

Cryptocurrency Price Prediction with News Sentiment:

A Reproducible Research Report from `crypto-price-analysis-news-sentiment`

Generated from experiment artifacts (Dec 2025)

December 16, 2025

Abstract

Cryptocurrency markets react quickly to information. This project builds an end-to-end pipeline that collects cryptocurrency news via public RSS feeds and market data via CoinGecko, extracts sentiment signals from text (VADER and TextBlob), engineers sentiment and market features, and evaluates machine learning models for (i) return regression and (ii) direction classification at hourly resolution.

This report is grounded in the saved experiment artifacts under `research_output/crypto-sentiment_research_2025/` (timestamp 20251216_084725). The experiment contains 301 news articles fetched from public RSS feeds and 10,805 hourly price rows across five cryptocurrencies. Correlation analysis shows statistically significant but small associations between aggregated sentiment and next-step returns (e.g., Pearson $r = 0.1204$ for `sentiment_mean`, $p = 3.6 \times 10^{-36}$). For predicting hourly percentage returns (`target_return`), the best test-set model is Ridge regression (RMSE = 0.7899, MAE = 0.5188, $R^2 = 0.0074$), indicating limited explained variance for return prediction under this setup. For direction classification (`target_direction`), XGBoost achieves test accuracy = 0.554 and ROC-AUC = 0.574.

1 Introduction

News and narrative influence crypto markets, but whether news sentiment yields actionable predictive signal at high frequency remains empirically unclear. This project evaluates a practical pipeline: collect news and prices, compute sentiment, build features, and compare models for prediction.

Research questions.

1. Do aggregated news sentiment features correlate with future cryptocurrency returns at hourly granularity?
2. How well can standard ML models predict hourly returns (regression) and direction (classification)?
3. Which engineered features are most influential in fitted models?

2 Data and Sources

2.1 Assets

The pipeline tracks five CoinGecko assets configured in `src/config.py`: Bitcoin, Ethereum, Solana, Cardano, and Polkadot.

2.2 Market data (prices, volume, market cap)

Hourly market data are fetched using the CoinGecko API (via `pycoingecko`) [1, 2].

2.3 News data (RSS)

News is fetched from public RSS feeds using `feedparser`. In the saved experiment, the dominant sources are listed in `results/experiment_results.json`. The feeds include: U.Today [3], Decrypt [4], CryptoPotato [5], CoinTelegraph [6], CoinDesk [7], CryptoNews [8], AMBCrypto [9], BeInCrypto [10], and Bitcoin Magazine [11].

2.4 Dataset summary (from artifacts)

Descriptive statistics are exported by the research pipeline:

- News articles fetched: 301 (see `results/experiment_results.json`)
- Hourly market rows: 10,805 (see `tables/price_statistics.tex`)
- Engineered modeling rows (after merge/targets): 10,795

Table 1: Descriptive Statistics of Cryptocurrency Prices

	mean	median	std	min	max
price	21755.8536	184.7793	41904.1405	0.3722	120000.0
volume	21454939092.5963	6644130079.7827	28086510646.0254	79138635.8263	200968700000.0
market_cap	532910549911.7969	101300958310.4291	802598048339.0220	3223436946.1576	2507866740000.0

3 Methods

3.1 Text preprocessing

News entries combine title and summary fields and are cleaned using lowercasing, URL/HTML removal, tokenization, stopwords removal, and lemmatization (NLTK WordNet).

3.2 Sentiment extraction

Sentiment is computed with:

- VADER (compound and class probabilities) [12]
- TextBlob polarity and subjectivity [13]

3.3 Feature engineering and targets

The engineered dataset used by the experiment contains market features (returns, moving averages, volatility, momentum, volume changes), aggregated sentiment features (mean/std/min/max, polarity/subjectivity aggregates), and meta features (hour, article count). Targets include future price, percentage return, and direction:

$$\text{target_return} = \left(\frac{\text{target_price} - \text{price}}{\text{price}} \right) \times 100, \quad \text{target_direction} = \mathbb{I}(\text{target_return} > 0).$$

3.4 Experimental design

The research pipeline uses a 60/20/20 train/validation/test split with fixed random seed 42. Models evaluated include Linear, Ridge, Random Forest, XGBoost, and LightGBM for regression; and Logistic, Random Forest, XGBoost, LightGBM for classification.

4 Results

4.1 Sentiment distribution

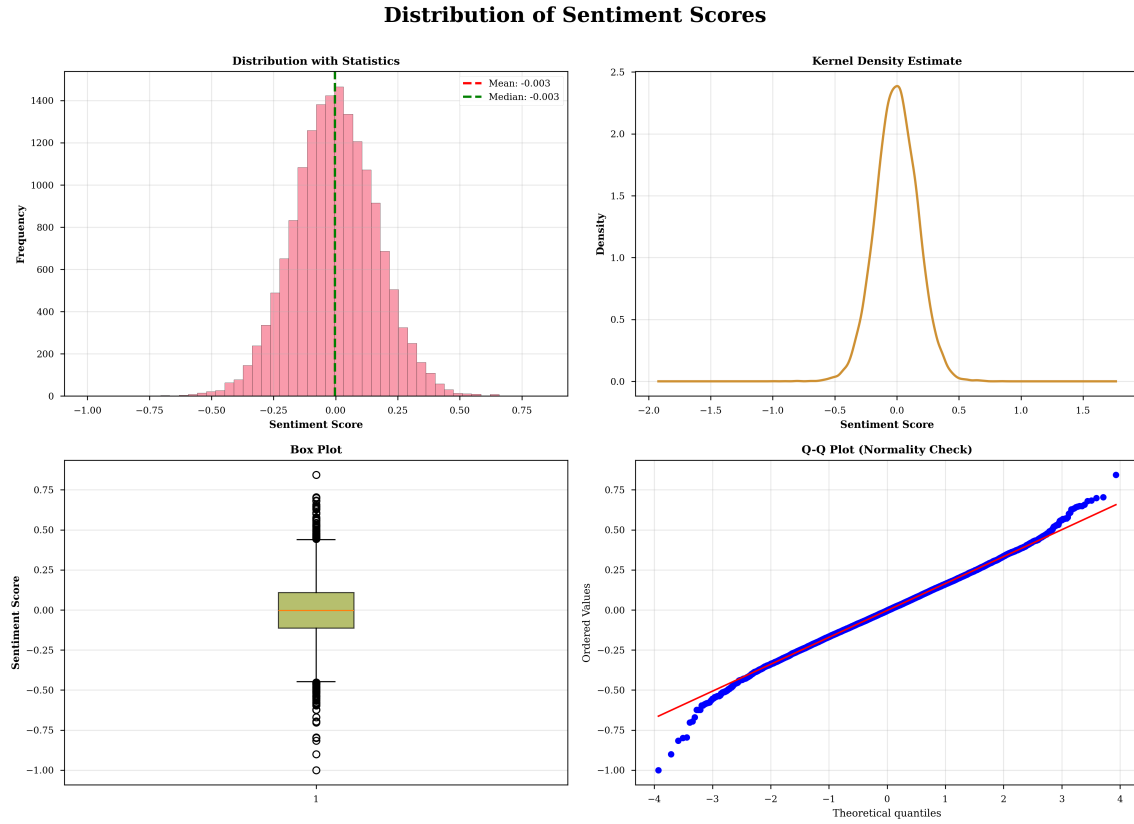


Figure 1: Sentiment distribution diagnostics (histogram, KDE, box plot, Q-Q plot).

4.2 Correlation analysis

Table 2: Correlation Analysis: Top Features vs Target Variable

	feature	pearson_corr	pearson_pvalue	pearson_significant	spearman_corr	spearman_pvalue
12	sentiment_mean	0.1204	0.0000	True	0.0812	0.0000
19	polarity_mean	0.1015	0.0000	True	0.0654	0.0000
17	negative_mean	-0.0863	0.0000	True	-0.0745	0.0000
14	sentiment_min	0.0841	0.0000	True	0.0320	0.0000
16	positive_mean	0.0770	0.0000	True	0.0648	0.0000
4	price_return	0.0695	0.0000	True	-0.0133	0.0000
15	sentiment_max	0.0447	0.0000	True	0.0171	0.0000
13	sentiment_std	-0.0367	0.0001	True	-0.0199	0.0000
18	neutral_mean	0.0182	0.0581	False	0.0212	0.0000
8	volatility_168h	0.0103	0.3046	False	0.0076	0.0000
1	market_cap	0.0071	0.4583	False	0.0034	0.0000
6	price_ma_168h	0.0068	0.4955	False	0.0064	0.0000
11	momentum_24h	-0.0068	0.4805	False	-0.0373	0.0000
0	price	0.0068	0.4799	False	0.0035	0.0000
9	volume_ma_24h	0.0067	0.4894	False	0.0078	0.0000

Feature Correlation Matrix

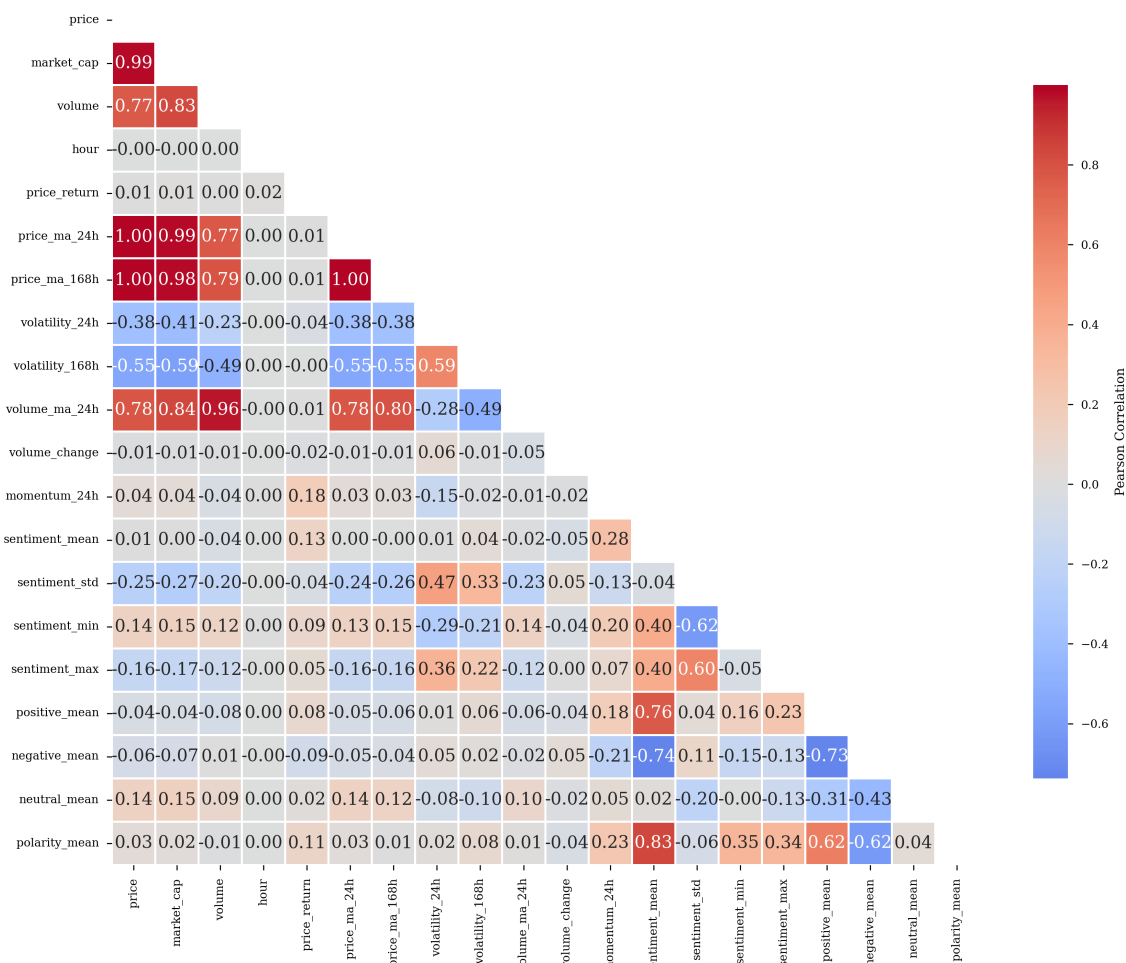


Figure 2: Correlation heatmap for selected engineered variables.

4.3 Regression performance (predicting target_return)

Table 3: Performance Comparison of Prediction Models

	model	mae	rmse	r2	mape	residual_mean	residual_std
1	ridge	0.5188	0.7899	0.0074	186.7192	-0.0104	0.7898
4	lightgbm	0.5155	0.7927	0.0004	207.5670	-0.0145	0.7926
0	linear	0.5107	0.7937	-0.0022	117.2563	-0.0112	0.7937
2	random_forest	0.5112	0.7998	-0.0176	167.5560	-0.0124	0.7997
3	xgboost	0.5210	0.8541	-0.1604	223.2444	-0.0004	0.8541

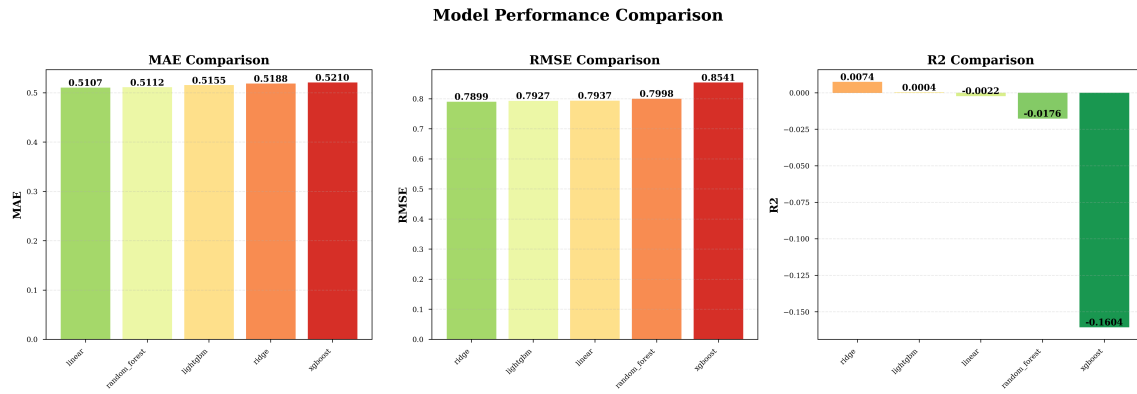


Figure 3: Model comparison across metrics (test split).

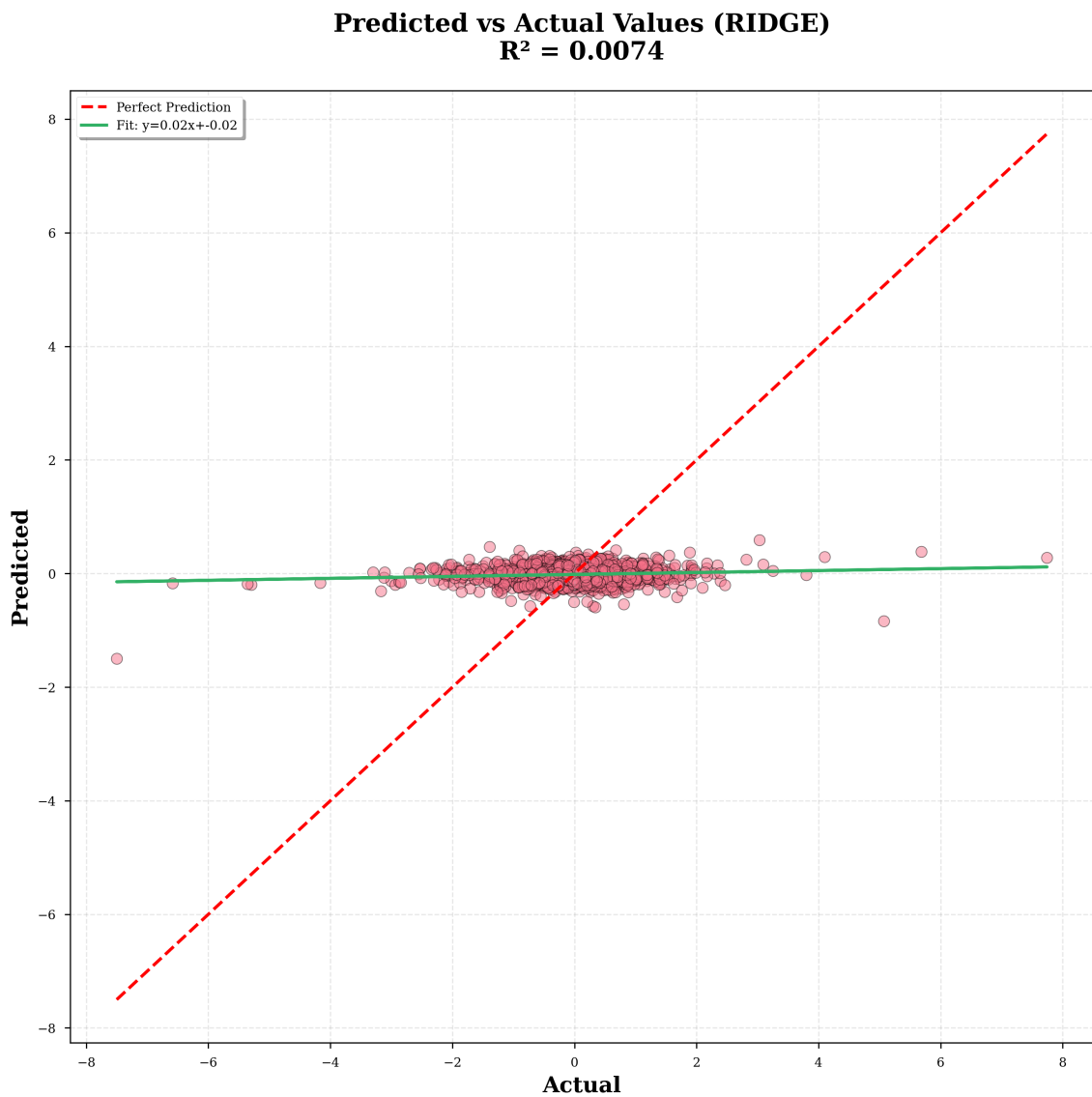


Figure 4: Predicted vs actual returns for the evaluated models.

Residual Analysis (RIDGE)

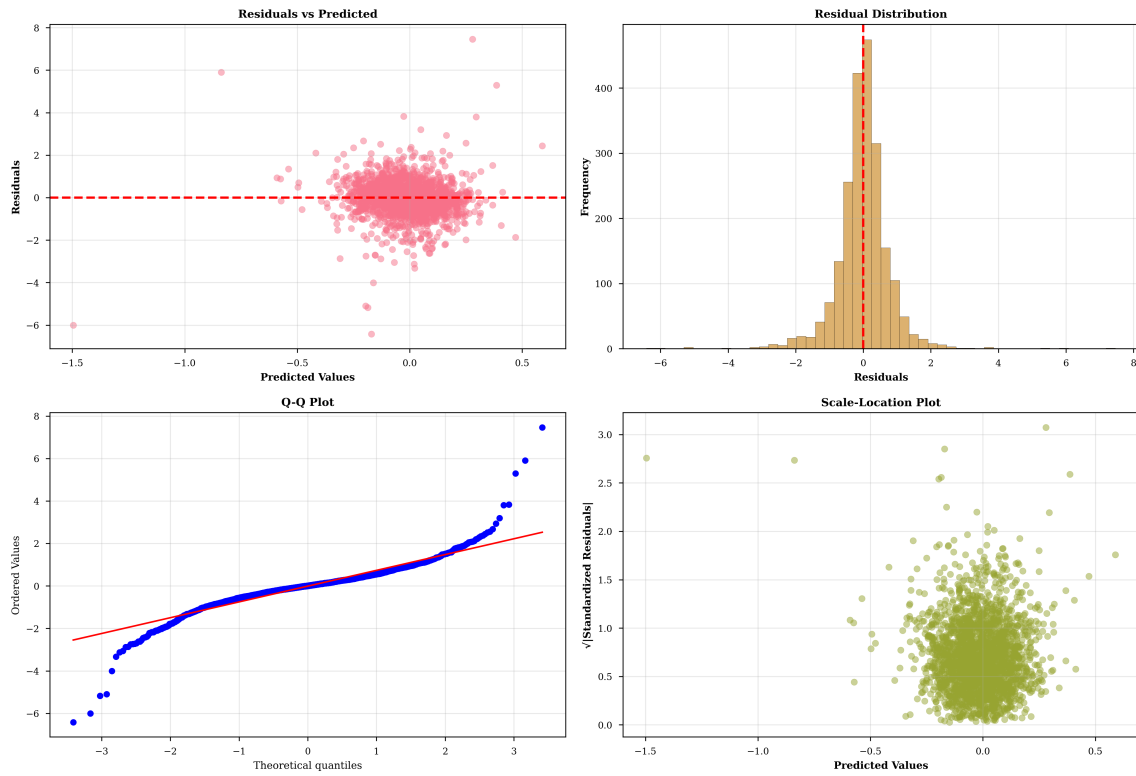


Figure 5: Residual analysis for the best test model in this run.

4.4 Feature importance

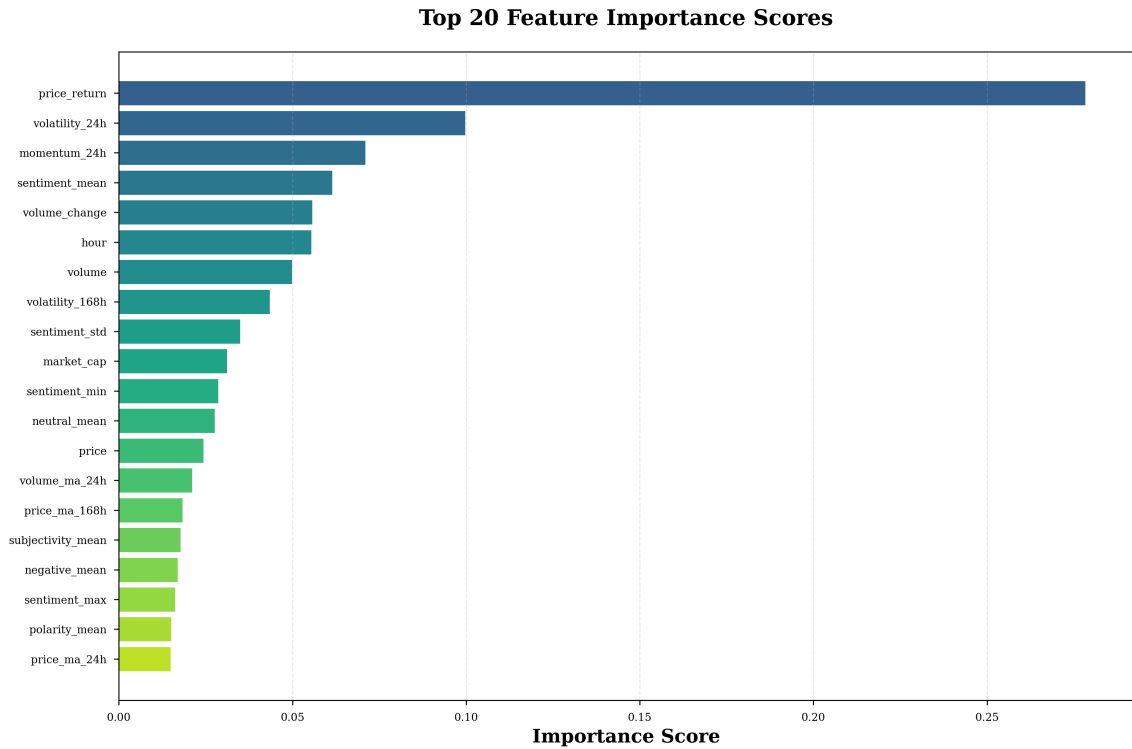


Figure 6: Top feature importances for the best model supporting importance in this run.

4.5 Direction classification (predicting target_direction)

The classification test-set results (from `tables/classification_comparison_test.csv`) indicate XGBoost as best performer with accuracy 0.554 and ROC-AUC 0.574.

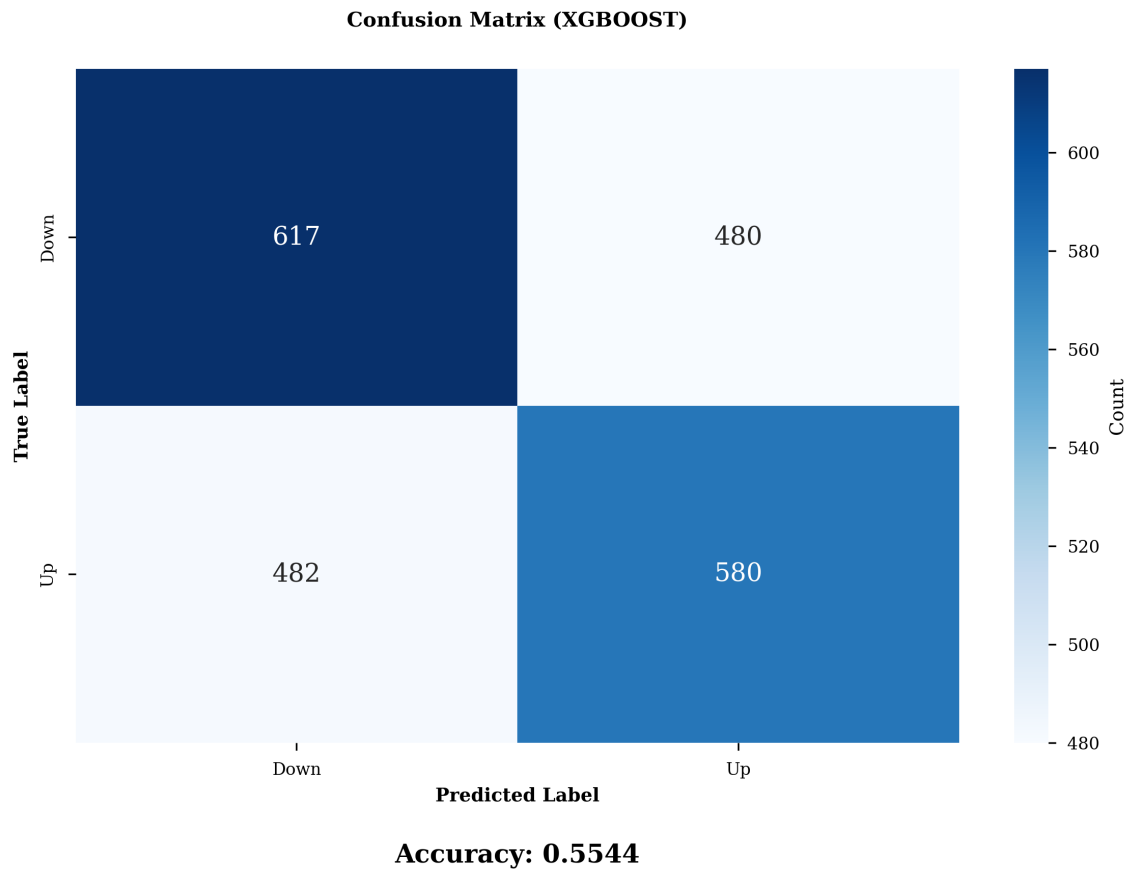


Figure 7: Confusion matrix for direction classification.

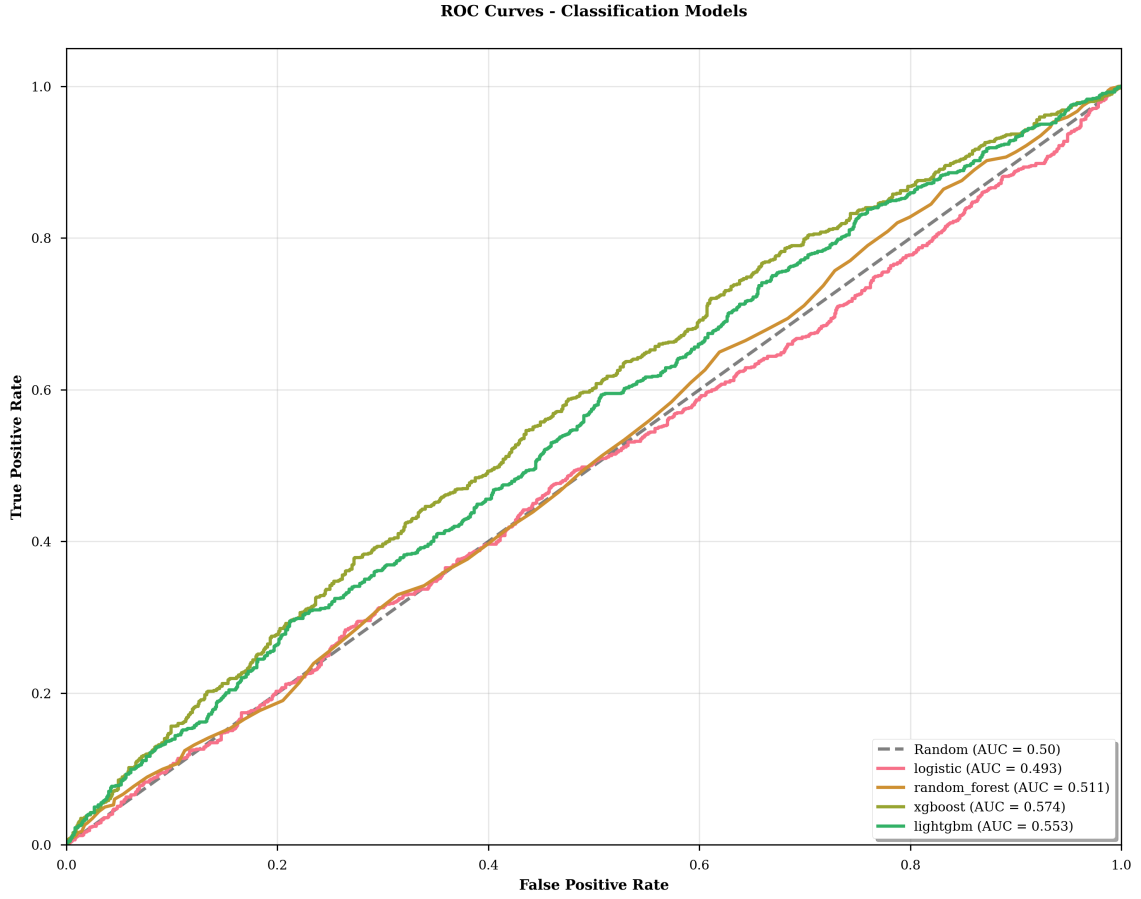


Figure 8: ROC curves for direction classification models.

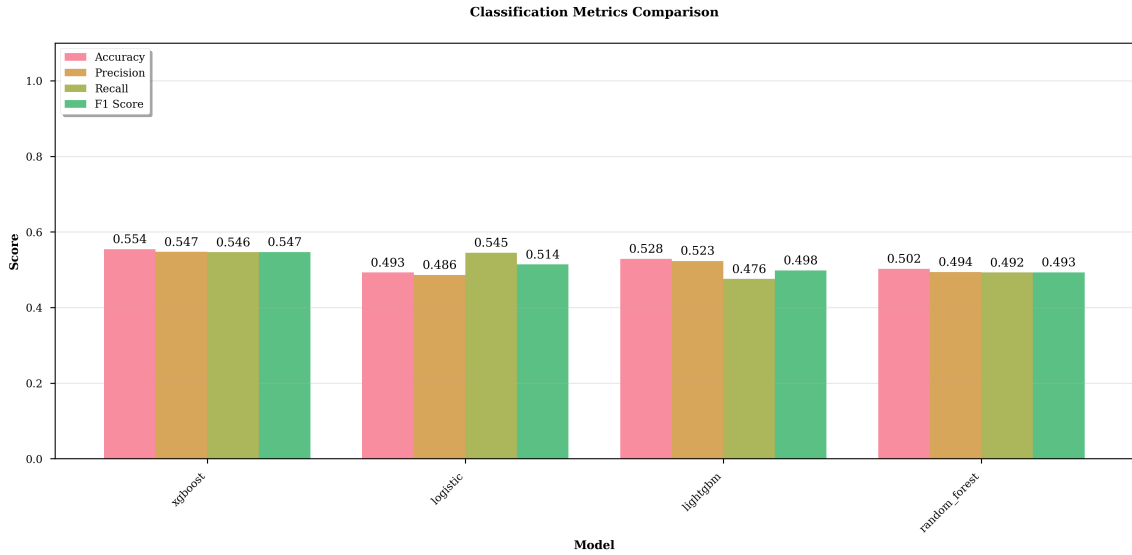


Figure 9: Classification metrics comparison across models.

5 Discussion

The experiment shows statistically significant but small sentiment–return correlations, while regression R^2 remains near zero on the test split, consistent with the difficulty of predicting

hourly returns. Classification performs modestly above chance ($\text{ROC-AUC} \approx 0.57$), suggesting weak separability for direction prediction under the current features and splitting protocol.

MAPE caveat. MAPE is unstable for return targets that cross or approach zero, and should not be interpreted as a primary metric for this task.

6 Limitations and Threats to Validity

- **Time-series leakage risk:** the experiment uses random splitting rather than strict walk-forward validation.
- **Timestamp alignment:** news publication timestamps are not consistently available in the saved artifact metadata, which can weaken causal alignment.
- **Non-stationarity:** cryptocurrency regimes change over time; results may not generalize.
- **Hypothesis test sensitivity:** very large effective sample sizes can make tiny effects statistically significant.

7 Reproducibility

To reproduce the experiment:

1. Install dependencies in `requirements.txt`.
2. Generate data (if needed): `python run_pipeline.py`.
3. Run the research pipeline: `python run_research_pipeline.py`.
4. Outputs appear under `research_output/crypto_sentiment_research_2025/`.

Acknowledgments

We thank open data providers and open-source maintainers. Market data are provided by CoinGecko [1]. News is collected from public RSS feeds cited in Section 2.

References

- [1] CoinGecko. *CoinGecko API*. <https://www.coingecko.com/en/api>. Accessed: 2025-12-16.
- [2] man-c. *pycoingecko: CoinGecko API wrapper for Python*. <https://github.com/man-c/pycoingecko>. Accessed: 2025-12-16.
- [3] U.Today. *RSS Feed*. <https://u.today/rss>. Accessed: 2025-12-16.
- [4] Decrypt. *RSS Feed*. <https://decrypt.co/feed>. Accessed: 2025-12-16.
- [5] CryptoPotato. *RSS Feed*. <https://cryptopotato.com/feed/>. Accessed: 2025-12-16.
- [6] CoinTelegraph. *RSS Feed*. <https://cointelegraph.com/rss>. Accessed: 2025-12-16.
- [7] CoinDesk. *RSS Feed*. <https://www.coindesk.com/arc/outboundfeeds/rss/>. Accessed: 2025-12-16.
- [8] CryptoNews. *RSS Feed*. <https://cryptonews.com/news/feed/>. Accessed: 2025-12-16.

- [9] AMBCrypto. *RSS Feed*. <https://ambcrypto.com/feed/>. Accessed: 2025-12-16.
- [10] BeInCrypto. *RSS Feed*. <https://beincrypto.com/feed/>. Accessed: 2025-12-16.
- [11] Bitcoin Magazine. *RSS Feed*. <https://bitcoinmagazine.com/.rss/full/>. Accessed: 2025-12-16.
- [12] C. J. Hutto and E. Gilbert. *VADER: A Parsimonious Rule-based Model for Sentiment Analysis of Social Media Text*. Proceedings of ICWSM (2014).
- [13] TextBlob. *TextBlob Documentation*. <https://textblob.readthedocs.io/>. Accessed: 2025-12-16.