**Stock Movement Prediction and Forecasting Using Reddit Sentiment and Real-Time Data**

**1. Scraping Process and Challenges**

To extract stock-related discussions, the Reddit API was used. The process involved:

1. **API Account Creation:** A developer account was created, and an application was registered to obtain the API credentials.
2. **Keyword-Based Data Extraction:** Initially, data was extracted using the keyword "stocks," which yielded posts about numerous companies. Analyzing multiple companies posed challenges in data consistency and prediction accuracy, so the focus was narrowed to Tesla-related discussions.
3. **Columns Extracted:** Posts related to Tesla were scraped, and the following key features were extracted:
   * Title: Main topic of discussion.
   * Score: Popularity of the post.
   * Upvote Ratio: Ratio of upvotes to total votes.
   * Number of Comments: Engagement level.
   * Created At: Post date and time.
   * Body: Detailed content of the post.

**Challenges Encountered:**

* **API Throttling:** Reddit API imposes rate limits, causing delays during high-volume extractions.
* **Data Cleaning:** Preprocessing unstructured data to make it analyzable took considerable time and effort, especially for a first-time user.
* **Reddit Data Format:** Understanding nested JSON responses from the API and mapping them into a tabular format was complex.

**Solutions:**

* Implemented pagination and rate limiting to handle API constraints.
* Used Natural Language Processing (NLP) techniques to preprocess text data efficiently for sentiment analysis.

**2. Features Extracted and Their Relevance to Stock Movement Prediction**

The following features were identified as crucial for the analysis:

|  |  |
| --- | --- |
| Feature | Relevance |
| Title | Highlights key stock-related discussions, aiding in sentiment analysis. |
| Score | Indicates the popularity and importance of the post. |
| Upvote Ratio | Helps measure overall community sentiment. |
| Number of Comments | Reflects community engagement, which can signal stock interest. |
| Created At | Allows alignment with real-time stock price data for trend analysis. |
| Body | Provides detailed context for sentiment and topic classification. |

Using NLP, sentiment scores were derived from the Title and Body columns, which were mapped to stock price trends (e.g., "positive" or "negative" sentiment).

**3. Model Development and Evaluation Metrics**

**Model Used:**

* Random Forest Classifier: Chosen for its robustness and ability to handle tabular data.

**Evaluation Metrics:**

The model was evaluated on its ability to predict stock movement (Up=1, Down=0) using sentiment scores:

* Accuracy: 76.09%
* Precision: 71.06%
* Recall: 76.09%
* F1-Score: 67.42%

**Confusion Matrix:**

|  |  |  |
| --- | --- | --- |
| **Actual vs Predicted** | **Predicted Down (0)** | **Predicted Up (1)** |
| Actual Down (0) | 1 | 6 |
| Actual Up (1) | 0 | 34 |

**Forecasting Model Performance:**

Real-time stock price data was integrated to build a forecasting model:

|  |  |
| --- | --- |
| **Metric** | **Score** |
| Accuracy | 65.63% |
| Precision | 64% |
| Recall | 60% |
| F1-Score | 62% |

Next-Day Forecast: The model predicted the next day's stock movement (Down = 0).

**4. Hyperparameter Tuning**

To optimize the Random Forest Classifier, RandomizedSearchCV was used with the following parameter grid:

|  |  |
| --- | --- |
| Parameter | Values Tested |
| n\_estimators | [50, 100, 200, 300] |
| max\_depth | [None, 10, 20, 30, 50] |
| min\_samples\_split | [2, 5, 10] |
| min\_samples\_leaf | [1, 2, 4] |
| max\_features | ['sqrt', 'log2', None] |
| bootstrap | [True, False] |

Best Parameters:

{'n\_estimators': 200, 'min\_samples\_split': 2, 'min\_samples\_leaf': 2, 'max\_features': 'log2', 'max\_depth': 10, 'bootstrap': True}

Post-Tuning Accuracy:

* Test Accuracy improved to 75%.

**5. Future Improvements and Suggestions**

1. **Enhanced Feature Engineering:**
   * Extract more granular features, such as sentiment trends over time.
   * Use financial indicators (e.g., RSI, MACD) to enrich the dataset.
2. **Integration of Multiple Data Sources:**
   * Incorporate data from news APIs (e.g., Google News, Bloomberg).
   * Add alternative social media data (e.g., Twitter) to refine sentiment analysis.
3. **Improved Models:**
   * Experiment with deep learning models like LSTMs or Transformers for sentiment-based time-series predictions.
   * Compare ensemble models (e.g., XGBoost, LightGBM).
4. **Real-Time Pipeline Development:**
   * Automate data scraping and analysis for live stock trend forecasting.
   * Deploy models as an API for real-time predictions.

**6. Objective & Accuracy**

In this analysis, we aim to compare the sentiment classification of Tesla-related posts with the actual stock movements for the following day. The goal is to determine whether the sentiment (positive, negative, or neutral) of the posts correlates with the stock price movement the next day, providing insights into how sentiment can potentially influence or reflect stock behavior.

After processing and comparing the sentiment data with stock price movements, the accuracy of our predictions was found to be 49.38%. This means that in approximately half of the cases, the sentiment classification successfully predicted whether the stock would go up or down on the following day.

**Conclusion**

The combination of Reddit sentiment data and real-time stock prices demonstrated the potential for accurate stock movement predictions. While the current approach achieved promising results, future work focused on enhanced features, diversified data sources, and advanced models could significantly improve prediction accuracy and applicability in dynamic markets.