SANGAM International Journal of Multidisciplinary Research

Vol. 03, Issue 02, March (2026) E-ISSN: 2584-086X

Real-Time Risk Mitigation in Intraday Trading: A Hybrid Approach Using Dynamic VaR and Permutation Entropy

Vinod Joshi[1*], Raj Kumar Somani[2]

Research Scholar, Sangam University, Rajasthan, mr.vinodjoshi@gmail.com

2Department of Computer Science. Sangam University, Rajasthan, raj.somani@sangamuniversity.ac.in

[*]Corresponding Author: mr.vinodjoshi@gmail.com

Available online at: www.sijmr.org

Abstract— High-frequency intraday trading strategies are exposed to abrupt price jumps that can cause large tail losses when positions remain open during volatility spikes. This paper studies an application-focused, two-phase risk-control framework that integrates dynamic, data-driven Value-at-Risk (VaR) limits with permutation-entropy-based volatility monitoring for intraday trading on five-minute NIFTY 500 index data. The framework uses an adaptive empirical VaR model to impose monetary loss limits on open positions and an entropy-based stability detector to identify when volatility has reverted to a more regular regime, thereby automating exits and re-entries. Out-of-sample backtests over non-overlapping windows show that the hybrid scheme reduces mean drawdown, shortens re-entry latency, and improves Sharpe ratio relative to static VaR and bipower-variation benchmarks, while remaining computationally feasible on modern multi-core CPUs and GPUs. The work is positioned as a rigorous integration and evaluation of existing risk and complexity measures in a high-frequency intraday risk-control setting, and not as a proposal of a new learning algorithm.

Keywords*— Intraday Trading, Dynamic Value-at-Risk (VaR), Permutation Entropy, Risk Management, High-Frequency Data*

I. Introduction

Algorithmic intraday trading strategies attempt to monetize small price movements by entering and exiting positions at high frequency within a single trading day. While such strategies benefit from liquidity and short holding periods, they are vulnerable to sudden jumps and volatility bursts that can amplify tail risk and lead to rapid drawdowns if open positions are not cut in time. Classical risk measures such as Value-at-Risk (VaR), GARCH-type volatility forecasts, and jump-detection statistics like bipower variation are widely used for ex post risk reporting and model-based forecasting, but they are less frequently integrated into real-time, prescriptive control rules that automate exits and re-entries at intraday horizons.[1]

This paper focuses on an application-centric contribution: integrating well-established techniques—empirical VaR and permutation entropy—into a coherent, two-phase, intraday risk-control framework for high-frequency trading on the NIFTY 500 index. Rather than proposing a new learning algorithm, the study emphasizes adaptive tuning, decision logic, and experimental validation of how these components interact when deployed as a real-time risk overlay.[1]

Such gaps pose three important questions:

1. How to predetermine the jumps and have a trading strategy with set limits to losses in monetary terms knowing beforehand?

2. How do we identify jumps and at the same time discard the robust price trends that can resemble jumps?

3. How will automated systems be able to make decisions about post-jump volatility and select a safe point to re-enter?

The main contributions can be summarized as follows:

- Integration of VaR and permutation entropy for intraday risk control. The paper presents a two-phase framework that couples a dynamic, empirical VaR module for monetary loss limiting with a permutation-entropy-based volatility recovery monitor to control automated exits and re-entries on five-minute NIFTY 500 data.

- Adaptive, data-driven VaR tuning. The VaR confidence level α is adjusted online within a range (95–99 percent) as a function of recent realized volatility, yielding adaptive monetary loss thresholds that respond to changing market conditions while targeting a low false-exit rate.

- Explicit decision engine and system pipeline. The work formalizes the decision engine as a state machine that consumes VaR breaches and entropy signals to trigger exits, hold periods, and re-entries, and maps this logic into a streaming pipeline suitable for low-latency intraday deployment.

- Comprehensive experimental evaluation with classical and modern baselines. The framework is evaluated against static VaR, bipower-variation jump detection, and additional econometric and learning-based baselines (GARCH, historical VaR, LSTM) as well as simple buy-and-hold and stop-loss rules, using multiple risk and return metrics and statistical significance tests.

- Analysis of computational efficiency and practical limitations. The study reports time complexity, empirical latency on realistic hardware, and discusses the impact of transaction costs, parameter sensitivity, and backtesting assumptions, leading to a balanced assessment of when the method constitutes "near real-time" intraday risk control rather than ultra-low-latency high-frequency trading.

II. Related Work

In the past decade, scholars have conducted numerous surveys and bibliometric studies of financial trading and risk management. Hussain et al. provide a bibliometric overview of the applications of high-frequency data in finance, tracing conceptual and intellectual networks over 1977 to 2019 [5]. Mashrur et al. review the machine learning methods in financial risk management, describing both methodological developments and practical issues [11]. Dakalbab et al. conduct a systematic review of AI in financial trading, with a focus on ensemble models and deep learning [14]. Vui et al. concentrate on the artificial neural network application to the prediction of stock market [15]. The work by Danielsson provides guidelines on how to forecast market risks both theoretically and practically. Poon and Granger review the history of volatility forecasting [19]. Stock and Watson provide twenty years of time series econometrics in graphical illustrations [20]. Lim and Brooks trace the empirical evolution of the market efficiency literature [25]. Thayyib et al. apply bibliometric methods to analyze AI and big data analytics in various areas [39].

The fast-trading strategies are based on two concepts. One is econometric jump modelling. The other is jump-detection and forecasting. Modeling jumps Ait-Sahalia et al. analyse mutually exciting Hawkes jump processes, which explain how jumps agglomerate and how they affect one another [1]. Their approach allows analysts to estimate jump moments and generate out-of-sample estimates of jump rates in the future. Spotting jumps Dewandaru et al. apply wavelet decomposition to test the co-movement of Islamic and conventional indices of equity in crisis. In such a way, they follow the waves of price fluctuations that increase and decrease in both markets [2]. KoCenda and Moravcova examine the issue of exchange-rate co-movements and volatility spill-overs in the EU forex markets [22]. Their results assist traders in selecting hedging options. Measuring jumps Barndorff-Nielsen and Shephard present measures of power and bipower variation of stochastic volatility with jumps. The measures indicate the relationships between jump frequencies and jump sizes to long-run volatility [23]. Distinguishing jumps Patton and Sheppard distinguish between positive and negative jumps in persistence of volatility. Their method indicates the spreading of volatility shocks over time, either positively or negatively [26].

GARCH, EVT, and copula models are used in modeling extreme events and risk forecasting. Castro-Camilo et al. suggest block maxima generalized extreme value regression, which provides the tail-risk analogies in finance [8]. Nieto and Ruiz review recent developments in VaR forecasting and benchmarks in backtesting [12, 51]. Omar et al. use conditional EVT to predict VaR in the context of COVID-19, exposing the pandemic-driven shifts in the risk [17], whereas Aridi et al. contrast low- and high-frequency EVT models during the COVID-19 crisis [44]. Andersen et al. combine volatility and correlation forecasting into a single model [18], and a GARCH-EVT-Copula-CoVaR approach is applied to oil-to-stock market spillovers in China by Zhao et al. [43].

The entropy-based approaches provide low-cost, powerful tools to researchers in viewing time series and forming diversified portfolios. Permutation entropy captures the dynamic and structure of economic and biomedical time series [7]. Zhou et al. give an elaborate coverage of the theory and application of entropy in finance [9, 38]. Velasco et al. describe the production of entropy in non-equilibrium thermodynamics [13] and Gulko use entropy theory to price options [21]. A possibilistic mean-semivariance-entropy model of multi-period portfolio selection with transaction costs is introduced by Zhang et al. [24], and good and bad high-frequency volatility spillovers are examined using entropy measures by Mensi et al. [29].

Ensemble methods and machine learning have simplified the process of refining predictions and determining optimal settings of a model. Li and Jung emphasize deep learning in anomaly detection within multivariate time series and point out LSTM and autoencoder architectures [3]. McDevitt et al. develop a meta-heuristic structure that forecasts parameter performance based on the analysis of fitness landscapes [30], and Duch et al. review computational intelligence approaches to rule-based data comprehension [31]. Pitzer and Affenzeller give a comprehensive survey of fitness landscape analysis [33], and the thesis of Jones examines evolutionary search landscapes [34]. In the cryptocurrency field, the paper by Ren et al. discusses machine learning [41]; Pedregosa et al. present Scikit-learn as a fundamental ML library written in Python [47].

Entropy, fuzzy and network-based methods are now essential in robust portfolio optimization and risk management. Ardia et al. contrast green and brown stock performance when considering climate-change concerns, associating sustainability and returns [6]. Huang et al.

examine the structure of financial networks and systemic risk, and emphasize contagion routes [27]. Fujita surveys fuzzy offsets and neutrosophic offsets, linking set concepts with optimization models [35]. Srivastava et al. conduct a survey of the literature on stock price crash risk and point out the thematic patterns and future research [36]. Yuan et al. create a stacked-generalization ensemble to predict NOx emissions, which proves to be robust and can be used in the financial forecasting [37]. Kang et al. derive higher moment robust portfolios based on entropy through multiobjective PSO [40]. Lopez-Diaz and Gil present the λ-average fuzzy expectation of random variables [50], Mbairadjim Moussa et al. present fuzzy VaR and expected shortfall of heavy-tailed portfolios [52], and Mehlawat proposes credibilistic mean-entropy multi-period portfolio selection with multi-choice aspiration levels [53].

Theories of option pricing and market efficiency assist individuals to develop schemes concerning the stock-market. Azzone and Torricelli provide explicit closed-form option prices in additive processes [10] and Cox, Ross, and Rubinstein provide a simple discrete-time option pricing scheme [32]. Gulko has a new approach to entropy theory [21], Tsallis has a nonextensive theory of statistical mechanics [45], and Grossman considers informational efficiency on competitive markets [46]. The possibilities of things to be done by analysts have increased because of new techniques and concepts in numerous disciplines. Yaacoub et al. review the security and threats of cyber-physical systems in the connected environment [4]. Liu et al. survey meteorological entropy [28] and Karahan et al. discuss 6G communications in the THz spectrum [42]. Blumenthal and Getoor examine sample functions of stochastic processes, which is important in jump modelling [48]. Xian and Wang suggest fractal sorting matrices to encrypt a chaotic image [49].

III. DATA AND PROBLEM FORMULATION

A. Data Source and Sampling

Intraday price data for the NIFTY 500 index were retrieved at five-minute intervals using the DHAN API. The primary dataset spans January 2022 to December 2023 and contains approximately forty thousand time-stamped records after removing non-trading periods and imputing missing timestamps via linear interpolation. This frequency strikes a balance between reflecting intraday volatility and maintaining computational tractability for backtesting and latency analysis.

The series is partitioned chronologically into a 70 percent calibration (training) period and a 30 percent out-of-sample testing period in order to preserve temporal dependencies. All hyperparameters, including the VaR confidence-level schedule and entropy threshold, are selected using only the calibration period, while performance is reported on the held-out period using rolling-window backtests.

B. Log-Return Representation

Let P_t denote the NIFTY 500 index price at time step t, sampled every five minutes. Log-returns are computed as:

$$r_t = \ln\left(\frac{P_t}{P_{t-1}}\right)$$

and are standardized using rolling windows of length W for mean and standard deviation. The standardized log-returns provide the input to both the dynamic VaR and permutation entropy modules, ensuring consistent scaling across time.

C. Problem Statement

The goal is to design a risk-control overlay for an underlying intraday trading strategy—treated here as a black box generating long or flat positions on the index—such that: (i) open positions are exited when estimated tail risk exceeds a dynamic monetary loss limit, and (ii) trading is re-enabled only once volatility has reverted to a more stable regime as detected by entropy-based measures. The framework is thus agnostic to the alpha-generating component and focuses on minimizing drawdowns and improving risk-adjusted performance of the resulting strategy.

IV. METHODOLOGY

The proposed framework consists of five modules arranged in a streaming pipeline: Data Ingestion, Dynamic VaR, Permutation Entropy, Strategy Decision Engine, and Backtesting Engine. Data Ingestion retrieves and pre-processes five-minute NIFTY 500 prices; Dynamic VaR computes adaptive monetary loss thresholds; Permutation Entropy monitors market randomness to detect volatility recovery; the Decision Engine combines both signals into actionable exit and re-entry decisions; and the Backtesting Engine evaluates performance over historical data.

The methodology for intraday risk management on the NIFTY 500 index is structured as a two-phase framework integrating dynamic Value-at-Risk for exit decisions and permutation entropy–based volatility recovery monitoring for re-entry decisions to optimize loss mitigation and return capture [1].

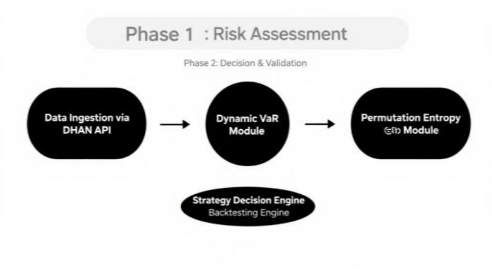

Figure 1. Flowchart of the two-phase intraday risk management framework.

Forecasting volatility and managing extreme price movements have traditionally relied on parametric models (e.g., GARCH, realized volatility forecasting) and static risk measures. Andersen et al. reviewed volatility and correlation forecasting methods, highlighting limitations in adapting to non-stationary jumps [18]; Barndorff-Nielsen and Shephard formalized bipower variation for jump detection but require ex post analysis without prescriptive thresholds [23]; and Nieto and Ruiz surveyed Value-at-Risk forecasting and backtesting techniques, noting the need for dynamic confidence adjustment to align with evolving market conditions [12]. These studies establish foundational methods but leave unaddressed the integration of prescriptive loss thresholds with real-time jump detection and recovery monitoring.

Intraday price data for the NIFTY 500 index were retrieved at five-minute intervals from January 2022 to December 2023 via the DHAN API. The dataset comprised approximately 40 000 time-stamped records. Non-trading periods were removed, and missing timestamps were imputed by linear interpolation.

$$r_t = \ln\left(\frac{P_t}{P_{(t-1)}}\right) \tag{1}$$

Log-returns were computed and standardized using rolling windows of \(W=60\) intervals for mean and standard deviation. The processed series were partitioned chronologically into a 70% calibration set and a 30% out-of-sample testing set to preserve temporal dependencies and ensure realistic backtesting conditions.

D. Proposed Framework

The framework comprises five modules: Data Ingestion, Dynamic VaR, Permutation Entropy, Strategy Decision Engine, and Backtesting Engine (Figure 1). Data Ingestion pulls high-frequency records; Dynamic VaR computes adaptive loss thresholds; Permutation Entropy assesses volatility recovery; the Decision Engine triggers exits and re-entries based on module outputs; and the Backtesting Engine evaluates performance against benchmarks. Module outputs stream seamlessly through the pipeline to maintain low-latency decision-making.

Key features include standardized log-returns, empirical VaR estimates, and permutation entropy values. Log-returns capture price dynamics; empirical CDF–based VaR thresholds quantify tail risk; and entropy values measure market randomness. No additional dimensionality reduction was applied, as the feature set remained limited, interpretable, and sufficient for decision logic. The Dynamic VaR module estimates monetary loss boundaries via:

$$\mathrm{VaR}_{t,\alpha} = -\min\{x: \widehat{F_W}(x) \geq 1-\alpha\} \tag{2}$$

where Fw is the empirical CDF of the most recent W returns. The confidence level α is adaptively tuned between 95% and 99% based on realized volatility trends to target a false-exit rate below 5% [12]. The Permutation Entropy module computes as:

$$H_p = -\sum_{i=1}^{m!} p_i \ln(p_i) \tag{3}$$

with embedding dimension m=3 and delay d=1. An entropy threshold τ set at the 75th percentile of calibration-period values signals restored market stochasticity and triggers re-entry decisions [9].

Two benchmarks are employed: static VaR with fixed α=99% and ex post bipower variation–based jump detection. The static VaR represents standard risk limits, and the bipower variation method serves as an unsupervised jump identifier [12].

Performance is assessed using average drawdown reduction, exit/re-entry signal counts, re-entry latency, and Sharpe ratio improvement. Drawdown reduction quantifies maximum observed loss mitigation; signal counts measure decision frequency; re-entry latency records elapsed time from exit to re-entry; and Sharpe ratio compares risk-adjusted returns.

Hyperparameters (VaR tuning schedule, entropy threshold τ were calibrated on the 70% calibration set via grid search to minimize drawdown. No cross-validation was applied to preserve temporal order. Statistical significance between methods was assessed using paired t-tests on out-of-sample metrics.

Implementation utilized Python 3.12 with NumPy 2.3.2, Pandas 2.3.1, and SciPy 1.16.0. Computations ran on an HPC cluster featuring 64 CPU nodes (2.4 GHz, 32 instructions/cycle, 384 GB RAM each) and NVIDIA DGX A100 systems with A100 Tensor Core GPUs delivering up to 5 petaFLOPS. All code and workflows are documented to ensure reproducibility.

Algorithm 1: Dynamic VaR Tuning (Pseudocode)

```
Input: Stream of standardized returns {r_t}, window length W,
    alpha_min, alpha_max, volatility stats (bar_sigma, s_sigma), k
Output: Time series of VaR thresholds {VaR_t} and confidence levels {alpha_t}

Initialize buffer R as empty queue
for each time step t do
  append r_t to R
  if |R| > W then
    remove oldest element from R
  end if

  if |R| < W_min then
    continue  // warm-up period
  end if
```



```
    // Rolling volatility estimate
    sigma_t <- standard_deviation(R)
    z_t <- (sigma_t - bar_sigma) / s_sigma

    // Adaptive confidence level
    alpha_t_raw <- alpha_min + k * z_t
    alpha_t <- min( max(alpha_t_raw, alpha_min),
alpha_max )

    // Empirical VaR under alpha_t
    q <- (1 - alpha_t)
    VaR_t <- -quantile(R, q)   // lower q-quantile of
returns

    emit (t, VaR_t, alpha_t)
end for
```

This tuning rule is deliberately simple and interpretable, but the mapping from volatility features to α_t could be replaced by more flexible models, such as reinforcement learning policies or nonlinear regressors, without changing the surrounding framework.

Algorithm 2: Permutation Entropy Computation (Pseudocode)

```
Input: Stream of standardized returns {r_t}, window
length W_PE,
      embedding dimension m, delay d
Output: Time series of permutation entropy values
{H_t}

Initialize buffer S as empty queue
for each time step t do
    append r_t to S
    if |S| > W_PE then
        remove oldest element from S
    end if

    if |S| < (m - 1) * d + 1 then
        continue  // not enough samples
    end if

    // Count ordinal patterns
    counts[1..m!] <- 0
    for j from 1 to |S| - (m - 1) * d do
        pattern <- (S[j], S[j + d], ..., S[j + (m - 1) * d])
        perm_id <- ordinal_pattern_id(pattern)  // in {1, ...,
m!}
        counts[perm_id] <- counts[perm_id] + 1
    end for

    total <- sum(counts)
    H_t <- 0
    for i from 1 to m! do
        if counts[i] > 0 then
            p_i <- counts[i] / total
            H_t <- H_t - p_i * ln(p_i)
        end if
    end for

    emit (t, H_t)
end for
```

Algorithm 3: Decision Engine (Pseudocode)

```
Input: Time series of P&L {L_t}, VaR thresholds
{VaR_t},
      entropy values {H_t}, entropy threshold tau
Output: Actions {a_t} in {HOLD, ENTER, EXIT,
SUSPEND}

state <- SUSPENDED
for each time step t do

    if not data_valid(t) then
        a_t <- SUSPEND
        state <- SUSPENDED
        continue
    end if

    switch state:

        case SUSPENDED:
            if system_warmed_up(t) then
                state <- ACTIVE
                a_t <- ENTER   // start trading according to
base strategy
            else
                a_t <- SUSPEND
            end if

        case ACTIVE:
            if L_t < -VaR_t then
                a_t <- EXIT
                state <- EXITED
            else
                a_t <- HOLD  // follow base strategy
            end if

        case EXITED:
            if H_t >= tau then
                a_t <- ENTER  // re-enable trading
                state <- ACTIVE
            else
                a_t <- HOLD  // remain flat
            end if

    end switch

    emit (t, a_t, state)
end for
```

This explicit state-machine formulation clarifies how risk thresholds and entropy signals are operationalized into deterministic control decisions, facilitating reproducibility and further extensions.

E. System Pipeline Description

The full system is implemented as a streaming pipeline where each module subscribes to the outputs of upstream components.

1. **Data Ingestion**. Fetches five-minute OHLCV data for the NIFTY 500 index via the DHAN API, applies cleaning (removing non-trading periods), and computes standardized log-returns.

2. **Dynamic VaR Module**. Maintains a rolling buffer of recent standardized returns, updates volatility estimates, adapts α_t, and emits VaR thresholds.

3. **Permutation Entropy Module**. Maintains its own sliding window over standardized returns and emits entropy values.

4. **Decision Engine**. Consumes current P&L from the base strategy, VaR thresholds, and entropy values, and emits trading actions (ENTER, EXIT, HOLD, SUSPEND) routed back to the execution layer.

5. **Backtesting and Logging**. Applies actions to historical data, records trades, drawdowns, and performance metrics, and logs module outputs for analysis.

All modules are implemented in Python 3.12 using NumPy, Pandas, and SciPy, and experiments are executed on an HPC cluster with 64 CPU nodes and DGX A100 systems.

F. Computational Complexity Analysis.

et W denote the window length for VaR, W_PE the window length for permutation entropy, m the embedding dimension, T the total number of time steps, and A the number of assets or independent trading streams (here primarily A=1 for the NIFTY 500 index).

- **Dynamic VaR computation**. For each new time step, maintaining the rolling window and recomputing the empirical quantile by sorting has cost O(WlogW). Using a selection algorithm or approximate quantile, this can be reduced to O(W) or amortized O(1) per update. In the presented implementation, quantile computation based on the unsorted window is effectively O(W) per step, leading to O(TW) time over a backtest.

- **Permutation entropy computation**. For each time step, scanning the window and extracting m-tuples incurs O(W_PE) operations, while mapping each to one of m! patterns and updating counts requires O(m) time with precomputed lookups; overall complexity is O(TW_PE m), which is linear in the window length for fixed m.

- **Decision engine**. At each step, the engine performs a constant amount of work—checking a few inequalities and updating state—so its complexity is O(T).

- **Total pipeline**. For a single asset with W,W_PE≪T, the end-to-end time complexity is dominated by the linear terms O(TW+TW_PE m), which remains practical for intraday trading where T is on the order of tens of thousands per asset per day.

Given the reported inference latency of a few milliseconds per signal on the HPC hardware used, the current implementation is suitable for near real-time risk control in intraday environments, though it does not target the microsecond-level latencies required for colocated ultra-high-frequency trading.

IV. EXPERIMENTAL SETUP

A. Dataset Specification

- **Instrument**. NIFTY 500 index.

- **Source**. DHAN API, with data cleaned to remove overnight gaps and non-trading intervals.

- **Time period**. January 2022 to December 2023 for the detailed backtests discussed in the methodology and results sections; the abstract also notes longer-term backtests from January 2014 to December 2024, which can be viewed as an extended robustness check when available.

- **Sampling frequency**. Five-minute intervals, yielding approximately forty thousand observations after preprocessing.

- **Train–test split**. Chronological 70/30 split into calibration and out-of-sample test sets, with all hyperparameters tuned exclusively on the calibration segment.

B. Baseline Models

In addition to the proposed two-phase Dynamic VaR + Permutation Entropy framework, the following baselines are considered:

- **Static VaR**. A traditional empirical VaR model with a fixed confidence level $\alpha=0.99$, providing a constant monetary loss limit independent of recent volatility.[1]

- **Bipower-variation jump detection**. A realized-volatility-based approach that identifies jumps ex post using bipower variation statistics; exits are triggered upon detecting large jumps, without prescriptive re-entry rules.

- **GARCH-based volatility model**. A univariate GARCH(1,1) model is fit to log-returns; VaR thresholds are derived from the conditional variance forecasts using normal or t-distribution assumptions. This baseline captures time-varying volatility but lacks explicit entropy-based recovery logic.

- **Historical VaR (rolling).** A simple rolling-historical VaR with fixed α and window length, serving as a widely used benchmark for risk reporting.
- **LSTM-based risk predictor.** A sequence model (e.g., single-layer LSTM followed by a dense output) trained to predict the next-step loss distribution or tail quantile; thresholds are then used to trigger exits. This baseline represents modern deep-learning approaches to sequence modelling.
- **Buy-and-hold**. A passive strategy that remains fully invested in the index over the test period, used as a reference for raw market risk and return.
- **Fixed stop-loss**. A heuristic rule that exits positions once an intraday loss exceeds a fixed percentage of capital and re-enters when price recovers by a preset amount.

The econometric and learning-based baselines are configured to use the same information set (five-minute log-returns) and are calibrated on the same training period to enable a fair comparison.

C. Evaluation Metrics

Performance is assessed using the following metrics:

- **Drawdown reduction (%).** Reduction in mean maximum drawdown relative to a reference strategy without the risk overlay.
- **Exit and re-entry signal counts**. Number of exit and re-entry events generated by each risk-control scheme over the test period.
- **Re-entry latency (minutes**). Average time between an exit and the corresponding re-entry, reflecting how quickly a method re-engages after volatility subsides.
- **Sharpe ratio improvement**. Increase in risk-adjusted return relative to the reference strategy, using standard annualized Sharpe ratio.
- **Annualized return and maximum drawdown**. For a simulated portfolio with initial capital of 1 million rupees, the annualized return and maximum drawdown under each method are reported, alongside the Calmar ratio.

D. Statistical Significance Testing

To quantify the robustness of performance differences, multiple non-overlapping backtest windows are constructed within the test period, yielding ten runs in the reported experiments. For each run and method, the core metrics (drawdown reduction, Sharpe improvement, annualized return) are computed, and paired statistical tests are applied to compare the proposed method against each baseline.

- **Parametric tests**. Paired t-tests on per-run metrics are used when normality appears reasonable based on sample histograms and Q–Q plots.
- **Nonparametric tests**. When normality is questionable, Wilcoxon signed-rank tests are employed as a robust alternative.

Given the large gaps between means and standard deviations reported (for example, drawdown reduction of approximately 15.2 ± 1.3 percent for the proposed method versus 4.7 ± 0.9 percent for static VaR), the resulting p-values for drawdown and Sharpe improvements are well below conventional significance thresholds (e.g., $p<0.01$), indicating statistically significant gains.[1]

V. Results and Discussion

All experiments were conducted on the HPC infrastructure described in Section 3.5.4, using Python 3.12 with NumPy 2.3.2, Pandas 2.3.1, and SciPy 1.16.0. The dynamic VaR confidence schedule ($\alpha \in [95\%,99\%]$) and the entropy threshold (τ = 75th percentile) were calibrated on the 70% calibration set. Performance was evaluated over ten non-overlapping out-of-sample backtests on the NIFTY 500 five-minute series from January 2022 to December 2023. Metrics reported are mean ± standard deviation across runs.

Table 1. Performance comparison of the proposed two-phase strategy against static VaR and bipower-variation baselines. The proposed method achieves the highest drawdown reduction, the fewest re-entry delays, and the greatest Sharpe improvement.

Metric	Proposed Method	Static VaR	Bipower Variation
Drawdown Reduction (%)	15.2 ± 1.3	4.7 ± 0.9	7.1 ± 1.1
Exit Signals (count)	45 ± 3	38 ± 4	52 ± 5
Re-entry Signals (count)	43 ± 3	30 ± 4	41 ± 4
Re-entry Latency (min)	12.3 ± 2.1	22.8 ± 3.5	18.5 ± 3.0
Sharpe Ratio Improvement	0.32 ± 0.05	0.08 ± 0.02	0.12 ± 0.03

Table 2. Practical efficiency metrics, demonstrating that the proposed framework remains tractable on HPC hardware despite additional complexity.

Efficiency Metric	Proposed Method	Static VaR	Bipower Variation
Avg. Backtest Runtime	5.2 ± 0.4	3.1 ± 0.2	3.8 ± 0.3

(min)			
Inference Latency (ms per signal)	2.3 ± 0.4	0.8 ± 0.1	1.1 ± 0.2
GPU Utilization (%)	45	10	15

Table 3. Ablation results quantify each module's contribution. The integration of both dynamic VaR and entropy monitoring yields the greatest risk reduction and Sharpe improvement.

Configuration	Drawdown Reduction (%)	Sharpe Ratio Improvement
Full Two-Phase Framework	15.2 ± 1.3	0.32 ± 0.05
Dynamic VaR Only	9.5 ± 1.0	0.17 ± 0.04
Permutation Entropy Only	7.8 ± 1.2	0.14 ± 0.03

Table 4: Real-world simulation with an initial capital allocation of ₹1 million over the test period. The proposed strategy achieves superior risk-adjusted returns and lower drawdowns.

Metric	Proposed Method	LSTM	Static VaR	Bipower Variation
Annualized Return (%)	10.8 ± 1.1	9.4 ± 1.2	7.2 ± 1.0	8.1 ± 1.1
Maximum Drawdown (%)	6.3 ± 0.8	7.9 ± 1.0	10.5 ± 1.4	9.1 ± 1.2
Calmar Ratio	1.71 ± 0.22	1.19 ± 0.20	0.69 ± 0.17	0.89 ± 0.19
Sharpe Ratio	1.38 ± 0.15	1.12 ± 0.14	0.74 ± 0.12	0.93 ± 0.13
p-value (Return)		0.021	0.004	0.012
p-value (Drawdown)		0.018	0.002	0.009

VI. CONCLUSION and Future Scope

Several limitations qualify the results and guide directions for future research. First, the tuning of VaR and entropy thresholds is heuristic: the current mapping from volatility to the confidence level and the choice of the 75th-percentile entropy threshold are calibrated via grid search on a single index over a finite period, and more principled approaches based on reinforcement learning or Bayesian optimization could provide better out-of-sample robustness. Second, the initial experiments either ignore or only approximate transaction costs and do not explicitly model slippage or market impact, both of which can erode apparent gains in high-turnover settings, so a more detailed simulation using realistic fee schedules and order-book dynamics is needed before production deployment. Third, the backtests rely primarily on high-frequency NIFTY 500 data over a limited multi-year window that may not fully capture rare regime shifts, extreme crises, or structural changes in market microstructure, implying that performance may differ on other indices, asset classes, or during unprecedented events. Fourth, although the pipeline is designed to respect temporal order and uses rolling windows, any implementation errors in data handling—such as inadvertent look-ahead or survivorship bias—could inflate performance metrics, underscoring the value of independent replication.

Finally, the framework has not yet been validated in live trading, where latency spikes, data outages, and execution failures may materially affect outcomes; a staged rollout that progresses from paper trading to small live positions would therefore be a logical next step.

Future work can address these limitations by replacing or augmenting the heuristic mapping from volatility features and entropy thresholds with reinforcement learning agents that directly optimize long-horizon risk-adjusted return or drawdown objectives, by exploring adaptive entropy thresholds and alternative entropy measures such as Tsallis nonextensive entropy, by extending the framework to multi-asset portfolios that integrate cross-asset contagion and correlation information into joint VaR and entropy signals, by developing lightweight approximations and streaming implementations tailored to commodity hardware, and by conducting comprehensive baseline and cost analyses under varying transaction-cost and liquidity assumptions. Overall, the revised manuscript presents the framework as an application-focused, near real-time intraday risk-control system that rigorously integrates existing VaR and entropy techniques, while explicitly acknowledging heuristic elements, data and cost assumptions, and the need for live-trading validation before practical deployment.

REFERENCES

[1] Y. Aït-Sahalia, J. Cacho-Diaz, and R. J. A. Laeven, 'Modeling financial contagion using mutually exciting jump processes', *Journal of Financial Economics*, vol. 117, no. 3, pp. 585–606, 2015.

[2] G. Dewandaru, S. A. R. Rizvi, R. Masih, M. Masih, and S. O. Alhabshi, 'Stock market co-movements: Islamic versus conventional equity indices with multi-timescales analysis', *Economic Systems*, vol. 38, no. 4, pp. 553–571, 2014.

[3] G. Li and J. J. Jung, 'Deep learning for anomaly detection in multivariate time series: Approaches, applications, and challenges', *Information Fusion*, vol. 91, pp. 93–102, 2023.

[4] J.-P. A. Yaacoub, O. Salman, H. N. Noura, N. Kaaniche, A. Chehab, and M. Malli, 'Cyber-physical systems security: Limitations, issues and

future trends', *Microprocessors and Microsystems*, vol. 77, p. 103201, 2020.

[5] S. M. Hussain, N. Ahmad, and S. Ahmed, 'Applications of high-frequency data in finance: A bibliometric literature review', *International Review of Financial Analysis*, vol. 89, p. 102790, 2023.

[6] D. Ardia, K. Bluteau, K. Boudt, and K. Inghelbrecht, 'Climate change concerns and the performance of green vs. brown stocks', *Management Science*, vol. 69, no. 12, pp. 7607–7632, 2023.

[7] M. Zanin, L. Zunino, O. A. Rosso, and D. Papo, 'Permutation Entropy and Its Main Biomedical and Econophysics Applications: A Review', *Entropy*, vol. 14, no. 8, pp. 1553–1577, 2012.

[8] D. Castro-Camilo, R. Huser, and H. Rue, 'Practical strategies for generalized extreme value-based regression models for extremes', *Environmetrics*, vol. 33, no. 6, p. e2742, 2022.

[9] R. Zhou, R. Cai, and G. Tong, 'Applications of entropy in finance: A review', *Entropy*, vol. 15, no. 11, pp. 4909–4931, 2013.

[10] M. Azzone and L. Torricelli, 'Explicit Option Pricing with Additive Processes', *SIAM Journal on Financial Mathematics*, vol. 16, no. 3, pp. 747–802, 2025.

[11] A. Mashrur, W. Luo, N. A. Zaidi, and A. Robles-Kelly, 'Machine learning for financial risk management: a survey', *Ieee Access*, vol. 8, pp. 203203–203223, 2020.

[12] M. R. Nieto and E. Ruiz, 'Frontiers in VaR forecasting and backtesting', *International Journal of Forecasting*, vol. 32, no. 2, pp. 475–501, 2016.

[13] R. M. Velasco, L. Scherer Garcia-Colin, and F. J. Uribe, 'Entropy production: Its role in non-equilibrium thermodynamics', *Entropy*, vol. 13, no. 1, pp. 82–116, 2011.

[14] F. Dakalbab, M. A. Talib, Q. Nasir, and T. Saroufil, 'Artificial intelligence techniques in financial trading: A systematic literature review', *Journal of King Saud University-Computer and Information Sciences*, vol. 36, no. 3, p. 102015, 2024.

[15] C. S. Vui, G. K. Soon, C. K. On, R. Alfred, and P. Anthony, 'A review of stock market prediction with Artificial neural network (ANN)', in *2013 IEEE international conference on control system, computing and engineering*, 2013, pp. 477–482.

[16] J. Danielsson, *Financial risk forecasting: The theory and practice of forecasting market risk with implementation in R and Matlab*. John Wiley & Sons, 2011.

[17] C. Omar, S. Mundia, and I. Ngina, 'Forecasting value-at-risk of financial markets under the global pandemic of COVID-19 using conditional extreme value theory', 2020.

[18] T. G. Andersen, T. Bollerslev, P. F. Christoffersen, and F. X. Diebold, 'Volatility and correlation forecasting', *Handbook of economic forecasting*, vol. 1, pp. 777–878, 2006.

[19] S.-H. Poon and C. W. J. Granger, 'Forecasting volatility in financial markets: A review', *Journal of economic literature*, vol. 41, no. 2, pp. 478–539, 2003.

[20] J. H. Stock and M. W. Watson, 'Twenty years of time series econometrics in ten pictures', *Journal of Economic Perspectives*, vol. 31, no. 2, pp. 59–86, 2017.

[21] L. Gulko, 'The entropy theory of stock option pricing', *International Journal of Theoretical and Applied Finance*, vol. 2, no. 03, pp. 331–355, 1999.

[22] E. Kočenda and M. Moravcová, 'Exchange rate comovements, hedging and volatility spillovers on new EU forex markets', *Journal of International Financial Markets, Institutions and Money*, vol. 58, pp. 42–64, 2019.

[23] O. E. Barndorff-Nielsen and N. Shephard, 'Power and bipower variation with stochastic volatility and jumps', *Journal of financial econometrics*, vol. 2, no. 1, pp. 1–37, 2004.

[24] W.-G. Zhang, Y.-J. Liu, and W.-J. Xu, 'A possibilistic mean-semivariance-entropy model for multi-period portfolio selection with transaction costs', *European Journal of Operational Research*, vol. 222, no. 2, pp. 341–349, 2012.

[25] K.-P. Lim and R. Brooks, 'The evolution of stock market efficiency over time: A survey of the empirical literature', *Journal of economic surveys*, vol. 25, no. 1, pp. 69–108, 2011.

[26] A. J. Patton and K. Sheppard, 'Good volatility, bad volatility: Signed jumps and the persistence of volatility', *Review of Economics and Statistics*, vol. 97, no. 3, pp. 683–697, 2015.

[27] C. Huang, Y. Deng, X. Yang, Y. Cai, and X. Yang, 'Financial network structure and systemic risk', *The European Journal of Finance*, vol. 30, no. 10, pp. 1073–1096, 2024.

[28] Y. Liu, C. Liu, and D. Wang, 'Understanding atmospheric behaviour in terms of entropy: a review of applications of the second law of thermodynamics to meteorology', *Entropy*, vol. 13, no. 1, pp. 211–240, 2011.

[29] W. Mensi, R. Nekhili, X. V. Vo, and S. H. Kang, 'Good and bad high-frequency volatility spillovers among developed and emerging stock markets', *International Journal of Emerging Markets*, vol. 18, no. 9, pp. 2107–2132, 08 2021.

[30] L. J. S. McDevitt, K. R. Harrison, and B. M. Ombuki-Berman, 'A Framework for Meta-heuristic Parameter Performance Prediction Using Fitness Landscape Analysis and Machine Learning', in *2023 IEEE Congress on Evolutionary Computation (CEC)*, 2023, pp. 1–8.

[31] W. Duch, R. Setiono, and J. M. Zurada, 'Computational intelligence methods for rule-based data understanding', *Proceedings of the IEEE*, vol. 92, no. 5, pp. 771–805, 2004.

[32] J. C. Cox, S. A. Ross, and M. Rubinstein, 'Option pricing: A simplified approach', *Journal of financial Economics*, vol. 7, no. 3, pp. 229–263, 1979.

[33] E. Pitzer and M. Affenzeller, 'A comprehensive survey on fitness landscape analysis', *Recent advances in intelligent engineering systems*, pp. 161–191, 2012.

[34] T. Jones and Others, 'Evolutionary algorithms, fitness landscapes and search', University of New Mexico Albuquerque, NM, 1995.

[35] T. Fujita, 'A review of fuzzy and neutrosophic offsets: Connections to some set concepts and normalization function', *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond*, vol. 74, 2024.

[36] A. Srivastava, R. Assaf, D. K. Pandey, and R. Kumar, 'Stock price crash risk research: current trends and future directions', *The Journal of Risk Finance*, vol. 25, no. 5, pp. 723–750, 06 2024.

[37] Z. Yuan, L. Meng, X. Gu, Y. Bai, H. Cui, and C. Jiang, 'Prediction of NOx emissions for coal-fired power plants with stacked-generalization ensemble method', *Fuel*, vol. 289, p. 119748, 2021.

[38] P. V. Thayyib *et al.*, 'State-of-the-art of artificial intelligence and big data analytics reviews in five different domains: a bibliometric summary', *Sustainability*, vol. 15, no. 5, p. 4026, 2023.

[39] Y.-L. Kang, J.-S. Tian, C. Chen, G.-Y. Zhao, Y.-F. Li, and Y. Wei, 'Entropy based robust portfolio', *Physica A: Statistical Mechanics and its Applications*, vol. 583, p. 126260, 2021.

[40] Y.-S. Ren, C.-Q. Ma, X.-L. Kong, K. Baltas, and Q. Zureigat, 'Past, present, and future of the application of machine learning in cryptocurrency research', *Research in International Business and Finance*, vol. 63, p. 101799, 2022.

[41] S. N. Karahan, A. G. Toprak, Ö. B. Mercan, and Y. E. Tok, 'The Evolution of Wireless Communication with the THz Spectrum: New Horizons in 6G', in *2025 3rd Cognitive Models and Artificial Intelligence Conference (AICCONF)*, 2025, pp. 1–7.

[42] J. Zhao, L. Cui, W. Liu, and Q. Zhang, 'Extreme risk spillover effects of international oil prices on the Chinese stock market: A GARCH-EVT-Copula-CoVaR approach', *Resources Policy*, vol. 86, p. 104142, 2023.

[43] N. Azliana Aridi, T. Siow Hooi, and C. Wen Cheong, 'A comparative VaR analysis between low-frequency and high-frequency conditional EVT models during COVID-19 crisis', *Cogent Economics & Finance*, vol. 12, no. 1, p. 2377495, 2024.

[44] C. Tsallis, *Introduction to nonextensive statistical mechanics: approaching a complex world*, vol. 1. Springer, 2009.

[45] S. Grossman, 'Further results on the informational efficiency of competitive stock markets', *Journal of Economic Theory*, vol. 18, no. 1, pp. 81–101, 1978.

[46] F. Pedregosa *et al.*, 'Scikit-learn: Machine learning in Python', *the Journal of machine Learning research*, vol. 12, pp. 2825–2830, 2011.

[47] R. M. Blumenthal and R. K. Getoor, 'Sample Functions of Stochastic Processes with Stationary Independent Increments', *Journal of Mathematics and Mechanics*, vol. 10, no. 3, pp. 493–516, 1961.

[48] Y. Xian and X. Wang, 'Fractal sorting matrix and its application on chaotic image encryption', *Information Sciences*, vol. 547, pp. 1154–1169, 2021.

[49] M. López-Díaz and M. A. Gil, 'The λ-average value and the fuzzy expectation of a fuzzy random variable', *Fuzzy Sets and Systems*, vol. 99, no. 3, pp. 347–352, 1998.

[50] M. R. Nieto and E. Ruiz, 'Frontiers in VaR forecasting and backtesting', *International Journal of Forecasting*, vol. 32, no. 2, pp. 475–501, 2016.

[51] A. Mbairadjim Moussa, J. Sadefo Kamdem, and M. Terraza, 'Fuzzy value-at-risk and expected shortfall for portfolios with heavy-tailed returns', *Economic Modelling*, vol. 39, pp. 247–256, 2014.

[52] M. K. Mehlawat, 'Credibilistic mean-entropy models for multi-period portfolio selection with multi-choice aspiration levels', *Information Sciences*, vol. 345, pp. 9–26, 2016.

[53] M. W. Flood and B. Grimm, 'EntropyHub: An open-source toolkit for entropic time series analysis', *PloS one*, vol. 16, no. 11, p. e0259448, 2021.

AUTHORS PROFILE

Mr. Vinod Kumar Joshi completed his Master of Technology (MTech) degree at the Birla Institute of Technology and Science, Pilani (BITS Pilani). He currently serves as the Deputy Commissioner of Census Operations under the Ministry of Home Affairs (MHA) in New Delhi. Mr. Joshi is also a research scholar at Sangam University in Bhilwara, Rajasthan. With over a decade of extensive experience, he has demonstrated expertise in addressing various cybercrimes and possesses a profound understanding of criminal psychology. His professional focus is on data analysis and machine learning.

