Project Report: Sentiment Analysis of Stock News Using Machine Learning

Introduction

This project focuses on analyzing the sentiment of news articles related to stocks and studying their impact on stock prices. The process involves gathering news headlines, preprocessing the text data, extracting features through TF-IDF, and building a machine learning model to classify sentiment. Furthermore, the model's effectiveness is evaluated using cross-validation, and its influence on stock trading is examined through various metrics and visual representations.

Data Collection and Preprocessing

Data Collection

- The dataset comprises stock-related news headlines stored in a DataFrame.
- This DataFrame includes columns for 'Date', 'Headline', and 'Stock Name'.

Text Preprocessing

- The text data is cleaned and standardized to remove noise.
- Preprocessing steps include converting text to lowercase, removing punctuation and stopwords, and applying stemming or lemmatization.

Feature Extraction

TF-IDF Vectorization

- The TfidfVectorizer from scikit-learn is utilized to transform the cleaned headlines into numerical features.
- TF-IDF (Term Frequency-Inverse Document Frequency) measures the significance of words across the dataset.

Model Building and Evaluation

Model Selection

- A Random Forest model (rf_model) is selected for sentiment classification due to its robustness and ability to handle high-dimensional data effectively.

Cross-Validation

- To assess the model's performance, 5-fold cross-validation is used. The dataset is divided into 5 parts, with the model being trained on 4 parts and validated on the remaining part. This process is repeated 5 times.
- The cross-validation score is the average accuracy across the 5 folds.

Performance Metrics and Visualization

Generating Signals

- Based on the predicted sentiment of news headlines, buy/sell signals are generated.
- Buy signals are associated with positive sentiments, while sell signals correspond to negative sentiments.

Performance Metrics

- Several metrics are calculated to evaluate the trading strategy:
    - **Final Portfolio Value**: The portfolio's value at the end of the analysis period.
    - **Sharpe Ratio**: A measure of risk-adjusted return.
    - **Number of Trades Executed**: Total count of buy/sell orders executed.
    - **Win Ratio**: The ratio of successful trades to the total number of trades.

Visualization

- **Closing Price and Buy/Sell Signals**: A plot showing the stock's closing price over time, with markers for buy and sell signals.
- **Portfolio Value Over Time**: A plot depicting the portfolio's value over time.

Results and Discussion

Cross-Validation Score

- The cross-validation score is computed and displayed. If the score is lower than anticipated, possible reasons might include model complexity, feature representation, data quality, class imbalance, and hyperparameter settings.

Analysis and Recommendations

- If the cross-validation score is unsatisfactory, consider the following:
    - **Model Complexity**: Adjust parameters like the number of trees or max depth in the Random Forest model.

- **Feature Engineering**: Enhance TF-IDF features by including bigrams/trigrams or other text features.
- **Data Quality**: Ensure the data is clean and pertinent.
- **Class Imbalance**: Use techniques such as SMOTE to balance the classes.
- **Hyperparameter Tuning**: Apply Grid Search or Random Search to find optimal hyperparameters.
- **Cross-Validation Strategy**: Utilize StratifiedKFold for imbalanced classes.

Conclusion

This project outlines the comprehensive process of performing sentiment analysis on stock-related news and evaluating its impact on stock prices using machine learning. The methodology encompasses data collection, preprocessing, feature extraction, model building, performance evaluation, and visualization. By addressing potential issues at each step, the model's performance can be enhanced, yielding valuable insights for stock trading strategies.