Earnings Call Sentiment: Informed Trading Strategy

Antoine Munier
EPFL
antoine.munier@epfl.ch

Micah Yo Maheo
EPFL
micah.maheo@epfl.ch

Robin Jaccard
EPFL
robin.jaccard@epfl.ch

Jeremy Di Dio
EPFL
jeremy.didio@epfl.ch

June 1, 2024

Abstract

This study explores the application of machine learning techniques to analyze the sentiment of earnings calls and develop trading strategies based on those sentiments. With access to a dataset comprising 18,755 earnings call transcripts, we implement various NLP models to extract sentiment signals. Our approach includes state-of-the-art machine learning algorithms, including transformers like FinBERT, optimized for financial contexts. The sentiments derived from earnings calls are then used to predict short-term stock returns, aiming to capture market reactions post-earnings announcements. Furthermore, we experiment with multiple trading strategies leveraging these sentiment signals to guide investment decisions. The performance of these strategies is assessed by back-testing on historical data, demonstrating the potential of sentiment analysis as a predictive tool.

Contents

1 Introduction

1.1 Earnings Calls

Earnings calls usually occur after a public company releases its financial statements for the current period. These are often in the form of discussions, where the management team of the company discusses in more detail the content of these financial statements, and moreover, they respond to investors' questions. These discussions are a major event and have often been used as indicators of a company's well-being and its prospective future [1, 2]. Typically, these earnings calls were manually analyzed to infer trading decisions [3]. This process is lengthy and meticulous but, fortunately, today's new technologies enable the automatic extraction of useful information from these earnings calls.

1.2 Financial Tone Analysis

With recent advances in computational technologies, there has been a significant increase in exploring various approaches to analyze earnings calls. Specifically, sentiment analysis can now be applied to these discussions to gain insights into the overall tone of the call. It has been researched and demonstrated that the specific tone of an earnings call can offer valuable information about the short-term future of the company [4]. Furthermore, recent advancements in natural language processing (NLP) techniques have the potential to significantly enhance the accuracy of tone analysis models, thereby providing more nuanced and insightful interpretations of these discussions [5]. The famous transformer architecture [6] has reshaped the landscape of natural language processing (NLP), achieving remarkable improvements over traditional methods. It did not take long for various players in the financial sector to adopt this technology. One standout example is BloombergGPT [7], a 50-billion parameters Large Language Model, especially designed for analyzing and discuss financial data.

1.3 Our Research

Building upon the established groundwork, this project aims to develop various trading strategies based on the distinct tone of each earnings call. Our approach primarily relies on a dataset that includes transcripts from numerous earnings calls. We then apply a sentiment extraction model to extract the nuanced tone encapsulated in each transcript. Then, armed with these insights and supplementary data, we design, implement, and evaluate multiple trading strategies.

2 Data

2.1 The Dataset

For this project, we used a freely available dataset composed of 18,755 earnings calls scraped from Motley Fool. Each row of the dataset contains the date, the quarter, the exchange, the ticker, and the transcript of the earnings call. Figure 3 shows an example of such earnings call transcript.

The dataset timeframe is starting from 2017-11-03 up to 2023-02-23.

2.2 Preprocessing

In order to correctly use the dataset, we performed some preprocessing and cleaning steps. These various selections and transformations are described below:

1. **Exchange selection**: We decided to focus on only two different exchanges (NASDAQ & NYSE) as they represent 99.8% of the dataset. This choice was made in order to discard exchanges with not enough data sample to infer correct assumptions.

2. **Cleaning of features**: We cleaned different features that contained multiple pieces of information in a single column. For example, the quarters were described as "YEAR-QUARTER" and we split it into two columns, Year and Quarter. We also discarded data points that had inconsistencies in the date.

3. **Transcript processing**: To further ease our future sentiment prediction, we performed different transformations to our transcripts. Namely, we removed punctuation and stop-words, converted to lowercase, and performed tokenization.

4. **Hour filtering**: To improve our future trading strategies, we chose to filter our dataset to include only earnings calls announced either before the market opens (9:30 am) or after the market closes (4:00 pm).

Following these preprocessing steps, our dataset comprised 10'319 unique earnings calls from $1'869$ companies.

2.3 Exploratory Data Analysis

In Appendix B, we outline several visualizations related to the exploratory analysis of the dataset.

2.3.1 Temporal Analysis

As highlighted in Figure 4, certain periods witness a significantly higher number of earnings calls being released. This phenomenon is primarily attributed to the common practice among companies of issuing quarterly earnings reports. Typically, the earnings season starts one or two weeks after the conclusion of each quarter (December, March, June, and September). Consequently, we expect a surge in the release of earnings by the majority of public companies during early to mid-January, April, July, and October, explaining the four pronounced peaks per year.

Figure 5 shows the specific hours at which earnings calls were released. As explained in Section 2.2, we have filtered out the earnings calls that occurred during the trading session. Consequently, we observe that earnings reports are announced only before and after market hours, mostly at 8am, 8:30am, 9am, 4:30pm, and 5pm.

2.3.2 Textual Analysis

Initially, we examined the word count distribution in earnings reports post pre-processing. On average, these reports contains around $55'000$ words per transcript, as depicted in Figure 6. However, the length of these reports does not necessarily correlate with the clarity of sentiment expressed. Upon manual review of several earnings calls, it became evident that such reports do not consistently show a clear sentiment, particularly in cases of unfavorable results, where speakers may choose not to emphasize about the negativity of the report. This observation is further supported by the most frequently occurring words in the transcripts, as illustrated in Figure 7. Notably, among the top 50 words, none distinctly express a clear sentiment. This will poses a first challenge for our subsequent analysis, described in Section 3, which aims to extract the sentimental polarity of each transcript.

2.4 Additional Data

Finally, to already prepare for the subsequent trading strategies tasks, we decided to scrape and add different data type to each transcripts. The following list enumerates the different pieces of information added to each earnings call:

- **Interest rate**: we added the interest rate (*CBOE Interest Rate 10 Year Treasury Note*) at the time of the earnings announcement, recognizing the significant influence of monetary policy on market sentiment and investor behaviour. Additionally, we consider the three-month returns on the interest rate leading up to the earnings event, offering insights into the trend during that period.
- **S&P 500 returns**: We incorporated the linear returns of the S&P 500 before the earnings call release for three different time horizons: one week, one month, and three months.
- **Past asset returns**: We integrated the returns of the corresponding asset on various time scales preceding the earnings announcement, including dividends in the computations. This includes the returns on the week, the month, and the three months preceding the earnings event. By incorporating these stock-specific metrics, we capture the recent performance of the company's stock.
- **Future asset returns**: For training and testing our trading strategies, we included future returns of the corresponding asset across various periods: overnight, one day, one week, and one month. The overnight period reflects the asset return during the market's closed hours, which includes the time when the earnings call takes place. For instance, if results are disclosed after Monday's market close, the overnight return spans from Monday's close to Tuesday's opening. Similarly, if results are announced Tuesday morning, overnight returns are computed between Monday's close and Tuesday's opening.

3 Methods

In this section, we present our methodology to first extract the specific sentiment of each earnings call and second predict future returns of the corresponding company.

3.1 Sentiment Analysis

In our specific context, the aim of our tone analysis is to determine whether a particular earnings call reflects positive or negative outcomes. Instead of using a simple binary classification, we aimed to incorporate a probability score that provides a more nuanced understanding between the two categories. We present in the following sections an extensive model comparison on a specifically selected benchmark.

3.1.1 Benchmark

Since our current dataset is unlabeled, comparing the results of different models on this dataset is challenging. Therefore, we decided to use another labeled dataset as a benchmark to assess the differences between the models. We selected the *FinancialPhraseBank* dataset [8] for this purpose. This dataset consists of over 4800 financial news headlines classified into three categories: Negative, Neutral, or Positive. To compare the performance of different models, we report Precision, Recall, and F1-score.

3.1.2 Models Selection

The subsequent paragraphs describe the different models we decided to explore. We selected three basic models and two more complex.

Bag of Words (BoW) The Bag of Words (BoW) model is a fundamental technique for text representation. This model simplifies text processing by treating a document as a collection of individual words, disregarding syntax and word order.

- **Mathematical Formulation**: In BoW, each document d in a corpus D is represented as a vector v_d in a high-dimensional space $\mathbb{R}^N$, where N is the size of the vocabulary. The i-th component of v_d is the count $n_{i,d}$ of the i-th word in the vocabulary within document d.

$$v_d = [n_{1,d}, n_{2,d}, \ldots, n_{N,d}]$$

- **Vector Space Model**: This representation allows for straightforward computation of document similarity using metrics like cosine similarity. However, the high dimensionality can lead to sparsity issues, and the model fails to capture semantic relationships between words.

Term Frequency-Inverse Document Frequency (TF-IDF) TF-IDF enhances the BoW model by weighting terms based on their importance within a document relative to the entire corpus, addressing some of the limitations of simple term frequency.

- **Term Frequency (TF)**: The term frequency $\text{tf}(t, d)$ of term t in document d is defined as the ratio of the count of t in d to the total number of terms in d.

$$\text{tf}(t, d) = \frac{n_{t,d}}{\sum_k n_{k,d}}$$

- **Inverse Document Frequency (IDF)**: The inverse document frequency $\text{idf}(t, D)$ measures the rarity of term t across all documents D. It is calculated as:

$$\text{idf}(t, D) = \log\left(\frac{|D|}{|\{d \in D : t \in d\}|}\right)$$

- **TF-IDF Calculation**: The TF-IDF score for a term t in document d is given by:

$$\text{tf-idf}(t, d, D) = \text{tf}(t, d) \times \text{idf}(t, D)$$

GloVe (Global Vectors for Word Representation) GloVe is an advanced unsupervised learning algorithm designed to generate dense word embedding that capture semantic meaning.

- **Co-occurrence Matrix**: GloVe constructs a word-word co-occurrence matrix X, where each entry X_{ij} represents the number of times word i co-occurs with word j within a specified context window across the corpus.

- **Objective Function**: The learning objective is to find word vectors w_i and context vectors $\tilde{w}_j$ such that their dot product approximates the logarithm of the co-occurrence probability:

$$\log(X_{ij}) \approx w_i^T \tilde{w}_j + b_i + \tilde{b}_j$$

  where b_i and $\tilde{b}_j$ are bias terms.

- **Optimization**: The model minimizes a weighted least squares objective function:

$$J = \sum_{i,j=1}^{V} f(X_{ij}) \left(w_i^T \tilde{w}_j + b_i + \tilde{b}_j - \log(X_{ij})\right)^2$$

  The weighting function $f(X_{ij})$ gives less importance to less frequent co-occurrences.

- **Pre-trained Models**: GloVe cannot be trained from scratch in this project due to computational constraints. Therefore, we use pre-trained models available at this link.

DistilRoBERTa Base Finetuned *DistilRoBERTa base* is an open-source fine-tuned model [9]. One of the most popular on the Hugging-Face platform. This model is a fine-tuned version of distilroberta-base. distilroberta-base is a distilled version of the RoBERTa-base model [10]. Model distillation is a technique aimed at reducing the size and complexity of machine learning models while retaining most of their performance capabilities. This process involves training a smaller, compact model, referred to as the student model, to replicate the behavior of a larger, more complex model, known as the teacher model.

FinBERT *FinBERT* [11] is an open-source LLM specifically trained on financial data and reportedly surpasses many approaches in financial NLP tasks. We used this pre-trained model, which adopts an encoder-only architecture [12], leveraging the robustness and fine-tuning potential of BERT for domain-specific sentiment analysis.

3.1.3 Classifiers

Since the BoW, TF-IDF, and GloVe models can only generate numerical representations of textual data and do not directly predict sentiment, we added and trained the following classifiers on top of each of these models:

- **Support Vector Machine (SVM)**: A supervised learning model known for its effectiveness in handling high-dimensional data and performing classification and regression analysis.
- **XGBoost**: An optimized distributed gradient boosting library designed for efficiency, flexibility, and portability, excelling in structured or tabular datasets.
- **Naive Bayes**: A simple yet powerful probabilistic classifier based on applying Bayes' theorem with strong (naive) independence assumptions between features.
- **Multi-Layer Perceptron (MLP)**: A class of feedforward artificial neural network, consisting of at least three layers of nodes, used for complex pattern recognition and classification tasks.

Model	Precision	Recall	F1-Score
BoW & SVM	0.75	0.75	0.75
BoW & XGBoost	0.81	0.80	0.79
BoW & Naive Bayes	0.75	0.76	0.74
BoW & MLP	0.76	0.76	0.76
TF-IDF & SVM	0.78	0.77	0.76
TF-IDF & XGBoost	0.78	0.78	0.77
TF-IDF & Naive Bayes	0.72	0.68	0.61
TF-IDF & MLP	0.74	0.77	0.74
GloVe & SVM	0.75	0.75	0.75
GloVe & XGBoost	0.75	0.76	0.74
GloVe & MLP	0.77	0.77	0.77
GloVe & Logistic Regression	0.75	0.77	0.75
Finbert	**0.90**	**0.89**	**0.89**
RoBERTa	0.86	0.86	0.86

Table 1: Performance Metrics by Model and Text Representation Method

3.1.4 Results

As expected, Table 1 shows that the transformer-based model fine-tuned on financial data, *FinBERT*, outperformed all other methods. Therefore, we selected this model for the subsequent steps of this project.

3.1.5 Application to Our Earnings Calls Dataset

Unlike financial news articles, earnings calls are much longer in their textual representation. Therefore, while the limited context window of FinBERT (512 tokens) was not an issue for the previous model selection analysis, it becomes a problem when computing the sentiment of each earnings call. Since the context window covers only the first 512 tokens of the earnings call, the model relied only on the initial portion, which typically includes a welcome message from the management team. This part is not usually indicative of the call's overall results and may mislead the analysis, especially if the management uses careful word selection to downplay bad results. To address this, we decided to apply a new filtering step to our dataset. Hence, we chose to exclude these sections and focus on the Q&A part, as it likely contains most of the useful information [13].

After this filtering process, we applied our model to the dataset, which provided three logits values, one for each class of Positive, Neutral, and Negative. To better interpret the results, we then applied the Softmax function to convert these logits into a probability distribution over the three classes.

Figure 1 shows the count of each sentiment prediction for our dataset. The sentiment was determined as the one with the highest probability among the three classes.

Figure 1: FinBERT Sentiment Distribution

3.2 Return Prediction Methodologies

The sentiments alone do not yield significant correlations with the returns, a finding that might initially appear counterintuitive. However, this observation highlights a fundamental aspect of market dynamics: sentiment, without broader contextual factors, offers limited predictive power. For instance, in scenarios where economic conditions are unfavorable, and sentiment aligns negatively but less negatively than the projections, investors may interpret such sentiment as positive news, leading to increased stock prices.

This highlights the need of incorporating more features beyond raw sentiment data to predict market reactions effectively. Hence, in our predictive model, we have incorporated additional features beyond sentiment analysis to improve the accuracy of predicting market reactions to earnings. As more precisely described in Section 2.4, we included the returns of the S&P 500 index on multiple time scales preceding the earnings announcement, the interest rate at the time of the earnings announcement, and the returns of the given stock on various time scales following and preceding the earnings announcement, including dividends in the computations.

These diverse features alongside sentiment analysis will help our model to more accurately anticipate market reactions to earnings by capturing contextual factors influencing investor sentiment and behavior.

3.2.1 Data Split

We integrated the sentiment results with the additional features and categorized the regressors into two groups: features without sentiment and features with sentiment. A 60/40 train-test split was chosen based on the sorted timestamps. The training set, ranging from 2017-11-03 to 2021-10-22 (6081 values), covers a broad temporal range to provide the model with diverse data. The test set, from 2021-10-22 to 2023-02-22 (4054 values), includes recent data points, ensuring the model's performance is evaluated on current trends. This split balances the need for sufficient training data with the necessity of a representative test set, particularly given the density of data points in 2021-2023 (Figure 4).

3.2.2 Linear Regression

Description Ordinary Least Squares (OLS) is a supervised technique in the field of machine learning and statistical modelling, particularly used for linear regression analysis. OLS is frequently employed to model and predict financial metrics, such as asset returns based on various economic indicators. Its simplicity and interpretability make it a preferred starting point for assessing the impact of different financial factors on outcomes of interest. The approach of OLS involves minimizing the sum of the squared differences between the observed values and the values predicted by the linear model. Squaring the residuals highlights the impact of larger errors and ensures that both positive and negative deviations from the predicted values are considered equitably.

Feature Choice We employed a thorough strategy to explore all feature combinations by iteratively running regressions on all possible sets of features, both incorporating and excluding sentiment variables. For second-order interactions, we ensured to include all corresponding individual terms. To mitigate (near) multicollinearity, we examined the correlation values as well as Variance Inflation Factors (VIF), making sure to exclude the neutral sentiment due to its inherent correlation with the two other sentiments.

Selection Criteria We chose the *adjusted* R^2 metric to select the optimal models in our training phase. Unlike R^2, which can be misleading by always increasing with more predictors, adjusted R^2 penalizes unnecessary ones, giving a more accurate assessment and helping to avoid overfitting. Furthermore, this goodness-of-fit approach allows for a pragmatic consideration of multicollinearity. While this affects the coefficient estimates (e.g. p-values), it does not directly impact predictions nor goodness-of-fit statistics. Since the primary objective of the regression is predictive accuracy, we kept this metric.

Finally, we selected the regression models with the highest adjusted R^2 for each of the four dependent variables across the four feature sets, resulting in a total of 16 models. The results are shown in Table 2.

The inclusion of sentiment as a feature consistently enhances the models' explanatory power. Though relatively small to begin with, the in-sample adjusted R^2 values are higher for the sentiment-inclusive

models (S and S w.i.) compared to their non-sentiment counterparts (NS and NS w.i.) across all time intervals.

Table 2: OLS Adjusted R-squared Values for Time Intervals (%)

	Features			
Time Interval	NS	S	NS (w.i.)	S (w.i.)
O/N	0.70	1.29	1.66	2.51
1D	2.92	2.99	5.45	5.54
1W	1.46	1.55	5.66	5.81
1M	6.23	6.25	9.87	10.26

Notation - O/N: overnight, 1D: one day, 1W: one week, 1M: one month,
NS: No Sentiment, S: Sentiment, w.i.: with interactions.

3.2.3 Random Forest regression

Description Random Forest belongs to the ensemble learning family, which combines multiple individual models to improve overall prediction accuracy.

The Random Forest algorithm operates by constructing a multitude of decision trees during the training phase. Each decision tree is built using a subset of the available features and a random selection of the training data. This randomness helps to diversify the individual trees. It reduces overfitting and improves the generalization capabilities of the model.

During the prediction phase, each decision tree in the forest independently generates a prediction, and the final prediction is determined by averaging the predictions across all trees. This ensemble approach allows Random Forest to effectively capture complex relationships within the data and produce robust predictions.

Parameter Tuning via Cross-Validation The optimal hyperparameters for our Random Forest are determined using cross-validation. We used a grid search approach over a predefined parameter grid to explore combinations of hyperparameters and identify the configuration that maximizes the model's performance. The parameter grid tested includes:

1. **n_estimators**: Number of trees in the forest, with values set to $[10, 50, 100]$. More trees generally improve the model's performance but also increase computational complexity.
2. **min_samples_split**: Minimum number of samples required to split an internal node, with options $[2, 6, 10]$. Higher values prevent the model from learning overly specific patterns, thus controlling overfitting.
3. **min_samples_leaf**: Minimum number of samples required to be at a leaf node, with choices $[1, 3, 4]$. This parameter helps in smoothing the model, particularly in regression scenarios, by making the model more robust to noise in the training data.
4. **max_depth**: Maximum depth of each tree, chosen from [None, $10, 20, 50$]. A deeper tree can capture more details of the data but might lead to overfitting. Setting this parameter to *None* allows the nodes to expand until all leaves are pure or until all leaves contain less than *min_samples_split* samples.

We do 3-fold cross-validation to evaluate each parameter combination's performance.

Feature Importance Ranking Random Forest provides built-in mechanisms for feature importance ranking, allowing us to assess the relative contributions of different features to the predictive performance.

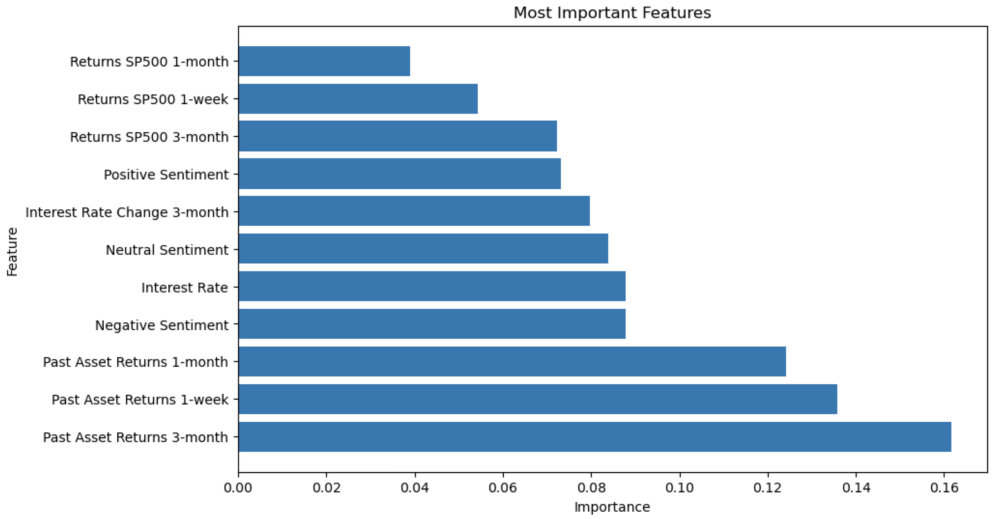

Figure 2: Importance of each feature when predicting the overnight returns.

The most important features are the past returns of the given asset, indicating momentum. Following this, negative sentiments are also significant, demonstrating that sentiment influences the decisions of the random forest model.

3.2.4 XGBoost

Description XGBoost (eXtreme Gradient Boosting) is a state-of-the-art machine learning algorithm. It belongs to the gradient boosting family, which sequentially builds an ensemble of weak learners to create a powerful predictive model.

The core principle behind XGBoost is to iteratively train a series of decision trees, with each subsequent tree aiming to correct the errors made by the previous ones. This iterative process allows XGBoost to gradually improve its predictive performance by focusing on the instances where the model has the highest prediction errors.

Parameter Tuning via Cross-Validation The optimal hyperparameters for our XGBoost model are also determined using cross-validation. The parameter grid tested includes:

1. **max_depth**: Maximum depth of each tree, with values set to $[2, 3, 4]$. A deeper tree can capture more details of the data but might lead to overfitting.
2. **learning_rate**: Also known as the step size, chosen from $[0.001, 0.01, 0.05]$.
3. **n_estimators**: Number of gradient boosted trees, with options $[100, 200, 300]$.
4. **subsample**: Proportion of the training instances sampled for training each tree, with choices $[0.8, 0.9, 1.0]$. Subsampling prevents overfitting by introducing more randomness into the model.
5. **colsample_bytree**: Proportion of features used for each tree, fixed at $[0.9]$ in this grid.

We do 3-fold cross-validation to evaluate each parameter combination's performance.

In order to compare the various methodologies, we decide to use the Root Mean Squared Error (RMSE) metric, which is computable for each approach. The results are shown in Table 3.

	Time Interval							
Method	O/N (S)	O/N (NS)	1D (S)	1D (NS)	1W (S)	1W (NS)	1M (S)	1M (NS)
Linear Regression	0.0792	0.0795	**0.0715**	**0.0715**	0.1123	0.1123	**0.1655**	**0.1655**
Linear Regression (w. i.)	0.0812	0.0814	0.0748	0.0745	0.1232	0.1224	0.1785	0.1763
Random Forest	0.0795	0.0807	0.0720	0.0719	0.1149	0.1154	0.1705	0.1710
XGBoost	**0.0788**	**0.0791**	0.0716	0.0717	**0.1121**	**0.1121**	0.1659	0.1659

Notation - O/N: overnight, 1D: one day, 1W: one week, 1M: one month,
NS: No Sentiment, S: Sentiment, w.i.: with interactions.

Table 3: Returns RMSE comparison for different methods (out-of-sample).

XGBoost generally performs better for shorter intervals (overnight and one week), while OLS is more accurate in one day and one month predictions. Linear regression with second-order interactions consistently shows higher RMSE, suggesting the added terms might not always be beneficial for these predictions (risk of overfitting). Random Forest performs moderately across all intervals but does not outperform XGBoost or Linear Regression in any specific time interval.

4 Trading strategies

4.1 Fixed Amount for All Earnings

This strategy involves investing a fixed amount of capital in each trade and closing the trade after a predetermined amount of time (one day, week or month). The decision to long or short an asset is based on the sign of the return's prediction. If we expect the asset's price to rise, we take a long position; if we expect the price to fall, we take a short position.

To evaluate the performance of this strategy, we build a portfolio that invests in every earnings of the test set. Then we compute the percentage gains of the portfolio over the full test set and scale it by the size of the test set to get the expected percentage gain by earning.

4.2 Proportional Amount for All Earnings

This strategy is similar to the "Fixed Amount for All Earnings" approach but introduces a key variation: it adjusts the investment amount based on the predicted return's magnitude. This means that larger investments are allocated to trades with higher predicted returns. Specifically, the investment for each asset is proportional to the absolute value of the predicted returns, aligning the investment size directly with the confidence in the return's magnitude.

Similar to the previous strategy, positions are taken in accordance with the sign of the predicted return: a long position if the prediction is positive and a short position if it is negative. The total percentage gains

of the portfolio are then normalized by dividing by the number of earnings in the test set to yield the expected gain per earning.

This adjustment allows the strategy to potentially capitalize more on trades where the model has higher certainty. However, a potential drawback is the inability to determine the required capital in advance.

4.3 Strongest Sentiment Percentile

This strategy focuses on leveraging the most extreme sentiments to dictate trading actions, aiming to capitalize on the assets that have the earnings call with the strongest positive or negative sentiment. The strategy goes long on assets with the most positive sentiment and short on those with the least positive (most negative) sentiment, based on a predefined sentiment percentile. Specifically, we use the top 5% for the most positive sentiment and the bottom 5% for the most negative sentiment.

For each trade, a fixed amount of capital is invested. The positions are held for a predetermined amount of time, similar to the other strategies.

To evaluate the effectiveness of this strategy, the performance is assessed by constructing a portfolio that only includes trades from assets that meet the sentiment criteria. The percentage gains of this portfolio are then normalized by the number of trades taken. Note that we do not use the predicted returns for this strategy.

4.3.1 Results

Table 4: Returns for Different Models Across Various Time Horizons (%)

		With sentiments				Without sentiments			
Model	Strategy	O/N	1D	1W	1M	O/N	1D	1W	1M
	Baseline	-	-	-	-	-0.28	0.19	0.43	-2.51
	Only Sentiments	-0.85	0.11	0.26	0.87	-	-	-	-
Linear Regression	Constant	0.18	0.36	0.28	-0.18	-0.17	0.34	0.22	-0.26
	Proportional	0.31	**0.70**	1.17	1.35	-0.43	0.70	1.18	1.34
Linear Regression (w. i.)	Constant	0.40	0.25	0.25	0.22	0.07	0.19	0.28	-0.15
	Proportional	0.39	0.53	0.83	**2.36**	0.0	0.47	0.79	**2.36**
Random Forest	Constant	0.44	0.15	0.43	-1.85	**0.17**	0.37	0.48	-1.75
	Proportional	0.64	0.62	1.23	0.55	-0.07	**0.82**	1.20	0.51
XGBoost	Constant	0.74	0.02	0.27	-2.5	-0.30	0.10	0.80	-2.56
	Proportional	**0.83**	0.46	**1.76**	0.87	-0.22	0.29	**1.74**	-2.49

Notation - O/N: overnight, 1D: one day, 1W: one week, 1M: one month, w.i.: with interactions.

Proportional vs. Constant Strategies The analysis of returns across various time intervals suggests that the proportional strategy generally outperforms the constant strategy. This trend is observed across different models, indicating a robust advantage when adjusting investment proportions based on stronger confidence in expected returns.

Impact of Sentiment Analysis Including sentiment data appears to improve the performance of overnight returns. However, it is important to note that sentiment data is not available at the start of the overnight period. For other time horizons, the impact of sentiment data on predictive power is less evident.

Observations on Market Reactions The results also hint at an under-reaction to earnings call sentiments, aligning with the concept of "Post-Earnings-Announcement Drift" [14] widely documented in financial literature. This phenomenon generally relates to the market's delayed response to the earnings "surprise" factor, which is the difference between actual and expected earnings. Our findings, particularly from the sentiment-only strategy where returns gradually increase, suggest a similar drift based on sentiment aspects of earnings calls. This indicates that the market may also under-react to the tone and content of disclosures during these calls. This under-reaction presents opportunities for investors who incorporate sentiment analysis into their strategies.

5 Conclusion

Our analysis of the sentiment from earnings calls using different machine-learning models has shown that, among the various methods tested, the FinBERT model demonstrated the best performance. Its ability to discern nuanced sentiments within financial texts proved superior, as reflected in our benchmarks and results.

We then attempted to predict stock returns over short to intermediate time frames using extracted sentiment data, while also incorporating other financial indicators such as interest rates and past asset returns. This approach aimed to express not just the sentiment but also the broader economic context influencing stock movements. We had more success including the sentiments for the overnight returns but it was not evident that including sentiments helped for the other time frames.

For the trading strategies, we developed and tested several approaches based on the sentiments and predictive returns from the earnings calls. The proportional investment strategy has proven to be the best strategy among those tested. It optimizes capital allocation based on the magnitude of predicted returns thus aligning investment with confidence levels. For the strategy based uniquely on the earnings call sentiment, we observe that market reactions are slightly delayed with respect to the earnings call. This phenomenon is best known as the *Post-Earnings Announcement Drift* in the literature.

In conclusion, our study highlights the potential influence of sentiment on financial returns, although quantifying this impact is challenging for several reasons. Firstly, the signal-to-noise ratio in finance is very low, making it difficult to extract meaningful signals from the data. Secondly, earnings calls occur shortly after earnings releases, complicating the distinction between the effects of sentiments from the earnings calls and the earnings releases themselves. Lastly, the effective size of our test set may not be as robust as it initially appears. Although our test set includes over 4,000 earnings calls, these data points are not entirely independent, as they are influenced by the same economic contexts and exhibit correlated returns.

To more definitively assert the presence of exploitable 'alphas', a larger dataset and the use of ensemble learning methods could enhance the significance and robustness of the results. We believe that this research paves the way for further investigations into how sentiment can be effectively integrated into trading strategies.

References

1. Kouwenberg, R., Vorst, T. & Monique, W. Options and Earnings Announcements: An Empirical Study of Volatility, Trading Volume, Open Interest and Liquidity. *European Financial Management* **6,** 149–171 (June 2000).
2. Medya, S., Rasoolinejad, M., Yang, Y. & Uzzi, B. *An Exploratory Study of Stock Price Movements from Earnings Calls* 2022. arXiv: 2203.12460 [q-fin.ST].
3. Heinrichs, A., Park, J. & Soltes, E. F. Who Consumes Firm Disclosures? Evidence from Earnings Conference Calls. *The Accounting Review* **94,** 205–231. ISSN: 0001-4826. eprint: https://publications.aaahq.org/accounting-review/article-pdf/94/3/205/37323/accr-52223.pdf. https://doi.org/10.2308/accr-52223 (May 2019).
4. Fu, X., Wu, X. & Zhang, Z. The Information Role of Earnings Conference Call Tone: Evidence from Stock Price Crash Risk. *Journal of Business Ethics* **173,** 643–660. https://ideas.repec.org/a/kap/jbuset/v173y2021i3d10.1007_s10551-019-04326-1.html (Oct. 2021).
5. Sohangir, S., Wang, D., Pomeranets, A. & Khoshgoftaar, T. M. Big Data: Deep Learning for financial sentiment analysis. *Journal of Big Data* **5.** https://api.semanticscholar.org/CorpusID:256400578 (2018).
6. Vaswani, A. *et al.* Attention Is All You Need. *CoRR* **abs/1706.03762.** arXiv: 1706.03762. http://arxiv.org/abs/1706.03762 (2017).
7. Wu, S. *et al. BloombergGPT: A Large Language Model for Finance* 2023. arXiv: 2303.17564 [cs.LG].
8. Malo, P., Sinha, A., Takala, P., Korhonen, P. J. & Wallenius, J. Good Debt or Bad Debt: Detecting Semantic Orientations in Economic Texts. *CoRR* **abs/1307.5336.** arXiv: 1307.5336. http://arxiv.org/abs/1307.5336 (2013).
9. Sanh, V., Debut, L., Chaumond, J. & Wolf, T. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. *ArXiv* **abs/1910.01108** (2019).
10. Liu, Y. *et al. RoBERTa: A Robustly Optimized BERT Pretraining Approach* 2019. arXiv: 1907.11692 [cs.CL].
11. Araci, D. FinBERT: Financial Sentiment Analysis with Pre-trained Language Models. *CoRR* **abs/1908.10063.** arXiv: 1908.10063. http://arxiv.org/abs/1908.10063 (2019).
12. Devlin, J., Chang, M., Lee, K. & Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *CoRR* **abs/1810.04805.** arXiv: 1810.04805. http://arxiv.org/abs/1810.04805 (2018).
13. Matsumoto, D., Pronk, M. & Roelofsen, E. What Makes Conference Calls Useful? The Information Content of Managers' Presentations and Analysts' Discussion Sessions. *The Accounting Review* **86,** 1383–1414 (July 2011).
14. Bernard, V. L. & Thomas, J. K. Evidence that stock prices do not fully reflect the implications of current earnings for future earnings. *Journal of Accounting and Economics* **13,** 305–340 (1990).

A Example of Earning Call Transcript

Prepared Remarks:
Operator
Good day, and welcome to the Bilibili 2020 Second Quarter Earnings Conference Call. Today's conference is being recorded.
At this time, I would like to turn the conference over to Juliet Yang, Senior Director of Investor Relations. Please go ahead.
Juliet Yang -- Senior Director of Investor Relations
Thank you, operator.
Please note the discussion today will contain forward-looking statements relating to the Company's future performance, and are intended to qualify for the Safe Harbor from liability, as established by the US Private Securities Litigation Reform Act. Such statements are not guarantees of future performance and are subject to certain risks and uncertainties, assumptions and other factors. Some of these risks are beyond the Company's control and could cause actual results to differ materially from those mentioned in today's press release and this discussion. A general discussion of the risk factors that could affect Bilibili's business and financial results is included in certain filings of the Company with the Securities and Exchange Commission. The Company does not undertake any obligation to update this forward-looking information, except as required by law.
During today's call, management will also discuss certain non-GAAP financial measures, for comparison purposes only. For a definition of non-GAAP financial measures and the reconciliation of GAAP to non-GAAP financial results, please see the 2020 second quarter financial results news release issued earlier today.
As a reminder, this conference call is being recorded. In addition, an investor presentation and a webcast replay of this conference call will be available on the Bilibili investor relations website at ir.bilibili.com.
Joining us today on the call from Bilibili's senior management are Mr. Rui Chen, Chairman of the Board and Chief Executive Officer; Ms. Carly Lee, Vice Chairwoman of the Board and Chief Operating Officer; and Mr. Sam Fan, Chief Financial Officer.
And I will now turn the call over to Mr. Fan, who will read the prepared remarks on behalf of Mr. Chen.
Xin Fan -- Chief Financial Officer
Thank you, Juliet, and thank you, everyone, for participating in our 2020 second quarter conference call. I'm pleased to deliver today's opening remarks on behalf of Mr. Chen.
The second quarter was another strong quarter of growth for Bilibili. Owing to our increasing diverse content and wider awareness of our brand, we are reaching a much broader audience. For the second quarter, MAUs were 172 million, up 55% and DAUs were up 52% to 51 million, both on a year-over-year basis. Mobile MAUs continued to be our fastest grower, reaching 153 million in the period, up 59% compared to the same period in 2019.
Along with the expanded user base, user engagement continues to be strong. In the second quarter, our users spent an average of 79 minutes per day on our Bilibili app, making us one of the most popular platforms among our peers. As we move into our peak summer season, with solid execution of our initiatives, we're looking forward to further building on our user growth momentum.
We continue to rollout more premium content and services, strengthen our monetization capabilities, and we are increasingly converting traffic to paid users. Our MPUs were up 105% year-over-year, reaching 12.9 million in the second quarter, and our paying ratio improved to 7.5%, compared to 5.7% from the same period last year. These increases fueled our top line growth and we once again reached record high revenues of RMB2.6 billion, beating the high-end of our guidance.
Our gross margin expanded to 23% from 16% in the second quarter of last year as we continued to realize more operating leverage.

Figure 3: Example of earning call transcript

B Exploratory Data Analysis Visualization

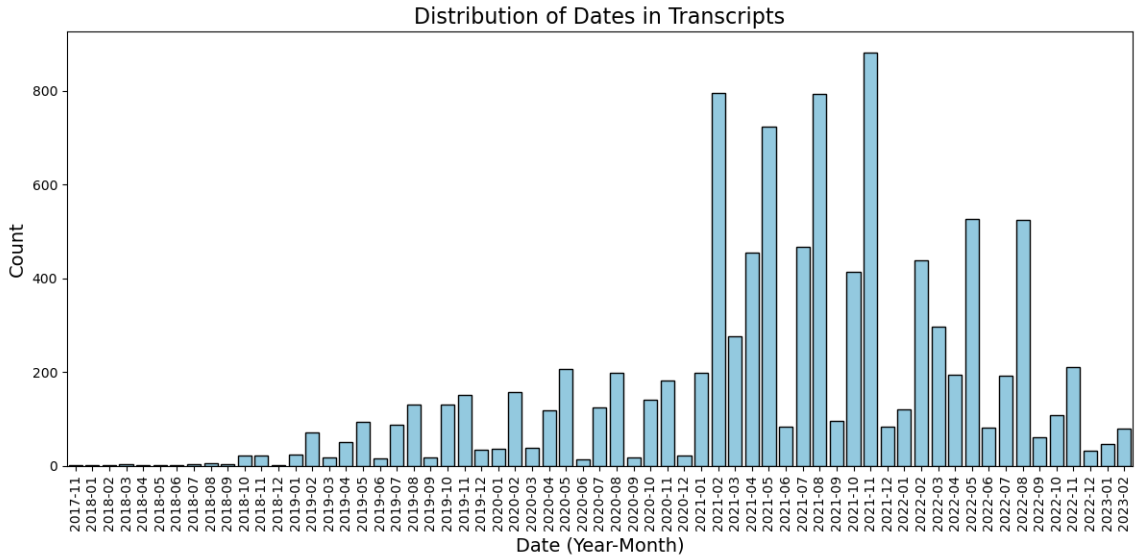

Figure 4: Date of the earning calls

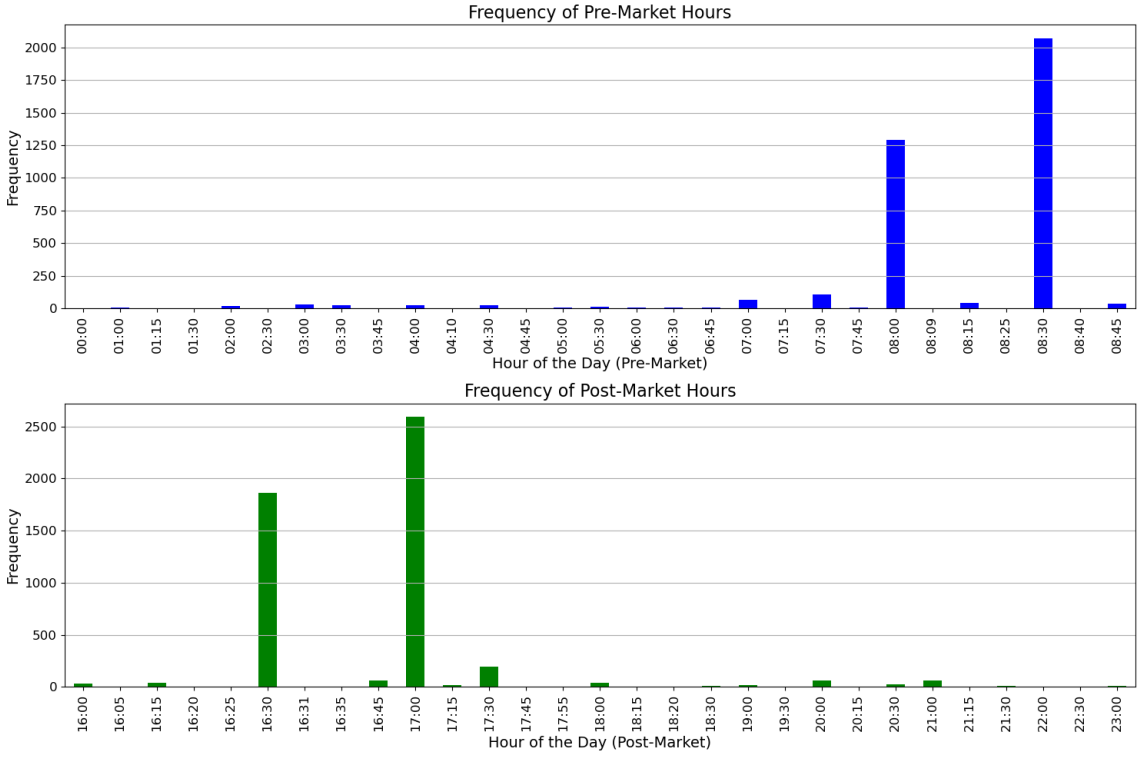

Figure 5: Hour of the earning calls

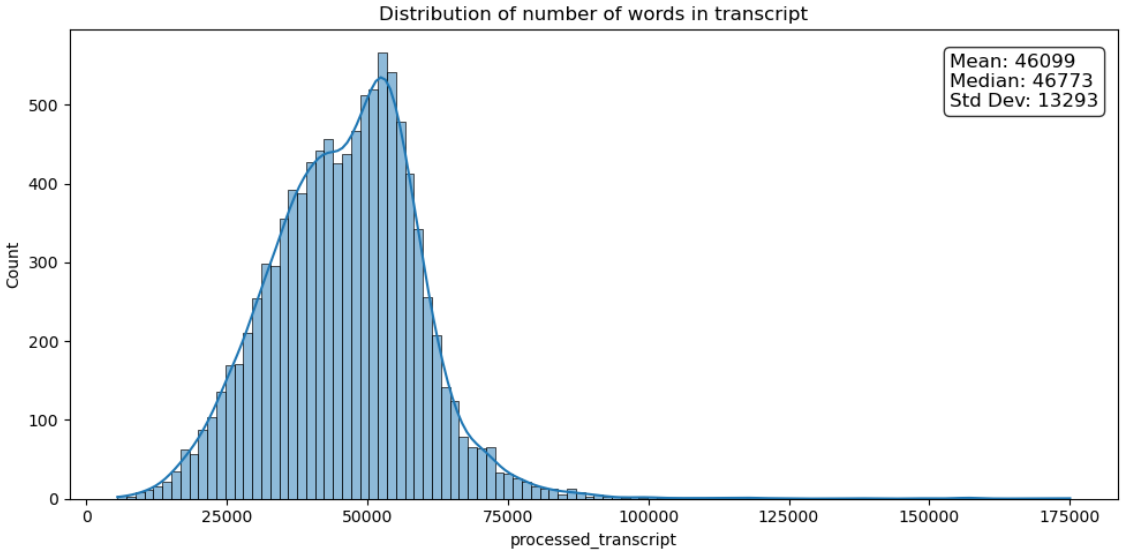

Figure 6: Distribution of the number of words

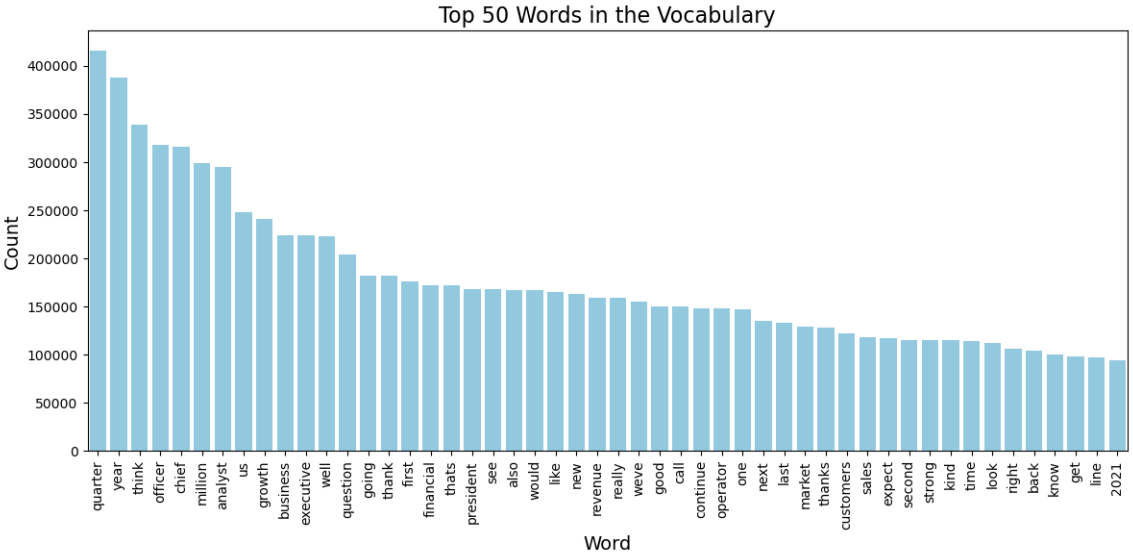

Figure 7: Top recurring words in the earning calls