

Financial News Sentiment Analyzer

Group Project Proposal

Project Title:

Financial News Sentiment Analyzer

Group Members:

1. Tandem Young
2. Jacob Meaders
3. Shreyash Gupta

Objective:

The goal of this project is to create an application that aggregates financial news from various sources, summarizes the articles using natural language processing (NLP) techniques, analyzes the sentiment of the news, and correlates the sentiment with movements in relevant financial markets. This will provide insights into how news sentiment impacts asset prices.

Background and Rationale:

Financial markets are heavily influenced by news and investor sentiment. Being able to quickly digest large volumes of news, determine the overall sentiment, and analyze how that sentiment affects asset prices would be extremely valuable for investors and traders.

Manually reading through financial news is time consuming. Using NLP to automatically summarize articles and gauge sentiment makes the process much more efficient. Correlating the sentiment data with price movements also provides predictive capabilities.

We chose this project to gain experience with NLP, sentiment analysis, data visualization, and to produce a practical tool for analyzing financial markets. It combines our interests in finance, data science, and software engineering.

Features and Functionality:

- Web scraper that collects financial news articles from major sources (Bloomberg, Reuters, WSJ, etc.)

- NLP model that summarizes articles to capture key points
- Sentiment analysis model that classifies article sentiment as positive, negative or neutral
- Visualization of sentiment scores over time
- Correlation of sentiment with price movements of major stock indices and sectors
- User-friendly web interface to display results and allow user to specify news sources and assets to analyze

Technologies and Libraries:

- Python for data collection, analysis, and backend
- NLTK and SpaCy for NLP and sentiment analysis
- Pandas and NumPy for data manipulation
- Matplotlib for data visualization
- Streamlit for web app user interface
- BeautifulSoup or Selenium for web scraping
- SQLite database to store news data

Data Sources:

- Financial news websites scraped using Python web scraping libraries
- Stock price and index data from Yahoo Finance or another financial data API
- News articles stored in SQLite database
- Price data stored as CSV files

Project Design:

- Web scraper module collects news articles and stores in database
- NLP pipeline retrieves articles, summarizes text, performs sentiment analysis, stores results
- Sentiment and price data loaded into Pandas for analysis and visualization
- Streamlit web app displays sentiment dashboard and allows user interaction
- App deployed using Heroku or AWS

Milestones and Timeline:

1. Project planning, design and setup (1 week)
2. Develop news web scraper (1 week)
3. Develop NLP pipeline for summarization and sentiment analysis (2 weeks)
4. Conduct sentiment-price correlation analysis (1 week)
5. Create data visualizations (1 week)
6. Build web app UI (2 weeks)
7. Deploy web app (1 week)
8. Testing and refinement (1 week)

Challenges and Risks:

- News site web scrapers may break if site structures change
- NLP sentiment analysis may not be accurate, especially for financial jargon
- Storing and processing large amounts of textual data could be computationally intensive
- Effectively visualizing correlations between sentiment and price
- Deploying and scaling web app

Learning Goals:

- Gain practical experience with NLP and sentiment analysis techniques
- Learn how to create insightful data visualizations
- Develop skills in full-stack development to create data product from end-to-end
- Understand challenges of working with unstructured textual data
- Explore correlations between news sentiment and financial asset prices

Additional Notes:

- Could extend project to analyze sentiment of earnings call transcripts or SEC filings
- Potential to productize sentiment insights for investment firms
- Consider applying more advanced NLP like transformers to improve performance