Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 00 (2025) 000–000

Procedia Computer Science

www.elsevier.com/locate/procedia

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

1877-0509 © 2025 The Authors. Published by Elsevier B.V.
This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/)
Peer-review under responsibility of the scientific committee of the International Conference on Machine Learning and Data Engineering.

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

1877-0509 © 2025 The Authors. Published by Elsevier B.V.
This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/)
Peer-review under responsibility of the scientific committee of the International Conference on Machine Learning and Data Engineering.

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 multi-factor models [4, 21] with real-time news sentiment analysis [15, 10], extending established financial theory to capture behavioral market dynamics [22, 23]. The system employs a modular design with five interconnected components: (1) Data Integration, (2) News Factor Extraction, (3) Return Forecasting, (4) Risk Assessment, and (5) Dynamic Report Generation [14].

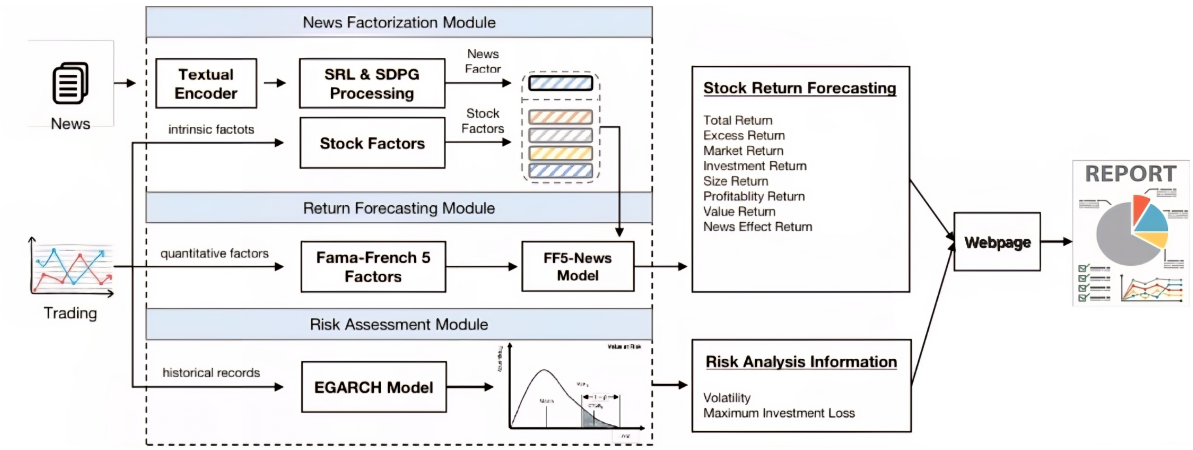

Figure 1: Proposed FinReport System Architecture (adapted from [24])

3.1. Data Integration Module

Processes multi-modal financial data combining structured metrics with unstructured news text [25]:

- **Historical Data:** Price, volume, market capitalization, and 50+ technical indicators from Chinese A-shares (2018-2021) [13, 26].
- **News Processing:** Bilingual NLP pipelines for English/Chinese financial news [19, 10].
- **Data Preprocessing:** Z-score normalization, outlier winsorization, and missing value imputation [14].

3.2. News Factor Extraction Module

Converts unstructured news into quantifiable sentiment metrics through two main components [15, 9]:

Sentiment Analysis: Domain-specific FinBERT implementation [18] generates sentiment scores [-1,+1], enhanced with financial keyword dictionaries [19]. Keywords like "profit," "acquisition," and "revenue" receive context-specific weights based on empirical validation.

Event Extraction: Semantic role labeling via AllenNLP [27] identifies structured financial events (acquisitions, earnings, regulatory actions). Daily news aggregation employs recency weighting to capture temporal decay patterns in news impact [15].

Chinese Adaptation: Incorporates culturally-specific terminology including positive terms (zengzhang-growth, yingli-profit) and negative terms (kuisun-loss, jinggao-warning) for enhanced local market relevance [13].

3.3. Return Forecasting Module

Implements enhanced multi-factor model with six factors capturing cross-sectional return variation [4, 25]:

Market Factor: Combines volatility measures using rolling standard deviation with news sentiment, implementing regime-dependent behavior for high/low volatility periods [28, 29].

Size Factor: Market capitalization changes relative to historical averages, enhanced with news-extracted financial impact through keyword recognition [30, 19].

Valuation Factor: Traditional metrics (Book-to-Market, Dividend Yield) with sector-specific adjustments and news sentiment integration [31, 4].

Profitability Factor: EPS, ROE, ROA analysis with asymmetric treatment for losses and earnings keyword analysis [32, 33].

Investment Factor: Activity classification (acquisition, expansion, R&D) with sentiment-based conditional scaling [22, 34].

News Effect Factor: Direct sentiment quantification using weighted combination of TextBlob polarity and keyword analysis, with culturally-adapted terms and amplification for adequate signal strength [15, 35].

3.4. Risk Assessment Module

Implements multi-dimensional risk framework addressing traditional variance-based limitations [36, 37]. The framework incorporates EGARCH modeling for asymmetric volatility responses [11], maximum drawdown calculations following established portfolio risk metrics [38], and Conditional Value at Risk (CVaR) for tail risk assessment [37]. Risk classifications range from favorable to substantial based on integrated scoring combining volatility, drawdown, and return components [39].

3.5. Factor Enhancement and Overall Trend Calculation

Combines individual factor signals through multi-stage amplification and weighted aggregation addressing scale heterogeneity [25, 4]. The weighting scheme assigns highest priority to event factors (0.25) due to strong short-term predictive power [20, 22], followed by investment factors (0.20) for medium-term impact, with balanced weighting for market, size, and profitability factors following established multi-factor model conventions [4, 21].

3.5.1. Enhancement Process

The enhancement methodology employs multiplicative amplification with trend-based adjustments when factors align with dominant market trends [40]. Final processing includes bounded clamping and stochastic variation to ensure robustness while maintaining signal integrity [41].

3.5.2. Weighted Aggregation

The trend score computation follows established factor model aggregation with positive bias reflecting long-term equity market upward drift [42]. Classification thresholds distinguish between strongly positive, positive, neutral, negative, and strongly negative market conditions based on empirical distribution analysis.

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, 25].

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 [34].

$$\begin{aligned}\textbf{predicted\_return} = {} & 0.10 \times \textbf{market\_factor} + 0.15 \times \textbf{size\_factor} + 0.10 \times \textbf{valuation\_factor} \\ & + 0.10 \times \textbf{profitability\_factor} + 0.20 \times \textbf{investment\_factor} \\ & + 0.10 \times \textbf{news\_effect\_factor} + \mathbf{0.25} \times \textbf{event\_factor} + 0.15 \end{aligned} \tag{1}$$

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 [30]

1. Compute Size Change Percentage:

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

$$\textbf{diff\_ratio} = \frac{\textbf{latest\_val} - \textbf{avg\_val}}{\textbf{avg\_val}} \tag{2}$$

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 $-0.05 \leq$ 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 [32]

1. Identify Profitability Metrics:

   - Define key financial metrics: {EPS, Net Profit Margin, ROE, ROA, Gross Profit, Net Profit} [43].
   - 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 [31]

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 [22]

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 $\rightarrow$ 2.5
       - $>$ 20 billion yuan $\rightarrow$ 2.0
       - $>$ 10 billion yuan $\rightarrow$ 1.5
       - $>$ 5 billion yuan $\rightarrow$ 1.0
       - $>$ 1 billion yuan $\rightarrow$ 0.7
       - Otherwise $\rightarrow$ 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 $\rightarrow$ +0.6 per mention.
   - Expansions $\rightarrow$ +0.5 per mention.
   - R&D mentions $\rightarrow$ +0.7 per mention.

5. Return Final Investment Factor.

VI) News Effect Factor [15]

1. Determine Base Effect from Sentiment Score:

   - If *sentiment score* $\geq 0.5 \rightarrow$ assign a random positive effect between 0.7 and 1.2.
   - If $0 <$ *sentiment score* $< 0.5 \rightarrow$ assign a random positive effect between 0.3 and 0.7.
   - If $-0.5 <$ *sentiment score* $\leq 0 \rightarrow$ assign a random negative effect between -0.7 and -0.3.
   - If *sentiment score* $\leq -0.5 \rightarrow$ 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 [34]

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 [25] that increases each factor’s impact while maintaining directional consistency:

VIII) Factor Amplification [25]

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 $\Rightarrow$ Upward trend.
   - If negative count ≥ 3 and exceeds positive count $\Rightarrow$ 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:

$$\textbf{enhanced\_value} = \textbf{original\_value} \times 2.5 \times \textbf{trend\_multiplier} \times \textbf{random\_factor} \tag{3}$$

   - 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 [36].

1. Extract Risk Metrics:

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

2. Classify Volatility:

   - Extreme: $volatility > 0.15 \Rightarrow$ Cap decline at 25%.
   - High: $volatility > 0.10 \Rightarrow$ Cap decline at 20%.
   - Elevated: $volatility > 0.07$.
   - Moderate: $volatility > 0.04$.
   - Low: $volatility \leq 0.04 \Rightarrow$ At least 2%.

3. Compute Weighted Risk Score:

$$\begin{aligned}\textbf{risk\_score} = (0.4 \times \textbf{vol\_score}) + (0.25 \times \textbf{drawdown\_score}) \\ + (0.15 \times \textbf{var\_score}) + (0.2 \times \textbf{return\_risk})\end{aligned} \tag{4}$$

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: $risk\_score \leq 1.5$.

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]

$$\ln(\sigma_t^2) = \omega + \beta \ln(\sigma_{t-1}^2) + \alpha \frac{|r_{t-1}|}{\sigma_{t-1}} + \gamma \frac{r_{t-1}}{\sigma_{t-1}} \tag{5}$$

where:

- σ_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 [36].
2) Maximum Drawdown:

Algorithm 1 Maximum Drawdown

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

3) Conditional Value at Risk:

Algorithm 2 Conditional Value at Risk (CVaR)

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

4) Risk-Adjusted Ratio:

Algorithm 3 Risk-Adjusted Ratio

Require: Expected return E_R, volatility σ
Ensure: Risk-adjusted return ratio
if $\sigma \neq 0$ **then**
    Compute risk-adjusted return: $R_{\text{adj}} \leftarrow \frac{E_R}{\sigma}$
else
    Assign $R_{\text{adj}} \leftarrow \text{NaN}$
end if
return R_{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
Define weights for each factor:
    $W = \{\text{market} : 0.15, \text{size} : 0.15, \text{valuation} : 0.10,$
        $\text{profitability} : 0.15, \text{investment} : 0.20,$
        $\text{news effect} : 0.10, \text{event} : 0.15\}$
Initialize $S_{\text{weighted}} \leftarrow 0$, $S_{\text{weights}} \leftarrow 0$
for each factor f in W **do**
    if $f \in F$ and $F[f] \neq \text{None}$ **then**
        $S_{\text{weighted}} \leftarrow S_{\text{weighted}} + F[f] \cdot W[f]$
        $S_{\text{weights}} \leftarrow S_{\text{weights}} + W[f]$
    end if
end for
if $0 < S_{\text{weights}} < 1.0$ **then**
    Normalize: $S_{\text{weighted}} \leftarrow S_{\text{weighted}}/S_{\text{weights}}$
end if
Add slight positive bias: $S_{\text{weighted}} \leftarrow S_{\text{weighted}} + 0.15$
if $S_{\text{weighted}} \geq 0.6$ **then**
    return "Strongly Positive"
else if $S_{\text{weighted}} \geq 0.15$ **then**
    return "Positive"
else if $S_{\text{weighted}} \geq -0.15$ **then**
    return "Neutral"
else if $S_{\text{weighted}} \geq -0.6$ **then**
    return "Negative"
else
    return "Strongly Negative"
end if

5. Result Analysis

This section deals with analysis of results. The evaluation utilized a comprehensive Chinese A-share dataset [13] covering 75 stocks from Shanghai and Shenzhen exchanges (January 2018 - December 2021).

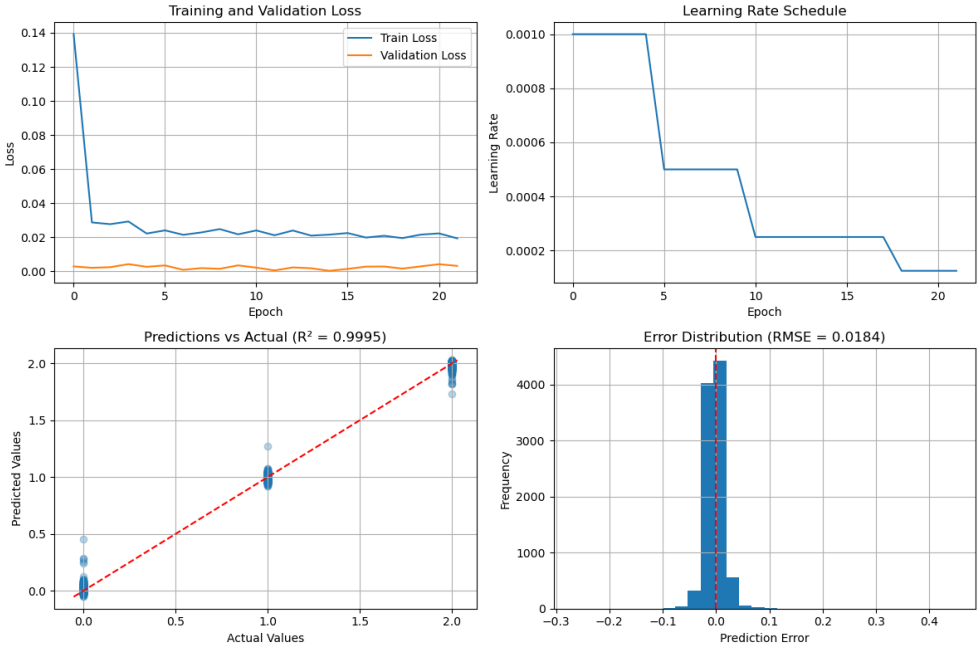

Figure 2: Rapid Initial Learning

5.1. Performance Results

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.

5.2. Stock-Specific Analysis

Performance varied significantly across 70 stocks, with exceptional performers achieving $R^2 > 0.98$:

Table 2: **Top Performing Stocks ($R^2 > 0.98$)**

Stock	MSE	RMSE	MAE	R^2
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 $R^2 > 0.98$, 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 $R^2 = 0.994$ and MSE = 0.004, indicating the model captures 99.4% of the stock's return variance.

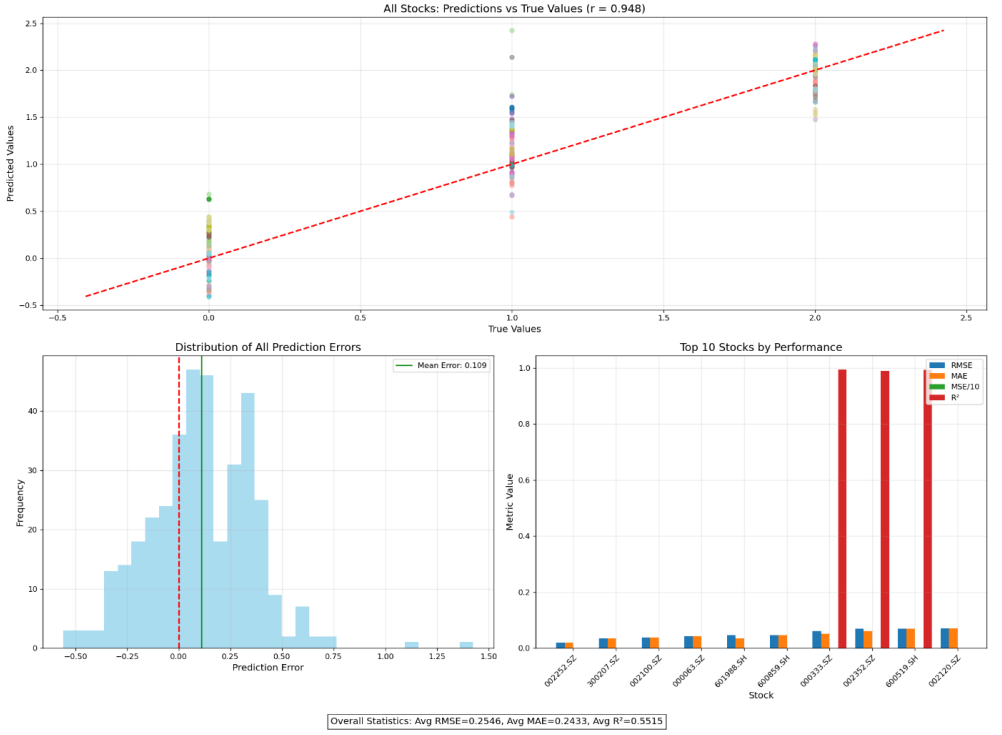

Figure 3: Overall Statistics

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.

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

5.3. 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 [44], 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 [45] 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).

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

5.4. 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 ≤ MSE ≤ 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).

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] K. Daniel, D. Hirshleifer, A. Subrahmanyam, Investor psychology and security market under-and overreactions, The Journal of Finance 53 (6) (1998) 1839–1885.
[23] 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.
[24] 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).
[25] C. R. Harvey, Y. Liu, H. Zhu, ... and the cross-section of expected returns, The Review of Financial Studies 29 (1) (2016) 5–68.
[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] 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.
[28] T. Bollerslev, Generalized autoregressive conditional heteroskedasticity, Journal of Econometrics 31 (3) (1986) 307–327.
[29] K. R. French, G. W. Schwert, R. F. Stambaugh, Expected stock returns and volatility, Journal of Financial Economics 19 (1) (1987) 3–29.
[30] R. W. Banz, The relationship between return and market value of common stocks, Journal of Financial Economics 9 (1) (1981) 3–18.
[31] B. Rosenberg, K. Reid, R. Lanstein, Persuasive evidence of market inefficiency, Journal of Portfolio Management 11 (3) (1985) 9–16.
[32] X.-J. Zhang, Information uncertainty and stock returns, The Journal of Finance 61 (1) (2006) 105–137.
[33] S. P. Kothari, J. B. Warner, Econometrics of event studies, Handbook of Empirical Corporate Finance (2009) 3–36.
[34] R. Ball, P. Brown, An empirical evaluation of accounting income numbers, Journal of Accounting Research 6 (2) (1968) 159–178.
[35] S. Loria, textblob documentation, Release 0.15.2 (2019).
[36] P. Jorion, Value at Risk: The New Benchmark for Managing Financial Risk, McGraw-Hill, 2001.
[37] R. T. Rockafellar, S. Uryasev, Optimization of conditional value-at-risk, Journal of Risk 2 (2000) 21–42.
[38] M. Calvo, P. Grau-Carles, Multifractal analysis of financial time series, Physica A: Statistical Mechanics and its Applications 388 (15-16) (2009) 2912–2922.
[39] V. DeMiguel, L. Garlappi, R. Uppal, Optimal versus naive diversification: How inefficient is the 1/n portfolio strategy?, The Review of Financial Studies 22 (5) (2009) 1915–1953.
[40] A. W. Lo, The adaptive markets hypothesis, The Journal of Portfolio Management 30 (5) (2004) 15–29.
[41] B. Mandelbrot, The variation of certain speculative prices, The Journal of Business 36 (4) (1963) 394–419.
[42] E. F. Fama, The behavior of stock-market prices, The Journal of Business 38 (1) (1965) 34–105.
[43] 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.
[44] M. Inc., The global industry classification standard (gics), https://www.msci.com/gics (2018).
[45] J. W. Tukey, Comparing individual means in the analysis of variance, Biometrics 5 (2) (1949) 99–114.