Stock Sentiment Analysis using Machine Learning

In the provided code, several steps are performed to analyze news articles related to Amazon (ticker symbol: AMZN) and their potential impact on stock trading decisions using sentiment analysis, machine learning, and financial modeling techniques. Here's an overview of what the code accomplishes and the rationale behind each step:

1. Data Collection and Preprocessing

- **Data Collection from API**: News articles related to Amazon are fetched from the New York Times Archive API using a Python script. This includes headlines and publication dates from 2012 to 2023.
- **Text Preprocessing**: Once fetched, the news articles are preprocessed. This involves converting text to lowercase, removing numbers, punctuation, and stopwords, and then lemmatizing the remaining tokens. The goal is to clean the text for better sentiment analysis and modeling.

2. Sentiment Analysis

- **VADER Sentiment Analysis**: The VADER (Valence Aware Dictionary and sEntiment Reasoner) sentiment analysis tool is used to assess the sentiment of each news headline. VADER is specifically designed for social media texts and provides polarity scores (positive, negative, neutral, compound) for sentiment analysis without requiring training data.

3. Financial Data Retrieval

- **Stock Price Data Retrieval**: Historical stock price data for Amazon (AMZN) from Yahoo Finance is downloaded using the `yfinance` library. This data includes daily Open, High, Low, Close, Adjusted Close prices, and Volume.

4. Feature Engineering

- **Feature Extraction**: Textual features (e.g., sentiment scores, average Word2Vec embeddings) are extracted from the preprocessed news articles to quantify sentiment and content relevance to Amazon's stock.
- **Label Generation**: Based on the daily returns of Amazon's stock, a binary label is created to indicate whether the return was positive (1) or negative (0). This serves as the target variable for classification.

5. Machine Learning Model Training and Evaluation

- **Model Selection**: Several classification models are chosen for predicting stock movements based on news sentiment and other features. Models include Support Vector Machines (SVM), Logistic Regression, Random Forest, AdaBoost, and others.

- **Grid Search and Cross-Validation**: For each model, hyperparameters are tuned using techniques like GridSearchCV or RandomizedSearchCV to optimize model performance. This helps in finding the best combination of parameters that maximize accuracy, precision, recall, and F1-score.
- **Model Evaluation**: Models are evaluated based on metrics such as accuracy, precision, recall, F1-score, and ROC AUC (Receiver Operating Characteristic Area Under Curve). These metrics provide insights into how well the models classify stock movements based on the extracted features.

6. Trading Strategy and Portfolio Management

- **Trading Strategy Simulation**: Using the predicted buy and sell signals from the classification models, a simulated trading strategy is implemented. This strategy buys Amazon stock when a positive sentiment signal is detected and sells when a negative sentiment signal occurs.
- **Portfolio Value Calculation**: The script calculates the final portfolio value based on simulated trading decisions, considering initial cash, positions held, buy/sell prices, and stock price movements over time.
- **Performance Metrics**: Key performance metrics like Sharpe Ratio, Maximum Drawdown, and Win Ratio are computed to evaluate the effectiveness and risk-adjusted returns of the trading strategy.

7. Visualization

- **Portfolio Value Visualization**: Results are visualized using matplotlib to plot the portfolio value over time. This includes annotating maximum portfolio values and visualizing buy and sell signals on stock price charts.

Methodology and Rationale

- **Integration of Text and Financial Data**: By combining sentiment analysis of news articles with financial data, the approach attempts to capture the impact of news sentiment on stock price movements. This integration provides a more comprehensive analysis compared to traditional purely quantitative approaches.
- **Machine Learning for Classification**: Classification models are used to predict stock movements based on sentiment and other extracted features. This supervised learning approach leverages labeled data (positive/negative returns) to train models and make predictions.
- **Simulation and Evaluation**: Simulated trading and portfolio management help in assessing the practical implications of sentiment-based trading strategies. Metrics such as Sharpe Ratio and Maximum Drawdown provide insights into risk and return characteristics.
- **Visualization for Interpretation**: Visualizations aid in understanding the performance of the trading strategy and the impact of sentiment signals on stock price movements. They also facilitate the interpretation of model predictions and trading decisions.

In summary, the code implements a data-driven approach combining sentiment analysis, machine learning, and financial modeling to explore the relationship between news sentiment and stock market performance. It demonstrates how sentiment from textual data can be used

to enhance trading decisions and portfolio management strategies, offering a blend of quantitative analysis and qualitative insights from news content.