Including Emotion & Sentiment in Multimodal Learning with Audio and Transcripts of Earnings Conference Calls for Predicting Volatility of Stock Prices

James Chapman
Department of Computer Science
Kansas State University
Manhattan, KS, USA
JamesChapman@ksu.edu

Nathan Diehl
Department of Electrical and Computer Engineering
Kansas State University
Manhattan, KS, USA
npdiehl@ksu.edu

John Woods
Department of Computer Science
Kansas State University
Manhattan, KS, USA
jwoods03@ksu.edu

***Abstract*—This project explores the use of multimodal deep learning to predict stock price volatility by leveraging both textual and audio data from earnings conference calls. Traditionally, stock volatility prediction relies on time-series models using historical price data, but recent advancements allow the incorporation of non-numeric data sources like financial news and company reports. In this study, both text transcriptions and audio recordings of earnings calls are utilized. We apply a Hierarchical Transformer-based Multi-task Learning (HTML) model to combine text and audio features, aiming to predict stock price volatility over multiple time frames (3, 7, 15, and 30 days). The text data is encoded using GloVe, RoBERTa, and BGE, while the audio data is processed through Praat for phonetic analysis and emotion2vec for emotion recognition. Our experimental setup trains 180 models using combinations of text and audio embeddings to optimize the prediction of stock volatility. Results indicate that RoBERTa and BGE text embeddings combined with Praat audio embeddings yield the best performance in volatility prediction. This approach demonstrates the potential of multimodal learning in financial forecasting. Future work may including the use of spectrograms, sentiment analysis, and incorporating news articles for improved accuracy.**

***Index Terms*—earnings call, multimodal deep learning, stock price volatility prediction**

I. Introduction

Predicting stock price volatility is a critical task in financial markets, as it provides insights into an asset's risk profile, enabling investors and analysts to make more informed decisions regarding risk management, portfolio allocation, and pricing strategies. Traditionally, volatility predictions have been based on time-series models that rely heavily on historical price data, such as ARIMA (AutoRegressive Integrated Moving Average) models, which are effective at capturing past price movements to forecast future trends [1]. However, with the rapid advancement of natural language processing (NLP) and machine learning techniques, the scope of data utilized for these predictions has expanded beyond traditional numerical datasets to include unstructured textual data, such as financial news articles, company reports, and earnings call transcripts. These textual data sources offer valuable insights into market sentiment, company performance, and external factors influencing stock behavior, which can significantly enhance the accuracy of volatility predictions.

Moreover, recent research has explored the potential of incorporating audio features, particularly from earnings calls, to gain a deeper understanding of the factors influencing stock prices. Earnings calls are key events where a company's management provides updates on its performance and future outlook, often coupled with a Q&A session with analysts. Verbal communication during these calls can reveal underlying sentiment, confidence, or uncertainty about a company's future, while vocal features such as tone, pitch, and emotion can provide additional context to the spoken words. These audio signals may serve as early indicators of market-moving information that is not always captured by text alone. Recent studies have suggested that analyzing these audio features can offer valuable predictive signals, but there remains a need for more sophisticated models to effectively capture and combine both text and audio data for enhanced predictions.

Building on these developments, this project leverages deep learning techniques to integrate both textual and audio data from earnings calls, aiming to predict stock price volatility more accurately. The key objective is to evaluate how each modality—text and audio—contributes to the prediction process. By using multimodal data sources, we seek to uncover deeper insights into market behavior, where the combination of verbal content (from text) and emotional cues (from audio) can provide a more comprehensive understanding of the factors driving stock price fluctuations.

II. Related Work

Much research has been done into predicting stock price volatility from earnings calls. Below, we highlight the foundational papers our project is built on that utilize the same datasets and benchmarks for volatility prediction. The following papers have been thoroughly examined, and we incorporate

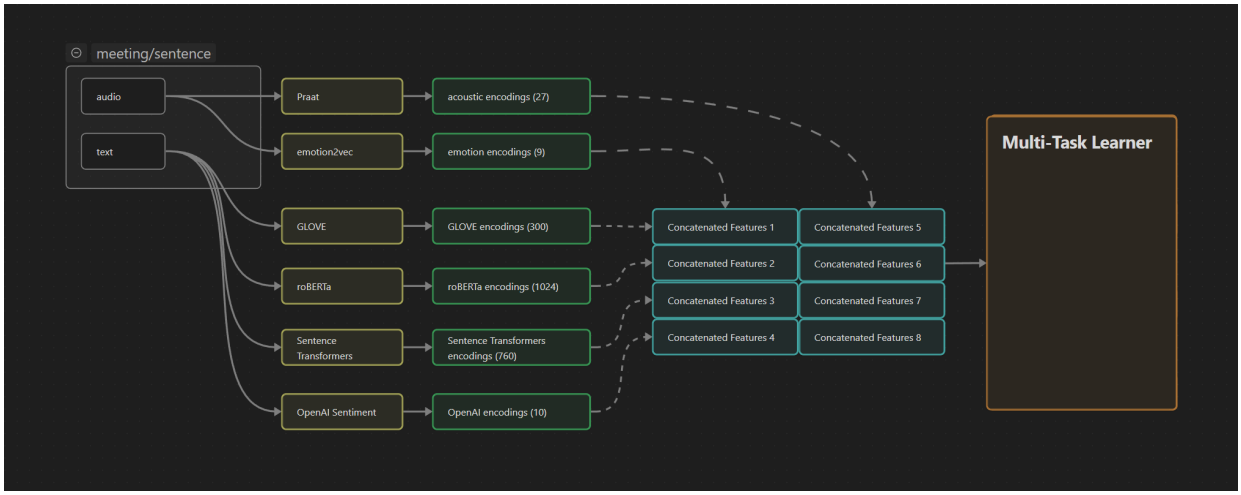

Fig. 1. Encoding Generating Diagram

their results in comparison to our own. The papers not only provide valuable background information, but also serve as a foundation for developing the code required for model training.

A. Predicting Stock Volatility Using Verbal and Vocal Cues

This paper [1] by Yu Qin and Yi Yang showed that vocal features could be used to predict a stock's risk level. While past research used textual information from official press releases and reports, Yu Qin and Yi Yang uses both text transcripts and audio recordings from investor meetings and earnings conference calls as a source of stock information.

Their research showed that a CEO's vocal features and emotions could make significant improvements in a models ability to make market predictions.

This paper is where the Earnings Call Dataset that we uses in our training comes from.

B. Multimodal Aligned Earnings Conference Call Dataset for Financial Risk Prediction

This paper [2] builds upon the work of Yu Qin and Yi Yang [1]. This paper has access to a much larger dataset, almost six times as many meetings spaced over a greater time interval. This allowed them to acheive much better results from their models.

This paper is also where the MAEC Dataset that we use in our training comes from.

C. HTML: Hierarchical Transformer-based Multi-task Learning for Volatility Prediction

This paper [3] introduces a new model architecture specifically designed for predicting stock volatility from audio and text data. The HTML (Hierarchical Transformer-based Multi-task Learning) model used a hierarchical transformer to examine the input data at multiple different scales, such as the sentence and paragraph levels. This more detailed examination of the data allows the model to gain a better understanding of what the CEO is saying in a given meeting. The HTML is also a Multi-task model, being trained with multiple different n-day volatility as targets.

The HTML model architecture was the starting point for designing our model architecture.

D. KeFVP: Knowledge-enhanced Financial Volatility Prediction

This paper [4] also introduces a new model architecture specifically designed for predicting stock volatility from audio and text data. The KeFVP (Knowledge-enhanced Financial Volatility Prediction) model is more complected model then HTML. In addition to the language processing that models like HTML do, KeFVP is injected with financial knowledge such as common financial metrics. This initial knowledge allows the model to better interpret the CEO's words in the meetings since the model has a better context for the financial vocabulary and concepts that CEO will be using.

E. Other Relevant Literature

Numerous other methods have been proposed to more accurately predict stock price volatility from multimodal earnings call data. Many of these also use the Earnings Call and MAEC datasets we are using.

HTML was improved by the authors to create NumHTML [5], which takes advantage of the quantity of numerical data present in the dataset. Instead of treating everything as plain text tokens, it uses structured adaptive pretraining to properly handle numerical data, which significantly improves performance. VolTAGE [6] uses a Graph Convolution Network instead of a transformer-based model to account for the correlations between stocks, which outperforms most other transformer-based methods on the dataset.

Large Language Models (LLMs) have also been applied to this problem. ECC Analyzer [7] uses LLMs to extract content that is more predictive of stock price volatility from earnings call transcripts. This information is used to augment the performance of multimodal deep learning models. RiskLabs

[8] uses LLMs to process large amounts of financial data, aggregating earnings calls, market-related time series data, and financial news to better predict stock volatility.

Additionally, research has been done examining the bias present in these datasets. [9] analyzes gender bias in the Earnings Call dataset when used to predict stock price volatility. The authors note how sensitive models are to gender-specific audio features and the underrepresentation of female executives in the dataset which leads to a significant difference in MSE. AMA-LSTM [10] attempts to rectify this by using adversarial learning to generate perturbations of the data, which reduces bias and variability when training multimodal models. It also resulted in improved accuracy over HTML. DocFin [11] incorporates additional structured financial data including earnings, profit-loss statements, and balance sheets to diversify the data used for multimodal learning. This ultimately improved accuracy by up to 12% and reduced gender bias by up to 30%.

III. Problem Statement

A. Multimodal Deep Learning

This project uses data from earnings conference calls to make predictions. This data comes in two forms, Audio recordings of the actual speech during the meeting and text transcripts of everything that was said. We intend to use both of these data sources to train the model on both the literal words spoken and the tone and pitch of the person speaking. We believe that combining these sources will allow our model to more accurately interpret what the speakers are saying in the earnings conference calls and thus be able to make more accurate predictions of how the stock market will be affected.

B. Volatility

The n-day volatility prediction is the predicted average volatility of a stock price over the following n days.

$$v[0,n] = \ln\left(\sqrt{\frac{1}{n}\sum_{i=1}^{n}(r_i - \bar{r})^2}\right)$$

Where the parameter r_i is the stock return on day i and $\bar{r}$ is the average stock return over n days. The stock return r_i is defined as:

$$r_i = \frac{P_i - P_{i-1}}{P_{i-1}}$$

And the parameter P_i is the adjusted closing price of the stock on day i. For single-day log volatility, we estimate it using the daily log absolute return:

$$v_n = \ln\left(\left|\frac{P_n - P_{n-1}}{P_{n-1}}\right|\right)$$

Where P_n is the adjusted closing price of the stock on day n and P_{n-1} is the adjusted closing price on the previous day. Our multi-task learning objective is to simultaneously predict these two quantities: $v[0,n]$: The average volatility over n days (the main task), and. v_n: The single-day volatility (the auxiliary task).

IV. Technical Approach

We use the Hierarchical Transformer-based Multi-task Learning (HTML) model [3], which allows us to combine text and audio data to predict stock price volatility. It uses a series of transformers to combine and encode the data for multi-task prediction. The process described in the following paragraphs is also depicted in figure 1.

For the audio data, Praat is used for sentence-level encoding of acoustic information and emotion2vec is used for sentence level encoding of emotion information.

For the text data, GLOVE and roBERTa are used to generate word-level encodings, the Sentence Transformers are used to generate sentence-level encodings and OpenAI calls are used to get high level sentiment encoding.

Once all of these types of encodings are generated, every combination of audio and text encoding are concatenated together. These combined features are then used as the input for the multi-task learner.

The training targets of the multi-task learner are the average and single-day volatility predictions for various lengths of time: 3, 7, 15, and 30 days.

The trained model generates average and single-day volatility predictions based on input text and audio from Earnings Conference Calls. The process of training and prediction is depicted in Figure 2.

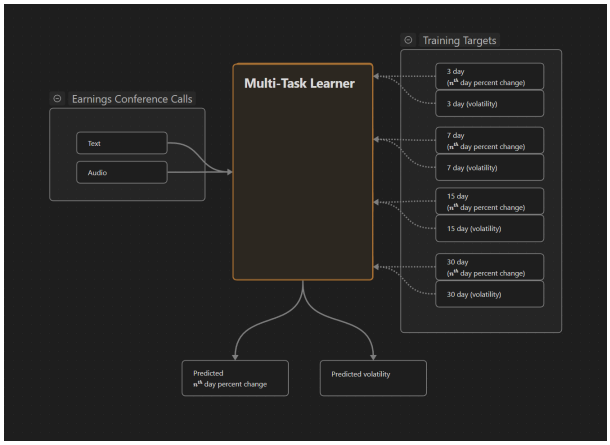

Fig. 2. Training Targets Diagram

V. Experimental setup

A. Data Used

1) Stock Price Data: We first use the Yahoo and Alphavantage APIs to obtain several years of stock price data. The n-day volatility and percent change of each stock is then calculated for 3, 7, 15, and 30 days after the earnings call. Each of the n-day metrics is a separate task for the Multi-Task Learner to train on.

2) Text Data: We experiment with using Glove [12], RoBERTa [13], and BGE [14] fine-tuned on Investopedia to get word embeddings for each sentence in the Earnings Call and MAEC datasets. Each embeddings is labeled with the

pitch	(mean, sd, min, max)
intensity	(mean, min, max)
pulse	(number)
periods	(mean, sd)
Unvoiced Fraction	
voice breaks	(number, degree)
Jitter	(local, local abs, rap, ppq5)
Shimmer	(local, dB, apq3, apq11, dda)
autocorrelation	(mean)
NHR	(mean)
HNR	(mean)
Length	

TABLE I
THE 27 AUDIO FEATURES GENERATED BY PRAAT

company/ticker, date, and sentence number for its original sentence.

3) Audio Data: Emotion2vec and the Praat methods in the Parselmouth library allows us to do audio processing on the .mp3 files and extract the useful audio features. We then using these relevant audio features for training rather then the larger original audio files.

B. Encoding Methods

1) GloVe: GloVe is a encoding model that generates static word to word embeddings. The model essentially generates a matrix who indices are the words uses and whose elements are a probability radio of the how related the two words are to each other.

2) RoBERTa: The RoBERTa encoding model generates contextualized embeddings that take into account not only the word in question, but also how the rest of the context around the words effect the meaning. These embeddings are also dynamic so a given embedding might not correspond to a specific words, but rather an idea or concept.

3) SentenceTransformer: Huggingface's SentenceTransformer model generates embeddings based on sentences rather then words. Rather then encoding the words or token, the SentenceTransformer pools all that information together into one encoding for the entire sentence. The theory is that combining this information together will result in an embedding that better represents the complex financial ideas contain in that CEO's sentences.

4) OpenAI - Sentiment Classification: Using OpenAI for Sentiment Classification generates sentence level encodings for each line in the Earnings Call Meet, but rather then creating a a vector representation in an abstract embedding space, it encode the sentence as one of 10 sentiments (Positive Outlook, Negative Outlook, Neutral/Factual, Cautiously Optimistic, Concerned/Uncertain, Strong/Confident, Weak/Insecure, Growth-Oriented, Cost-Conscious, Risk-Acknowledging).

5) Praat: The python Parselmouth library is a Python wrapper of the open source Praat C++ library. Praat is an audio processing library that gives measurements of Phonetics and Voice analysis on audio files. The types of audio features generated by Praat are shown in the table below.

6) emotion2vector: emotion2vec is an emotion recognition model. It takes in raw audio files and then returns a probability distribution of 9 emotions: Angry, Disgusted, Fearful, Happy, Sad, Surprised, Neutral, Other, Unknown. We then use Softmax on this output to get a one-hot encoding of the emotion vector.

C. Research Questions

Are main questions is how effective are the embedding methods at representing financial data from earning conference call recordings. Our text embedding systems range from word level, to contextual tokens, to sentence level. Our audio embedding system range from signal processes to emotional predictions. The six embedding systems range in complexity, and our objective is to find the embedding system that add them most explanatory power.

D. Experiments

This project experiments with pairs of embedding methods. We train models on every of text embedding system and every audio system combination. Since each meeting extracts different information from the data, our hope is to find a pairing that has larger coverage of the data then any embedding system on its own.

E. Hyperparameters

The hyperparameters for our HTML model are listed in Table II and based off of the ones used in [3]. Our only modification is to decrease the learning rate by an order of magnitude since the models were converging too quickly.

Hyperparameters	Value
max_norm	1.0
num_epochs	10
warmup_steps	1000
batch_size	4
dropout	0.5
heads	2
depth	2
lr	2e-6

TABLE II
HTML HYPERPARAMETERS

VI. DATA

A. Earnings Call Dataset

The Earnings Call Dataset [1] is a collection of data from earning conference calls of various companies on multiple dates. This dataset include 572 earning conference call meetings from the year 2017. Each earning conference calls has a transcript of the meeting in the form of a .txt file and a folder of .mp3 files with the audio recoding of the meeting. The .txt files are formatted which each sentence of the meeting on its own line. The .mp3 files are each an audio recoding of one sentence in the meeting corresponding to the sentences in the .txt transcript. We uses the Praat methods from the Parselmouth Python library to extract audio features from these .mp3 files.

	EC Dataset					MAEC 2015					MAEC 2016					
Model	$\overline{MSE}$	MSE_3	MSE_7	MSE_{15}	MSE_{30}	$\overline{MSE}$	MSE_3	MSE_7	MSE_{15}	MSE_{30}	$\overline{MSE}$	MSE_3	MSE_7	MSE_{15}	MSE_{30}	**Overall Average**
Vpast	1.120	2.990	0.890	0.420	0.230	0.696	1.599	0.560	0.339	0.284	0.691	1.544	0.571	0.362	0.288	0.836
Price LSTM	0.750	1.970	0.460	0.320	0.240	-	-	-	-	-	-	-	-	-	-	0.750
BiLSTM + ATT	0.740	1.980	0.490	0.320	0.220	-	-	-	-	-	-	-	-	-	-	0.740
HAN(Glove)	0.600	1.430	0.440	0.310	0.200	-	-	-	-	-	-	-	-	-	-	0.600
MDRM(Audio)	0.580	1.370	0.420	0.290	0.200	0.630	1.425	0.488	0.320	0.285	0.618	1.426	0.488	0.311	0.259	0.609
MDRM(Text+Audio)	0.580	1.370	0.420	0.290	0.200	0.514	1.194	0.440	0.231	0.231	0.579	1.287	0.479	0.300	0.249	0.558
HTML(Text)	0.460	1.180	0.380	0.250	0.170	-	-	-	-	-	-	-	-	-	-	0.460
HTML(Text+Audio)	0.400	1.050	0.340	0.230	0.160	-	-	-	-	-	-	-	-	-	-	0.400
VolTAGE	0.310	0.630	**0.260**	0.200	0.130	-	-	-	-	-	-	-	-	-	-	0.310
KeFVP	**0.300**	**0.610**	0.291	**0.183**	**0.114**	**0.204**	**0.418**	**0.187**	**0.122**	**0.087**	0.318	0.445	0.279	0.303	0.177	**0.274**
glove + Praat	0.390	0.743	0.384	0.248	0.186	0.292	0.517	0.294	0.196	0.161	0.224	0.371	0.222	0.166	0.138	0.302
Roberta + Praat	0.344	0.672	0.326	0.210	0.168	0.278	0.514	0.268	0.183	0.145	0.206	**0.346**	0.201	0.150	0.126	0.276
bge + Praat	0.348	0.676	0.314	0.223	0.177	0.277	0.510	0.270	0.178	0.148	**0.204**	**0.346**	0.197	0.148	0.124	0.276
bge_base + Praat	0.355	0.692	0.333	0.233	0.161	0.284	0.526	0.268	0.184	0.158	0.210	0.358	0.209	0.148	0.123	0.283
investopedia + Praat	0.363	0.693	0.346	0.248	0.165	0.277	0.498	0.269	0.194	0.145	0.209	0.350	0.202	0.156	0.128	0.283
glove + Praat + emotion2vec	0.388	0.717	0.372	0.255	0.208	0.291	0.530	0.283	0.190	0.162	0.222	0.363	0.223	0.163	0.138	0.300
Roberta + Praat + emotion2vec	0.368	0.723	0.329	0.245	0.176	0.281	0.514	0.272	0.183	0.153	0.209	0.347	0.200	0.160	0.129	0.286
bge + Praat + emotion2vec	0.356	0.678	0.346	0.221	0.179	0.275	0.497	0.263	0.179	0.159	0.205	0.350	0.200	0.146	**0.122**	0.278
bge_base + Praat + emotion2vec	0.366	0.711	0.329	0.233	0.191	0.277	0.503	0.263	0.193	0.150	0.214	0.366	0.205	0.157	0.127	0.286
investopedia + Praat + emotion2vec	0.359	0.675	0.335	0.234	0.192	0.278	0.508	0.267	0.184	0.153	0.206	0.352	0.195	**0.145**	0.130	0.281
glove + Praat + sentiment																
Roberta + Praat + sentiment																
bge + Praat + sentiment																
bge_base + Praat + sentiment		currently running					currently running					currently running				
investopedia + Praat + sentiment																

Fig. 3. The full results for all the models we trained. Top group of rows is existing models, the ones below are ours. Columns are grouped by dataset, and display the MSE for 3, 7, 15, and 30-day volatility (MSE_3, MSE_7, MSE_{15}, MSE_{30}), as well as the mean MSE for all intervals ($\overline{MSE}$). Numbers in bold are the best result overall, numbers underlined are the best result for our models.

B. MAEC Dataset

The MAEC: A Multimodal Aligned Earnings Conference Call Dataset for Financial Risk Prediction [2] is a collection of data from earning conference calls of various companies on multiple dates. This dataset include 3443 earning conference call meetings from the years 2016-2017. Each earning conference calls incudes a .txt file with a transcript of the meeting and a .csv file with data about the audio features. Like the Earning Call Dataset, each sentence in the .txt transcript file is placed to its own line. The .csv file contains a row of audio features for each sentence in the .txt transcript.

VII. Results

Comining both the Earnings Call and MAEC datasets gives us over 4000 meetings to work with, each with around 500 sentences. With this data, we trained 180 different models. This includes 15 different embedding method combinations (created from 5 text embedding methods, 2 audio embedding methods, and using OpenAI GPT-4o mini sentiment embeddings), 4 targets (3, 7, 15, and 30 day volatility), and 3 years of data (2015, 2016, and 2017).

All the results for our project can be seen in Figure 3. It is clear that KeFVP is still state of the art in 2015 and 2017, but our models do achieve state of the art performance in 2016. For audio embeddings, only using Praat worked best for 3-day volatility, but adding emotion2vec improved performance for longer time frames. For text embeddings, Investopedia and BGE seem to work the best across the board.

We also found interesting results in other years, despite not being state of the art. In 2017, using only Praat audio embeddings provided the best performance for all volatility intervals. For text embeddings, RoBERTa and BGE take the crown for best performance. In 2015, using Praat and emotion2vec for audio embeddings works better for shorter volatility intervals, but Praat only works better for longer volatility intervals. Similar to 2017, BGE and RoBERTa are the best performing text embeddings, with BGE performing better in the short term and RoBERTa only performing best for the 30-day volatility interval.

Looking at all years, we found that BGE + Praat and RoBERTa + Praat are tied for best performance overall, and only very slightly worse (+0.73% error) than the current state of the art, KeFVP. It is interesting to note how MSE decreases across the board as the volatility interval lengthens. This makes sense, however, since stocks tend to be more stable in the long term, which makes volatility easier to predict.

VIII. Conclusions

To summarize, our project uses multimodal learning to predict stock price volatility from transcripts and audio recordings of earnings calls. We tested different combinations of text and audio embeddings with the HTML model. Our novel contributions include applying a pretrained emotion2vec model to extract sentiment from audio and using the OpenAI GPT-4o mini LLM to perform sentiment analysis on the transcripts. We tested against 3, 7, 15, and 30 day volatility targets to assess model performance in short and long timeframes, and found that RoBERTa and BGE text embeddings with Praat audio embeddings result in the best model performance across much of the data. We were even able to beat prior state of the art performance in multimodal stock volatility prediction in 2016 earnings call meetings.

While we achieved many successes in this project, we also faced some challenges. We had issues collecting dividend-adjusted closing prices for some stocks, which meant we were missing volatility for those stocks. We tried emailing authors

of similar works such as KeFVP, but they also faced the same challenges. We also faced significant setbacks due to the ongoing Beocat maintenance this semster, which made it difficult to train our models. In fact, we are still running experiments with the OpenAI sentiment analysis and unfortunately cannot include the results in this report. However, we intend to continue working on this project after this class is over, with many ideas in mind for future work as described below.

IX. Future Work

A. Using Spectrograms for Audio Analysis

Instead of processing the audio files as-is, it may be beneficial to convert the audio into spectrogram images that can be passed into a Convolutional Neural Network. Considering how much more developed image classification models are than audio classification models, image models may be better at extracting features or emotion from images of audio data. While we didn't see improvement by using emotion2vec, works such as [15] have been able to achieve up to 68% accuracy in classifying emotion from spectrograms.

B. Large Language Models

While we only tried one prompt for our experiments with GPT-4o mini sentiment classification, it may be worth experimenting with different prompts. Works such as [16] have shown success in prompt engineering for financial sentiment analysis, and prior experiments from the homework have shown that a good prompt can significantly improve accuracy.

We also had the idea of expanding our usage of LLMs and leveraging them for more advanced reasoning. By feeding in the transcipts of each meeting to the LLM, we can then ask it questions to extract additional information from the meetings. [17] applied LLMs to earnings calls to extract sentiment, temporal orientation, and language clarity from them, but it could be interesting to ask an LLM more general questions such as the health of the company or whether the stock price will go up or down.

C. Retrieval Augmented Generation

Another way to improve LLM responses is to employ Retrieval Augmented Generation (RAG). Using this would allow us to supply financial knowledge to the model for more accurate and informed answers, similar to KeVFP [4]. Works including [18] and [19] have seen greater accuracy from LLMs using RAG with earnings calls.

D. Incorporating News about Companies

While earnings calls provide useful information about the internal perceptions of a company, it may also be beneficial to include external perceptions in the dataset. News articles about the company can provide this information. [20] used Sentiment Analysis to find that news articles are a significant predictor of stock price volatility, which is backed up by [21].

E. New Targets

Another interesting approach could be to look at a different target from stock price volatility altogether. Stock price movement is a binary indicator of whether the stock price moved up or down, and is a much simpler target to learn. Works such as [22], [23], and [24] have found success using earnings calls to predict stock price movement.

X. Data & Code

A. Original Earnings Call Dataset

https://github.com/GeminiLn/EarningsCall_Dataset

B. MAEC Dataset

https://github.com/Earnings-Call-Dataset

C. Target Volatility Data

https://github.com/hankniu01/KeFVP/tree/main/price_data

D. Code

https://github.com/JamesChapmanNV/Earnings_call_project

References

[1] Y. Qin and Y. Yang, "What you say and how you say it matters: Predicting stock volatility using verbal and vocal cues," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics* (A. Korhonen, D. Traum, and L. Màrquez, eds.), (Florence, Italy), pp. 390–401, Association for Computational Linguistics, July 2019.

[2] J. Li, L. Yang, B. Smyth, and R. Dong, "Maec: A multimodal aligned earnings conference call dataset for financial risk prediction," in *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, CIKM '20, (New York, NY, USA), p. 3063–3070, Association for Computing Machinery, 2020.

[3] L. Yang, T. L. J. Ng, B. Smyth, and R. Dong, "Html: Hierarchical transformer-based multi-task learning for volatility prediction," in *Proceedings of The Web Conference 2020*, WWW '20, ACM, Apr. 2020.

[4] H. Niu, Y. Xiong, X. Wang, W. Yu, Y. Zhang, and W. Yang, "KeFVP: Knowledge-enhanced financial volatility prediction," in *Findings of the Association for Computational Linguistics: EMNLP 2023* (H. Bouamor, J. Pino, and K. Bali, eds.), (Singapore), pp. 11499–11513, Association for Computational Linguistics, Dec. 2023.

[5] L. Yang, J. Li, R. Dong, Y. Zhang, and B. Smyth, "Numhtml: Numeric-oriented hierarchical transformer model for multi-task financial forecasting," 2022.

[6] R. Sawhney, P. Khanna, A. Aggarwal, T. Jain, P. Mathur, and R. R. Shah, "VolTAGE: Volatility forecasting via text audio fusion with graph convolution networks for earnings calls," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (B. Webber, T. Cohn, Y. He, and Y. Liu, eds.), (Online), pp. 8001–8013, Association for Computational Linguistics, Nov. 2020.

[7] Y. Cao, Z. Chen, Q. Pei, N. J. Lee, K. P. Subbalakshmi, and P. M. Ndiaye, "Ecc analyzer: Extract trading signal from earnings conference calls using large language model for stock performance prediction," 2024.

[8] Y. Cao, Z. Chen, Q. Pei, F. Dimino, L. Ausiello, P. Kumar, K. P. Subbalakshmi, and P. M. Ndiaye, "Risklabs: Predicting financial risk using large language model based on multi-sources data," 2024.

[9] R. Sawhney, A. Aggarwal, and R. R. Shah, "An empirical investigation of bias in the multimodal analysis of financial earnings calls," in *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* (K. Toutanova, A. Rumshisky, L. Zettlemoyer, D. Hakkani-Tur, I. Beltagy, S. Bethard, R. Cotterell, T. Chakraborty, and Y. Zhou, eds.), (Online), pp. 3751–3757, Association for Computational Linguistics, June 2021.

[10] S. Wang, T. Ji, J. He, M. ALMutairi, D. Wang, L. Wang, M. Zhang, and C.-T. Lu, "AMA-LSTM: Pioneering robust and fair financial audio analysis for stock volatility prediction," in *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 6: Industry Track)* (Y. Yang, A. Davani, A. Sil, and A. Kumar, eds.), (Mexico City, Mexico), pp. 379–386, Association for Computational Linguistics, June 2024.

[11] P. Mathur, M. Goyal, R. Sawhney, R. Mathur, J. Leidner, F. Dernoncourt, and D. Manocha, "DocFin: Multimodal financial prediction and bias mitigation using semi-structured documents," in *Findings of the Association for Computational Linguistics: EMNLP 2022* (Y. Goldberg, Z. Kozareva, and Y. Zhang, eds.), (Abu Dhabi, United Arab Emirates), pp. 1933–1940, Association for Computational Linguistics, Dec. 2022.

[12] J. Pennington, R. Socher, and C. Manning, "GloVe: Global vectors for word representation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (A. Moschitti, B. Pang, and W. Daelemans, eds.), (Doha, Qatar), pp. 1532–1543, Association for Computational Linguistics, Oct. 2014.

[13] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "Roberta: A robustly optimized bert pretraining approach," 2019.

[14] S. Xiao, Z. Liu, P. Zhang, and N. Muennighoff, "C-pack: Packaged resources to advance general chinese embedding," 2023.

[15] A. Satt, S. Rozenberg, and R. Hoory, "Efficient emotion recognition from speech using deep learning on spectrograms," in *Interspeech 2017*, pp. 1089–1093, 2017.

[16] R. Ahmed, S. A. Rauf, and S. Latif, "Leveraging large language models and prompt settings for context-aware financial sentiment analysis," in *2024 5th International Conference on Advancements in Computational Sciences (ICACS)*, pp. 1–9, 2024.

[17] T. Cook, S. Kazinnik, A. Hansen, and P. McAdam, "Evaluating local language models: An application to bank earnings calls," *Fed. Reserve Bank Kans. City Res. Work. Pap.*, Nov. 2023.

[18] Y. Juan, C.-C. Chen, H.-H. Huang, and H.-H. Chen, "Co-trained retriever-generator framework for question generation in earnings calls," *arXiv preprint arXiv:2409.18677*, 2024.

[19] B. Zhang, H. Yang, T. Zhou, M. Ali Babar, and X.-Y. Liu, "Enhancing financial sentiment analysis via retrieval augmented large language models," in *Proceedings of the fourth ACM international conference on AI in finance*, pp. 349–356, 2023.

[20] J.-L. Seng and H.-F. Yang, "The association between stock price volatility and financial news – a sentiment analysis approach," *Kybernetes*, vol. 46, pp. 1341–1365, Jan 2017.

[21] A. Atkins, M. Niranjan, and E. Gerding, "Financial news predicts stock market volatility better than close price," *The Journal of Finance and Data Science*, vol. 4, no. 2, pp. 120–137, 2018.

[22] L. E. Solberg and J. Karlsen, "The predictive power of earnings conference calls : predicting stock price movement with earnings call transcripts," Master's thesis, Norwegian School of Economics Bergen, 2018.

[23] Z. Ma, G. Bang, C. Wang, and X. Liu, "Towards earnings call and stock price movement," 2020.

[24] S. Medya, M. Rasoolinejad, Y. Yang, and B. Uzzi, "An exploratory study of stock price movements from earnings calls," 2022.

APPENDIX

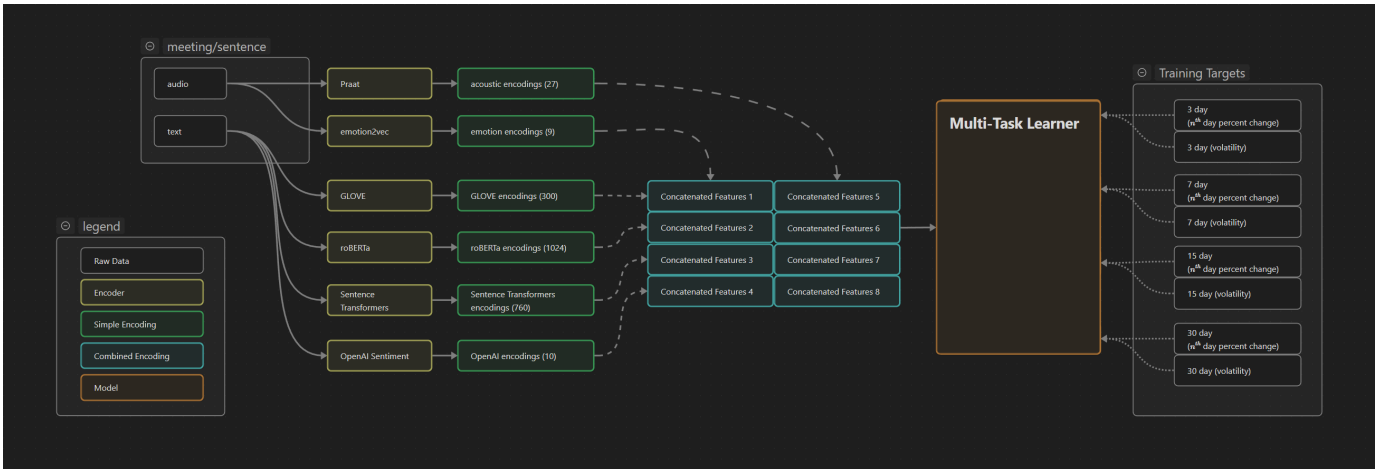

Fig. 4. The full architecture of our experiment pipeline. This is a combination of Figures 1 and 2 from earlier in the paper.