_2^2\Bigg] \quad (12)$$

where $\epsilon \sim \mathcal{N}(0, \mathbf{I})$ characterizes market microstructure noise, forcing the objective function to distinguish persistent pricing signals ($\mathbf{y}_0$) from transient market frictions (ϵ).

4 Experiments

In this section, we first introduce the basic setup of the experiment, including the baselines and evaluation metrics. Subsequently, in Section 4.2, we present our main results and comparisons with some baseline models. In Section 4.3, we will conduct a detailed analysis of the experimental results from several aspects, including the limitations of LLMs, the impact of model parameters and the ablation studies of the model. Due to space limitations, parts that are not presented, such as hyperparameter settings, can be referred to in Appendix C.

4.1 Experiment Settings

Baselines: For single time series modality, the selection of baseline is based on Wang et al. (2024). PatchTST (Nie et al., 2022) owns the best performance in short-term forecasting, which is Transformer-based. TimesNet (Wu et al., 2022) is

CNN-based. Mamba (Gu and Dao, 2023) is RNN-based. TimeLLM (Jin et al., 2023) is LLM-based.

For considering both text and time series modalities, we select the following six baselines for new-driven prediction: MAN-SF (Sawhney et al., 2020), TimeLLM (Jin et al., 2023), Qwen2.5-7B-Instruct, Qwen2.5-7B (Yang et al., 2024), Llama-2-7b, Llama-2-7b-chat (Touvron et al., 2023).

Metric: We select the Symmetric Mean Absolute Percentage Error (SMAPE) and Mean Absolute Scaled Error (MASE) as metrics, which focus on absolute errors and reduce the impact of outliers, providing reliable evaluations of forecast accuracy across different methodologies.

4.2 Main Results

As illustrated by Table 1 and Table 2, the experimental results offer several key insights into asset pricing. Models that rely solely on time series data face inherent limitations, as demonstrated by the performance degradation of most baselines with longer forecasting horizons. For instance, PatchTST's SMAPE rises from 2.02 to 2.97 as the forecast window expands from 2 to 10. This highlights the difficulty of extracting sufficient predictive signals from temporal patterns alone.

Although LLMs show strong capabilities in text comprehension, their limitations in numerical reasoning become apparent when applied to time series forecasting. The Qwen2.5-7B variants and Llama-2-7b fall short of our model's performance by a significant margin—1.25 SMAPE compared to 0.103 at a forecasting length of 2 when incorporating news. This underscores that while textual understanding is valuable, it cannot replace the need for robust numerical processing in financial prediction tasks.

Our model excels by leveraging two complementary mechanisms. Its inherent stochastic modeling aligns with the characteristics of financial time series, as shown by consistently superior MASE scores in pure time series tasks. Additionally, the effective integration of event descriptions boosts predictive power, with a 20-50% improvement in SMAPE over LLM baselines when news is incorporated. The model's performance continues to improve with increasing time lags, with SMAPE rising from 0.103 to 0.205, further demonstrating its robustness for extended forecasting windows.

4.3 Analysis

Instruction Tuning for LLMs Table 3 reveals fundamental limitations of LLMs in temporal analysis for asset pricing. While instruction-tuned models like QWEN2.5-7B-INSTRUCT show modest improvements (3.6% SMAPE reduction, 10.2% MASE reduction), the absolute performance remains poor compared to traditional time series models. Notably: The best-performing instruction-tuned model (QWEN2.5-7B-INSTRUCT) still achieves only 47.43 MASE, where values >1 indicate worse performance than naive forecasts. Instruction tuning provides marginal benefits (average 8.7% SMAPE improvement across models) that fail to meaningfully close the performance gap. Base models and their instruction-tuned variants show similar error patterns, suggesting architectural rather than training limitations. These results challenge the conventional wisdom that instruction tuning adapts LLMs effectively to specialized domains. The persistent high errors in both zero-shot and tuned configurations reveal fundamental limitations in LLMs' ability to: (1) model temporal dependencies, (2) understand financial time series patterns, and (3) perform numerical reasoning with market data. This suggests current LLM architectures may be inherently unsuitable for time-sensitive financial forecasting tasks, regardless of tuning approaches.

Choice of Textual Encoder The experimental results in Table 4, demonstrate limited sensitivity of model performance to textual encoder selection. As shown in the table, varying encoder architectures (110M-355M parameters) yield comparable prediction accuracy across both metrics: SMAPE fluctuates within a narrow 0.37–0.53 range while MASE remains confined to 3.25–3.92. Notably, parameter-efficient DistilBERT (66M) achieves only marginally higher errors compared to larger counterparts like RoBERTa-large (355M), with a modest 0.16 SMAPE and 0.67 MASE degradation despite 5.4× fewer parameters. This robustness suggests that textual features primarily serve as auxiliary signals rather than dominant predictive factors in our framework. The observed consistency across encoder variants implies that downstream market-adaptive fusion mechanisms effectively mitigate potential information bottlenecks from text representations, prioritizing numeric market dynamics over linguistic nuances in asset pricing.

Modeling Residuals for Traditional Asset Pricing In our approach, we model the residuals from

Forecasting lengths	PatchTST		TimesNet		Mamba		TimeLLM		Ours	
	SMAPE	MASE	SMAPE	MASE	SMAPE	MASE	SMAPE	MASE	SMAPE	MASE
len = 2	2.02	47.23	0.18	3.61	1.58	35.12	1.01	23.64	0.18	7.23
len = 4	2.30	53.46	0.26	5.21	2.30	53.54	1.39	29.88	0.16	1.21
len = 6	2.58	61.77	0.33	6.93	1.60	33.76	2.71	63.02	0.23	1.82
len = 8	2.75	64.05	0.42	8.83	1.63	35.21	2.98	64.74	0.38	2.50
len = 10	2.97	69.02	0.46	9.71	1.69	37.42	3.85	70.11	0.56	3.12

Table 1: The selected baselines and our model take only time series as input. For our model, this means removing the text encoder and MHA module, degrading it into a conditional generation problem based on past time series. The `len` represents the prediction lengths of 2, 4, 6, 8, and 10. All models are trained on the training set. We use the background color to mark the optimal results, and the background color to mark the suboptimal results.

Time Lag	MAN-SF		TimeLLM		Qwen2.5-7B-Instruct		Qwen2.5-7B		Llama-2-7b		Llama-2-7b-chat		Ours	
	SMAPE	MASE	SMAPE	MASE	SMAPE	MASE	SMAPE	MASE	SMAPE	MASE	SMAPE	MASE	SMAPE	MASE
len = 2	1.52	33.16	1.32	29.63	1.25	28.07	1.32	29.28	1.48	32.49	1.42	31.52	0.11	1.63
len = 4	1.78	39.57	1.41	30.07	1.47	32.12	1.54	34.32	1.67	37.00	1.63	36.50	0.15	2.12
len = 6	2.05	45.42	3.79	70.24	1.69	37.33	1.76	39.59	1.88	41.87	1.82	40.17	0.18	2.47
len = 8	2.31	52.02	3.91	72.03	1.91	42.27	1.98	44.63	2.09	46.51	2.01	45.10	0.22	2.81
len = 10	2.57	51.78	4.05	74.88	2.13	47.43	2.20	49.97	2.30	51.33	2.25	50.45	0.37	3.25

Table 2: We select the rationality of the baseline to ensure fair comparisons. All models are trained on the training set. For large language models, we use a similar model size of 7 billion (7b) for comparison. Additionally, for the base model, we incorporate special tokens during inference. We use background color to mark the optimal results, and the background color to mark the suboptimal results.

Model	Zero Shot		Instruction Tuning	
	SMAPE	MASE	SMAPE	MASE
QWEN2.5-7B-INSTRUCT	2.47	52.79	2.13	47.43
QWEN2.5-7B	2.51	55.63	2.20	49.97
LLAMA-2-7B	2.74	59.03	2.30	51.33
LLAMA-2-7B-chat	2.46	52.37	2.25	50.45

Table 3: We fine-tuned four LLMs using full parameter fine-tuning. The training was conducted on 4×A800 GPUs, with a batch size of 5 per GPU. The learning rate was set to 3×10^{-5}, and we did not use a learning rate scheduler, keeping it constant throughout the training. The training lasted for 4 epochs. For the prediction task, we set the `len` to 10.

Encoder	Parameters	SMAPE	MASE
BERT-base-cased	110M	0.37	3.25
BERT-base-uncased	110M	0.44	3.48
DistilBERT-base-uncased	66M	0.53	3.92
RoBERTa-base	125M	0.47	3.58
RoBERTa-large	355M	0.41	3.42

Table 4: We tested different text encoders in our framework, where "M" denotes millions. The **Bert-base-cased** model (Devlin, 2018) is case-sensitive, suitable for tasks like named entity recognition. The **Bert-base-uncased** model (Devlin, 2018) is case-insensitive and more efficient for tasks where capitalization is not critical. The **Distilbert-base-uncased** model (Sanh, 2019) is a smaller, faster version of BERT. The **Roberta-base** (Liu, 2019) and **Roberta-large** (Liu, 2019) are optimized for improved performance on complex tasks.

CAPM, Fama3, and Fama5 by training the network with news and historical time series as input. The labels for training are the residuals that CAPM, Fama3, and Fama5 cannot explain. The residuals are then explained using the predicted results, with key metrics such as coefficients, R-squared, and p-values used to evaluate the interpretability of our method. The results of modeling different residuals are shown in Table 5. Our model incorporates inherent randomness, but it still maintains compat-

Model	Coefficient	R-Squared	p-value
CAPM Residuals	3.35**	0.18	0.008
Fama3 Residuals	3.41**	0.19	0.004
Fama5 Residuals	2.09**	0.27	0.006

Table 5: This table shows the results of modeling residuals for CAPM (Sharpe, 1964), Fama3 (Fama and French, 1992), and Fama5 (Fama and French, 2015) with key performance metrics: Coefficient, R-Squared, and p-value. The Coefficient indicates the strength and direction of the relationship between variables, R-Squared measures the proportion of variance explained by the model, and p-value assesses the statistical significance of the results. The significance levels are indicated as follows: * $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$. More introductions about these asset pricing models is provided in Appendix E

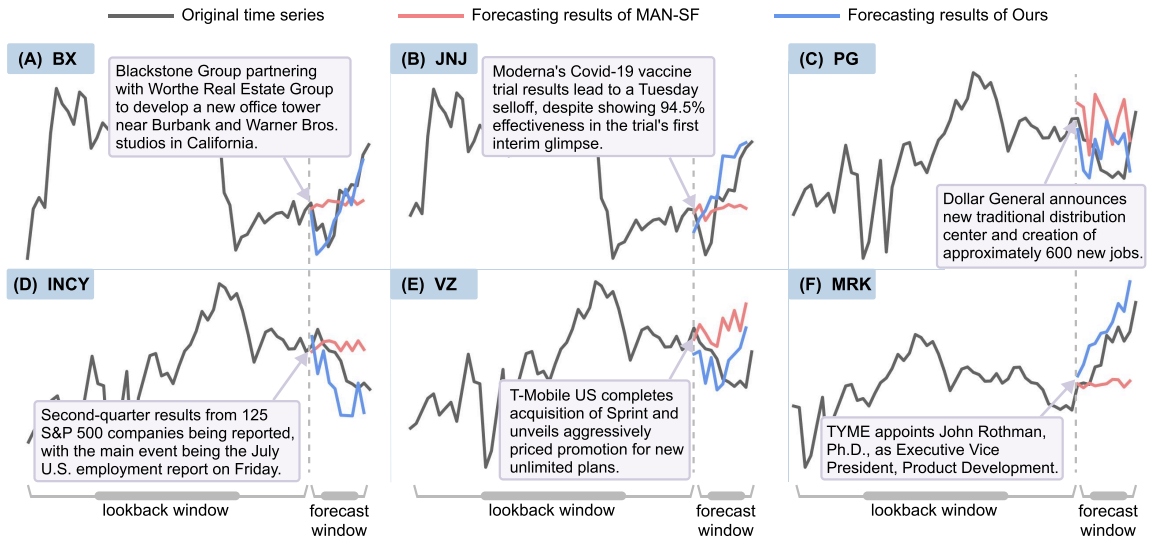

Figure 3: We selected six representative news articles to showcase the effectiveness of latent diffusion. The gray line represents the historical window.

ibility with classical asset pricing theories. The first mechanism is the underlying diffusion process. This module allows our model to exhibit a high correlation with traditional stochastic discount factors, such as those from the Fama models, ensuring that the randomness introduced does not deviate from the established asset pricing framework.

Ablation Study From the ablation experiments shown in Table 6, we conclude disabling cross-attention mechanisms causes catastrophic performance collapse - SMAPE doubles (0.37→0.76) and MASE deteriorates by 21% (3.25→3.93), confirming that market-text interaction modeling constitutes the framework's analytical core. These results demonstrate that simple feature concatenation (via pooling) fails to capture nuanced cross-modal dependencies essential for asset pricing, while the text encoder primarily serves as auxiliary regularization rather than a primary signal source.

Model Variant	SMAPE ↓	MASE ↓	Speed ↑
Full Model	0.37	3.25	1.31
w/o Text Encoder	0.56	3.14	3.24
w/o Cross-Attention	0.76	3.93	3.85

Table 6: By removing the cross - attention mechanism, we employ pooling operations to align the text representations with the temporal representations. The speed is measured in steps per second.

Case Study Figure 3 illustrates our latent diffusion framework's capability to capture complex event-driven market dynamics. By unifying heterogeneous events—from strategic expansions (BX) to clinical trial volatility (MRNA)—the model effectively separates fundamental drivers (e.g., T-Mobile's pricing strategy effects) from transient noise (e.g., Moderna's post-news selloff). Its cross-modal fusion dynamically weights textual and numerical signals, enabling robust predictions for both rapid shocks (vaccine updates) and gradual shifts (infrastructure investments). The diffusion mechanism's inherent noise suppression further mitigates overreaction to superficial triggers (TYME leadership changes), prioritizing material economic impacts across diverse event types.

5 Limitions and Conclusion

In this work, we propose a novel multimodal latent diffusion framework for event-driven asset pricing. This framework addresses critical challenges in aligning time series with textual event representations, while explicitly modeling the stochastic nature of financial systems. The extensive empirical validation demonstrates superior predictive performance over conventional baselines. Additionally, its compatibility with classical asset pricing theories ensures interpretability, which is an often overlooked but critical aspect in financial applications.

Despite these strengths, the framework's performance is closely tied to data quality. Noisy, ambiguous, or contextually incomplete event descriptions may propagate errors through the latent alignment process, potentially undermining the framework's ability to accurately isolate causal relationships between events and price movements. Thus, the efficacy of our approach may be diminished when the input data lacks clarity or coherence.

References

Defu Cao, Furong Jia, Sercan O Arik, Tomas Pfister, Yixiang Zheng, Wen Ye, and Yan Liu. 2023. Tempo: Prompt-based generative pre-trained transformer for time series forecasting. *arXiv preprint arXiv:2310.04948*.

Zihan Chen, Lei Nico Zheng, Cheng Lu, Jialu Yuan, and Di Zhu. 2023. Chatgpt informed graph neural network for stock movement prediction. *arXiv preprint arXiv:2306.03763*.

Mayank Daswani, Mathias MJ Bellaiche, Marc Wilson, Desislav Ivanov, Mikhail Papkov, Eva Schnider, Jing Tang, Kay Lamerigts, Gabriela Botea, Michael A Sanchez, et al. 2024. Plots unlock time-series understanding in multimodal models. *arXiv preprint arXiv:2410.02637*.

Jacob Devlin. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Eugene F Fama. 1970. Efficient capital markets. *Journal of finance*, 25(2):383–417.

Eugene F Fama and Kenneth R French. 1992. The cross-section of expected stock returns. *the Journal of Finance*, 47(2):427–465.

Eugene F Fama and Kenneth R French. 2015. A five-factor asset pricing model. *Journal of financial economics*, 116(1):1–22.

Nate Gruver, Marc Finzi, Shikai Qiu, and Andrew G Wilson. 2024. Large language models are zero-shot time series forecasters. *Advances in Neural Information Processing Systems*, 36.

Albert Gu and Tri Dao. 2023. Mamba: Linear-time sequence modeling with selective state spaces. *arXiv preprint arXiv:2312.00752*.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*.

Jianing Hao, Zhuowen Liang, Chunting Li, Yuyu Luo, Jie Li, and Wei Zeng. 2024. Vistr: Visualizations as representations for time-series table reasoning. *arXiv preprint arXiv:2406.03753*.

Ming Jin, Shiyu Wang, Lintao Ma, Zhixuan Chu, James Y Zhang, Xiaoming Shi, Pin-Yu Chen, Yuxuan Liang, Yuan-Fang Li, Shirui Pan, et al. 2023. Time-llm: Time series forecasting by reprogramming large language models. *arXiv preprint arXiv:2310.01728*.

Kai Kim, Howard Tsai, Rajat Sen, Abhimanyu Das, Zihao Zhou, Abhishek Tanpure, Mathew Luo, and Rose Yu. 2024. Multi-modal forecaster: Jointly predicting time series and textual data. *arXiv preprint arXiv:2411.06735*.

Jun Li, Che Liu, Sibo Cheng, Rossella Arcucci, and Shenda Hong. 2024. Frozen language model helps ecg zero-shot learning. In *Medical Imaging with Deep Learning*, pages 402–415. PMLR.

Benjamin Lindemann, Timo Müller, Hannes Vietz, Nasser Jazdi, and Michael Weyrich. 2021. A survey on long short-term memory networks for time series prediction. *Procedia Cirp*, 99:650–655.

Haoxin Liu, Shangqing Xu, Zhiyuan Zhao, Lingkai Kong, Harshavardhan Kamarthi, Aditya B Sasanur, Megha Sharma, Jiaming Cui, Qingsong Wen, Chao Zhang, et al. 2024a. Time-mmd: A new multi-domain multimodal dataset for time series analysis. *arXiv preprint arXiv:2406.08627*.

Haoxin Liu, Zhiyuan Zhao, Jindong Wang, Harshavardhan Kamarthi, and B Aditya Prakash. 2024b. Lst-prompt: Large language models as zero-shot time series forecasters by long-short-term prompting. *arXiv preprint arXiv:2402.16132*.

Yinhan Liu. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 364.

Yong Liu, Guo Qin, Xiangdong Huang, Jianmin Wang, and Mingsheng Long. 2024c. Autotimes: Autoregressive time series forecasters via large language models. *arXiv preprint arXiv:2402.02370*.

Yu Ma, Rui Mao, Qika Lin, Peng Wu, and Erik Cambria. 2023. Multi-source aggregated classification for stock price movement prediction. *Information Fusion*, 91:515–528.

Yuqi Nie, Nam H Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. 2022. A time series is worth 64 words: Long-term forecasting with transformers. *arXiv preprint arXiv:2211.14730*.

Venkata Sasank Pagolu, Kamal Nayan Reddy, Ganapati Panda, and Babita Majhi. 2016. Sentiment analysis of twitter data for predicting stock market movements. In *2016 international conference on signal processing, communication, power and embedded system (SCOPES)*, pages 1345–1350. IEEE.

Konstantinos I Roumeliotis and Nikolaos D Tselikas. 2023. Chatgpt and open-ai models: A preliminary review. *Future Internet*, 15(6):192.

V Sanh. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.

Ramit Sawhney, Shivam Agarwal, Arnav Wadhwa, and Rajiv Shah. 2020. Deep attentive learning for stock movement prediction from social media text and company correlations. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8415–8426.

Matthias Scherf, Xenia Matschke, and Marc Oliver Rieger. 2022. Stock market reactions to covid-19 lockdown: A global analysis. *Finance research letters*, 45:102245.

William F Sharpe. 1964. Capital asset prices: A theory of market equilibrium under conditions of risk. *The journal of finance*, 19(3):425–442.

Chenxi Sun, Hongyan Li, Yaliang Li, and Shenda Hong. 2023. Test: Text prototype aligned embedding to activate llm's ability for time series. *arXiv preprint arXiv:2308.08241*.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Yuxuan Wang, Haixu Wu, Jiaxiang Dong, Yong Liu, Mingsheng Long, and Jianmin Wang. 2024. Deep time series models: A comprehensive survey and benchmark.

Haixu Wu, Tengge Hu, Yong Liu, Hang Zhou, Jianmin Wang, and Mingsheng Long. 2022. Timesnet: Temporal 2d-variation modeling for general time series analysis. *arXiv preprint arXiv:2210.02186*.

Zhijian Xu, Yuxuan Bian, Jianyuan Zhong, Xiangyu Wen, and Qiang Xu. 2024. Beyond trend and periodicity: Guiding time series forecasting with textual cues. *arXiv preprint arXiv:2405.13522*.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, et al. 2024. Qwen2. 5 technical report. *arXiv preprint arXiv:2412.15115*.

Xinli Yu, Zheng Chen, Yuan Ling, Shujing Dong, Zongyi Liu, and Yanbin Lu. 2023. Temporal data meets llm–explainable financial time series forecasting. *arXiv preprint arXiv:2306.11025*.

Yuan Yuan, Zhaojian Li, and Bin Zhao. 2025. A survey of multimodal learning: Methods, applications, and future. *ACM Computing Surveys*.

Wenbin Zhang and Steven Skiena. 2010. Trading strategies to exploit blog and news sentiment. In *Proceedings of the international AAAI conference on web and social media*, volume 4, pages 375–378.

A Theoretical Analysis

Notation Setup We have P as the physical probability measure, Q as the risk-neutral probability measure, $M_t := \frac{dQ}{dP}\big|_{F_t}$ as the stochastic discount factor (Radon-Nikodym derivative), and $y_t \in \mathbb{R}^n$ as the asset return process at diffusion step t. Under the physical probability measure P, the conditional probability density is denoted by $p_t(y \mid x, s)$, while under the risk-neutral probability measure Q, the conditional probability density is represented by $q_t(y \mid x, s)$.

Theorem 1. (Diffusion-SDF Duality) Under no-arbitrage conditions with complete markets, the denoising process $\{\mathbf{y}_t\}_{t=0}^T$ defined by Equations (6-9) implicitly learns the stochastic discount factor M_t through its noise prediction mechanism ϵ_θ. Specifically, the learned score function $\nabla_{\mathbf{y}} \log p_t(\mathbf{y}|\mathbf{x}, \mathbf{s})$ corresponds to the logarithmic gradient of the risk-neutral measure.

Proof. Define the continuous-time limit of the forward process using the Ornstein-Uhlenbeck parameterization:

$$d\mathbf{y}_t = -\frac{1}{2}\beta(t)\mathbf{y}_t dt + \sqrt{\beta(t)} d\mathbf{W}_t^{\mathbb{P}}$$

where $\mathbf{W}_t^{\mathbb{P}}$ is a $\mathbb{P}$-Brownian motion. The associated reverse-time process under $\mathbb{Q}$ becomes:

$$d\mathbf{y}_t = \left[-\frac{1}{2}\beta(t)\mathbf{y}_t - \beta(t)\nabla_{\mathbf{y}} \log p_t(\mathbf{y}_t|\mathbf{x}, \mathbf{s})\right] dt + \sqrt{\beta(t)} d\mathbf{W}_t^{\mathbb{Q}}$$

By Girsanov's theorem, the measure change satisfies:

$$\frac{d\mathbb{Q}}{d\mathbb{P}} = \exp\left(-\int_0^T \nabla_{\mathbf{y}} \log p_t(\mathbf{y}_t|\mathbf{x}, \mathbf{s}) \cdot d\mathbf{W}_t^{\mathbb{P}} - \frac{1}{2}\int_0^T \|\nabla_{\mathbf{y}} \log p_t(\mathbf{y}_t|\mathbf{x}, \mathbf{s})\|^2 dt\right)$$

Thus identifying $M_t \propto d\mathbb{Q}/d\mathbb{P}|_{\mathcal{F}_t}$. From asset pricing theory, the fundamental pricing equation states:

$$\mathbb{E}_t^{\mathbb{P}}[M_{t+1} R_{t+1}] = 1$$

where $R_{t+1} = \mathbf{y}_{t+1}/\mathbf{y}_t$. Substituting the measure change relation:

$$\mathbb{E}_t^{\mathbb{Q}}[R_{t+1}] = \mathbb{E}_t^{\mathbb{P}}\left[\frac{M_{t+1}}{M_t} R_{t+1}\right] = 1$$

The score function emerges through the Fokker-Planck equation for the reverse process:

$$\frac{\partial p_t}{\partial t} = -\nabla_{\mathbf{y}} \cdot [\mu_\theta p_t] + \frac{1}{2}\beta(t)\Delta p_t$$

Substituting μ_θ from Equation (9), we derive:

$$\nabla_{\mathbf{y}} \log p_t(\mathbf{y}|\mathbf{x}, \mathbf{s}) = \mathbb{E}^{\mathbb{Q}}\left[\frac{M_T}{M_t} \nabla_{\mathbf{y}} \log q_T(\mathbf{y}_T|\mathbf{x}, \mathbf{s}) \,\middle|\, \mathcal{F}_t\right]$$

This establishes the score function as a weighted expectation of future SDF-adjusted gradients.

Reformulate the denoising objective using Doob-Meyer decomposition:

$$\epsilon_\theta(\mathbf{y}_t, t, \mathbf{x}, \mathbf{s}) = \underbrace{\mathbb{E}^{\mathbb{P}}[\mathbf{y}_0|\mathbf{y}_t]}_{\text{Physical expectation}} - \underbrace{\mathbb{E}^{\mathbb{Q}}[\mathbf{y}_0|\mathbf{y}_t]}_{\text{Risk-neutral expectation}}$$

Substituting the Radon-Nikodym derivative:

$$\epsilon_\theta = \int \mathbf{y}_0 \left(\frac{p_t(\mathbf{y}_0|\mathbf{y}_t)}{q_t(\mathbf{y}_0|\mathbf{y}_t)} - 1\right) q_t(\mathbf{y}_0|\mathbf{y}_t) d\mathbf{y}_0$$

This reveals ϵ_θ as the covariance between future returns and SDF innovations:

$$\epsilon_\theta = \mathbb{C}\text{ov}^{\mathbb{P}}(\mathbf{y}_0, M_{0t}|\mathbf{y}_t)$$

where $M_{0t} = M_t/M_0$.

The training objective $\mathcal{L}(\theta)$ in Equation (12) induces a variational problem:

$$\min_\theta \mathbb{E}^{\mathbb{P}}\left[\|\epsilon_\theta - \mathbb{C}\text{ov}^{\mathbb{P}}(\mathbf{y}_0, M_{0t}|\mathcal{F}_t)\|^2\right]$$

First-order conditions yield:

$$\mathbb{E}^{\mathbb{P}}\left[\epsilon_\theta \frac{\partial M_{0t}}{\partial \theta}\right] = 0 \quad \forall \theta$$

This orthogonality condition enforces the Law of One Price: assets with identical exposure to SDF risk must have equal expected returns under $\mathbb{P}$. □

B Dataset

The origin 792,684 news articles are sourced from Dow Jones News Services and the Wall Street Journal, and stored as structured XML files. The structured dataset comprises eight key variables, including {Publication_datetime, Company_code, Company_code, Title, Body, Word_count}. Using the 'Company_code' variable, we filtered and identified 129,753 news articles about individual S&P

500 firms, covering the period from March 8, 2001, to June 30, 2024.

The corresponding daily stock price data for S&P 500 firms from 2001 to 2024 was obtained from the Center for Research in Security Prices (CRSP) database. The collected variables include Date, Volume, Open, High, Low, Close, and Company_code. To align the stock price data with the news data, we used the publication date of each news article and the company code as the reference point. Specifically, news published before the close of the market was associated with the same trading day, while news published after the market close was assigned to the next trading day. By matching the Company code and Publication date between the two datasets, we constructed (news, price) pairs, resulting in a total of 126,521 pairs. Detailed descriptions of the variables within these pairs are provided in Table 7. To identify stocks exhibiting significant price changes influenced by news events within a given time lag (denoted as i), we employed the Bollinger Bands methodology. The bands are calculated as $BB = MA_N \pm K \times SD(MA_N)$, where MA_N is the moving average over N days, $SD(MA_N)$ is the standard deviation of the moving average, K is set as 2, and N is set as 20 because $MA20$ represents a monthly average. If the stock price on day i, denoted as $price_i$, satisfies the condition $price_i \geq MA20 + 2SD(MA20)$ or $price_i \leq MA20 - 2SD(MA20)$, we consider the (news, price) pair meets the 'jump' criterion. The relationship between the specific time lag i and the selected pairs is shown in Table 8.

C Hyperparameters

We conducted extensive experiments and performed a grid search to determine the following hyperparameters, as shown in the Table 9. These settings provide a good balance between training speed and performance. During the training process, our textual encoder is frozen, which means that the number of parameters requiring fine-tuning is relatively small.

D Prompts

System Setting You are an advanced AI system capable of understanding and processing temporal and textual data. Your task is to predict future financial time series values using historical data and relevant news articles. Leverage statistical analysis, natural language processing, and machine learning to generate accurate predictions.

Input The input consists of two components: **News Article**, which is a recent news headline or content in text format, and **Historical Financial Data**, a time series of past financial values represented as numerical data.

Output The output should strictly follow the format outlined below. Ensure the following structure is maintained: - The output must be a JSON object. - It should include a "predictions" field containing an array of prediction objects. - Each prediction object must include a "date" field (in the format "YYYY-MM-DD") and a "value" field (as a numerical value).

Input/Output Examples:

Sample Input

```
{
  "news": {
    "title": "Company X Launches New Product...",
    "content": "..."
  },
  "time_series": {
    "2023-10-01": 100.5,
    "2023-10-02": 101.2,
    "2023-10-03": 102.3,
    ...
  }
}
```

Sample Output

```
{
  "predictions": [
    {"date": "2023-11-20", "value": 121.5},
    ...
  ]
}
```

E Asset Pricing Models

In asset pricing theory, the Capital Asset Pricing Model (CAPM), the Fama-French Three-Factor Model (Fama 3), and the Fama-French Five-Factor Model (Fama 5) represent significant frameworks for explaining the expected return on assets by accounting for various market factors. Below is a brief overview of each model.

E.1 The Capital Asset Pricing Model (CAPM)

The CAPM, introduced by *William Sharpe* in 1964, seeks to explain an asset's expected return in relation to its sensitivity to market risk. The model assumes rational investors, efficient markets, and a risk-free asset. CAPM posits that the return on an

Variable	Description
Publication_date	News article publication date. If the news article was officially published before the close of market, this variable records the same date, else it marks the next date.
Company_code	Unique identifier or code for the relevant company. A unique code that identifies the company mentioned in the news and the prices.
Title	Title of news article. A brief headline that summarizes the main topic or event described in the news article.
Body	The detailed news content.
Word_count	Number of total word count in the body of the news article.
Volume	The number of shares traded on the publication date.
Open	The opening price of the corresponding company on the publication date.
High	The highest stock price of the corresponding company on the publication date.
Low	The lowest stock price of the corresponding company on the publication date.
Close	The closing price of the corresponding company on the publication date.

Table 7: The variables in the collected news articles dataset.

Time Lag	Lag=1	Lag=2	Lag=3	Lag=4	Lag=5	Lag=6	Lag=7	Lag=8	Lag=9	Lag=10
Influenced Rate	0.2361	0.2251	0.2067	0.1915	0.1842	0.1811	0.1750	0.1758	0.1706	0.1737
Influenced Pairs	29,869	28478	26148	24224	23308	22907	22141	22238	21586	21979

Table 8: The influenced paired for different time lags.

asset is determined by its exposure to the overall market's risk, represented by the asset's beta (β_i).

The formula for CAPM is:

$$E(R_i) = R_f + \beta_i \, (E(R_m) - R_f)$$

where $E(R_i)$ is the expected return on asset i, R_f is the risk-free rate, and β_i is the asset's sensitivity to the market return $E(R_m)$. CAPM's simplicity is one of its strengths, though it has faced criticism for not accounting for other factors that influence asset returns, which led to the development of more sophisticated models.

E.2 The Fama-French Three-Factor Model (Fama 3)

In 1993, CAPM is extended by introducing the Fama3. They found that two additional factors—*size* and *value*—helped explain stock returns more effectively. The model builds on the CAPM framework by adding two components: the *SMB (Small Minus Big)* factor, which represents the return difference between small-cap and large-cap stocks, capturing the *size effect*, and the *HML (High Minus Low)* factor, which measures the return difference between value stocks (high book-to-market ratio) and growth stocks (low book-to-market ratio), capturing the *value effect*.

The Fama-French Three-Factor Model is represented as:

$$\begin{aligned} E(R_i) = R_f &+ \beta_i \, (E(R_m) - R_f) \\ &+ \beta_{SMB} \cdot SMB \\ &+ \beta_{HML} \cdot HML \end{aligned}$$

This model significantly improves upon CAPM by addressing the role of firm size and value characteristics in determining asset returns.

E.3 The Fama-French Five-Factor Model (Fama 5)

Fama 5, introduced in 2015, further extends the Three-Factor Model by adding two more factors: *profitability* and *investment*. The model acknowledges that a firm's profitability and investment strategies can influence its stock returns, providing a more comprehensive explanation of asset pricing. The two new factors are *RMW (Robust Minus Weak)*, which reflects the return difference between highly profitable and less profitable firms, and *CMA (Conservative Minus Aggressive)*, which captures the return difference between firms with conservative vs. aggressive investment strategies.

Hyperparameter	Default Value	Description
num_epochs	10	Number of training epochs, default 10
batch_size	32	Training batch size, default 32
lr	1×10^{-5}	Learning rate, default 1×10^{-5}
d_model	128	Dimensionality of the model, default 128
latent_space_dim	512	Latent space dimension, default 512
dropout	0.1	Dropout rate, default 0.1
nhead	4	Number of attention heads in the MHA block, default 4
num_layers	2	Number of layers in the MHA block, default 2

Table 9: Hyperparameter Settings

The Five-Factor Model is expressed as:

$$\begin{aligned} E(R_i) = R_f + \beta_i \left(E(R_m) - R_f\right) \\ + \beta_{SMB} \cdot SMB + \beta_{HML} \cdot HML \\ + \beta_{RMW} \cdot RMW + \beta_{CMA} \cdot CMA \end{aligned}$$

By adding the RMW and CMA factors, the Five-Factor Model provides a deeper and more nuanced understanding of asset pricing, incorporating firm-specific characteristics such as profitability and investment behavior.