A Comparative Study of Evaluation Metrics for Long-Document Financial Narrative Summarization with Transformers

Nadhem Zmandar[1 [0000−0002−3087−6762]], Mahmoud El-Haj[1 [0000−0002−6136−3898]], and Paul Rayson[1 [0000−0002−1257−2191]]

UCREL NLP group,
School of Computing and Communications,
Lancaster University, UK
{n.zmandar,m.el-haj,p.rayson}@lancaster.ac.uk

Abstract. There are more than 2,000 listed companies on the UK's London Stock Exchange, divided into 11 sectors who are required to communicate their financial results at least twice in a single financial year. UK annual reports are very lengthy documents with around 80 pages on average. In this study, we aim to benchmark a variety of summarisation methods on a set of different pre-trained transformers with different extraction techniques. In addition, we considered multiple evaluation metrics in order to investigate their differing behaviour and applicability on a dataset from the Financial Narrative Summarisation (FNS 2020) shared task, which is composed of annual reports published by firms listed on the London Stock Exchange and their corresponding summaries. We hypothesise that some evaluation metrics do not reflect true summarisation ability and propose a novel BRUGEscore metric, as the harmonic mean of ROUGE-2 and BERTscore. Finally, we perform a statistical significance test on our results to verify whether they are statistically robust, alongside an adversarial analysis task with three different corruption methods.

Keywords: Long Document sumamrization · Evaluation Metrics · Benchmarking.

1 Introduction

With the proliferation of firms worldwide, the amount of financial disclosures and financial texts (or narratives) in various languages and formats has risen dramatically. Consequently, the study of natural language processing (NLP) methods that automatically summarize content has become a rapidly growing research area [22] [8].
In fact, financial reporting and communication requirements have expanded significantly in recent years, particularly following the 2008 financial crisis. Financial communications and investor relations management are becoming increasingly critical to the financial markets and fund management industry. Regulated financial markets mandate that all listed companies regularly communicate their financial activities to stakeholders by publishing financial reports and other financial narratives.
Financial narratives are employed by firms to communicate with their stakeholders, including investors, shareholders, customers, employees, financial analysts, regulators,

lenders, rating agencies, and suppliers. Through financial communications, stakeholders can assess how well the company is creating value.
The aim of this study is to create and evaluate summarization benchmarks for UK financial narratives, investigate the effect of long document methods, and examine their interactions with various metrics, including ROUGE, in order to assess their suitability for this domain. Additionally, we will introduce a statistical testing method for system-generated financial summaries and the novel BRUGEscore.

2 Background

Summarizing text is a complex task, and standard evaluation metrics such as accuracy, recall, and precision are not suitable for text summarization. In recent years, several metrics have been introduced that are specifically designed for evaluating the quality of machine-generated summaries. In this study, we used the following metrics:

- **ROUGE**: Recall-Oriented Understudy for Gisting Evaluation is a metric used to evaluate the quality of machine-generated summaries by comparing them with a set of human-produced reference summaries. ROUGE measures the number of overlapping textual units, such as n-grams or word sequences, between the generated summary and the reference summaries.
- **BERTScore**: BERTScore is an embedding-based evaluation metric that aligns generated and reference summaries on a token level. Token alignments are computed to maximize the cosine similarity between contextualized token embeddings from the BERT transformer.
- **BARTScore**: BARTScore is an unsupervised evaluation metric used for generative tasks such as machine translation, text summarization, and text generation. It offers a number of variants, depending on the language model used, that can be flexibly applied to evaluate generated text from different perspectives such as informativeness, fluency, or factuality.
- **METEOR**: METEOR computes an alignment between candidate and reference sentences by mapping unigrams in the generated summary to 0 or 1 unigrams in the reference, based on stemming, synonyms, and paraphrastic matches.
- **Bleurt**: Bleurt is a transfer learning-based metric for natural language generation that compares a candidate summary with a reference summary to determine how well the candidate summary conveys the meaning of the reference summary.
- **BRUGEscore**: BRUGEscore is our novel proposed metric, calculated as the harmonic mean of ROUGE-2 and BERTscore. It combines elements of word overlap and embedding cosine similarity into a single score.

Table 1 provides a summary of the features of these metrics, including whether they are embedding-based or n-gram-based.

2.1 Related work

Text summarization has shown promising applications in the financial domain [7]. Prior works in this field have explored a range of approaches. The Summariser system [15]

Metric	Embeddings	Language Model	n-gram
ROUGE	No	N.A	n-gram
BERTScore	Yes	Roberta Large	1-gram
BARTScore	Yes	Bart Large	1-gram
METEOR	No	N.A	1-gram
Bleurt	Yes	BERT-lg	Sequence
BRUGEscore	Yes	N.A	2-gram

Table 1. Summary of the features of the evaluation metrics used in this study

employed sentence linkage heuristics, while a query-based company-tailored summarization system was proposed in [9]. Recently, statistical features with heuristic approaches were used to summarise financial textual disclosures [3]. The Financial Narrative Summarisation (FNS) task of the Multiling 2019 workshop involved generating structured summaries from financial narrative disclosures. The FNS 2020 task [6] resulted in the first large scale experimental results and state-of-the-art summarisation methods applied to financial data, focusing on annual reports produced by UK firms listed on the London Stock Exchange (LSE). The participating systems used a variety of techniques, ranging from rule-based extraction methods to traditional machine learning methods and high-performing deep learning models.

Prior works on UK annual report summarization include [16], who used a transformer-based encoder-decoder extractive summarisation approach based on the T5 pre-trained model. Abhishek Singh [20] proposed a Pointer Network and T5-based summarization approach to extract relevant narrative sentences in a particular order to have a logical flow in the summary. Lei Li [13] used Determinantal Point Processes to build a Statistical learning Extractive Financial Narrative auto Summarizer. Jaime Baldeon Suarez, [1] combined financial word embeddings and knowledge-based features for financial text summarisation, and Moreno La Quatra [11] developed an end-to-end training framework for financial report summarisation in English.

In comparison to prior works, we explore the impact of different transformer model architectures, the task and data used to pre-train transformer models, as well as correlations between automated metrics within the task of summarising UK annual reports. Our work is distinct as UK annual reports are long, unstructured in plain text, technically written, and subjective. Our study aims to address the challenging components of Financial Narrative Summarisation, and this effort is further promoted by the 2021 Financial Narrative Summarisation task (FNS 2021) in the FNP 2021 workshop.

To address the memory efficiency issue of transformers, we cannot simply pass the entire input annual report and gold standard to the model and fine-tune it. Instead, we need to determine which parts of the report to pass to the transformer. Through dataset analysis, we found that the gold standards are typically extracted from the first third of the report, where the chairman or CEO message and financial highlights are usually located. Therefore, we will pass the first k tokens to the model, where k depends on the model architecture, pre-training, and memory efficiency. Then, the model will be trained to predict the first n tokens of the system summary. On the test dataset of 500 UK annual reports, the model will predict the first n tokens, and we will continue the extraction of

the remaining k tokens by determining which part of the report matches the predicted n tokens. This approach transforms the summarization problem into a task of predicting the start of the summary, allowing us to adapt sequence-to-sequence transformer models to summarize long documents where the reference summary is a continuous extracted part of the original text. We refer to this technique as the block-based summarization approach. This technique surpasses the memory efficiency issue of some transformers and is motivated by the fact that reference summaries are extracted from the financial annual report as a block. To our knowledge, this is the best approach for adapting encoder-decoder transformer models to summarize long documents.

We describe several techniques for summarization in this paper, including transformer-based [16], reinforcement learning-based [23], unsupervised learning using LSA, BERT extractive [14], and SBERT extractive summarisation [19]. We also compare the results of these techniques to four toplines and baseline summarizes, as we show later in the papers, and finally, we use Lead-1000 (the first 1000 words) as a strong baseline summarizer [17].

The block-based summarization approach is described as a method of adapting sequence-to-sequence transformer models to summarize long documents where the reference summary is a continuous extracted part of the original text [16]. RL-based summarization is also discussed as a suitable approach for maximizing a predefined metric [23]. Finally, we briefly explain LSA [10], BERT extractive [14], and SBERT extractive [19] as unsupervised techniques that can be used to identify important sentences in a document.

3 Dataset

The dataset used for this study is composed of UK annual reports in English from the financial summarisation shared task (FNS 2021) [22]. It contains 3,863 annual reports for firms listed on the London Stock Exchange (LSE) covering the period between 2002 and 2017, with an average length of 52,000 tokens. The dataset also includes 9,873 gold standard summaries. The dataset is randomly split into training (75%), testing, and validation (25%). Table 2 shows the dataset details.

Data Type	Train	Validate	Test
Report full text	3,000	363	500
Gold summaries	9,873	1,250	1,673

Table 2. FNS 2021 Shared Task Dataset

The dataset used for training presents several gold standard summaries for each annual report (between one and five) [22]. We wanted to use multiple references to make the process more objective since we did not have a human-generated reference summary as a good gold standard. The gold standards used in this study were Financial Highlights, Letter to the Shareholders, Financial Statements, and Auditor's Report.

4 Experimental Work

In our experiments, we used various transformer models used in the study, including the T5 transformer [18], LongFormer Encoder-Decoder [2], as well as BART, Pegasus, and BERT [12] [21] [4].

In our study, we investigated whether using multiple gold standard summaries would improve the performance of summarization models. To fine-tune the models, we first considered the issue of gold summary standards. We trained T5, Pegasus, and BART using two different strategies. The first strategy involved using all available gold standards, which meant creating multiple pairs for each report. The second strategy was to choose only one gold summary that maximized the ROUGE metric [23], which was the aim of the FNS task. Our preliminary study found that training on a multi-referenced dataset did not significantly improve the ROUGE result and was computationally expensive. Therefore, we chose to train our models using only one reference summary per annual report. We set our reward function as ROUGE-2 and selected the gold standard summary that maximized the ROUGE-2 score with the annual report. This enables our system sumamrisers to maximize the Rouge metric with all the reference summaries.

For hyperparameter search, a comprehensive grid search is a common approach. However, due to the significant computational power and time required, we opted for a simpler strategy. We selected hyperparameters that maximize the input length and target length for our models, as detailed in Table 3. In this study, we used metrics that

Transformer	model_name	max_input	max_target	batch_size	train_epochs
T5	t5-small	4096	512	4	5
LED base	allenai/led-base-16384	8000	1000	4	5
LED large	allenai/led-large-16384	4096	512	4	5
Pegasus	google/pegasus-large	1024	256	4	5
BART	facebook/bart-base	1024	128	4	5

Table 3. description of hyperparameters during training on the FNS dataset

support multiple references to evaluate the performance of our models. To compute the score between the system summary and all the gold standards, we used the Rouge.2.0 java jar[1] file for ROUGE evaluation. We removed English stop-words but did not use an English stemmer. For other metrics, we used the implementation from the original authors or the implementation of the Hugging Face team on the datasets library. Table 4 provides a summary of the results. We compared the best version of each transformer model with different baselines and toplines, as well as our new BRUGEscore. F1 scores were reported for each metric, including four variants of the rouge score (R1, R2, R-L, R-SU4), BERT and BART scores, Meteor and Bleurt scores. To compute the embedded representation, we used the Roberta-large-mnli and Bart-large-mnli language models for BERTScore and BARTScore, respectively.

[1] `https://github.com/kavgan/ROUGE-2.0`

The results suggest that model-based metrics give good results on the financial dataset, and that Bleurt is not a suitable metric to evaluate system performance. T5 is the best text-to-text model for the dataset, performing well alongside Longformer Encoder-Decoder. LED base is memory-efficient and performs very well on the dataset, while LED Large did not perform as well due to limited GPU memory. The BRUGEscore shows a harmonic mean between the Rouge2 score and BERT score, giving an equilibrium between sentence semantics and exact 2-gram matching. Lead-1000 is a strong benchmark in this task, indicating the superiority of transformer-based summarisation over deep learning and reinforcement learning methods.

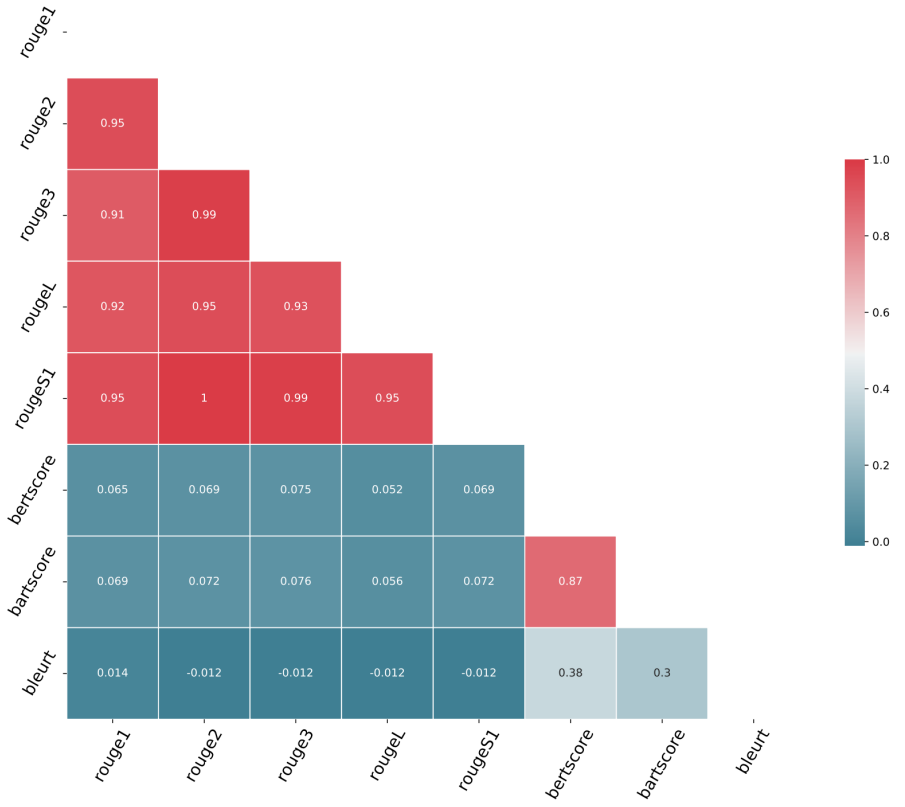

Fig. 1. Correlation Matrix of Scores Produced using T5

Figure 1 shows the correlation matrix of different evaluation metrics' scores using summaries produced by the T5 transformer models which was pre-trained on the FNS test dataset[2]. The correlation plot shows that the different variants of the ROUGE

[2] We only display the T5 matrix as it aligns with our conclusion, and the matrices of the other transformers exhibit similar patterns.

metric are highly correlated, motivating the use of only one ROUGE variant in the evaluation process. Additionally, BERTScore and BARTScore are highly correlated, while BERTscore, BARTscore, and Bleurt are not correlated with the different variants of ROUGE.

System / Metric	R-1/F	R-2/F	R-L/F	R-SU4/ F	BE/F	BA/F	bleurt	meteor	BR
T5-Small-96	0.496	0.374	0.487	0.417	0.910	0.830	-0.836972	0.184	0.530
LED-base-128	0.492	0.370	0.484	0.413	0.899	0.816	-0.849750	0.182	0.524
Pegasus	0.476	0.350	0.467	0.394	0.847	0.759	-0.925372	0.174	0.495
BART	0.453	0.317	0.440	0.365	0.852	0.774	-0.928474	0.176	0.462
Lead-1000	0.443	0.307	0.431	0.356	0.774	0.694	-1.039358	0.162	0.440
RNN-LSTM-RL	0.459	0.270	0.431	0.268	0.761	0.647	-1.027724	0.175	0.399
MUSE-topline	0.433	0.234	0.419	0.253	0.756	0.655	-1.045138	0.163	0.357
LSA	0.321	0.140	0.287	0.187	0.782	0.651	-0.945594	0.160	0.237
SBERT-extractive	0.322	0.139	0.276	0.187	0.781	0.647	-0.973918	0.159	0.236
BERT-extractive	0.312	0.134	0.263	0.182	0.771	0.632	-0.987254	0.121	0.228
LexRank	0.264	0.120	0.253	0.140	0.732	0.580	-1.051438	0.088	0.206
POLY-BASELINE	0.274	0.105	0.212	0.135	0.723	0.565	-1.060618	0.109	0.183
TextRank	0.172	0.070	0.242	0.079	0.727	0.576	-1.074088	0.088	0.128

Table 4. F-measure scores for Rouge-1, Rouge-2, Rouge-L, SU4, BERTScore, BARTScore, Bleurt, and Meteor, ranked based on Rouge-2 F1 measure. The abbreviations used are BE for BERT score (roberta-large-mnli), BA for BART score (bart-large-mnli), and BR for BRUGEscore.

5 Statistical significance

To compare the performance of two algorithms or models, we need to prove that the evaluation metric, denoted by ‘e’, is greater for one system than the other. However, this is not sufficient as we also need to check the statistical significance of the difference in performance between the two algorithms. The common practice in NLP is to claim superiority of one algorithm over another only if the difference in results is statistically significant. To do that, we use significance levels and p-values to determine whether the test results are statistically significant, to avoid false discoveries. We follow the guidelines from the Hitchhiker’s Guide to Testing Statistical Significance in NLP" [5]. We model our problem as a “no difference” (null hypothesis H0) or “difference” (H1) and choose the bootstrap test to verify the significance of our results. We apply our test to the difference between the series of results generated by each system, report the p-values of ROUGE-2, ROUGE-L, BERTscore, and Bleurt score as shown in Tables 6 to 9 in Appendix A. We present the p-values of ROUGE-2, ROUGE-L, BERTscore, and Bleurt score in the tables obtained through the Bootstrap method. These p-values, when compared to the significance level (0,1), indicate the significance of the performance difference between the two systems. Cells that are not coloured red indicate a statistically insignificant difference, allowing us to claim with 90% confidence that system one system outperforms the other using a specific metric.

6 Adversarial analysis

To assess the robustness of the metrics, we also conducted an adversarial analysis on the predicted summaries. Adversarial attacks are text perturbations designed to test the effectiveness of the metrics. Our experiments involved corrupting a set of summaries generated by the T5 small model, which was the best-performing model on the test dataset. We tested the ability of the metrics to resist different sources of noise using **a) BERT mask-filling, b) word-dropping, and c) word permutation** methods. BERT mask-filling and word-dropping are derived from the method used to pre-train BLEURT, while word permutation tests the metrics' sensitivity to syntax by swapping the ordering of two adjacent tokens in the summary. We chose four values of chunks to avoid creating a bias in the distribution of corrupted tokens: 4, 6, 8, and 10. By uniformly distributing the corruption across the text, we can evaluate how well the metrics reflect the difference between the corrupted and uncorrupted text. We anticipate that higher-quality summaries will be more robust to noise. Word-dropping simulates some of the common issues that can arise with extractive summarization. BERT mask-filling is a denoising encoding task that is challenging for BERT score since it assumes that the predicted word by a BERT model is better in this context than the original word in the system summary. Word permutation will penalize the n-gram based metrics but will favour model-based metrics like BERT score and BART score.

To compare the original and corrupted summaries, we use a strict comparison where the original summary must be strictly better than the corrupted one. Table 5 shows the results for the three adversarial tasks with a chunk length of 10. The accuracy value represents the percentage of non-corrupted summaries that received better scores than their corrupted counterparts. An accuracy of 0.00 indicates that the corrupted and non-corrupted summaries received the same scores, as with ROUGE-1 during the word permutation corruption test. This is because ROUGE-1 is insensitive to syntax.

BERTScore and BARTScore achieved an accuracy score of 60% across the three different tasks. These results suggest that **ROUGE** is better suited for extractive summarization while model-based metrics are more suitable for abstractive summarization. ROUGE evaluates summaries on a word-by-word basis, whereas model-based metrics consider the context as a whole. The results also show that ROUGE-2 performed best on the word permutation and BERT mask-filling tasks, while ROUGE-3 performed best on the word dropping task. When the corruption is applied to a single token in a sentence, it disrupts the n-gram sequence, which impacts ROUGE-n when n is greater than 1. Bleurt returned poor results, confirming that it is more suitable for comparing different models than evaluating a single model.

7 Conclusion and Future Work

This paper tackled the task of automatic financial extractive summarisation of UK annual reports using various transformer models and unsupervised baselines. We proposed a set of model-based evaluation metrics, including a new metric called BRUGEscore, which outperformed ROUGE metric variants. We analyzed the results and performed adversarial analysis on the system-generated summaries to verify the robustness of the

Metric	Word dropping_10 (%)	Word Permutation_10 (%)	Bert Mask filling_10(%)
ROUGE-1	0.826	0.000	0.982
ROUGE-2	0.958	1	0.99
ROUGE-3	0.968	0.998	0.992
ROUGE-S1	0.958	1	0.99
ROUGE-S2	0.946	0.996	0.992
ROUGE-L	0.922	0.978	0.99
ROUGE-SU4	0.88	0.994	0.992
BERTScore	0.608	0.63	0.668
BARTScore	0.636	0.6	0.656
BLEURT	0.556	0.632	0.574

Table 5. Mean accuracy by metric on the three corruption tasks. We apply three types of corruptions on the system generated summaries. We create a corruption every 10 chunks. Each metric is used to score the original and the corrupted versions of these summaries. This task should give the uncorrupted version a higher score to make sure that the metric is sensitive to corrupted summaries. The results reported shows the accuracy by metric on this task. All standard deviations were small (less than 0.2%). The experiments were performed on the FNS dataset using the best performing system which is the small version of T5 transformer

metrics. In the future, we plan to perform a human evaluation task on our dataset, measure the correlation with existing evaluation metrics, and work on improving the quality of the reference summaries. All PyTorch models are hosted on a private huggingface repository and will be released once the paper is accepted.

8 Limitations

The lack of gold standards, specifically human-generated summaries by domain experts, is the biggest technical challenge facing the financial text summarisation research community. We currently use extracted sections from annual reports as gold summaries. Furthermore, the results are limited to this English dataset, and the performance of evaluation metrics on other languages cannot be guaranteed, especially for language model-based models that are pretrained on English. Financial datasets are also large and scalable, requiring significant computational capacities. Finally, the jargon used in financial disclosures is different from 'general' language, and there is an urgent need to pre-train financial-specific language models for use in such studies.

References

1. Baldeon Suarez, J., Martínez, P., Martínez, J.L.: Combining financial word embeddings and knowledge-based features for financial text summarization UC3M-MC System at FNS-2020. In: Proceedings of the 1st Joint Workshop on Financial Narrative Processing and MultiLing Financial Summarisation. pp. 112–117. COLING, Barcelona, Spain (Online) (Dec 2020), https://aclanthology.org/2020.fnp-1.19
2. Beltagy, I., Peters, M.E., Cohan, A.: Longformer: The Long-Document Transformer. arXiv:2004.05150 [cs] (Dec 2020), http://arxiv.org/abs/2004.05150, arXiv: 2004.05150

3. Cardinaels, E., Hollander, S., White, B.J.: Automatic summarization of earnings releases: attributes and effects on investors' judgments. Review of Accounting Studies **24**(3), 860–890 (2019)
4. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: 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). pp. 4171–4186. Association for Computational Linguistics, Minneapolis, Minnesota (Jun 2019). https://doi.org/10.18653/v1/N19-1423, https://aclanthology.org/N19-1423
5. Dror, R., Baumer, G., Shlomov, S., Reichart, R.: The hitchhiker's guide to testing statistical significance in natural language processing. In: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). vol. 1, p. 1383–1392 (2018)
6. El-Haj, M., Litvak, M., Pittaras, N., Giannakopoulos, G., et al.: The financial narrative summarisation shared task (fns 2020). In: Proceedings of the 1st Joint Workshop on Financial Narrative Processing and MultiLing Financial Summarisation. pp. 1–12 (2020)
7. El-Haj, M., Rayson, P., Walker, M., Young, S., Simaki, V.: In search of meaning: Lessons, resources and next steps for computational analysis of financial discourse. Journal of Business Finance & Accounting **46**(3-4), 265–306 (2019)
8. El-Haj, M., Zmandar, N., Rayson, P., AbuRa'ed, A., Litvak, M., Pittaras, N., Giannakopoulos, G., Kosmopoulos, A., Carbajo-Coronado, B., Moreno-Sandoval, A.: The financial narrative summarisation shared task (FNS 2022). In: Proceedings of the 4th Financial Narrative Processing Workshop @LREC2022. pp. 43–52. European Language Resources Association, Marseille, France (Jun 2022), https://aclanthology.org/2022.fnp-1.6
9. Filippova, K., Surdeanu, M., Ciaramita, M., Zaragoza, H.: Company-oriented extractive summarization of financial news. In: Proceedings of the 12th Conference of the European Chapter of the ACL (EACL 2009). pp. 246–254 (2009)
10. Gong, Y., Liu, X.: Generic text summarization using relevance measure and latent semantic analysis. In: SIGIR '01 (2001)
11. La Quatra, M., Cagliero, L.: End-to-end Training For Financial Report Summarization. In: Proceedings of the 1st Joint Workshop on Financial Narrative Processing and MultiLing Financial Summarisation. pp. 118–123. COLING, Barcelona, Spain (Online) (Dec 2020), https://aclanthology.org/2020.fnp-1.20
12. Lewis, M., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A., Levy, O., Stoyanov, V., Zettlemoyer, L.: 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. pp. 7871–7880. Association for Computational Linguistics, Online (Jul 2020). https://doi.org/10.18653/v1/2020.acl-main.703, https://aclanthology.org/2020.acl-main.703
13. Li, L., Jiang, Y., Liu, Y.: Extractive Financial Narrative Summarisation based on DPPs. In: Proceedings of the 1st Joint Workshop on Financial Narrative Processing and MultiLing Financial Summarisation. pp. 100–104. COLING, Barcelona, Spain (Online) (Dec 2020), https://aclanthology.org/2020.fnp-1.17
14. Miller, D.: Leveraging bert for extractive text summarization on lectures (2019)
15. de Oliveira, P.C.F., Ahmad, K., Gillam, L.: A financial news summarization system based on lexical cohesion. In: Proceedings of the International Conference on Terminology and Knowledge Engineering, Nancy, France (2002)
16. Orzhenovskii, M.: T5-LONG-EXTRACT at FNS-2021 shared task. In: Proceedings of the 3rd Financial Narrative Processing Workshop. pp. 67–69. Association for Computational Linguistics, Lancaster, United Kingdom (15-16 Sep 2021), https://aclanthology.org/2021.fnp-1.12

17. Radev, D.R., Jing, H., Styś, M., Tam, D.: Centroid-based summarization of multiple documents. Information Processing & Management **40**(6), 919–938 (2004)
18. Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., Liu, P.J.: Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. arXiv:1910.10683 [cs, stat] (Jul 2020), http://arxiv.org/abs/1910.10683, arXiv: 1910.10683
19. Reimers, N., Gurevych, I.: Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP). pp. 3982–3992. Association for Computational Linguistics, Hong Kong, China (Nov 2019). https://doi.org/10.18653/v1/D19-1410, https://aclanthology.org/D19-1410
20. Singh, A.: PoinT-5: Pointer Network and T-5 based Financial Narrative Summarisation. In: Proceedings of the 1st Joint Workshop on Financial Narrative Processing and MultiLing Financial Summarisation. pp. 105–111. COLING, Barcelona, Spain (Online) (Dec 2020), https://aclanthology.org/2020.fnp-1.18
21. Zhang, J., Zhao, Y., Saleh, M., Liu, P.J.: PEGASUS: Pre-training with Extracted Gap-sentences for Abstractive Summarization. arXiv:1912.08777 [cs] (Jul 2020), http://arxiv.org/abs/1912.08777, arXiv: 1912.08777
22. Zmandar, N., El-Haj, M., Rayson, P., Abura'Ed, A., Litvak, M., Giannakopoulos, G., Pittaras, N.: The financial narrative summarisation shared task FNS 2021. In: Proceedings of the 3rd Financial Narrative Processing Workshop. pp. 120–125. Association for Computational Linguistics, Lancaster, United Kingdom (15-16 Sep 2021), https://aclanthology.org/2021.fnp-1.22
23. Zmandar, N., Singh, A., El-Haj, M., Rayson, P.: Joint abstractive and extractive method for long financial document summarization. In: Proceedings of the 3rd Financial Narrative Processing Workshop. pp. 99–105. Association for Computational Linguistics, Lancaster, United Kingdom (15-16 Sep 2021), https://aclanthology.org/2021.fnp-1.19

Appendix A

T5-Small-96	
LED-BASE-128	0.0448
LED-BASE-256	0.0161 0.3353
LED-BASE-1000	0.0042 0.1930 0.3210
BART	0.0000 0.0000 0.0000 0.0000
mBART	0.0000 0.0000 0.0000 0.0000 0.4943
PEGASUS	0.0000 0.0000 0.0000 0.0000 0.2440 0.2429
T5-MULTI-REFERENCES	0.0000 0.0000 0.0000 0.0000 0.2373 0.2306 0.4825
T5-Small-256	0.0000 0.0000 0.0000 0.0000 0.0106 0.0064 0.0378 0.0351
T5-Small-512	0.0000 0.0000 0.0000 0.0000 0.0015 0.0022 0.0112 0.0059 0.2930
PEGASUS-MULTI-REFERENCES	0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0001
BART-MULTI-REFERENCES	0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0077
LSA	0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0550 0.0000
SBERT-EXTRACTIVE	0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0300 0.0000 0.2948
LEAD-1000	0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0093 0.0172
LED-LARGE	0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0068 0.0132 0.4912
BERT-EXTRACTIVE	0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.1234 0.1335
RNN-LSTM-RL	0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0001 0.0001 0.0000
MUSE	0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0329
LEXRANK	0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
TEXTRANK	0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0423
POLY	0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0003 0.1001

Table 6. The p-values of the BERT score results using the Bootstrap test are presented in each column, where column i includes the p-values of system i and the p-values of the remaining n-i systems.

T5-Small-96	
LED-BASE-128	0.2622
LED-BASE-256	0.1766 0.3579
LED-BASE-1000	0.0774 0.2204 0.3232
PEGASUS	0.0000 0.0001 0.0004 0.0015
BART	0.0000 0.0000 0.0003 0.0011 0.4284
mBART	0.0000 0.0001 0.0001 0.0009 0.4227 0.4961
LSA	0.0000 0.0000 0.0000 0.0000 0.1042 0.1160 0.1261
SBERT-EXTRACTIVE	0.0000 0.0000 0.0000 0.0000 0.0010 0.0006 0.0011 0.0134
T5-Small-256	0.0000 0.0000 0.0000 0.0000 0.0020 0.0013 0.0008 0.0142 0.4225
T5-MULTI-REFERENCES	0.0000 0.0000 0.0000 0.0000 0.0014 0.0008 0.0013 0.0113 0.3947 0.4955
PEGASUS-MULTI-REFERENCES	0.0000 0.0000 0.0000 0.0000 0.0008 0.0009 0.0015 0.0066 0.3410 0.4104 0.4304
BERT-EXTRACTIVE	0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0004 0.1452 0.2396 0.2400 0.3016
T5-Small-512	0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0233 0.0598 0.0539 0.0906 0.1589
RNN-LSTM-RL	0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0004 0.0002 0.0005 0.0008 0.0177
LEAD-1000	0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0001 0.0000 0.0005 0.1594
LED-LARGE	0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0007 0.1439 0.4598
BART-MULTI-REFERENCES	0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0006 0.1427 0.4502 0.4757
MUSE	0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0002 0.0586 0.3071 0.3319 0.3543
LEXRANK	0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0164 0.1609 0.1706 0.1829 0.2828
POLY	0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0015 0.0284 0.0433 0.0390 0.0723 0.1871
TEXTRANK	0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0014 0.0004 0.0018 0.0026 0.0171 0.0946

Table 7. The p-values of the Bleurt score results using the Bootstrap test are presented in each column, where column i includes the p-values of system i and the p-values of the remaining n-i systems.

T5-Small-96	
LED-BASE-128	0.1243
LED-BASE-256	0.0777 0.1649
LED-BASE-1000	0.0453 0.0874 0.2068
PEGASUS	0.0000 0.0000 0.0000 0.0000
T5-MULTI-REFERENCES	0.0000 0.0000 0.0000 0.0000 0.0003
T5-Small-256	0.0000 0.0000 0.0000 0.0000 0.0001 0.2906
PEGASUS-MULTI-REFERENCES	0.0000 0.0000 0.0000 0.0000 0.0000 0.1807 0.3979
T5-Small-512	0.0000 0.0000 0.0000 0.0000 0.0000 0.0524 0.2062 0.2628
BART	0.0000 0.0000 0.0000 0.0000 0.0000 0.0003 0.0023 0.0089 0.0187
mBART	0.0000 0.0000 0.0000 0.0000 0.0000 0.0007 0.0021 0.0091 0.0191 0.0000
BART-MULTI-REFERENCES	0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0021 0.2190 0.2158
LED-LARGE	0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0526 0.0550 0.1431
LEAD-1000	0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0273 0.0300 0.0724 0.4301
RNN-LSTM-RL	0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
MUSE	0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
LSA	0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
SBERT-EXTRACTIVE	0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.3925
BERT-EXTRACTIVE	0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0070 0.0002
LEXRANK	0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
POLY	0.0000 0.0000
TEXTRANK	0.0000 0.0000

Table 8. The p-values of the Rouge-2 score results using the Bootstrap test are presented in each column, where column i includes the p-values of system i and the p-values of the remaining n-i systems.

T5-Small-96	
LED-BASE-256	0.1075
LED-BASE-128	0.1330 0.7523
LED-BASE-1000	0.0520 0.1245 0.0740
PEGASUS	0.0000 0.0000 0.0000 0.0000
T5-MULTI-REFERENCES	0.0000 0.0000 0.0000 0.0000 0.0000
PEGASUS-MULTI-REFERENCES	0.0000 0.0000 0.0000 0.0000 0.0000 0.2039
T5-Small-256	0.0000 0.0000 0.0000 0.0000 0.0000 0.2694 0.5332
T5-Small-512	0.0000 0.0000 0.0000 0.0000 0.0000 0.1049 0.3750 0.3654
BART	0.0000 0.0000 0.0000 0.0000 0.0000 0.0010 0.0138 0.0086 0.0159
mBART	0.0000 0.0000 0.0000 0.0000 0.0000 0.0012 0.0156 0.0075 0.0157 0.0000
BART-MULTI-REFERENCES	0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0008 0.0015 0.2342 0.2341
LEAD-1000	0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0220 0.0204 0.0375
LED-LARGE	0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0483 0.0394 0.1032 0.5720
RNN-LSTM-RL	0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0154 0.0156 0.0451 0.4762 0.4233
MUSE	0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0001 0.0000
LSA	0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
SBERT-EXTRACTIVE	0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
BERT-EXTRACTIVE	0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
LEXRANK	0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 0.0031
TEXTRANK	0.0000 0.0000
POLY	0.0000 0.0000

Table 9. The p-values of the Rouge-L score results using the Bootstrap test are presented in each column, where column i includes the p-values of system i and the p-values of the remaining n-i systems.