**Unveiling Internet Sentiment through News: Insights from LLMs**

**Abstract:**

In this paper I present a novel system to estimate cumulative online sentiment of users on different domains based on news articles. This task requires gathering the most influential news articles daily, analyzing their sentiment, and creating an algorithm to aggregate the metric. To achieve this, I introduce a Large Language Model-based online dashboard. The news articles are obtained through web scraping techniques. By using a prompt-engineered Phi-2 LLM as the core of my system, I obtain accuracies of over 74%.

**Introduction:**

Understanding internet sentiment expressed through news articles has emerged as a pivotal endeavor in deciphering public opinion dynamics and its ramifications across various domains. The exponential growth of digital content has presented both opportunities and challenges in distilling meaningful insights from the vast sea of textual data. In this context, the ability to accurately analyze and interpret internet sentiment holds immense significance for stakeholders ranging from investors and policymakers to researchers and media analysts. However, traditional sentiment analysis approaches often fall short in capturing the nuanced sentiment nuances inherent in online discourse, necessitating more sophisticated techniques.

Recognizing the limitations of conventional sentiment analysis methods, this study endeavors to leverage the capabilities of Large Language Models (LLMs) to unravel the complexities of internet sentiment expressed in news articles. By harnessing advanced natural language processing techniques, we aim to address the pressing need for more robust sentiment analysis frameworks capable of discerning subtle shifts in public sentiment over time. Motivated by the increasing importance of internet sentiment analysis in decision-making processes across diverse sectors, the development of our system seeks to provide actionable insights into societal attitudes, market sentiment, and socio-political trends.

In this paper, we elucidate the rationale behind the creation of our internet sentiment analysis system and its potential applications in various domains. By delving into the problem statement, the significance of internet sentiment analysis, and the rationale behind our system's development, we lay the foundation for exploring the intricacies of sentiment analysis in the digital age.

**Related Work:**

The study by Kemal Kirtac et al. [1] reveals that hedge funds with higher sentiment beta, indicating a stronger response to market sentiment, tend to outperform those with lower sentiment beta. This suggests that skilled hedge fund managers (approximately 10%) demonstrate the ability to time market sentiment effectively, which correlates with higher sentiment beta and better fund performance. Notably, the return spread between the top and bottom deciles of hedge funds ranked by sentiment beta is as large as 0.59% per month on a risk-adjusted basis, further validating the efficacy of sentiment-based strategies.

The paper by Kelvin Du et al. [2] highlights the evolution of Financial Sentiment Analysis (FSA) techniques, emphasizing the shift from traditional classification models to advanced Large Language Models (LLMs). LLMs have demonstrated superior performance in sentiment analysis by leveraging deep learning and pre-trained language models, which are more effective in capturing the nuances and complexities of financial texts. This advancement has significantly contributed to the accuracy and reliability of sentiment analysis in financial contexts, making LLMs a preferred choice for extracting sentiment data that can inform trading decisions.

Lastly, Small Language Models (SLMs) have shown promising potential to outperform their larger counterparts in the realm of sentiment analysis. Notably, Microsoft Research’s Phi-2 [3], a 2.7 billion-parameter SLM, has demonstrated its remarkable ability to match or exceed the capabilities of base language models with significantly more parameters. The compact size of Phi-2 also makes it an ideal candidate for fine-tuning on specific tasks, potentially leading to more accurate sentiment analysis that can enhance trading strategies. Additionally that due to their smaller size, Phi-2 offers faster inference times, which can be crucial for trading algorithms, where timely decision-making is essential.

**Methodology:**

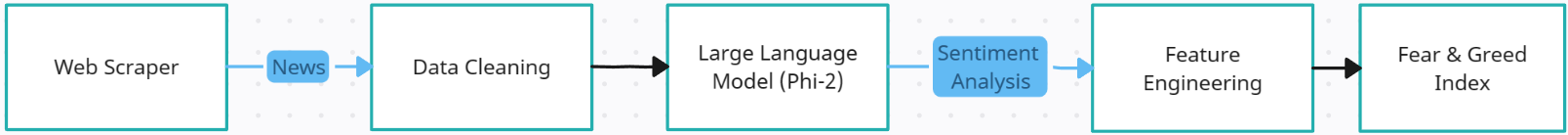


Fig. 1: Overview of System.

1. ***Web Scraper***



Fig. 2: Web Scraper Process.

The methodology employed for obtaining the latest news articles from the internet centered on leveraging web scraping techniques, with a particular focus on the Wall Street Journal (WSJ) website. The purpose was to gather a diverse array of news articles relevant to financial markets and the economy.

As we developed the methodology, we had to keep a few things in mind. First, we needed to make sure we were following the rules and acting ethically. This meant choosing websites that allowed us to scrape their data legally and without violating anyone's privacy. Second, we had to consider how easy it would be to actually get the data we needed from each website. Some websites have complicated structures or security measures that make it hard to scrape data from them. After weighing these factors, we decided to focus on the Wall Street Journal because it met our criteria well—it had a wide range of content and was relatively easy to scrape compared to other options.

To extract the necessary information from the WSJ website, we searched for relevant HTML tags, particularly <a> and <p> tags, within each HTML response for every article. We scraped the title, content, article link, and publication time from these tags.

By carefully addressing these factors and employing this method, our web scraping technique effectively collected important news articles. This formed the basis for further examination and understanding within the Fear and Greed index framework.

1. ***Data Processer***

The methodology for the Data Cleaning block aimed to format the data and remove duplicate news articles, ensuring the integrity and efficiency of subsequent analysis. This step was crucial, especially considering the need to run the web scraper multiple times a day to obtain real-time news updates.

The implementation involved leveraging Python's Regex (regular expressions) and Pandas libraries. Initially, the focus was on identifying and filtering out duplicate articles scraped during multiple runs of the web scraper.

Furthermore, during data inspection, it was observed that certain special characters appeared in the text, lacking significance in English or potentially disrupting downstream analysis. To address this, Python's Regex capabilities were utilized to systematically remove such characters from the text data. This ensured that the text input for subsequent analysis remained clean and devoid of unnecessary noise.

1. ***Phi-2 Small Language Model***

Fig. 3: Phi-2 Usage Process.

In the LLM (Language Model) block, we utilized Phi-2, a smaller-sized LLM developed by Microsoft with 2.78 billion parameters, to conduct sentiment analysis on news articles obtained from our web scraping process.

To achieve this, we employed Prompt Engineering combined with Two-shot prompting. Prompt Engineering involves designing specialized prompts or instructions to guide the LLM in performing specific tasks, such as sentiment analysis. Two-shot prompting entails providing the model with both the prompt and the text to analyze.

Before finalizing the prompt, we conducted experiments using a supervised financial news dataset to refine it, ensuring it effectively captured article sentiment.

In practice, we first load the LLM and initialize the prompt. Then, we iterate through each news article, appending the prompt as context before feeding it to the LLM. The model then makes a sentiment prediction for each article, which we save in our dataframe.

Additionally, we adjusted the Phi-2 LLM settings by specifying max\_new\_tokens=1 to reduce computation time without sacrificing analysis accuracy.

1. ***Feature Engineering***

The Feature Engineering block serves two primary purposes: first, to categorize news articles into specific subdomains such as 'Real-Estate' and ‘Economy’, and second, to calculate sentiment scores on a daily, weekly, and monthly basis using aggregation techniques tailored for each subgroup. This block is implemented using Natural Language Processing (NLP) techniques along with Pandas for data manipulation and an Exponential Moving Average (EMA) algorithm.

Initially, the sentiment data is loaded, with sentiment scores designated as the target variables. Leveraging the URLs of the articles, a function extracts the subdomain to categorize each article accordingly. This process creates a new feature, enabling subsequent analysis at the subdomain level.

------------------------------------------------------------------------------------------------------------------------

EMA computation Formula:

Given a series of sentiment scores *S*1​,*S*2​,*S*3​,...,*Sn*​ over a specified time window, the EMA at time *t* is calculated as:

***EMAt*​** = (***St*​** × 2/(***N***+1)​) + (***EMAt*−1**​ × (1 − 2/(***N***+1)​))

*Where:*

***St*** *represents the sentiment score at time t.*

***N*** *is the span parameter, indicating the number of periods to consider for the EMA calculation.*

***EMAt−1*** *is the EMA value from the previous time period.*

------------------------------------------------------------------------------------------------------------------------

For the EMA computation, two types of data are required. Firstly, a sliding bucket or window is employed to display sentiment scores over past intervals. Additionally, a current EMA function is utilized, focusing solely on sentiment scores from the present day. These algorithms generate sentiment scores across three different timeframes: 4-day (Short), 12-day (Medium), and 30-day (Long).

1. ***Fear and Greed Styled Dashboard***

A graph and diagram of a person

Description automatically generated with medium confidence

Fig. 4: Fear & Greed Index Dashboard with Plotly and Streamlit.

The Data Visualization block serves the purpose of offering a comprehensive overview of online sentiment across various economic sectors. To achieve this, Plotly and Streamlit were employed as the primary visualization tools.

The implementation consists of two main visual elements: a line chart and a gauge chart. These were chosen due to their familiarity, mirroring the format often seen in popular sentiment indices such as CNN's fear and greed index.

The line chart provides a historical perspective, allowing users to track sentiment trends over time. On the other hand, the gauge chart offers an immediate snapshot of the current sentiment status. This dual approach caters to different user preferences and analytical needs.

To enhance user experience and flexibility, interactive features were incorporated. Users can select the desired time frame, choosing between short, medium, and long-term sentiment views. Additionally, they have the option to focus on specific economic sectors. In cases where no sector is specified, the visualization defaults to displaying the overall sentiment across all sectors, ensuring accessibility and ease of use for all users.

**Experiments:**

In this section, we outline the experimental setup and procedures conducted to evaluate the performance of the Phi-2 sentiment analysis model. The experiments aimed to assess the effectiveness of prompt engineering techniques and the impact of different prompt configurations on sentiment classification accuracy.

**Objective**

The objective of the experiments was to verify the sentiment classification accuracies achieved by the Phi-2 model under various prompt engineering strategies and configurations.

**Dataset Used**

The primary dataset utilized for evaluating the Phi-2 model's sentiment analysis capabilities was the Financial Phrasebank. This dataset comprises 4,850 financial news articles annotated with sentiment labels, including neutral, positive, and negative sentiments. The distribution of sentiments within the dataset is as follows: 60% neutral, 28% positive, and 12% negative.

|  |  |
| --- | --- |
| **Sentiment** | **Rows of Data** |
| Neutral | 1,391 |
| Positive | 570 |
| Negative | 303 |

Table. 1: Financial Phrasebank Dataset Distribution.

**Prompt Engineering Tests**

To optimize the Phi-2 model's performance, several prompt engineering strategies were tested and evaluated using the Financial Phrasebank dataset. Through iterative experimentation, insights were gained regarding the effectiveness of different prompt formulations. Notable findings include:

|  |  |  |
| --- | --- | --- |
| **Added to Prompt** | **Location** | **Accuracy Change** |
| Spaces | Before new text line | NaN |
| Spaces | After new text line | + 2.1% |
| Spaces | Both sides of text line | + 1.5% |
| Concise Phrasing | In prompt | + 1% per 3 tokens |
| Zero Indexing | Classification statement | + 20.7% |

Table. 2: Prompt Engineering Tests on Phi-2.

These experiments demonstrated the importance of carefully crafting prompts tailored to the sentiment analysis task and dataset characteristics.

**N-Shot Prompting Evaluation**

In addition to prompt engineering, the Phi-2 model's performance was evaluated using N-shot prompting techniques. N-shot prompting involves providing the model with multiple examples of each sentiment class to enhance its understanding and classification capabilities. The results of N-shot prompting experiments are summarized below:

|  |  |
| --- | --- |
| **N-shot** | **Accuracy** |
| 0 | 0.635 |
| 1 | 0.687 |
| 2 | 0.743 |
| 3 | 0.385 |

Table. 3: N-Shot Prompting Tests (Overall).

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **N-Shot** | **Precision** | | | **Recall** | | | **F1 Score** | | |
| **Class ->** | **Neg.** | **Neut.** | **Pos.** | **Neg.** | **Neut.** | **Pos.** | **Neg.** | **Neut.** | **Pos.** |
| 0 | 0.471 | 0.632 | 0.088 | 0.082 | 0.974 | 0.102 | 0.14 | 0.766 | 0.182 |
| 1 | 0.49 | 0.671 | 0.977 | 0.828 | 0.822 | 0.771 | 0.615 | 0.738 | 0.143 |
| 2 | 0.702 | 0.738 | 0.93 | 0.917 | 0.922 | 0.212 | 0.795 | 0.82 | 0.345 |
| 3 | 0.342 | 0.946 | 0.382 | 0.99 | 0.05 | 0.881 | 0.508 | 0.095 | 0.532 |

Table. 4: N-Shot Prompting Tests (Class wise).

Different examples yielded varying accuracies. For instance, the sentence "Its 168 asset management experts manage assets worth over EUR 35 billion" achieved higher accuracy, while "In January-September 2009, the Group's net interest income increased to EUR 112.4 mn from EUR 74.3 mn in January-September 2008" resulted in significantly lower accuracy, at **0.441** compared to **0.68**.

These experiments highlighted the effectiveness of N-shot prompting in enhancing the Phi-2 model's sentiment classification performance, particularly when provided with sufficient diverse examples for each sentiment class.

Overall, the experiments conducted provided valuable insights into the effectiveness of prompt engineering and N-shot prompting techniques in optimizing the Phi-2 sentiment analysis model for financial sentiment classification tasks.

**Conclusion:**

In conclusion, my study has led to the development of the Phi-2 sentiment analysis model, specifically tailored for financial sentiment classification tasks. Through the utilization of advanced prompt engineering techniques and the exploration of N-shot prompting strategies, I have significantly improved the model's capability to discern nuanced sentiment expressions within financial text data.

The experiments conducted using the Financial Phrasebank dataset have provided invaluable insights into the effectiveness of various prompt configurations and the influence of N-shot prompting on sentiment classification accuracy. By meticulously processing data, optimizing the performance of Large Language Models (LLMs), and creating a comprehensive dashboard, I have addressed the growing demand for accurate sentiment analysis tools in deciphering internet sentiment.

Looking forward, my focus remains on broadening the applicability of the Phi-2 model to encompass a broader spectrum of financial sentiment analysis tasks. Through continuous refinement of methodologies and leveraging advancements in natural language processing, I aim to contribute to the development of robust sentiment analysis frameworks that empower decision-makers across various sectors to extract actionable insights from online discourse.

Additionally, it's essential to highlight the appropriate and inappropriate use cases for the Phi-2 model. It should be utilized for market sentiment analysis, aiding in market timing decisions, and serving as a confirmation tool. However, it's not suitable for predicting market movements, serving as a sole indicator for decision-making, or short-term trading strategies.

**References:**

[1] [Sentiment Trading and Hedge Fund Returns (wiley.com)](https://onlinelibrary.wiley.com/doi/epdf/10.1111/jofi.13025?saml_referrer)

[2] [Financial Sentiment Analysis: Techniques and Applications (acm.org)](https://dl.acm.org/doi/pdf/10.1145/3649451)

[3] [Phi-2: The surprising power of small language models - Microsoft Research](https://www.microsoft.com/en-us/research/blog/phi-2-the-surprising-power-of-small-language-models/)

Github Repo: