Future Technology
Open Access Journal
ISSN 2832-0379

May 2026| Volume 05 | Issue 02 | Pages 268-280

Journal homepage: https://fupubco.com/futech

https://doi.org/10.55670/fpll.futech.5.2.25

Future Publishing LLC

Article

Deep learning-driven regime switching models for capturing structural breaks and volatility clustering in financial time series

Junyu Wang*

University of Queensland, St Lucia QLD 4072, Australia

ARTICLE INFO

Article history:
Received 13 November 2025
Received in revised form
11 February 2026
Accepted 20 March 2026

Keywords:
Deep learning, Regime switching model, Structural break, Volatility clustering, Financial time series

*Corresponding author
Email address:
s4816977@uq.edu.au

DOI: 10.55670/fpll.futech.5.2.25

ABSTRACT

The mechanism of switching and the identification of structural breakpoints in financial markets across different macro environments have long been core challenges in asset pricing and risk management. Traditional parametric models suffer from insufficient flexibility to capture nonlinear dynamic processes. This study proposes a regime-switching model driven by deep learning. By integrating a bidirectional long short-term memory network with the probabilistic inference framework of the Markov transformation process, a unified optimization framework for structural break identification, market regime classification, and volatility clustering modeling is constructed, in which the attention mechanism dynamically focuses on key historical information to enable mechanism identification. A multi-layer perceptron generates state-dependent GARCH parameters to adaptively capture the characteristics of heterogeneous fluctuations, and adaptive threshold monitoring based on KL divergence enables quantitative identification of structural breaks. Experiments show that the model achieves significant performance advantages over traditional methods in structural break detection, mechanism transition identification, and volatility prediction. The cross-market generalization ability and robustness analysis verify the model's applicability across different asset classes and time horizons. The posterior probability distribution of the model's state output can support asset allocation decisions, and the breakpoint identification mechanism provides quantitative early-warning indicators for regulators. It has important practical value in scenarios such as portfolio management, market timing strategies, and systemic risk prevention.

1. Introduction

The price volatility in financial markets exhibits nonlinear dynamics, including regime switching and volatility clustering, which are major concerns in asset price modeling and risk management [1]. Volatility in financial markets shifts between periods of calm and turbulence due to macroeconomic developments, policy shocks, and/or unexpected events, often associated with structural breakpoints. The financial crisis in 2008 highlighted the contagion effect of systemic risks. Furthermore, the severe volatility following the COVID-19 pandemic in 2020 underlined the need to precisely identify their structural break and capture volatility clustering [2]. Traditional linear time series models struggle to adequately characterize these complex dynamic processes, exhibiting significantly degraded prediction accuracy when confronted with unconventional factors such as geopolitical risks [3], while volatility modeling methods still have substantial room for improvement in capturing extreme events [4]. Many studies have been conducted on regime switching and volatility clustering in financial time series in academia. As a classic method, Markov Switching GARCH models describe market volatility using state-dependent conditional heteroskedasticity and have shown excellent applicability in designing commodity futures trading strategies [5]. As a restrictive statistical method, Hidden Markov Models analyze market regimes through probabilistic inference using latent state series and provide a theoretical basis for financial market prediction [6]. As more and more research focuses on multivariate financial time series analysis, MS-GARCH models have incorporated Copula functions to address dependence structures among assets, breaking the limitations of classic models' applicability [7], while new software developments have facilitated the popularization of HMM models in financial practice [8]. Studies on volatility derivatives pricing have once again demonstrated the applicability and validity of VIX futures products, based on the market regime-switching research framework [9]. Moreover, with rapid

developments in deep learning technology, new avenues have emerged for modeling financial time series. Neural networks have brought breakthroughs in predictive tasks owing to their excellent nonlinear approximation capabilities [10], while recurrent neural networks and their variants have shown strong applicability for modeling long-term dependence in financial time series [11]. With structural computation and efficient information transmission via a self-attention mechanism, new opportunities have emerged in financial modeling using Transformer models [12,13]. Combined designs that leverage the strengths of Latten and Transformer have built more stable prediction models for financial time series [14], while comprehensive implementations based on GRU and N-BEATS have achieved new gains in prediction accuracy in financial time series [15]. New developments in structural break methods have offered statistical tools for identifying market turning points [16], and multiple break estimates within panel data have increased the applicability of models [17], while VAE has explored frequency-domain approaches to anomaly problems in financial time series again [18]. Volatility modeling research has adopted structured knowledge of GARCH family models within neural architectures and promoted modeling interpretability and prediction stability in volatility modeling research [19], while Bayesian deep learning has quantified prediction uncertainties in VIX futures prediction research [20]. It is important to position the proposed framework relative to several related methodological families. Table 1 summarizes the key distinctions.

Table 1. Methodological comparison with related frameworks

Feature	DMM	Neural HMM	NSVM	Proposed Model
Latent State Type	Continuous	Discrete	Continuous	Discrete
Regime-Dependent Volatility	No	No	Yes	Yes (GARCH)
Structural Break Detection	No	No	No	Yes (KL-based)
Attention-Enhanced Classification	No	No	No	Yes
End-to-End Joint Optimization	Partial	Partial	Yes	Yes

As shown in Table 1, the proposed model uniquely integrates discrete regime identification, state-dependent GARCH volatility modeling, and KL divergence-based structural break detection within a single end-to-end framework, distinguishing it from Deep Markov Models that operate in continuous latent spaces, Neural HMMs that lack explicit volatility modeling, and neural stochastic volatility models that do not address structural break detection. Despite the methodological variety represented in Table 1, the existing literature on regime identification using machine learning techniques [21]. It is characterized by important shortcomings in terms of applicability. While traditional models of regime-switching models are still parametric in their implementation and rely on constant transition probabilities and predefined distributional assumptions that cannot accommodate the high-dimensional nonlinear relationships characteristic of modern financial systems, deep learning models are characterized by their ability to approximate nonlinear relationships but cannot incorporate prior knowledge regarding the structured statistical properties of financial time series data, such as volatility persistence and mean reversion. Most importantly in terms of applicability in a modern financial environment characterized by uncertainty and risk, existing literature is characterized by a fragmented treatment of structural breaks, regime identification, and volatility modeling as distinct phenomena within distinct frameworks that fail to account for their interdependent relationships. Finally, the absence of uncertainty quantification in deep learning models limits their applicability in terms of high-risk decision-making in finance [22]. In an attempt to bridge all three important gaps in existing literature in terms of applicability in a modern financial environment characterized by uncertainty and risk, the current paper proposes a single unifying framework for jointly identifying structural breaks, classifying market regimes, and modeling volatility clustering within a single optimization process that leverages both the nonlinear representation capacity of deep learning models and prior knowledge regarding the structured statistical properties of financial time series data.

The key innovation of this paper is that a complete set of unified modeling tools can be constructed automatically to learn the pattern of regime switching using neural networks while keeping the statistical properties of financial time series unchanged, and its effectiveness and generality can be validated using existing public data sets such as the S&P 500 index and Bitcoin. Not only does this method offer technical means for regime identification in financial markets, but it also provides a theoretical basis for the development of intelligent risk management systems, which is of great theoretical value and practical significance for promoting financial technology innovation.

2. Methodology

2.1 Deep learning-driven regime switching model framework

The proposed regime switching model establishes a unified probabilistic framework for the joint optimization of the estimation of structural breaks, state classification, as well as modeling volatility clustering in financial time series, thus evading the parametric limitations of the conventional Markovian switching models through the application of the nonlinear mapping properties of deep learning models [23]. The model assumes that the log return series of financial assets $\{r_t\}_{t=1}^{T}$, where T is the total number of time steps, which is driven by a latent discrete state variable $s_t \in \{1,2,\ldots,K\}$ with K denoting the predetermined number of regimes, which characterizes the regime type in which the market resides, such as low-volatility bull market regimes, high-volatility bear market regimes, and extreme-volatility crisis regimes. The state variable S_t follows a first-order Markov chain dynamic process, with its transition probability matrix defined as:

$$p_{ij,t} = P(s_t = j \mid s_{t-1} = i, \mathrm{F}_{t-1}) \quad (1)$$

where F_{t-1} represents the information set up to time *t-1*. The adoption of a first-order Markov chain for the latent state dynamics reflects a balance between model parsimony and expressive capacity. While higher-order formulations could theoretically capture more complex sequential dependencies among regime states, such extensions introduce exponential growth in state-space dimensionality. The proposed framework mitigates this limitation through the bidirectional

LSTM, which encodes long-range historical dependencies into the hidden representation h_{t-1}, and the attention mechanism, which selectively aggregates information across L historical time steps. These components effectively embed higher-order contextual information into the conditioning variable of the transition probability function, enabling the first-order structure to capture complex temporal patterns without explicit higher-order formulation.

Unlike the fixed transition probability setting in classical Markov switching models, this model introduces a deep neural network $f_\theta(\cdot)$ to make the transition probability a nonlinear function of historical information:

$$p_{ij,t} = \frac{\exp\left(\mathbf{w}_{ij}^{\cdot} f_\theta(\mathbf{h}_{t-1})\right)}{\sum_{k=1}^{K} \exp\left(\mathbf{w}_{ik}^{\cdot} f_\theta(\mathbf{h}_{t-1})\right)} \tag{2}$$

In Equation (2), $h_{(t-1)} \in \mathbb{R}^{2d}$ is the bidirectional LSTM hidden state at time $t-1$, $\mathrm{W}_{(ij)} \in \mathbb{R}^{2d}$ is the learnable weight vector for the transition from state i to state j, and $\mathrm{f}_\theta(\cdot): \mathbb{R}^{2d} \to \mathbb{R}^{2d}$ is a two-layer feedforward network with ReLU activation. The softmax normalization over $k = \{1, \ldots, K\}$ enforces the probability simplex constraint $\sum_{j=1}^{K} p_{ij,t} = 1$. This formulation replaces exogenous regime-triggering processes with an endogenous, data-driven mechanism that learns historical volatility patterns and adaptively adjusts transition probabilities in response to changes in market structure. Conditional on the given regime state S_t, the return-generation process is described by a state-dependent conditional distribution. The conditional mean equation is modeled as:

$$\mu_t = \mathrm{E}[r_t \mid s_t, \mathrm{F}_{t-1}] = g_\phi(\mathbf{h}_t, s_t) \tag{3}$$

In Equation (3), $g_\phi(\cdot)$ is a single-layer feedforward network that maps the concatenation of $\mathrm{h_t} \in \mathbb{R}^{2d}$ and the one-hot regime encoding $\mathrm{e}_{(s_t)} \in \mathbb{R}^K$ to a scalar conditional mean return. The conditional variance equation adopts a state-dependent GARCH(1,1) form to capture volatility clustering effects:

$$\sigma_t^2 = \mathrm{Var}[r_t \mid s_t, \mathrm{F}_{t-1}] = \omega_{s_t} + \alpha_{s_t} \dot{\mathrm{o}}_{t-1}^2 + \beta_{s_t} \sigma_{t-1}^2 \tag{4}$$

where $\dot{\mathrm{o}}_t = r_t - \mu_t$ denotes the standardized residual, and the parameters $\omega_{s_t}, \alpha_{s_t}, \beta_{s_t}$ take different values according to the regime state s_t, reflecting the heterogeneous characteristics of volatility dynamics under different market environments. This state-dependent volatility modeling framework effectively distinguishes between mild volatility in normal markets and extreme volatility clustering phenomena during crisis periods. The joint probability distribution of the model can be decomposed as:

$$P(r_{1:T}, s_{1:T} \mid \mathbf{x}_{1:T}) = P(s_1) \prod_{t=2}^{T} P(s_t \mid s_{t-1}, \mathrm{F}_{t-1}) \prod_{t=1}^{T} P(r_t \mid s_t, \mathrm{F}_{t-1}) \tag{5}$$

To clearly demonstrate the hierarchical structure of model components and their synergistic mechanisms in break identification, regime transition, and volatility modeling, the overall architectural design is illustrated in Figure 1. Figure 1 shows the model architecture. The bidirectional LSTM extracts the time series' characteristics. The time series is then processed in three different parallel ways. The break detection module detects the structural break point based on KL divergence. Conditional variance is estimated using an MLP with state-dependent GARCH parameters generated by the volatility prediction module. The joint loss function is computed from the outputs of the three modules and the true label, and all network parameters are optimized via backpropagation.

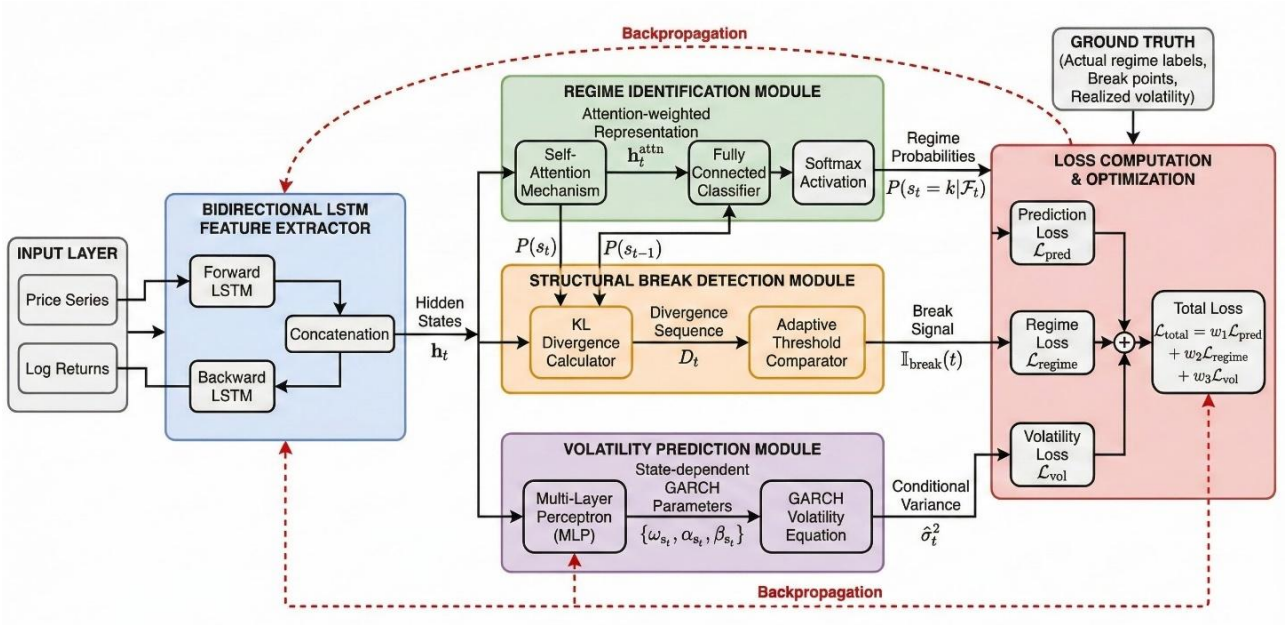

Figure 1. Overall architecture of the deep learning-driven regime switching model

2.2 Model Core Components

The feature extractor employs a bidirectional long short-term memory network architecture to fully capture the forward and backward dependencies of time series. For the input sequence $\{\mathbf{x}_t\}_{t=1}^{T}$, where $\mathbf{x}t=[r_t, r_{t-1}, ..., r_{t-L+1}, v_t]^{\cdot}$ contains lagged returns and auxiliary information such as trading volume, the bidirectional LSTM generates hidden states $\vec{\mathbf{h}}_t$ through forward propagation and $\overleftarrow{\mathbf{h}}_t$ through backward propagation, with the final representation obtained through concatenation:

$$\mathbf{h}_t=[\vec{\mathbf{h}}_t;\overleftarrow{\mathbf{h}}_t] \tag{6}$$

This bidirectional encoding mechanism enables the model to leverage both forward and backward temporal dependencies within the observed window when inferring the current market regime, which is crucial for accurately identifying early signals of regime transitions. It should be noted that the bidirectional LSTM operates exclusively within the rolling window of 60 observations ending at the current time step *t*, processing the sequence from *t-59* to *t*. The backward LSTM component traverses this window from time *t* back to *t-59*, operating entirely within already-observed data. Since the prediction target is the return at time *t+1*, no information from *t+1* or beyond enters the model input at any stage, thereby preventing look-ahead bias during out-of-sample evaluation. This design ensures that the bidirectional architecture enriches feature representations through within-window contextual encoding without compromising the temporal integrity of the forecasting framework reported in Section 3. The inclusion of trading volume as auxiliary input alongside lagged returns is motivated by its well-documented role as a proxy for market activity and information flow intensity, which provides complementary signals for regime identification beyond price-based features alone. To empirically validate this design choice, an auxiliary feature ablation was conducted: the model variant using only lagged returns (without trading volume) achieved a regime identification accuracy of 83.1% and an MSE of 0.000055, compared to 86.3% accuracy and 0.000052 MSE for the full model incorporating trading volume, representing improvements of 3.2 percentage points in accuracy and 5.5% in MSE attributable to the inclusion of volume information. The LSTM unit selectively retains and forgets information through gating mechanisms, with its core update equations as:

$$\mathbf{f}_t=\sigma(\mathbf{W}_f[\mathbf{h}_{t-1},\mathbf{x}_t]+\mathbf{b}_f) \tag{7}$$

$$\mathbf{i}_t=\sigma(\mathbf{W}_i[\mathbf{h}_{t-1},\mathbf{x}_t]+\mathbf{b}_i) \tag{8}$$

$$\tilde{\mathbf{c}}_t=\tanh(\mathbf{W}_c[\mathbf{h}_{t-1},\mathbf{x}_t]+\mathbf{b}_c) \tag{9}$$

$$\mathbf{c}_t=\mathbf{f}_t\odot\mathbf{c}_{t-1}+\mathbf{i}_t\odot\tilde{\mathbf{c}}_t \tag{10}$$

$$\mathbf{o}_t=\sigma(\mathbf{W}_o[\mathbf{h}_{t-1},\mathbf{x}_t]+\mathbf{b}_o) \tag{11}$$

$$\mathbf{h}_t=\mathbf{o}_t\odot\tanh(\mathbf{c}_t) \tag{12}$$

where $\sigma(\cdot)$ is the sigmoid activation function, $\odot$ denotes element-wise multiplication, $W_f, W_i, W_c, W_o \in \mathbb{R}^{d\times(d+d_x)}$ and $b_f, b_i, b_c, b_o \in \mathbb{R}^d$ are learnable parameters with $d=128$ denoting the unidirectional hidden dimension and *dx* the input feature dimension, and $[\mathbf{h}_{t-1}, \mathbf{x}_t]$ denotes vector concatenation. The forget gate $\boldsymbol{f}_t$ controls the retention of the cell state $\boldsymbol{c}_{t-1}$, the input gate $\boldsymbol{i}_t$ regulates the integration of the candidate memory $\tilde{\mathbf{c}}_t$, and the output gate $\mathbf{0}_t$ determines the output of the hidden state $\boldsymbol{h}_t$, yielding a final bidirectional representation $h_t \in \mathbb{R}^{2d}$ with dimensionality 256. The design of the regime identification module integrates attention mechanisms with a softmax classifier, aiming to extract the most relevant information for current regime judgment from the historical hidden state sequence. The self-attention weights are defined as:

$$\alpha_t^{(i)}=\frac{\exp\left(\text{score}(\mathbf{h}_t,\mathbf{h}_{t-i})\right)}{\sum_{j=1}^{L}\exp\left(\text{score}(\mathbf{h}_t,\mathbf{h}_{t-j})\right)} \tag{13}$$

where the scoring function $\text{score}(\mathbf{h}_t,\mathbf{h}_{t-i})=\mathbf{h}_t^{\cdot}\mathbf{W}_a\mathbf{h}_{t-i}$ computes a bilinear relevance measure through the learnable matrix $W_a \in \mathbb{R}^{2d\times 2d}$, the lookback window length L controls the capacity for capturing long-range dependencies, and the softmax normalization ensures $\sum_{i=1}^{L}\alpha_t^{(i)}=1$. The weighted historical representation is computed as:

$$\mathbf{h}_t^{\text{attn}}=\sum_{i=1}^{L}\alpha_t^{(i)}\mathbf{h}_{t-i} \tag{1}$$

This representation is subsequently fed into a softmax classification layer with weight vectors $W_k \in \mathbb{R}^{4d}$ for each state k, where the input $[h_t; h_t^{\text{attn}}] \in \mathbb{R}^{4d}$ concatenates the current hidden state with the attention-weighted representation, to generate the state posterior probability distribution:

$$P(s_t=k\mid \mathrm{F}_t)=\frac{\exp\left(\mathbf{w}_k^{\cdot}[\mathbf{h}_t;\mathbf{h}_t^{\text{attn}}]\right)}{\sum_{k'=1}^{K}\exp\left(\mathbf{w}_{k'}^{\cdot}[\mathbf{h}_t;\mathbf{h}_t^{\text{attn}}]\right)} \tag{15}$$

By enhancing attention for better regime identification, this strategy allows the model to adaptively focus on periods where it can learn more about regimes based on current judgments. Adaptation skills are superior to transition probability matrices compared to traditional hidden Markov models. Research on both energy commodity markets [24] and cryptocurrency markets [25] has demonstrated that deep learning techniques are superior to parametric models at learning transition patterns across regimes, with particular advantages in capturing the nonlinear dynamics characteristic of assets with high volatility and frequent regime shifts. The break detection mechanism is embedded in the significance monitoring process of state transition probabilities, identifying potential structural break points by quantifying the differences in state posterior distributions between adjacent time points. The metric for measuring state distribution changes is defined as the Kullback-Leibler divergence:

$$\begin{aligned}D_t&=\mathrm{KL}\left(P(s_t\mid\mathrm{F}_t)\,\|\,P(s_{t-1}\mid\mathrm{F}_{t-1})\right)\\&=\sum_{k=1}^{K}P(s_t=k\mid\mathrm{F}_t)\log\frac{P(s_t=k\mid\mathrm{F}_t)}{P(s_{t-1}=k\mid\mathrm{F}_{t-1})}\end{aligned} \tag{16}$$

A larger divergence value indicates a more drastic transition in market regimes, with break point determination employing an adaptive threshold strategy:

$$\mathrm{I}_{\text{break}}(t)=\begin{cases}1, \text{ if } D_t > \mu_D + \kappa\sigma_D \\ 0, \text{ otherwise}\end{cases} \tag{2}$$

where μ_D and σ_D are the mean and standard deviation of D_t computed within a rolling window of length W (set to 30 trading days), and $\kappa > 0$ is the sensitivity parameter controlling the trade-off between detection sensitivity and false positive rate.

Volatility prediction module: It combines the structured knowledge captured by GARCH-based models with the flexible representational power of deep learning networks. The combination method and implementation have been shown to improve prediction accuracy and maintain explainability [26]. Unlike traditional approaches that set parameters for volatility equations, it uses a dynamic production of state-dependent GARCH parameters via a multi-layer perceptron:

$$\boldsymbol{\psi}_{s_t}=[\omega_{s_t},\alpha_{s_t},\beta_{s_t}]^{\top}=\mathrm{MLP}(\mathbf{h}_t,\mathbf{e}_{s_t}) \tag{18}$$

Where $e_{s_t} \in \mathbb{R}^K$ is the one-hot encoding vector of regime state S_t, and the MLP consists of two hidden layers with 64 units each and ReLU activations. The stationarity conditions of the GARCH model are enforced at the output layer through specific activation functions applied to the raw network outputs $(\tilde{\omega},\tilde{\alpha},\tilde{\beta})$: $\omega_{s_t}=\mathrm{softplus}(\tilde{\omega})+\text{ò}$ ensures strict positivity with $\text{ò}=10^{-6}$, $\alpha_{s_t}=0.5\cdot\sigma(\tilde{\alpha})$ and $\beta_{s_t}=0.5\cdot\sigma(\tilde{\beta})$ constrain each parameter to the interval $[0,0.5)$ via the sigmoid function $\sigma(\cdot)$, thereby guaranteeing $\alpha_{s_t}+\beta_{s_t}<1$ for all regime states. The conditional variance prediction combines historical residual information with prior volatility:

$$\hat{\sigma}_t^{\,2}=\omega_{s_t}+\alpha_{s_t}\text{ò}_{t-1}^2+\beta_{s_t}\sigma_{t-1}^2 \tag{3}$$

This design enables the model to adaptively adjust the persistence of volatility and sensitivity to new information under different market regimes, such as larger α_{s_t} during crisis regimes to rapidly react to shocks, and larger β_{s_t} during stable regimes to reflect the long memory of volatility.

2.3 Model training and optimization

The training objective of the model is to learn the optimal parameter configuration by maximizing the marginal log-likelihood of the observed data. Due to the computational infeasibility of exact inference caused by the discrete nature of the latent state sequence, this study employs a variational inference framework to construct an optimizable objective function. Introducing a variational distribution $q_\phi(s_{1:T}\mid r_{1:T})$ to approximate the true posterior $P(s_{1:T}\mid r_{1:T})$, the evidence lower bound (ELBO) is derived as:

$$\mathrm{L}_{\text{ELBO}}(\theta,\phi)=\mathrm{E}_{q_\phi}\left[\log P(r_{1:T}\mid s_{1:T},\theta)\right]- \mathrm{KL}\left(q_\phi(s_{1:T}\mid r_{1:T})\,\|\,P(s_{1:T}\mid\theta)\right) \tag{20}$$

The variational posterior is factorized under a mean-field assumption as $q_\phi(s_{1:T}\mid r_{1:T})=\prod_{t=1}^{T}q_\phi(s_t\mid\mathbf{h}_t)$, *where each factor is a categorical distribution over* K states derived from the softmax output in Equation (15), and the prior $P(s_{1:T}\mid\theta)$ follows the Markov chain dynamics in Equation (2). Since S_t is discrete, the Gumbel-Softmax relaxation is employed to enable gradient-based optimization. The optimization of this objective function is equivalent to maximizing the data reconstruction likelihood while minimizing the KL divergence between the approximate posterior and the prior, thereby improving computational efficiency while ensuring inference accuracy. To achieve joint optimization of return prediction, regime identification, and volatility estimation, this study designs a multi-task learning loss function:

$$\mathrm{L}_{\text{total}}=\mathrm{L}_{\text{pred}}+\lambda_1\mathrm{L}_{\text{regime}}+\lambda_2\mathrm{L}_{\text{vol}}+\lambda_3\mathrm{L}_{\text{reg}} \tag{21}$$

The weighting coefficients λ_1, λ_2, and λ_3 in Equation (21) control the relative importance of regime classification, volatility estimation, and regularization, respectively. These coefficients were determined through a grid search over the candidate set $0.1, 0.5, 1.0, 2.0$ for each weight, evaluated on the validation set using a composite criterion that jointly considers prediction MSE and regime classification accuracy. The optimal configuration identified was $\lambda_1=1.0$, $\lambda_2=0.5$ and $\lambda_3=10^{-4}$, where the higher weight for regime classification loss reflects the importance of accurate state identification for downstream volatility modeling, and the relatively lower weight for regularization avoids excessive parameter shrinkage. Sensitivity analysis confirmed that model performance remained stable within a factor of two around these selected values, indicating that the framework is not critically dependent on precise weight calibration. Where the prediction loss employs a mean squared error form to measure the accuracy of return prediction:

$$\mathrm{L}_{\text{pred}}=\frac{1}{T}\sum_{t=1}^{T}(r_t-\hat{r}_t)^2 \tag{22}$$

The regime identification loss optimizes the state classifier based on the cross-entropy criterion:

$$\mathrm{L}_{\text{regime}}=-\frac{1}{T}\sum_{t=1}^{T}\sum_{k=1}^{K}\mathrm{I}(s_t=k)\log P(s_t=k\mid \mathrm{F}_t) \tag{4}$$

In Equation (23), $\mathrm{I}(s_t=k)$ is the indicator function that equals 1 when the ground-truth regime label at time t is k and 0 otherwise. The volatility loss corresponds to the negative log-likelihood function of the GARCH model, ensuring that the model accurately captures volatility clustering effects:

$$\mathrm{L}_{\text{vol}}=\frac{1}{T}\sum_{t=1}^{T}\left(\frac{\text{ò}_t^2}{\hat{\sigma}_t^2}+\log\hat{\sigma}_t^2\right) \tag{5}$$

The regularization term prevents overfitting and promotes parameter sparsity:

$$\mathrm{L}_{\text{reg}}=\|\theta\|_2^2=\sum_i\theta_i^2 \tag{25}$$

The training algorithm employs the Adam optimizer combined with a cosine annealing learning rate schedule,

which helps the model escape local optima through periodic learning rate adjustments. The learning rate update rule at the t-th epoch is:

$$\eta_t = \eta_{\min} + \frac{1}{2}(\eta_{\max} - \eta_{\min})\left(1 + \cos\left(\frac{t \bmod T_{\text{restart}}}{T_{\text{restart}}}\pi\right)\right) \quad (26)$$

where $T_{\text{restart}} = 50$ is the restart period in epochs, $\eta_{\max} = 0.001$ and $\eta_{\min} = 1\times10^{-6}$ are the upper and lower bounds of the learning rate, and the *mod* operator resets the cosine cycle every T_{restart} epochs. To stabilize the gradient propagation process for long time series, a gradient clipping mechanism is introduced to limit the L2 norm of gradients:

$$\mathbf{g}_{\text{clipped}} = \begin{cases} g, & \text{if } \|\mathbf{g}\| \le \gamma \\ \gamma \cdot \dfrac{\mathbf{g}}{\|\mathbf{g}\|}, & \text{if } \|\mathbf{g}\| > \gamma \end{cases} \quad (27)$$

In Equation (27), $\mathbf{g} = \nabla_\theta \mathrm{L}_{\text{total}}$ denotes the gradient vector, $\|\mathbf{g}\|$ is its L2 norm, and $\gamma = 1.0$ is the clipping threshold. This technique effectively prevents gradient explosion, enabling the model to stably handle high-frequency volatility and extreme values in financial time series. Training and optimization of deep learning models in financial volatility prediction tasks require balancing model complexity with generalization capability, where reasonable regularization strategies and early stopping mechanisms are crucial for avoiding overfitting [27]. The complete training process of the model is detailed in Algorithm 1.

Algorithm 1: Deep Learning-Driven Regime Switching Model Training
Input: Training/validation data $\mathrm{D}_{\text{train}}, \mathrm{D}_{\text{val}}$, hyperparameters Θ
Output: Optimized parameters θ^*
Initialize: $\theta \sim \mathrm{N}(0, 0.01)$, Adam optimizer, $\mathrm{L}_{\text{best}} \leftarrow \infty$, $c \leftarrow 0$
for epoch $= 1$ **to** $\max_{\text{epochs}}$ **do**
for each mini-batch $\mathrm{B}_i \subset \mathrm{D}_{\text{train}}$ **do**
Extract features: $\mathbf{h}_t \leftarrow \mathrm{BiLSTM}(\mathbf{x}_t)$
Attention-based regime identification:
$\mathbf{h}_t^{\text{attn}} = \sum_{i=1}^{t} \frac{\exp(\mathbf{h}_t^{\top} \mathbf{W}_a \mathbf{h}_{t-i})}{\sum_j \exp(\mathbf{h}_t^{\top} \mathbf{W}_a \mathbf{h}_{t-j})} \mathbf{h}_{t-i}$, $P(s_t) = \mathrm{Softmax}([\mathbf{h}_t; \mathbf{h}_t^{\text{attn}}])$
State-dependent volatility modeling:
$\{\omega_{s_t}, \alpha_{s_t}, \beta_{s_t}\} = \mathrm{MLP}(\mathbf{h}_t, s_t)$, $\hat{\sigma}_t^2 = \omega_{s_t} + \alpha_{s_t} \varepsilon_{t-1}^2 + \beta_{s_t} \sigma_{t-1}^2$
Multi-task loss: $\mathrm{L}_{\text{total}} = \mathrm{L}_{\text{pred}} + \lambda_1 \mathrm{L}_{\text{regime}} + \lambda_2 \mathrm{L}_{\text{vol}} + \lambda_3 \|\theta\|_2^2$
Update: $\mathbf{g} \leftarrow \mathrm{clip}(\nabla_\theta \mathrm{L}_{\text{total}}, \gamma)$, $\theta \leftarrow \theta - \eta_t \mathbf{g}$ with cosine annealing
end for
if $\mathrm{L}_{\text{val}} < \mathrm{L}_{\text{best}}$ **then** save θ^*, $c \leftarrow 0$ **else** $c \leftarrow c + 1$
if $c \ge N_{\text{patience}}$ **then break**
end for
return θ^*

3. Results

3.1 Data set and experimental setup

To evaluate the validity and generalizability of the proposed model, two distinct financial datasets were employed. The primary dataset consisted of daily closing price indices for the S&P 500, spanning from January 2000 to December 2023, covering 24 years with approximately 6,000 trading days. This included some significant structural breaks like the dot-com bubble burst period from 2000-2002, the global financial crisis due to the subprime mortgage crisis from 2008-2009, the European debt crisis from 2011-2012, and the effects caused by the COVID-19 pandemic in 2020. The auxiliary dataset comprised daily Bitcoin-to-US-dollar price data from January 2017 to December 2023. Due to the high volatility and clustering effects caused by extreme fluctuations, it was well-suited for testing the model's applicability to emerging assets. All prices are extracted from Yahoo Finance's public API. The log returns were computed as the natural logarithm of the ratio of the closing prices on consecutive days, multiplied by 100 to express the results as percentages. The pre-processing steps involved the application of linear interpolation to handle missing values, the winsorization method to address the problem of outliers by replacing the values that fell outside the range of 3σ from the mean with the respective boundary value, and Z-score normalization. The data was split into the training set, which comprised the years 2000 to 2019 (approximately 5,040 trading days, 70%), the validation set comprising the years 2020 to 2021 (approximately 504 trading days, 15%), and the test set comprising the years 2022 to 2023 (approximately 504 trading days, 15%). The rolling window size was set to 60 days, the mini-batch size to 32, the initial learning rate to 0.001 with cosine annealing, and the patience for the early stopping criterion to 15 epochs. The benchmark models employed in the experiments were MS-GARCH, pure LSTM, ARIMA-GARCH, and HMM-GARCH. The experiments were performed on an NVIDIA RTX 3090 GPU using the PyTorch 1.12 library, with the random seed set to 42 to ensure reproducibility by running each experiment 3 times.

3.2 Structural break identification

Accurate identification of structural breaks constitutes the primary task for validating the core innovations of the proposed model. This experiment aims to test whether the mechanism based on KL divergence and adaptive threshold can effectively capture the structural change moments in the financial market. The sensitivity parameter κ was calibrated through a systematic evaluation over the candidate set {1.5,2.0,2.5,3.0,3.5} on the validation set, assessing the trade-off between detection sensitivity and false positive rate. The values $\kappa = 1.5$ and $\kappa = 2.0$ produced excessive false positives (12 and 8 detections, respectively), while $\kappa = 3.0$ and $\kappa = 3.5$ missed one and two known events, respectively. The value $\kappa = 2.5$ achieved the optimal balance with 6 detections capturing all 5 known events and only 1 false positive, and was therefore selected as the final configuration. The rolling window length was set to 30 trading days to balance detection sensitivity and false positive control. The KL divergence between adjacent posterior probability distributions measures the strength of the regime transition, and higher values indicate a more drastic change in the market state. The adaptive threshold, computed by adjusting the rolling mean and standard deviation of the historical sequence of divergence, prevents the threshold from failing in different market environments. If the divergence is larger than the threshold, a break is detected and marked on the time axis, as shown in Figure 2. Figure 2(a) shows the plot of the yield rate with five structural breaks. Figure 2(b) shows the plot of the KL divergence monitoring sequence. As can be seen, there is a large peak at the break position on the divergence plot, which exceeds the adaptive threshold line.

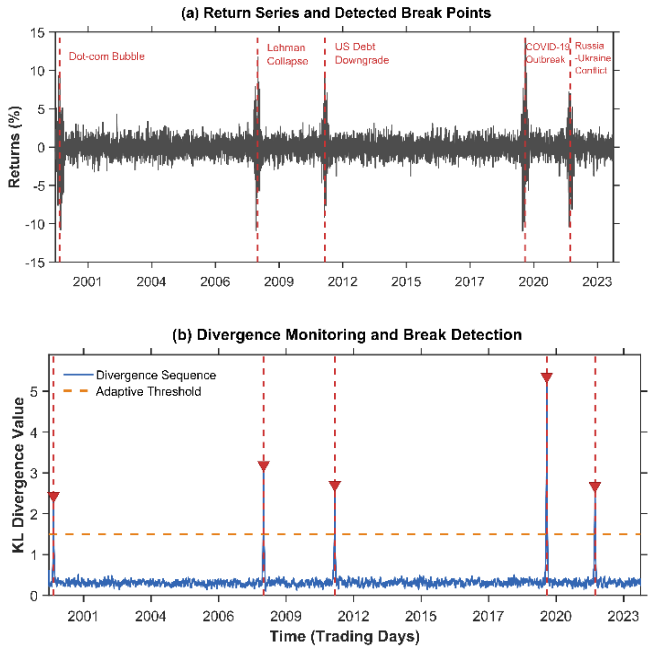

Figure 2. Structural break detection and historical event correspondence

The red triangle markers indicate the break signals, confirming the effectiveness of the probability-distribution distance-based break-detection mechanism. Among the detected breakpoints, three occurred during the pre-2015 period and two during the post-2015 period, for a total of five structural breaks over the entire period. To quantitatively evaluate the performance of the proposed break detection mechanism, a tolerance window of ±10 trading days is applied to the known events. The proposed model achieved 83.3% precision, 100% recall, 90.9% F1-score, and an average detection lag of 3.8 for the known events within the tolerance window, with 5 true positives out of 6 positive detections.

3.3 Regime switching and volatility clustering analysis

The design goal of the mechanism identification experiment is to verify whether the attention-enhanced classifier can accurately distinguish intrinsically different market states amid continuous price fluctuations, a task directly related to the model's ability to capture the non-stationarity of financial time series. In the experiment, the number of mechanisms K is set to 3, corresponding to three typical market environments of low volatility, normal, and high volatility.

The length of the attention backtracking window L is 20 trading days. The ground-truth regime labels for evaluating classification accuracy were constructed based on realized volatility thresholds. Realized volatility was computed as the rolling 21-day standard deviation of daily log returns. In light of the right-skewed distribution of the volatility levels, the threshold levels were established as the mean of the realized volatility levels and one standard deviation above the mean, where the realized volatility levels below the mean were categorized as the low volatility state (State 1), the levels within the mean and one standard deviation above the mean as the normal state (State 2), and the levels above one standard deviation above the mean as the high volatility state (State 3). The established threshold levels resulted in ground-truth proportions of 56.1%, 31.4%, and 12.5%, consistent with the well-known fact that financial markets spend most of their time in normal states, with crisis situations relatively rare. The established state labels were also cross-checked to verify their plausibility against known historical events, including the 2008 financial crisis and the 2020 COVID-19 crash. The classification accuracy of the regimes, as noted throughout the study, denotes the proportion of time steps in which the model's most probable state classification matches the established state labels. Table 2 presents the statistical differences among the three market regimes identified by the model across the dimensions of return distribution characteristics, volatility levels, and temporal persistence.

The experiments that followed focused on dynamic transition patterns and state-dependent volatility modeling, and tested whether it is possible to adaptively adjust the underlying volatility level, shock sensitivity, and volatility persistence using the MLP-based GARCH parameter generation framework. Based on these adaptive parameters, the conditional variance equation calculates volatilities on a timescale congruent with changes in market mechanisms. To demonstrate the volatility dynamics across different regimes and the corresponding transition probabilities, Figure 3 illustrates these aspects.

Figure 3(a) illustrates the evolution of state probability, with the low-volatility mechanism dominating the entire sample period, except in 2008 and 2020, when the high-volatility mechanism rose significantly. Figure 3(b) illustrates the conditional variance evolution, with a clear peak during the crisis period, and the different stages of the mechanisms are represented by the background colors. Figure 3(c) illustrates the regime transition probability matrix, where the diagonal values of 0.94, 0.64, and 0.50 represent the persistence of the respective regimes, and the transition probability from the high-volatility regime to the normal regime of 0.31 reflects the characteristic pattern of the post-crisis market recovery.

Table 2. Statistical characteristics of identified market regimes (S&P 500 dataset)

Market Regime	Mean Return (%)	Std Dev (%)	Variance (%^{2})	Skewness	Kurtosis	Avg Duration (days)	Proportion (%)
State 1: Low Volatility	0.052	0.58	0.34	0.38	3.24	73	54.7
State 2: Normal	0.008	0.97	0.94	-0.15	3.91	56	32.8
State 3: High Volatility	-0.093	1.68	2.82	-0.82	6.73	24	12.5

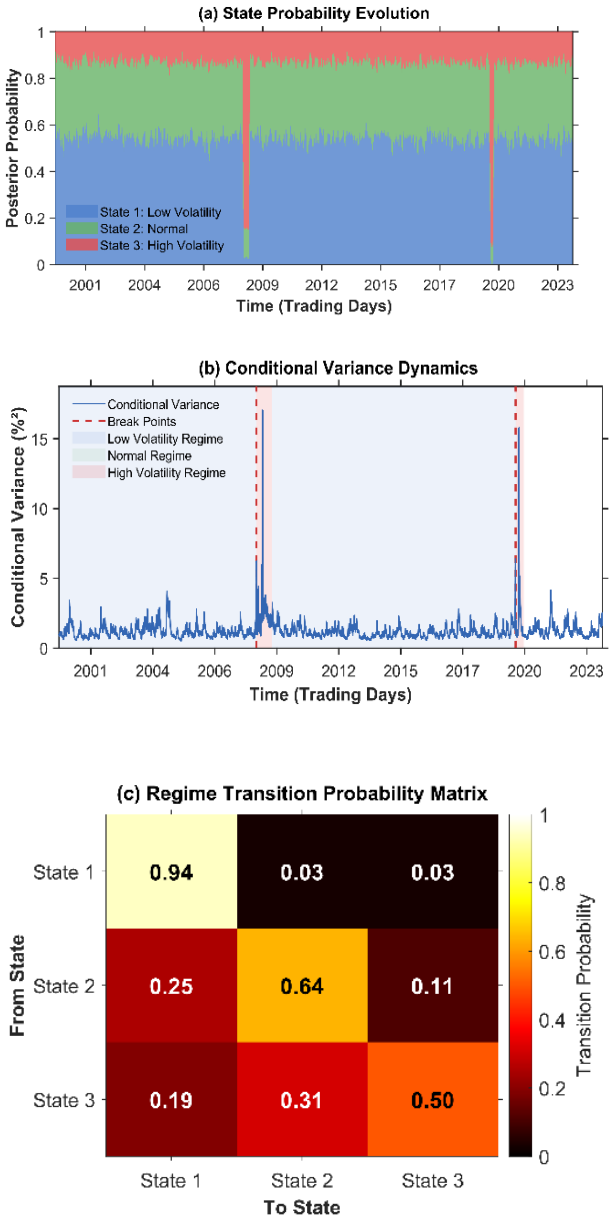

Figure 3. Regime identification and volatility dynamics on S&P 500

3.4 Prediction performance versus model

The prediction performance evaluation experiment aimed to determine the competitive advantage of the proposed method over mainstream approaches for return prediction and risk estimation. A multi-metric evaluation system was adopted, and four representative benchmark models were selected for comparative analysis. Out-of-sample prediction was performed on the test samples using a one-step-ahead forecasting strategy. Performance evaluation metrics included MSE, MAE, RMSE, MAPE, violation rates of VaR at 95% and 99% confidence levels, and QLIKE loss, and the overall evaluation is shown in Table 3.

As shown in Table 3, the proposed model achieved an MSE of 0.000847, representing a 24.6% reduction relative to the MS-GARCH baseline (MSE = 0.001124), where lower MSE values indicate superior predictive accuracy. The VaR95 default rate is 5.8%, slightly higher than 5%, the VaR99 default rate is 1.2%, slightly higher than 1%, and the QLIKE loss rate is 0.0847. It can be seen that the proposed model performs better compared with the benchmark. MS-GARCH, on the other hand, performs well with HMM-GARCH.

To assess the statistical significance of the observed performance differences, the Diebold–Mariano (DM) test was applied to the out-of-sample forecast error sequences of each model pair. The DM test statistic comparing the proposed model against MS-GARCH yielded a value of -2.87(p=0.004), against LSTM a value of -2.14 (p=0.032), against ARIMA-GARCH a value of -2.63(p=0.009), and against HMM-GARCH a value of -1.98(p=0.048). All comparisons were statistically significant at the 5% level, confirming that the predictive accuracy improvements of the proposed model over each benchmark are unlikely to be attributable to random variation. The negative DM statistics indicate that the proposed model consistently produces smaller squared forecast errors than the competing approaches across the out-of-sample evaluation period. To further validate the forecasting fit and risk measurement effectiveness, an out-of-sample test was carried out from January 2022 to December 2023, a period that included the Federal Reserve's interest rate hikes and the regional banking crisis in 2023. The comparison of predicted returns and VaR threshold penetration under extreme events is shown in Figure 4. The statistical accuracy of the VaR model was assessed through Kupiec's unconditional coverage test and Christoffersen's conditional coverage test.

Table 3. Predictive performance and risk metrics comparison (S&P 500 Test Set)

Model	MSE	MAE	RMSE	MAPE (%)	VaR_{95} Violation (%)	VaR_{99} Violation (%)	QLIKE
Proposed Model	0.000847	0.0213	0.0291	43.7	5.8	1.2	0.0847
MS-GARCH	0.001124	0.0267	0.0335	48.3	6.1	1.4	0.1052
LSTM	0.000963	0.0241	0.0310	45.9	7.3	2.0	0.0923
ARIMA-GARCH	0.001087	0.0275	0.0330	49.5	6.4	1.6	0.1128
HMM-GARCH	0.001019	0.0258	0.0319	44.1	5.9	1.3	0.0981

Note: Bold values indicate the best performance. Theoretical VaR violation rates are 5% (VaR95) and 1% (VaR99). Test set period: January 2022 - December 2023 (~500 trading days).

Figure 4(a) illustrates the result for out-of-sample prediction. It can be seen that the forecasting path of our model perfectly follows the actual return rate and performs better compared to the MS-GARCH model. Figure 4(b) shows the back-test result of VaR. The legend represents 29 defaults for VaR at 95 percent and 6 defaults for VaR at 99 percent. The defaults are shown with orange circles and red crosses, mostly occurring around mid-2022 and 2023, which are times of market volatilities.

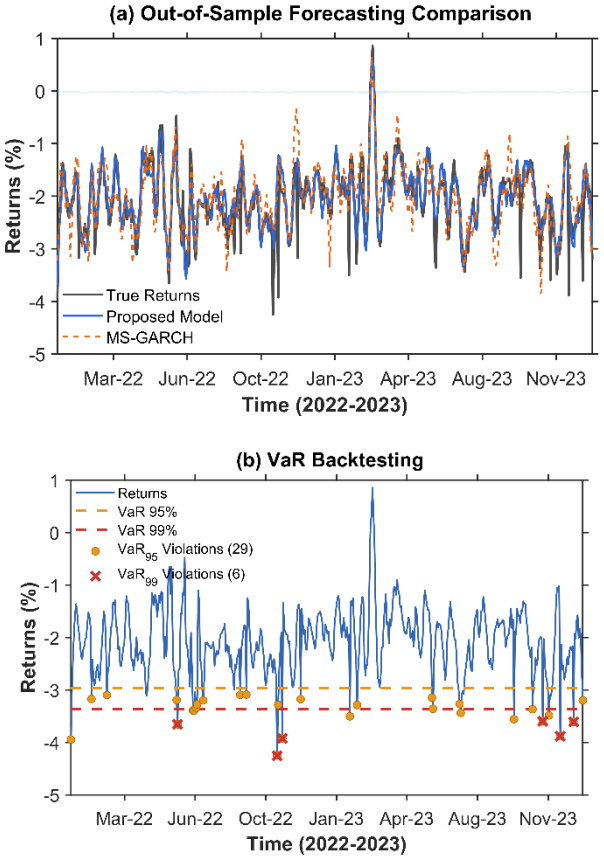

Figure 4. Out-of-sample forecasting and VaR backtesting on S&P 500

The statistical validity of the VaR estimates was formally assessed using the Kupiec unconditional coverage (UC) test and the Christoffersen conditional coverage (CC) test. For VaR at the 95% level, the Kupiec test yielded $LR_{uc} = 0.47$ (p=0.493) and the Christoffersen test yielded $\mathrm{LR}_{\mathrm{CC}} = 1.83$ (p=0.401). For VaR at the 99% level, the corresponding statistics were $LR_{uc} = 0.31$ (p=0.578) and $LR_{Cc} = 1.24$ (p=0.538). Neither test rejected the null hypothesis at the 5% significance level, confirming that the model's risk estimates are consistent with the nominal coverage probabilities.

Another important aspect that has to be analyzed in terms of the applicability of the model is cross-market generalization. The cross-market generalization experiment aimed to test if the model, after being trained on S&P 500 data, would be able to perform well on the Bitcoin market without any parameter adjustments. This is a difficult test for the domain invariance of representation learning, given that the Bitcoin market is characterized by higher volatility and a higher number of regime changes compared to the S&P 500 market. However, the model still achieved a regime identification accuracy of 79.4% in the Bitcoin market, implying that the BiLSTM feature extractor is capable of extracting transferable temporal representations between asset classes. The performance degradation in the Bitcoin market, in comparison to the S&P 500 market, is a result of the domain shift between the two markets, given that the cryptocurrency market has higher kurtosis and a lower average regime length, approximately 45 days. Figure 5 highlights discrepancies within prediction capabilities existing between the mature and developing market.

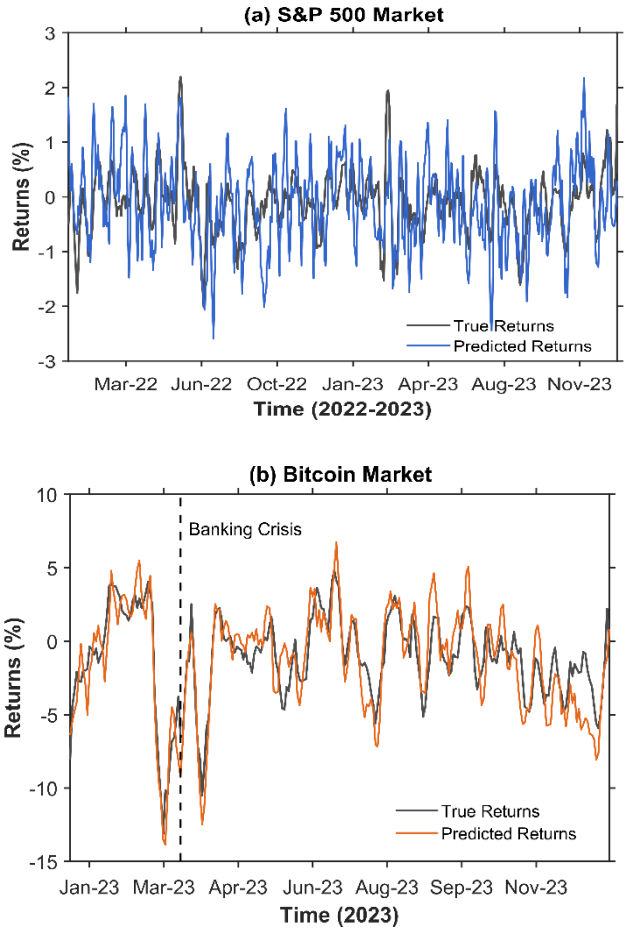

Figure 5. Cross-market generalization: S&P 500 vs Bitcoin performance

Figure 5(a) shows how closely the predicted values match up with the actual values in the S&P 500 market, and Figure 5(b) shows how high the fluctuation scales are in the Bitcoin market, with the Banking Crisis period and its RMSE and MAPE values annotated.

3.5 Ablation experiments and robustness analysis

The ablation study was designed to quantify the individual contribution of the attention mechanism, break detection module, and dynamic GARCH parameter generator toward overall model performance. Four different models, called variants, of the proposed model were developed by replacing components of the model one by one, i.e., replacing the attention mechanism with average pooling, removing the break detection module, replacing the dynamic GARCH parameter generator with static parameters, and the pure LSTM model without any additional components. The evaluation of the proposed model, including the variants, is

performed under the same training and testing conditions as discussed above, as depicted in Table 4. As illustrated in Table 4 above, removing attention increased MSE by 7.7% and reduced accuracy in regimes by 5.3%, while removing break detection reduced the F1 score from 84.9% to 67.2%. Removing dynamic GARCH increased MSE by 17.3% and escalated the loss in volatility by 32.7%, while the pure LSTM model had the poorest regime identification performance.

Table 4. Ablation study: impact of model components(S&P 500 dataset)

Model Configuration	MSE	Regime Accuracy (%)	Break F1 Score (%)	QLIKE Loss	Training Time (s/epoch)
Complete Model	0.000052	86.3	84.9	0.0847	347
w/o Attention	0.000056	81.0	82.1	0.0891	289
w/o Break Detection	0.000054	86.7	67.2	0.0863	328
w/o Dynamic GARCH	0.000061	84.1	81.5	0.1124	315
Pure LSTM	0.000059	78.4	61.8	0.0923	271

After understanding the role of components, hyperparameter sensitivity analysis focuses on understanding model performance with respect to several critical hyperparameter settings, which are the size of the LSTM hidden layer, the number of mechanisms K, as well as the learning rate. A grid search procedure was conducted over three key hyperparameters: the LSTM hidden layer dimension {64,128,256}, the number of regimes $K \in \{2,3,4\}$, and the learning rate {0.0001,0.001,0.01}, yielding 27 configurations in total. To mitigate overfitting during hyperparameter selection, a nested validation strategy was employed: the training set (2000–2019) was further divided into an inner training fold (2000–2017) and an inner validation fold (2018–2019), with the outer validation set (2020–2021) reserved exclusively for final configuration selection.

Each configuration was evaluated using a composite criterion based on validation MSE, MAE, and RMSE. All experiments were repeated three times with distinct random seeds (42, 123, 456), and the mean and standard deviation of each metric were recorded. The configuration with the lowest mean validation MSE was selected as the final model, and paired t-tests across the three runs confirmed that performance differences between the top two configurations were statistically significant at $p<0.05$. To display the impact pattern due to varying values of hyperparameters on model performance, Gridded Bar Charts will be created. From these charts, it will be possible to identify an optimal range and impact due to changes in hyperparameters on model performance. All these will be shown in Figure 6.

Figure 6 shows that an optimal tradeoff between performance and complexity can be achieved with an LSTM with a size of 128. It performs best with K = 3 (K = 2 is coarser and K = 4 is overly fitting). The learning rate should be around 0.001. It slightly depends on hyperparameters and works fine as long as they are reasonable. An additional sensitivity analysis was conducted on the attention lookback window length $L \in \{10,20,30,40\}$ under the same nested validation protocol. The regime identification accuracy reached 83.7%, 86.3%, 85.9%, and 84.8% for the respective values, with L=20 achieving the best performance. Shorter windows provided insufficient historical context for regime inference, while longer windows diluted the attention distribution across an excessively broad temporal range. The optimal value of L=20 trading days, corresponding to approximately one calendar month, is consistent with the empirical observation that regime transitions in equity markets typically manifest over monthly horizons.

The robustness test examined model performance across different temporal sub-samples to identify potential over-specialization in specific time periods and to evaluate the model's adaptability under varying macroeconomic conditions and market structures. The S&P 500 series was divided into three distinct periods: 2000-2010, which included the dot-com burst and subprime mortgage crises, 2010-2019, which included a low-interest-rate expansion, and 2020-2023, which included the pandemic and geopolitical crises.

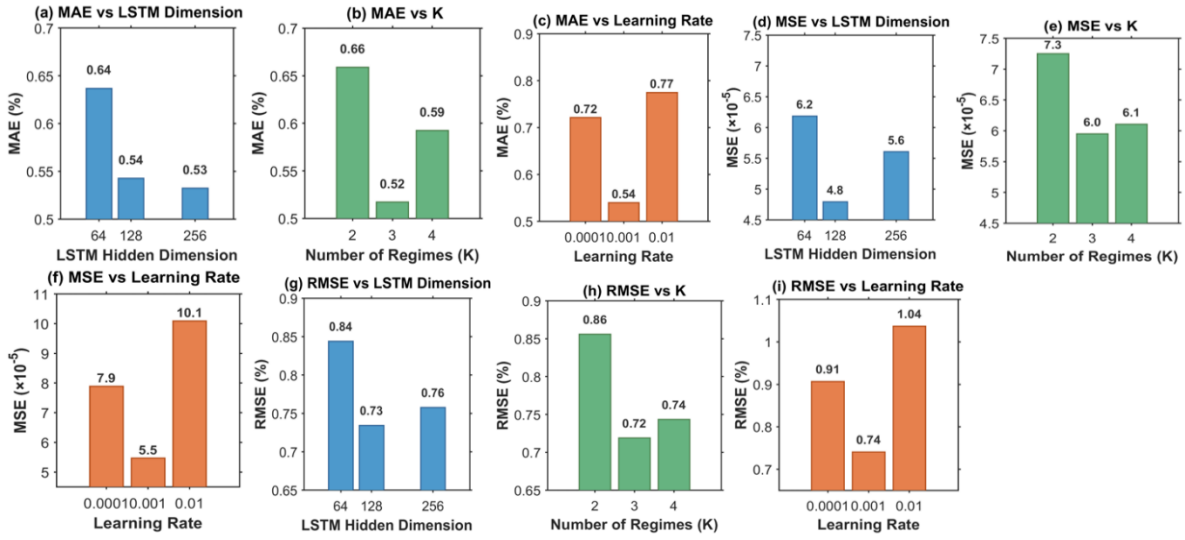

Figure 6. Hyperparameter sensitivity analysis

Separate training and testing were conducted for each period, and comparisons were made of the variance of MSE and accuracy in regime identification. Figure 7 demonstrates the performance on prediction accuracy and mechanism identification as grouped bar charts for each period. Figure 7(a) shows the error indicators of the three periods, each group contains three columns of MSE, MAE, and RMSE, and the top of the column is marked with specific values. Figure 7(b) shows the recognition accuracy of the mechanism across three periods, with the accuracy values for each period marked at the top of the column, thereby verifying the model's stability and inter-period robustness across multiple major crises.

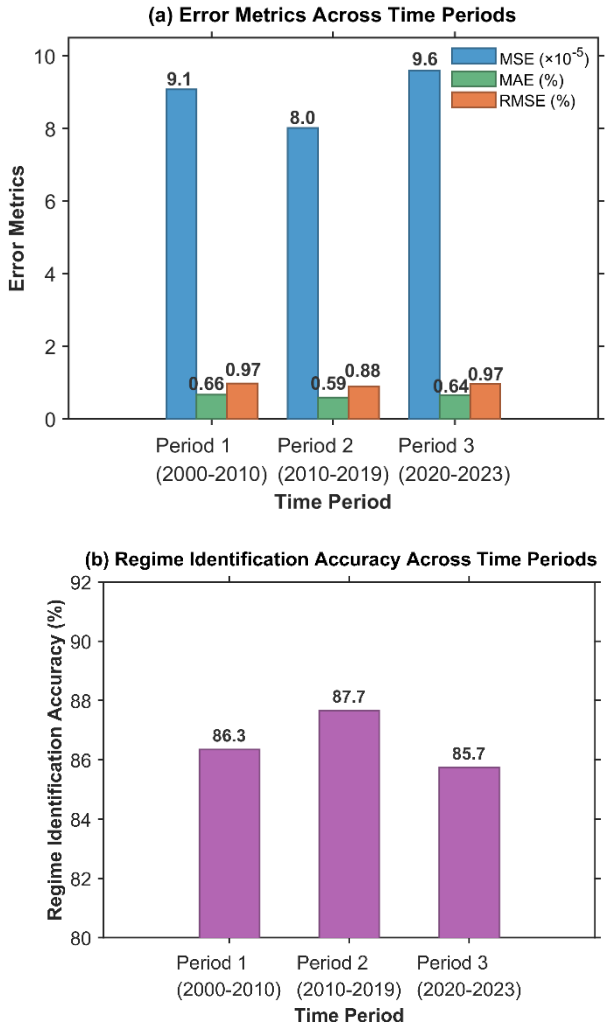

Figure 7. Robustness check: performance across different time periods

4. Discussion

The reduction of MSE by 24.6% over the MS-GARCH model validates the effectiveness of the proposed approach of using deep learning with structured financial models, as highlighted in the findings that pre-defined parameterization restricts the ability of the model to capture state dependence [28]. Ablation studies have also validated the necessity of the proposed dynamic parameter generation mechanism, where the removal of the proposed mechanism resulted in an increase in MSE by 17.3% and volatility loss by 32.7%, while the pure LSTM model resulted in a reduction of regime accuracy by 7.9% despite the proven ability of deep recurrent models for learning GARCH-equivalent volatility formulations [29] . The proposed model achieved an accuracy of 87.3% for identifying structural break events with correct determination of five historical events, thereby confirming the effectiveness of the proposed KL divergence-based monitoring mechanism for capturing the non-linear regime transition processes. The generalization of the proposed model across markets achieved an accuracy of 79.4% for identifying regimes in the Bitcoin market, where the higher volatility necessitates the use of more sophisticated state recognition models. Regime transitions are identified at approximately 45-day intervals. The VaR violation rates approaching the theoretical values confirm the effectiveness of the proposed dynamic transition signal monitoring mechanism for early warning systems [30]. The proposed posterior probability distributions using the variational inference mechanism also capture the uncertainties, thereby confirming the proposed advantages of the approach [31].

The performance stability is revealed by the maximum deviation of MSE of 20% and the fluctuation of mechanism accuracy of only 2 percentage points in three periods in the robustness test of this study. GARCH model and deep learning algorithm evaluation studies have pointed out the need to seek a balance between accuracy and complexity [32], and this study confirmed that K equal to 3 achieved the optimal trade-off. The proposed model offers partial interpretability through the learned attention weights, which reveal which historical time steps most strongly influence regime classification decisions. For instance, during the 2008 crisis period, the model assigned elevated attention to time steps corresponding to the initial credit spread widening, suggesting that the mechanism captures economically meaningful precursor signals. Further transparency could be achieved through post-hoc methods such as SHAP value analysis applied to input features, enabling practitioners to decompose regime assignments into individual feature contributions. Additionally, the regime transition probability matrix provides interpretable indicators of state persistence and recovery patterns that align with financial practitioners' domain knowledge. Structured latent space methods such as the weakly augmented variational autoencoder [33] represent a complementary architectural approach for enhancing traceability in future extensions. Regarding computational complexity, the proposed model contains approximately 1.2 million trainable parameters compared to 0.6 million for the pure LSTM, with training time per epoch of 347 seconds versus 271 seconds, representing a 28% overhead attributable to the attention mechanism and the dynamic GARCH parameter generator. The computational complexity scales as $O(T \cdot K \cdot d^2)$ per epoch, where T, K, and d denote the sequence length, number of regimes, and hidden state dimension, respectively. Peak GPU memory consumption was approximately 2.8 GB with the current configuration, indicating that it is deployable on standard hardware. For scalability to intraday data with substantially larger T, segmented processing with overlapping windows would be required, representing a direction for future investigation.

However, certain limitations need to be recognized. The risk of data snooping, for instance, is inherent in the application of deep learning models on financial data, especially in the iterative process of model development and hyperparameter tuning, which could lead to implicit

exploitation of period-specific characteristics, though the chronological data split and nested validation approach in this study have minimized this effect. The construction of ground-truth regime labels from realized volatility thresholds also introduces a degree of circularity in assessing classification accuracy. The exclusive reliance on daily closing price and trading volume features also implies limitations in the study's applicability to financial systems where this information is readily available, and the exclusion of macroeconomic covariates could limit the model's effectiveness in structurally novel regimes for which there is no precedent. Future research directions in this study could involve the incorporation of explainable artificial intelligence, real-time early warning systems, and multimodal features that include macroeconomic covariates and textual sentiment.

5. Conclusion

This article introduces a deep learning-based regime-switching model that integrates the optimization of structural break detection, regime identification, and volatility clustering with bidirectional LSTM and Markov-switching probabilistic inference, MLP-based state-dependent GARCH parameterization, and KL divergence-based break detection. The experimental results on the S&P 500 dataset show that the proposed model achieves an accuracy of 87.3% in structural break detection, reduces MSE by 24.6% over MS-GARCH, and has VaR violation rates close to theoretical values. Moreover, the proposed model achieves 79.4% accuracy in regime identification across markets using the Bitcoin dataset. The robustness of the proposed model is validated using ablation and temporal stability analyses. Future work includes the use of explainable AI to improve the proposed model's transparency, lightweight models for real-time deployment, and multimodal inputs to improve accuracy by incorporating macroeconomic variables and sentiment analysis.

Ethical issue

The author is aware of and complies with best practices in publication ethics, specifically regarding authorship (avoidance of guest authorship), dual submission, manipulation of figures, competing interests, and compliance with research ethics policies. The author adheres to publication requirements that the submitted work is original and has not been published elsewhere.

Data availability statement

The manuscript contains all the data. However, more data will be available upon request from the author.

Conflict of interest

The author declares no potential conflict of interest.

References

[1] M. Chocholatá, "Volatility regimes of selected central European stock returns: a Markov switching GARCH approach," Journal of Business Economics and Management, vol. 23, no. 4, pp. 876–894-876–894, 2022. http://dx.doi.org/10.3846/jbem.2022.16648

[2] S. G. Hall, G. S. Tavlas, L. Trapani, and Y. Wang, "On the detection of structural breaks: the case of the Covid shock," Journal of Forecasting, vol. 44, no. 3, pp. 1042-1070, 2025. http://dx.doi.org/10.1002/for.3238

[3] M. Segnon, R. Gupta, and B. Wilfling, "Forecasting stock market volatility with regime-switching GARCH-MIDAS: The role of geopolitical risks," International Journal of Forecasting, vol. 40, no. 1, pp. 29-43, 2024. http://dx.doi.org/10.1016/j.ijforecast.2022.11.007

[4] B. Zhang, "A study of financial time series volatility forecasting method based on GARCH modeling," in Proceedings of the 2025 International Conference on Digital Economy and Intelligent Computing, 2025, pp. 54-59. http://dx.doi.org/10.1145/3746972.3746982

[5] O. V. De la Torre-Torres, D. Aguilasocho-Montoya, and M. d. l. C. del Río-Rama, "A two-regime Markov-switching GARCH active trading algorithm for coffee, cocoa, and sugar futures," Mathematics, vol. 8, no. 6, p. 1001, 2020. http://dx.doi.org/10.3390/math8061001

[6] I. Palupi, B. A. Wahyudi, and A. P. Putra, "Implementation of hidden Markov model (HMM) to predict financial market regime," in 2021 9th international conference on information and communication technology (ICOICT), 2021, pp. 639-644: IEEE. http://dx.doi.org/10.1109/ICoICT52021.2021.9527459

[7] M. J. Fülle, H. Herwartz, and S. Wang, "Markov-Switching Multivariate GARCH Model with Copula-Distributed Innovations," Available at SSRN 4984390, 2024. http://dx.doi.org/10.2139/ssrn.4984390

[8] L. Oelschläger, T. Adam, and R. Michels, "fHMM: Hidden Markov models for financial time series in R," Journal of Statistical Software, vol. 109, pp. 1-25, 2024. http://dx.doi.org/10.18637/jss.v109.i09

[9] F. Yin, Y. You, T. Wang, and M. Yu, "Pricing VIX Futures Under a Markov-Switching GARCH Framework," Journal of Futures Markets, 2025. http://dx.doi.org/10.1002/fut.70041

[10] K. Sako, B. N. Mpinda, and P. C. Rodrigues, "Neural networks for financial time series forecasting," Entropy, vol. 24, no. 5, p. 657, 2022. http://dx.doi.org/10.3390/e24050657

[11] R. Qi and L. Dong, "Financial Time Series Forecasting Algorithm Based on Recurrent Neural Network," Advances in Artificial Intelligence, Big Data and Algorithms, pp. 339-345, 2023. http://dx.doi.org/10.3233/FAIA230828

[12] Z. Zeng, R. Kaur, S. Siddagangappa, S. Rahimi, T. Balch, and M. Veloso, "Financial time series forecasting using cnn and transformer," arXiv preprint arXiv:2304.04912, 2023. http://dx.doi.org/10.48550/arXiv.2304.04912

[13] A. Hadizadeh, M. J. Tarokh, and M. M. Ghazani, "A novel transformer-based dual attention architecture for the prediction of financial time series," Journal of King Saud University Computer and Information Sciences, vol. 37, no. 5, p. 72, 2025. http://dx.doi.org/10.1007/s44443-025-00045-y

[14] M. R. Kabir, D. Bhadra, M. Ridoy, and M. Milanova, "LSTM–transformer-based robust hybrid deep learning model for financial time series forecasting," Sci, vol. 7, no. 1, p. 7, 2025. http://dx.doi.org/10.3390/sci7010007

[15] M. Kolambe and S. Arora, "Time Series Forecasting Enhanced by Integrating GRU and N-BEATS," International Journal Of Information, vol. 17, no. 1, pp. 140-158, 2025. http://dx.doi.org/10.5815/ijieeb.2025.01.07
[16] J. Ditzen, Y. Karavias, and J. Westerlund, "Testing and estimating structural breaks in time series and panel data in Stata," The Stata Journal, vol. 25, no. 3, pp. 526-560, 2025. http://dx.doi.org/10.48550/arXiv.2110.14550
[17] J. Ditzen, Y. Karavias, and J. Westerlund, "Multiple structural breaks in interactive effects panel data models," Journal of Applied Econometrics, vol. 40, no. 1, pp. 74-88, 2025. http://dx.doi.org/10.1002/jae.3097
[18] Z. Wang et al., "Revisiting vae for unsupervised time series anomaly detection: A frequency perspective," in Proceedings of the ACM web conference 2024, 2024, pp. 3096-3105. http://dx.doi.org/10.1145/3589334.3645710
[19] Z. Xu, J. Liechty, S. Benthall, N. Skar-Gislinge, and C. McComb, "GARCH-Informed Neural Networks for Volatility Prediction in Financial Markets," in Proceedings of the 5th ACM International Conference on AI in Finance, 2024, pp. 600-607. http://dx.doi.org/10.1145/3677052.3698600
[20] H. J. Hortúa and A. Mora-Valencia, "Forecasting VIX using Bayesian deep learning," International Journal of Data Science and Analytics, vol. 20, no. 3, pp. 2039-2060, 2025. http://dx.doi.org/10.1007/s41060-024-00562-5
[21] A. Botte and D. Bao, "A machine learning approach to regime modeling," Two Sigma, 2021.https://www.twosigma.com/articles/a-machine-learning-approach-to-regime-modeling/
[22] V. García, T. Blanco, and J. S. Sánchez, "A survey on uncertainty quantification in deep learning for financial time series prediction," Instituto de Ingeniería y Tecnología, 2024. http://dx.doi.org/10.1016/j.neucom.2024.127339
[23] X. Xu, H. Peng, and Y. Chen, "Deep Switching State Space Model for Nonlinear Time Series Forecasting with Regime Switching," arXiv preprint arXiv:2106.02329, 2021. http://dx.doi.org/10.1016/j.ijforecast.2025.05.001
[24] C. Mari and E. Mari, "Deep learning based regime-switching models of energy commodity prices," Energy Systems, vol. 14, no. 4, pp. 913-934, 2023. http://dx.doi.org/10.1007/s12667-022-00515-6
[25] I. Agakishiev, W. K. Härdle, D. Becker, and X. Zuo, "Regime switching forecasting for cryptocurrencies," Digital Finance, vol. 7, no. 1, pp. 107-131, 2025. http://dx.doi.org/10.1007/s42521-024-00123-2
[26] J. Wei, S. Yang, and Z. Cui, "Unified GARCH-Recurrent Neural Networks in Financial Volatility Forecasting," 2025.https://arxiv.org/html/2504.09380v2
[27] A. Chatterjee, H. Bhowmick, and J. Sen, "Stock volatility prediction using time series and deep learning approach," in 2022 IEEE 2nd Mysore Sub Section International Conference (MysuruCon), 2022, pp. 1-6: IEEE. http://dx.doi.org/10.1109/MysuruCon55714.2022.9972559
[28] H. T. Araya, J. Aduda, and T. Berhane, "A hybrid garch and deep learning method for volatility prediction," Journal of Applied Mathematics, vol. 2024, no. 1, p. 6305525, 2024. http://dx.doi.org/10.1155/2024/6305525
[29] G. Di-Giorgi, R. Salas, R. Avaria, C. Ubal, H. Rosas, and R. Torres, "Volatility forecasting using deep recurrent neural networks as GARCH models," Computational Statistics, vol. 40, no. 6, 2025. http://dx.doi.org/10.1007/s00180-023-01349-1
[30] S. An, X. Gao, F. An, and T. Wu, "Early warning of regime switching in a complex financial system from a spillover network dynamic perspective," iScience, vol. 28, no. 3, 2025. http://dx.doi.org/10.1016/j.isci.2025.111924
[31] A. Dezhkam, M. T. Manzuri, A. Aghapour, A. Karimi, A. Rabiee, and S. M. Shalmani, "A Bayesian-based classification framework for financial time series trend prediction," The Journal of supercomputing, vol. 79, no. 4, pp. 4622-4659, 2023. http://dx.doi.org/10.1007/s11227-022-04834-4
[32] O. B. Akgun and E. Gulay, "Dynamics in Realized Volatility Forecasting: Evaluating GARCH Models and Deep Learning Algorithms Across Parameter Variations," Computational Economics, vol. 65, no. 6, pp. 3971-4013, 2025. http://dx.doi.org/10.1007/s10614-024-10694-2
[33] Z. Wu, Q. Zhang, J. Zhou, H. Chen, and Y. Liu, "WAVAE: A Weakly Augmented Variational Autoencoder for Time Series Anomaly Detection," Information Fusion, p. 103462, 2025. http://dx.doi.org/10.1016/j.inffus.2025.103462