Forecasting Stock Volatility following earnings calls release of S&P1500 companies

Abstract

Earning calls have a substantial impact on the volatility in the stock price and can be used to analyse a stock's risk level. Along with textual features, the vocal features and voice tones while delivering the earning call are also depictive of the firm's performance. This approach uses an audio encoder and text encoder for obtaining the embeddings and features; this combined encoded representation has been used along with past volatility values to predict the volatility of the stock over the next t=3 days. The model has been built on the earning calls data available for S&P 1500 companies.

Method Overview

1) Data Acquisition: Here, the MAEC(A Multimodal Aligned Earnings Conference Call Dataset for Financial Risk Prediction) Dataset [1] has been used. This consists of text-audio paired earnings call, based on S&P 1500 companies. The MAEC dataset consists of multiple earning calls each consisting of a transcripts text file and low-level audio features, where each earning call is divided into multiple audio clips - and the text and audio features have been aligned for each of these audio clips. For this task, a smaller subset of data has been used consisting of earning calls made between 1 st Jan 2018 to 31 st June 2018. The ground values consisting of the closing prices 3 days before and 3 days after the earnings call was scraped from Yahoo Finance[2]. After filtering a total of 298 earning calls from 238 different companies are present in the dataset. Assuming that the longest document has N sentences or audio clips, the embeddings and audio features for all earning calls have been zero-padded so that the final embedding dimension for each call is N x 768 and the audio feature dimension is N x 29(For this subset of data max audio clips, N = 444)

2) Text Encoder: To extract text embeddings or features for the transcripts, pre-trained Sentence-BERT sentence embeddings which were developed using Siamese BERT-networks [3], have been used to represent each sentence in any earning call. Hence, each sentence is represented as a 768-dimension vector. For encoding this, the embeddings are passed through a BiLSTM layer (units = 128, return_sequences = true) and each hidden state is then passed to a Dense layer (units = 1). Hence the final text encoder output for each earning call has dimension N x 1, where N is the maximum audio clips in a call. The BiLSTM layer allows us to extract the sequential context of each sentence in the entire transcript and hence can be used to model the text embeddings for the transcript.

3) Audio Encoder: As the MAEC dataset has audio clip aligned audio features for each sentence in the earning call, these have been used as features for each sentence. This consists of 29 sound acoustic features such as pitch, intensity, voice breaks, etc. [4]. For each earning call, the audio features are represented as N x 29, where N is the maximum number of audio clips in an earning call, as mentioned in the Text Encoder section. For audio encoding also, the feature vector is passed through a BiLSTM layer (units=64, return_sequences = true) and each hidden state corresponding to N audio clips in an earning call is passed on to a Dense layer(units =1). The final audio encoder output for each call is N x 1.

4) Feature Fusion: The outputs from both the encoders have been combined to form a N x 2 feature value for each earning call, Two approaches have been tested one using a Convolution layer (4 x 2 kernel) and one using an LSTM over this fused input. In both approaches, a Dense Layer(units=1) is applied on the output. The layers have been reshaped wherever necessary.

5) Past Volatility: Since this is a prediction problem, past values for volatility taken into account can help to make the prediction better. Here, the volatility of the stock over a period of 3 days prior to the earning call is input as a feature to the model. The output of feature fusion and past volatility is combined to give a final output - prediction of stock volatility for 3 days post earning call. For computing the past volatility and the ground truth values for volatility, equation 1 from [1] has been used.

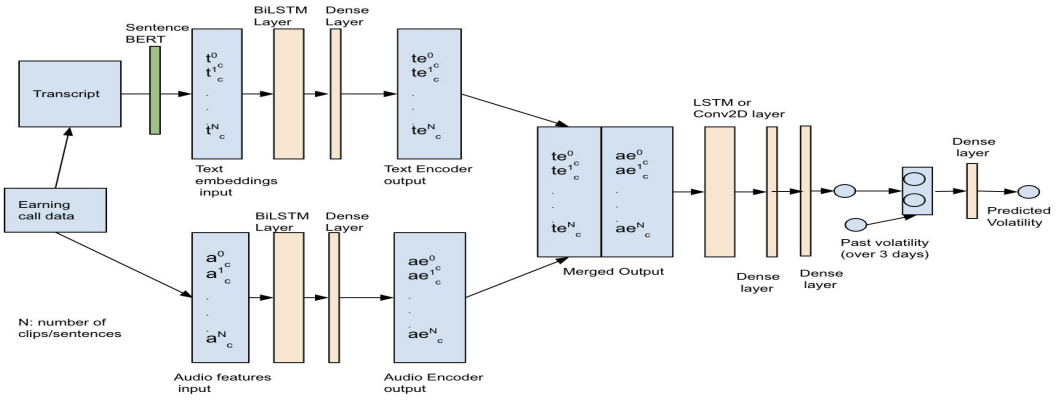

Fig1: Overview of the Architecture Used

Training and Setup: As the data is time-dependent, random shuffling might cause data leakage. The list of earning calls have been sorted according to the date to prevent this. The model has been trained on 200 earning calls and tested on 98 earning calls. All the layers have been implemented using TensorFlow and Keras API. Each model was trained for 25 epochs using the Adam optimizer and the loss metric was set as mean_squared_error. As the feature embeddings and volatility values consist of a large number of negative values, the tanh and linear activation functions have been used.

Method	Mean Squared Error on Test Data	**Performance and Inferences**
Using Convolution Layer over combined audio-text features with past volatility feature	1.9354	The table alongside denotes the mean squared error obtained using 2 different networks over the merged-encoded output. Using LSTM layer performs better than Convolution layer, attributing to the fact that LSTM is able to model the sequential nature of the earning call text and audio better.
Using LSTM Layer over combined audio-text features with past volatility feature	0.8470	

Table 1: Performance of the Model

Conclusion

Using Bidirectional LSTMs to encode the text and audio features of an earning call, followed by feature fusion and using past volatility values can be successfully used to predict the volatility in the stock following an earning call. This also takes into account the acoustics along with the text as that is also a factor for analysing company performance. Further improvements can include developing high-level features from the audio data and also testing the performance of other sentence embedding methods to evaluate model performance.

The repository for the approach can be found here: https://github.com/vishwa27yvs/Stock-Volatility-Forecasting

References:

[1] Jiazheng Li, Linyi Yang, Barry Smyth, and Ruihai Dong. 2020. 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). Association for Computing Machinery, New York, NY, USA, 3063–3070. DOI:https://doi.org/10.1145/3340531.3412879

[2] https://in.finance.yahoo.com/

[3] Reimers, N., Gurevych, I.: Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing. pp. 3982–3992. EMNLP '19, Hong Kong, China (2019), https://github.com/UKPLab/sentence-transformers

[4] Audio Features Used: 'Mean pitch', 'Standard deviation', 'Minimum pitch', 'Maximum pitch','Mean intensity', 'Minimum intensity', 'Maximum intensity','Number of pulses', 'Number of periods', 'Mean period', 'Standard deviation of the period', 'Fraction of unvoiced','Number of voice breaks', 'Degree of voice breaks', 'Jitter local','Jitter local absolute', 'Jitter rap', 'Jitter ppq5', 'Jitter ddp', 'Shimmer local', 'Shimmer local dB', 'Shimmer apq3', 'Shimmer apq5', 'Shimmer apq11', 'Shimmer dda', 'Mean autocorrelation', 'Mean NHR', 'Mean HNR', 'Audio Length'