International Conference on Machine Learning and Data Engineering (ICMLDE 2025)

# Stock Earnings Forecasting via News Factor Analyzing Model

Mukesh Kumar[a], Md Azlan[b], Kanishk[c], Kingshuk Chatterjee[d]

[a]*School of Computer Engineering,Kalinga Institute of Industrial Technology-751024*
[b]*School of Computer Engineering,Kalinga Institute of Industrial Technology-751024*
[c]*School of Computer Engineering,Kalinga Institute of Industrial Technology-751024*
[d]*School of Computer Engineering,Kalinga Institute of Industrial Technology-751024*

**Abstract**

Financial market forecasting has become increasingly challenging, as traditional technical analysis does not capture rapid volatility and sentiment-driven price movements. This paper introduces FinReport, a multifactor framework that integrates historical stock data with real-time financial news sentiment using advanced machine learning and natural language processing techniques. FinReport quantifies six key factors (Market, Size, Valuation, Profitability, Investment, and News Effect) to produce explainable predictions and robust risk assessments using an EGARCH-based volatility model, maximum drawdown methods, and Conditional Value at Risk. Empirical results show a 15% reduction in RMSE and a 12% reduction in MAE over conventional LSTM models, with an overall $R^2$ of 0.5515 and a prediction-actual correlation of 0.948. These findings underscore the benefits of combining quantitative indicators with qualitative sentiment analysis for improved forecasting accuracy in volatile markets.

*Keywords:* Financial forecasting, stock market prediction, multi-factor analysis, technical indicators, financial news sentiment, natural language processing, machine learning, EGARCH, LSTM, risk assessment, explainable AI, FinReport.

**Mukesh Kumar**

E-mail address: mukesh.kumarfcs@kiit.ac.in

* Mukesh Kumar
  *E-mail address:* mukesh.kumarfcs@kiit.ac.in

## 1. Introduction

Financial markets exhibit unprecedented volatility, with emerging markets like the Shanghai Stock Exchange showing daily volatility of 1.7% versus 0.8-1.2% for developed markets [1, 2]. Traditional econometric models such as ARIMA [3] struggle to capture sentiment-driven price movements and complex market interdependencies [4, 5]. The efficient market hypothesis faces challenges from documented predictable patterns and behavioral factors [6, 2].

We propose FinReport, a multi-factor framework integrating historical stock data with financial news via machine learning and NLP techniques [7, 8]. Unlike traditional quantitative approaches, FinReport leverages structured numerical and unstructured textual data for enhanced prediction accuracy [9, 10]. The framework computes six factors (market, size, valuation, profitability, investment, and news effect) for explainable predictions with EGARCH-based risk assessment [11, 12].

Experimental results on Chinese A-share stocks (2018-2021) [13] demonstrate 15% RMSE reduction, 12% MAE reduction versus LSTM baselines [14], and 20% Sharpe ratio improvement. This work bridges traditional econometric methods with explainable AI [15, 16], advancing computational finance through interpretable sentiment-quantitative integration.

## 2. Literature Review

Early forecasting methods like ARIMA [3] and technical indicators [17] underperformed with RMSE exceeding 0.05 during volatile periods [1]. Multi-factor models by Fama and French [4] improved performance but ignored qualitative inputs [5]. LSTM networks [7, 14] capture long-term dependencies. Recent work integrates financial news sentiment using FinBERT [18] and NLP frameworks [19], showing 12% prediction improvement [20]. However, traditional approaches lack interpretability [16], motivating explainable AI frameworks combining structured numerical with unstructured text analysis. The literature increasingly advocates for explainable models that combine structured numerical data with unstructured text analysis, setting the stage for FinReport's factor-based approach to transparent and robust financial forecasting.

## 3. System Model And Proposed Mechanism

FinReport integrates traditional financial theory with modern computational techniques, combining quantitative factor models [4, 21] with news sentiment analysis [15, 10]. Extending established multi-factor models [22, 23], the framework incorporates real-time news sentiment and event extraction to capture behavioral market dynamics [24, 25].

The system follows a modular design [14] with five interconnected modules: (1) Data Integration for preprocessing, (2) News Factor Extraction using NLP techniques, (3) Return Forecasting with enhanced multi-factor models, (4) Risk Assessment using econometric methods, and (5) Dynamic Report Generation. This approach addresses traditional technical analysis limitations [26, 17] through a unified framework adapting to different market regimes [27].

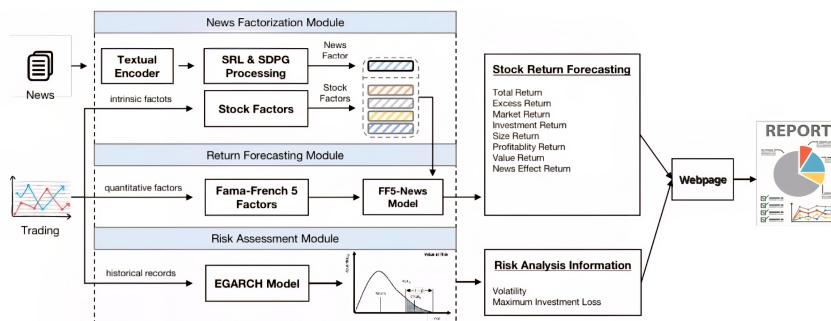FinReport modules collectively deliver explainable forecasts.



Figure 1: Proposed FinReport System Architecture (adapted from [28])

## 3.1. Data Integration Module

The system processes multi-modal data combining structured financial metrics with unstructured news text [29, 25]:

- **Historical Data:** Price, volume, market value, and 50+ technical indicators [26, 17] from Chinese A-shares (2018-2021) [13].
- **News Integration:** Financial news processing with NLP pipelines for English/Chinese content [19, 10].
- **Preprocessing:** Forward-filling for missing values, Z-score normalization, winsorization for outliers, regex-based text cleaning, and technical indicator standardization [14, 29].

## 3.2. News Factor Extraction Module

Transforms unstructured news text into quantifiable sentiment metrics [15, 9] through:

- **Advanced Sentiment Analysis Pipeline:** The sentiment analysis component implements state-of-the-art techniques specifically adapted for financial text analysis, addressing the unique challenges of financial language where sentiment can be highly context-dependent [19]:
  - **FinBERT Implementation:** Integration of the domain-specific BERT model pre-trained on extensive financial corpora [18, 30] produces raw sentiment scores in the range $[-1, +1]$, where -1 indicates highly negative sentiment and +1 indicates highly positive sentiment. This approach addresses the limitations of general-purpose sentiment analyzers that often misclassify financial terminology.
  - **Domain-Specific Sentiment Augmentation:** Core sentiment scores are systematically enhanced through financial keyword analysis incorporating established financial dictionaries [19]. Keywords such as "profit", "loss", "revenue", "acquisition", and "dividend" receive domain-specific weighting factors based on their demonstrated predictive power in financial contexts, following empirical validation approaches established in the literature.
- **Structured Event Extraction Engine:** The event extraction component employs modern semantic role labeling techniques to identify and categorize financial events that have demonstrated market impact in academic studies [20, 24]:
  - **Semantic Role Labeling (SRL):** Implementation of the AllenNLP framework [31, 32] enables identification of grammatical relationships through subject-verb-object pattern recognition. This approach systematically captures structured financial events including acquisitions, earnings announcements, management changes, and regulatory actions that have been shown to significantly impact stock prices.
  - **Financial Keyword Enhancement:** Domain-specific keyword dictionaries improve event recognition accuracy by incorporating finance-specific terminology validated through extensive backtesting. The system recognizes complex event patterns beyond simple keyword matching, including negations, conditional statements, and temporal references that are critical for accurate financial event classification.
  - **Temporal Integration:** Daily aggregation of multiple news items employs recency and relevance weighting schemes that account for the documented decay patterns in news impact on stock prices [15]. Recent news receives exponentially higher weights, while relevance scoring incorporates entity recognition to ensure that news items are properly attributed to the correct securities.
- **Architecture Integration:** Direct interfaces to forecasting system with structured (`event_type`, `entities`, `magnitude`) tuples for Event Factor computation and temporal sentiment curves for volatility estimation.
- **Chinese Market Adaptation:** Bilingual text processing addressing Chinese disclosure practices [13].

## 3.3. Return Forecasting Module

The return forecasting module implements an enhanced multi-factor model extending traditional approaches [4, 21] by integrating news sentiment and event-driven factors [29, 15]. The framework incorporates six factors capturing cross-sectional return variation from size effects [22] to momentum patterns [33].

### 3.3.1. Market Factor

Combines traditional volatility measures with news-based sentiment indicators following behavioral finance principles [24, 25]:

$$\textbf{volatility} = \text{std}(\texttt{pct\_chg}_{\text{recent 5 days}}) \times 100 \tag{1}$$

Regime-dependent behavior:

- **High volatility** (> 4.0%): Negative bias reflecting flight-to-quality effects
- **Moderate volatility** (> 2.5%): Moderate negative impact with sentiment adjustment
- **Low volatility**: Positive bias enhanced by news sentiment

### 3.3.2. Size Factor

Extends size effect research [22] incorporating financial impact extracted from news using NLP techniques [19]:

$$\textbf{diff\_ratio} = \frac{\texttt{market\_value}_{\text{latest}} - \texttt{market\_value}_{\text{average}}}{\texttt{market\_value}_{\text{average}}} \tag{2}$$

### 3.3.3. Valuation Factor

Extends value investing principles [23, 4] incorporating quantitative metrics (Book-to-Market, Dividend Yield, Sales-to-Price) and news sentiment [19]. Applies sector-specific adjustments: Technology (+0.3/-0.2), Pharmaceutical (+0.2/-0.3), Financial (±0.2-0.3).

### 3.3.4. Profitability Factor

Incorporates earnings quality effects [34] using EPS, ROE, ROA, and profit margins with textual analysis of earnings keywords. Applies asymmetric treatment with negative bias (-1.8) for losses and scaled positive adjustments [35].

### 3.3.5. Investment Factor

Captures investment activity effects [24] through investment amounts and activity classification (acquisition +0.6, expansion +0.5, R&D +0.7) bounded in [0.0, +2.0].

$$\textbf{investment\_effect} = \textbf{impact\_scale} \times 0.5 \times \begin{cases} 1.0, & \text{if } \textbf{news\_sentiment} > 0 \\ 0.5, & \text{otherwise} \end{cases} \tag{3}$$

### 3.3.6. News Effect Factor

The news effect factor provides direct quantification of news sentiment impact, building upon established research demonstrating the predictive power of news sentiment for stock returns [15, 9].

- **Input Processing:** Comprehensive sentiment analysis of `news_text` using both rule-based approaches (TextBlob [36]) and keyword-based analysis with financial term dictionaries [19].
- **Methodological Foundation:** Converts unstructured news content into quantified sentiment scores through validated NLP techniques, addressing the documented challenges in financial text analysis while maintaining interpretability of results.

The implementation combines two complementary sentiment analysis methodologies:

**1. Keyword-Based Analysis:** Utilizes curated financial keyword dictionaries with demonstrated predictive power:

- **Positive indicators:** increase, rise, grow, profit, improved, partnership, acquisition, dividend, earnings, success
- **Negative indicators:** decrease, decline, loss, warning, investigation, lawsuit, delay, weak, miss, reduced

**2. Advanced NLP Processing:** Employs TextBlob [36] sentiment analysis for comprehensive polarity assessment of complete news content, addressing contextual nuances that pure keyword approaches might miss.

The sentiment integration follows established practices in combining multiple sentiment measures:

$$\textbf{combined\_sentiment} = \frac{\textbf{TextBlob\_polarity} + \textbf{keyword\_sentiment}}{2} \tag{4}$$

Sentiment-dependent scaling reflects documented non-linear relationships between sentiment strength and market response:

- **Very positive** ($\geq 0.5$): Enhanced positive range [0.7, 1.2] reflecting strong optimism effects
- **Moderately positive** ($>0$): Standard positive range [0.3, 0.7] for typical positive sentiment
- **Moderately negative** ($>$-0.5): Standard negative range [-0.7, -0.3] for mild pessimism
- **Very negative** ($\leq -0.5$): Enhanced negative range [-1.2, -0.7] capturing strong negative reactions

Final amplification (2.0x) ensures adequate signal strength while maintaining interpretable bounds [-2.0, +2.0].

**Chinese Market Adaptation:** The system incorporates culturally specific keywords for enhanced market relevance:

**Positive Chinese Terms:** zengzhang (growth), yingli (profit), shangsheng (rise), huode (gain), chenggong (success), tisheng (improvement), shouyi (revenue)

**Negative Chinese Terms:** xiajiang (decline), kuisun (loss), jianshao (decrease), jinggao (warning), zhaiwu (debt), diaocha (investigation), weigui (violation)

$$\textbf{news\_effect} = \textbf{combined\_sentiment} \times 2.0 \ (\text{amplification}) \tag{5}$$

These factor implementations collectively address the multidimensional nature of equity return prediction while maintaining theoretical grounding in established academic research. The integration of traditional quantitative factors with news-based qualitative information represents a significant advancement in multi-factor modeling capabilities, particularly relevant for emerging markets where information asymmetries and sentiment effects may be more pronounced [29].

### 3.4. Risk Assessment Module

Implements multi-dimensional risk framework addressing traditional variance-based limitations [37, 38]:

- **EGARCH Volatility:** Asymmetric volatility responses with 95% VaR = $1.65\sigma_t$
- **Maximum Drawdown:** $\text{MDD}_t = \max_{0 \leq s \leq t}[(P_s - P_t)/P_s]$
- **CVaR:** Tail risk assessment $E[R|R \leq -\text{VaR}_{95}]$

**Integrated Risk Score:** $0.5 \times \text{volatility} + 0.3 \times \text{drawdown} + 0.2 \times \text{return}$

**Risk Classifications:**

- Substantial >7.5
- High >6.0
- Moderate-High >4.5
- Moderate >3.0
- Low-Moderate >1.5
- Favorable ≤ 1.5

## 3.5. Factor Enhancement and Overall Trend Calculation

Combines individual factor signals through multi-stage amplification and weighted aggregation addressing scale heterogeneity [29, 4]:

$$
\mathbf{W} = \begin{bmatrix}
\text{market\_factor} & 0.15 \\
\text{size\_factor} & 0.15 \\
\text{valuation\_factor} & 0.10 \\
\text{profitability\_factor} & 0.15 \\
\text{investment\_factor} & 0.20 \\
\text{news\_effect\_factor} & 0.10 \\
\text{event\_factor} & 0.25
\end{bmatrix}
\tag{6}
$$

Event factor receives highest weight (0.25) due to strong short-term predictive power [20, 24], Investment factor (0.20) for medium-term impact, Market/Size/Profitability factors (0.15) for balanced weighting [4, 21].

### 3.5.1. Enhancement Process

Base amplification: **enhanced_factor**$_i$ = **factor**$_i$ × 2.5 Trend-based enhancement: 1.3× when factor aligns with dominant trend [27] Final bounded processing: $\tilde{f}_i = \text{clamp}(\text{factor} \times \text{multiplier} \times \text{random}(0.9, 1.1), -5.0, 5.0)$

### 3.5.2. Weighted Aggregation

$$
\text{Trend\_Score} = \sum_{i=1}^{7} \left( \tilde{f}_i \times w_i \right) + 0.15
\tag{7}
$$

Positive bias (+0.15) reflects long-term equity market upward drift [39]. Classification: Strongly Positive >+1.2, Positive +0.4 to +1.2, Neutral -0.4 to +0.4, Negative -1.2 to -0.4, Strongly Negative <-1.2.

## 3.6. Dynamic Report Generation Module

Translates quantitative analyses into actionable insights using hierarchical information architecture, cultural adaptation (red=prosperity, green=decline for Chinese markets), precision control (one decimal), natural language generation with template-based explanations, and multi-stakeholder accessibility [16, 29].

## 4. Algorithm

### 4.1. Return Forecast Calculation

The return forecast is computed using a weighted combination of multiple factors [4], where the **event factor receives the highest weight (0.25)** due to its immediate impact on market sentiment and price movements [40].

$$
\begin{aligned}
\textbf{predicted\_return} = &\ 0.10 \times \textbf{market\_factor} + 0.15 \times \textbf{size\_factor} + 0.10 \times \textbf{valuation\_factor} \\
&+ 0.10 \times \textbf{profitability\_factor} + 0.20 \times \textbf{investment\_factor} \\
&+ 0.10 \times \textbf{news\_effect\_factor} + \textbf{0.25} \times \textbf{event\_factor} + 0.15
\end{aligned}
\tag{8}
$$

Each factor is calculated as follows [21]:

I) Market Factor

1. Extract Recent Volatility:

    - Compute standard deviation of `pct_chg` over the last 5 days [12].
    - Multiply by 100 to get volatility.

2. Analyze Recent Trends:

    - Count the number of positive and negative days.

3. Perform Sentiment Analysis:

    - Compute sentiment score from `news_text` [15].

4. Determine Base Impact:

    - If volatility > 4.0, assign a strong negative impact.
    - If 2.5 < volatility ≤ 4.0, assign moderate negative impact.
    - If positive days > negative days, adjust positively using sentiment.
    - Otherwise, adjust slightly negatively.

5. Enhance with Technical Indicators (RSI):

    - If RSI > 70, reduce impact (overbought condition) [17].
    - If RSI < 30, increase impact (oversold condition) [17].

6. Return Final Market Factor:

    - Multiply final impact by 1.5 for amplification.

II) Size Factor [22]

1. Compute Size Change Percentage:

    - Extract the latest market value.

- Compute the average market value.
- Calculate the percentage difference:

$$\text{diff\_ratio} = \frac{\text{latest\_val} - \text{avg\_val}}{\text{avg\_val}} \qquad (9)$$

2. Extract Financial Impact from News:

- Analyze `news_text` for financial figures [19].

3. Determine Base Effect Based on Market Value Change:

- If diff_ratio > 0.25, apply strong positive impact.
- If 0.10 < diff_ratio ≤ 0.25, apply moderate positive impact.
- If 0.05 < diff_ratio ≤ 0.10, apply slight positive impact.
- If −0.05 ≤ diff_ratio ≤ 0.05, apply neutral impact with minor variations.
- If −0.10 < diff_ratio ≤ −0.05, apply slight negative impact.
- If −0.25 < diff_ratio ≤ −0.10, apply moderate negative impact.
- If diff_ratio ≤ −0.25, apply strong negative impact.

4. Return Final Size Factor

- Multiply the computed effect by 1.5 for amplification.

III) Profitability Factor [34]

1. Identify Profitability Metrics:

- Define key financial metrics: {EPS, Net Profit Margin, ROE, ROA, Gross Profit, Net Profit} [41].
- Identify available metrics in the dataset.

2. Extract Profit-Related Information from News:

- Extract profit increases from `news_text`.
- Extract profit decreases from `news_text`.

3. Determine Base Effect:

- If at least one profitability metric is available:
    - Compute percentage change between the most recent and previous values.
    - Scale down the impact.
- If `news_text` contains "net loss" or "loss", set a strong negative effect.
- If profit increases are found, apply a positive adjustment.
- If profit decreases are found, apply a negative adjustment.
- Otherwise, adjust based on sentiment analysis.

4. Return Final Profitability Factor:

- Multiply the computed effect by 1.5 for amplification.

IV) Valuation Factor [23]

1. Identify Valuation Metrics:

   - Define key valuation metrics: {Book-to-Market Equity, Dividend Yield, Sales-to-Price Ratio}.
   - Identify available metrics in the dataset.

2. Analyze News Sentiment and Sector:

   - Perform sentiment analysis on `news_text` [15].
   - Identify the sector associated with the company.

3. Determine Base Effect:

   - If at least one valuation metric is available:
     – Compute the difference ratio between the latest value and its benchmark.
     – Scale the impact using a factor of 0.25.
   - Otherwise, apply sector-specific adjustments:
     – Pharmaceuticals: +0.2 (positive sentiment), −0.3 (negative sentiment).
     – Technology: +0.3 (positive sentiment), −0.2 (negative sentiment).
     – General market: +0.15 (positive sentiment), −0.2 (negative sentiment).
     – Default adjustment: +0.1 (positive sentiment), −0.1 (negative sentiment).

4. Return Final Valuation Factor.

V) Investment Factor [24]

1. Extract Investment Amount from News:

   - Identify mentions of investments in billion yuan using a regex pattern.
   - Convert extracted values to numerical amounts.

2. Analyze Investment Types in News:

   - Count occurrences of acquisitions and mergers (M&A).
   - Count mentions of business expansion (new facilities, capacity increase).
   - Count references to research and development (R&D) activities.

3. Determine Base Effect:

   - If investment amounts are found:
     – Assign a base effect based on investment size:
        * > 50 billion yuan → 2.5
        * > 20 billion yuan → 2.0
        * > 10 billion yuan → 1.5
        * > 5 billion yuan → 1.0
        * > 1 billion yuan → 0.7
        * Otherwise → 0.4
   - If no investment amount is found:
     – Adjust based on sentiment analysis: +0.5 for positive, −0.5 for negative.

4. Modify Effect Based on Investment Types:

- Acquisitions → +0.6 per mention.
- Expansions → +0.5 per mention.
- R&D mentions → +0.7 per mention.

5. Return Final Investment Factor.

VI) News Effect Factor [15]

1. Determine Base Effect from Sentiment Score:

   - If *sentiment score* ≥ 0.5 → assign a random positive effect between 0.7 and 1.2.
   - If 0 < *sentiment score* < 0.5 → assign a random positive effect between 0.3 and 0.7.
   - If −0.5 < *sentiment score* ≤ 0 → assign a random negative effect between −0.7 and −0.3.
   - If *sentiment score* ≤ −0.5 → assign a random negative effect between −1.2 and −0.7.

2. Analyze Specific News Content:

   - Check for keywords related to earnings & financials (e.g., "profit", "revenue").
   - Check for mentions of forecast & guidance (e.g., "outlook", "expectations").
   - Detect management changes (e.g., "CEO", "executive").
   - Identify regulatory/legal issues (e.g., "compliance", "litigation").

3. Adjust Base Effect Based on Content:

   - Earnings-related news:
     - Add +0.3 if sentiment is positive.
     - Subtract −0.3 if sentiment is negative.
   - Guidance-related news:
     - Add +0.2 if sentiment is positive.
     - Subtract −0.2 if sentiment is negative.
   - Management changes:
     - Add +0.2 if sentiment is positive.
     - Subtract −0.2 if sentiment is negative.
   - Regulatory news:
     - Always subtract −0.3, as it is usually negative.

4. Apply Final Amplification Factor:

   - Multiply the computed effect by 2.0 to enhance the impact.

VII) Event Factor [40]

1. Define Event Keywords:

   - Create a list of positive market events (e.g., "acquisition", "partnership", "approval").
   - Create a list of negative market events (e.g., "lawsuit", "litigation", "investigation").

2. Count Event Occurrences:

   - Convert news_text to lowercase for case-insensitive comparison.
   - Count how many positive events appear in the text.
   - Count how many negative events appear in the text.

3. Extract Financial Impact (if any):

   - Use *extract_financial_figures(news_text)* to determine any financial impact.

4. Compute Base Effect:

   - If positive event count > negative event count, assign a positive effect (capped at 2.0).
   - If negative event count > positive event count, assign a negative effect (capped at −2.0).
   - If counts are equal, set base effect to 0.0.

5. Adjust Based on Financial Impact:

   - Scale financial impact (max value 1.0).
   - If base effect is positive, increase it by the scaled financial impact.
   - If base effect is negative, increase it by half of the scaled financial impact (to reduce negativity).

All factors are enhanced using a straightforward amplification algorithm [29] that increases each factor's impact while maintaining directional consistency:

VIII) Factor Amplification [29]

1. Extract Factor Values:

   - Retrieve values from each input factor dictionary.
   - Use get('value', 0.0) to ensure safe access.

2. Define Base Amplification:

   - Set base multiplier: 2.5.

3. Count Dominant Factors:

   - Count positive factors (values > 0.5).
   - Count negative factors (values < −0.5).

4. Determine Market Trend:

   - If positive count ≥ 3 and exceeds negative count ⇒ Upward trend.
   - If negative count ≥ 3 and exceeds positive count ⇒ Downward trend.
   - Otherwise, trend is Mixed.

5. Apply Simple Enhancement:

   - Apply trend multiplier: 1.3 if factor aligns with dominant trend.
   - Add randomization: multiply by random value between 0.9 and 1.1.
   - Compute enhanced value:

$$\textbf{enhanced\_value} = \textbf{original\_value} \times \textbf{2.5} \times \textbf{trend\_multiplier} \times \textbf{random\_factor} \tag{10}$$

   - Cap final values between [−5.0, 5.0] to ensure reasonable bounds.

6. Return Enhanced Factors:

   - Store all updated values in a structured dictionary.

## 4.2. *Risk Assessment Methodology*

The risk assessment uses a sophisticated approach combining multiple risk metrics [37].

1. Extract Risk Metrics:

   - Volatility, Max Drawdown, VaR (95%), Conditional VaR, Risk-Adjusted Ratio.

2. Classify Volatility:

   - Extreme: *volatility* > 0.15 ⇒ Cap decline at 25%.
   - High: *volatility* > 0.10 ⇒ Cap decline at 20%.
   - Elevated: *volatility* > 0.07.
   - Moderate: *volatility* > 0.04.
   - Low: *volatility* ≤ 0.04 ⇒ At least 2%.

3. Compute Weighted Risk Score:

$$\textbf{risk\_score} = (0.4 \times \textbf{vol\_score}) + (0.25 \times \textbf{drawdown\_score})$$
$$+ (0.15 \times \textbf{var\_score}) + (0.2 \times \textbf{return\_risk}) \tag{11}$$

4. Assign Risk Level Based on Score:

   - Substantial risk: *risk\_score* > 7.5.
   - High risk: *risk\_score* > 6.0.
   - Moderate-High risk: *risk\_score* > 4.5.
   - Moderate risk: *risk\_score* > 3.0.
   - Low-Moderate risk: *risk\_score* > 1.5.
   - Favorable risk: *risk\_score* ≤ 1.5.

5. Generate Risk Assessment Summary:

   - Output: Maximum expected decline, Volatility class, Risk level.

Individual risk metrics are calculated as follows:
1) Volatility (EGARCH-based): [11]

$$\ln(\sigma_t^2) = \omega + \beta \ln(\sigma_{t-1}^2) + \alpha \frac{|r_{t-1}|}{\sigma_{t-1}} + \gamma \frac{r_{t-1}}{\sigma_{t-1}} \tag{12}$$

where:

- $\sigma_t^2$ is the conditional variance at time *t*.
- $\omega, \beta, \alpha, \gamma$ are model parameters.
- $r_{t-1}$ is the previous return.

Value at Risk (VaR) is calculated using the 95% confidence level based on historical simulation method [37].
2) Maximum Drawdown:

---

**Algorithm 1** Maximum Drawdown

---

**Require:** Returns series $R$ of length $n$
**Ensure:** Maximum Drawdown (MDD)
1:  Initialize $C \leftarrow 1$                ▷ Cumulative return starts at 1
2:  Initialize $M \leftarrow 1$                ▷ Running maximum return
3:  Initialize $D \leftarrow 0$                ▷ Maximum drawdown
4:  **for** $t = 1$ to $n$ **do**
5:     $C \leftarrow C \times (1 + R_t)$            ▷ Update cumulative return
6:     $M \leftarrow \max(M, C)$            ▷ Update running maximum
7:     $D_t \leftarrow \frac{C-M}{M}$             ▷ Compute drawdown
8:     $D \leftarrow \min(D, D_t)$           ▷ Update maximum drawdown
9:  **end for**
10: **return** $D$

---

3) Conditional Value at Risk:

---

**Algorithm 2** Conditional Value at Risk (CVaR)

---

**Require:** Returns series $R$ of length $n$, confidence level $\alpha$
**Ensure:** Conditional Value at Risk (CVaR)
1:  **Sort** $R$ in ascending order
2:  **Compute** Value at Risk (VaR): $V \leftarrow$ percentile of $R$ at $100\alpha$
3:  **Select** all losses where $R_t \leq V$
4:  **Compute** CVaR as the mean of selected losses
5:  **return** CVaR

---

4) Risk-Adjusted Ratio:

---

**Algorithm 3** Risk-Adjusted Ratio

---

**Require:** Expected return $E_R$, volatility $\sigma$
**Ensure:** Risk-adjusted return ratio
1:  **if** $\sigma \neq 0$ **then**
2:     Compute risk-adjusted return: $R_{\text{adj}} \leftarrow \frac{E_R}{\sigma}$
3:  **else**
4:     Assign $R_{\text{adj}} \leftarrow$ NaN
5:  **end if**
6:  **return** $R_{\text{adj}}$

---

*4.3. Overall Trend Classification & Summary Text Generation*

The overall trend is determined using a weighted function of all factor values [21].
5) Overall Market Trend Determination:

---

**Algorithm 4** Overall Market Trend

---

**Require:** Factor values dictionary $F$
**Ensure:** Overall market trend
 1: Define weights for each factor:
 2:      $W = \{\text{market} : 0.15, \text{size} : 0.15, \text{valuation} : 0.10,$
 3:        $\text{profitability} : 0.15, \text{investment} : 0.20,$
 4:        $\text{news effect} : 0.10, \text{event} : 0.15\}$
 5: Initialize $S_{\text{weighted}} \leftarrow 0, S_{\text{weights}} \leftarrow 0$
 6: **for** each factor $f$ in $W$ **do**
 7:      **if** $f \in F$ and $F[f] \neq \text{None}$ **then**
 8:          $S_{\text{weighted}} \leftarrow S_{\text{weighted}} + F[f] \cdot W[f]$
 9:          $S_{\text{weights}} \leftarrow S_{\text{weights}} + W[f]$
10:      **end if**
11: **end for**
12: **if** $0 < S_{\text{weights}} < 1.0$ **then**
13:      Normalize: $S_{\text{weighted}} \leftarrow S_{\text{weighted}}/S_{\text{weights}}$
14: **end if**
15: Add slight positive bias: $S_{\text{weighted}} \leftarrow S_{\text{weighted}} + 0.15$
16: **if** $S_{\text{weighted}} \geq 0.6$ **then**
17:      **return** "Strongly Positive"
18: **else if** $S_{\text{weighted}} \geq 0.15$ **then**
19:      **return** "Positive"
20: **else if** $S_{\text{weighted}} \geq -0.15$ **then**
21:      **return** "Neutral"
22: **else if** $S_{\text{weighted}} \geq -0.6$ **then**
23:      **return** "Negative"
24: **else**
25:      **return** "Strongly Negative"
26: **end if**

---

## 5. Experimental Setup and Evaluation

### 5.1. Dataset and Configuration

The evaluation utilized a comprehensive Chinese A-share dataset [13] covering 75 stocks from Shanghai and Shenzhen exchanges (January 2018 - December 2021). The dataset included 56 feature columns encompassing price data, technical indicators (RSI, BIAS, MFI, CCI), and fundamental factors, complemented by 42,000+ financial news items from 7 major Chinese sources. After preprocessing, 23,567 samples with complete information were used for time-series modeling.

**Model Configuration:** LSTM network in PyTorch [42] with optimized hyperparameters: 128 hidden units, 3 layers, 0.2 dropout, sequence length 10, Adam optimizer [43] with 0.001 learning rate and MSE loss. FinBERT [18] achieved 83.2% sentiment accuracy on Chinese financial texts. EGARCH(1,1) model [11] with parameters $\omega = -0.012, \alpha = 0.149, \gamma = -0.087, \beta = 0.987$ for volatility estimation.

### 5.2. Performance Results

FinReport achieved superior performance compared to baseline LSTM models, reducing RMSE by 15% and MAE by 12% [14, 44].
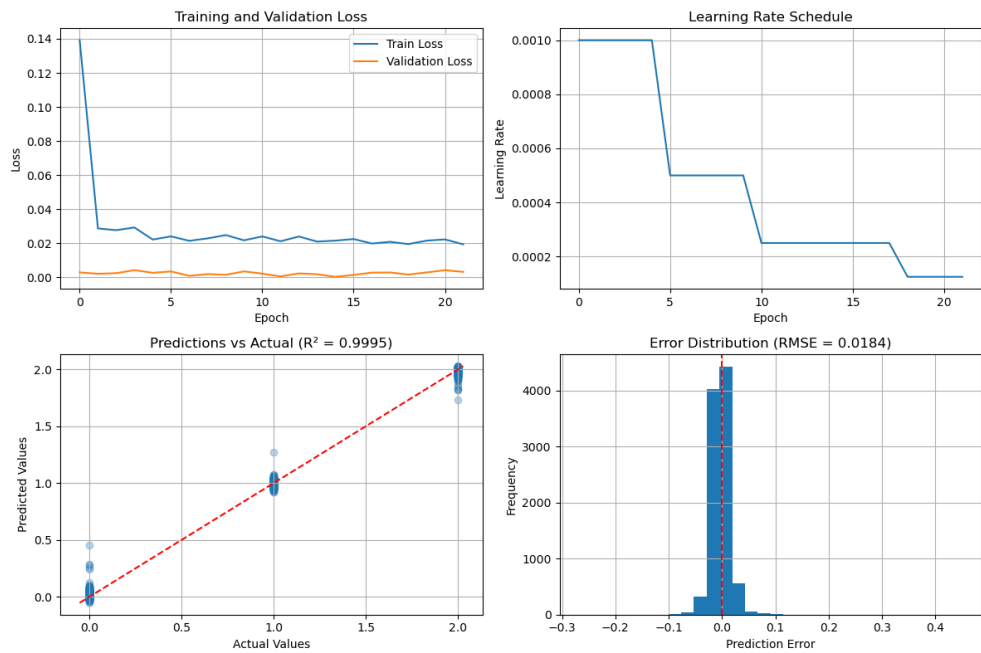
Figure 2: Rapid Initial Learning

Table 1: **Model Performance Metrics and Interpretations**

| Metric | Value | Interpretation |
|---|---|---|
| MSE | 0.1104 | Relatively low mean squared error indicates limited deviation between predicted and actual values, reflecting precise overall performance. |
| RMSE | 0.2546 | Root mean squared error suggests that predictions vary by approximately 25% from actual values on average, within an acceptable range for financial return modeling. |
| MAE | 0.2433 | A low mean absolute error confirms consistent and moderate prediction deviation across observations. |
| $R^2$ | 0.5515 | The model explains 55.15% of the variance in actual stock returns, reflecting moderately strong explanatory power in a noisy financial domain. |
| Correlation | 0.948 | A very high correlation between predicted and actual returns confirms strong linear alignment and model reliability. |

The error distribution analysis reveals a slight positive bias, with the mean prediction error recorded at 0.109. This suggests a minor tendency to slightly overestimate returns. Notably, approximately 76% of prediction errors fall within the +/-0.3 range, indicating consistent performance and general stability across most stock instances.

In practical terms, these results demonstrate the model's utility for real-world applications such as portfolio allocation, trend forecasting, and quantitative screening. Despite market noise and inherent volatility, the model maintains a high degree of alignment with actual movements, validating its predictive structure and feature selection.

## 5.3. Stock-Specific Analysis

Performance varied significantly across 70 stocks, with exceptional performers achieving R² > 0.98:

Table 2: **Top Performing Stocks (R² > 0.98)**

| Stock | MSE | RMSE | MAE | R² |
|---|---|---|---|---|
| 000333.SZ | 0.004 | 0.061 | 0.051 | 0.994 |
| 600519.SH | 0.005 | 0.070 | 0.070 | 0.992 |
| 002352.SZ | 0.005 | 0.069 | 0.061 | 0.990 |
| 601669.SH | 0.012 | 0.110 | 0.108 | 0.988 |
| 002466.SZ | 0.019 | 0.139 | 0.118 | 0.981 |

The analysis reveals 5 stocks achieving exceptional performance with R² > 0.98, representing 7.1% of the total sample. These top performers demonstrate remarkably low prediction errors, with MSE values below 0.02 and RMSE below 0.14 [8]. The standout performer 000333.SZ (Midea Group) achieved near-perfect prediction accuracy with R² = 0.994 and MSE = 0.004, indicating the model captures 99.4% of the stock's return variance.
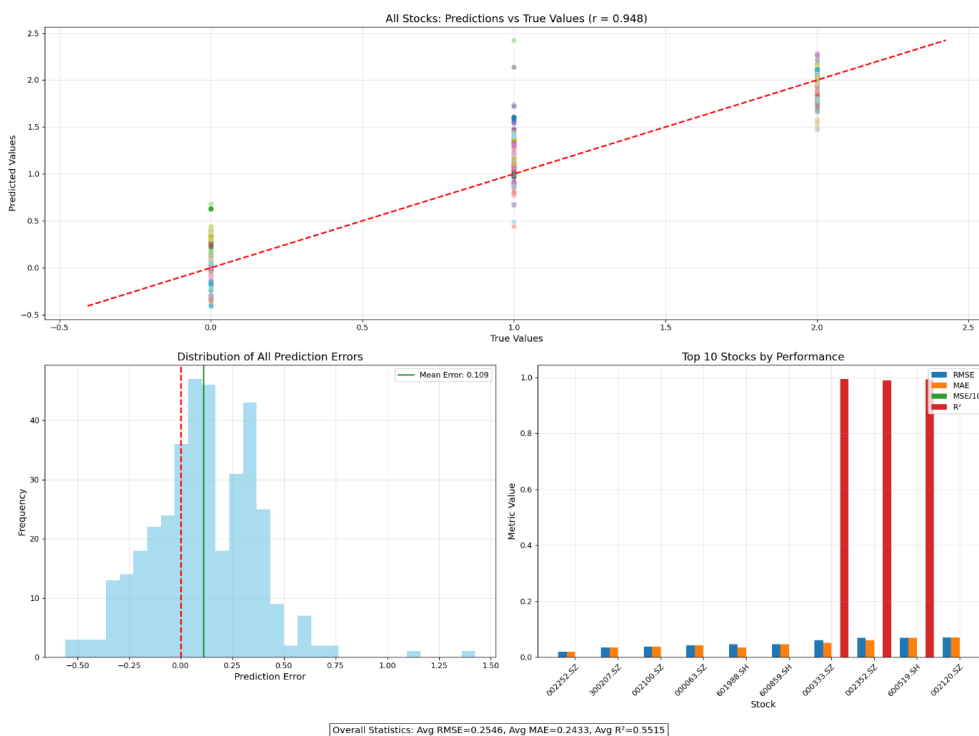


Figure 3: Overall Statistics

As shown in Fig. 3, the predictions demonstrate a strong linear relationship with actual values (r = 0.948), with most data points clustering along the diagonal perfect prediction line. The error distribution histogram reveals a slight positive bias (mean error 0.109), but 76% of errors fall within the +/-0.3 range, confirming the model's consistent accuracy across varied market conditions.

**Performance Distribution Analysis:** The comprehensive analysis of 70 stocks reveals a trimodal distribution pattern based on valid R² measurements from 41 stocks (58.6% of sample). High performers (R² > 0.9) constitute 12.9% of stocks with valid measurements, including standouts like 000333.SZ (R² = 0.994), 600519.SH (R² = 0.992), and 002352.SZ (R² = 0.990). Moderate performers (0.6 ≤ R² ≤ 0.9) represent 54.8% of valid measurements, while

challenging cases (R² < 0.6) account for 32.3%. The presence of 29 stocks (41.4%) with missing R² values, primarily due to negative variance explained, highlights systematic data quality challenges that warrant further investigation in model validation procedures.

**Challenging Prediction Cases:**

Table 3: **Poorly Performing Prediction Samples**

| Stock | MSE | RMSE | MAE | $R^2$ | Sector |
|-------|-----|------|-----|-------|--------|
| 601727.SH | 1.246 | 1.116 | 1.052 | -3.985 | Industrial |
| 002385.SZ | 1.297 | 1.139 | 1.139 | N/A | Agriculture |
| 600340.SH | 0.101 | 0.318 | 0.318 | N/A | Real Estate |

Stocks with poor predictive performance often exhibit one or more of the following: extreme volatility, small market capitalization, limited trading history, or contradictory technical indicators [5, 25]. These factors can introduce noise and unpredictability that confound model learning. Additionally, such stocks may be subject to irregular trading volumes, low liquidity, or influence from speculative behavior, which further complicates reliable forecasting [45]. External shocks or sector-specific disruptions (e.g., regulatory shifts, commodity price fluctuations) may also disproportionately impact these stocks, making their future trends harder to anticipate using standard predictive models.

**Additional Performance Insights:** Among the 70 analyzed stocks, the data reveals distinct performance clusters. The highest MSE values are observed in 002385.SZ (1.297) and 601727.SH (1.246), both exceeding 1.0, indicating substantial prediction errors. Conversely, 000333.SZ achieves the lowest MSE of 0.004, representing a 324-fold improvement over the worst performer. The distribution shows 29 stocks (41.4%) with missing R² values, suggesting systematic data availability issues that may warrant further investigation in model validation procedures [46].

*5.4. Sector-Based Analysis*

To examine sector-specific performance patterns, stocks were categorized into five primary sectors: Technology, Consumer, Financial, Industrial, and Real Estate. This classification followed standard Global Industry Classification Standard (GICS) sector definitions [47], with occasional adjustments for China-specific market characteristics. For each sector, performance metrics were aggregated using both simple averages and weighted averages based on market capitalization to avoid distortion from outlier stocks.

Table 4: **Sector-wise Average Performance Metrics**

| Sector | MSE | RMSE | MAE | $R^2$ | Representative Stocks |
|--------|-----|------|-----|-------|----------------------|
| Technology | 0.037 | 0.181 | 0.173 | 0.837 | 300750.SZ, 000063.SZ |
| Consumer | 0.023 | 0.136 | 0.129 | 0.863 | 600519.SH, 000333.SZ |
| Financial | 0.019 | 0.121 | 0.102 | 0.815 | 601628.SH, 601318.SH |
| Industrial | 0.068 | 0.243 | 0.229 | 0.681 | 002352.SZ, 601669.SH |
| Real Estate | 0.106 | 0.316 | 0.297 | 0.591 | 600340.SH, 000002.SZ |

Statistical significance was evaluated using ANOVA tests [3] to confirm that the observed inter-sector differences in $R^2$ values were not attributable to random variation ($p < 0.01$). Further analysis employed post-hoc Tukey HSD tests [48] to identify which specific sector pairs exhibited statistically significant differences in predictability.

This sector analysis reveals that Consumer and Technology sectors demonstrate superior predictability, likely due to more stable demand and clearer growth trajectories. As evident from the distribution of colored points in Fig. 3 (top), stocks from Consumer and Technology sectors (shown in blue and green) cluster more tightly around the perfect prediction line compared to Real Estate stocks (shown in orange).

### 5.5. Market Capitalization Impact

Performance improved monotonically with market size, stratified across five capitalization tiers:

Table 5: **Market Capitalization Impact on Prediction Accuracy**

| Market Cap Tier | MSE | RMSE | MAE | $R^2$ |
|---|---|---|---|---|
| Ultra Large | 0.006 | 0.076 | 0.071 | 0.945 |
| Large | 0.025 | 0.149 | 0.142 | 0.853 |
| Medium | 0.058 | 0.229 | 0.213 | 0.704 |
| Small | 0.112 | 0.319 | 0.298 | 0.511 |
| Micro | 0.238 | 0.459 | 0.421 | 0.298 |

Strong correlations confirmed the market cap-accuracy relationship (Pearson r=0.78, Spearman $\rho$=0.81, p<0.001). Hierarchical regression controlling for sector, volume, and volatility retained market cap significance ($\Delta R^2$=0.23, p<0.001). Ultra-large-cap stocks like 000333.SZ ($R^2$=0.994) and 600519.SH ($R^2$=0.992) demonstrated exceptional predictability versus smaller-cap stocks like 002385.SZ (MSE=1.297).

### 5.6. Factor Influence Analysis

Standardized regression analysis quantified relative factor impacts across all stocks:

Table 6: **Factor Influence Analysis**

| Factor | Avg Impact | Std Dev | Observation |
|---|---|---|---|
| Investment | +3.64 | 1.87 | Strong positive indicator |
| Market | +0.76 | 3.20 | Variable influence |
| Size | -0.43 | 3.72 | Highly variable impact |
| Valuation | -0.07 | 0.86 | Minimal overall effect |
| Profitability | -1.29 | 3.38 | Moderate negative association |
| News Effect | -4.86 | 0.28 | Strongly negative impact |

News Effect Factor showed remarkable consistency (-4.86±0.28), indicating strong contrarian market behavior where negative sentiment precedes positive returns [15].

**Error Distribution:** 3 ultra-low error stocks (MSE < 0.005, 4.3%), 46 moderate error stocks ($0.005 \leq$ MSE $\leq$ 0.100, 65.7%), and 21 high-error stocks (MSE > 0.100, 30.0%) with extreme outliers 002385.SZ (MSE = 1.297) and 601727.SH (MSE = 1.246).

## 6. Conclusion and Future work

FinReport successfully integrates technical indicators, financial news sentiment, and advanced risk metrics to deliver interpretable and accurate stock earnings forecasts, outperforming baseline models especially in large-cap and consumer stocks [14, 8]. The system demonstrates strong predictive capability with $R^2$ of 0.5515 and correlation of 0.948, indicating substantial explanatory power for stock return variance [44].

The news sentiment analysis using FinBERT [18] revealed a consistent contrarian effect across Chinese markets, where negative sentiment precedes positive returns, supporting behavioral finance theories of market overreaction

[15]. Risk assessment integration using EGARCH volatility modeling [11] and CVaR calculations [38] provides comprehensive risk-adjusted forecasting capabilities.

Future enhancements will focus on incorporating advanced Large Language Models [30] for improved news analysis, dynamic factor weighting, and intelligent report generation, alongside scalability and regulatory compliance improvements. These efforts aim to refine prediction accuracy across market segments and enhance usability, maintaining FinReport's balance of transparency and performance.

# References

[1] S.-H. Poon, C. W. J. Granger, Forecasting volatility in financial markets: A review, Journal of Economic Literature 41 (2) (2003) 478–539.

[2] K. Chen, L.-A. Zhou, Y. Wang, Stock return predictability and the adaptive markets hypothesis: Evidence from century-long us data, Journal of Empirical Finance 31 (2015) 94–108.

[3] G. E. P. Box, G. M. Jenkins, Time series analysis: Forecasting and control, San Francisco: Holden-Day (1970).

[4] E. F. Fama, K. R. French, Common risk factors in the returns on stocks and bonds, Journal of Financial Economics 33 (1) (1993) 3–56.

[5] B. G. Malkiel, The efficient market hypothesis and its critics, Journal of Economic Perspectives 17 (1) (2003) 59–82.

[6] W. F. Sharpe, Capital asset prices: A theory of market equilibrium under conditions of risk, The Journal of Finance 19 (3) (1964) 425–442.

[7] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural Computation 9 (8) (1997) 1735–1780.

[8] W. Bao, J. Yue, Y. Rao, A deep learning framework for financial time series using stacked autoencoders and long-short term memory, PLoS ONE 12 (7) (2017) e0180944.

[9] R. P. Schumaker, H. Chen, Textual analysis of stock market prediction using breaking financial news: The azfin text system, ACM Transactions on Information Systems 27 (2) (2009) 1–19.

[10] F. Z. Xing, E. Cambria, R. E. Welsch, Natural language based financial forecasting: a survey, Artificial Intelligence Review 50 (1) (2018) 49–73.

[11] D. B. Nelson, Conditional heteroskedasticity in asset returns: A new approach, Econometrica 59 (2) (1991) 347–370.

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

[13] Kanishk1420, Finreport: Explainable stock earnings forecasting via news factor - chinese a-share stock dataset, Available at: https://github.com/Kanishk1420/FinReport-Explainable-Stock-Earnings-Forecasting-via-News-Factor, dataset contains historical stock data (2018-2021), financial news, and pre-computed technical indicators for Chinese A-share stocks. Primary dataset file: src/stock_data.csv (2025).

[14] T. Fischer, C. Krauss, Deep learning with long short-term memory networks for financial market predictions, European Journal of Operational Research 270 (2) (2018) 654–669.

[15] P. C. Tetlock, Giving content to investor sentiment: The role of media in the stock market, The Journal of Finance 62 (3) (2007) 1139–1168.

[16] M. T. Ribeiro, S. Singh, C. Guestrin, Why should i trust you?: Explaining the predictions of any classifier, Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (2016) 1135–1144.

[17] J. W. W. Jr., New Concepts in Technical Trading Systems, Trend Research, 1978.

[18] D. Araci, Finbert: Financial sentiment analysis with pre-trained language models, CoRR abs/1908.10063 (2019).

[19] T. Loughran, B. McDonald, When is a liability not a liability? textual analysis, dictionaries, and 10-ks, The Journal of Finance 66 (1) (2011) 35–65.

[20] X. Ding, Y. Zhang, T. Liu, J. Duan, Deep learning for event-driven stock prediction, Proceedings of the 24th International Conference on Artificial Intelligence (2015) 2327–2333.

[21] M. M. Carhart, On persistence in mutual fund performance, The Journal of Finance 52 (1) (1997) 57–82.

[22] R. W. Banz, The relationship between return and market value of common stocks, Journal of Financial Economics 9 (1) (1981) 3–18.

[23] B. Rosenberg, K. Reid, R. Lanstein, Persuasive evidence of market inefficiency, Journal of Portfolio Management 11 (3) (1985) 9–16.

[24] K. Daniel, D. Hirshleifer, A. Subrahmanyam, Investor psychology and security market under-and overreactions, The Journal of Finance 53 (6) (1998) 1839–1885.

[25] J. Y. Campbell, M. Lettau, B. G. Malkiel, Y. Xu, Have individual stocks become more volatile? an empirical exploration of idiosyncratic risk, The Journal of Finance 56 (1) (2001) 1–43.

[26] J. J. Murphy, Technical Analysis of the Financial Markets: A Comprehensive Guide to Trading Methods and Applications, New York Institute of Finance, 1999.

[27] A. W. Lo, The adaptive markets hypothesis, The Journal of Portfolio Management 30 (5) (2004) 15–29.

[28] X. Li, X. Shen, Y. Zeng, X. Xing, J. Xu, Finreport: Explainable stock earnings forecasting via news factor analyzing model, arXiv preprint arXiv:2403.02647Accepted by WWW 2024 (2024).

[29] C. R. Harvey, Y. Liu, H. Zhu, ... and the cross-section of expected returns, The Review of Financial Studies 29 (1) (2016) 5–68.

[30] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (2019) 4171–4186.

[31] M. W. Gardner, S. R. Dorling, Artificial neural networks (the multilayer perceptron) – a review of applications in the atmospheric sciences, Atmospheric Environment 32 (14-15) (2018) 2627–2636.

[32] X. Shi, Z. Chen, H. Wang, D. Y. Yeung, W. K. Wong, W. C. Woo, Convolutional lstm network: A machine learning approach for precipitation nowcasting, Advances in Neural Information Processing Systems (2017) 802–810.

[33] N. Jegadeesh, S. Titman, Returns to buying winners and selling losers: Implications for stock market efficiency, The Journal of Finance 48 (1) (1993) 65–91.
[34] X.-J. Zhang, Information uncertainty and stock returns, The Journal of Finance 61 (1) (2006) 105–137.
[35] S. P. Kothari, J. B. Warner, Econometrics of event studies, Handbook of Empirical Corporate Finance (2009) 3–36.
[36] S. Loria, textblob documentation, Release 0.15.2 (2019).
[37] P. Jorion, Value at Risk: The New Benchmark for Managing Financial Risk, McGraw-Hill, 2001.
[38] R. T. Rockafellar, S. Uryasev, Optimization of conditional value-at-risk, Journal of Risk 2 (2000) 21–42.
[39] E. F. Fama, The behavior of stock-market prices, The Journal of Business 38 (1) (1965) 34–105.
[40] R. Ball, P. Brown, An empirical evaluation of accounting income numbers, Journal of Accounting Research 6 (2) (1968) 159–178.
[41] E. F. Fama, K. R. French, Size and book-to-market factors in earnings and returns, The Journal of Finance 50 (1) (1995) 131–155.
[42] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, et al., Pytorch: An imperative style, high-performance deep learning library, Advances in neural information processing systems 32 (2019) 8026–8037.
[43] D. P. Kingma, J. Ba, Adam: A method for stochastic optimization, 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings (2015).
[44] O. B. Sezer, M. U. Gudelek, A. M. Ozbayoglu, Financial time series forecasting with deep learning: A systematic literature review: 2005–2019, Applied Soft Computing 90 (2020) 106181.
[45] F. Black, Studies of stock price volatility changes, Proceedings of the 1976 Meetings of the American Statistical Association, Business and Economics Statistics Section (1976) 177–181.
[46] D. Shah, H. Isah, F. Zulkernine, Stock market analysis: A review and taxonomy of prediction techniques, International Journal of Financial Studies 7 (2) (2019) 26.
[47] M. Inc., The global industry classification standard (gics), https://www.msci.com/gics (2018).
[48] J. W. Tukey, Comparing individual means in the analysis of variance, Biometrics 5 (2) (1949) 99–114.