mathematics

MDPI

Article

A Multi-Source Data Fusion and PSO-Transformer Decision Support System for Gold Price Volatility Forecasting and Risk Warning

Xueping Diao [1,2,3,*], Xueqiang Lv [1,2], Weizheng Wang [3,4] and Junyi Wu [1]

[1] School of Computer Science, Qinghai Normal University, Xining 810008, China; lxq@bistu.edu.cn (X.L.); 202333341014@stu.qhnu.edu.cn (J.W.)
[2] Beijing Key Laboratory of Internet Culture and Digital Dissemination Research, Beijing Information Science and Technology University, Beijing 100192, China
[3] Bank of Qinghai Co., Ltd., Xining 810000, China; a1963436@adelaide.edu.au
[4] School of Computer and Mathematical Sciences, The University of Adelaide, Adelaide, SA 5005, Australia
* Correspondence: 202333341003@stu.qhnu.edu.cn; Tel.: +86-187-9719-1293

Abstract

Gold volatility responds to shocks and sentiment, yet the two are rarely fused, and forecasts seldom become actionable. We link a multi-source fusion subsystem to a risk-warning layer through a Transformer whose three pre-registered hyperparameters are searched by particle swarm optimization (PSO-Transformer). It fuses video sentiment, headline scoring and market factors screened to 20 of 21 candidates; the decision side maps the predicted realized variance RV_t/M_t (bp^2, non-annualized) into three traffic-light tiers of a calibrated predictive distribution. On 5-min spot gold data (January–June 2025, 126 trading days), the PSO-Transformer attains a one-step MAE of 8.038, a 36.5% average improvement over seven baselines under protocol P1, where selection and evaluation share one window; under the leak-free protocol P2, the same pipeline attains 10.16 against 8.82 for the GRU, so the first place reflects the P1 design, not an independent out-of-sample comparison. Diebold–Mariano gains over GARCH, SVM and XGBoost survive a Holm correction. Walk-forward backtesting over 59 origins with persistence-anchored alerts yields 67.8% tier accuracy, 78.8% high-risk precision and 92.9% recall, with the neural engine fed into the same rule attaining 45.5% against 61.4%. One instrument, six months: a documented architecture, not a demonstrated transfer.

Keywords: multi-source data fusion; PSO-Transformer; gold volatility forecasting; multimodal sentiment analysis; risk warning system

MSC: 68T07; 91G20; 90C59

Check for updates

Academic Editor: Codruta Mare

Received: 14 September 2026
Revised: 22 October 2026
Accepted: 2 November 2026
Published: 9 November 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

1. Introduction

As the most representative safe-haven asset in global financial markets [1], gold has evolved from a physical store of value into a highly financialized risk-hedging instrument. The spot gold market (XAU/USD), with global electronic trading, second-level price updates and high leverage, is acutely sensitive to macroeconomic shocks and to shifts in the tail risk of the wider commodity market [2], so accurate volatility forecasting matters for asset allocation, hedging efficiency and systemic risk identification [1].

Prior work in this direction has advanced in two steps, and the third is the one this paper takes. The first extracted sentiment from text, and financial sentiment analysis is by

now a mature field with its own surveys [3]. The second turned to the audio and video channel: Zadeh et al. [4] introduced the tensor fusion network (TFN), which represents the linguistic, visual and acoustic modalities as a single outer-product tensor and models unimodal, bimodal and trimodal interactions explicitly, and that representation is the core of the extraction module of Section 3. Three studies are closest to the setting of this paper: Alzazah et al. [5] find video-news sentiment to carry a stronger link to market fluctuations than conventional text news, Zhang and Zhang [6] build a monthly video sentiment index from investors' facial expressions that predicts stock returns out of sample, and Liu and Huang [7] fuse news and social-media signals inside a multimodal deep learning framework. In all three, the video signal enters as an additional regressor of an otherwise conventional model, it is not aligned in time with a high-frequency market panel, and the resulting forecast is not connected to any decision rule, which is the gap this paper addresses (Section 2.5 and Table 1). That third step consists of the two things the three studies leave undone: aligning a video sentiment channel in time with a high-frequency market panel and connecting the resulting forecast to a decision rule. Section 2 reviews the wider methodological literature—volatility models, text sentiment, Transformer architectures and metaheuristic tuning—and Table 1 contrasts the closest studies with the present framework along five dimensions. *Extended discussion of this section, with the further detail and the intermediate quantities it carries, is given in Supplementary Section S5.1.*

1. **Temporal synchronization of news, video and market data.** The obstacle the forecasting literature raises most often is that news items, macroeconomic releases and high-frequency price series are stamped on different clocks, so that joining them by calendar date can let information published after the target window enter the input. Here, every daily factor is stamped at or before 20:30 Beijing time of its stamping day—the video channel by the publication time of the individual video, the news and macroeconomic channels by the last 5-min bar at or before that clock time, the price and technical channels by the last available bar of the day—while the modeled target is the realized variance of the following trading day, so that no factor can carry information from the day it is used to predict (Section 3.1; Supplementary Table S10 states the rule predictor by predictor).
2. **A time-aligned multimodal sentiment index for gold volatility.** TFN video embeddings and Qwen-2.5 headline scores are combined with the price, technical, commodity and macroeconomic blocks in one daily factor matrix under that alignment rule. The index moves with the information flow, because it is refreshed by every item that clears the materiality filter, but it is not weighted: items enter with equal standing under the fixed, pre-registered aggregation rule of Section 3.1, no recency decay and no per-item time-varying weight is applied at any stage, and the only weights estimated in the fusion stage are the Random Forest importances of Section 4.2, computed once on the training prefix, which decide which columns to retain rather than how to weight them (Supplementary Section S5.8).
3. **A two-stage architecture that separates fusion from learning.** The first stage assembles the candidate matrix and applies Random Forest selection: the released auxiliary file has 26 columns, five of which are near-constant placeholders that cannot serve as independent predictors, which leaves the 21 candidates—the 20 exogenous predictors of Table 2 together with the one-day-lagged realized volatility RV_{t-1}—of which 20 are retained and listed in Table 3 (Sections 4.2 and 6.1). The second stage trains a Transformer on the retained block over a fixed 22-day sliding window and searches its three pre-registered hyperparameters (Supplementary Table S7) by particle swarm optimization instead of tuning them by hand (Sections 5.1–5.3). The engine attains the best value on all six error metrics at the one-step and two-step horizons and on five of

the six at the five-step horizon (Sections 6.3–6.6), and the gains over the econometric and machine-learning baselines are confirmed by Diebold–Mariano tests (Section 6.7).

4. **A risk warning decision layer.** The forecast is mapped to graded alerts by thresholding a calibrated conditional predictive distribution rather than the point forecast, which is what allows the extreme tiers to be populated at all and closes the distance between a numerical prediction and an actionable signal (Sections 7.2–7.4).

How to read this paper: the main text keeps the complete route of the study—problem, data, factor system, method, experiments, risk-warning rule and conclusions—with every number, table and figure of the paper; Supplementary Section S5 carries the extended derivations, qualifications and further detail for each section, and each subsection of Section S5 is announced by an italic pointer in the main text.

2. Related Work

This section reviews five strands of literature that inform the present study: volatility forecasting methods; sentiment analysis in financial markets; Transformer architectures for time series prediction; PSO-based hyperparameter optimization; and systems approaches in financial modeling.

2.1. Volatility Forecasting Methods

Volatility modeling has a long tradition in financial econometrics [8]. The generalized autoregressive conditional heteroskedasticity (GARCH) model [9] and its extensions, notably the EGARCH [10] and GJR-GARCH [11] specifications, capture volatility clustering and asymmetric leverage effects through parametric conditional variance equations. The heterogeneous autoregressive-realized volatility (HAR-RV) model [12] decomposes realized volatility into daily, weekly and monthly components and achieves a strong in-sample fit through long-memory modeling of high-frequency data, while the drivers of gold futures volatility have been examined directly in a structural vector autoregression [13], and geopolitical risk indices have been used to forecast gold volatility [14].
Extended discussion of this subsection, with the further detail and the intermediate quantities it carries, is given in Supplementary Section S5.2.

2.2. Sentiment Analysis in Financial Markets

The role of investor sentiment in asset pricing has gained increasing empirical support, with comprehensive surveys documenting the evolution of sentiment analysis methods across domains including finance [3]. Chuang et al. [15] embed measures of financial uncertainty in a GARCH-MIDAS model with variable selection and find that they carry information for gold market volatility, and Luo et al. [16] show that sentiment indicators improve oil and gold volatility forecasts once structural breaks are accounted for.
Extended discussion of this subsection, with the further detail and the intermediate quantities it carries, is given in Supplementary Section S5.3.

2.3. Transformer Models for Financial Time Series Prediction

The Transformer architecture [17], originally designed for natural language processing, has been increasingly adopted for financial time series forecasting, where self-attention captures long-range dependencies that recurrent models miss, and has been extended to probabilistic forecasting with exogenous variables [18]. Deep learning ensembles fusing convolutional and long short-term memory components have also improved on individual models [19], Lim et al. [18] introduced the Temporal Fusion Transformer for multi-horizon forecasting, and Zhou et al. [20] proposed the Informer with ProbSparse attention for efficient long-horizon prediction, while Wan et al. [21] demonstrated multi-head attention

for cross-variable dependencies in financial series, and meta-learning has been applied to the forecasting of non-stationary financial time series [22].

The Transformer's predictive performance is nevertheless highly sensitive to hyperparameter configuration—attention heads, hidden dimensions, learning rate and batch size [18]—and manual tuning is laborious, subjective and prone to suboptimal convergence [23]. Our study embeds a particle swarm optimization (PSO) algorithm within the Transformer architecture, and two qualifications align that claim with what the engine actually searches. First, the search space is three-dimensional and fixed before any model is fitted—the model dimension d_{model}, the number of training epochs and the batch size—so it covers two of the four quantities listed above and adds the epoch count, while the number of attention heads, the learning rate and the remaining entries of Supplementary Table S7 are held fixed. Second, PSO is a heuristic population search, so the configuration it returns is the best one found inside that space rather than a certified optimum (Section 5).
Extended discussion of this subsection, with the further detail and the intermediate quantities it carries, is given in Supplementary Section S5.4.

2.4. PSO-Based Hyperparameter Optimization

PSO, proposed by Kennedy and Eberhart [24], is a population-based meta-heuristic inspired by the social foraging of bird flocks. Combining evolutionary algorithms with Transformer architectures for time series forecasting has begun to attract attention, but the integration of PSO with a Transformer for financial volatility forecasting remains limited: to the best of our knowledge no existing work applies PSO to Transformer hyperparameters for gold volatility prediction while incorporating multimodal sentiment factors from video sources, as Table 1 summarizes (Section 2.5).
Extended discussion of this subsection, with the further detail and the intermediate quantities it carries, is given in Supplementary Section S5.5.

2.5. Systems Approaches in Financial Modeling

Systems thinking is increasingly applied to financial market analysis, recognizing that market behavior emerges from the interaction of multiple subsystems rather than from individual factors in isolation. Salisu et al. [25] model the interaction between gold returns and the tail risk of the wider commodity system, using conditional autoregressive value-at-risk measures of own and oil tail risk as predictors of gold return behavior; Sakowski et al. [26] developed volatility-based risk assessment frameworks integrating multiple market indicators into composite monitoring systems; and McNeil et al. [27] provided a comprehensive treatment of quantitative risk measurement, including threshold-based alerting. Table 1 contrasts our system with the closest prior work along five dimensions.
Extended discussion of this subsection, with the further detail and the intermediate quantities it carries, is given in Supplementary Section S5.6.

3. Multi-Source Data Fusion System

This section describes the multi-source data fusion system, which constitutes the data perception layer of the overall prediction and decision support system. Its core function lies in perceiving, extracting, aligning and fusing heterogeneous information streams from multiple data sources into a unified, high-quality factor pool suitable for downstream volatility modeling. The overall architecture of the system is shown in Figure 1.

The output of the multi-source data fusion system is a 20-dimensional factor pool that serves as the input to the adaptive learning engine (Section 5). On the Federal Open Market Committee (FOMC) announcement day of the running example, this system ingests the 5-min K-line bars together with the video and news items published around the decision,

aligns them to the common event-time reference described in Section 6.1, and outputs the factor vector that the engine consumes.

Table 1. Comparison of the proposed framework with existing approaches.

Study	Data Modality	Sentiment Source	Model Architecture	Optimization	Systems Integration
GARCH [9]	Structured	None	Econometric	MLE	No
HAR-RV [12]	Structured	None	Econometric	OLS	No
Tang and Zhong [2]	Structured	None	AR + tail risk	OLS	No
Luo et al. [16]	Structured + text	Sentiment indices	Predictive reg.	OLS	No
Tang et al. [28]	Structured	None	PSO-GRU	PSO	No
Zhou et al. [20]	Structured	None	Transformer	Manual	No
Sakowski et al. [26]	Structured	None	Threshold-based	Rule-based	Partial
This study	**Structured + video + text**	**Video + news text**	**PSO-Transformer**	**PSO**	**Full**

Bold entries mark the row of the present study. Abbreviations: MLE, maximum likelihood estimation; OLS, ordinary least squares.

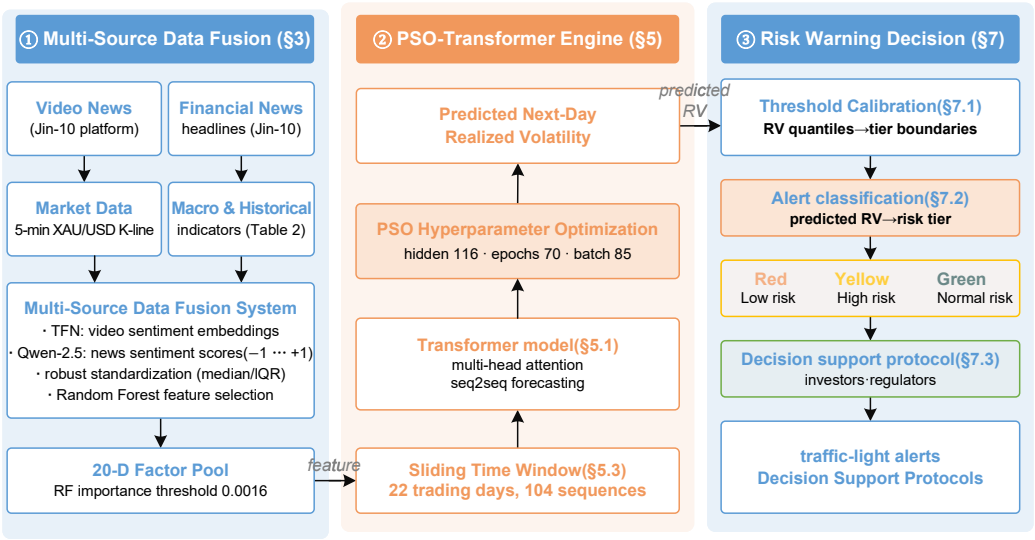


Figure 1. Overall architecture of the proposed gold volatility prediction and decision support system. The system comprises two core subsystems (multi-source data fusion system and risk warning decision system) bridged by a PSO-Transformer adaptive learning engine. The PSO optimization loop (right side of center panel) automatically tunes the three pre-registered Transformer hyperparameters of Supplementary Table S7; the remainder of the architecture is fixed. The arrows denote the flow of information from the data sources through the fusion and the learning stages to the decision output. **Labels printed inside the panel.** The news branch is annotated *"sentiment scores (−1 …+1)"* because −1 to +1 is the raw output range of the Qwen-2.5 scorer of Section 3.2; that score is an intermediate quantity, mapped by Equation (1) onto the five ordinal grades that actually enter the factor pool, as set out in Section 3.1. The three numerical labels inside the panel carry the settings used in this work and need no reinterpretation: the factor pool retained by Section 4.2 is 20-dimensional; the 22-day window of Equations (8) and (9) slides over 104 daily sequences; and the triple printed in the optimization block is the frozen (116, 70, 85) of Section 6.2.

- **Multimodal Sentiment Extraction Module:** Extracts emotional signals from financial news videos using a TFN, capturing visual, acoustic and linguistic modalities simultaneously.
- **Text-Based Sentiment Scoring Module:** Generates fine-grained sentiment scores from news headlines using the Qwen-2.5 large language model.
- **Realized Volatility Computation Module:** Calculates nonparametric volatility measures from high-frequency intraday price data.

- **Feature Selection Module:** Assesses variable importance using Random Forest and filters the factor pool to retain core predictors.

Extended discussion of this section, with the further detail and the intermediate quantities it carries, is given in Supplementary Section S5.7.

3.1. Multimodal Sentiment Extraction Module

This module extracts emotional signals from financial news videos by employing a TFN [4]. The TFN explicitly models all possible interaction combinations among unimodal, bimodal and trimodal features through outer product operations in high-order tensor spaces. Its architecture comprises three intramodal embedding subnetworks, each targeting a distinct modality: the linguistic subnetwork processes transcribed speech content, the visual subnetwork encodes facial expressions and gestures, and the acoustic subnetwork captures intonation, rhythm and pitch. The classifier is validated on 200 hand-annotated videos, on which tensor fusion attains a weighted F_1 of 0.68 against 0.61 for early fusion and 0.54 for a text-only baseline; the validation record of the classifier, including the items the study does not record, is documented in Supplementary Section S2.1.

Two multimodal fusion strategies are compared in Figure 2. In early fusion (left), the visual, auditory and linguistic features are directly concatenated into a single vector; this is structurally simple but prone to losing interaction information between modalities. In tensor fusion (right), outer product operations ($\otimes$) explicitly model all possible interaction relationships, including unimodal, bimodal and trimodal combinations, in high-dimensional space, forming a high-order tensor rich in cross-modal interaction information.

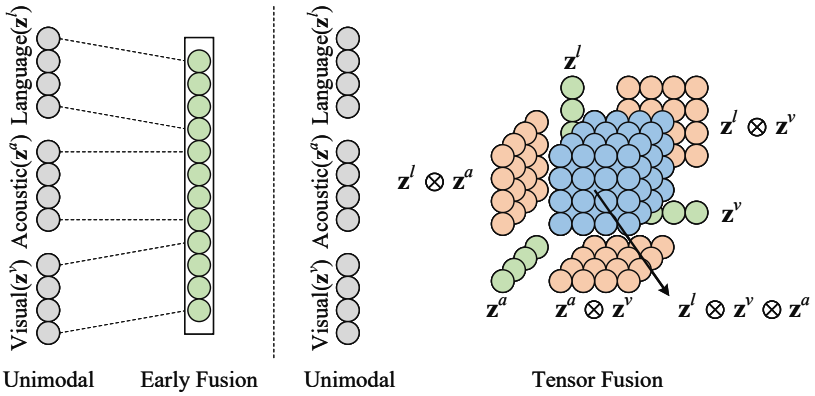


Figure 2. (**Left**): Early fusion (multimodal concatenation). (**Right**): Tensor fusion with three types of subtensors: unimodal, bimodal and trimodal. The outer product operation ($\otimes$) captures all interaction combinations across modalities. The fused representation is read out as a five-class label, which supplies the classifier's own validation target; that label is not the quantity the forecasting model consumes, because the video channel of the factor matrix carries the separate non-negative scalar of Section 3.1, whose construction from the video records of the auxiliary file is written out in Supplementary Section S2.2. The coloured circles denote the unimodal feature vectors, coloured by modality (linguistic, visual and acoustic).

Temporal alignment of the news, video and market channels. The emotional factor is aligned with the basic K-line data of XAU/USD by the publication time of each video rather than by the date on which the video was retrieved, so that the video channel is synchronized with market volatility in the time dimension. Three properties of the alignment are worth stating explicitly, because the forecasting design rests on them. First, all daily factors are stamped on one clock: the sentiment channels and the auxiliary macroeconomic columns at the 20:30 Beijing time snapshot of the stamping day, the price and technical columns at the

last available 5-min bar of that day, with Supplementary Table S10 stating the rule predictor by predictor together with the time-zone convention and the lag it implies (Section 6.1). Second, the ordering of inputs and target is enforced rather than assumed: the target of a sample is the realized variance of day $t + 1$ while every input is stamped on a day no later than t, so no factor can carry information from the day it is used to predict. Third, the aggregation of the two sentiment channels is fixed and pre-registered: each channel is reduced to one daily scalar by that same snapshot rule, and no per-item weighting is applied at any stage, the sentinel 0.001 marking a day on which no material item was recorded.

The two sentiment channels: what is weighted, and how each daily value is constructed. The question this paragraph answers is whether the individual information factors of Section 3.1 are weighted into a dynamic index, or whether the sentiment channel is the raw -1 to $+1$ score shown on the news branch of Figure 1; neither is the case, and the construction is fixed at every step. Three objects must be kept apart, because only one of them reaches the forecasting model. First, the raw score $s \in [-1, +1]$ that Qwen-2.5 returns for a headline is an intermediate quantity: it is stored and printed in the panel of Figure 1, but it is passed through Equation (1) before it is used, so the model never consumes it. Second, the TFN produces for each video a fused representation and a posterior over the five impact classes; the posterior validates the classifier and is not exported, while the scalar that the video channel carries is a separate non-negative quantity: it is read from the video column of the auxiliary file by the explicit rule written out in Supplementary Section S2.2, and it is never recomputed from the posterior. Third, the only weights estimated in the fusion system are the Random Forest importances of Section 4.2, used to keep or drop candidate columns rather than to weight the values of the retained ones; in particular, no weight attached to an individual news item or video varies over time, the embeddings of a video being combined by the fixed outer-product fusion whose parameters are estimated once during fine-tuning and then frozen. The daily series is therefore built by the following fixed sequence of steps. (i) Each video is mapped to one non-negative scalar and each headline to one integer grade $g(s) \in \{0, 1, 2, 3, 4\}$ through Equation (1). (ii) Each stream is reduced to a single daily value by taking the item published most recently before the 20:30 Beijing snapshot of the stamping day, not by averaging the items of the day and not by weighting recent items more heavily.

Extended discussion of this subsection, with the further detail and the intermediate quantities it carries, is given in Supplementary Section S5.8.

3.2. Text-Based Sentiment Scoring Module

Under the efficient market hypothesis [29], financial news serves as a key medium for transmitting macroeconomic dynamics, policy adjustments and changes in market sentiment, shaping investors' cognitive frameworks and emotional responses, which, in turn, affect asset price fluctuations [16]. Building on foundational advances in word embeddings [30] and pretrained language models [31], we adopt the Qwen-2.5 large language model [31] as the core engine for extracting sentiment signals from unstructured financial news text. A domain-adapted encoder of the FinBERT family is the conventional default for financial text, and the properties of this task that favor the instruction-tuned scorer here—the headlines are not in English, the factor requires a graded intensity rather than a three-way polarity label, the question is conditioned on the impact on gold prices, and the scorer has to be reproducible without an external service—are set out in Supplementary Section S5.9.

To calibrate the scores, we manually annotated a subset of 500 news headlines on the same -1 to $+1$ scale: the Pearson correlation between human and LLM scores was 0.81 and

the Spearman rank correlation 0.78, indicating strong agreement. The pipeline for extracting the news sentiment factor is depicted in Figure 3.

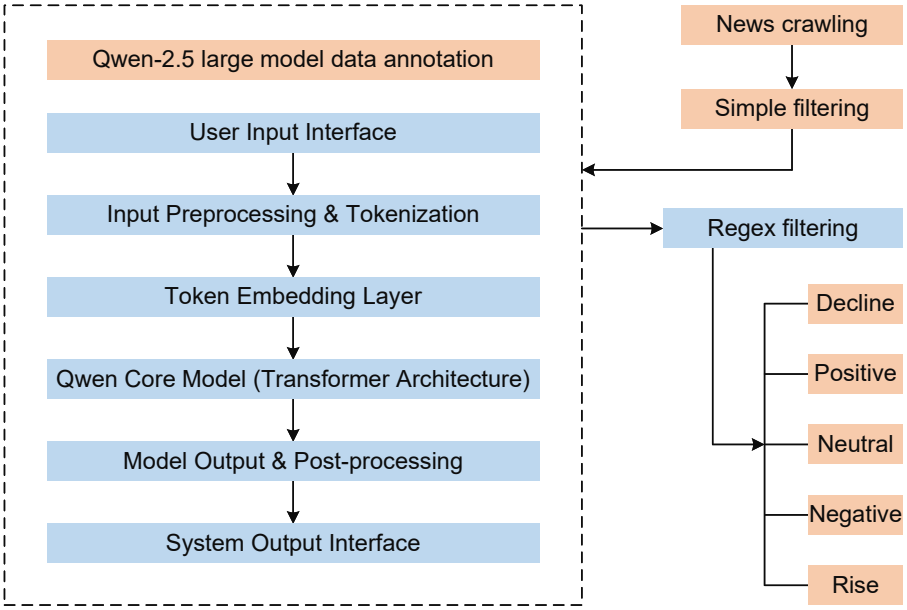


Figure 3. Pipeline for extracting textual sentiment scores from Jin-10 financial news headlines using the Qwen-2.5 large language model. Raw sentiment scores range from −1 (strongly negative) to +1 (strongly positive); they are then discretized onto the five-level ordinal scale defined in Equation (1), so that each headline contributes one integer grade; the daily news factor is the value of the last headline before the 20:30 Beijing snapshot, stored as the sentinel 0.001 when that grade is neutral, and it is the only text-side quantity the forecasting model consumes; the matching rule for the video channel and the value it contributes to the same matrix are stated in Section 3.1. The arrows denote the order in which each headline passes through the pipeline.

Five-level ordinal encoding of the news channel and the sentinel convention shared by the two channels. The news channel enters the factor pool on the five-level ordinal scale of Equation (1) and the video channel on the non-negative scalar of Section 3.1; the two are therefore not expressed in the same units, and their Random Forest importances in Table 3 are not directly comparable. Both are non-negative, both mark a daily value with no material signal by the same sentinel 0.001, and each can be removed independently in the ablation of Section 6.5. The continuous news score $s \in [-1, +1]$ produced by Qwen-2.5 is discretized by equal-width binning of the score interval into five integer grades $g \in \{0, 1, 2, 3, 4\}$,

$$g(s) = \min\big(4,\ \max(0,\ \lfloor 2.5\,(s+1)\rfloor)\big), \qquad s \in [-1, +1], \tag{1}$$

which partitions $[-1, +1]$ into the half-open intervals $[-1, -0.6)$, $[-0.6, -0.2)$, $[-0.2, 0.2)$, $[0.2, 0.6)$ and $[0.6, 1]$. The grades carry a directional interpretation that coincides with the five video-sentiment levels of Section 3.1: 0 = extremely bearish, 1 = bearish, 2 = neutral, 3 = bullish and 4 = extremely bullish. Applying Equation (1) yields a strongly neutral-centered distribution in the news channel: pooled over the 34,571 five-minute bars of the 126-day sample, the neutral grade, stored as the sentinel 0.001, accounts for 82.9% of the bars, while grades 0, 1, 3 and 4 account for 2.5%, 3.7%, 4.8% and 6.1%, respectively; on the daily 20:30 snapshot that supplies the daily factor value, the sentinel marks 94 of the 126 trading days, and the remaining 32 days carry a non-neutral grade. The video channel is sparser in its discrete part and continuous elsewhere: the sentinel covers 88.6% of the bars; the values 0, 1 and 4 appear on 2129 bars (6.2%) and are exactly the bars recorded as non-neutral; and the remaining 1808 bars (5.2%) carry a magnitude between 0.01 and

957.4, whose released grade is nevertheless neutral. On the daily snapshots, the sentinel marks 98 of the 126 days, 6 days carry the values 0 (2 days) and 4 (4 days) and are the six days the classifier calls non-neutral, and the other 22 days carry a non-zero magnitude with a neutral grade. The rule that produces these values from the video records is written out in Supplementary Section S2.2.

Extended discussion of this subsection, with the further detail and the intermediate quantities it carries, is given in Supplementary Section S5.9.

3.3. Realized Volatility Computation Module

In high-frequency financial data analysis, the sum of squared intraday returns—the *realized variance*, which the literature most often calls realized volatility and denotes RV—is widely adopted for its ability to depict market dynamics with high precision. As a non-parametric volatility measure constructed from intraday high-frequency price observations, RV directly reflects the actual fluctuation level of asset price paths by summing squared fine-grained returns within a trading day [32,33]. Under the ideal assumptions of no noise and continuous trading, when the sampling frequency approaches infinity, the realized variance converges in probability to the quadratic variation—the integrated variance—of the latent price process, an asymptotic property that establishes RV as a consistent estimator of the daily integrated variance of the latent price, not of its square root. The calculation method follows Andersen et al. [32]:

$$RV_t = \sum_{i=1}^{M} r_{t,i}^2 \tag{2}$$

where $r_{t,i}$ denotes the log return of the i-th 5-min interval on day t, and M is the number of intraday intervals, computed as $r_{t,i} = \ln(C_{t,i}/C_{t,i-1})$ on the continuous 5-min price series so that the first interval of each trading session also carries the overnight return; throughout the paper, returns are expressed in basis points, and $r_{t,i}^2$ and RV_t are therefore measured in bp^2. Two storage conventions matter for reproducing the sum. The squared return of each bar is kept in its own column at three-decimal precision in bp^2, and it is that column that is summed in Equation (2), not the square of the return column: the return column carries four decimals, a resolution of one basis point, so squaring it sets to zero the 9.75% of bars whose $|r| < 0.5$ bp and understates the daily realized variance by 0.22% at the median, 0.27% at the mean and up to 1.04% on the worst day of the sample. The very first bar of the sample has no predecessor, its return is zero by convention, and it is the only bar of which this is true. The overnight component is retained rather than excluded, and it is not a rounding detail: the 01:00 bar alone contributes about 3.6% of the pooled daily realized variance of the sample. Because the number of observed 5-min intervals varies slightly from day to day ($M_t \in [243, 276]$, mean 274.37), the quantity modeled in Section 6 and mapped to risk tiers in Section 7 is the per-interval realized variance $RV_t/M_t = M_t^{-1}\sum_{i=1}^{M_t} r_{t,i}^2$, expressed in bp^2 per 5-min interval. Following common usage, we continue to call the forecasting target *realized volatility* and write RV_t for it; strictly speaking, the sum of squared returns is the *realized variance*, and that—not its square root—is the numerical object modeled, tabulated and thresholded throughout the paper. Unless stated otherwise, every value reported below refers to this per-interval quantity, and no annualization and no other rescaling is applied anywhere in the paper.

Extended discussion of this subsection, with the further detail and the intermediate quantities it carries, is given in Supplementary Section S5.10.

3.4. Feature Selection Module

The feature selection module assesses variable importance and filters the factor pool accordingly to retain core predictors, thereby ensuring the input quality of the downstream

adaptive learning engine. The complete factor system is presented in Table 2; because these candidate variables are sampled at different frequencies and become observable at different times, the exact availability and alignment rule of every one of them—frequency, stamping time, time-zone convention, lag and alignment with the forecast origin—is stated in Supplementary Table S10 of Section 6.1.

Table 2. Impact factor indicators for gold volatility prediction: 20 exogenous candidate predictors in five dimensions (price, trend, market sentiment, historical and transaction information) plus the forecast target, which is not a candidate. The candidate matrix of the Random Forest has 21 columns, namely, these 20 predictors together with the one-day-lagged realized volatility RV_{t-1}; the contemporaneous target RV_t, equivalently RV_t/M_t on the reported scale, is never a column of it.

Category	Variables
Forecast target	Realized Volatility RV_t (reported as RV_t/M_t)—*not a predictor and not a candidate column: the model and the selector receive only its one-day lagged history RV_{t-1}, which is counted among the 21 candidates*
Price Information	Open Price, Close Price, High Price, Low Price, Adj_Close
Trend Indicators	trend-following: MA10, MACD, Signal_Line, Histogram; bounded oscillator: RSI
Market Sentiment	Video Sentiment Factor, Public Sentiment Factor
Historical Information	Intraday Return Volatility, Spread, VIX_H_L, Oil_H_L, Gold_H_L, Lag_Log_WP
Transaction Variables	Tick Volume, Log_WP

Italic entries mark the forecast target row, which is not a candidate predictor. Abbreviations: MA, moving average; MACD, moving average convergence divergence; RSI, relative strength index; VIX, volatility index.

Selection criteria for the factor set. Two tiers of selection are involved in the construction of the factor pool, and they are kept apart because they answer different questions. The accounting of the released auxiliary file belongs to the first tier: the file contains 26 columns, namely, the twenty exogenous predictors of Table 2 in their five dimensions (five price columns, five technical columns, two sentiment columns, six historical columns and two transaction columns), the squared five-minute return column from which the realized-volatility target is aggregated, and the five macroeconomic and exchange-traded fund (ETF) columns.
Extended discussion of this subsection, with the further detail and the intermediate quantities it carries, is given in Supplementary Section S5.11.

1. *Channel coverage.* Each candidate must represent one of the five information channels through which the gold-volatility literature documents a response: price discovery, trend and positioning, market sentiment, realized-risk history and liquidity or transaction intensity. The five dimensions of Table 2 are that coverage requirement rather than a post hoc label, and no channel is left without a representative.
2. *A documented relation to gold or commodity volatility.* A candidate is admitted only if its channel has an established empirical relation to volatility, which is the criterion that populates the trend, historical-risk and transaction blocks. The parameters of the technical block are conventional rather than fitted.
3. *Observability at the forecast origin.* Every candidate must be measurable at or before the origin—the 23:55 Beijing close of the stamping day for the market, technical and historical columns, the 20:30 snapshot for the two sentiment columns—and no value dated on the target day or later may enter. Supplementary Table S10 documents this requirement variable by variable.
4. *Availability and continuity over the sample.* A documented relation is necessary but not sufficient: the series must also exist, and exist continuously, in the released sample.

This is the criterion that removes the five macro/ETF columns, and it is the reason the macroeconomic block is absent from the predictor pool rather than present with an untested importance.

5. *Non-redundancy within the pool*. No two candidates may be near-duplicates of each other. The criterion removes the five placeholders a second time and restricts each technical family to one standard member instead of several parameterizations of the same series; the post hoc screening of Section 4.2 then measures how far the entrants still are from one another.

4. Data Processing

This section describes the data preprocessing pipeline that connects the multi-source data fusion system to the adaptive learning engine. Two key operations are performed: robust standardization to normalize heterogeneous feature scales and Random-Forest-based feature selection to distill the factor pool.

Extended discussion of this section, with the further detail and the intermediate quantities it carries, is given in Supplementary Section S5.12.

4.1. Data Standardization

Data standardization maps variables with diverse dimensions, scales and distributional characteristics into a unified numerical range, thereby mitigating interference in model training caused by scale discrepancies among features. This study adopts robust scaling, which transforms input features using statistical measures resistant to outliers, specifically the median and the interquartile range (IQR), thereby alleviating the impact of extreme values while preserving their relative characteristics in the original dataset. The transformation is defined as follows:

$$X_{\text{scaled}} = \frac{X - \text{median}}{\text{IQR}} \tag{3}$$

where median represents the median of the sample data, and IQR denotes the interquartile range. Three properties of this step belong to the procedure rather than to the formula. First, the two statistics are estimated on the training prefix of the daily series only and are then applied unchanged to every later day, so that the scaling introduces no information from the evaluation dates; the same rule and the same fitted statistics are used for all ten models of this paper. Second, the target is standardized by the same rule with the same training-prefix fit, and every metric reported in Section 6 is computed after the inverse transformation, that is, on the original RV_t/M_t scale, so that no reported number is expressed in scaled units. Third, no clipping, winsorization or trimming is applied: extreme returns are the object of study for a volatility target and are retained, and it is the median-and-IQR denominator rather than a cut-off that limits their leverage. A guard exists for a degenerate column: if the interquartile range of a column is numerically zero, the divisor is replaced by one instead of being left at zero, a guard that is inactive in this sample because no retained column is constant.

4.2. Feature Selection

The predictive performance of deep learning models depends largely on the quality, relevance and representativeness of the input features [34]. Excessively low feature dimensions may cause underfitting due to information deficiency, whereas excessively high dimensions may introduce redundant variables or irrelevant noise, thereby increasing model complexity and the risk of overfitting. This study employs Random Forest feature ranking [35] to assess the contribution of each candidate variable to volatility prediction; Table 3 presents the feature importance scores. Two diagnostics accompany the ranking.

Because the candidate columns are far from mutually orthogonal, Table S4 screens them for multicollinearity—pairwise Pearson correlations together with variance inflation factors—and shows that its first three rows are collinear blocks whose members are deterministic transforms of one price series, so their importances are read jointly rather than one by one, and that no remaining pair approaches that degeneracy. Table S5 then recomputes the attribution with exact TreeSHAP values, and the shares reproduce the ordering of Table 3 at a Spearman correlation of 0.96 with the same single rejection at the 0.0016 cutoff. The cutoff is an a priori value rather than a fitted one, and Table S6 reports how far the forecast moves when it is varied, so the retention decision is not read as a tuned quantity.
Extended discussion of this subsection, with the further detail and the intermediate quantities it carries, is given in Supplementary Section S5.13.

Table 3. Variable importance by Random Forest model. The scores are the normalized Gini importances of a 500-tree forest fitted to the training segment on the 21 candidate columns, that is, the 20 exogenous predictors of Table 2 together with the one-day-lagged realized volatility RV_{t-1} (the contemporaneous target is not among them); ✓ marks the variables retained at the 0.0016 cutoff of Section 4.2 and × the one variable rejected.

Variable	Importance	Retained
Realized Volatility (RV_{t-1})	0.451087	✓
Oil_H_L	0.124830	✓
Histogram	0.084059	✓
Spread	0.041604	✓
Gold_H_L	0.035713	✓
Signal_Line	0.032587	✓
Intraday Return Volatility	0.032103	✓
Low_Price	0.030690	✓
Tick Volume	0.023152	✓
MA10	0.019780	✓
RSI	0.017408	✓
Open Price	0.017303	✓
Close Price	0.012688	✓
Lag_Log_WP	0.012245	✓
High Price	0.011810	✓
Public Sentiment Factor	0.011501	✓
Adj_Close	0.011494	✓
Log_WP	0.010525	✓
MACD	0.010086	✓
Video Sentiment Factor	0.008169	✓
VIX_H_L	0.001165	×

Abbreviations: MA, moving average; MACD, moving average convergence divergence; RSI, relative strength index; VIX, volatility index.

5. PSO-Transformer Adaptive Learning Engine

This section presents the adaptive learning engine that bridges the multi-source data fusion system and the risk warning decision system. The engine comprises four components: the Transformer architecture for nonlinear temporal modeling, the PSO algorithm for the automatic search over the three pre-registered hyperparameters of Supplementary Table S7, the sliding time window method for sequential data processing, and the evaluation criteria for performance assessment.
Extended discussion of this section, with the further detail and the intermediate quantities it carries, is given in Supplementary Section S5.14.

5.1. Transformer Model

The multi-factor volatility prediction problem can be formulated as a sequence-to-sequence (seq2seq) learning task [17]. Under this framework, the Transformer addresses the complex requirements of multi-factor modeling through its self-attention mechanism, which captures long-range dependencies between any two time steps in the input sequence, thereby overcoming the vanishing and exploding gradient limitations inherent in traditional recurrent neural networks [36,37]. The core of the Transformer is the scaled dot-product attention,

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{4}$$

where Q, K and V denote the query, key and value matrices, respectively, and d_k is the dimension of the key vectors. The multi-head attention mechanism extends this by partitioning the attention computation into h parallel heads,

$$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, \ldots, \text{head}_h)W^O \tag{5}$$

where $\text{head}_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V)$, which enables the model to learn different representation patterns of factors in multiple subspaces in parallel [17,38].

Because the self-attention mechanism does not rely on sequential recursive processing, the Transformer exhibits strong parallel computing capabilities during training, improving both training efficiency and scalability [17]. Traditional multi-step prediction strategies fall into two paradigms: the iterative method (Figure 4), which uses a single-step model to extrapolate future values progressively but suffers from error accumulation, and the direct method (Figure 5), which builds independent models for each prediction step but ignores the dependency structure across steps.

Extended discussion of this subsection, with the further detail and the intermediate quantities it carries, is given in Supplementary Section S5.15.

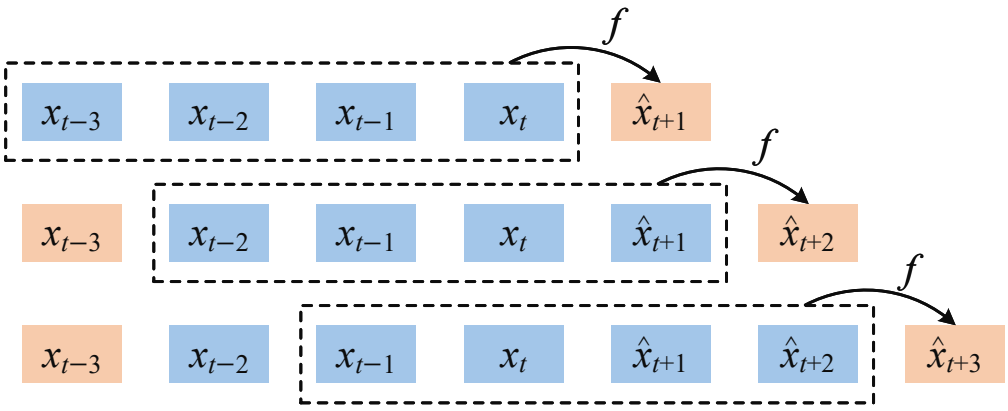


Figure 4. Iterated multi-step forecasting. A single-step model is applied recursively, feeding each prediction back as input, so prediction errors accumulate over the horizon.

5.2. Particle Swarm Optimization Algorithm

Within the adaptive learning engine, PSO serves as the hyperparameter optimization engine. As introduced in Section 2.4, PSO maintains a set of candidate solutions (particles) within the solution space; by using each particle's historical best experience and the globally shared optimal information, it collaboratively updates velocity and position, contracting towards a common region of the search space rather than towards a certified optimum. The velocity and position updates follow:

$$v_i^{t+1} = \omega v_i^t + c_1 r_1 (\text{pbest}_i^t - x_i^t) + c_2 r_2 (\text{gbest}^t - x_i^t) \tag{6}$$

$$x_i^{t+1} = x_i^t + v_i^{t+1} \tag{7}$$

where v_i^t denotes the velocity of particle i at iteration t, x_i^t represents its position, ω is the inertia weight [39,40], c_1 and c_2 are learning rates, pbest_i^t is the best position visited so far by particle i, gbest^t is the best position visited so far by the swarm—the standard PSO notation, which names the best point found rather than a certified global optimum—and $r_1, r_2 \in [0, 1]$ are random numbers.

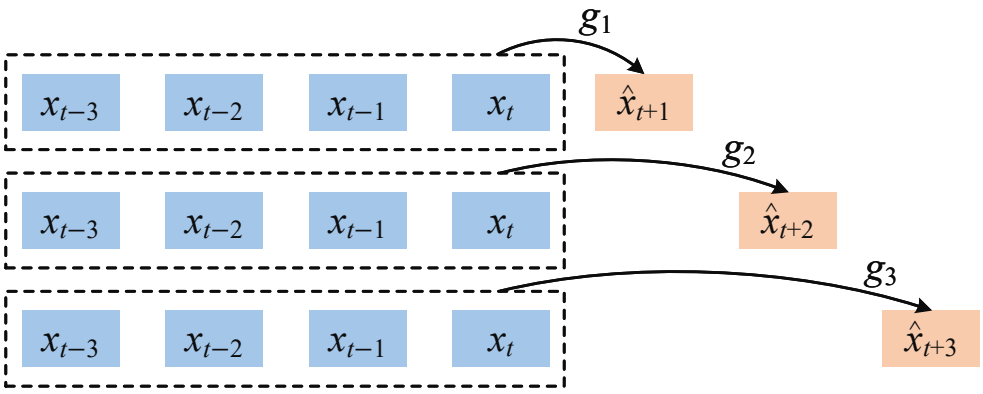


Figure 5. Direct multi-step forecasting. An independent model is trained for each prediction step, avoiding error accumulation but ignoring the dependency structure across steps.

Integer handling, boundary treatment and termination. All three searched quantities are integers, whereas Equations (6) and (7) move a particle in continuous space, so a particle is evaluated at the rounded position $\text{round}(x_i^t)$, and the configuration reported is the rounded position of the best particle found. Boundaries are handled by projection rather than by reflection or damping: after the velocity update, the position is clipped to the box, $x_i^{t+1} \leftarrow \text{clip}\left(x_i^t + v_i^{t+1}, \text{LB}, \text{UB}\right)$ with $\text{LB} = (40, 18, 64)$ and $\text{UB} = (128, 80, 256)$, so a particle that reaches a bound stays on it until the velocity term pulls it back; no separate velocity clipping is applied, because the clipped position already bounds the effective step. The search runs for a fixed 20 iterations, i.e., 600 model evaluations for a swarm of 30 particles, with no patience-style early stopping at the swarm level, although each candidate network is early-stopped on its validation loss, so the iterative curve of Table 4 may be flat for several consecutive iterations.

Extended discussion of this subsection, with the further detail and the intermediate quantities it carries, is given in Supplementary Section S5.16.

Table 4. Hyperparameter evolution during PSO iteration. The MAE column is the selection criterion of Section 5.2, computed on the validation window of Supplementary Table S11 (under protocol P1, this window is the held-out test window). The MAE column is in bp^2 per 5-min interval.

Iteration	Model Dimension	Epochs	Batch Size	MAE
1	96	73	155	7.150
5	96	73	155	7.150
10	96	73	155	7.150
15	116	70	85	6.684
20	116	70	85	6.684

5.3. Sliding Time Window Method

The sliding time window method processes time series data by setting a fixed-length window on the time axis and sliding it step-wise forward to conduct local statistical analysis. Over the 126 retained trading days, this yields 104 daily sliding-window samples, of which the first 83 form the training set and the final 21 the out-of-sample test window, a configu-

ration that balances historical memory capacity with responsiveness to recent dynamics (Figure 6) [12].

Formal definition of the sliding-window samples. Let $t = 1, \dots, D$ index the retained trading days, and let M_t denote the number of 5-min bars recorded on day t; in our sample, $D = 126$ and $243 \le M_t \le 276$ with mean 274.37. With window length $W = 22$ and a step of one trading day, the j-th sample is the ordered pair

$$\begin{aligned} \mathcal{S}_j &= \left(\mathbf{X}_{j-W:j-1}, y_j\right), \qquad \mathbf{X}_{j-W:j-1} = [\mathbf{x}_{j-W}, \dots, \mathbf{x}_{j-1}] \in \mathbb{R}^{W \times d}, \\ y_j &= \frac{RV_{t_j}}{M_{t_j}} = \frac{1}{M_{t_j}} \sum_{i=1}^{M_{t_j}} r_{t_j,i}^2, \qquad j = W+1, \dots, D, \end{aligned} \tag{8}$$

where $\mathbf{x}_t$ collects the $d = 20$ retained daily factors of day t, and $r_{t,i}$ is the 5-min log return of Equation (2). The number of generated daily samples is therefore

$$J = D - W = 126 - 22 = 104, \tag{9}$$

of which the first $\lfloor 0.8J \rfloor = 83$ form the training set, and the remaining 21 form the out-of-sample test window. Because the step equals one trading day, every sample predicts a *different* target day, so the number of generated sequences (104) and the number of independent forecast origins (104) coincide. This is the central methodological difference from a bar-level sliding window, where consecutive windows overlap by $(W-1)/W$, and N_{gen} sequences carry only $\lfloor L/W \rfloor$ non-overlapping forecast origins (Supplementary Table S8); reporting N_{gen} in place of the independent count inflates the apparent sample size by a factor of W.

Extended discussion of this subsection, with the further detail and the intermediate quantities it carries, is given in Supplementary Section S5.17.

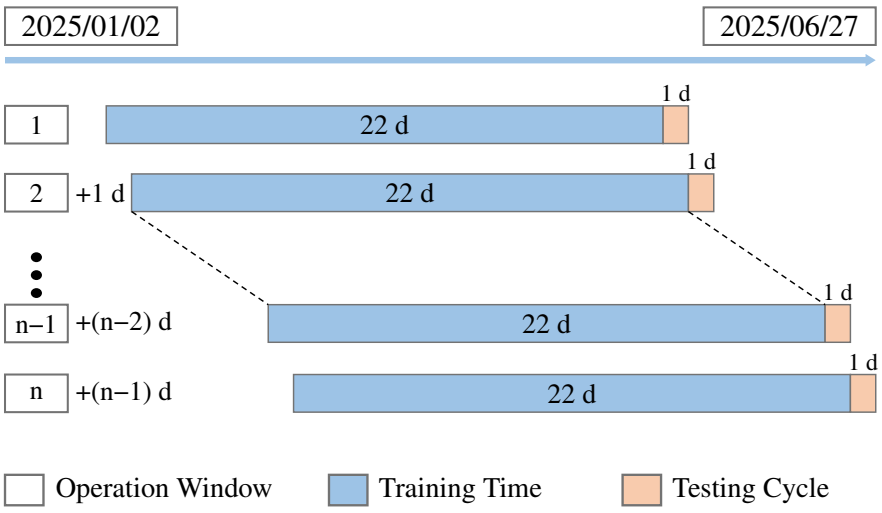


Figure 6. Sliding time window structure. Window length $W = 22$ trading days and a prediction horizon of one trading day; consecutive windows are displaced by exactly one trading day (the +1 d offsets), so that the $J = 104$ windows defined in Section 5 predict 104 distinct target days, and the number of generated sequences coincides with the number of independent forecast origins (Supplementary Table S8). The time axis spans the sample, 2 January–27 June 2025.

5.4. Model Evaluation Criteria

To assess prediction accuracy from several angles, six complementary error metrics are employed:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$$

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \times 100\%$$

$$SMAPE = \frac{1}{n}\sum_{i=1}^{n}\frac{|y_i - \hat{y}_i|}{(|y_i| + |\hat{y}_i|)/2} \times 100\%$$

$$RMSPE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(\frac{y_i - \hat{y}_i}{y_i}\right)^2}$$

where y_i is the true value, $\hat{y}_i$ is the predicted value, and n is the number of observations. MAPE and SMAPE are expressed in percent, the factor 100 appearing in their definitions above, whereas RMSPE carries no such factor and is therefore reported as a proportion; the columns of Table 5 and of Tables 6–12 follow this convention, and their headers label the three scale-dependent columns explicitly. The identity RMSPE × 100 ≥ MAPE, which a root mean square must satisfy once both quantities are placed on the common percent scale, holds for every model and horizon reported here, so a tabulated RMSPE above unity is a proportion above one rather than a percentage above 100%: as a percentage, the XGBoost entry of Table 6 is 157.8%, and the RMSPE 0.415 of the one-step PSO-Transformer row of Table 7 is the proportion 41.5%, to be read against the MAPE 26.76% of that row. The name of the measure is retained from the volatility literature even though the quantity computed here is a ratio and not a percentage. The target y_i is the per-interval realized variance RV_t/M_t of Section 3.3, expressed in bp^2 per 5-min interval without annualization, so MAE and RMSE are reported in bp^2 per 5-min interval, MSE in its square, while MAPE, SMAPE and RMSPE are dimensionless.

6. Empirical Research: System Evaluation

Throughout this section, the in-sample and held-out test MAE of Table 5, and the rolling-origin error of Supplementary Table S15, are the quantities used to discuss overfitting, while all benchmark comparisons use the test-set MAE (Tables 6–12). This section evaluates the performance of the integrated prediction and decision support system; the experimental workflow is illustrated in Figure 7. Unless a caption states otherwise, every table of Sections 6.3–6.8 is computed under **protocol P1** of Section 6.1: the 21 days of 30 May–27 June 2025 serve both as the validation sample on which the hyperparameters and the best training epoch are selected and as the evaluation window on which the errors are then measured, so the P1 figures are within-tuning-window results and not a fully independent out-of-sample comparison. **Protocol P2** is the leak-free counterpart, with a disjoint validation period of 20 trading days (2–29 May 2025) and a test window read once; it is reported for the proposed system and the two fixed recurrent baselines in Supplementary Sections S5.19 and S5.24 (Supplementary Table S18), and it does not preserve the P1 ranking of the proposed system (Section 8.1).

Extended discussion of this section, with the further detail and the intermediate quantities it carries, is given in Supplementary Section S5.18.

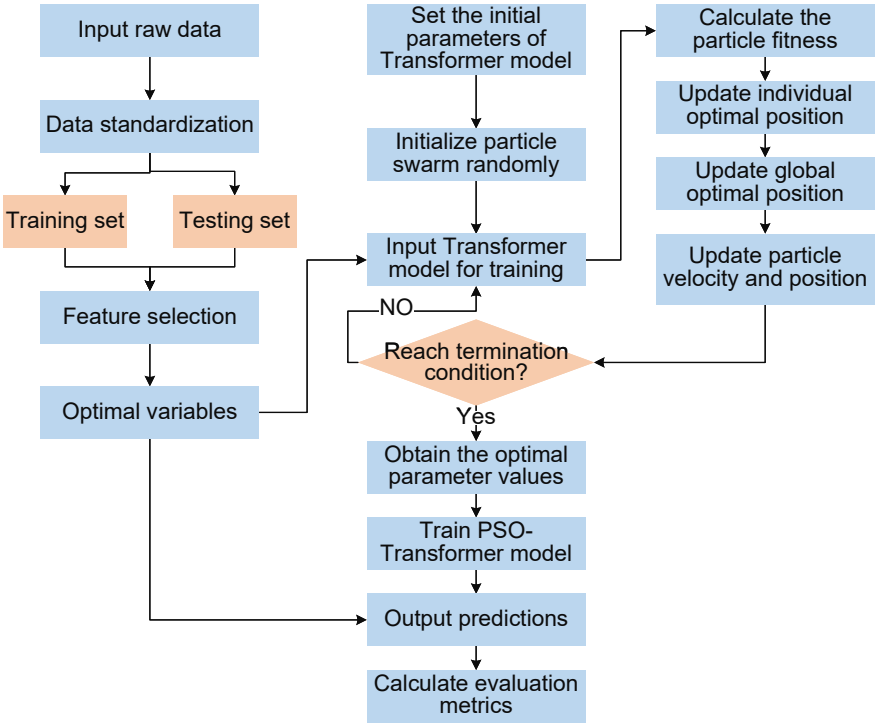


Figure 7. Training workflow of the PSO-Transformer adaptive learning engine. The particle swarm iteratively evaluates hyperparameter configurations on the validation split and returns the best configuration for final training.

6.1. Data Description

The sample comprises 126 trading days covering 2 January–27 June 2025. No trading day inside that window was removed for incomplete records: 127 weekdays fall between those two dates; the single weekday absent from the source file is Good Friday, 18 April 2025, on which the spot-gold market is closed, and 1 January 2025 lies outside the window, so no date was dropped and none were aggregated away. The five macro/ETF columns of that auxiliary file—gold ETF holdings, non-farm payrolls, China and US CPI and the policy rate—are not used as predictors: under the snapshot rule of Supplementary Table S10, they behave as a single placeholder series repeated five times (minimum pairwise correlation 0.99995 across the 126 days, and the same value 0.001 on between 100 and 103 of them), so they are removed by the non-redundancy criterion of Section 3.4, which leaves the 21 candidate columns listed in Table 3. The reported runs were executed on a Linux node with one NVIDIA GeForce RTX 4090 D under an isolated Python 3.12.13 virtual environment; Supplementary Table S14 lists the version of every library stage by stage, and the two supplementary analyses computed on a second, newer stack are identified in Supplementary Section S5.19, and the dependency list shipped with the code gives lower bounds for those libraries rather than pinned versions, as Supplementary Sections S2 and S5.19 record.

The sequence of samples and what the tabulated significance means. The experiments form a four-step sequence whose samples do not coincide. The models are fitted on the training segment; the PSO search of Section 5.2 and the grid search of Section 6.6 select their hyperparameters and their best epoch on a validation sample; the forecasts and the error metrics of Tables 6–12 are computed on the 21 held-out days of 30 May–27 June 2025; and the pairwise tests of Section 6.7 are then applied to those out-of-sample errors. Under protocol P1, which governs every table of Sections 6.3–6.8, the validation sample is those same 21 days, so the selection that precedes the test and the test itself are not disjoint, and the significance reported in Section 6.7 is to be read as conditional on the tuning window;

protocol P2 (the leak-free variant) separates the two by moving the selection to 2–29 May 2025, and it is reported for the proposed system and the two recurrent baselines without a significance table (Supplementary Section S5.24). Supplementary Tables S12 and S13 give the segment consumed by each step and the sequence itself, in order, and Supplementary Section S5.19 gives the same accounting in continuous form.

Extended discussion of this subsection, with the further detail and the intermediate quantities it carries, is given in Supplementary Section S5.19.

Table 5. Training and held-out test MAE comparison (one-step, with sentiment factors). Under protocol P1 of Supplementary Table S11, the estimation sample is the 83 pre-test sliding-window samples, and the held-out segment is the final 21 trading days (30 May–27 June 2025), so the second column is the test-set MAE, and the pair is the overfitting diagnostic discussed in the text; protocol P2 of Section 6.1 adds a separate 20-day validation segment between the two, at the cost of a different configuration. The MAE values are in bp^2 per 5-min interval, on the non-annualized scale of Section 3.3.

Model	Training MAE	Test MAE (Held Out)	Gap
GARCH	29.390	14.981	−14.409
HAR-RV	24.701	12.920	−11.781
LSTM	27.957	8.925	−19.032
GRU	28.008	8.730	−19.277
Transformer	13.608	8.610	−4.998
PSO-Transformer	10.143	8.038	−2.105
SVM	12.624	25.311	+12.687
XGBoost	0.051	38.026	+37.974

MAE is reported on the paper-comparable scale (RV_t/M_t, i.e., the mean squared 5-min return of the day, in bp^2); deep learning values are computed from the seed-averaged forecast over 20 independent runs, whereas the deterministic baselines are reported as single runs; the run-to-run mean and standard deviation of the deep-learning MAE are given in Supplementary Table S16. The Gap column reports the held-out test MAE minus the training MAE, and this pair of columns is the overfitting diagnostic read in the text.

6.2. PSO Hyperparameter Optimization

The global best is unchanged through the tenth iteration, improves at the eleventh and again at the twelfth, and then remains constant for the rest of the search, as the convergence curve of Figure 8 and the iteration record of Table 4 show.

By the twelfth iteration, the search has settled on its best configuration: model dimension $d_{\text{model}} = 116$, number of training epochs = 70 and batch size = 85, corresponding to a validation MAE of 6.684. This number is the PSO selection criterion of Section 5.2, evaluated on the validation window of Supplementary Table S11; under protocol P1, which produced Tables 6–12, that window coincides with the held-out test window, whereas under protocol P2, it is the 20-day validation period. The search runs on the same single-seed budget as the PSO row of Table 9, 30 particles over 20 generations, i.e., 600 model evaluations, which, on a single NVIDIA RTX 4090 GPU, cost about six minutes (369 s, or 0.62 s per evaluation).

Extended discussion of this subsection, with the further detail and the intermediate quantities it carries, is given in Supplementary Section S5.20.

6.3. One-Step Forecasting Without Sentiment Factors

To establish a baseline for evaluating the incremental contribution of the data fusion system's sentiment modules, this subsection reports one-step prediction results excluding market sentiment information, using only price and fundamental data. All models, including the econometric models (GARCH, HAR-RV), the traditional machine learning models (SVM, XGBoost) and the deep learning models (LSTM, GRU, Transformer, PSO-

Transformer), are evaluated on the same target; the econometric and machine-learning models are deterministic and were evaluated once, while each deep learning model was independently trained and evaluated 20 times under different random seeds, and for each metric, the tabulated value is computed from the 20-run seed-averaged forecast, with the run-to-run means and standard deviations of the same metric reported in Supplementary Table S16. All values in this subsection are measured under protocol P1 of Section 6.1, whose validation sample is also the evaluation window.

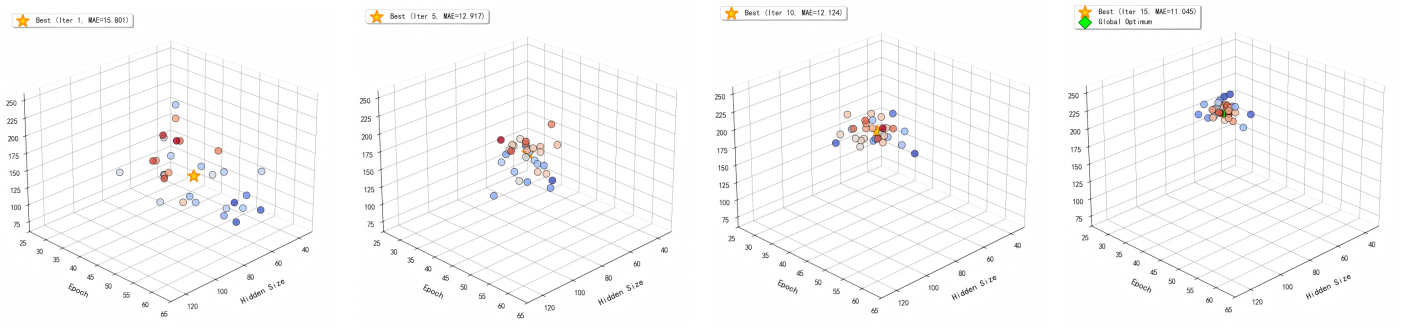


Figure 8. Particle position evolution during PSO optimization. Each circle denotes one particle of the swarm. Iterations 1, 5, 10 and 15 are shown.

Specifications of the benchmark models. The seven benchmark models fall into three families, and their equations are stated here because the two one-step tables must be read against them. Every model forecasts the same target as the proposed system, the per-bar realized volatility $y_t = RV_t/M_t$ of Equation (2); each baseline is fitted to the daily realized volatility RV_t, and its forecast is divided by the number of intraday bars M_t of the forecast date before the metrics are computed, so the tabulated numbers of all models are on one scale. *Econometric recursions.* Both GARCH variants are estimated on the daily realized volatility of the training segment, the first 100 of the 126 trading days, with an AR(1) conditional mean

$$y_t = \mu + \phi y_{t-1} + \varepsilon_t, \qquad \varepsilon_t = \sigma_t z_t, \tag{10}$$

with $z_t \sim \mathcal{N}(0,1)$ drawn independently across t, and a GARCH(1,1) conditional variance

$$\sigma_t^2 = \omega + \alpha \varepsilon_{t-1}^2 + \beta \sigma_{t-1}^2 \tag{11}$$

for GARCH, replaced by

$$\sigma_t^2 = \omega + \left(\alpha + \gamma\, \mathbf{1}\{\varepsilon_{t-1} < 0\}\right)\varepsilon_{t-1}^2 + \beta \sigma_{t-1}^2 \tag{12}$$

for GJR-GARCH, where $\mathbf{1}\{\cdot\}$ is the indicator of a negative return shock. Both are fitted jointly by Gaussian quasi-maximum likelihood, and their point forecast is the fitted conditional mean of Equation (10) evaluated at the last observation available at the forecast origin s,

$$\hat{y}_{s+h|s} = \hat{\mu} + \hat{\phi}\, y_s, \tag{13}$$

so the conditional-variance recursion enters the likelihood but not the tabulated point forecast; the QLIKE entries of these two rows in Supplementary Table S19 are therefore the loss of that level forecast. HAR-RV [12] is the OLS regression

$$\hat{y}_{s+h|s} = \hat{c} + \hat{\beta}_d\, y_s + \hat{\beta}_w\, \bar{y}_s^{(5)} + \hat{\beta}_m\, \bar{y}_s^{(22)}, \tag{14}$$

estimated on the same training segment, where $\bar{y}_s^{(J)} = J^{-1} \sum_{i=0}^{J-1} y_{s-i}$ is the average of the last J observations, so that $\bar{y}_s^{(5)}$ and $\bar{y}_s^{(22)}$ are the weekly and the monthly average of the

realized volatility. *Neither recursion contains an exogenous regressor*: both are univariate in the realized-volatility series, so their rows are numerically identical in Tables 6 and 7, as Section 6.4 explains.

Two augmented versions of the same recursions place the retained factor block inside the equation that is forecast from, and they are stated here so that the benchmark family is complete. Write $z_{k,s}$ for the k-th column of the retained block of Section 4.2 on day s, in the robustly scaled units of Section 4.1; HARX(1,d,w,m) adds the block to Equation (14),

$$\hat{y}_{s+h|s} = \hat{c} + \hat{\beta}_d\, y_s + \hat{\beta}_w\, \bar{y}_s^{(5)} + \hat{\beta}_m\, \bar{y}_s^{(22)} + \sum_{k=1}^{K} \hat{\gamma}_k z_{k,s}, \tag{15}$$

and GARCH-X adds it to the conditional mean of Equations (10) and (11),

$$y_t = \mu + \phi\, y_{t-1} + \boldsymbol{\gamma}^{\top}\mathbf{z}_{t-1} + \varepsilon_t, \qquad \varepsilon_t = \sigma_t z_t, \qquad \sigma_t^2 = \omega + \alpha\varepsilon_{t-1}^2 + \beta\sigma_{t-1}^2, \tag{16}$$

so that the point forecast of Equation (13) becomes $\hat{y}_{s+h|s} = \hat{\mu} + \hat{\phi}\, y_s + \hat{\boldsymbol{\gamma}}^{\top}\mathbf{z}_s$. Both augmented versions were estimated on the same 78 origins and scored on the same 21 days under protocol P1 as the restricted recursions (least squares for HARX, Gaussian quasi-maximum likelihood for GARCH-X), and the restricted rows were re-estimated along that same code path rather than transcribed. Neither augmentation yields a forecast that the evaluation window supports: HARX fits the training segment far better than the restricted recursion (R^2 0.293 → 0.530), yet its held-out MAE rises from 12.937 to 65.200, and restricting it to the eleven columns that are neither price levels nor their moving averages still raises it to 27.396, both differences being significant in favor of the restricted recursion; GARCH-X moves the one-step MAE from 8.976 to 8.210, a difference the same test does not resolve ($p = 0.35$), and its full-block coefficients are not identified by 78 daily observations, the likelihood rising while the evaluation-window error worsens. Supplementary Section S3.1 and Tables S23 and S24 give the estimation protocol, the re-estimated restricted rows, the coefficient-level detail and the significance tests. The tabulated rows of Tables 6 and 7 therefore keep the univariate form of Equations (11) and (14): on this sample, the exogenous information is recovered by the learned models of Sections 5 and 6.5, not by a parametric recursion.

Run-to-run stability of the deep-learning results. The deep-learning entries of Tables 7, 11 and 12 summarize twenty independent training runs, and a summary is incomplete without its dispersion: Supplementary Table S16 reports, for each model and horizon, the mean and standard deviation of the test MAE over twenty runs together with the coefficient of variation, every run drawing its own weight initialization and batch ordering from the seed assigned to it, while the training and validation windows are held fixed, so the spread is generated entirely by the initialization. Two features stand out. First, the architectures differ sharply in stability: the LSTM is the most stable at every horizon (coefficient of variation 4.8–6.5%), the bare Transformer the least stable (19.4–27.4%), and the PSO-Transformer the most stable member of the Transformer family (8.5–15.3%). Second, stability and accuracy are not the same quantity, and the ordering of the four architectures is not invariant to the way the twenty runs are summarized: under the seed-averaged forecast, the proposed system has the lowest one-step and two-step MAE of the four and the GRU the lowest five-step MAE, whereas under the per-run average, it is best at one step but is overtaken by the LSTM at two and five steps, so the multi-step ranking of the deep architectures is unresolved, and the five-step margin over the best recurrent baseline lies inside the run-to-run variation.

Extended discussion of this subsection, with the further detail and the intermediate quantities it carries, is given in Supplementary Section S5.21.

Table 6. One-step prediction results without market sentiment factors. Computed under protocol P1 (Section 6.1). Scale convention of Section 5.4: MAPE and SMAPE are percentages; RMSPE is a dimensionless proportion.

Model	MSE (bp^2)	RMSE (bp^2)	MAE (bp^2)	MAPE (%)	SMAPE (%)	RMSPE (prop.)
GARCH	315.636	17.766	14.981	42.126	53.018	0.478
HAR-RV	227.481	15.082	12.920	47.851	34.184	0.701
LSTM	119.943	10.952	8.895	30.393	25.893	0.451
SVM	706.964	26.589	24.475	88.484	54.484	1.155
XGBoost	1609.917	40.124	36.078	127.060	68.492	1.578
GRU	116.222	10.781	8.682	29.926	25.276	0.447
Transformer	119.232	10.919	8.814	29.515	25.388	0.432
PSO-Transformer	**115.245**	**10.735**	**8.401**	**27.974**	**24.427**	**0.424**
Persistence	188.355	13.724	10.586	35.586	31.045	0.596
Rolling mean	332.537	18.236	15.459	53.737	38.683	0.705

Metrics are computed on the paper-comparable scale (RV_t/M_t, in bp^2 per 5-min interval, non-annualized). Deep learning results are computed from the seed-averaged forecast over 20 independent runs, whereas the econometric and machine-learning baselines are deterministic and are reported as single runs. The last two rows are deterministic benchmark rules—the random-walk rule $\hat{y}_{t+h|t} = y_t$ and the 21-day rolling mean of the realized volatility observed up to the origin—evaluated on exactly the same forecast origins as the other rows and are not averaged quantities; the equations and the estimation protocol of the seven benchmark models are given in Section 6.3. Best value per column is shown in bold.

Table 7. One-step prediction results with market sentiment factors (full feature set). Computed under protocol P1 (Section 6.1). Scale convention of Section 5.4: MAPE and SMAPE are percentages; RMSPE is a dimensionless proportion.

Model	MSE (bp^2)	RMSE (bp^2)	MAE (bp^2)	MAPE (%)	SMAPE (%)	RMSPE (prop.)
GARCH	315.636	17.766	14.981	42.126	53.018	0.478
HAR-RV	227.481	15.082	12.920	47.851	34.184	0.701
LSTM	119.041	10.911	8.925	30.424	26.047	0.449
SVM	750.973	27.404	25.311	91.414	55.725	1.195
XGBoost	1720.814	41.483	38.026	130.092	70.928	1.573
GRU	119.992	10.954	8.730	30.943	25.398	0.477
Transformer	126.732	11.258	8.610	28.047	24.650	0.420
PSO-Transformer	**112.024**	**10.584**	**8.038**	**26.758**	**23.525**	**0.415**
Persistence	188.355	13.724	10.586	35.586	31.045	0.596
Rolling mean	332.537	18.236	15.459	53.737	38.683	0.705

All metrics are computed on the paper-comparable scale (RV_t/M_t, in bp^2 per 5-min interval, non-annualized); the deep-learning values are computed from the 20-run seed-averaged forecast, the deterministic baselines are single runs, the run-to-run means and standard deviations of the deep-learning MAE are reported in Supplementary Table S16, and the last two rows are the deterministic random-walk rule $\hat{y}_{t+h|t} = y_t$ and the 21-day rolling mean of the realized volatility observed up to the origin, and the equations and the estimation protocol of the benchmark models are given in Section 6.3. Best value per column is shown in bold.

6.4. One-Step Forecasting with Multimodal Sentiment Factors

Building on the baseline established in Section 6.3, this subsection presents one-step prediction results with the full feature set, including the sentiment embeddings extracted by the TFN from videos (Section 3.1) and the sentiment scores from Qwen-2.5 analysis of news headlines (Section 3.2). The two econometric baselines, GARCH and HAR-RV, are univariate recursions on the realized-volatility series and contain no slot for a sentiment regressor, so their rows in Table 7 are numerically identical to their rows in Table 6; the

comparison they support is between a univariate volatility recursion and the complete fusion system, not between two models that consume the same information set. Every value below is measured under protocol P1 of Section 6.1, so the ordering it reports is conditional on the tuning window; the leak-free protocol P2 does not place the proposed system first, its seed-averaged one-step MAE being 10.16 against 8.82 for the GRU on the same days; and that comparison is given in Supplementary Sections S5.19 and S5.24.
Extended discussion of this subsection, with the further detail and the intermediate quantities it carries, is given in Supplementary Section S5.22.

6.5. Ablation Study

To quantify the contribution of each system component, we conduct systematic ablation experiments by removing one component at a time from the full model. Three variants are constructed: (1) *w/o Video Sentiment*, in which the TFN-extracted video sentiment factor is excluded from the data fusion system, retaining the text-based sentiment signal and the conventional financial indicators; (2) *w/o Public Sentiment*, in which the news-headline sentiment scores produced by Qwen-2.5 are excluded, retaining the video sentiment factor and the conventional indicators; and (3) *w/o PSO*, in which the PSO optimization engine is replaced by standard Transformer training under default hyperparameters (model dimension 64, epochs 50, batch size 128). The ablation rows are measured under protocol P1 of Section 6.1, so the increments they report are window-specific.

Table 8 reports the ablation results. Removing the video sentiment factor raises the one-step MAE slightly, from 8.038 to 8.119 (+1.0%), so that in this sample, the video signal is close to redundant with the price and technical blocks. Removing the public (news) sentiment factor leaves the MAE essentially unchanged (8.036, −0.03%), and removing PSO raises the MAE by 8.0% (from 8.038 to 8.680).
Extended discussion of this subsection, with the further detail and the intermediate quantities it carries, is given in Supplementary Section S5.23.

Table 8. Ablation study: one-step prediction performance (with sentiment factors). Computed under protocol P1 (Section 6.1). Scale convention of Section 5.4: MAPE and SMAPE are percentages; RMSPE is a dimensionless proportion.

Configuration	MSE (bp^2)	RMSE (bp^2)	MAE (bp^2)	MAPE (%)	SMAPE (%)	RMSPE (prop.)
Full Model	**112.024**	**10.584**	8.038	26.758	23.525	0.415
w/o Video Sentiment	114.210	10.687	8.119	26.828	23.637	**0.410**
w/o Public Sentiment	113.299	10.644	**8.036**	**26.724**	**23.482**	0.417
w/o PSO (default hyperparameters)	127.940	11.311	8.680	28.250	24.840	0.421

The Full Model row is the PSO-Transformer configuration of Table 7; the three variant rows are obtained by re-running the Random Forest selection and the PSO-Transformer on the reduced feature pool. Best value per column is shown in bold.

6.6. Comparison with Other Optimization Methods

To evaluate whether PSO provides advantages over alternative hyperparameter optimization strategies within the adaptive learning engine, we compare the PSO-Transformer with grid search and random search. All three search the same box—the model dimension in $[40, 128]$, the epochs in $[18, 80]$ and the batch size in $[64, 256]$—and all three are scored by the identical fitness function, the validation-set MAE of a model trained on the same training segment and evaluated on the same validation window, so the comparison isolates the search rule and not the data or the protocol. Grid search is *exhaustive* over a pre-registered lattice of $5 \times 5 \times 4 = 100$ points—the model-dimension levels $\{40, 60, 80, 100, 128\}$, the epoch levels $\{18, 34, 50, 66, 80\}$ and the batch-size levels $\{64, 128, 192, 256\}$—random search draws

600 configurations uniformly from the box, and PSO evaluates 30 particles over 20 iterations, again, 600 configurations. The stated ranges contain $89 \times 63 \times 193 = 1{,}082{,}151$ integer configurations, so "exhaustive" is meant over the 100-point lattice and not over every integer point of the box, and the grid is in fact the *cheapest* of the three searches here, at 100 evaluations against the 600 of each stochastic searcher. None of the three stops early: grid and random search exhaust their scheduled candidate list, and PSO runs its full 20 generations, retaining the best-so-far particle whose best value was first reached at generation 12. The evaluation counts, and the measured wall-clock times of each method are reported under Table 9. First, the fitness is the validation-set MAE, so each row of Table 9 is a value of the selection criterion itself: the configuration named in a row is the one that minimized the number reported beside it, re-measured under the paper's seed-averaging convention, and the table therefore ranks the three search rules *on their own objective* rather than three configurations on an external yardstick.

Extended discussion of this subsection, with the further detail and the intermediate quantities it carries, is given in Supplementary Section S5.24.

Table 9. Comparison of optimization methods: one-step prediction performance. Scale convention of Section 5.4: MAPE and SMAPE are percentages; RMSPE is a dimensionless proportion.

Optimization Method	Model Dimension	Epochs	Batch Size	MSE (bp^2)	RMSE (bp^2)	MAE (bp^2)	MAPE (%)
Grid Search	100	80	64	**103.758**	**10.186**	**7.900**	**26.603**
Random Search	116	70	125	112.210	10.593	8.059	26.763
PSO	116	70	85	114.509	10.701	8.120	26.929

Grid search evaluates a pre-registered $5 \times 5 \times 4$ lattice of 100 points—the model-dimension levels 40/60/80/100/128, the epoch levels 18/34/50/66/80 and the batch-size levels 64/128/192/256, which span the endpoints of the search box—i.e., 100 model evaluations; random search draws 600 configurations uniformly from the same box; PSO uses 30 particles over 20 iterations, i.e., 600 evaluations. None of the three terminates early: grid and random search exhaust their scheduled candidate list, and PSO runs its full 20 generations and returns the best-so-far particle, whose best value was first reached at generation 12 and never improved afterwards (Section 6.2). All three optimize the identical fitness, the validation-set MAE of a model trained on the same training segment and scored on the same validation window; all three report the validation-MAE-optimal configuration; and the metrics are then recomputed on that same window from the 20-run seed-averaged forecast. Because the fitness window is the evaluation window under protocol P1, the three rows are values of the search criterion rather than errors on data held out from the search; the leak-free counterpart of this comparison exists for the PSO search only and is reported in Supplementary Section S5.24. The three searches were timed in a single session on one NVIDIA RTX 4090 GPU: the 100 grid points took 50 s, the 600 random draws 270 s and the 600 PSO evaluations 369 s—0.50, 0.45 and 0.62 s per evaluation, respectively—so the two 600-evaluation searches cost about five and seven times the grid. Because every method calls the same fitness function, the wall-clock of a method here is governed by how many evaluations it performs, and the 100-point grid is the cheapest of the three, while the two 600-evaluation searches are the most expensive. The grid is therefore not the costly method at this budget; its cost grows multiplicatively in the number of hyperparameters, which is the sense in which it becomes prohibitive in a larger space, whereas the 600 evaluations of PSO are independent of the dimensionality of the search.

6.7. Statistical Significance Testing

To verify that the observed differences in predictive accuracy are statistically significant rather than attributable to random variation, we conduct the Diebold–Mariano (DM) test [41], which compares the predictive accuracy of two competing models by evaluating whether the mean difference in their forecast loss series is significantly different from zero. Let $L(\hat{y}_{A,t}, y_t)$ and $L(\hat{y}_{B,t}, y_t)$ denote the loss functions (squared error loss in this study) of

models A and B at time t, and define the loss differential as $d_t = L(\hat{y}_{A,t}, y_t) - L(\hat{y}_{B,t}, y_t)$. The DM test statistic is

$$DM = \frac{\bar{d}}{\sqrt{\hat{V}(\bar{d})}}, \tag{17}$$

where $\bar{d} = \frac{1}{T}\sum_{t=1}^{T} d_t$ is the sample mean of the loss differentials, and $\hat{V}(\bar{d})$ is a consistent estimator of their long-run variance, computed with a heteroskedasticity-and-autocorrelation-consistent (HAC) estimator in order to allow for serial correlation in the daily loss differentials; because the reported tests are one-step, no overlapping multi-step losses enter the series. The differentials are drawn from the evaluation window of protocol P1 (Section 6.1), which is also the window on which the hyperparameters and the best epoch are selected, so what the tests establish is that the differences on those days are large relative to their own sampling variation, not that they would recur on a disjoint sample. We use the Newey–West estimator with a Bartlett kernel,

$$\hat{V}(\bar{d}) = \frac{1}{T}\Big[\hat{\gamma}_0 + 2\sum_{k=1}^{L}\Big(1 - \frac{k}{L+1}\Big)\hat{\gamma}_k\Big], \qquad \hat{\gamma}_k = \frac{1}{T}\sum_{t=k+1}^{T}(d_t - \bar{d})(d_{t-k} - \bar{d}), \tag{18}$$

where $\hat{\gamma}_k$ is the sample autocovariance of the loss differentials at lag k, and L is the Bartlett bandwidth.

Protocol of the test. Six aspects of the implementation are stated explicitly, because they determine the degrees of freedom that the test actually has. Three of them are stated here—the length of the series, the role of the twenty training repetitions and the overlap of the forecasts—and the other three, namely, what t indexes, the HAC kernel and bandwidth and the treatment of multiple comparisons, are given in Supplementary Section S5.25. *(i) Length of the series.* Each test uses $T = 21$ loss differentials, one for every day of the out-of-sample window: because the sliding window advances by one trading day, each day contributes exactly one forecast origin and one non-overlapping forecast, so T equals the number of *trading days* in the test window—not the number of generated sequences (104) nor the number of 5-min observations from which the daily realized volatility is aggregated. *(iii) Role of the twenty training repetitions.* Repeating the network training twenty times does not create twenty independent histories of the market, and the repetitions are *not* treated as twenty loss-differential series: every row of Table 10 is computed from a single series of $T = 21$ differentials, in which the forecast of the proposed system is its twenty-run seed-averaged forecast, the repetitions reducing the simulation noise of that one forecast and entering the test in no other way. *(v) Overlapping forecasts.* All entries of Table 10 are one-step-ahead tests, so consecutive loss differentials are not mechanically overlapping, and no multi-step correction is needed.

Extended discussion of this subsection, with the further detail and the intermediate quantities it carries, is given in Supplementary Section S5.25.

6.8. Multi-Step Forecasting with Multimodal Sentiment Factors

The one-step results demonstrate short-term prediction capability. To evaluate system performance across error accumulation, trend continuation and structural stability, two-step and five-step out-of-sample prediction experiments are conducted. Both horizons are evaluated under protocol P1 of Section 6.1, with the validation sample equal to the evaluation window, and they therefore carry the same qualification as the one-step results.

Table 10. Diebold–Mariano test results: each competing model (model A) vs. the PSO-Transformer (model B), one-step ahead, squared error loss. The differentials are computed on the evaluation window of protocol P1 (Section 6.1); the rows are named in the order of the DM definition, so the model on the left of a row is model A, and the PSO-Transformer on the right is model B; $d_t = e^2_{A,t} - e^2_{B,t}$ is therefore the competitor's squared error minus the proposed system's; a *positive* statistic means that the competing model named first is the less accurate of the two, and a negative statistic would favor the competitor.

Comparison (Model ***A*** vs. Model ***B***)	DM Statistic	***p*-Value**	Holm ***p***	Significant at 5%
XGBoost vs. PSO-Transformer	5.61	<0.001	<0.001	Yes
SVM vs. PSO-Transformer	4.90	<0.001	<0.001	Yes
GARCH vs. PSO-Transformer	3.18	0.001	0.007	Yes
HAR-RV vs. PSO-Transformer	2.06	0.040	0.159	No
Transformer vs. PSO-Transformer	1.38	0.169	0.507	No
LSTM vs. PSO-Transformer	0.70	0.487	0.974	No
GRU vs. PSO-Transformer	0.53	0.596	0.974	No

$T = 21$ one-step loss differentials per row, one per trading day of the held-out window; the twenty training repetitions enter only through the seed-averaged forecast of the proposed system, so they do not multiply the series; Newey–West Bartlett variance with bandwidth $L = 0$ for the reported one-step tests, the $L = 2$ sensitivity being recorded in Section 6.7; two-sided critical value ± 1.96; the last column gives Holm step-down adjusted p-values over the seven comparisons. The GARCH and HAR-RV rows are the tabulated recursions of Tables 6 and 7 (one-step MAE 14.981 and 12.920), not the recursions re-estimated inside the HARX/GARCH-X exercise of Section 6.3, whose errors and statistics are reported in Supplementary Tables S23 and S24. Section 6.7 states the protocol in full.

Table 11. Two-step prediction performance with full features. Computed under protocol P1 (Section 6.1). Scale convention of Section 5.4: MAPE and SMAPE are percentages; RMSPE is a dimensionless proportion.

Model	MSE (bp^2)	RMSE (bp^2)	MAE (bp^2)	MAPE (%)	SMAPE (%)	RMSPE (prop.)
GARCH	386.254	19.653	15.238	39.880	52.295	0.493
HAR-RV	283.496	16.837	14.886	54.639	38.753	0.755
LSTM	123.770	11.125	9.365	31.401	27.292	0.443
SVM	742.633	27.251	25.102	90.883	55.403	1.191
XGBoost	2284.832	47.800	40.479	144.394	71.133	1.869
GRU	123.510	11.113	9.017	31.304	26.234	0.463
Transformer	122.064	11.048	9.261	31.193	27.032	0.452
PSO-Transformer	**117.051**	**10.819**	**8.791**	**29.465**	**25.658**	**0.440**
Persistence	299.724	17.313	15.655	50.611	44.998	0.675
Rolling mean	366.062	19.133	16.449	56.623	40.499	0.724

Every row is a direct two-step forecast of the protocol described in the text—one model per horizon, the one-step PSO configuration reused, the seed-averaged forecast of the twenty runs—evaluated on the 21 held-out days during 30 May–27 June 2025. Best value per column is shown in bold.

The ablation results in Tables 13 and 14 show that the component contributions are horizon-dependent. At the two-step horizon, removing the PSO engine raises the MAE by 5.6% (8.791 to 9.287), while removing the video and the news sentiment factors changes it by +0.5% and −0.1%, respectively, mirroring the one-step picture. At the five-step horizon, the PSO engine matters most (removing it raises the MAE by 11.5%, from 9.140 to 10.189), and the sentiment inputs become materially useful for the first time: removing the news factor costs 6.3% and removing the video factor 7.4%.

Extended discussion of this subsection, with the further detail and the intermediate quantities it carries, is given in Supplementary Section S5.26.

Table 12. Five-step prediction performance with full features. Computed under protocol P1 (Section 6.1). Scale convention of Section 5.4: MAPE and SMAPE are percentages; RMSPE is a dimensionless proportion.

Model	MSE (bp^2)	RMSE (bp^2)	MAE (bp^2)	MAPE (%)	SMAPE (%)	RMSPE (prop.)
GARCH	369.981	19.235	16.273	43.517	55.316	0.479
HAR-RV	290.660	17.049	13.635	52.585	35.636	0.771
LSTM	133.881	11.571	9.480	32.536	27.531	0.479
SVM	670.406	25.892	23.542	87.014	53.234	1.166
XGBoost	2450.794	49.505	43.171	156.845	74.885	2.057
GRU	129.972	11.401	9.229	32.948	26.807	0.500
Transformer	145.994	12.083	10.188	32.230	30.228	**0.428**
PSO-Transformer	**126.836**	**11.262**	**9.140**	**31.106**	**26.676**	0.467
Persistence	260.179	16.130	12.919	43.193	36.089	0.624
Rolling mean	461.481	21.482	18.410	65.302	43.945	0.844

Every row is a direct five-step forecast of the protocol described in the text, evaluated on the 20 held-out days 2–27 June 2025; this window is one day shorter than the one- and two-step windows, because the sequence constructor consumes the last four usable windows at this horizon. Best value per column is shown in bold, so that the horizon at which each model is competitive is visible directly.

6.9. Case Study: Revisiting the FOMC Running Example

To close the loop of the running example introduced in Section 1, we examine the March 2025 FOMC episode. The announcement day is 19 March 2025: the FOMC statement is released at 14:00 New York time, i.e., 02:00 Beijing time on 20 March, whereas the last input of the model is the 23:55 Beijing bar of 19 March, so the prediction for 20 March is a strictly pre-announcement forecast, and its error measures how much of the policy shock the pre-event information set already contained. Using these inputs, the PSO-Transformer predicted a next-day realized volatility of 26.52, whereas the variant without sentiment factors predicted 27.19; all values quoted here and in Table 15 are on the scale of Equation (2), that is, bp^2 per 5-min interval without annualization.

Extended discussion of this subsection, with the further detail and the intermediate quantities it carries, is given in Supplementary Section S5.27.

Table 13. Ablation study: two-step prediction performance. Computed under protocol P1 (Section 6.1). Scale convention of Section 5.4: MAPE and SMAPE are percentages; RMSPE is a dimensionless proportion.

Configuration	MSE (bp^2)	RMSE (bp^2)	MAE (bp^2)	MAPE (%)	SMAPE (%)	RMSPE (prop.)
Full Model	117.051	10.819	8.791	29.465	**25.658**	0.440
w/o Video Sentiment	**109.255**	**10.453**	8.838	**28.827**	25.870	**0.391**
w/o Public Sentiment	117.005	10.817	**8.778**	28.925	25.669	0.417
w/o PSO (default hyperparameters)	122.534	11.070	9.287	31.207	27.114	0.448

The Full Model row is the PSO-Transformer configuration of Table 11. All rows use the direct two-step protocol and the 21-day held-out window of that table. Best value per column is shown in bold.

Table 14. Ablation study: five-step prediction performance. Computed under protocol P1 (Section 6.1). Scale convention of Section 5.4: MAPE and SMAPE are percentages; RMSPE is a dimensionless proportion.

Configuration	MSE (bp^2)	RMSE (bp^2)	MAE (bp^2)	MAPE (%)	SMAPE (%)	RMSPE (prop.)
Full Model	**126.836**	**11.262**	**9.140**	**31.106**	**26.676**	0.467
w/o Video Sentiment	139.195	11.798	9.812	31.741	28.959	**0.428**
w/o Public Sentiment	138.781	11.781	9.719	31.923	28.542	0.441
w/o PSO (default hyperparameters)	146.789	12.116	10.189	32.217	30.205	0.429

The Full Model row is the PSO-Transformer configuration of Table 12. All rows use the direct five-step protocol and the 20-day held-out window of that table. Best value per column is shown in bold.

6.10. Scheduled High-Impact Events: A Multi-Event Summary

The running example of Section 6.9 rests on a single date, and one date cannot carry a claim about event days. To see whether the behavior observed on 19 March 2025 is typical of scheduled events rather than of that particular afternoon, an event set is fixed from public calendars before any outcome is read, and the system is evaluated on every member of it that the sample contains. Three families of releases with a documented effect on gold and a fixed publication time are used: the FOMC decisions of the Federal Reserve (statement at 14:00 New York time), the CPI releases of the Bureau of Labor Statistics and its Employment Situation reports, the last two at 08:30 New York time.
Extended discussion of this subsection, with the further detail and the intermediate quantities it carries, is given in Supplementary Section S5.28.

7. Risk Warning Decision System

This section describes the risk warning decision system, which constitutes the output layer of the integrated prediction and decision support system. Its core function lies in translating quantitative volatility forecasts into structured, actionable risk alerts for market participants. In the backtest of Section 7.4, the alert rule is anchored to a persistence volatility model rather than to the neural engine, so the tier accuracy reported there characterizes the alert layer with that anchor, and the PSO-Transformer's own walk-forward warning result is weaker (Supplementary Table S22). The system comprises three functional modules (Figure 9):

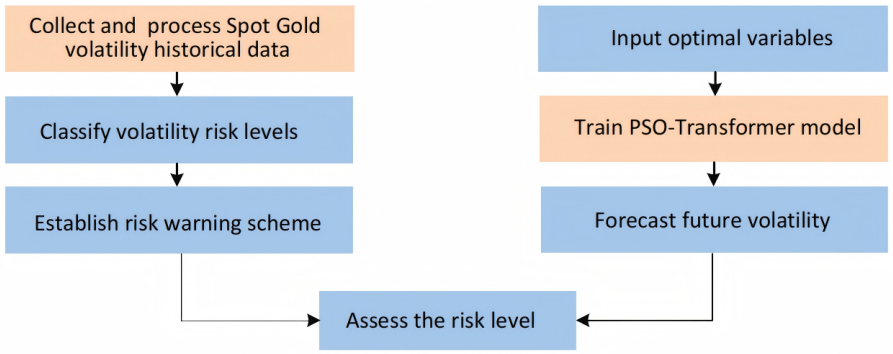


Figure 9. Architecture of the risk warning decision system. The system receives predicted realized volatility values from the adaptive learning engine, calibrates thresholds based on historical quantiles, classifies market conditions into three risk tiers and issues the corresponding risk-management guidance. The volatility model that supplies the prediction is a design choice: the backtest of Section 7.4 anchors the rule to a persistence model, and its tier accuracy therefore measures the alert layer rather than the neural engine (Supplementary Table S22). The tiers describe the dispersion of predicted realized volatility, not its direction.

Extended discussion of this section, with the further detail and the intermediate quantities it carries, is given in Supplementary Section S5.29.

- **Threshold Calibration Module:** Determines risk boundaries based on realized volatility quantiles and distributional properties.
- **Alert Classification Module:** Maps predicted volatility values to discrete risk tiers using calibrated thresholds.
- **Decision Support Protocol Module:** Translates alert levels into risk-management responses for market participants in general, without any recommendation to alter the direction of exposure.

7.1. Threshold Calibration Module

The thresholds are set to the median and the third quartile of the empirical distribution of the per-interval realized variance RV_t/M_t of Section 3.3. Over the 126 trading days of the sample, these quantiles are 30.571 and 51.391, and over the 105-day calibration sub-sample that precedes the out-of-sample window, they are 29.797 and 55.398, all four numbers being in bp^2 per 5-min interval on the non-annualized scale defined in Section 3.3. The two full-sample values describe the tier geometry for exposition; every out-of-sample result reported in this paper uses thresholds estimated without the test window, namely, the calibration-sub-sample values 29.797 and 55.398 or, in the walk-forward backtest, thresholds recomputed causally at each origin. Everything evaluated below is therefore rolling: the thresholds are recomputed at every forecast origin from the data available at that origin, and this expanding-window procedure—not the full-sample quantiles quoted above, which serve only to describe the geometry of the tiers—is the primary backtesting result of this paper, reported with its own confusion matrix and its own class-specific metrics.
Extended discussion of this subsection, with the further detail and the intermediate quantities it carries, is given in Supplementary Section S5.30.

Table 15. FOMC case study: input factors, model predictions and risk-tier classification outcomes. All realized-variance values are in bp^2 per 5-min interval, on the non-annualized scale of Section 3.3: the 18.34 and 18.60 entries are the observed values and 26.52 / 27.19 the model predictions. Every value of the table is reproduced in the date-level record `forecasts/fomc_case_study.csv` of the reproduction package, which also lists the cut-off, the input block, the seeds and the configuration of the refit. Bold entries mark the tier labels and the match indicators, and the green checkmark marks a case in which the predicted tier coincides with the observed tier.

Item	Value	Source
FOMC announcement day	19 March 2025	public FOMC calendar; Section 6.9
Video sentiment factor (V1, cut-off-day snapshot)	0.001 (no-signal sentinel)	Section 3.1
Public sentiment score (V2, cut-off-day snapshot)	4 (extremely bullish)	Section 3.2
Training windows available before the event	33 (targets 3 February–19 March 2025)	`forecasts/fomc_case_study.csv`
Observed RV on the announcement day (paper scale)	18.34	Section 3.3 (Equation (2))
Observed next-day RV (paper scale)	18.60	Section 3.3 (Equation (2))
PSO-Transformer predicted next-day RV (sentiment-augmented)	26.52	`forecasts/fomc_case_study.csv`
No-sentiment predicted next-day RV (sentiment-free)	27.19	`forecasts/fomc_case_study.csv`
Observed tier	**Green** (<30.571)	Section 7.2 (Table 16)
Full-system predicted tier	**Green** (<30.571) ✓ Match	Section 7.2 (Table 16)
No-sentiment predicted tier	**Green** (<30.571) ✓ Match	Section 7.2 (Table 16)
Absolute error (full system)	7.92	|18.60–26.52|
Absolute error (no sentiment)	8.59	|18.60–27.19|
Error reduction	7.8%	(8.59–7.92)/8.59

7.2. Alert Classification Module

Accordingly, levels of the per-interval realized variance RV_t/M_t (in bp^2 per 5-min interval, non-annualized) are categorized into three risk tiers, as shown in Table 16. (1) When RV_t/M_t falls within $[0, 30.571)$, the day is classified as a low-risk zone (green), indicating stable market conditions. (2) When it falls within $[30.571, 51.391)$, it is defined as a normal risk zone (yellow), in which market fluctuations remain acceptable but require continued monitoring. (3) When it reaches or exceeds 51.391, entering $[51.391, +\infty)$, it is identified as a high-risk warning zone (red), which activates the risk control mechanisms of Section 7.3.

Table 16. Risk warning level classification based on the per-interval realized variance RV_t/M_t. The threshold values are in bp^2 per 5-min interval and are not annualized. The last column illustrates the risk-management response that a volatility state of this kind would call for; it is not a trading recommendation. The tiers describe the dispersion of next-day realized volatility and say nothing about its direction, so a high-volatility state is compatible with either a positive or a negative expected return, no tier recommends an increase in exposure, and no return or portfolio model is estimated in this paper.

Early Warning Color	Range of RV_t/M_t (bp^2 per 5-min)	Risk Level	Risk-Management Response (Illustrative)
Green	$[0, 30.571)$	Low Risk	Standard monitoring; the volatility state alone indicates no change to risk limits
Yellow	$[30.571, 51.391)$	Normal Risk	Review hedging or stop-loss coverage; keep exposures inside the risk limits already set
Red	$[51.391, +\infty)$	High Risk	Consider trimming exposure or hedging in order to lower volatility risk; re-check margin and limits

From point thresholds to a distributional alert rule. A direct implementation of the three tiers thresholds the point forecast: day t is labeled green when $\hat{y}_t < L_t$, red when $\hat{y}_t \geq U_t$ and yellow otherwise. Such a rule is only as informative as the dispersion of $\hat{y}_t$. A forecast that is optimal in mean squared error is deliberately compressed towards the conditional mean, so it under-populates the two extreme tiers even when the ranking of the days is correct, and Section 7.4 quantifies this for the PSO-Transformer and shows that the point rule never raises a red alert. We therefore define the alert as the Bayes-optimal decision over the predictive *distribution* of next-day realized volatility rather than over a single number. Let F_t denote the cumulative predictive distribution of y_t formed from the information available at the close of day $t-1$, and let L_t and U_t be the lower and upper thresholds in force at origin t. The three tiers then carry the probabilities

$$p_t^{\text{lo}} = F_t(L_t), \qquad p_t^{\text{mid}} = F_t(U_t) - F_t(L_t), \qquad p_t^{\text{hi}} = 1 - F_t(U_t), \tag{19}$$

and the alert issued for day t is the maximum a posteriori tier

$$a_t = \underset{k \in \{\text{lo, mid, hi}\}}{\arg\max} \; p_t^k, \tag{20}$$

which minimizes the probability of a misclassification under any loss that is symmetric across tiers. The rule uses only the two thresholds defined above and a predictive distribution: it introduces no free tuning parameter, and the point-threshold rule is recovered as the degenerate case in which F_t is a point mass at the forecast. In the empirical work below, F_t is built by pairing a one-step volatility model $\hat{y}_t$ with the empirical distribution of its past one-step residuals $\{y_s - \hat{y}_s\}_{s<t}$; Supplementary Section S5.31 states the split-conformal construction and why it calibrates the tier probabilities of Equation (19) by construction

rather than by assertion, and Supplementary Table S22 reports the alerting performance of this rule together with that of the point-threshold rule and of four alternative one-step volatility models. In the backtest reported in this paper, the model that supplies F_t is the persistence model $\hat{y}_t = y_{t-1}$, not the PSO-Transformer; Section 7.4 compares the two. *Extended discussion of this subsection, with the further detail and the intermediate quantities it carries, is given in Supplementary Section S5.31.*

7.3. Decision Support Protocol Module

The decision support protocol module provides tier-specific responses to the classified risk level.

Scope of these recommendations. The tiers produced by the two modules above concern the dispersion of next-day realized volatility. They are not forecasts of the direction of the gold price, and a higher predicted volatility does not imply a higher expected return: a turbulent day can close either way. The protocol below is therefore confined to risk-management responses—monitoring, hedging coverage and exposure limits—and it contains no recommendation to increase a position at any tier, because such a recommendation would require a return or portfolio model, which this paper does not estimate. *Extended discussion of this subsection, with the further detail and the intermediate quantities it carries, is given in Supplementary Section S5.32.*

- **Red zone (high risk):** When the system enters the red zone, the protocol calls for risk-averse participants to consider risk-reducing steps—trimming exposure or hedging—with the object of lowering volatility risk rather than of improving expected return.
- **Yellow and green zones (normal and low risk):** In the yellow and green zones the probability of a turbulent day remains comparatively low. Participants keep to a normal monitoring cadence and stay inside the risk limits they have already set. A green or yellow alert is not a signal to add risk: it reports that the volatility state is unremarkable, which is silent about the direction of the price.

7.4. Backtesting Evaluation of the Risk Warning System

To empirically assess the reliability of the risk warning decision system, we adopt a walk-forward, or rolling-origin, backtest. At each forecast origin t, the alert for day t is formed from information dated $t-1$ and earlier only: the one-step volatility model that anchors the alert rule—the persistence model, whose split-conformal predictive distribution is the object that Equations (19) and (20) threshold—is re-estimated on $\{y_1, \ldots, y_{t-1}\}$, its conformal residual sample is accumulated over the same window, the two thresholds are recomputed as the median and the third quartile of $\{y_1, \ldots, y_{t-1}\}$, and Equations (19) and (20) are applied. The procedure is restarted from scratch at every origin, so the evaluation is genuinely out of sample and, unlike a single fixed split, uses the whole sample as test data. The first 45 forecast origins (3 February–4 April 2025) are the burn-in that builds the residual sample, and the remaining 59 (7 April–27 June 2025) form the evaluation window, which contains the 21-day held-out window of the point-forecast comparison.

Which model anchors the alert and what the alternative gives. The alert rule is anchored to the persistence model by design, and the configuration reported in Tables 17 and 18 is that anchor. Supplementary Table S22 compares it with the three other volatility models that can supply the predictive distribution, and the persistence anchor attains the highest tier accuracy of the four (0.678, against 0.576 for HAR-RV, 0.610 for quantile regression and 0.455 for the network). The recursively re-estimated PSO-Transformer is that fourth candidate, and the loop it closes is reported here rather than left to the supplement: its split-conformal residual sample exists only from the 45th origin onward, so the rule can be evaluated on 44 of the 59 origins, and on those origins, it attains an accuracy of 0.455 against

the 0.614 that the persistence anchor reaches on the same 44 origins. The accuracy reported in this subsection is therefore a property of the alert layer with a persistence anchor and not of the neural engine, and the two are not interchangeable: the point-forecast tables of Sections 6.4–6.8 measure the engine, while the alert tables of this subsection measure the rule. The engine is not uniformly worse at this task: its red-tier probability orders the turbulent days better than the persistence anchor does (AUC 0.929 against 0.915), and its lower tier accuracy comes from raising more red alerts than the tier geometry requires, spending precision while keeping the same recall. Supplementary Section S5.33 states the comparison in full.

Extended discussion of this subsection, with the further detail and the intermediate quantities it carries, is given in Supplementary Section S5.33.

8. Results and Discussion

This section synthesizes the empirical findings at both the system and component levels, interprets the contributions of each subsystem, compares results with existing literature and discusses practical implications and limitations. Unless a passage states otherwise, every accuracy figure quoted in Sections 8 and 9 is a protocol P1 value in the sense of Section 6.1, measured on the 21 days that also served as the validation sample, and hence, a within-tuning-window result rather than a fully independent out-of-sample one (protocol P2 counterpart: Supplementary Sections S5.19 and S5.24).

Table 17. Confusion matrix of the walk-forward risk alerts (59 out-of-sample forecast origins, 7 April–27 June 2025). The alert is the distributional rule of Equations (19) and (20) anchored to the persistence model and not to the PSO-Transformer; Supplementary Table S22 compares the two. Rows are predicted tiers and columns are realized tiers.

Predicted\Actual	Green (Low)	Yellow (Normal)	Red (High)	Total Predicted
Green (Low)	8	9	1	18
Yellow (Normal)	1	6	1	8
Red (High)	2	5	26	33
Total Actual	**11**	**20**	**28**	**59**

The alert rule is Equation (20) applied to the conformal persistence predictive distribution; the thresholds are the expanding median and third quartile of the realized volatility observed up to the forecast origin. Bold entries mark the row and column totals.

Table 18. Per-tier evaluation metrics for the walk-forward risk alerts (59 forecast origins), computed with the alert anchored to the persistence model; the PSO-Transformer's own walk-forward alerts are compared with them in Supplementary Table S22. Precision, recall and F1 are proportions in $[0, 1]$ and not percentages, and the overall accuracy is reported on the same scale.

Risk Tier	Precision	Recall	F1 Score
Green (Low)	0.444	0.727	0.552
Yellow (Normal)	0.750	0.300	0.429
Red (High)	0.788	0.929	0.852
Overall Accuracy	**0.678**	(balanced accuracy 0.652; macro-F1 0.611; 40 of 59 days correct)	

The red-tier probability p_t^{hi} of the persistence anchor attains AUC = 0.915 and a Brier score of 0.126, the pairing reported for that model in Supplementary Section S5.33. The recursively re-estimated network of Supplementary Table S22 ranks the red tier better (AUC = 0.929 and a Brier score of 0.118 on the 44 origins at which its residual sample exists) but attains the lowest red-tier F1 of the four volatility models (0.684). Interval coverage: 49.2% of the realized values fall inside the 50% predictive interval and 79.7% inside the 80% interval. The degenerate rule that always issues a yellow alert attains an overall accuracy of 0.339. Bold entries mark the overall-accuracy row.

8.1. Summary of Key Findings

The empirical results yield four principal findings. First, the multi-source data fusion system's sentiment modules carry measurable but modest explanatory power for gold volatility: the Random Forest importance analysis (Table 3) ranks the Public Sentiment factor (0.011501) 16th and the Video Sentiment factor (0.008169) 20th among the 21 candidate columns, and both clear the retention threshold, yet the price-, commodity- and technical-indicator blocks dominate, which is consistent with the sparse publication of sentiment-bearing items in this sample.

Second, within this sample, the deep-learning family holds the lower error at every horizon: across the one-, two- and five-step tables, the sequence models sit below both econometric baselines and below the two flattened-input learners on the reported metrics, most narrowly at the five-step horizon (Sections 6.4 and 6.8). That ordering is a property of the window studied rather than a general statement about deep architectures in financial time series, and the five-step margins in particular fall inside the run-to-run variation.

Third, under protocol P1—in which the hyperparameters and the best epoch are selected on the same 21 days that are then evaluated—the proposed system achieves the best value on all six error metrics at the one-step and two-step horizons, and on five of the six at the five-step horizon. In one-step prediction, it attains MAE = 8.038, an average accuracy improvement of 36.5% over the seven benchmark models, and the Diebold–Mariano test (Section 6.7) confirms that the one-step gains over GARCH, SVM and XGBoost remain statistically significant at the 5% level after a Holm correction for the seven comparisons, whereas the differences from HAR-RV, the LSTM, the GRU and the standard Transformer are not. This ordering is a property of the P1 design and not of a fully independent out-of-sample comparison: when the same pipeline is re-searched under the leak-free protocol P2, whose 20-day validation period is disjoint from the test window, its seed-averaged one-step MAE is 10.16 against 8.82 for the GRU, and the first place is not preserved (Supplementary Sections S5.19 and S5.24).

Fourth, the ablation studies (Tables 8, 13 and 14) attribute the largest contribution to the PSO optimization engine at every horizon (+8.0% on the one-step MAE when it is replaced by default hyperparameters, +5.6% at the two-step and +11.5% at the five-step horizon), whereas the sentiment inputs barely matter at short horizons (removing the video factor raises the one-step MAE by 1.0%, and removing the news factor leaves it essentially unchanged) and become useful only at the five-step horizon (+7.4% and +6.3% respectively). *Extended discussion of this subsection, with the further detail and the intermediate quantities it carries, is given in Supplementary Section S5.34.*

8.2. Why Do Video Sentiment Factors Matter?

The ablation results show that the incremental value of the video sentiment factor is horizon-dependent rather than absent: removing it raises the one-step MAE by 1.0%, the two-step MAE by 0.5% and the five-step MAE by 7.4%, so the channel becomes materially useful only where the model can no longer rely on the short-horizon persistence of the volatility process. The case study of Section 6.9 shows, in addition, how the sentiment block can sharpen the point forecast on one event day, and the multi-event summary of Section 6.10 shows that the behavior is not general to the scheduled releases of the sample: as a mechanism, the episode is informative; as a performance claim, it is not.
Extended discussion of this subsection, with the further detail and the intermediate quantities it carries, is given in Supplementary Section S5.35.

8.3. The Role of PSO Optimization

The comparison with grid search and random search (Table 9) shows that the three optimizers reach comparable accuracy under the same validation criterion and the same training protocol. PSO converges to model dimension 116, epochs 70 and batch size 85 and attains MAE = 8.120; grid search exhausts its pre-registered $5 \times 5 \times 4 = 100$-point lattice and converges to (100, 80, 64) with MAE = 7.900, and random search to (116, 70, 125) with MAE = 8.059. PSO is therefore 2.8% and 0.8% above the two competing searches in this particular comparison. Three qualifications matter. First, these three numbers are values of the selection criterion on the very window that also selects them (Section 6.6), not errors measured on data held out from the search: under protocol P1, the validation window is the evaluation window. Second, "exhaustive" is meant over that lattice of 100 points and not over every integer configuration of the box: the stated ranges contain $89 \times 63 \times 193 = 1{,}082{,}151$ of them, and both random search and PSO visit 600. Third, the comparison does not show PSO to be the cheaper search: the grid spends 100 evaluations while each stochastic searcher spends 600, and PSO still trails the grid by 2.8%, so what the swarm buys is dimensional scalability rather than a lower cost at this size. Fourth, as with the accuracy tables, this ordering does not survive a disjoint evaluation: when the search is repeated under protocol P2 the searched configuration lands at 10.16 against 8.82 for the GRU on the same disjoint window (Supplementary Sections S5.19 and S5.24), so the P1 ranking of the three searches is a within-window result in the same sense as the accuracy tables of Sections 6.3–6.8.

Extended discussion of this subsection, with the further detail and the intermediate quantities it carries, is given in Supplementary Section S5.36.

8.4. Comparison with Existing Literature

Our results are consistent with several strands of existing literature while offering extensions. The improved performance of deep learning models over GARCH and HAR-RV is consistent with the nonlinearity that this literature already anticipates: Bollerslev [9] introduced GARCH so that the conditional variance could respond to past shocks, and Nelson [10] and Glosten et al. [11] documented the asymmetric response that a symmetric linear specification misses; our framework extends beyond these studies by integrating multimodal sentiment factors through a structured data fusion system.

The role of sentiment factors is more nuanced in our results than in the earlier literature. Luo et al. [16] documented an association between sentiment indicators and oil and gold volatility once structural breaks are accounted for; in our sample, the news and video sentiment factors clear the Random Forest retention threshold but carry an importance one to two orders of magnitude below that of the realized-volatility and technical-indicator blocks, and removing them changes the average one-step error by at most five percent.

The effectiveness of PSO for hyperparameter search is consistent with the PSO–GRU findings of Tang et al. [28] for Bitcoin realized volatility and with the deep-network tuning of Lorenzo et al. [23], here in the Transformer architecture and on gold volatility; whether the parallel extends beyond this sample is not tested here. All three comparisons are drawn from the protocol P1 evaluation window of Section 6.1 and inherit its qualification: the orderings observed inside that window are not claimed to survive a disjoint test window, and under the leak-free protocol P2, the proposed system does not lead (Section 8.1).

Extended discussion of this subsection, with the further detail and the intermediate quantities it carries, is given in Supplementary Section S5.37.

8.5. System-Level Contributions

Beyond the individual component improvements discussed above, the integrated system delivers three system-level benefits that emerge from the coordination of its two core subsystems and the bridging adaptive learning engine:

- **End-to-end information flow:** The multi-source data fusion system constructs a rich, multimodal input representation that captures both structured market dynamics and unstructured sentiment signals. This input is processed by the PSO-Transformer adaptive learning engine, which extracts nonlinear temporal patterns, and the risk warning decision system then converts the numerical forecasts into graded alerts. The path from fused inputs to graded alerts is traced end to end by the ablation of Section 6.5, which measures what each stage contributes; the contributions are complementary but far from equal, as the next item records.
- **Complementarity of system components:** The ablation results indicate that the components contribute very unevenly at the horizon that matters for alerting. Removing PSO degrades the one-step performance by 8.0%, whereas removing either sentiment input leaves the MAE changed by at most one percent. The combined system achieves MAE = 8.038 under protocol P1 (Section 6.1), which is statistically indistinguishable from the news-ablated variant (8.036) and lower than the other two variants. The components therefore address different aspects of the problem in a complementary but highly unbalanced way: the data fusion system enriches the input space and materializes as a measurable gain only at the five-step horizon, not at the one-step horizon at which the alerts are raised (Section 6.5), while the adaptive optimization engine dominates the average error at every horizon.
- **Bridging prediction and decision-making:** Most volatility forecasting models produce point estimates or confidence intervals that require expert interpretation. By integrating a risk warning decision system with quantile-based risk grading and level-appropriate action protocols, the proposed system reduces this gap and provides interpretable outputs usable in risk-management protocols within the scope of Section 7.3; it is not a trading rule, because no expected-return model is estimated in this paper. The backtesting results (Section 7.4) show that the warning system—with the alert rule anchored to the persistence model, the configuration that Supplementary Table S22 identifies as the most accurate of the four volatility models that can supply the predictive distribution—attains 67.8% overall tier-classification accuracy (balanced accuracy 65.2%) over 59 walk-forward forecast origins, with a high-risk precision of 78.8% and a recall of 92.9%—a property of the alert layer with this anchor rather than of the neural engine, which fed into the same rule attains 45.5% against the anchor's 61.4% on the 44 origins where both can be scored (Supplementary Table S22)—and that its 50% and 80% predictive intervals cover the realized volatility 49.2% and 79.7% of the time. The design choice that carries the alert layer is not the placement of the thresholds, which are stated directly as the median and the 75th percentile of the realized-variance distribution, but the object that is thresholded: a mean-squared-error point forecast is compressed towards the conditional mean and therefore under-populates the extreme tiers, whereas a calibrated conditional predictive distribution supplies exactly the tier probabilities that the alert rule consumes. Driving the alert with a distributional layer rather than with a point forecast raises the high-risk F1 from zero to 0.852 on the same sample; the point-forecast rule it replaces is scored on the forecasts of protocol P1, and the 0.852 belongs to the distributional rule with the persistence anchor, the network's own forecast giving a red-tier F1 of 0.684 when it is passed through the same rule (Supplementary Table S22). This feature is uncommon in the academic literature on volatility forecasting and is the part of the design we would expect to carry over most

readily, since it requires no new data; it is nevertheless established here on a single six-month sample.

8.6. Practical Implications

From the perspective of application value, spot gold is a critical global safe-haven asset, and the proposed system, integrating multi-source data fusion, intelligent adaptive learning and structured risk warning, offers several practical contributions. The traffic-light alert scheme (Table 16) translates quantitative predictions into interpretable risk categories that a risk desk can act on; its usefulness to supervisory authorities is a possible extension of the present work rather than a result this study establishes.
Extended discussion of this subsection, with the further detail and the intermediate quantities it carries, is given in Supplementary Section S5.38.

8.7. Limitations and Future Directions

Although this study has produced encouraging results, several limitations remain. First, the empirical analysis is based on data from January to June 2025 (126 trading days, of which the final 21 form the held-out test window). While the 5-min data provide a rich intraday record, the daily aggregation leaves only 104 sliding-window observations (83 training and 21 test) for the point-forecast evaluation, and the limited calendar span may not capture the full spectrum of market regimes, including prolonged bull, bear and sideways phases. The walk-forward backtest of the risk layer uses 59 forecast origins, which is enough to exercise the high-risk tier but not enough to estimate its performance precisely, and the 21-day window available to the point-forecast comparison limits the power of the Diebold–Mariano tests; enlarging the out-of-sample period is therefore the single most important extension for future work.
Extended discussion of this subsection, with the further detail and the intermediate quantities it carries, is given in Supplementary Section S5.39.

9. Conclusions

This paper has proposed and evaluated an integrated prediction and decision support system for spot gold (XAU/USD) volatility forecasting, addressing two fundamental systemic gaps in existing approaches: the absence of a structured multi-source data fusion system and the absence of a risk warning decision system. On the sample studied, the complete system comprises three coordinated components:

1. **Multi-source data fusion system:** Integrates multimodal video sentiment embeddings extracted by a TFN, text-based news sentiment scores generated by the Qwen-2.5 large language model and conventional financial indicators into a unified factor pool, with Random-Forest-based feature selection ranking and retaining the candidate factors (20 of the 21 columns in this sample), which keeps the input pool interpretable rather than maximally wide.
2. **PSO-Transformer adaptive learning engine:** Embeds PSO into the Transformer architecture to search three pre-registered hyperparameters automatically—the model dimension, the number of training epochs and the batch size, with the remaining architecture fixed (Supplementary Table S7)—achieving MAE = 8.038 in one-step prediction and an average accuracy improvement of 36.5% over all benchmark models (GARCH, HAR-RV, LSTM, GRU, SVM, XGBoost and standard Transformer) under protocol P1, in which the hyperparameter and epoch selection window is the evaluation window, as well as on the 21-day evaluation window of this sample, the best value on all six error metrics at the one-step and two-step horizons and on five of the six at the five-step horizon. The leak-free protocol P2, whose validation period is disjoint

from the test window, does not preserve that first place: the same pipeline attains 10.16 against 8.82 for the GRU. The P1 figures are therefore within-tuning-window results, and no ranking obtained inside the tuning window is presented in this paper as a fully out-of-sample result. Diebold–Mariano tests confirm that the gains over GARCH, SVM and XGBoost remain statistically significant at the 5% level after a Holm correction for multiple comparisons, while the differences from HAR-RV, the recurrent models and the standard Transformer baseline are not.

3. **Risk warning decision system:** Establishes a three-tier traffic-light alert scheme whose lower and upper thresholds are the median and the third quartile of the realized-variance distribution, and issues each alert as the maximum a posteriori tier of a calibrated conditional predictive distribution rather than by thresholding a point forecast, with risk-management guidance for market participants in general; the supervisory use of the alerts is offered as a possible application rather than as a demonstrated benefit. Backtesting walk-forward over 59 forecast origins, with the alert anchored to the persistence model's calibrated predictive distribution, yields an overall tier-classification accuracy of 67.8% (balanced accuracy 65.2%), a high-risk precision of 78.8% and a recall of 92.9%—a property of the alert layer with this anchor rather than of the neural engine, which, fed into the same rule, reaches a lower walk-forward tier accuracy (45.5% against 61.4% for the anchor on the 44 origins where both can be scored; Supplementary Table S22)—with 50% and 80% predictive intervals that cover the realized volatility 49.2% and 79.7% of the time. Thresholding the point forecast instead produces a high-risk F1 of zero on the same sample under protocol P1, because the mean-squared-error forecast is compressed towards the conditional mean; the distributional rule is what makes the high-risk tier operational.

Ablation studies at the component level attribute the dominant contribution to the adaptive learning engine's PSO optimization (+8.0% MAE increase at the one-step horizon, +5.6% at the two-step horizon and +11.5% at the five-step horizon when it is replaced by default hyperparameters), whereas removing either sentiment input leaves the average error essentially unchanged at short horizons and becomes measurable only at the five-step horizon.

Extended discussion of this section, with the further detail and the intermediate quantities it carries, is given in Supplementary Section S5.40.

Copyright Statement. All figures and tables presented in this manuscript are original and were created by the authors. No third-party copyrighted material is used, and no permissions are required.

Supplementary Materials: The following supporting information can be downloaded at https://www.mdpi.com/article/10.3390/math14214115/s1. The following supporting information accompanies this paper and is cited in the text as *Supplementary Table S#* and *Supplementary Section S#*. Tables S1–S3 support Section 3 (sentiment channels and factor definitions); Tables S4–S6 support Section 4.2 (correlation screening, SHAP attribution and cutoff sensitivity); Tables S7 and S8 support Section 5 (implemented configuration and forecast origins); Tables S9–S15 support Section 6.1 (sample calendar, predictor alignment, the two chronological protocols, the experiment sequence, the software stack and the rolling-origin refit); Tables S16–S19 support Sections 6.4–6.8 (run-to-run stability, sentiment parameterization, search dispersion and QLIKE loss); Tables S20 and S21 support Section 6.10 (scheduled high-impact events); Table S22 supports Section 7.4 (comparison of alert rules), and Tables S23 and S24 support Section 6.3 (the exogenous-augmented recursions, their Diebold–Mariano tests and their coefficients). Section S1 collects the supplementary tables and names the main-text section each of them supports; Section S2 documents the preprocessing pipeline step by step and carries the validation record of the video sentiment classifier; Section S3 collects additional analyses and qualifications; and Section S4 gives the file inventory of the reproduction package. Section S5

gives, subsection by subsection, the extended discussion of the corresponding main-text section, together with the intermediate quantities that the main text does not print. It recomputes nothing and restates no value to a different figure, so wherever a quantity appears in both places it is the same quantity, and each subsection of Section S5 is announced by an italic pointer in the main text.

Author Contributions: Conceptualization, X.D. and X.L.; methodology, X.D., X.L., W.W. and J.W.; software, X.D., X.L. and W.W.; validation, X.D., X.L., W.W. and J.W.; formal analysis, X.D. and X.L.; investigation, X.D., X.L., W.W. and J.W.; resources, X.D., X.L. and W.W.; data curation, X.D., X.L. and W.W.; writing—original draft, X.D.; writing—review and editing, X.D.; visualization, X.D., X.L. and J.W.; supervision, X.D., X.L. and J.W.; project administration, X.D. and X.L.; funding acquisition, X.D., X.L. and J.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the China National University Student Innovation and Entrepreneurship Development Program (Grant Nos. 202611104287 and 202511104025), the Qinghai IoT Key Laboratory (Grant No. 2017-ZJ-Y21), the Hebei IoT Monitoring Technology Innovation Center (Grant No. 21567693H), the Hebei Major Science and Technology Support Program (Grant No. 202622003010002) and the Hebei Provincial Innovation Capacity Improvement Program (Grant No. 25350802D).

Institutional Review Board Statement: Not applicable. This study did not involve humans or animals.

Informed Consent Statement: Not applicable.

Data Availability Statement: Both the dataset and code supporting the findings of this study are publicly available. The primary dataset (`gold_5m_data.csv`), along with all supporting code, has been deposited in the Zenodo repository at https://zenodo.org/records/23354872 (accessed on 1 November 2026).

Acknowledgments: During the preparation of this work, the authors used AI-assisted technologies (DeepSeek V4 Pro and GLM-4) for language editing and readability improvement and DeepSeek V4 Pro for code debugging and code snippet generation during data analysis. All AI-generated output was reviewed and edited by the authors, and all AI-generated code was independently reviewed, tested and modified prior to use. The authors take full responsibility for the content of the publication.

Conflicts of Interest: Xueping Diao and Weizheng Wang were employed by the Bank of Qinghai Co., Ltd. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript: CPI, consumer price index; DM, Diebold–Mariano; EMH, efficient market hypothesis; ETF, exchange-traded fund; FOMC, Federal Open Market Committee; GARCH, generalized autoregressive conditional heteroskedasticity; GRU, gated recurrent unit; HAR-RV, heterogeneous autoregressive model of realized volatility; IQR, interquartile range; LLM, large language model; LSTM, long short-term memory; MAE, mean absolute error; MAPE, mean absolute percentage error (reported in percent); MSE, mean squared error; PSO, particle swarm optimization; QLIKE, quasi-likelihood loss for volatility forecasts (reported as a dimensionless quantity; Supplementary Table S19); Qwen, Tongyi Lab large language model series (Qwen-2.5 used in this study); RLHF, reinforcement learning from human feedback; RMSE, root mean squared error; RMSPE, root mean squared percentage error (reported as a dimensionless proportion, that is, without the factor 100; Section 5.4); RV, realized variance (sum of squared 5-min returns, reported as RV_t/M_t in bp^2 per 5-min interval; called realized volatility in the text; Section 3.3); SMAPE, symmetric mean absolute percentage error (reported in percent); SVM, support vector machine; TFN, tensor fusion network; Transformer, a deep learning architecture based on the self-attention mechanism; XAU/USD, spot gold quoted in US dollars; XGBoost, eXtreme gradient boosting.

References

1. Baur, D.G.; McDermott, T.K. Is gold a safe haven? International evidence. *J. Bank. Financ.* **2010**, *34*, 1886–1898. [CrossRef]
2. Tang, Y.; Zhong, J. Predicting gold volatility: Exploring the impact of extreme risk in the international commodity market. *Finance Res. Lett.* **2023**, *58*, 104491. [CrossRef]
3. Tan, K.L.; Lee, C.P.; Lim, K.M. A Survey of Sentiment Analysis: Approaches, Datasets, and Future Research. *Appl. Sci.* **2023**, *13*, 4550. [CrossRef]
4. Zadeh, A.; Chen, M.; Poria, S.; Cambria, E.; Morency, L.P. Tensor Fusion Network for Multimodal Sentiment Analysis. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (EMNLP)*; Association for Computational Linguistics (ACL): Stroudsburg, PA, USA, 2017; pp. 1103–1114. [CrossRef]
5. Alzazah, F.; Cheng, X.; Gao, X. Predict Market Movements Based on the Sentiment of Financial Video News Sites. In *2022 IEEE 16th International Conference on Semantic Computing (ICSC)*; IEEE: New York, NY, USA, 2022; pp. 103–110. [CrossRef]
6. Zhang, H.; Zhang, W. Facial expression sentiment in videos and stock returns. *SSRN Electron. J.* **2023**, SSRN ID 4569776. [CrossRef]
7. Liu, Y.; Huang, Y. A multimodal deep learning framework for constructing a market sentiment index from stock news. *Big Data Res.* **2025**, *41*, 100535. [CrossRef]
8. Andersen, T.G.; Bollerslev, T.; Diebold, F.X.; Labys, P. Modeling and Forecasting Realized Volatility. *Econometrica* **2003**, *71*, 579–625. [CrossRef]
9. Bollerslev, T. Generalized autoregressive conditional heteroskedasticity. *J. Econom.* **1986**, *31*, 307–327. [CrossRef]
10. Nelson, D.B. Conditional heteroskedasticity in asset returns: A new approach. *Econometrica* **1991**, *59*, 347–370. [CrossRef]
11. Glosten, L.R.; Jagannathan, R.; Runkle, D.E. On the relation between the expected value and the volatility of the nominal excess return on stocks. *J. Finance* **1993**, *48*, 1779–1801. [CrossRef]
12. Corsi, F. A Simple Approximate Long-Memory Model of Realized Volatility. *J. Financ. Econom.* **2009**, *7*, 174–196. [CrossRef]
13. Wang, H.; Sheng, H.; Zhang, H.w. Influence factors of international gold futures price volatility. *Trans. Nonferrous Met. Soc. China* **2019**, *29*, 2447–2454. [CrossRef]
14. Li, X.; Guo, Q.; Liang, C.; Umar, M. Forecasting gold volatility with geopolitical risk indices. *Res. Int. Bus. Financ.* **2023**, *64*, 101857. [CrossRef]
15. Chuang, O.C.; Gupta, R.; Pierdzioch, C.; Shu, B. Financial Uncertainty and Gold Market Volatility: Evidence from a Generalized Autoregressive Conditional Heteroskedasticity Variant of the Mixed-Data Sampling (GARCH-MIDAS) Approach with Variable Selection. *Econometrics* **2024**, *12*, 38. [CrossRef]
16. Luo, J.; Demirer, R.; Gupta, R.; Ji, Q. Forecasting oil and gold volatilities with sentiment indicators under structural breaks. *Energy Econ.* **2022**, *105*, 105751. [CrossRef]
17. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention Is All You Need. In Proceedings of the Advances in Neural Information Processing Systems 30 (NeurIPS 2017), Long Beach, CA, USA, 4–9 December 2017; pp. 5998–6008.
18. Lim, B.; Arık, S.O.; Loeff, N.; Pfister, T. Temporal Fusion Transformers for interpretable multi-horizon time series forecasting. *Int. J. Forecast.* **2021**, *37*, 1748–1764. [CrossRef]
19. He, K.; Yang, Q.; Ji, L.; Pan, J.; Zou, Y. Financial Time Series Forecasting with the Deep Learning Ensemble Model. *Mathematics* **2023**, *11*, 1054. [CrossRef]
20. Zhou, H.; Zhang, S.; Peng, J.; Zhang, S.; Li, J.; Xiong, H.; Zhang, W. Informer: Beyond Efficient Transformer for Long Sequence Time-Series Forecasting. In *Proceedings of the 35th AAAI Conference on Artificial Intelligence (AAAI 2021)*; Association for the Advancement of Artificial Intelligence (AAAI): Palo Alto, CA, USA, 2021; Volume 35, pp. 11106–11115. [CrossRef]
21. Wan, L.; Tao, Y.; Wang, J.; Zhu, W.; Tang, C.; Zhou, G. A multi-scale multi-head attention network for stock trend prediction considering textual factors. *Appl. Soft Comput.* **2025**, *171*, 112388. [CrossRef]
22. Hong, A.; Gao, M.; Gao, Q.; Peng, X.H. Non-stationary financial time series forecasting based on meta-learning. *Electron. Lett.* **2023**, *59*, e12681. [CrossRef]
23. Lorenzo, P.R.; Nalepa, J.; Sanchez Ramos, L.; Ranilla Pastor, J. Hyper-parameter Selection in Deep Neural Networks Using Parallel Particle Swarm Optimization. In *Proceedings of the Genetic and Evolutionary Computation Conference Companion (GECCO '17)*; ACM (Association for Computing Machinery): New York, NY, USA, 2017; pp. 1864–1871. [CrossRef]
24. Kennedy, J.; Eberhart, R. Particle swarm optimization. In *Proceedings of ICNN'95—International Conference on Neural Networks*; IEEE: New York, NY, USA, 1995; Volume 4, pp. 1942–1948. [CrossRef]
25. Salisu, A.A.; Adediran, I.; Omoke, P.C.; Tchankam, J.P. Gold and tail risks. *Resour. Policy* **2023**, *80*, 103154. [CrossRef]
26. Sakowski, P.; Sieradzki, R.; Ślepaczuk, R. Systemic Risk Indicator Based on Implied and Realized Volatility. *Qual. Quant.* **2026**, *60*, 6389–6427. [CrossRef]
27. McNeil, A.J.; Frey, R.; Embrechts, P. *Quantitative Risk Management: Concepts, Techniques and Tools*, revised ed.; Princeton Series in Finance; Princeton University Press: Princeton, NJ, USA, 2015.

28. Tang, X.; Song, Y.; Jiao, X.; Sun, Y. On Forecasting Realized Volatility for Bitcoin Based on Deep Learning PSO–GRU Model. *Comput. Econ.* **2024**, *63*, 2011–2033. [CrossRef]
29. Fama, E.F. Efficient Capital Markets: A Review of Theory and Empirical Work. *J. Financ.* **1970**, *25*, 383–417. [CrossRef]
30. Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G.S.; Dean, J. Distributed Representations of Words and Phrases and their Compositionality. In Proceedings of the Advances in Neural Information Processing Systems 26 (NIPS 2013), Lake Tahoe, NV, USA, 5–8 December 2013; pp. 3111–3119.
31. Yang, A.; Yang, B.; Zhang, B.; Hui, B.; Zheng, B.; Yu, B.; Li, C.; Liu, D.; Huang, F.; Wei, H.; et al. Qwen2.5 Technical Report. *arXiv* **2024**, arXiv:2412.15115. [CrossRef]
32. Andersen, T.G.; Bollerslev, T.; Diebold, F.X.; Labys, P. The Distribution of Realized Exchange Rate Volatility. *J. Am. Stat. Assoc.* **2001**, *96*, 42–55. [CrossRef]
33. Liu, L.Y.; Patton, A.J.; Sheppard, K. Does anything beat 5-minute RV? A comparison of realized measures across multiple asset classes. *J. Econom.* **2015**, *187*, 293–311. [CrossRef]
34. Guyon, I.; Elisseeff, A. An Introduction to Variable and Feature Selection. *J. Mach. Learn. Res.* **2003**, *3*, 1157–1182. [CrossRef]
35. Breiman, L. Random Forests. *Mach. Learn.* **2001**, *45*, 5–32. [CrossRef]
36. Bengio, Y.; Simard, P.; Frasconi, P. Learning long-term dependencies with gradient descent is difficult. *IEEE Trans. Neural Netw.* **1994**, *5*, 157–166. [CrossRef] [PubMed]
37. Hochreiter, S.; Schmidhuber, J. Long Short-Term Memory. *Neural Comput.* **1997**, *9*, 1735–1780. [CrossRef] [PubMed]
38. Niu, Z.; Zhong, G.; Yu, H. A review on the attention mechanism of deep learning. *Neurocomputing* **2021**, *452*, 48–62. [CrossRef]
39. Shi, Y.; Eberhart, R. A modified particle swarm optimizer. In Proceedings of the 1998 IEEE International Conference on Evolutionary Computation Proceedings. IEEE World Congress on Computational Intelligence (Cat. No.98TH8360), Anchorage, AK, USA, 4–9 May 1998; pp. 69–73. [CrossRef]
40. Nickabadi, A.; Ebadzadeh, M.M.; Safabakhsh, R. A novel particle swarm optimization algorithm with adaptive inertia weight. *Appl. Soft Comput.* **2011**, *11*, 3658–3670. [CrossRef]
41. Diebold, F.X.; Mariano, R.S. Comparing Predictive Accuracy. *J. Bus. Econ. Stat.* **1995**, *13*, 253–263. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.