ECTSum: A New Benchmark Dataset For Bullet Point Summarization of Long Earnings Call Transcripts

Rajdeep Mukherjee[1]* **Abhinav Bohra**[1] **Akash Banerjee**[1] **Soumya Sharma**[1]
Manjunath Hegde[2] **Afreen Shaikh**[2] **Shivani Shrivastava**[2] **Koustuv Dasgupta**[2]
Niloy Ganguly[1,3] **Saptarshi Ghosh**[1] **Pawan Goyal**[1]
[1] Department of Computer Science and Engineering, IIT Kharagpur, India
[2] Goldman Sachs Data Science and Machine Learning Group, India
[3] Leibniz University of Hannover, Germany

Abstract

Despite tremendous progress in automatic summarization, state-of-the-art methods are predominantly trained to excel in summarizing short newswire articles, or documents with strong layout biases such as scientific articles or government reports. Efficient techniques to summarize financial documents, discussing facts and figures, have largely been unexplored, majorly due to the unavailability of suitable datasets. In this work, we present **ECTSum**, a new dataset with *transcripts of earnings calls* (ECTs), hosted by publicly traded companies, as documents, and experts-written short *telegram-style bullet point* summaries derived from corresponding ***Reuters*** articles. ECTs are long unstructured documents without any prescribed length limit or format. We benchmark our dataset with state-of-the-art summarization methods across various metrics evaluating the content quality and factual consistency of the generated summaries. Finally, we present a simple yet effective approach, **ECT-BPS**, to generate a set of bullet points that precisely capture the important facts discussed in the calls.

1 Introduction

Earnings Calls, typically a teleconference or a webcast, are hosted by publicly traded companies to discuss important aspects of their quarterly (10-Q), or annual (10-K) earnings reports, along with current trends and future goals that help financial analysts and investors to review their price targets and trade decisions (Givoly and Lakonishok, 1980; Richard Frankel and Skinner, 1999; Bowen et al., 2002; Keith and Stent, 2019). The corresponding call transcripts (called **Earnings Call Transcripts**, abbreviated as **ECT**s) are typically in the form of long unstructured documents consisting of thousands of words. Hence, it requires a great deal of time and effort, even on the part of trained analysts, to quickly summarize the key facts covered in these transcripts. Given the importance of these calls, they are often summarized by media houses such as *Reuters* and *BusinessWire*. The scale of such effort, however, calls for the development of efficient methods to automate this task which in turn necessitates the creation of a benchmark dataset.

- QUARTERLY EARNINGS PER SHARE $1.52.
- QUARTERLY TOTAL NET SALES $97.28 BILLION VERSUS $89.58 BILLION REPORTED LAST YEAR.
- BOARD OF DIRECTORS AUTHORIZED AN INCREASE OF $90 BILLION TO THE EXISTING SHARE REPURCHASE PROGRAM.
- QUARTERLY IPHONE REVENUE $50.57 BILLION VERSUS $47.94 BILLION REPORTED LAST YEAR.

Table 1: **ECTSum**: Excerpt from the *Reuters* article[1] corresponding to the ECT[2] for **Apple Q2 2022**.

Towards this goal, we present **ECTSum**, a new benchmark dataset for bullet-point summarization of long ECTs. As discussed in Section 3.2, first we crawled around 7.4K ECTs from *The Motley Fool*[3], posted between January 2019 and April 2022, corresponding to the *Russell 3000 Index* companies[4]. ***Reuters*** was chosen to be the source of our target summaries, per consultation with domain experts, since the expert-written articles posted on *Reuters* effectively capture the key takeaways from earnings calls. However, searching for *Reuters* articles corresponding to the collected ECTs was especially challenging, since the task was non-trivial. Given the fact that not all calls are tracked, after carefully performing data cleaning and addressing pairing issues, we arrive at a total of **2,425 document-summary pairs** as part of the dataset.

What makes *ECTSum* truly different from others is the way the summaries are written. Instead of containing well-formed sentences, the articles

*Corresponding author: rajdeep1989@iitkgp.ac.in

[1] https://tinyurl.com/yc3z9sbj
[2] https://tinyurl.com/uyby3vh4
[3] https://www.fool.com/earnings-call-transcripts/
[4] https://www.investopedia.com/terms/r/russell_3000.asp

Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, pages 10893–10906
December 7-11, 2022

contain *telegram-style bullet-points* precisely capturing the important metrics discussed in the earnings calls. A sample reference summary from our dataset corresponding to the 2nd quarter 2022 earnings call of *Apple* is shown in Table 1. There are several other factors that make *ECTSum* a challenging dataset. First, the document-to-summary **compression ratio of 103.67** is the **highest** among existing long document summarization datasets with comparable document lengths (Table 2). Hence, in order to do well, trained models need to be highly precise in capturing the most relevant facts discussed in the ECTs in as few words as possible.

Second, existing long document summarization datasets such as Arxiv/PubMed (Cohan et al., 2018), BigPatent (Sharma et al., 2019), FNS (El-Haj et al., 2020), and GovReport (Huang et al., 2021), have fixed document layouts. ECTs, on the other hand, are free-form documents with salient information spread throughout the text (please refer Section 3.3). Hence, models can no longer take advantage of learning any stylistic signals (Kryściński et al., 2021). Third, the average length of ECTs is around 2.9K words (before tokenization). On the other hand, neural models employing BERT (Devlin et al., 2019), T5 (Raffel et al., 2020), or BART (Lewis et al., 2020) as document encoders cannot process documents longer than 512/1024 tokens. Hence, despite achieving state-of-the-art performances on short-document summarization datasets such as CNN/DM (Nallapati et al., 2016), Newsroom (Grusky et al., 2018), and XSum (Narayan et al., 2018), etc., such models cannot be readily applied to effectively summarize ECTs.

We benchmark the performance of several representative summarization approaches (Section 5.1) from both supervised and unsupervised paradigms, on our newly proposed dataset. Among supervised, we select state-of-the-art methods from extractive, abstractive, and long document summarization literature. Finally, given the pattern of source transcripts and target summaries, we present **ECT-BPS**, a simple yet effective pipeline approach for the task of ECT summarization (Section 4). Specifically, it consists of an **extractive summarization** module followed by a **paraphrasing** module. While, the former is trained to identify salient sentences from the source ECT, the latter is trained to paraphrase ECT sentences to short abstractive telegram-style bullet-points that precisely capture the numerical values and facts discussed in the calls.

In order to demonstrate the challenges of the proposed *ECTSum* dataset, competing methods are evaluated on several metrics that assess the *content quality* and *factual consistency* of the model-generated summaries. These metrics are discussed in Section 5.2. We discuss the comparative results of all considered methods against automatic evaluation metrics in Section 5.4. Given the complex nuances of financial reporting, we further conduct a human evaluation experiment (survey results reported in Section 5.5) where we hire a team of financial experts to manually assess and compare the summaries generated by *ECT-BPS*, and those of our strongest baseline. Overall, both automatic and manual evaluation results show **ECT-BPS** to outperform strong state-of-the-art baselines, which demonstrates the advantage of a simple approach.

Our contributions can be summarized as follows:

- We present **ECTSum**, the first long document summarization dataset in the finance domain that requires models to process long unstructured earning call transcripts and summarize them in a few words while capturing crucial metrics and maintaining factual consistency.
- We propose **ECT-BPS**, a simple approach to effectively summarize ECTs while ensuring factual correctness of the generated content. We establish its better efficacy against strong summarization baselines across all considered metrics evaluating the content quality and factual correctness of model-generated summaries.
- Our dataset and codes are publicly available at https://github.com/rajdeep345/ECTSum

2 Related Works

Automatic text summarization, *extractive* (Nallapati et al., 2017; Zhong et al., 2020), *abstractive* (Zhang et al., 2019; Lewis et al., 2020), as well as *long document summarization* (Zaheer et al., 2020; Beltagy et al., 2020) have seen tremendous progress over the years (Huang et al., 2020). Several works also exist on *controllable summarization* (Mukherjee et al., 2020; Amplayo et al., 2021) and, in specific domains, such as *disaster* (Mukherjee et al., 2022), and *legal* (Shukla et al., 2022). However, the field of financial data summarization remains largely unexplored, primarily due to the unavailability of suitable datasets. Passali et al. (2021) have recently compiled a financial news summarization dataset consisting of around 2K *Bloomberg* articles with corresponding human-written sum-

maries. However, similar to other popular *newswire* datasets such as CNN/DM (Nallapati et al., 2016), Newsroom (Grusky et al., 2018), XSum (Narayan et al., 2018), the documents (news articles) themselves are only a few hundred words long, hence limiting the practical importance of model generated summaries (Kryściński et al., 2021).

To the best of our knowledge, *FNS* (El-Haj et al., 2020) is the only available financial summarization dataset, released as part of the *Financial Narrative Summarization Shared Task 2020* [5]. In *FNS*, annual reports of UK firms constitute the documents, and a subset of *narrative* sections from the reports are given verbatim as reference summaries. However, *ECTSum* differs from *FNS* on several accounts.

First, our target summaries consist of a small set of telegram-style bullet-points, whereas the ones in *FNS* are large extractive portions from respective source documents. Second, *ECTSum* has a very high document-to-summary *compression ratio* (refer Section 3.3), because of which the models are expected to generate extremely concise summaries of around 50 words from lengthy unstructured ECTs, around 2.9K words long. In contrast, the expected length of model-generated summaries on *FNS* is around 1000 words. Finally, the models developed on *FNS* are specifically trained to identify and summarize the *narrative* sections, while completely ignoring others containing facts, and figures that reflect the firm's annual financial performance. Excluding these key performance indicators from summaries limits their practical utility to stakeholders. Models trained on *ECTSum*, on the other hand, are specifically expected to capture salient financial metrics such as sales, revenues, current trends, etc. in as few words as possible.

Previously, Cardinaels et al. (2018) had attempted to summarize earnings calls using standard unsupervised approaches. We are however the first to propose and exhaustively benchmark a large scale financial long document summarization dataset involving earnings call transcripts.

3 Dataset

This section describes our dataset, **ECTSum**, including the data sources, and the steps taken to sanitize the data, in order to obtain the document-summary pairs. Finally, we conduct an in-depth analysis of the dataset and report its statistics.

[5] http://wp.lancs.ac.uk/cfie/fns2020/

3.1 Data Collection

ECTs of listed companies are publicly hosted on *The Motley Fool* [6]. We crawled the web pages corresponding to all available ECTs for the *Russell 3000 Index* companies[7] posted between January 2019 and April 2022. In the process, we obtain a total of 7,389 ECTs. The HTML web pages were parsed using the BeautifulSoup[8] library. ECTs typically consist of two broad sections: *Prepared Remarks*, where the company's financial results, for the given reporting period, are presented; and *Question and Answers*, where call participants ask questions regarding the presented results. We only consider the unstructured text corresponding to the *Prepared Remarks* section to form the source documents.

Collecting expert-written summaries corresponding to these ECTs was a far more challenging task. *Reuters*[9] hosts a huge repository of financial news articles from around the world. Among these, are articles, written by analysts, that summarize earnings calls events in the form of a few bulleted points (see Table 1). After manually going through several such articles, and after consulting experts from *Goldman Sachs, India*, we understood that these articles precisely capture the key takeaways[10] from earnings calls. Accordingly, using the company codes and dates of the earnings call events corresponding to the collected ECTs, we crawled *Reuters* web pages to search for relevant articles. We obtained 3,013 *Reuters* articles in the process.

3.2 Data Cleaning and Pairing

Cleaning the ECTs: Almost all earnings calls (and hence the corresponding transcripts) begin with an introduction by the call moderator/operator. We remove these statements since they do not relate to the financial results discussed thereafter. Some calls directly start with the *Questions and Answers*, in which case we exclude them from the collection.

Cleaning the summaries: For the *Reuters* (summary) articles, first we performed simple preprocessing to split the text into sentences. In many articles, we observed sentences ending with the phrase REFINITIV IBES DATA. Such sentences report estimates made by *Refinitiv*[11] analysts on the

[6] https://www.fool.com/earnings-call-transcripts/
[7] https://www.investopedia.com/terms/r/russell_3000.asp
[8] https://crummy.com/software/BeautifulSoup/
[9] https://www.reuters.com/business/
[10] https://tinyurl.com/27ehcxzf
[11] https://tinyurl.com/2p9e6kh2

Dataset	# Docs.	Coverage	Density	Comp. Ratio	# Tokens	
					Doc.	Summary
ARXIV/PUBMED (Cohan et al., 2018)*	346,187	0.87	3.94	31.17	5179.22	257.44
BILLSUM (Kornilova and Eidelman, 2019)†	23,455	-	4.12	13.64	1813.0	207.7
BIGPATENT (Sharma et al., 2019)*	1,341,362	0.86	2.38	36.84	3629.04	116.67
GOVREPORT (Huang et al., 2021)†	19,466	-	7.60	19.01	9409.4	553.4
BOOKSUM Chapters (Kryściński et al., 2021)*	12,293	0.78	1.69	15.97	5101.88	505.32
ECTSum	2,425	0.85	2.43	**103.67**	2916.44	49.23

Table 2: Comparing the statistics of ECTSum dataset with existing long document summarization datasets. The numbers for the datasets marked with * are copied from Kryściński et al. (2021), whereas the ones marked with † are copied from Huang et al. (2021). Numbers which were not reported are left blank. ECTSum has the highest *compression ratio* among all the datasets while having comparable *coverage* and *density* scores.

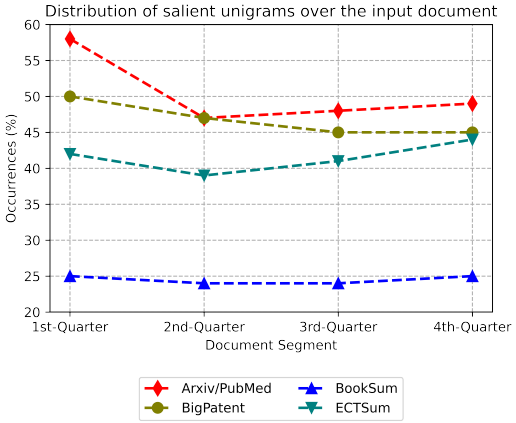


Figure 1: Salient unigram distribution in four equally sized segments of the source text. Higher percentages indicate higher unigram overlap. Percentages more than 25 indicate there are repetitions.

earnings of publicly traded companies. We remove these sentences as they **do not** correspond to the actual results discussed in the earnings calls (as understood from our discussion with financial experts). In the process, we make our target summaries factually consistent with the source documents.

Creating Document-Summary Pairs: In order to automate the process of pairing an ECT with its corresponding *Reuters* article, first we made sure that the article mentions the same company code as the ECT, and second, it is posted either on the same day or at max one day after the earnings event. Please refer to Section A.1 for more details. After obtaining the automatically-matched pairs, the authors manually and independently cross-checked 200 randomly selected ECT(document)-*Reuters*(summary) pairs. We found all the pairs to be properly matched. The process thus ensures accuracy at the cost of obtaining a smaller amount of (sanitized) data. The dataset can however be easily extended as future earnings calls are covered by media houses, such as *Reuters*, and *BusinessWire*.

3.3 Statistics and Analysis

The data cleaning and pairing process described above resulted in a total of **2,425 document-summary pairs**, with average document length of around 2.9K words and average target summary length of around 50 words. We randomly split the data to form the train (70%), validation (10%) and test (20%) sets. In Table 2, we report various dataset statistics, as defined by Grusky et al. (2018), for the *ECTSum* corpus and compare them with the existing long document summarization datasets. While *Coverage* quantifies the extent to which a summary is derivative of a text, *Density* measures how well the word sequence of a summary can be described as a series of extractions. Our scores of 0.85 (Coverage) and 2.43 (Density) are fairly comparable with other datasets. These indicate that although our target summary sentences are short abstractive texts, they are fairly derivable from the ECT content. Our document-to-summary **compression ratio** score of 103.67 is overwhelmingly **higher than any other dataset**. This makes *ECTSum* challenging to work on, and requires models to be trained in a way so that they can capture relevant information in as few words as possible. Both these factors motivated the design of our proposed approach, ECT-BPS (refer to Section 4).

Following prior works (Huang et al., 2021; Kryściński et al., 2021), we further assess whether the target summary content is confined to certain portions of the source document. For this, we plot, in Fig. 1, the percentage distribution of *salient unigrams* (target summary words excluding stopwords) in four equally sized segments of the source text. We observe that the salient content is evenly distributed across all the four segments of the source documents. This property requires models, trained on *ECTSum*, to process the entire document in order to generate a high quality summary.

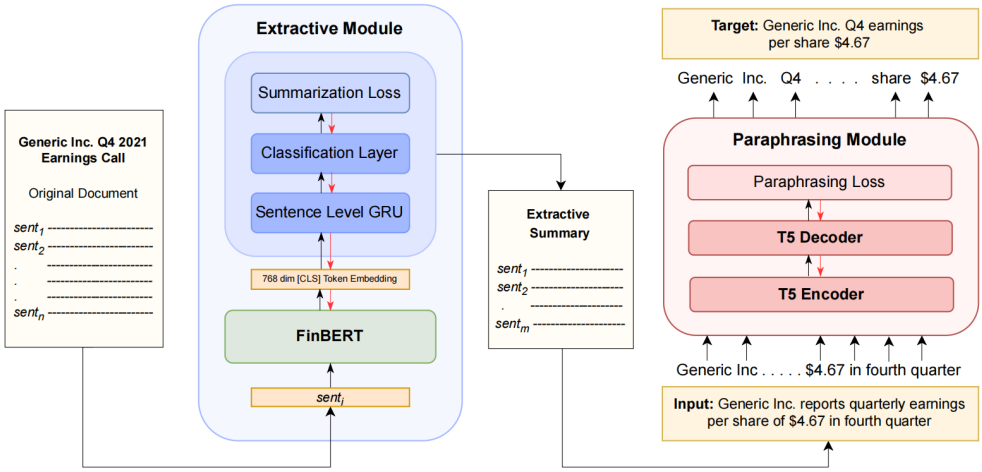


Figure 2: ECT-BPS: Our Proposed Summarization Framework. It consists of an *Extractive Module* that is trained to select highly salient sentences from the source document. The *Paraphrasing Module* is then trained to paraphrase the ECT sentences to the (*Reuters*) format of target summary sentences.

4 The ECT-BPS Framework

We observe some important properties of the *Reuters* reference summaries. They contain a high percentage of word overlap with the source ECT documents. However, they are *not* extractive, rather contain a small set of abstractive bullet-points. It seems as if the analysts writing these summaries first selected some crucial parts of the ECT, before compressing them into a bullet-point format. These properties of the reference summaries motivated us to design a two-stage pipeline approach for summarizing ECTs. Our proposed model **ECT-BPS** contains two separately trained modules/blocks – (1) an *Extractive* block that is trained to identify the most relevant sentences from the input ECT document, and (2) a *Paraphrasing* block that is trained to rephrase the extracted ECT sentences to the format of target (*Reuters*) sentences, thereby generating a set of bullet points. Figure 2 gives an overview of our proposed architecture.

4.1 The Extractive Module

We leverage and suitably modify the architecture of *SummaRuNNer* (Nallapati et al., 2017) to design our extractive module. The vanilla SummaRuNNer consists of a two-layer bi-directional GRU-RNN. The first layer works at the *word-level* to learn contextualized word representations, which are then average-pooled to obtain sentence representations. We replace this layer by **FinBERT** (Yang et al., 2020), a BERT model pre-trained on financial communication text, and use it to obtain the individual sentence representations. The second layer of bi-directional GRU-RNN works at the *sentence-level* to learn contextualized representations of the input ECT sentences. We then obtain the document representation d using the hidden state vectors of sentences from this second layer of bi-directional GRU-RNN as follows:

$$d = tanh(W_d \frac{1}{N_d} \sum_{i=1}^{N_d} [h_i^f, h_i^b] + b) \quad (1)$$

where h_i^f and h_i^b respectively represent the hidden state vectors of the forward, and backward GRUs corresponding to s_i, the i^{th} sentence of the input ECT document. W_d and b represent the weight and bias parameters, respectively. N_d represents the number of sentences in the document.

Each sentence s_i is sequentially revisited in a second pass where a *classification layer* (Fig. 2) takes a binary decision regarding its inclusion in the summary as follows:

$$P(y_i = 1) = f(h_i, sum_i, d, p_i^a, p_i^r, \nu_i) \quad (2)$$

Here, h_i represents a non-linear transformation of $[h_i^f, h_i^b]$. sum_i represents the intermediate representation of the summary formed till s_i is visited. p_i^a, and p_i^r respectively represent the absolute and relative positional embeddings corresponding to s_i. Please refer to Nallapati et al. (2017) for more details. We add a parameter ν_i that is set to 1 if s_i contains numerical values, and 0 otherwise. Keeping in mind the nature of the target summary sentences, that predominantly discuss metrics and numbers, ν_i

guides the classifier to give higher weightage to sentences containing numerical values. Therefore, for each sentence s_i, its *content* $f(h_i)$, *salience* given the document context $f(h_i, d)$, *novelty* considering the summary already formed ($f(h_i, sum_i)$, positional importance, and the fact whether it contains monetary figures, are all taken into account while deciding upon its summary membership.

4.2 The Paraphrasing Module

As depicted in Fig. 2, we fine-tune T5 (Raffel et al., 2020) to paraphrase the input ECT sentences to the telegram-style (*Reuters*) format of target summary sentences. During this paraphrasing, special care is taken to ensure that the numerical values in the input sentences are not rephrased wrongly (hallucinated). More specifically, during training we replace the numerical values in the input sentences with placeholders such as [num-one], [num-two], etc. After obtaining the paraphrased sentences, we replace the placeholders with their original values by performing a simple post-processing step.

4.3 Training and Inference

Target Summary for Extractive Module. Corresponding to each sentence (hereby referred to as the 'target sentence') in the reference summary (obtained from *Reuters*), first we greedily search for a document sentence (using *regular* expressions) that captures all the numerical values mentioned in the target sentence. In case of multiple matches, we select all such document sentences. If no match is found, we select the document sentence that is most similar to the target sentence, in terms of cosine similarity between their embeddings obtained using Google's *Universal Sentence Encoder* (Cer et al., 2018). The selected set of document sentences serve as the *target summary* for training the *Extractive Module*. We train this module by minimizing the *Binary Cross Entropy* loss between the predicted and the true sentence labels.

For training the *Paraphrasing Module*, each sentence in the target summary for the *Extractive Module* becomes the source while the corresponding reference summary sentence becomes the target. The module is trained by minimizing the *Cross-Entropy* loss between the predicted and target tokens.

During **inference**, a test ECT document is sent as input to the trained *Extractive Module*. Sentences corresponding to the extractive summary thus obtained are paraphrased using the trained *Paraphrasing Module* to obtain the final summary.

5 Experiments and Results

In this section, we first enumerate the baselines and evaluation metrics. We then describe our experimental setup, followed by a detailed discussion of our main results. We then report the design and results of a human evaluation experiment conducted to manually assess and compare ECT-BPS-generated summaries with those of competing baselines. We end this section with a qualitative analysis of model-generated summaries.

5.1 Baselines

We evaluate and compare the summarization performance of a wide range of representative algorithms corresponding to various categories on the *ECT-Sum* corpus. The categories together with their specific algorithms are enumerated below:

1. **Unsupervised Approaches: LexRank** (Erkan and Radev, 2004), **DSDR** (He et al., 2012), **PacSum** (Zheng and Lapata, 2019).
2. **Extractive Approaches: SummaRuNNer** (Nallapati et al., 2017), **BertSumEXT** (Liu and Lapata, 2019), **MatchSum** (Zhong et al., 2020).
3. **Abstractive Approaches: BART** (Lewis et al., 2020), **Pegasus** (Zhang et al., 2019), **T5** (Raffel et al., 2020).
4. **Long Document Summarizers: BigBird** (Zaheer et al., 2020), **LongT5** (Guo et al., 2021), **Longformer Encoder Decoder (LED)** (Beltagy et al., 2020).

For more details, please refer to the **appendix** A.2.

5.2 Evaluation Metrics

1. For evaluating the content quality of model-generated summaries, we consider **ROUGE** (Lin, 2004), and **BERTScore** (Zhang et al., 2020). We report the F-1 scores corresponding to ROUGE-1, ROUGE-2, and ROUGE-L.
2. For assessing the factual correctness of the generated summaries, we consider **SummaC$_{\text{CONV}}$** (Laban et al., 2022), a recently proposed NLI-based factual inconsistency detection model.
3. **Num-Prec.**: Accurate reporting of monetary figures is crucial in the financial domain. However, quantity hallucination is a known problem in abstractive summaries (Zhao et al., 2020). In order to evaluate the correctness of values captured in summaries, especially the abstractive ones, we define *Num-Prec.* as the fraction of numerals/values in the model-generated summaries that appear in the source text. Please refer A.3.

Model	ROUGE-1	ROUGE-2	ROUGE-L	BERTScore	Num-Prec.	SummaC$_{CONV}$
Unsupervised						
LexRank (Erkan and Radev, 2004)	0.122	0.023	0.154	0.638	1.00	1.00
DSDR (He et al., 2012)	0.164	0.042	0.200	0.662	1.00	1.00
PacSum (Zheng and Lapata, 2019)	0.167	0.046	0.205	0.663	1.00	1.00
Extractive						
SummaRuNNer (Nallapati et al., 2017)	0.273	0.107	0.309	0.647	1.00	1.00
BertSumExt (Liu and Lapata, 2019)	0.307	0.118	0.324	0.667	1.00	1.00
MatchSum (Zhong et al., 2020)	0.314	0.126	0.335	0.679	1.00	1.00
Abstractive						
BART (Lewis et al., 2020)	0.327	0.153	0.361	0.692	0.594	0.431
Pegasus (Zhang et al., 2019)	0.334	0.185	0.375	0.708	0.783	0.444
T5 (Raffel et al., 2020)	0.363	0.209	0.413	0.728	0.796	0.508
Long Document Summarizers						
BigBird (Zaheer et al., 2020)	0.344	0.252	0.400	0.716	0.844	0.452
LongT5 (Guo et al., 2021)	0.438	0.267	0.471	0.732	0.812	0.516
LED (Beltagy et al., 2020)	0.450	0.271	0.498	0.737	0.679	0.439
Ours						
ECT-BPS w/o Paraphrasing	0.313	0.137	0.351	0.714	1.00	1.00
ECT-BPS	**0.467**	**0.307**	**0.514**	**0.764**	**0.916**	**0.518**

Table 3: Comparison of representative summarizers against automatic evaluation metrics. Best scores are **bold**-ed. For *Num-Prec.* and *SummaC$_{CONV}$*, we highlight the best scores among *abstractive* methods (reasons in Section 5.4). **ECT-BPS**-generated summaries **score the highest** on both content quality as well as factual consistency.

5.3 Experimental Setup

As discussed in Section 4.3, we train the two modules of ECT-BPS separately. For respectively training the *extractive* (and *paraphrasing*) modules, we initialize the FinBERT[12] (and T5[13]) parameters using pre-trained weights from Huggingface (Wolf et al., 2020). In the extractive module, all other parameters were set as defined in Nallapati et al. (2017). The *Extractive* (*Paraphrasing*) module is trained end-to-end with Adam Optimizer with a learning rate of 1e-5 (2e-5) and batch size 8 (16).

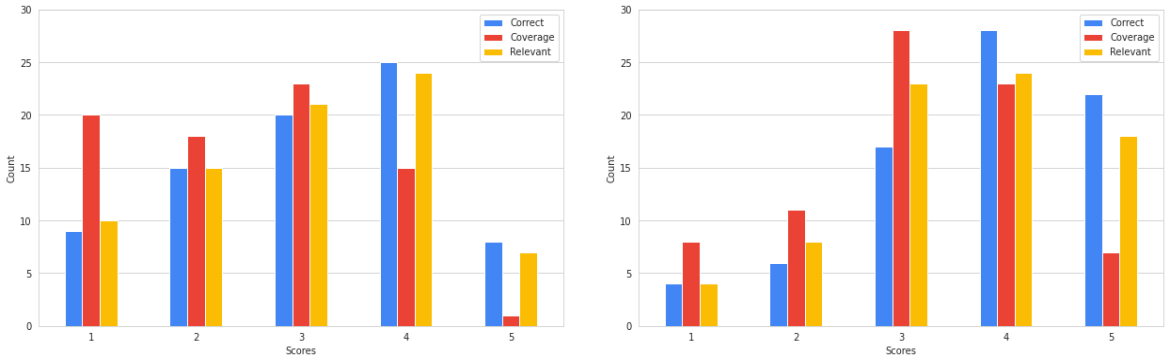
Among the baselines, BART[14] and Pegasus[15] model parameters were initialized with weights pre-trained on financial data. For others, the *base* version of their respective models were used to initialize the parameter weights. All other model hyperparameters were initialized with default values as specified in the respective papers.

All models, including the ECT-BPS modules, were trained end-to-end with hyperparameters fine-tuned on the validation set (recall that we used a 70:10:20 ratio as train:validation:test split). In each case, the model with the lowest validation loss was used to evaluate the test set. All experiments were performed on a Tesla P100-PCIE (16GB) GPU. BART (1024), Pegasus (512), and T5 (512) have limitations on the length of input text that they can process. Since ECTs contain around 2.9K words on an average, for training these *abstractive* methods, we divided the source documents into multiple chunks, each with length less than or equal to their respective `max_token_len`. Corresponding target summaries were made by selecting a subset of all target summary sentences that were entailed by the sentences in the document chunk under consideration. During inference, a small summary (max 32 tokens) was generated from each document chunk. The unique sentences from all such short summaries were concatenated to produce the overall summary for the entire document. Our *ECTSum* dataset, and codes, including baselines, are publicly available on our *GitHub*[16] repository.

5.4 Main Results

Table 3 reports the performance of all competing methods on the test set. All the *unsupervised* methods perform poorly, thereby highlighting the domain-specific nature of the ECT summarization task, and hence the need for supervised training. Among the supervised *extractive* methods, *MatchSum*, a state-of-the-art extractive summarizer, has the best scores across all metrics. Here, we would like to highlight the advantage of the modifications we made to the vanilla *SummaRuNNer* code. Our *Extractive Module*, ECT-BPS w/o Paraphrasing, when compared to *SummaRuNNer*, achieves 18.7% improvement on average across all the *ROUGE*

[12] https://huggingface.co/ProsusAI/finbert
[13] https://huggingface.co/ramsrigouthamg/t5_paraphraser
[14] https://tinyurl.com/26wwaf2e
[15] https://tinyurl.com/mrwpj8mj
[16] https://github.com/rajdeep345/ECTSum



(a) LED summaries **(b) ECT-BPS summaries**

Figure 3: Histogram distribution for human evaluation scores assigned to model-generated summaries.

Model	Correctness	Relevance	Coverage
Summary-level scores (over 75 summaries)			
LED better	18 (24%)	23 (31%)	21 (28%)
ECT-BPS better	45 (60%)	44 (59%)	48 (64%)
Both equally good	12 (16%)	8 (11%)	6 (8%)
Expert-level Scores (over 10 experts)			
LED better	3	3	2
ECT-BPS better	7	7	8

Table 4: Results for the manual evaluation of model-generated summaries by a team of 10 financial experts.

scores, and 10.4% improvement in *BERTScore*. This also makes our *Extractive Module* the best performing extractive method across all metrics. Please note that the *Num-Prec.* and $SummaC_{CONV}$ scores for all extractive summarizers are always 1.00 because the summary sentences are taken verbatim from the source documents.

Among the *abstractive* methods, *Pegasus* and *BART*, despite being initialized with weights pre-trained on financial data, could not match the performance of *T5*. Interestingly, both *T5* (0.508) as well its long version, *LongT5* (0.516), have very good factual consistency scores. These observations led us to select T5 as the backbone of our *paraphrasing* module. *LED* performs better on token overlap metrics (Rouge and BERTscore) but has poor factual consistency scores, highlighting the issue of hallucination in abstractive summarizers (King et al., 2022). To conclude, despite the understandably good performance of *long document summarizers* on the ECT summarization task, our simple extract-then-paraphrase approach, ECT-BPS, establishes the state-of-the-art performance with overall 6.8% better *ROUGE* scores, 3.67% better *BERTScore* scores, 8.5% better *Num-Prec.* scores, and 0.4% better factual consistency scores over the respective strongest baselines.

5.5 Evaluation by Financial Domain Experts

Given the complex nuances of the financial domain, we get the model-generated summaries evaluated by a team 10 analysts/experts working with *Goldman Sachs Data Science and Machine Learning Group, India* who were well-versed with the concepts of financial reporting, earnings calls, etc. For this, we create a survey with 75 randomly chosen test set ECTs and their corresponding summaries generated by ECT-BPS and *LED*, our strongest baseline. Each survey form (please refer to an example[17]) was divided into 5 sections. In each section, the participants were required to go through an entire ECT (*Motley Fool* link provided), and evaluate the two summaries (randomly placed, identity not revealed) on three quality metrics – ***factual correctness***, ***relevance*** and ***coverage*** as defined below:

- **Factual Correctness**: For each summary sentence, the task was to asses if it can be supported by the source ECT.
- **Relevance**: For each summary sentence, the task was to asses if it captures pertinent information relative to the ECT.

The final *correctness/relevance* score of the summary is then determined based on the percentage of sentences that are factually correct/relevant as follows: 5 (>80%), 4 (>60% & $\leq$ 80%), 3 (>40% & $\leq$ 60%), 2 (>20% & $\leq$ 40%), 1 ($\leq$20%). It is to be noted here that *factual correctness* is an objective metric, whereas *relevance* is a subjective metric. For **Coverage**, the participants were instructed to assign a score to the overall summary (on a *Likert* scale of 1-5) based upon their impression about the amount/coverage of relevant content present in it.

Participants were adequately remunerated for their involvement in the task. The summary of

[17] https://forms.gle/pWtexZqM9TXGGoCAA

Summary	Evaluation Scores		
	Correct	Relevant	Coverage
LED-Generated			
q2 revenue rose 27 percent to $667 million.	✓	✓	
sees q3 adjusted earnings per share $12.80 to **$13.90**.	✗	✗	
qtrly adjusted net income per diluted share $3.15.	✓	✓	1
sees fy earnings per common share to be in range of **$12** - $13.00.	✗	✗	
sees 2021 revenue $2.74 billion to $2.791 billion.	✓	✓	
ECT-BPS-Generated			
sees q3 adjusted earnings per share $3.35 to $3.55.	✓	✓	
sees fy ~~**adjusted**~~ earnings per share $12.80 to $13.00.	✗	✗	
sees fy revenue $2.74 billion to $2.79 billion.	✓	✓	3
q2 revenue rose 27 percent to $667 million.	✓	✓	
q2 earnings per share $2.30.	✓	✓	

Table 5: Comparing the summaries generated by LED and ECT-BPS for a given ECT (details in Section 5.6). Parts marked in red are wrongly generated. ECT-BPS better preserves the correctness of generated numbers.

results obtained from this survey are presented in Table 4. At a summary/sample level, respectively for 60% (45/75) and 59% (44/75) of the cases, the summaries generated by ECT-BPS were found to contain more number of factually correct and relevant sentences than the corresponding LED-generated summaries. For 16% and 11% of the cases respectively, the scores for *correctness* and *relevance* were the same for both models. Also, 64% of the times, the participants found ECT-BPS-generated summaries to have a broader *coverage*. When we checked the results of individual experts, 70% of the participants (7 out of 10) found ECT-BPS-generated summaries to be better with respect to *correctness*, and *relevance*. On the other hand, 8 out of 10 participants found ECT-BPS-generated summaries to have a broader *coverage*.

The distribution of absolute scores assigned to the summaries are shown in Fig. 3 as a histogram plot. Here again we find that ECT-BPS-generated summaries are majorly scored ≥ 3 across all three metrics, whereas the majority of LED summaries are scored ≤ 3. Overall, the survey results were found to be comprehensively in favor of ECT-BPS.

5.6 Qualitative Analysis

In Table 5, we qualitatively compare the summaries generated by LED and ECT-BPS corresponding to the earnings call transcript for FleetCor Technologies Inc Q2 2021.[18] The expert evaluation scores corresponding to this pair are also reported. We observe that LED wrongly produces a few monetary values which make the corresponding sentences factually incorrect. Whereas, ECT-BPS maintains the correctness of generated numbers. This may be attributed to our strategy of replacing numbers with placeholders while training the *Paraphrasing* module (please refer to Section 4.2 for details). ECT-BPS however makes a factual error in the second sentence where it misses the word *adjusted*. In the finance domain *adjusted earnings per share* is different from *earnings per share*. These nuances necessitates further research on the ECTSum corpus, and financial summarization in general.

[18] https://tinyurl.com/mph93w46

6 Conclusion

To our knowledge, **ECTSum** is the first large-scale long document summarization dataset in the finance domain. Our documents consist of free-form lengthy transcripts of company earnings calls. Target summaries consist of a set of telegram-style bullet points derived from corresponding *Reuters* articles that cover the calls. Drawing observations from the nature of source transcripts and target summaries, we also propose a simple, yet effective *extract-then-paraphrase* approach, **ECT-BPS**, that establishes state-of-the-art performance over strong summarization baselines across several metrics.

ECTSum is an extremely challenging dataset given the high document-to-summary compression ratio. Moreover, it is highly extendable as future earnings calls are covered by media houses, such as *Reuters*, *BusinessWire*, etc. Finally, it is a very specialized one which would otherwise have costed a lot of time and resources if one had to hire experts to write the reference summaries. The mere observation that these summaries are created by (expert) analysts and can be leveraged automatically is a major milestone of the paper. We believe our contributions to the dataset and methodology will attract future research in the finance domain.

Limitations

In this work, we have restricted ourselves to collecting reference summaries, corresponding to an earnings call transcript, from a single data source, *Reuters*, in our case. Articles summarizing the earnings calls are however published on other media websites as well, for example *CNBC*. In future, we can enrich both the quantity as well as the quality of the dataset, in a scalable manner, by collecting more than one articles from multiple sources thereby resulting in multiple reference summaries corresponding to a single source document.

Despite performing substantially better than strong baseline summarization algorithms, our proposed model ECT-BPS is still a pipeline approach. To our advantage, the improvement in scores over the baselines probably overcomes the increase in the number of model parameters by following an extract-then-paraphrase approach. In future, we would definitely like to address this shortcoming by designing a unified model.

The factual consistency scores obtained using $SummaC_{CONV}$ are generally low across all methods. This gives us ample scope of improvement which in turn calls for further investigation into the nature of factual errors being made by various approaches. A deeper analysis of dataset nuances can also lead us to interesting ideas to improve performance.

Ethics Statement

Given the impact of our proposed contributions on the financial community in particular, and wider research community in general, our dataset and codes have been publicly released. Our document-summary pairs are derived from public/open domain. Still, we may ask users, intending to access our data, to provide a self declaration that the data is to be used solely for research purposes.

References

Reinald Kim Amplayo, Stefanos Angelidis, and Mirella Lapata. 2021. Aspect-controllable opinion summarization. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6578–6593, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

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

Robert M. Bowen, Angela K. Davis, and Dawn A. Matsumoto. 2002. Do conference calls affect analysts' forecasts? *The Accounting Review*, 77(2):285–316.

Eddy Cardinaels, Stephan Hollander, and Brian J. White. 2018. Automatic summaries of earnings releases: Attributes and effects on investors' judgments.

Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St. John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, Brian Strope, and Ray Kurzweil. 2018. Universal sentence encoder for English. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 169–174, Brussels, Belgium. Association for Computational Linguistics.

Arman Cohan, Franck Dernoncourt, Doo Soon Kim, Trung Bui, Seokhwan Kim, Walter Chang, and Nazli Goharian. 2018. A discourse-aware attention model for abstractive summarization of long documents. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 615–621, New Orleans, Louisiana. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Mahmoud El-Haj, Ahmed AbuRa'ed, Marina Litvak, Nikiforos Pittaras, and George Giannakopoulos. 2020. The financial narrative summarisation shared task (FNS 2020). In *Proceedings of the 1st Joint Workshop on Financial Narrative Processing and MultiLing Financial Summarisation*, pages 1–12, Barcelona, Spain (Online). COLING.

Günes Erkan and Dragomir R. Radev. 2004. Lexrank: Graph-based lexical centrality as salience in text summarization. *J. Artif. Int. Res.*, 22(1):457–479.

Dan Givoly and Josef Lakonishok. 1980. Financial analysts' forecasts of earnings: Their value to investors. *Journal of Banking & Finance*, 4(3):221–233.

Max Grusky, Mor Naaman, and Yoav Artzi. 2018. Newsroom: A dataset of 1.3 million summaries with diverse extractive strategies. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 708–719, New Orleans, Louisiana. Association for Computational Linguistics.

Mandy Guo, Joshua Ainslie, David Uthus, Santiago Ontanon, Jianmo Ni, Yun-Hsuan Sung, and Yinfei Yang. 2021. Longt5: Efficient text-to-text transformer for long sequences. *arXiv preprint arXiv:2112.07916*.

Zhanying He, Chun Chen, Jiajun Bu, Can Wang, Lijun Zhang, Deng Cai, and Xiaofei He. 2012. Document summarization based on data reconstruction. In *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence*, AAAI'12, page 620–626. AAAI Press.

Dandan Huang, Leyang Cui, Sen Yang, Guangsheng Bao, Kun Wang, Jun Xie, and Yue Zhang. 2020. What have we achieved on text summarization? In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 446–469, Online. Association for Computational Linguistics.

Luyang Huang, Shuyang Cao, Nikolaus Parulian, Heng Ji, and Lu Wang. 2021. Efficient attentions for long document summarization. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1419–1436, Online. Association for Computational Linguistics.

Katherine Keith and Amanda Stent. 2019. Modeling financial analysts' decision making via the pragmatics and semantics of earnings calls. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 493–503, Florence, Italy. Association for Computational Linguistics.

Daniel King, Zejiang Shen, Nishant Subramani, Daniel S. Weld, Iz Beltagy, and Doug Downey. 2022. Don't say what you don't know: Improving the consistency of abstractive summarization by constraining beam search.

Anastassia Kornilova and Vladimir Eidelman. 2019. BillSum: A corpus for automatic summarization of US legislation. In *Proceedings of the 2nd Workshop on New Frontiers in Summarization*, pages 48–56, Hong Kong, China. Association for Computational Linguistics.

Wojciech Kryściński, Nazneen Rajani, Divyansh Agarwal, Caiming Xiong, and Dragomir Radev. 2021. Booksum: A collection of datasets for long-form narrative summarization. *arXiv preprint arXiv:2105.08209*.

Philippe Laban, Tobias Schnabel, Paul N. Bennett, and Marti A. Hearst. 2022. SummaC: Re-Visiting NLI-based Models for Inconsistency Detection in Summarization. *Transactions of the Association for Computational Linguistics*, 10:163–177.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Yang Liu and Mirella Lapata. 2019. Text summarization with pretrained encoders. 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 3730–3740, Hong Kong, China. Association for Computational Linguistics.

Rajdeep Mukherjee, Hari Chandana Peruri, Uppada Vishnu, Pawan Goyal, Sourangshu Bhattacharya, and Niloy Ganguly. 2020. Read what you need: Controllable aspect-based opinion summarization of tourist reviews. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '20, page 1825–1828, New York, NY, USA. Association for Computing Machinery.

Rajdeep Mukherjee, Uppada Vishnu, Hari Chandana Peruri, Sourangshu Bhattacharya, Koustav Rudra, Pawan Goyal, and Niloy Ganguly. 2022. Mtlts: A multi-task framework to obtain trustworthy summaries from crisis-related microblogs. In *Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining*, WSDM '22, page 755–763, New York, NY, USA. Association for Computing Machinery.

Ramesh Nallapati, Feifei Zhai, and Bowen Zhou. 2017. Summarunner: A recurrent neural network based sequence model for extractive summarization of documents. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, February 4-9, 2017, San Francisco, California, USA*, pages 3075–3081. AAAI Press.

Ramesh Nallapati, Bowen Zhou, Cicero dos Santos, Caglar Gulcehre, and Bing Xiang. 2016. Abstractive text summarization using sequence-to-sequence RNNs and beyond. In *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, pages 280–290, Berlin, Germany. Association for Computational Linguistics.

Shashi Narayan, Shay B. Cohen, and Mirella Lapata. 2018. Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1797–1807, Brussels, Belgium. Association for Computational Linguistics.

Tatiana Passali, Alexios Gidiotis, Efstathios Chatzikyriakidis, and Grigorios Tsoumakas. 2021. Towards

human-centered summarization: A case study on financial news. In *Proceedings of the First Workshop on Bridging Human–Computer Interaction and Natural Language Processing*, pages 21–27, Online. Association for Computational Linguistics.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Marilyn Johnson Richard Frankel and Douglas J. Skinner. 1999. An empirical examination of conference calls as a voluntary disclosure medium. *Journal of Accounting Research*, 37(1):133–150.

Eva Sharma, Chen Li, and Lu Wang. 2019. BIGPATENT: A large-scale dataset for abstractive and coherent summarization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2204–2213, Florence, Italy. Association for Computational Linguistics.

Abhay Shukla, Paheli Bhattacharya, Soham Poddar, Rajdeep Mukherjee, Kripabandhu Ghosh, Pawan Goyal, and Saptarshi Ghosh. 2022. Legal case document summarization: Extractive and abstractive methods and their evaluation. *arXiv preprint arXiv:2210.07544*.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Yi Yang, Mark Christopher Siy UY, and Allen Huang. 2020. Finbert: A pretrained language model for financial communications.

Manzil Zaheer, Guru Guruganesh, Kumar 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. In *Advances in Neural Information Processing Systems*, volume 33, pages 17283–17297. Curran Associates, Inc.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter J. Liu. 2019. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with BERT. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Zheng Zhao, Shay B. Cohen, and Bonnie Webber. 2020. Reducing quantity hallucinations in abstractive summarization. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2237–2249, Online. Association for Computational Linguistics.

Hao Zheng and Mirella Lapata. 2019. Sentence centrality revisited for unsupervised summarization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6236–6247, Florence, Italy. Association for Computational Linguistics.

Ming Zhong, Pengfei Liu, Yiran Chen, Danqing Wang, Xipeng Qiu, and Xuanjing Huang. 2020. Extractive summarization as text matching. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6197–6208, Online. Association for Computational Linguistics.

A Appendix

A.1 Creating Document-Summary Pairs

In order to automate the process of pairing an ECT with its corresponding *Reuters* article, first we made sure that the article mentions the same company code as the ECT, and second, it is posted either on the same day or at max one day after the earnings event. In some cases, we found multiple articles for the same ECT. Upon manual inspection, we classified them into two broad categories: (1) (multiple) articles summarizing the same earnings call, but in parts; (2) articles covering news not directly related to the earnings call. We took the articles of the first category, and merged their distinct sentences into one summary file. After obtaining the automatically-matched pairs, the authors manually and independently cross-checked 200 randomly selected ECT(document)-*Reuters*(summary) pairs. We found all the pairs to be properly matched. The process thus ensures accuracy at the cost of obtaining a smaller amount of (sanitized) data. The dataset can however be easily extended as future earnings call events are covered by media houses, such as *Reuters*, *CNBC*, and *BusinessWire*. We also propose to release subsequent versions of *ECTSum* on our *Github*[19] repository.

A.2 Baselines

We evaluate and compare the summarization performance of a wide range of representative algorithms on the *ECTSum* corpus as briefed below:

A.2.1 Unsupervised Approaches

- **LexRank** (Erkan and Radev, 2004) uses a graph-based lexical centrality metric to score and summarize the document sentences.
- **DSDR** (He et al., 2012) produces a summary consisting of sentences that can best reconstruct the original document.
- **PacSum** (Zheng and Lapata, 2019) is a graph-based algorithm that redefines sentence centrality by taking into account their relative positions in the document to build a directed graph to be used for document summarization.

A.2.2 Extractive Approaches

- **SummaRuNNer** (Nallapati et al., 2017): Vanilla version of our *Extractive Module* (Section 4.1).
- **BertSumEXT** (Liu and Lapata, 2019) takes pre-trained BERT (Devlin et al., 2019) as the sentence encoder and an additional Transformer as the document encoder. A classifier on sentence representations is used for sentence selection.
- **MatchSum** (Zhong et al., 2020) generates a set of candidate summaries from the output of *BertSumEXT*. The candidate that matches best with the document is considered as the final summary.

A.2.3 Abstractive Approaches

- **BART** (Lewis et al., 2020) introduces a denoising autoencoder for pre-training sequence to sequence tasks including summarization.
- **Pegasus** (Zhang et al., 2019) introduces a novel pre-training strategy, *Gap Sentence Generation*, especially suitable for abstractive summarization.
- **T5** (Raffel et al., 2020) systematically applies transfer learning techniques for seq-to-seq generation tasks, including summarization.

A.2.4 Long Document Summarizers

- **BigBird** (Zaheer et al., 2020) applies sparse, global, and random attentions to overcome the quadratic dependency of BERT while preserving the properties of full-attention models. Consequently, it can handle longer context.
- **LongT5** (Guo et al., 2021) extends the original T5 encoder with *Transient Global* attentions to handle long inputs. The model is pre-trained using the PEGASUS strategy.
- **Longformer Encoder Decoder (LED)** (Beltagy et al., 2020) is a *Longformer* variant for supporting long document generative seq-to-seq tasks. It uses an attention mechanism that scales linearly with sequence length, making it easy to process documents of thousands of tokens or longer.

A.3 Evaluation Metrics

For evaluating the content quality and factual correctness of the model-generated summaries, we consider the following evaluation metrics:

- **ROUGE** (Lin, 2004) measures the textual overlap (n-grams, word sequences) between the generated summary and the reference summary. In this work, we report the F-1 scores corresponding to ROUGE-1, ROUGE-2, and ROUGE-L.

[19] https://github.com/rajdeep345/ECTSum

- **BERTScore** (Zhang et al., 2020) aligns the generated and target summaries on a token-level and uses BERT to compute their similarity scores. It correlates better with human judgements. We installed the latest version (0.3.11) of BERTScore from its official implementation[20], and calculated the scores with the recommended NLI model MICROSOFT/DEBERTA-XLARGE-MNLI.

- **Num-Prec.**: Accurate reporting of facts and monetary figures is crucial in the financial domain. Extractive summaries are always expected to contain values that appear in the source text. However, quantity/numeral hallucination is a known problem in abstractive summaries, which prior works (Zhao et al., 2020) have attempted to reduce. Here, we define *Num-Prec.* as the fraction of numerals/values in the model-generated summaries that are consistent with the source text. We use this metric to specifically evaluate the precision/correctness with which abstractive summarizers generate values.

- **SummaC$_{\text{CONV}}$** (Laban et al., 2022) is a recently proposed NLI-based factual inconsistency detection model based on aggregation of sentence-level entailment scores for each pair of input document and summary sentences. We used the official implementation[21] of *SummaC* to obtain the scores for all model-generated summaries.

[20] https://github.com/Tiiiger/bert_score
[21] https://github.com/tingofurro/summac