Sentiment Analysis in Finance: Exploring Techniques for Effective Classification of Financial Text Data

Qinggang Zeng, Jason Chow and Yize Dai
Stony Brook University
May 5, 2023

Abstract

This project explores various approaches to sentiment classification of financial field-related sentences. We experimented with different models and strategies, including fine-tuning on small-sized pre-trained models, zero-shot and few-shot learning on Large Language Models (LLMs). Our objective was to identify the most effective method for sentiment classification of financial text data. We evaluated our models on a dataset of financial news articles and reports, and compared their performance using metrics such as precision, recall, and F1 score. Our findings suggest that the model that provides the best results for sentiment classification in the financial domain out of all of our solution approaches is the GPT-3.5 Turbo model when given 6 correctly classified samples. Overall, this project provides valuable insights for practitioners and researchers working on sentiment analysis in finance.

Introduction

The financial industry generates vast amounts of text data every day, including news articles, company reports, social media posts, and more. Analyzing this data to determine the sentiment behind it can provide valuable insights for investors, analysts, and other stakeholders. However, sentiment analysis of financial text presents unique challenges, including the complexity of financial language and the need for accurate classification of nuanced sentiments. In addition, texts related to finance are often mixed with a large number of numbers rather than just words.

These numbers largely constitute the core meaning of the text, and it is difficult to accurately infer sentiment through this digital information.

The overall goal of our project is to find a better way to perform sentiment analysis on financial-related sentences. Specifically, we aim to develop a model that can accurately classify a sentence as either positive, negative, or neutral. We have experimented with various machine learning techniques, including fine-tuning on pre-trained models, zero-shot learning, and few-shot learning on Large Language Models (LLMs).

In order to identify the most effective approach, we evaluate the performance of each model using standard metrics such as precision, recall, and F1-score. Our ultimate objective is to identify the best-performing model that can accurately classify financial sentences into their corresponding sentiment classes based on these metrics.

In this paper, we will present our methodology, data collection, and experimental results. Our contributions include a thorough evaluation of different sentiment analysis techniques on a dataset of financial news articles and reports. Our findings will provide insights for practitioners and researchers working in the field of sentiment analysis in finance, and help to advance our understanding of this challenging and important problem.

In our experiments, we find that few-shot prompting on LLMs is much more effective than zero-shot prompting. Few-shot prompting on Flan-T5-XXL, a model with 11 billion parameters, outperformed GPT-3 Davinci, a model with 175 billion parameters, using zero-shot prompting (Weighted F1 = 0.469728 vs 0.431026).

Ideas

Since we have limited time and computational resources, we need to find the least time-consuming approach that can provide accurate results. To achieve this goal, we decided to use online large language models, which can provide a balance between performance and resource utilization. To compare the performance of different approaches, we experimented with two options, few-shot and zero-shot learning. Both approaches have expected good performance.

We also want to explore the potential benefits of fine-tuning a pre-trained model approach. This approach can leverage the knowledge already learned by a pre-trained model on a large dataset of general language data and adapt it to a smaller dataset of financial-related sentences. This can theoretically lead to higher accuracy in sentiment classification and is suitable for scenarios where a large amount of labeled financial data is available.[1]

[1] The idea was not completed due to one of our members abruptly quitting on the deadline day

zero-shot LLM:
zero-shot LLM approach is a quick and efficient solution that does not require any training on the specific financial sentiment classification task. It leverages a pre-existing language understanding and knowledge to classify financial sentences, making it an attractive option for scenarios where resources are limited. To implement the zero-shot LLM approach, we chose to use two of the most popular language models, GPT-3 and GPT-Neo. These models are widely used in natural language processing tasks and are known for their high performance and accuracy.

Few-shot LLM:
The few-shot prompting approach with LLMs is a small optimization over the zero-shot approach. It involves showing K examples of a given task to the model at inference time in order to guide its prediction. Using few-shot prompting leads to a considerable reduction in the amount of task-specific data needed, compared to approaches such as fine-tuning. However, the performance of few-shot prompting has been shown to be much worse compared to that of state-of-the-art fine-tuned models. For this task, few-shot prompting (K=6 and K=15) was used with two LLMs: GPT-3.5 (gpt-3.5-turbo) and Flan-T5 (flan-t5-xxl).

Experimental Setup

Dataset:
The financial_phrasebank dataset is a polar sentiment dataset that consists of 4840 English-language financial news sentences categorized by sentiment. Unlike most datasets, the dataset is not split into train/validation/test, but instead divided by agreement rate of 5-8 annotators. It provides four possible configurations based on the percentage of agreement of annotators. Each data instance in the financial_phrasebank dataset consists of two fields: a "sentence" field and a "label" field. The "sentence" field contains a tokenized line from the financial news dataset, while the "label" field contains a label corresponding to the sentiment of the sentence. The sentiment label can be one of three classes: "positive", "negative", or "neutral". Note that these dataset labels were annotated by 16 people with adequate background knowledge of financial markets, including three researchers and 13 master's students at Aalto University School of Business with majors primarily in finance, accounting, and economics. This dataset was created based on the findings of the paper *'Good Debt or Bad Debt: Detecting Semantic Orientations in Economic Texts'* (Malo et al., 2014), which identified a lack of high-quality training data for building models to perform sentiment analysis on financial texts.

Model:
GPT-3 (Brown et al., 2020) and GPT-Neo (Black et al., 2021) are used for the zero-shot LLM solution. GPT-3 is a large language model developed by OpenAI. It is based on the transformer

architecture and has 175 billion parameters, making it the largest publicly available language model to date. GPT-3 can perform a variety of natural language processing tasks, including language translation, question-answering, and text generation. It is trained on a diverse range of internet text data and can generate human-like responses to text prompts.

GPT-Neo is another transformer-based language model developed by EleutherAI. It is an open-source alternative to GPT-3 and has been trained on a large corpus of internet text data. GPT-Neo has a smaller number of parameters than GPT-3, with the largest version having 2.7 billion parameters. Despite its smaller size, GPT-Neo has shown promising results on a variety of natural language processing tasks, and it has been used in several research studies.

Both GPT-3 and GPT-Neo are capable of zero-shot learning, meaning that they can theoretically perform well on tasks without any fine-tuning or additional training. This is possible because the models are trained on a diverse range of text data and can use their learned knowledge to perform a wide variety of natural language processing tasks.

GPT-3.5 Turbo, an improved version of GPT-3 Davinci, was used for the few-shot prompting solution. ChatGPT is a popular chatbot that is based on GPT-3.5 Turbo. Its architecture is based on GPT-3's, however a technique known as "Reinforcement Learning with Human Feedback" (RLHF) was used in this improved model's development. In RLHF, human feedback is used to adjust the model's behavior. With RLHF, GPT-3.5 Turbo generates more informative and impartial responses as well as rejects improper questions, as well as those that are beyond its scope of knowledge (Fu et al., 2022).

Flan-T5 (Chung et al., 2022) was another model that was used in the few-shot prompting solution. Flan-T5 is an enhanced version of the T5 model that has been fine tuned on over 1000 additional tasks covering more languages. It is a transformer based model which treats every language problem as a text-to-text problem. For this task, Flan-T5-XXL was used, which is a pretrained model with 11 billion parameters.

Method:

In this study, a zero-shot language model (LLM) solution was implemented using the API from GPT-3. Given that the API did not offer a specific setting for classification, several prompts were used and evaluated for their appropriateness. It was found that the prompt "Please analyze the following text for sentiment with only Neutral or Positive or Negative as output, no other discussion needed: Text: {prompt} Sentiment:" provided the most reasonable results.

The all_degree sub dataset from financial_phrasebank was utilized as the input for the experiments, and different temperature settings (0.1, 0.2, 0.4, 0.6, 0.8) and sub-models (davinci, curie, babbage) were tested. Note that all_degree sub dataset has a 100% agreement rate of 5-8

annotators. Additionally, for comparison purposes, the EleutherAI/gpt-neo-2.7B from the Transformers package was also employed in the experiments. Moreover, it is worth mentioning that GPT-Neo offers a specific API setting for classification tasks.

To implement the few-shot prompting solution using GPT-3.5 Turbo, the OpenAI API was used. Due to the rate limits of the API, batches of four sentences were sent per request for sentiment classification. Two system prompts, followed by the four sentences packed into an array format were sent to the API. The expected response was an array of sentiments. The exact prompt that was sent is shown in Figure 1.

```
 {'role': 'system',
  'content': "Perform sentiment analysis on each of the sentences given. Six example sentences
and their correct sentiments will be given. Reply with either 'negative', 'neutral', or
'positive' in an array."},
 {'role': 'system',
  'content': "Sentence: The international electronic industry company Elcoteq has laid off tens
of employees from its Tallinn facility ; contrary to earlier layoffs the company contracted the
ranks of its office workers , the daily Postimees reported .\nSentiment: negative\n\nSentence:
According to Gran , the company has no plans to move all production to Russia , although that
is where the company is growing .\nSentiment: neutral\n\nSentence: With the new production
plant the company would increase its capacity to meet the expected increase in demand and would
improve the use of raw materials and therefore increase the production profitability
.\nSentiment: positive\n\nSentence: A tinyurl link takes users to a scamming site promising
that users can earn thousands of dollars by becoming a Google ( NASDAQ : GOOG ) Cash advertiser
.\nSentiment: negative\n\nSentence: Technopolis plans to develop in stages an area of no less
than 100,000 square meters in order to host companies working in computer technologies and
telecommunications , the statement said .\nSentiment: neutral\n\nSentence: According to the
company 's updated strategy for the years 2009-2012 , Basware targets a long-term net sales
growth in the range of 20 % -40 % with an operating profit margin of 10 % -20 % of net sales
.\nSentiment: positive\n\n"},
 {'role': 'user',
  'content': '["FINANCING OF ASPOCOMP \'S GROWTH Aspocomp is aggressively pursuing its growth
strategy by increasingly focusing on technologically more demanding HDI printed circuit boards
PCBs .", "For the last quarter of 2010 , Componenta \'s net sales doubled to EUR131m from
EUR76m for the same period a year earlier , while it moved to a zero pre-tax profit from a
pre-tax loss of EUR7m .", "In the third quarter of 2010 , net sales increased by 5.2 % to EUR
205.5 mn , and operating profit by 34.9 % to EUR 23.5 mn .", "Operating profit rose to EUR 13.1
mn from EUR 8.7 mn in the corresponding period in 2007 representing 7.7 % of net sales .",
"Operating profit totalled EUR 21.1 mn , up from EUR 18.6 mn in 2007 , representing 9.7 % of
net sales ."]'}
```

Figure 1: An example of a prompt sent to the OpenAI API for chat completion using the model "gpt-3.5-turbo" for the few-shot (K=6) prompting solution. Text colored in green represents system prompts used to guide the model. The first system prompt is the instruction given to the model. The second system prompt contains the 6 correctly classified sentences separated by two newline characters. Text colored in red represents the user prompt that contains the 4 sentences to be classified packed into an array format.

To implement the few-shot prompting solution using Flan-T5-XXL, the HuggingFace Inference API was used. Sentences to be classified were sent one per request to the API using a prompt

based on that of the second system prompt in Figure 1. The expected response is the predicted sentiment of the sentence sent.

The subset "sentences_50agree" was used in the implementation of the few-shot prompting solution. For each of the models used, K=6 and K=15 were used to observe the effect in varying the number of examples shown to the model. The examples shown to each model were kept constant across each K to isolate the effect of the number of examples shown on model performance.

Evaluation Metrics:

The precision, recall, and F1 score are widely used evaluation metrics in classification tasks for natural language processing problems, including sentiment analysis. Precision for a class *i* refers to the proportion of correct predictions of class *i* out of all instances where class *i* was predicted. Recall for a class *i*, on the other hand, measures the proportion of correct predictions of class *i* out of the number of instances that are truly of class *i*. The F1 score for a class *i* is the harmonic mean of the precision and recall for class *i*, which takes into account both metrics and provides a single measure of model performance. These three metrics can be aggregated in three different ways: macro-average, weighted average, and micro-average. The macro average of a metric is simply the arithmetic mean of the metric across each class. The weighted average of a metric accounts for class imbalance by computing the average of the metric across each class weighted by the number of actual occurrences of that class. The micro average of a metric aggregates the contribution of each class to compute the averaged metric, and is ideal for scenarios with a class imbalance. In our study, we have utilized these three evaluation metrics and aggregation methods to measure the performance of solutions.

Result

Zero-shot LLM:

In the results of our study, we found that the Curie and Babbage models from GPT-3 had significantly lower performance in detecting sentiment with F1 scores of 0.27 and 0.31 respectively when compared to the Davinci model. This was observed in the same temperature setting of 0.2. We also performed experiments with different temperature settings using the Davinci model and the results are presented in the following chart and plots. The results from GPT-Neo are also presented below.

GPT-3 Davinci temperature = 0.1				
Class	Precision	Recall	F1	Support
0	0.940541	0.574257	0.713115	303
1	0	0	0	1391
2	0.269971	0.984211	0.423716	570
weighted	0.193846	0.324647	0.202117	
macro	0.403504	0.519489	0.378944	
micro	0.324647	0.324647	0.324647	

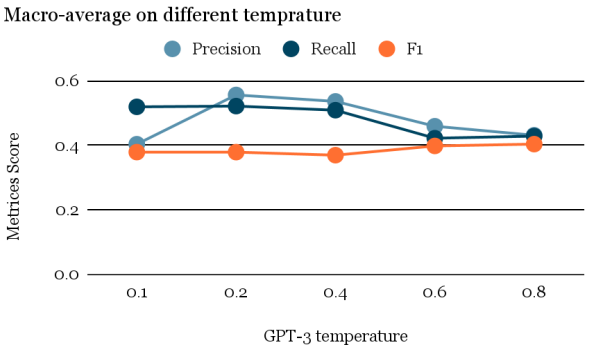
GPT-3 Davinci temperature = 0.2				
Class	Precision	Recall	F1	Support
0	0.898477	0.584158	0.708	303
1	0.5	0.00215672	0.00429492	1391
2	0.270742	0.978947	0.424173	570
weighted	0.49561	0.325972	0.204186	
macro	0.556407	0.521754	0.378823	
micro	0.325972	0.325972	0.325972	

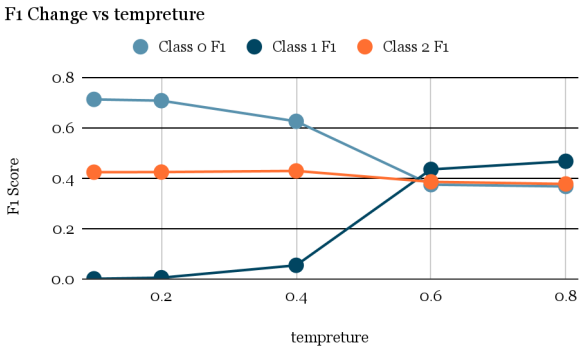
GPT-3 Davinci temperature = 0.4				
Class	Precision	Recall	F1	Support
0	0.742081	0.541254	0.625954	303
1	0.590909	0.0280374	0.0535347	1391
2	0.276176	0.957895	0.42874	570
weighted	0.531902	0.33083	0.224608	
macro	0.536389	0.509062	0.36941	
micro	0.33083	0.33083	0.33083	

GPT-3 Davinci temperature = 0.6				
Class	Precision	Recall	F1	Support
0	0.505618	0.29703	0.37422	303
1	0.594022	0.342919	0.434822	1391
2	0.278254	0.626316	0.385321	570
weighted	0.502691	0.408127	0.414249	
macro	0.459298	0.422088	0.398121	
micro	0.408127	0.408127	0.408127	

GPT-3 Davinci temperature = 0.8				
Class	Precision	Recall	F1	Support
0	0.399225	0.339934	0.367201	303
1	0.612602	0.377426	0.467082	1391
2	0.281984	0.568421	0.376963	570
weighted	0.500806	0.420495	0.431026	
macro	0.43127	0.428594	0.403749	
micro	0.420495	0.420495	0.420495	

GPT-Neo				
Class	Precision	Recall	F1	Support
0	0.06	0.128713	0.0818468	303
1	0.855769	0.0639827	0.119064	1391
2	0.306623	0.812281	0.445192	570
weighted	0.611011	0.261042	0.196191	
macro	0.407464	0.334992	0.215368	
micro	0.261042	0.261042	0.261042	





Few-shot LLM:

We find that the model that performed best across all metrics is GPT-3.5 Turbo when given 6 examples. This is quite a surprising result as one would expect the model that is given more examples to perform better. The GPT-3.5 Turbo model performed similarly for K=6 and K=15, with the model that is given 15 examples performing marginally better when predicting neutral

and positive sentiment in terms of F1-score. The Flan-T5-XXL model performed as expected with performance scaling with the number of examples given. It also expectedly performed worse than GPT-3.5 Turbo, which has 175 billion parameters compared to Flan-T5-XXL's 11 billion. The results of each experiment are presented below.

Few-Shot Performance Summary				
Model	Accuracy	Precision	Recall	F1
Flan-T5-XXL (K=6)	0.503306	0.781051	0.503306	0.432228
Flan-T5-XXL (K=15)	0.525978	0.777404	0.525978	0.469728
GPT-3.5 Turbo (K=6)	**0.624793**	**0.753473**	**0.624793**	**0.614931**
GPT-3.5 Turbo (K=15)	0.624094	0.746019	0.624094	0.614571

Flan-T5-XXL Few-Shot (K=6)				
Class	Precision	Recall	F1	Support
0	0.641387	0.983389	0.776393	602
1	0.994118	0.176225	0.299380	2877
2	0.392427	0.982366	0.560822	1361
weighted	0.781051	0.503306	0.432228	
macro	0.675977	0.713993	0.545532	
micro	0.503306	0.503306	0.503306	

GPT3.5-Turbo Few-Shot (K=6)				
Class	Precision	Recall	F1	Support
0	0.697226	0.960133	0.807827	602
1	0.903632	0.423705	0.576905	2877
2	0.460932	0.901543	0.609993	1361
weighted	0.753473	0.624793	0.614931	
macro	0.687263	0.761794	0.664908	
micro	0.624793	0.624793	0.624793	

Flan-T5-XXL Few-Shot (K=15)				
Class	Precision	Recall	F1	Support
0	0.623003	0.976628	0.760728	599
1	0.984424	0.219903	0.359499	2874
2	0.407385	0.974963	0.574653	1358
weighted	0.777404	0.525978	0.469728	
macro	0.671604	0.723831	0.564960	
micro	0.525978	0.525978	0.525978	

GPT3.5-Turbo Few-Shot (K=15)				
Class	Precision	Recall	F1	Support
0	0.667055	0.956594	0.786008	599
1	0.894851	0.429367	0.580296	2874
2	0.465870	0.889543	0.611491	1358
weighted	0.746019	0.624094	0.614571	
macro	0.675925	0.758502	0.659265	
micro	0.624094	0.624094	0.624094	

Analysis and Discussion

Our findings indicate that increasing the GPT-3 temperature setting has a significant impact on the F1 score for different sentiment classes. Specifically, we found that increasing the temperature leads to an increase in the F1 score for class 1, but a decrease in the F1 score for class 0. This suggests that the optimal temperature setting for the sentiment classification task may depend on the class distribution in the dataset. One reason for the decrease in the F1 score for class 0 with an increase in the GPT-3 temperature setting could be the model's tendency to generate more diverse and creative responses with higher temperatures. This could result in the model producing more neutral or positive sentiment responses for negative sentiment inputs, leading to misclassifications and a decrease in the F1 score for class 0.

We also found that for class 0 (negative sentiment), the F1 score, recall, and precision are high at lower temperatures, indicating that this class is easier to predict accurately at lower temperatures. However, for class 1 (neutral sentiment), the performance is very poor, with an F1 score close to 0 even at lower temperatures. This could indicate that the model is struggling to distinguish

between neutral and negative sentiment, or that there are not enough training examples for this class. The poor performance of class 1 at all temperature settings could be due to the inherent difficulty of distinguishing between neutral and negative sentiment. Since fewer financial dataset was provided to the GPT-3 at the time it was developed, it lacks diversity and generalization in the financial field.

Additionally, we observed that class 2 (positive sentiment) consistently has low precision, but high recall and a stable F1 score. This suggests that the model may be incorrectly predicting some instances as positive sentiment, but it rarely misclassifies instances of the other classes as belonging to class 2. This could be due to an imbalance in the training data or the model's bias towards more positive language.

The analysis of GPT-Neo's performance on financial sentiment classification suggests that the model is also not performing well on this task. The precision and recall scores for class 0 (negative) and class 1 (neutral) are both very low, indicating that the model is struggling to correctly identify negative and neutral sentiments. The precision score for class 2 (positive) is moderate, but the recall score is not as high as one might hope. The F1 scores for the dataset are low, indicating that the model is not performing well on this task.

Overall, it is important to note that while GPT-3 and GPT-Neo are powerful language models with impressive capabilities, they may not always be the best choice for zero-shot sentiment classification tasks like in the financial field. Our findings indicate that GPT-3 and GPT-Neo struggle with accurately predicting neutral sentiment and may require significant fine-tuning to achieve high performance in this class.

For the few-shot prompting approach, our findings indicate that increasing the number of samples shown to the Flan-T5-XXL model improves the model’s performance across all metrics (accuracy, weighted precision, weighted recall, weighted f1 score). Notably, the model improved in its classification of sentences of neutral (F1 = 0.299 vs 0.354) and positive (F1 = 0.561 vs 0.575) sentiment, trading some performance in its classification of sentences of negative (F1 = 0.776 vs 0.761) sentiment. The overall result makes sense from a theoretical standpoint as the model is shown more examples of correctly classified financial sentences, and therefore should be more capable of classifying the sentiment of other financial sentences.

The findings for GPT-3.5 Turbo were less expected, as our findings indicate that increasing the number of samples shown to the model had little to no effect on the model’s performance. In fact, when the model was shown less samples (K=6), the model performed marginally better than when it was shown more samples (K=15) across all metrics. However, when more samples were shown to the model, it tended to slightly improve in its classification of sentences of neutral (F1 = 0.577 vs 0.580) and positive (F1 = 0.610 vs 0.611) sentiment, while trading off some

performance in its classification of sentences of negative (F1 = 0.808 vs 0.786) sentiment. A possible explanation for this is the fact that ChatGPT, which runs on the GPT-3.5 Turbo model, may have sacrificed its in-context learning ability for the ability to model dialogue context (Fu et al., 2022).

Overall, in the few-shot prompting approach, the GPT-3.5 Turbo model outperformed the Flan-T5-XXL model across all metrics. This is expected as GPT-3.5 Turbo has 175 billion parameters, compared to Flan-T5-XXL's 11 billion parameters. With more parameters, a model is able to represent more complicated functions, and in the context of natural language processing, the model is able to better understand and model the intricacies of natural language, allowing the model to perform different tasks more successfully, like sentiment analysis. The performance of both models can definitely be improved by tuning different hyperparameters, such as changing the way the prompt is structured, modifying batch size, and modifying model-specific hyperparameters.

Compared to the model with the best settings found in the zero-shot prompting approach (GPT-3 Davinci, Temperature=0.8), the worst model from the few-shot prompting approach (Flan-T5-XXL, K=6) still performed better across all weighted metrics. This is quite surprising given the difference in the number of parameters between each model. However, this shows how powerful few-shot prompting can be compared to zero-shot prompting -- a model with 11 billion parameters can outperform a model with 175 billion parameters just by passing it 6 correctly classified samples.

The best model out of all of our solution approaches is the GPT-3.5 Turbo model when given 6 correctly classified samples across every metric (Accuracy = 0.624793, Weighted Precision = 0.753473, Weighted Recall = 0.624793, Weighted F1 = 0.614931). A trend that is noticed across all models independent of the solution approach is that the models seemed to do worse in sentences of neutral sentiment. This may be due to the fact that sentences that have a very slight positive or negative sentiment are flagged by the models as either positive or negative, but are in reality neutral. For example, a sentence where the best model predicts positive while it's in reality neutral is: "As a result of the share issue , the number of the company 's shares increases by 391,630 shares and the share capital by a total of EUR 15,665.20 ." The sentence contains positive keywords such as "increases by" and two positive numbers. To the untrained eye, the sentence does indeed seem to convey a positive sentiment. Therefore, the model in that case just lacks the knowledge needed to fully understand the sentence and its implications. The model can also be seen making large errors, incorrectly classifying positive sentences as negative: "Nokia also noted the average selling price of handsets declined during the period , though its mobile phone profit margin rose to more than 22 percent from 13 percent in the year-ago quarter ." The sentence is a complex one with decreasing and increasing keywords as well as mentions of different time periods. The model was most likely tricked by the first part of the sentence where

the average selling price declined, while the part that actually has an effect on stock price -- “mobile phone profit margin rose to more than 22 percent from 13” was probably ignored by the model. The model also seems to be swayed by punctuation as noted by the following neutral sentence that was classified as positive: “Welcome !” The sentence alone without context can be seen as one of positive sentiment due to the exclamation point. However, in the standpoint of finance, the sentence has no meaning or in this case, no effect on any stock price.

Code

https://drive.google.com/drive/folders/1jKXzCZCTzF_JC4akf-kVmfAWeF6U4-6U?usp=sharing

Learning Outcomes

Working on this project we got the opportunity to develop a more comprehensive understanding of the capabilities and limitations of large language models. We also gained a deeper understanding of how these models work, including exploring the architecture and components of GPT models such as the transformer-based encoder-decoder architecture, self-attention mechanisms, and pre-training objectives. Furthermore, we learned about analyzing the performance of language models by examining metrics like precision, recall and F1 scores, and understanding the importance of dataset selection and evaluation methodologies.

In addition to that, we improved our skills in sending and receiving requests from an API, as well as getting around rate limits using techniques such as batching, timeouts, exponential backoff. We’ve also learnt about what zero_shot and few-shot prompting is and how it can be used as a powerful tool when used in conjunction with large language models. We’ve also explored the history of GPT models, especially the evolution of GPT3 to GPT3.5 with new techniques such as reinforcement learning through human feedback and instruction tuning.

Contributions

Qingguang Zeng: Implemented and evaluated the zero-shot LLM solution for sentiment classification using GPT-3 and GPT-Neo. Conducted the experiments, analyzed the results, and provided insights on the strengths and weaknesses of this approach. Also, contributed significantly to the writing of the report, including drafting most of the sections and organizing the overall structure.

Jason Chow: Implemented and evaluated the few-shot LLM solution for sentiment classification using GPT-3.5 and Flan-T5. Sample analysis on a model, analysis on few-shot compared to zero-shot approaches.

Reference

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., Hesse, C., . . . Amodei, D. (2020). Language Models are Few-Shot Learners. ArXiv. /abs/2005.14165

Malo, P., Sinha, A., Takala, P., Korhonen, P., & Wallenius, J. (2013). Good Debt or Bad Debt: Detecting Semantic Orientations in Economic Texts. ArXiv. /abs/1307.5336

Black, Sid, Leo, Gao, Wang, Phil, Leahy, Connor, & Biderman, Stella. (2021). GPT-Neo: Large Scale Autoregressive Language Modeling with Mesh-Tensorflow (1.0). Zenodo. https://doi.org/10.5281/zenodo.5297715

Fu, Y., Peng, H., & Khot, T. (2022, December 11). How Does GPT Obtain Its Ability? Tracing Emergent Abilities of Language Models to Their Sources. [web log]. Retrieved from https://yaofu.notion.site/How-does-GPT-Obtain-its-Ability-Tracing-Emergent-Abilities-of-Language-Models-to-their-Sources-b9a57ac0fcf74f30a1ab9e3e36fa1dc1.

Chung, H. W., Hou, L., Longpre, S., Zoph, B., Tay, Y., Fedus, W., Li, Y., Wang, X., Dehghani, M., Brahma, S., Webson, A., Gu, S. S., Dai, Z., Suzgun, M., Chen, X., Chowdhery, A., Castro-Ros, A,. Pellat, M., Robinson, K., …, Wei, J. (2022). Scaling Instruction-Finetuned Language Models. ArXiv. abs/2210.11416