Summarizing Financial Reports with Positional Language Model

1st Natalia Vanetik
Dept. Of Software Engineering
Shamoon Academic College
Beer Sheva, Israel
natalyav@sce.ac.il

2nd Elizaveta Podkaminer
Dept. Of Software Engineering
Shamoon Academic College
Beer Sheva, Israel
elizapo@ac.sce.ac.il

3rd Marina Litvak
Dept. Of Software Engineering
Shamoon Academic College
Beer Sheva, Israel
marinal@sce.ac.il

***Abstract*—Financial reports are essential for the decision-making processes of various stakeholders, containing vast amounts of both quantitative and qualitative data. As the business world becomes increasingly intricate, stakeholders require a swift means to understand a company's financial status. Text summarization is a useful tool in this regard, aiming to present long texts concisely without losing their essence. Given the complexity and the structured nature of financial documents, summarizing them poses a significant challenge. This paper suggests a method employing Positional Language Models (PLMs), a subset of non-neural language models that assess the sequence of tokens in input data, for financial report summarization. Our proposed method is unsupervised, eliminating the need for training and ensuring computational efficiency for lengthy documents.**

***Index Terms*—extractive summarization, positional language model, financial summarization, multilingual**

I. Introduction

Financial reports play a pivotal role in the decision-making processes of investors, stakeholders, and business executives. These reports often contain vast amounts of data, ranging from balance sheets and income statements to notes about company operations and future projections. Being able to generate concise, understandable summaries can greatly expedite the assessment process, allowing stakeholders to quickly grasp a company's financial health, performance, and potential risks. As the global business landscape becomes more complex, the ability to rapidly interpret financial data becomes crucial for timely and informed decision-making.

Text summarization refers to the process of distilling lengthy pieces of information, such as documents and articles, into shorter, coherent texts that retain the most salient and relevant points of the original content. There are three main types of summarization techniques: extractive, abstractive, and compressive. Extractive summarization involves selecting whole sentences or segments directly from the source material to construct the summary. In contrast, abstractive summarization goes a step further by paraphrasing and generating new sentences to convey the original content. Compressive summarization involves shortening the original sentences by removing redundant words or phrases while maintaining their fundamental meaning. Each of these methods has its advantages and challenges, and the choice often depends on the specific application and desired outcome. Surveys of modern summarization techniques can be found in [1]–[4].

Financial reports are intricate documents that meld quantitative data with qualitative narratives, often laden with industry-specific vocabulary and structured following strict regulatory guidelines that are different for every country. The main challenge in generating report summaries is to ensure that summaries capture not just the numerical facts but also the nuances and implications embedded in the narrative sections, like management discussions or notes. Furthermore, ensuring coherence and avoiding the loss of interconnected insights contained in these texts during summarization is crucial.

A series of Financial Narrative Processing (FNP) workshops dedicated to knowledge extraction from financial data has been held from 2018 till now [5]–[10]. Starting in 2020, these workshops include the financial narrative summarization (FNS) task, producing the first large-scale summarization results applied to financial data. Initially, the task focused on English annual reports produced by UK firms listed on the London Stock Exchange (LSE), but as of recently reports from other countries in additional languages–Spanish and Greek–have been added to the data.

Positional language models (PLMs) are a class of non-neural language models that are aware of the position or sequence of tokens in the input data; they were first introduced in [11] and were initially used for soft passage retrieval. The core concept of PLM is to create a distinct language model for every position within a document. Then, the document is evaluated based on the combined scores from its PLMs, rather than using a singular language model typically defined for the whole document. PLMs have demonstrated their value and cost-efficiency across various Natural Language Processing (NLP) domains, including information retrieval [12], [13] and language generation [14].

To tackle the challenging problem of financial report summarization, we propose a method that utilizes PLMs to score text chunks and then produces a coherent and quality summary from top-scored chunks. With the help of PLM, we produce scores for every word and every paragraph in a document, where the scores express the frequency behavior of words throughout the document. This method is unsupervised and thus does not require training; moreover, it is computationally efficient and thus especially suitable for long documents.

This paper is organized as follows. Section II describes background and related work. Section III describes the data. In Section IV we describe our method and summarization pipeline. Experimental evaluation is covered in Section V. Finally, Section VI concludes our work.

II. Related Work

Automatic text summarization is a challenging process, especially when it comes to summarizing long texts. Most recent datasets contain documents with an average length of 3,000 tokens [15], [16], and the majority of pre-trained Language Models (LMs) only support input with 512 to 1024 tokens [17], [18]. The main recent approaches to long text summarization include graph-based methods [19], [20], RNN-based approaches [21], [22], and transformer-based methods [23] are suitable for long texts (most notably, the models that employ Longformer [24].

Financial reports differ greatly from news pieces in at least four ways: length, structure, format, and language. A recent series of FNP and related workshops [5], [7]–[10] and a series of The Financial Narrative Summarisation (FNS) Shared Tasks [8], [9], [25] shows that there is growing interest in the application of automatic and computer-aided approaches for extracting, summarizing, and analyzing both qualitative and quantitative financial data. But before these workshops, there had been very few attempts to summarize financial reports [26], whereas the majority of works had concentrated on summarizing financial news [27]–[31].

Both traditional machine learning techniques (such as MNB, SVM, etc.) and deep neural networks (such as BILSTM, CNN, GRUs, etc.) were adopted by teams participating in the FNS tasks over the years. In particular, rule-based extraction methods were used in [32]–[35]; traditional machine learning methods were adopted in [33], [34], [36]–[38]; and high performing deep learning models, including Transformers and LMs, were the focus of works [23], [33]–[35], [39]–[44]. The text representation is also very diverse among the proposed approaches. Basic morphological and structure features were applied in [36], [45], syntactic features were used in [33], semantic vectors using word embeddings (including multilingual vectors) were applied in [36], [39], [44], and the hierarchical structure of reports was employed in [32], [42]. The vast majority of methods suggested for financial summarization are extractive and include a variety of ranking mechanisms. This is the natural option for lengthy papers because, when dealing with really large documents, global optimization procedures are much slower than ranking. The variety of methods include application of T-5 language model [46], BERT-based representation [47], [48], LSTM [48], neural node embeddings [48], unsupervised clustering [47], and extraction of word sequences extraction [49], hybrid extractive-abstractive summarization with pointer network generators [50].

One of the main challenges reported by the FNS [8], [9], [25] series participants was the length of annual reports (around 60,000 words). In addition, participants argued that extracting both text and structure from PDF files with numerous tables, charts, and numerical data resulted in noisy texts. Such feedback highlighted challenging components of the Financial Narrative Summarisation.

Trying to address this challenge, we focus on extractive summarization with a special emphasis on the computational efficiency that a PLM can supply.

III. The Data

Financial reports supplied by the shared task organizers come in three language-specific datasets – English, Greek, and Spanish. The datasets include financial report documents accompanied by carefully compiled human-generated summaries. The contents of these reports were obtained using Optical Character Recognition (OCR) technology extracted from the original PDF documents.

Annual reports published by UK companies listed on The London Stock Exchange (LSE) between 2002 and 2017 are included in the English dataset. [51], [52] The annual reports of firms between the years 2019 and 2020 are included in the Greek dataset. These reports, which were initially in PDF format, range in length from 100 to 300 pages. Compared to English reports, Greek reports are less organized. The distribution of the Spanish dataset, which is obtained from the FinT-esp corpus [53] is comparable to that of the Greek dataset. The reports' publication dates range from 2014 to 2018. The source is in PDF format and has a total of 40 to 400 pages.

The English dataset was randomly split by the task organizers into training (75%), testing, and validation (a total of 25%). For Greek and Spanish, the split was 60% for training and 40% for testing and validation. Gold summaries for the training and validation parts were available for the training, and gold summaries for the test parts remained hidden. In the English dataset, each report is accompanied by an average of three gold standard summaries, while the Greek and Spanish datasets contain two corresponding gold standard summaries for each report. The average length of gold summary and reports for the validation set of the FNS 2023 data is 968 and 55,389 words in English, 655 and 22,011 in Greek, and 547 and 26,021 in Spanish respectively. More detailed data statistics are given in Table I. For the FNS-2023 shared task the dataset provided contains a larger number of texts, with an increase in the number of texts in English and Greek in comparison to the data of FNS-2022 [8].

IV. The Method

A. The Pipeline

Our summarization pipeline contains three main steps depicted in Fig. 1. We start with text preprocessing which involves cleaning and structuring the text (described in Section IV-B. After preprocessing, the text is optionally fed into a BERT-based summarizer to produce large summaries (see detailed description in Section IV-C). Finally, the document or a generated summary undergoes paragraph scoring using a PLM (see Section IV-D). The PLM scores the paragraphs

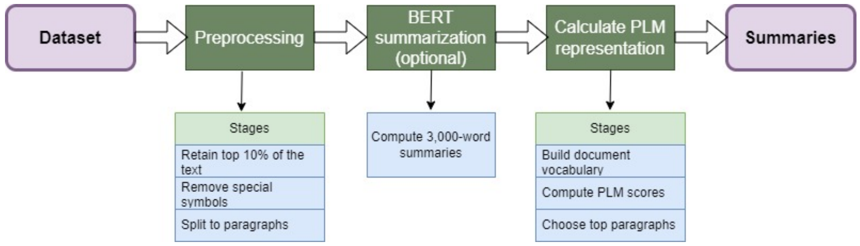

Fig. 1. The pipeline of our approach

TABLE I
DATA STATISTICS

		English		
data	train	validation	test	total
annual reports	3,050	413	550	4,013
gold summaries	10,007	1,383	1,804	13,194
gold per report	3.28	3.35	3.38	3.29
		Greek		
data	train	validation	test	total
annual reports	212	50	50	312
gold summaries	424	100	100	624
gold per report	2	2	2	2
		Spanish		
data	train	validation	test	total
annual reports	162	50	50	262
gold summaries	324	100	100	524
gold per report	2	2	2	2

based on their coherence, enabling the final selection or reordering of paragraphs to ensure the summary is not just concise, but also contextually accurate and fluent.

B. Preprocessing

The first step in data preprocessing involves retaining the first 10% of text from each report. This approach has been proven successful in previous shared tasks by several authors [47], [54]. To verify this approach, we have computed Rouge [55] scores for various percentages of the text for the FNS 2023 validation set (see Table II). From this table, we can see that the scores are more plausible when the first 10% of the text is retained. The next data preprocessing step includes cleaning the text of extraneous characters and debris and keeping only words, numbers, punctuation, and end-of-line (EOL) characters. For metric computation and the identification of critical sections within the text, we have decided to operate on a paragraph level. Therefore, the third stage of text preprocessing includes segmenting the text into paragraphs based on the EOL characters.

TABLE II
PARTIAL DOCUMENT RETENTION

text percentage	Rouge 1-F	Rouge 2-F	Rouge L-F
5%	0.3754	0.1907	0.3516
10%	0.3922	0.2207	0.3710
15%	0.3544	0.1648	0.3290
100%	0.2480	0.0700	0.2195

C. BERT Extractive Summarization

BERTSUM [56] is a tool for extractive text summarization based on the Bidirectional Encoder Representations from Transformers (BERT). This method selects significant sentences from the source text to create a coherent summary by utilizing a pre-trained BERT model. The summarizer ranks sentences by their importance to generate concise and informative summaries, making it valuable for summarizing long documents or articles. We used a publically available implementation of BERTSUM called ”bert-extractive-summarizer” at https://pypi.org/project/bert-extractive-summarizer, described in [57]. By default, it utilizes the ’bert-base-uncased’ pre-trained model [17] for its operations.

We set the length of summaries for BERTSUM to 3,000 words (this length was chosen empirically), whereas the final target summaries have a length of 1,000 words. When used with the FNS-2020 dataset, BERT summarization was proven to be beneficial in achieving higher metric scores, whereas, on the FNS-2023 data, the outcomes are less straightforward, as shown in Table III.

TABLE III
BERT SUMMARIZATION SCORES FOR THE FNS 2020 AND 2023 ENGLISH VALIDATION SETS

year	metric	R	P	F
2020	Rouge-1	0.4545	0.4096	0.3941
2020	Rouge-2	0.2868	0.2376	0.2325
2020	Rouge-L	0.4367	0.3909	0.3774
2023	Rouge-1	0.4952	0.4190	0.4132
2023	Rouge-2	0.3370	0.2695	0.2651
2023	Rouge-L	0.4774	0.4013	0.3970

D. PLM Scoring

The key idea of PLM is to define a language model for each position of a document [11], and then score a document or its part based on the scores of their PLMs. The fundamental concept underpinning PLMs lies in the ability to compute a score for each vocabulary element at any given position within a document. This score quantifies the element’s relevance within that specific position, taking into account its proximity to other occurrences of the same element across the entire document. Essentially, the closer an element is to the position under consideration, the higher its corresponding score. This behavior allows the model to capture the importance of elements within the broader context of the entire text.

Let D be a document and i be a position within this document. The document is considered to be an ordered sequence of elements, which in our case are words (tokens). Formally, a value of PLM P for word w at position i in D is computed as

$$P(w|i) = \frac{\sum_{j=1}^{|D|} c(w,j) \times f(i,j)}{\sum_{w' \in V} \sum_{j=1}^{|D|} c(w',j) \times f(i,j)}$$

where $c(w,j) \in \{0,1\}$ denotes the presence of a term w at position j, $|D|$ denotes the length of D (the number of terms in it), V is the document's vocabulary, and $f(i,j)$ is the propagation function that measures the distance between i and j. We used a Triangle kernel as a propagation function $f(i,j)$ (see (1)). We chose the σ value of 25, following the work of [14].

$$f(i,j) \begin{cases} 1 - \frac{i-j}{\sigma} & \text{if } |i-j| \leq \sigma \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

Fig. 2 shows how the triangle kernel behaves for different values of σ.

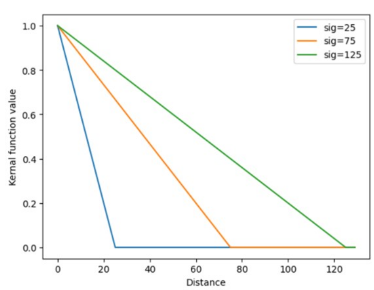

Fig. 2. Charts of $P(w)$ for words that appear once in the document (left) and words that appear more than once (right). The X axis denotes the position of a word in the document.

Thus, P is a function defined for every word w in the document. The function's domain is word positions; it behaves differently for terms that appear only once in a document and repeated terms. Fig. 3 illustrates these functions. To utilize

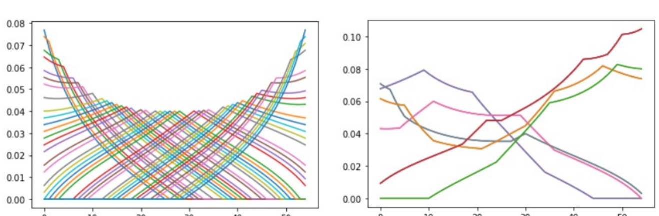

Fig. 3. Charts of $P(w)$ for words that appear once in the document (left) and words that appear more than once (right). The X axis denotes the position of a word in the document.

the PLM method for financial summarization, it is necessary to calculate the sum of products of word matrices and the propagation function $f(i,j)$. We have created the following matrices:

- propagation matrix PM, where the propagation function values between every two positions in the text are stored;
- text matrix TM containing all words present in the processed text and their respective indices within the text.

Matrix multiplication $PM \times TM$ yielded a matrix of words and PLM values $P(w)$ for all words w in a document. For each paragraph p of a document, its PLM value $PLM(p)$ was calculated as the sum of PLM values for each word divided by the word count in the paragraph:

$$PLM(p) = \frac{\sum_{w \in p} P(w)}{|p|}$$

Finally, we added the paragraphs with the highest PLM score to the summary, such that in total it contains no more than 1000 words. The paragraph order in summaries corresponds to their positions in the original texts. The pipeline of this approach is depicted in Fig. 4. For efficient computation, we

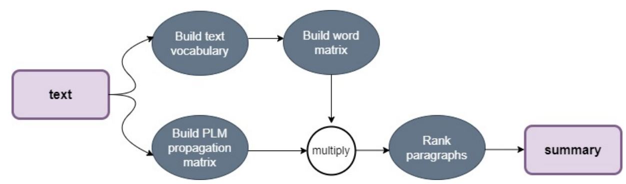

Fig. 4. Summarization with PLM

used SciPY sparse matrices [58].

V. Experimental Evaluation

A. Software and Hardware Setup

All experiments were conducted on a standalone desktop with Intel(R) Core(TM) i7-1065G7 CPU and 16GB DRAM. The system was implemented in Python 3.11.6. We used Python Rouge implementation of [59], and applied NLTK tokenization [60] to split the texts into words.

B. Models and Baselines

Our baselines include TopK which takes the first K words in every document (with $K = 1,000$) and BERT summarization we applied to the partial documents retained after preprocessing (described in Section IV-B). We tested two of our models – a pure PLM summarization that does not include a summarization step (denoted PLMsumm), and a PLM summarization applied to BERT summaries with 3,000 words (denoted PLMsumm+BERT).

C. Results

We can see that on the FNS 2020 English data, BERT summarization is beneficial in achieving superior metric scores, whereas, on the FNS 2023 data, the outcomes are less straightforward. In Greek and Spanish datasets, a simple approach with PLM applied on 10% of a text outperforms PLM with BERT selection.

TABLE IV
SCORES OF PLM MODELS AND BASELINES ON FNS 2020 AND 2023 VALIDATION SETS

year	model	Rouge 1-F	Rouge 2-F	Rouge L-F
		English		
2020	PLMsumm	0.3621	0.1975	0.3438
2020	BERT+PLMsumm	**0.4369**	**0.2848**	**0.4249**
2023	PLMsumm	0.3716	0.1969	0.3506
2023	BERT+PLMsumm	0.3761	0.1911	0.3523
2023	TopK	0.1644	0.0294	0.1398
2023	BERTsumm	**0.3771**	**0.2127**	**0.3584**
		Greek		
2023	PLMsumm	**0.1262**	**0.0350**	**0.1173**
2023	BERT+PLMsumm	0.1178	0.0289	0.1093
2023	TopK	0.0820	0.0107	0.0783
2023	BERTsumm	0.1172	0.0291	0.1096
		Spanish		
2023	PLMsumm	**0.2280**	**0.1026**	**0.2006**
2023	BERT+PLMsumm	0.2099	0.0898	0.1865
2023	TopK	0.1743	0.0378	0.1461
2023	BERTsumm	0.2167	0.0894	0.1934

Table IV also contains the Rouge scores of baselines applied to validation sets of the FNS 2023 dataset. Note that the scores of BERT extractive summarization (denoted BERTsumm) are significantly higher for the English data because English summaries are extracts of the original documents, while Greek and Spanish summaries are abstractive summaries.

One prominent advantage of our method is its computational efficiency. The heaviest part of our method is BERT summarization (if used), while PLM computing is very fast, taking less than 2 seconds on average per document. We report the average runtimes of preprocessing and PLM computing in Table V for the PLMsumm model that processes the first 10% of texts (as described in Section IV-B, and for the BERT+PLMsumm that uses BERT summarization as its initial step. For the latter model, we also report the time spent on BERT summarization.

TABLE V
AVERAGE PER-DOCUMENT RUNTIMES OF OUR MODEL ON THE FNS 2023 VALIDATION SETS

text size	preprocessing	PLM computing	BERT summarization
		English	
first 10%	0.0050s	1.2195s	N/A
3,000 words	0.0002s	0.0270s	100s
		Greek	
first 10%	0.0020s	0.3600s	N/A
3,000 words	0.0010s	0.0159s	78s
		Spanish	
first 10%	0.0030s	0.3000s	N/A
3,000 words	0.0010s	0.0164s	50.4s

VI. CONCLUSIONS AND FUTURE WORK

This paper introduces a novel approach to summarizing long and noisy financial reports. The extractive summarization is performed in two stages: filtering out the noisy part of a document with two alternative methods and then applying the Positional Language Model to the selected part. The main advantage of the proposed method is its time efficiency and that it does not require training data. The evaluation of the proposed method shows that a simple unsupervised approach with PLM applied on the first 10% of a document slightly outperforms a supervised BERT summarizer on Greek and Spanish reports despite the proposed method having a very significant supremacy in time complexity.

In our future research, we aim to investigate the impact of modifications in PLM configurations and parameters on summarization outcomes. Specifically, we plan to explore how using training data can serve as a means to fine-tune the parameter σ. Additionally, we seek to determine whether the consideration of word proximity, as opposed to allowing greater distances between words, yields superior results. Another aspect of our investigation involves assessing whether the utilization of more advanced pre-trained summarization models, such as GPT [61] or T5 [62], can enhance the overall quality of summarization. Finally, we intend to investigate whether or not changing our method's granularity from paragraphs to sentences will improve our results.

REFERENCES

[1] H. Y. Koh, J. Ju, M. Liu, and S. Pan, "An empirical survey on long document summarization: Datasets, models, and metrics," *ACM computing surveys*, vol. 55, no. 8, pp. 1–35, 2022.

[2] J. Wang, F. Meng, D. Zheng, Y. Liang, Z. Li, J. Qu, and J. Zhou, "A survey on cross-lingual summarization," *Transactions of the Association for Computational Linguistics*, vol. 10, pp. 1304–1323, 2022.

[3] C. Ma, W. E. Zhang, M. Guo, H. Wang, and Q. Z. Sheng, "Multi-document summarization via deep learning techniques: A survey," *ACM Computing Surveys*, vol. 55, no. 5, pp. 1–37, 2022.

[4] A. Alomari, N. Idris, A. Q. M. Sabri, and I. Alsmadi, "Deep reinforcement and transfer learning for abstractive text summarization: A review," *Computer Speech & Language*, vol. 71, p. 101276, 2022.

[5] M. El-Haj, P. Rayson, and A. Moore, *The first financial narrative processing workshop (FNP 2018)*, 2018.

[6] R. Juge, I. Bentabet, and S. Ferradans, "The fintoc-2019 shared task: Financial document structure extraction," in *Proceedings of the Second Financial Narrative Processing Workshop (FNP 2019)*, 2019, pp. 51–57.

[7] M. El-Haj, "Multiling 2019: Financial narrative summarisation," in *Proceedings of the Workshop MultiLing 2019: Summarization Across Languages, Genres and Sources*, 2019, pp. 6–10.

[8] M. El-Haj, M. Litvak, N. Pittaras, G. Giannakopoulos *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*, 2020, pp. 1–12.

[9] N. Zmandar, M. El-Haj, P. Rayson, M. Litvak, G. Giannakopoulos, N. Pittaras *et al.*, "The financial narrative summarisation shared task fns 2021," in *Proceedings of the 3rd Financial Narrative Processing Workshop*, 2021, pp. 120–125.

[10] M. El-Haj, P. Rayson, and N. Zmandar, "Proceedings of the 4th financial narrative processing workshop@ lrec2022," in *Proceedings of the 4th Financial Narrative Processing Workshop@ LREC2022*, 2022.

[11] Y. Lv and C. Zhai, "Positional language models for information retrieval," in *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, 2009, pp. 299–306.

[12] F. Boudin, J.-Y. Nie, and M. Dawes, "Positional language models for clinical information retrieval," in *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, 2010, pp. 108–115.

[13] L. Wang, C. Huang, M. Wang, Z. Xue, and Y. Wang, "Neuropred-plm: an interpretable and robust model for neuropeptide prediction by protein language model," *Briefings in Bioinformatics*, vol. 24, no. 2, p. bbad077, 2023.

[14] M. Vicente, C. Barros, and E. Lloret, "Statistical language modelling for automatic story generation," *Journal of Intelligent & Fuzzy Systems*, vol. 34, no. 5, pp. 3069–3079, 2018.
[15] P. Manakul and M. J. Gales, "Long-span summarization via local attention and content selection," *arXiv preprint arXiv:2105.03801*, 2021.
[16] M. Zaheer, G. Guruganesh, K. A. Dubey, J. Ainslie, C. Alberti, S. Ontanon, P. Pham, A. Ravula, Q. Wang, L. Yang *et al.*, "Big bird: Transformers for longer sequences," *Advances in neural information processing systems*, vol. 33, pp. 17 283–17 297, 2020.
[17] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "BERT: pre-training of deep bidirectional transformers for language understanding," *CoRR*, vol. abs/1810.04805, 2018. [Online]. Available: http://arxiv.org/abs/1810.04805
[18] J. Zhang, Y. Zhao, M. Saleh, and P. J. Liu, "Pegasus: pre-training with extracted gap-sentences for abstractive summarization (2019)," *arXiv preprint ArXiv:1912.08777*, 1912.
[19] X. Liang, S. Wu, M. Li, and Z. Li, "Improving unsupervised extractive summarization with facet-aware modeling," in *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, 2021, pp. 1685–1697.
[20] Y. Dong, A. Mircea, and J. C. Cheung, "Discourse-aware unsupervised summarization of long scientific documents," *arXiv preprint arXiv:2005.00513*, 2020.
[21] W. Xiao and G. Carenini, "Extractive summarization of long documents by combining global and local context," *arXiv preprint arXiv:1909.08089*, 2019.
[22] J. Pilault, R. Li, S. Subramanian, and C. Pal, "On extractive and abstractive neural document summarization with transformer language models," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2020, pp. 9308–9319.
[23] U. Khanna, S. Ghodratnama, A. Beheshti *et al.*, "Transformer-based models for long document summarisation in financial domain," in *Proceedings of the 4th Financial Narrative Processing Workshop@ LREC2022*, 2022, pp. 73–78.
[24] I. Beltagy, M. E. Peters, and A. Cohan, "Longformer: The long-document transformer," *arXiv preprint arXiv:2004.05150*, 2020.
[25] M. El-Haj, N. Zmandar, A. A. Paul Rayson, M. Litvak, N. Pittaras, G. Giannakopoulos, A. Kosmopoulos, B. Carbajo-Coronado, and A. Moreno-Sandoval, "The financial narrative summarisation shared task (fns 2022)," in *Proceedings of the 4th Financial Narrative Processing WorkshopLREC 2022*, 2022, pp. 43–52.
[26] M. Isonuma, T. Fujino, J. Mori, Y. Matsuo, and I. Sakata, "Extractive summarization using multi-task learning with document classification," in *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 2017, pp. 2101–2110.
[27] K. Filippova, M. Surdeanu, M. Ciaramita, and H. Zaragoza, "Company-oriented extractive summarization of financial news," in *Proceedings of the 12th Conference of the European Chapter of the ACL (EACL 2009)*, 2009, pp. 246–254.
[28] C. C. Yang and F. L. Wang, "Automatic summarization for financial news delivery on mobile devices." in *WWW (Posters)*, 2003.
[29] P. C. F. de Oliveira, K. Ahmad, and L. Gillam, "A financial news summarization system based on lexical cohesion," in *Proceedings of the International Conference on Terminology and Knowledge Engineering, Nancy, France*, 2002.
[30] E. Baralis, L. Cagliero, and T. Cerquitelli, "Supporting stock trading in multiple foreign markets: a multilingual news summarization approach," in *Proceedings of the Second International Workshop on Data Science for Macro-Modeling*, 2016, pp. 1–6.
[31] Y. Zhang, E. Chen, and W. Xiao, "Extractive-abstractive summarization with pointer and coverage mechanism," in *Proceedings of 2018 International Conference on Big Data Technologies*, 2018, pp. 69–74.
[32] M. Litvak, N. Vanetik, and Z. Puchinsky, "Sce-summary at the fns 2020 shared task," in *Proceedings of the 1st Joint Workshop on Financial Narrative Processing and MultiLing Financial Summarisation*, 2020, pp. 124–129.
[33] A. Vhatkar, P. Bhattacharyya, and K. Arya, "Knowledge graph and deep neural network for extractive text summarization by utilizing triples," in *Proceedings of the 1st Joint Workshop on Financial Narrative Processing and MultiLing Financial Summarisation*, 2020, pp. 130–136.
[34] P. Arora and P. Radhakrishnan, "Amex ai-labs: An investigative study on extractive summarization of financial documents," in *Proceedings of the 1st Joint Workshop on Financial Narrative Processing and MultiLing Financial Summarisation*, 2020, pp. 137–142.
[35] A. A. Azzi and J. Kang, "Extractive summarization system for annual reports," in *Proceedings of the 1st Joint Workshop on Financial Narrative Processing and MultiLing Financial Summarisation*, 2020, pp. 143–147.
[36] J. B. Suarez, P. Martínez, and J. L. Martínez, "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*, 2020, pp. 112–117.
[37] M. El-Haj and A. Ogden, "Financial narrative summarisation using a hybrid tf-idf and clustering summariser: AO-Lancs system at FNS 2022," in *Proceedings of the 4th Financial Narrative Processing Workshop@ LREC2022*, 2022, pp. 79–82.
[38] N. Shukla, A. Vaid, R. Katikeri, S. Keeriyadath, and M. Raja, "Dimsum: Distributed and multilingual summarization of financial narratives," in *Proceedings of the 4th Financial Narrative Processing Workshop@ LREC2022*, 2022, pp. 65–72.
[39] R. Agarwal, I. Verma, and N. Chatterjee, "Langresearchlab_nc at fincausal 2020, task 1: A knowledge induced neural net for causality detection," in *Proceedings of the 1st Joint Workshop on Financial Narrative Processing and MultiLing Financial Summarisation*, 2020, pp. 33–39.
[40] A. Singh, "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*, 2020, pp. 105–111.
[41] M. La Quatra and L. Cagliero, "End-to-end training for financial report summarization," in *Proceedings of the 1st Joint Workshop on Financial Narrative Processing and MultiLing Financial Summarisation*, 2020, pp. 118–123.
[42] S. Zheng, A. Lu, and C. Cardie, "Sumsum@ fns-2020 shared task," in *Proceedings of the 1st Joint Workshop on Financial Narrative Processing and MultiLing Financial Summarisation*, 2020, pp. 148–152.
[43] M. Pant and A. Chopra, "Multilingual financial documentation summarization by team_tredence for fns2022," in *Proceedings of the 4th Financial Narrative Processing Workshop@ LREC2022*, 2022, pp. 112–115.
[44] N. Foroutan, A. Romanou, S. Massonnet, R. Lebret, and K. Aberer, "Multilingual text summarization on financial documents," in *Proceedings of the 4th Financial Narrative Processing Workshop@ LREC2022*, 2022, pp. 53–58.
[45] L. Li, Y. Jiang, and Y. Liu, "Extractive financial narrative summarisation based on dpps," in *Proceedings of the 1st Joint Workshop on Financial Narrative Processing and MultiLing Financial Summarisation*, 2020, pp. 100–104.
[46] M. Orzhenovskii, "T5-long-extract at fns-2021 shared task," in *Proceedings of the 3rd Financial Narrative Processing Workshop*, 2021, pp. 67–69.
[47] T. Gokhan, P. Smith, and M. Lee, "Extractive financial narrative summarisation using sentencebert based clustering," in *Proceedings of the 3rd Financial Narrative Processing Workshop*, 2021, pp. 94–98.
[48] M. Litvak and N. Vanetik, "Summarization of financial reports with AMUSE," in *Proceedings of the 3rd Financial Narrative Processing Workshop*, 2021, pp. 31–36.
[49] S. Krimberg, N. Vanetik, and M. Litvak, "Summarization of financial documents with tf-idf weighting of multi-word terms," in *Proceedings of the 3rd Financial Narrative Processing Workshop*, 2021, pp. 75–80.
[50] N. Zmandar, A. Singh, M. El-Haj, and P. Rayson, "Joint abstractive and extractive method for long financial document summarization," in *Proceedings of the 3rd Financial Narrative Processing Workshop*, 2021, pp. 99–105.
[51] M. El-Haj, P. Rayson, S. Young, and M. Walker, "Detecting document structure in a very large corpus of uk financial reports," 2014.
[52] M. El-Haj, P. Rayson, P. Alves, C. Herrero-Zorita, and S. Young, "Multilingual financial narrative processing: Analyzing annual reports in english, spanish, and portuguese," in *Multilingual Text Analysis: Challenges, Models, And Approaches*. World Scientific, 2019, pp. 441–463.
[53] A. Moreno-Sandoval, A. Gisbert, and H. Montoro, "Fint-esp: A corpus of financial reports in spanish," *Fuster, et al., editors, Multiperspectives in analysis and corpus design*, pp. 89–102, 2020.
[54] N. Vanetik, M. Litvak, and S. Krimberg, "Summarization of financial reports with tiber," *Machine Learning with Applications*, vol. 9, p. 100324, 2022.

[55] C.-Y. Lin, "Rouge: A package for automatic evaluation of summaries," in *Text summarization branches out*, 2004, pp. 74–81.
[56] Y. Liu, "Fine-tune bert for extractive summarization," *arXiv preprint arXiv:1903.10318*, 2019.
[57] D. Miller, "Leveraging bert for extractive text summarization on lectures," *arXiv preprint arXiv:1906.04165*, 2019.
[58] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. J. Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. Carey, İ. Polat, Y. Feng, E. W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, P. van Mulbregt, and S. . Contributors, "SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python," *Nature Methods*, vol. 17, pp. 261–272, 2020.
[59] pltrdy, "ROUGE metrics implementation in python," 2023, [Online; accessed 18-October-2023]. [Online]. Available: https://github.com/pltrdy/rouge
[60] S. Bird, E. Klein, and E. Loper, "NLTK: Natural language toolkit," 2021, [Online; accessed 18-October-2023]. [Online]. Available: https://www.nltk.org/
[61] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, "Improving language understanding by generative pre-training," *arXiv preprint arXiv:1803.08493*, 2018.
[62] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu, "Exploring the limits of transfer learning with a unified text-to-text transformer," *arXiv preprint arXiv:1910.10683*, 2019.