Forecasting Earnings Surprises from Conference Call Transcripts

Ross Koval[1,3], **Nicholas Andrews**[2], **and Xifeng Yan**[1]

[1]University of California, Santa Barbara
[2]Johns Hopkins University
[3]AJO Vista
rkoval@ucsb.edu

Abstract

There is a multitude of textual data relevant to the financial markets, spanning genres such as financial news, earnings conference calls, and social media posts. Earnings conference calls are one of the most important to information flow as they reflect a direct communication between company executives, financial analysts, and large shareholders. Since these calls contain content that is forward-looking in nature, they can be used to forecast the future performance of the company relative to market expectations. However, they typically contain over 5,000 words of text and large amounts of industry jargon. This length and domain-specific language present problems for many generic pretrained language models. In this work, we introduce a novel task of predicting earnings surprises from earnings call transcripts and contribute a new long document dataset that tests financial understanding with complex signals. We explore a variety of approaches for this long document classification task and establish some strong baselines. Furthermore, we demonstrate that it is possible to predict companies' future earnings surprises from solely the text of their conference calls with reasonable accuracy. Finally, we probe the models through different interpretability methods and reveal some intuitive explanations of the linguistic features captured that go beyond traditional sentiment analysis.

1 Introduction

There is a multitude of textual data relevant to the financial markets, spanning genres such as financial news, earnings conference calls, analyst recommendation reports, social media posts, and regulatory filings. Earnings conference calls are one of the most important datasets relevant to the information flow in equity markets because they reflect a direct communication between company executives and financial analysts (Brown et al., 2004).

Many public companies in the US hold earnings conference calls quarterly, in which their executives discuss the recent performance of the firm, their prospects, and answer questions from financial analysts covering their firms. Typically, companies report results quarterly at a lag of 4-6 weeks from the end of the previous period, and hold a conference call shortly thereafter. Therefore, the company executives have substantial knowledge into the next period's results when the call is held, providing a rare opportunity to detect textual indicators, such as tone and emotion in executive and analyst language patterns. These diverse signals, which can vary from clear sentiment to more subtle signs of deception (Larcker and Zakolyukina, 2012) or obfuscation (Bushee et al., 2018), may reveal important information about the current and future prospects of the company and be used to forecast its future earnings surprises. Earnings surprises, which measure the operating performance of a company relative to market expectations, are highly followed by equity investors and often result in high magnitude stock returns and volatility (Doyle et al., 2006).

Input:	"...we continued our positive momentum in the first quarter reporting comp sales that accelerated from our strong fourth quarter performance. during the quarter, we drove market share gains and better than expected profitability by capitalizing on the advantages of our business model with dynamic marketing, compelling brands, and providing our customers with the preferred beauty shopping experience..."
Output:	Positive Surprise, $P(y = 1) = 0.95$

Figure 1: Paragraph of a sample transcript from the Validation Set that resulted in a positive surprise prediction from the model.

In this paper, we consider the problem of using the textual content of earnings call transcripts to forecast future earnings surprises. It is important to note that this is a challenging task because the forecasting horizon is long (∼3 months), producing a lot of uncertainty between the forecast and

Findings of the Association for Computational Linguistics: ACL 2023, pages 8197–8209

event data, and there are legal restrictions about what is allowed to be disclosed to the public during the call. As a result, it is not clear *a priori* if the content of call transcripts contains sufficient task signal to outperform even uninformed baselines.

In the broader literature, there has been growing interest in the relevance of textual content to financial markets that has increasingly grown in sophistication in recent years. Some initial attempts used the Harvard General Inquirer IV-4 (HGI) dictionary to measure word polarity in financial text but found there to be domain mismatch (Price et al., 2012). Then, Loughran and Mcdonald (2011) constructed their own financial-specific sentiment dictionary (LM). Interestingly, they found that 75% of words with strong negative polarity in the HGI dictionary had a neutral sentiment in a financial context (e.g. liability, tax, excess, etc.). Further, Ke et al. (2019) develop a supervised learning method to identify sentiment in WSJ news articles, and found that over 40% of the most negative words identified by their model are not present in the LM dictionary because they are not negative in the context of regulatory filings. In other words, even within the financial domain, there can be a genre mismatch between different types of financial text. This motivated us in this work to explore models that are well attuned to the language of earnings conference calls.

However, the length of these conference calls, typically ranging from 5,000 to 10,000 words per transcript, creates some challenges because many of the popular pretrained language models only support a maximum length of 512 tokens. While there have been many advances in developing Efficient Transformers that reduce the time complexity of the self-attention mechanism, these methods are still computationally expensive to pretrain and there does not yet exist a variety of pretrained versions, such as BERT (Devlin et al., 2019) and RoBERTa (Liu et al., 2019), which both contain many variants targeting specific domains, such as biology (Lee et al., 2020), medicine (Gu et al., 2021), law (Chalkidis et al., 2020), finance (Yang et al., 2020a), and many others. As far as we know, there does not exist equivalent domain specific pretrained versions of these Efficient Transformers.

In this work, we explore a variety of approaches to this novel long document classification task and make the following contributions.

1. We introduce a novel task of using the text from earnings conference calls to make long horizon predictions of future company earnings surprises and explore a variety of approaches to this long document classification task (§3).

2. We contribute a new long document dataset that tests financial language understanding with complex signals that we anticipate to be of broader interest to the computational linguistics community (§3.1).

3. We explore a variety of approaches for this task, including simple bag-of-words models as well as long document Transformers, such as Efficient Transformers and tailored Hierarchical Transformer models, establishing the state of the art (§4).

4. We demonstrate that it is possible to predict companies' future earnings surprises with reasonable accuracy from solely the textual content of their most recent conference calls (§5).

5. We probe the best model through different interpretability methods to reveal some intuitive explanations of the linguistic features captured, which indicates that our model is learning more powerful features than just traditional sentiment (§6).

We release the dataset and sample code at: `https://github.com/rosskoval/fc-es-ccts`

2 Related Work

2.1 Long Document Classification

There are generally two different approaches to modeling long documents. First, there is a class of Efficient Transformer models that were designed for long documents, such as Transformer-XL (Dai et al., 2019), Longformer (Beltagy et al., 2020), BigBird (Zaheer et al., 2020), and Reformer (Kitaev et al., 2020), which modify the self-attention mechanism in the original implementation to make it more efficient to accommodate longer contexts. These models have been shown to excel at long document understanding tasks, such as classification, question-answering, and summarization.

Alternatively, a hierarchical attention approach can be used. In Hierarchical Attention Networks

(Yang et al., 2016), the authors model a document in a hierarchical fashion, viewing a sentence as a sequence of words and a document as a sequence of sentences, and use self-attention and recurrent networks to produce document representations. More recent work on Hierarchical Transformers (HTs) extend this approach to use pretrained language models for segment-level embeddings and Transformers for document-level representations (Pappagari et al., 2019; Yang et al., 2020b; Zhang et al., 2019; Mulyar et al., 2019). This approach naturally supports longer sequence lengths due to the product of multiple self-attention mechanisms. Although Efficient Transformers are attractive in principle, there is much less availability of them pretrained in different domains and languages, such as mBERT (Devlin et al., 2019) and XLM-RoBERTa (Conneau et al., 2020).

2.2 Financial Prediction

Since earning conference calls are highly followed by most investors, there has been a considerable amount of research performed on them. For instance, Larcker and Zakolyukina (2012) use data on subsequent financial restatements to analyze conference calls for deceptive behavior. Frankel et al. (2018), apply traditional ML models to extract the sentiment of conference calls and use it to predict subsequent analyst revisions. However, as far as we know, this is the first work that uses deep learning to directly learn to predict earnings surprises from conference call transcripts in an end-to-end manner.

In Qin and Yang (2019), the authors propose a deep multi-modal regression model that jointly leverages textual and audio data from a small sample of conference calls to predict short-term stock volatility. Sawhney et al. (2020a,b) build on that work to leverage the network structure of stock correlations with graph networks and financial features to make joint predictions. Similarly, Sang and Bao (2022) propose a multi-modal structure that jointly models the textual dialogue in the call and the company network structure. Further, Yang et al. (2020a, 2022) propose a multi-modal model that leverages the numerical content of financial text.

In Huang et al. (2022), the authors release a pretrained language model for the financial domain, termed FinBERT, and show that their model understands financial text significantly better than competing methods in many aspects, including sentiment and ESG. They constructed the model by pretraining the BERT-base architecture from scratch with a custom vocabulary on a large corpus of English text from the financial domain, consisting of conference calls, corporate filings, and analyst recommendation reports, that is commensurate in size to the BERT pretraining corpus.

3 Problem

3.1 Data

We collected manually transcribed English conference calls from the largest publicly trade companies in the US (MSCI USA Index) over January 2004 to December 2011, from FactSet Document Distributor. We also source Reported Earnings per Share (EPS) and Analyst Consensus Estimates of EPS from FactSet Fundamentals and Consensus Estimates, respectively. To focus on the largest and most actively traded companies in the US, we filter the data such that all companies in the sample have market capitalizations above $1B USD and daily average trading volume over $50M USD. Then, we temporally partition the data into train, validation, and test sets. We use transcripts that occurred between 2005 and 2009 as the training set, those that occurred in 2010 as the validation set, and those that occurred in 2011 as the test set. It is important to note that these sets much be temporally disjoint and monotonically ascending in time to avoid look-ahead bias. We provide summary statistics in Table 1.

	Train	Validation	Test
Start Date	Jan 2004	Jan 2010	Jan 2011
End Date	Dec 2009	Dec 2010	Dec 2011
Sample Size	4,056	524	588
Avg # of Words	9,016	8,907	9,010
Max # of Words	26,130	19,853	17,650
Avg # of Sentences	390	409	415
Avg # of Words per Sentence	25	22	22

Table 1: Summary Statistics of the Earnings Call Transcript dataset on each sample split.

3.2 Supervised Learning Task

We propose the task to predict the direction of the next earnings surprise ES from solely the textual content of the most recent transcript as a supervised learning task. Therefore, the input is a raw, unsegmented, English transcript with maximum number of words L_T. We set L_T to be 12,000 words (∼20,000 BERT tokens) for computational and memory constraints, and because less than 10% of the transcripts in the sample are longer than that length. We select the Standardized Unexpected Earnings (SUE; Latane and Jones, 1979) as our measure of earnings surprise, which is defined to be the difference between the reported EPS of the company and the analyst consensus estimate of the EPS, scaled by the inverse of the standard deviation (dispersion) of the analyst forecasts. We measure the analyst consensus estimate as the mean of all latest valid analyst forecasts, collected 1-month following the last earnings call transcript (the one we are using to make the prediction), which serves are the closest approximation to forward-looking market expectations; this allows analysts to update their forecasts based off their perception of the conference call and recently reported company results, and yields a more challenging, but potentially more rewarding, task than if we collected analyst forecasts at an earlier time horizon. We note that there is roughly a 3-month time horizon between the earnings call and the reporting of the next earnings surprise, making this long horizon prediction task particularly challenging.

$$ES = \frac{RepEPS - Avg(EstEPS)}{Std(EstEPS)}$$

$$y = \begin{cases} 0, & ES \leq -\delta \\ 1, & ES \geq \delta \end{cases}$$

We binarize the continuous value, such that an ES above δ corresponds to a label of +1 and represents a positive surprise, while an ES below $-\delta$ corresponds to a label of 0 and represents a negative surprise. We select a value for δ of 0.10 as a balance between the sample size and the significance of the events, such that about $\frac{1}{4}$ of events are positive surprises and $\frac{1}{4}$ of events are negative surprises. We discard transcripts which do not result in a material earnings surprise. Since this does not translate to a perfect class label balance, we randomly down-sample the majority class, such that there is an equal 50/50 split of positive surprises and negative surprises in each sample split, to more easily interpret the results. Thus, we can use accuracy score as our primary evaluation metric. We use binary cross-entropy as the loss function for this binary classification task.

While the underlying earnings surprise metric is continuous, the importance of the metric to the market is often more binary. In general, market participants are more interested in the direction of the surprise than the precise magnitude of it and typically react accordingly insofar as the surprise is a "material" event. However, there are neutral cases when the company beats or misses the forecast by a small margin (i.e. [-0.10, 0.10] in this work) in which market reaction is typically lower. These boundary cases are generally difficult for the model to learn from at this long horizon because the ex-ante true probability is approximately random and there is often some form of earnings management involved. Therefore, we choose to focus on the most important earnings surprise events and disregard the neutral class.

4 Methods

4.1 Approach

We provide a wide variety of baseline models for this task, consisting of a combination of traditional and neural models, and establish several strong baselines as well as the state-of-the-art. While the current literature suggests that domain adaptation methods, such as language model finetuning on a domain-relevant corpus, is beneficial when using generic pretrained language models in out-of-distribution tasks, (Han and Eisenstein, 2019; Gururangan et al., 2020), this additional pretraining is computationally expensive and requires large-scale datasets to be effective. Therefore, we explore the use of existing pretrained language models.

4.2 Bag-of-Words

For the classical ML baselines, we select bag-of-words with n-grams (BOW) with TF-IDF weighting (Salton and Buckley, 1988), and Logistic (Logistic) and Gradient Boosted Decision Trees (GBDT) as classifiers. We also provide a simple dictionary-based model that uses the proportion of words in each category of the Loughran and McDonald (LM) financial sentiment dictionary as features to a Logistic classifier. Addition-

ally, we consider both general (BERT-Sent) and domain-specific (FinBERT-Sent) pretrained sentiment classifiers that are applied at the sentence level and aggregated with simple majority-rule voting. BERT-Sent is BERT-base-uncased (Devlin et al., 2019) finetuned on IMDB movie reviews (Maas et al., 2011) and FinBERT-Sent is FinBERT (Huang et al., 2022) finetuned on manually labeled sentences from financial analyst research reports.

Given that the positive autocorrelation of earnings surprises is documented in the financial literature (Kama, 2009), we provide a simple autoregressive time-series baseline AR(1) that fits a logistic classifier on the continuous value of the firm's previous earnings surprise. In general, the autocorrelation beyond the most recent quarter is much lower. While the resulting performance is far below that of the best long-document models we provided, it is important to note that the signal contained in the text is likely largely distinct from and complementary to the information contained in the lagged surprise variables.

4.3 Short Context Models

For the short context models that only support up to 512 tokens of text, we consider multiple variants of FinBERT, including first, last, and random 512 tokens (truncation), as well as various forms of aggregation, including mean & max pooling over time, to aggregate segment embeddings into document representations.

4.4 Hierarchical Transformers

We provide various forms of Hierarchical Transformers (HTs) with different pretrained segment encoders, and train them end-to-end on our supervised learning task. HTs take a hierarchical approach by dividing each long document into shorter non-overlapping segments of maximum length L. To do so, we use greedy sentence chunking to recursively add sentences to each segment until the length of the segment exceeds L. We choose this segmentation strategy to avoid breaking up and mixing sentences into chunks to better preserve their syntax and semantics. Since it is not clear what the optimal value of L should be, we treat it as an additional hyperparameter and tune it over $\{32, 64, 128\}$.

Segment Encoder

We initialize the segment encoder with pretrained models, which typically supports a maximum sequence length of 512 tokens. We explore BERT (Devlin et al., 2019) and FinBERT (Huang et al., 2022). This model produces contextualized embeddings of all tokens in each segment and we extract the last hidden state of the first [CLS] token as our segment representation (Devlin et al., 2019).

Document Encoder

We use the standard Transformer architecture (Vaswani et al., 2017) with multi-head self-attention and sinusoidal positional encodings as the document encoder. The document encoder is responsible for taking the segment embeddings and producing contextualized segment representations by allowing each segment to attend to all other segments in the document and share information. We apply max pooling over time and concatenate with the first state to arrive at our document representation. We tune the number of layers over $\{2, 3, 4\}$ and set the number of attention heads in each layer to be 6.

4.5 Efficient Transformers

We select BigBird (Zaheer et al., 2020) as our Efficient Transformer baseline model because it has been shown to exhibit state-of-the-art performance on long document classification and question-answering tasks (Zaheer et al., 2020). The model applies a combination of local (sliding window), random, and global attention to sparsely approximate the full self-attention matrix. We also experimented with Longformer (Beltagy et al., 2020) but found BigBird to be more efficient and effective on this task, likely due to the increased number of global attention tokens and smaller attention window sizes. This result may suggest that the number of global attention tokens can be traded-off against the attention window size to improve efficiency and maintain effectiveness in Efficient Transformer models.

Since there are no versions of BigBird pretrained on the financial domain, we use the RoBERTa-base checkpoint that was warm-started from RoBERTa-base and further pretrained on a large corpus of long documents. We continue the MLM pretraining process on our in-task dataset to adapt to financial language (Gururangan et al., 2020). We tune the block size over $\{32,\ 64,\ 84\}$ and the number of random blocks over $\{3,\ 4,\ 5\}$. Please see Appendix A for more details.

We also provide two simple, heuristic-based extraction baselines in which we first extract the

most salient (*a priori*) sentences in each transcript, defined to be those that contain forward-looking statements (FLSE) according to Li (2010) or positive/negative sentiment words (LMSE) according to the LM dictionary, and pass the resulting abridged text to BigBird, thereby reducing the text length by about 75% and 67%, respectively, and potentially avoiding the truncation of the most relevant sentences. However, we do not find these extraction steps to be effective and discuss them further below.

4.6 Implementation Details

We train all models for a maximum of 10 epochs and select the checkpoint with the highest validation accuracy for further evaluation. BigBird and Hierarchical Transformers both contain approximately 130M parameters. We include more details on the implementation and training process in Appendix A.

5 Results

5.1 Comparison

Model	Test Accuracy
AR(1)	58.33%
BERT-Sent	49.58%
FinBERT-Sent	51.27%
LM + Logistic	60.20%
BOW + Logistic	67.68%
BOW + GBDT	**71.43%**
FinBERT – First 512	63.44%
FinBERT – Last 512	62.24%
FinBERT – Random 512	62.07%
FinBERT + Mean Pooling	66.84%
FinBERT + Max Pooling	**73.28%**
BigBird	**75.87%**
BigBird + MLM	74.30%
BigBird + FLSE	65.65%
BigBird + LMSE	72.98%
Hierarchical BERT	70.24%
Hierarchical FinBERT	**76.56%**

Table 2: Comparison of classifier performance on the test set across the neural and non-neural baselines. The best model in each class is indicated in bold.

As shown in Table 2, the pretrained sentiment models with simple majority-rule voting, BERT-Sent and FinBERT-Sent, fail to perform much better than random chance, indicating the difficulty of the task. Surprisingly, we observe that a simple bag-of-ngrams with TF-IDF weighting performs comparatively well on this dataset when used with GBDTs, outperforming many of the simple neural baselines. This indicates that substantial signal can be captured through non-linear interaction of normalized unigram/bigram features. It also indicates the models that try to reduce the length of text through either various forms of truncation or simple aggregation, likely dilute the signal and do not possess the ability to identify and capture the most salient portions of the transcript.

Further, we observe the importance of domain alignment within the Hierarchical Transformer models, with FinBERT performing significantly better than BERT for this task. Given earnings conference calls were a large component of the pretraining corpus, the benefit of FinBERT is expected. However, we do not find that domain adaptation of BigBird via further MLM pretraining to be beneficial in this setting, likely because of the small size of the training set.

Interestingly, we find that the addition of FLSE and LMSE is detrimental to the performance of BigBird, and that FLSE performs considerably worse than the best simple baselines, suggesting that the signal contained in the task requires the additional text to contextualize statements about forward-looking performance with information about the past and present, and supports the view that a model that can simultaneously process the full transcript in an end-to-end manner is required for strong performance on this task.

5.2 Document Length

# Tokens	Test Accuracy
1,000	70.09%
2,000	72.64%
5,000	74.34%
10,000	75.72%
20,000	76.56%

Table 3: Comparison of model performance when truncating the text in the test set at various length cutoffs in terms of number of FinBERT tokens.

In Table 3, we apply the trained Hierarchical FinBERT model to the test set with different truncation lengths. Our results also indicate the performance degradation when truncating long documents to shorter lengths and demonstrate the need for models to support the ability to process longer documents, as there is more than a 6% gap in test set accuracy between using the first 1,000 tokens and using the first 20,000 tokens, indicating that

truncation loses valuable information contained in the middle of the transcript. While the first and last portions of the transcript may be the most relevant, it is clear that the middle portions also contain predictive value.

5.3 Training Efficiency

Model	Time
Hierarchical FinBERT	1.00
BigBird	1.40
Longformer	1.79

Table 4: Comparison of model finetuning times per epoch normalized to Hierarchical FinBERT.

In Table 4, we observe that the hierarchical structure of our model is quite efficient for processing long documents (>20K tokens). Compared to BigBird and Longformer, which can only process the first 4,096 tokens, we observe an approximately 50% speed up in finetuning time.

6 Model Interpretability and Analysis

Since Hierarchical FinBERT was the best performing model, we probe its predictions through a variety of interpretability methods to better understand the linguistic features important to this task.

6.1 LIME

First, we conduct Local Interpretable Model-agnostic Explanations (LIME) (Ribeiro et al., 2016) as a word-level attribution test, which constructs a sparse linear model across the perturbed neighborhood of each sample to approximate the influence of the top 10 features (words) to the model's prediction. We sort by the words that have the highest feature importance summed over 100 random samples from test set (given length of the transcripts, performing this analysis on the full test set was not computationally feasible).

Top Words for Positive Surprise:	higher, strong, great, raising, beat, congrats, outperformance, benefitted, quarter, formidable, improvement
Top Words for Negative Surprise:	bad, challenging, negatives, impacted, caused, tough, unfavorable, because, capacity, offset

Figure 2: Top Words according to largest LIME weights summed over a random selection of 100 samples from the test set.

As shown in Figure 2, many of the phrases with the highest importance have a strong sentiment attached to them. For instance, "congratulations" and "great quarter" are important features, which are used by financial analysts to praise the performance of the company. Since it appears that the top positive words are more intuitive and have larger magnitude weights, we conjecture that positive sentiment is more easily expressed, while negative sentiment may often manifest in what is not said. We also note that words such as "because" and "caused" may be used to try to explain away poor performance.

Category	# words	% words	% sentences	coeff	p-value
Positive	347	1.29	21.69	53.73	0.000
Negative	2345	0.65	11.99	-38.56	0.000
Uncertain	297	0.88	15.61	16.90	0.117
Litigious	903	0.13	2.50	14.59	0.284
Constraining	184	0.10	1.30	-31.97	0.050
Strong Modal	19	0.48	9.39	-9.51	0.106
Weak Modal	27	0.40	7.56	43.04	0.017

Table 5: Multivariate Linear Regression of model predictions onto LM financial sentiment variables on the test set. All standard errors are clustered by firm and year-month, and covariates include year-month fixed effects to account for intra-firm residual correlation. The p-value represents a two-sided significance test.

6.2 LM Sensitivity Analysis

To further understand model behavior, we perform another interpretability test using the LM financial dictionary and the predictions of Hierarchical FinBERT. We provide an overview of the summary statistics of the dictionary and results in Table 5.

To do so, we compute the proportion of total words in each transcript that belong to each LM category as financial sentiment variables. Then, we extract the model predictions $(P(y = 1))$ on the test set and regress them onto the financial sentiment variables. We observe that the model's predictions are positively associated with positive financial sentiment, and negatively associated with negative and constraining financial sentiment.

We also see smaller associations with strong modal (negative), litigious (positive), and uncertain (positive), but these are less statistically significant. We note that the LM dictionary was created based on word meaning in a sample of firm regulatory filings, which are distinct in style from conference calls, so there may be some domain mismatch. While some of the variables are sta-

tistically significant at the 95% level, the linear model has an adjusted R^2 of 10.3% (without the fixed effects), indicating that the trained model is capturing more than just LM sentiment. We note the negative relationship with strong modal words, such as "must," "best," "clearly," which may be used in persuasive writing to convince the audience of a particular viewpoint, and we conjecture that this may be a potential sign of executives trying to control the market narrative towards their company. In fact, Loughran and Mcdonald (2011) find that firms with higher proportions of strong modal words in their regulatory filings are more likely to report material weakness in their accounting controls. Conversely, there appears to be a positive relationship with weak modal and uncertain words, such as "may," "depends," "appears," which may be a reflection of executives's honest portrayal of their expectations about the future.

6.3 LM Sentence Masking

We also conduct a masking-based interpretability method in which we remove all sentences in the test set that contain at least one word from any LM financial sentiment category and perform inference on the masked test set. We report the model performance in Table 6. We observe that while the trained model is capturing the sentiment conveyed by each category of words, it is not overly reliant on any single category and appears to be multi-faceted and balanced in its ability to utilize multiple types of signals inherent in the transcripts. This indicates that the model is relatively robust to perturbations in the input space, suggesting that it may be less susceptible to manipulation (e.g. if executives try to avoid certain words or phrases they think the market would react negatively to) than keyword based approaches.

LM Category	Accuracy
None	76.56%
Positive	70.41%
Negative	72.11%
Uncertain	69.73%
Litigious	69.39%
Constraining	69.90%
Strong Modal	69.56%
Weak Modal	70.69%

Table 6: Best model performance on the test set after dropping all sentences containing at least 1 word in each respective LM financial sentiment category.

6.4 Forward-Looking Statements

Since a typical conference call contains information about the past, present, and future performance of the company, we wish to understand the importance of forward-looking content to the predictions. In particular, we define sentences that contain certain keywords, such as "will," "expect," "believe," etc., to be forward-looking statements (FLS) according to Li (2010). We then examine the relationship between the number of forward-looking statements in the text (NFLS) and the model performance in Figure 3. Further, we observe that model performance generally increases for larger values of NFLS. We conjecture that higher values of NFLS provide the model with more signal about the firm's future prospects.

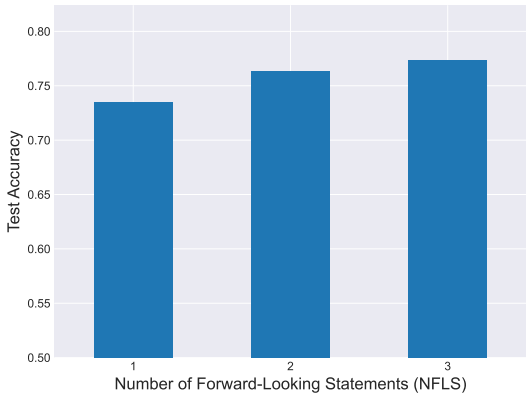

Figure 3: Breakdown of model performance on test set by tercile of number of sentences in the text that contain forward-looking statements.

6.5 Comparison

While the Hierarchical Transformer models are able to process almost the full transcript in an efficient manner, the best model (FinBERT) only outperforms BigBird by about 0.70 absolute percentage points, even though it is able to process more than 5 times the number of tokens. This result seems to indicate that the BigBird architecture, which applies the global attention simultaneously with local attention rather than in a hierarchical fashion, is more effective in this setting, perhaps because of the ability of the model to inject global context into the token-level representations. Therefore, we would expect BigBird to outperform the HTs if it could be extended to support longer sequence lengths and/or adapted to the financial domain. However, given that BigBird is already 50% slower than the HTs, the training

time may become intractable without adjusting to significantly smaller block sizes, and we leave it to future research to identify the best approaches to efficiently extend Efficient Transformer models, such as BigBird, to support longer sequence lengths (Phang et al., 2022).

7 Conclusion

In conclusion, we propose a novel task that uses transcripts from earnings conference calls to predict future earnings surprises. We formulate the problem as a long document classification task and explore a variety of different approaches to address it. While the length and language of the calls presents challenges for generic pretrained language models, we establish several strong baselines. We demonstrate that it is possible to predict companies' future earnings surprises with reasonable accuracy from the solely the text of their earnings conference calls. Further, we probe the model through multiple interpretability methods to uncover intuitive linguistic features that go beyond traditional sentiment analysis.

Limitations

Our experiments demonstrate that it is possible to analyze company executive and analyst language during earnings calls and use it to predict future earnings surprises with reasonable accuracy that is well above random chance. We acknowledge that the dataset contains events that result in significant (in magnitude) earnings surprises so the performance numbers do not directly translate to a live trading setting in which many events do not result in material surprises. We also note that predicting future earnings surprises is correlated with but not equivalent to predicting future stock returns so more work must be done to translate our results into an actual trading strategy that is out of the scope of this paper.

Ethics Statement

We acknowledge that our Earnings Conference Call dataset contains English transcripts from the largest US-based companies so it is possible that some populations may be underrepresented in this sample. We plan to extend this work to international companies and conference calls held in other languages in the future.

Acknowledgements

We thank the anonymous reviewers for their thoughtful comments. We would also like to thank AJO Vista and FactSet for providing access to and permission to release the data. The authors are solely responsible for the content and views expressed in this publication do not reflect those of the affiliated institutions.

References

Iz Beltagy, Matthew E Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*.

Stephen Brown, Stephen A Hillegeist, and Kin Lo. 2004. Conference calls and information asymmetry. *Journal of Accounting and Economics*, 37(3):343–366.

Brian J Bushee, Ian D Gow, and Daniel J Taylor. 2018. Linguistic complexity in firm disclosures: Obfuscation or information? *Journal of Accounting Research*, 56(1):85–121.

Ilias Chalkidis, Manos Fergadiotis, Prodromos Malakasiotis, Nikolaos Aletras, and Ion Androutsopoulos. 2020. Legal-bert: The muppets straight out of law school. *arXiv preprint arXiv:2010.02559*.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Édouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451.

Zihang Dai, Zhilin Yang, Yiming Yang, Jaime G Carbonell, Quoc Le, and Ruslan Salakhutdinov. 2019. Transformer-xl: Attentive language models beyond a fixed-length context. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2978–2988.

Jacob Devlin, Ming Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*, pages 4171–4186.

Jeffrey T Doyle, Russell J Lundholm, and Mark T Soliman. 2006. The extreme future stock returns following i/b/e/s earnings surprises. *Journal of Accounting Research*, 44(5):849–887.

Richard M. Frankel, Jared N. Jennings, and Joshua A. Lee. 2018. Using natural language processing to assess text usefulness to readers: The case of conference calls and earnings prediction. *SSRN Electronic Journal*.

Jiuxiang Gu, Jason Kuen, Vlad I. Morariu, Handong Zhao, Nikolaos Barmpalios, Rajiv Jain, Ani Nenkova, and Tong Sun. 2021. Unified pretraining framework for document understanding. volume 1.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A Smith. 2020. Don't stop pretraining: Adapt language models to domains and tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8342–8360.

Xiaochuang Han and Jacob Eisenstein. 2019. Unsupervised domain adaptation of contextualized embeddings for sequence labeling. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4238–4248.

Allen H Huang, Hui Wang, and Yi Yang. 2022. Finbert: A large language model for extracting information from financial text. *Contemporary Accounting Research*.

Itay Kama. 2009. On the market reaction to revenue and earnings surprises. *Journal of Business Finance & Accounting*, 36(1-2):31–50.

Zheng Tracy Ke, Bryan T Kelly, and Dacheng Xiu. 2019. Predicting returns with text data. Technical report, National Bureau of Economic Research.

Nikita Kitaev, Łukasz Kaiser, and Anselm Levskaya. 2020. Reformer: The efficient transformer. *arXiv preprint arXiv:2001.04451*.

David F Larcker and Anastasia A Zakolyukina. 2012. Detecting deceptive discussions in conference calls. *Journal of Accounting Research*, 50(2):495–540.

Henry A Latane and Charles P Jones. 1979. Standardized unexpected earnings–1971-77. *The journal of Finance*, 34(3):717–724.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2020. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240.

Feng Li. 2010. The information content of forward-looking statements in corporate filings-a naïve bayesian machine learning approach. *Journal of Accounting Research*, 48.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Tim Loughran and Bill Mcdonald. 2011. When is a liability not a liability? textual analysis, dictionaries, and 10-ks. *Journal of Finance*, 66.

Andrew Maas, Raymond E Daly, Peter T Pham, Dan Huang, Andrew Y Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies*, pages 142–150.

Andriy Mulyar, Elliot Schumacher, Masoud Rouhizadeh, and Mark Dredze. 2019. Phenotyping of clinical notes with improved document classification models using contextualized neural language models. *arXiv preprint arXiv:1910.13664*.

Raghavendra Pappagari, Piotr Zelasko, Jesus Villalba, Yishay Carmiel, and Najim Dehak. 2019. Hierarchical transformers for long document classification. In *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 838–844. IEEE.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Jason Phang, Yao Zhao, and Peter J Liu. 2022. Investigating efficiently extending transformers for long input summarization. *arXiv preprint arXiv:2208.04347*.

S McKay Price, James S Doran, David R Peterson, and Barbara A Bliss. 2012. Earnings conference calls and stock returns: The incremental informativeness of textual tone. *Journal of Banking & Finance*, 36(4):992–1011.

Yu Qin and Yi Yang. 2019. 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*, pages 390–401.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. "why should i trust you?" explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1135–1144.

Gerard Salton and Christopher Buckley. 1988. Term-weighting approaches in automatic text retrieval. *Information processing & management*, 24(5):513–523.

Yunxin Sang and Yang Bao. 2022. Predicting corporate risk by jointly modeling company networks and dialogues in earnings conference calls. *arXiv preprint arXiv:2206.06174*.

Ramit Sawhney, Piyush Khanna, Arshiya Aggarwal, Taru Jain, Puneet Mathur, and Rajiv Ratn Shah. 2020a. 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)*, pages 8001–8013.

Ramit Sawhney, Puneet Mathur, Ayush Mangal, Piyush Khanna, Rajiv Ratn Shah, and Roger Zimmermann. 2020b. Multimodal multi-task financial risk forecasting. In *Proceedings of the 28th ACM international conference on multimedia*, pages 456–465.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Ho Chung Wu, Robert Wing Pong Luk, Kam Fai Wong, and Kui Lam Kwok. 2008. Interpreting tf-idf term weights as making relevance decisions. *ACM Transactions on Information Systems*, 26.

Linyi Yang, Jiazheng Li, Ruihai Dong, Yue Zhang, and Barry Smyth. 2022. Numhtml: Numeric-oriented hierarchical transformer model for multi-task financial forecasting. *arXiv preprint arXiv:2201.01770*.

Linyi Yang, Tin Lok James Ng, Barry Smyth, and Riuhai Dong. 2020a. Html: Hierarchical transformer-based multi-task learning for volatility prediction. In *Proceedings of The Web Conference 2020*, pages 441–451.

Liu Yang, Mingyang Zhang, Cheng Li, Michael Bendersky, and Marc Najork. 2020b. Beyond 512 tokens: Siamese multi-depth transformer-based hierarchical encoder for long-form document matching. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, pages 1725–1734.

Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. 2016. Hierarchical attention networks for document classification. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1480–1489, San Diego, California. Association for Computational Linguistics.

Manzil Zaheer, Guru Guruganesh, Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontanon, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, and Amr Ahmed. 2020. Big bird: Transformers for longer sequences. volume 2020-December.

Xingxing Zhang, Furu Wei, and Ming Zhou. 2019. Hibert: Document level pre-training of hierarchical bidirectional transformers for document summarization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5059–5069.

A Appendix

A.1 Implementation Details and Training Process

We use Scikit-learn and XGBoost to develop the non-neural baseline models. We develop all Transformer-based models in PyTorch and source all pretrained checkpoints from HuggingFace.

For the BOW models, we remove stop words, create both unigrams and bigrams from the resulting 50,000 most frequent phrases vocabulary, and apply Term Frequency-Inverse Document Frequency weighting (TF-IDF; Salton and Buckley, 1988; Wu et al., 2008) to create features. For the CNN model, we initialize the word embeddings with pretrained weights from GLOVE (100D; Pennington et al., 2014), select the 50,000 most frequent words as our vocabulary, and truncate all transcripts after the first 12,000 words.

We perform all experiments on a single Tesla A100 GPU with 40GB in memory. We use AdamW to optimize all parameters. We tune the hypeparamters of each neural model by conducting a limited grid search over learning rates $\in \{5e-6, 1e-5, 5e-5\}$, weight decay $\in \{1e-4, 1e-3, 1e-2\}$ and batch size $\in \{32, 64, 128\}$, based off validation set accuracy score. For computational constraints, we train all models using FP16 precision training, and apply gradient checkpointing to satisfy GPU memory constraints, and clip gradient norms. It takes approximately 10 minutes per epoch of supervised finetuning for the Hierarchical Transformer models and 15 minutes per epoch of training for BigBird with block size of 64.

We conduct the MLM pretraining process for BigBird on the training set for a maximum of 10 epochs or until the MLM loss on the validation set increases. This pretraining process takes multiple days of run time and indicates the difficulty of pretraining these Efficient Transformers models on domain relevant text. We tune the block size over $\{32,\ 64,\ 84\}$ and the number of random blocks over $\{3,\ 4,\ 5\}$.

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
Yes, in the dedicated limitations section.

☑ A2. Did you discuss any potential risks of your work?
Yes, in the dedicated limitations and ethics statement sections.

☑ A3. Do the abstract and introduction summarize the paper's main claims?
Left blank.

☒ A4. Have you used AI writing assistants when working on this paper?
Left blank.

B ☑ Did you use or create scientific artifacts?

Left blank.

☑ B1. Did you cite the creators of artifacts you used?
(3)

☑ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
(3)

☑ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
(3)

☐ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
Not applicable. The text transcripts are from publicly available earnings conference calls for US-based public companies.

☑ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
(3.1)

☑ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
(3.1)

C ☑ Did you run computational experiments?

(5)

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
(4.6) and Appendix (A)

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
(4.6) and Appendix (A)

☑ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
(5.1)

☑ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
Appendix (A)

D ☒ Did you use human annotators (e.g., crowdworkers) or research with human participants?

Left blank.

☐ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
Not applicable. Left blank.

☐ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
Not applicable. Left blank.

☐ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
Not applicable. Left blank.

☐ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
Not applicable. Left blank.

☐ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
Not applicable. Left blank.