A Comparison of VADER and Subjectivity-Enhanced VADER

1. Introduction

Financial news strongly influences investor behavior and market trends. Analyzing the sentiment of news headlines can provide useful signals for predicting stock market movements. Traditional methods like VADER measure positive, negative, and neutral sentiment but cannot distinguish between opinions and factual statements. To address this, we combine VADER sentiment scores with a subjectivity/objectivity score from a CNN trained on the Cornell Subjectivity Dataset. These features are integrated with market data to predict stock movements and forecast stock prices.

2. Lexicon-Based Sentiment Analysis

VADER (Valence Aware Dictionary for Sentiment Reasoning) is a lexicon-based method that assigns sentiment scores to text, including positive, negative, neutral, and a combined score. Although VADER effectively captures general sentiment, it cannot distinguish between subjective opinions and objective statements. As a result, factual sentences may receive misleading sentiment scores, adding noise to predictive models. To address this limitation, we introduce a subjectivity-aware feature, called SubjObj_Score, which is generated by a convolutional neural network trained on a dataset labeled for subjectivity detection.

3. Subjectivity/Objectivity Classification

3.1 Cornell Subjectivity Dataset

The Cornell Subjectivity Dataset, developed by Bo Pang and Lillian Lee, is commonly used for subjectivity classification in natural language processing. It consists of 10,000 sentences evenly divided between subjective and objective categories. The subjective subset includes 5,000 sentences from movie reviews that express opinions or emotions, while the objective subset contains 5,000 factual plot summaries. For supervised learning, subjective sentences are labeled as 0 and objective sentences as 1. The sentences and labels are combined into a dataset that serves as the basis for training the model.

3.2 Word Embedding Using GloVe

Neural networks require numerical input, so each sentence was converted into a vector using pre-trained GloVe word embeddings. GloVe (Global Vectors) represents words as dense vectors that capture their semantic relationships. In this study, each word was mapped to a 300-dimensional vector from the glove.6B.300d dataset. Sentences were tokenized and either padded or truncated to a fixed length of 40 tokens. Words not found in the GloVe vocabulary were assigned random vectors. This preprocessing ensured that all sentences had consistent numerical representations for input into the neural network.

3.3 1D Convolutional Neural Network (CNN) Architecture

A one-dimensional convolutional neural network (CNN) was developed in PyTorch to classify sentences as either subjective or objective. The network included three convolutional layers with kernel sizes of 3, 4, and 5, each containing 100 filters, designed to capture patterns of consecutive words (n-grams). ReLU activation was applied to introduce non-linearity, followed by max-pooling layers to extract the most important features from each convolution. A dropout rate of 0.5 was used to reduce overfitting, and the final fully connected layer employed a sigmoid activation function for binary classification. The dataset was divided into training (80%), validation (10%), and test (10%) sets to ensure unbiased evaluation. The trained CNN generates a subjectivity score (SubjObj_Score) for each headline, which is then incorporated as an additional feature in stock market prediction models.

4. Stock Market Movement Prediction (Classification)

For each trading day, daily sentiment features were created by averaging the SubjObj_Scores from the top 25 news headlines. These sentiment features were combined with financial indicators, including Open, High, Low, Close, Volume, and Adjusted Close prices. The complete feature set used for predicting stock market movements included the VADER sentiment scores (compound, positive, negative, neutral) along with the stock market variables and the SubjObj_Score. Several machine learning models were then applied to classify whether the stock market would go up or down, including XGBoost, Random Forest, Logistic Regression, K-Nearest Neighbors, Decision Tree, and Gaussian Naive Bayes. Model performance was evaluated using standard metrics such as accuracy, precision, recall, F1-score, and confusion matrices.

Table 4.1. Classification performance using VADER sentiment analysis.

Model	Train Accuracy	Test Accuracy	Precision	Recall	F1
LogisticRegression	**0.8635**	**0.8783**	**0.8785**	**0.8783**	**0.8783**
LDA	**0.8585**	**0.8757**	**0.8757**	**0.8757**	**0.8756**
SVM	**0.8547**	**0.8624**	**0.8629**	**0.8624**	**0.8624**
RandomForest	**0.6913**	**0.5714**	**0.5761**	**0.5714**	**0.5607**
XGBoost	**0.8641**	**0.5714**	**0.5829**	**0.5714**	**0.5508**
NaiveBayes	**0.5798**	**0.5582**	**0.5700**	**0.5582**	**0.5316**
KNN	**0.6462**	**0.5370**	**0.5383**	**0.5370**	**0.5251**

The comparison of table 4.1 and 4.2 shows that incorporating CNN-based subjectivity features alongside VADER sentiment leads to a modest improvement in predictive performance for certain models. Logistic Regression benefits the most, with test accuracy increasing from 0.878 to 0.884, while SVM and Naive Bayes show slight gains. In contrast, LDA and KNN exhibit decreased performance, and ensemble methods such as Random Forest and XGBoost remain largely unchanged. These results indicate that adding subjectivity information enhances feature representation for some algorithms, particularly linear models, but its effect is model-dependent.

Table 4.2. Classification performance using VADER + CNN (Subjectivity) sentiment features.

Model	Train Accuracy	Test Accuracy	Precision	Recall	F1
LogisticRegression	**0.8635**	**0.8836**	**0.8838**	**0.8836**	**0.8836**
SVM	**0.8528**	**0.8571**	**0.8576**	**0.8571**	**0.8571**
LDA	**0.6293**	**0.6190**	**0.6191**	**0.6190**	**0.6186**
XGBoost	**0.6468**	**0.5741**	**0.5926**	**0.5741**	**0.5459**
RandomForest	**0.6969**	**0.5688**	**0.5744**	**0.5688**	**0.5560**
NaiveBayes	**0.5767**	**0.5635**	**0.5768**	**0.5635**	**0.5372**
KNN	**0.6374**	**0.4947**	**0.4929**	**0.4947**	**0.4890**

5. Stock Price Prediction (Regression)

In addition to predicting market direction, we also forecasted actual stock prices using a Long Short-Term Memory (LSTM) neural network. The model focused on predicting the Adjusted Close price of the Dow Jones Industrial Average (DJIA).

5.1 Dataset Preparation

The dataset combined stock market indicators with sentiment features, including VADER scores and the SubjObj_Score. The data were divided by date into a training set (before January 1, 2015) and a test set (after January 1, 2015). All features and the target variable were scaled using MinMaxScaler. For the LSTM model, sequences of 10 consecutive days were used as input, with the next day's Adjusted Close price as the target.

5.2 LSTM Model Architecture

The LSTM model was built using TensorFlow Keras and consisted of three LSTM layers with 128, 64, and 32 units, each followed by dropout layers to prevent overfitting. A final dense layer was used to produce the predicted stock price. The model was trained with the Adam optimizer using mean squared error (MSE) as the loss function for up to 200 epochs, with early stopping applied if the validation loss did not improve for 20 consecutive epochs.

Model performance was assessed using several metrics, including mean squared error (MSE), root mean squared error (RMSE), mean absolute error (MAE), R^2 score, and mean absolute percentage error (MAPE). Predicted stock prices were also compared to actual values to visually evaluate the model's accuracy.

The results in tables 5.1 and 5.2 indicate that Incorporating CNN-based subjectivity features alongside VADER sentiment significantly enhances LSTM forecasting performance. The mean squared error (MSE) decreases from 68,085.87 to 52,829.72, and the root mean squared error (RMSE) reduces from 260.93 to 229.85, indicating improved prediction accuracy. The mean absolute error (MAE) declines from 205.31 to 176.10, while the coefficient of determination (R^2) increases from 0.8382 to 0.8744, reflecting a stronger alignment between predicted and actual values. The mean absolute percentage error (MAPE) also improves from

1.19% to 1.03%. These results demonstrate that integrating subjectivity-aware sentiment features strengthens the model's capacity to capture market dynamics and enhances the reliability of stock price forecasts.

Table 5.1 Forecasting performance using VADER sentiment features with LSTM

MSE	**68085.8677**
RMSE	**260.9327**
MAE	**205.3070**
R^2	**0.8382**
MAPE	**1.19%**

Table 5.2. Forecasting performance using VADER + CNN (Subjectivity) sentiment features with LSTM.

MSE	**52829.7150**
RMSE	**229.8472**
MAE	**176.1031**
R^2	**0.8744**
MAPE	**1.03%**

6. Results

The CNN-based subjectivity classifier effectively distinguished between subjective and objective sentences, allowing the extraction of meaningful SubjObj_Score features. Machine learning models that included these scores performed better in predicting stock market movements than models using only VADER sentiment.

For regression, the LSTM model achieved low prediction errors and high R^2 scores, showing that combining sentiment and subjectivity features with historical stock data captures market patterns well and improves forecasting accuracy. The training and validation loss curves indicated stable convergence without overfitting.

7. Conclusion

This study introduces a hybrid approach to analyzing financial news sentiment by combining VADER sentiment scores with subjectivity/objectivity features extracted from a CNN trained on the Cornell Subjectivity Dataset. The results show that including subjectivity information

enhances both the prediction of stock market direction and the forecasting of actual stock prices. By integrating sentiment with market data, this framework offers a reliable method for using textual information to improve financial forecasting.

References

1. Hutto, C. J., & Gilbert, E. E. (2014). VADER: A parsimonious rule-based model for sentiment analysis of social media text. In *Proceedings of the 8th International Conference on Weblogs and Social Media (ICWSM-14)* (pp. 216–225). ICWSM. https://doi.org/10.1609/icwsm.v8i1.14550
2. Williams, B. (2025?). *Polarity and subjectivity in sentiment analysis: Explained*. Insight7. Retrieved [date], from https://insight7.io/polarity-and-subjectivity-in-sentiment-analysisexplained

3. Hutto, C. J., & Gilbert, E. E. (2014). VADER: A parsimonious rule-based model for sentiment analysis of social media text. In Proceedings of the 8th International Conference on Weblogs and Social Media (ICWSM-14) (pp. 216–225). ICWSM.

4. Biswas, S., Young, K., & Griffith, J. (2022, November 5). A comparison of automatic

labelling approaches for sentiment analysis. arXiv preprint arXiv:2211.02976.

https://doi.org/10.48550/arXiv.2211.02976

5. Thapa, B. (2022, April 26). Sentiment analysis of cybersecurity content on Twitter and Reddit. arXiv preprint arXiv:2204.12267. https://doi.org/10.48550/arXiv.2204.12267

6. Pang, B., & Lee, L. (2004, June). *Subjectivity dataset, version 1* [Data set]. Cornell University. Retrieved from https://www.cs.cornell.edu/people/pabo/movie-reviewdata/subjdata.README.1.0.txt
7. Pang, B., & Lee, L. (2004). A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL-04)* (pp. 271–278). Association for Computational Linguistics. https://doi.org/10.3115/1218955.1218990
8. Cornell dataset http://www.cs.cornell.edu/people/pabo/movie-review-data
9. Raychev, V., & Nakov, P. (2019, November 28). Language-independent sentiment analysis using subjectivity and positional information. *arXiv preprint arXiv:1911.12544***.** https://doi.org/10.48550/arXiv.1911.

110. Sun, J. (2016, August). Daily News for Stock Market Prediction, Version 1. Retrieved from https://www.kaggle.com/aaron7sun/stocknews