Stock Market News Sentiment Analysis

Natural Language Processing

David Lundvall
04/11/2025

Contents / Agenda

- Executive Summary
- Business Problem Overview and Solution Approach
- EDA Results
- Data Preprocessing
- Model Performance Summary

Executive Summary

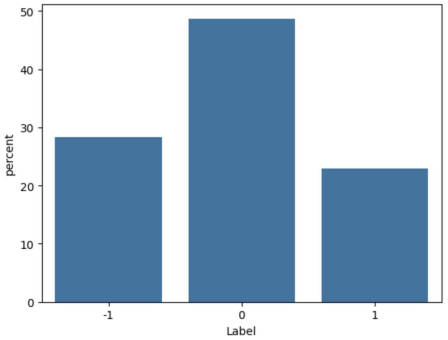
- The AI team has built an LLM to assist financial analysts and increase the efficiency and effectiveness of their day-to-day activities
- This tool will help them to quickly sift through and accurately interpret news and media reports' potential positive or negative impact on stock prices and the market
- We recommend running a 30 day pilot with a subset of analysts to understand and measure the impact the model has on their investment strategies

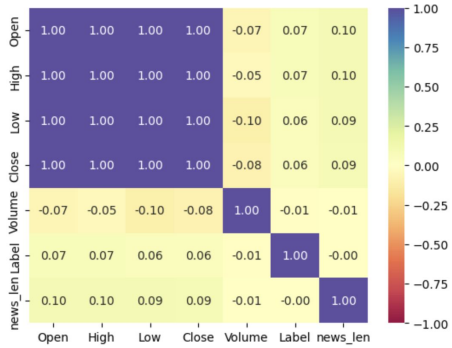
Business Problem Overview and Solution Approach

- Problem
  - Financial analysts are having a difficult time consuming and analyzing the large volume of news related to the economy and stock market
  - The investment firm needs to innovate and have sophisticated tools which analysts can leverage to analyze market sentiment and integrate this information into their investment strategies
- Solution
  - The data science AI team went ahead to tackle this problem and used stock-related news article data to train several models to provide positive or negative sentiment
  - The result is a tuned model that can quickly and accurately provide analysts the top 3 positive and top 3 negative news articles so they can focus on the macro & micro picture to help with their stock decisions and strategy

EDA Results

- The dataset that was used had 349 news articles dated between 01/02/2019 – 04/30/2019
- The average news article was 49 words in length
- The dataset is imbalanced with almost 50% of the news articles labeled “neutral,” while ~30% were labeled “negative” and ~20% were labeled “positive”
- All of the price data (open, close, high, low) is all highly-correlated, but none of the other data showed any correlation





Data Preprocessing

- There were no duplicate values and there were no missing values
- The original dataset was split into train, validation, and test datasets with 286, 21, and 42 rows
- The target dataset contained “Label” data which indicated the article’s sentiment
- For Word2Vec model, the news articles dataset was used to extract and create 4,682 word embeddings in it’s vocabulary
- The Stanford GloVe model, which has 400k words in its vocabulary, was used to create word embeddings for the GloVe model
- The Sentence Transformer model that was used is the “all-MiniLM-L6-v2”

Sentiment Analysis - Model Evaluation Criterion

- The F1-score metric was used during the evaluation and determining the best model
- F1-score was used is because the dataset was imbalanced and it is a good metric to use in cases like this
- Both precision and recall were important and needed to be considered together because we wanted to minimize false positives and false negatives

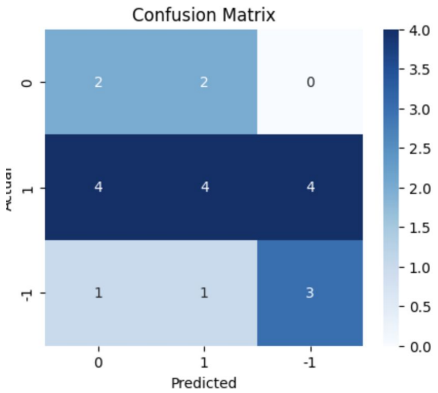
Sentiment Analysis - Model Building

- Three models were used for sentiment analysis: Word2Vec, GloVe, and Sentence Transformer
- Word2Vec is an NLP technique for obtaining vector representations of words which capture information about the meaning of the word based on the surrounding words. The algorithm estimates these representations by modeling text in a large corpus.
- GloVe in an NLP technique that learns word embeddings by analyzing co-occurrence statistics from a large corpus of text, capturing both semantic and syntactic relationships between words.
- Sentence Transformer is an NLP technique that captures the semantic meaning of sentences, going beyond the capabilities of traditional word embeddings.

Sentiment Analysis - Base Model Building

- The models were tested using these classifiers: gradient boosting, random forest, and decision tree
- For Word2Vec, decision tree was found to have the best F1-score of 0.429
- For GloVe, random forest was found to have the best F1-score of 0.427
- For Sentence Transformer, random forest was found to have the best F1-score of 0.505

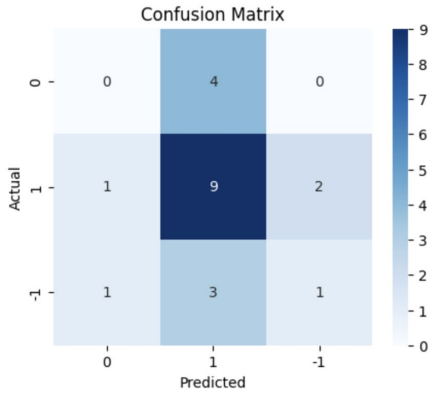
Word2Vec



Validation performance:

	Accuracy	Recall	Precision	F1
0	0.428571	0.428571	0.482993	0.428913

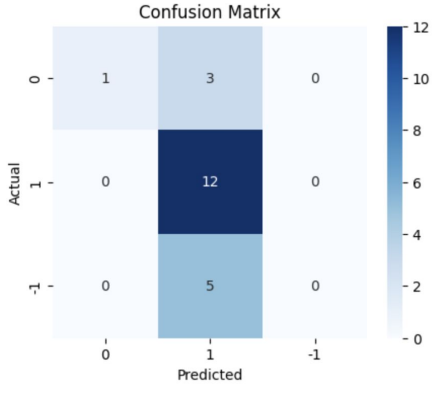
GloVe



Validation performance:

	Accuracy	Recall	Precision	F1
0	0.47619	0.47619	0.400794	0.426871

Sentence Transformer



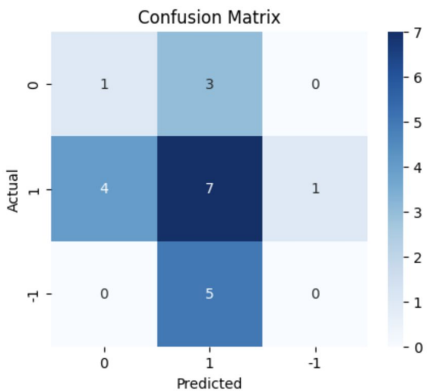
Validation performance:

	Accuracy	Recall	Precision	F1
0	0.619048	0.619048	0.533333	0.504762

Sentiment Analysis - Model Improvement

- The models were tuned and tested using the best classifier from the base model
- For Word2Vec, decision tree had an F1-score of 0.338 (it went down)
- For GloVe, random forest had an F1-score of 0.531 (it went up)
- For Sentence Transformer, random forest had an F1-score of 0.416 (it went down)

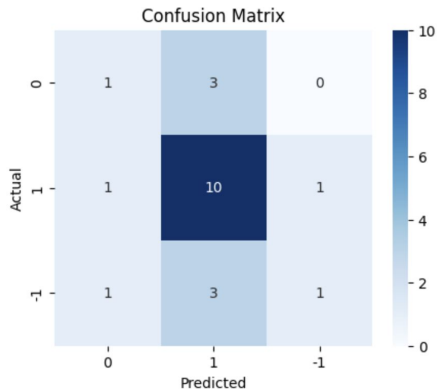
Word2Vec



Validation performance:

	Accuracy	Recall	Precision	F1
0	0.380952	0.380952	0.304762	0.338624

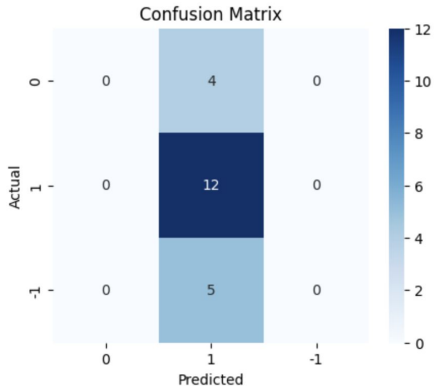
GloVe



Validation performance:

	Accuracy	Recall	Precision	F1
0	0.571429	0.571429	0.539683	0.530612

Sentence Transformer



Validation performance:

	Accuracy	Recall	Precision	F1
0	0.571429	0.571429	0.326531	0.415584

Sentiment Analysis – Model Performance Comparison

- All of the base models were overfit on the training data and had 100% performance
- The tuned models were trained using the best classifier from the base model and the model parameters shown below
- The tuned models were not overfit, but unfortunately 2 of the 3 actual had worse F1-scores
- The best performing model by F1-score was the tuned GloVe model with 0.505

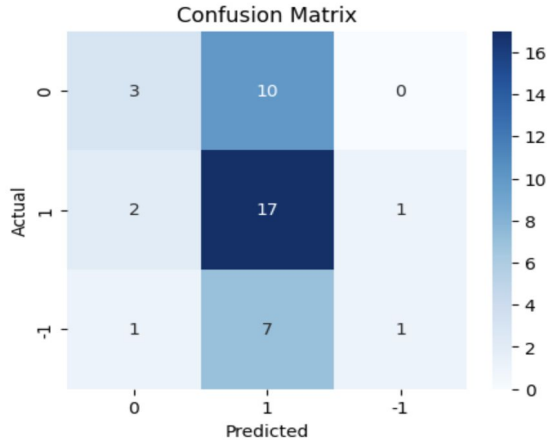
```
parameters = {
    'max_depth': np.arange(3,7),
    'min_samples_split': np.arange(5,12,2),
    'max_features': ['log2', 'sqrt', 0.2, 0.4]
}
```

Validation performance comparison:

	Base Model (Word2Vec)	Base Model (GloVe)	Base Model (Sentence Transformer)	Tuned Model (Word2Vec)	Tuned Model (GloVe)	Tuned Model (Sentence Transformer)
Accuracy	0.428571	0.476190	0.619048	0.523810	0.571429	0.571429
Recall	0.428571	0.476190	0.619048	0.523810	0.571429	0.571429
Precision	0.482993	0.400794	0.533333	0.314286	0.539683	0.326531
F1	0.428913	0.426871	0.504762	0.392857	0.530612	0.415584

Sentiment Analysis – Final Model

- The best performing model by F1-score was the tuned GloVe model with 0.505
- Yet this model underperformed on the test data with an F1-score of 0.437



Test performance for the final model:

	Accuracy	Recall	Precision	F1
0	0.5	0.5	0.5	0.436529

Content Summarization – Data Preprocessing

- We used the same stock news dataset (stock_news.csv) from the first modeling exercise
- We grouped the news by week and separated each article using a “ || ” delimiter, which gave us 18 weeks of data

Content Summarization – Modeling Approach

- For content summarization and sentiment analysis, we downloaded (via HuggingFace) an LLM from Meta: TheBloke/Mistral-7B-Instruct-v0.2-GGUF
- The Llama model used a layer of 100 GPUs and a context window of 4500
- The Llama model's max_tokens were set to 1024, the temperature was set to 0 (to be as predictable as possible), and top_p was set to 0.1 (so only the top tokens are considered to make the output more deterministic)
- The model was trained to summarize the news from the week by identifying the top three positive and negative events that are most likely to impact the price of the stock

Content Summarization – Sample Input

- The prompt below was used to get the top 3 positive and top 3 negative news for each week
- Fine-tuning was needs so that neutral results weren't classified into either positive or negative
- Clarity on the JSON output format was very important so that the week number wasn't added

```
#Define the prompt for this task
prompt = """
    Your role is as an expert financial analyst specializing in stock market and economy news analysis.
    Assess the overall sentiment of each news article and classify it into one of the following categories:
    - "Positive"
    - "Negative"
    - "Neutral"

    After that is complete, identify the up to the top three positive events for the stock price (there can be between 0 and 3) and
    up to the top three negative events for the stock price (there can be between 0 and 3).

    Return the output in the specified JSON format, ensuring consistency and providing proper formatting.
    Ensure that all values in the JSON are formatted as strings.
    The format of the key name must be labeled exactly in the provided format and not altered, do not add a number for the week to the key.
    The value in the key-value pair is to be provided by you.
    Do not add any additional key-value pairs; only use what is specified for positive and negative events, do not include neutral events.
    This is the JSON format you should return which shows the key-value pairs (the key name should be exactly as it is here with no alterations):
    {
        "Week Positive Events": ["your_sentiment_prediction",]
        "Week Negative Events": ["your_sentiment_prediction",]
    }

    Only return the JSON, do NOT return any other text or information. Make sure the JSON is valid and well-formatted.
"""
```

Content Summarization – Sample Output

- Below is sample output resulting from the provided prompt and dataset
- 3 negative events and 2 positive events were found for this particular week and the JSON format looks good, demonstrating that the prompt instructions were followed


```
{
  "Week Negative Events":
      ["Apple lowered its fiscal Q1 revenue guidance to $84 billion from earlier estimates of $89-$93
billion due to weaker than expected iPhone sales.",
       "Apple cut its fiscal first quarter revenue forecast from $89-$93 billion to $84 billion due to
weaker demand in China and fewer iPhone upgrades.",
       "Apple Inc. lowered its quarterly sales forecast for the fiscal first quarter, underperforming
analysts' expectations due to slowing Chinese economy and trade tensions."],
  "Week Positive Events":
      ["Roku Inc has announced plans to offer premium video channels on a subscription basis through
its free streaming service, The Roku Channel.",
       "Apple CEO Tim Cook discussed the company's Q1 warning on CNBC, attributing US-China trade
tensions as a factor but also mentioning projected Services revenue exceeding $10.8 billion in Q1."]
}
```

Content Summarization – Raw Model Output

- Below is a snapshot of the resultant dataframe
- Each week includes both the negative event articles and the positive event articles

	Key Events
0	{ "Week Negative Events": ["Apple lowe...
1	{ "Week Positive Events": ["AMS develo...
2	{ "Week Negative Events": ["U.S. stock...
3	{ "Week Positive Events": ["IBM's stoc...
4	{ "Week Negative Events": ["Caterpilla...

Content Summarization – Final Output

- First, the key events data was normalized into a dataframe which contained the week's negative events in one column and the week's positive events in another column (looking like just the two columns on the right below)
- Then that dataframe was then concatenated with the original data, the key events column containing the raw JSON was dropped, and you can see the results below

	Date	News	Week Negative Events	Week Positive Events
0	2019-01-06	The tech sector experienced a significant dec...	[Apple lowered its fiscal Q1 revenue guidance ...	[Roku Inc has announced plans to offer premium...
1	2019-01-13	Sprint and Samsung plan to release 5G smartph...	[Geely forecasts flat sales for 2019 due to ec...	[AMS develops new 3D facial recognition featur...
2	2019-01-20	The U.S. stock market declined on Monday as c...	[U.S. stock market declined due to concerns ov...	[Dialog Semiconductor reported fourth quarter ...
3	2019-01-27	The Swiss National Bank (SNB) governor, Andre...	[The Dow, S&P 500, and Nasdaq experienced sign...	[IBM's stock price increased after hours due t...
4	2019-02-03	Caterpillar Inc reported lower-than-expected ...	[Caterpillar Inc reported lower-than-expected ...	[Apple reported spending over $60 billion with...