SUMSUM@FNS-2020 Shared Task

Siyan Zheng, Anneliese Lu, Claire Cardie
Department of Computer Science, Cornell University, Ithaca, NY 14853
{sz488, yl668}@cornell.edu
cardie@cs.cornell.edu

Abstract

This paper describes the SUMSUM systems submitted to the Financial Narrative Summarization Shared Task (FNS-2020). We explore a section-based extractive summarization method tailored to the structure of financial reports: our best system parses the report Table of Contents (ToC), splits the report into narrative sections based on the ToC, and applies a BERT-based classifier to each section to determine whether it should be included in the summary. Our best system ranks 4th, 1st, 2nd and 17th on the Rouge-1, Rouge-2, Rouge-SU4, and Rouge-L official metrics, respectively. We also report results on the validation set using an alternative set of Rouge-based metrics that measure performance with respect to the best-matching of the available gold summaries.

1 Introduction

The number of financial reports available each year is increasing rapidly. Consequently, exhaustive reading of such documents has become unreasonably laborious. Automatic summarization methods could greatly simplify this task. The Financial Narrative Summarization Shared Task for 2020 (FNS-2020) aims to study the application of automatic summarization methods to annual reports from UK firms listed on The London Stock Exchange (LSE) (El-Haj et al., 2020). These reports, compared to those written by U.S. firms, exhibit a much less rigid structure, which makes summarization a challenging task.

In recent years, recurrent neural networks (RNNs) (e.g. (Nallapati et al., 2016)) and transformer-based neural networks (e.g. BERT (Devlin et al., 2019)) have been widely and successfully employed for extractive summarization in numerous text genres. We had hoped to employ such models for the FNS-2020 shared task. Unfortunately, the length of documents is beyond the models' limits: with an average training document length of 6086 tokens, RNNs and transformers struggle to encode useful hidden representations in an end-to-end fashion. As a result, we hypothesized that some pre-selection over the input text is mandatory to obtain reasonable performance. In particular, we determined (see Section 3) that *most sentences in the gold-standard report summaries come from a single narrative section of the report*. Therefore, we ultimately built our system based on a classification-based extractive summarization approach that uses BERT-based models (Devlin et al., 2019) simply to determine the section(s) to extract as the summary subject to truncation constraints (as described in Section 3).

2 Data Preprocessing

Data Type	Training Set	Validation Set	Testing Set	Total
full reports	3000	363	500	3863
gold summaries	9873	1250	1673	12796

Table 1: Dataset statistics. (El-Haj et al., 2020)

As shown in Table 1, on average there are no fewer than 3 gold-standard summaries provided for each annual report, with some reports containing up to 7 gold-standard summaries (El-Haj et al., 2020). In

Proceedings of the 1st Joint Workshop on Financial Narrative Processing and MultiLing Financial Summarisation, pages 148–152
Barcelona, Spain (Online), December 12, 2020.

addition, summaries in the training set are highly extractive: there is a >99% unigram overlap between each summary and corresponding report. That is to say, except for an extremely small number of gold summaries, all other summaries are comprised of phrases and sentences of its respective full report — even given the spelling and formatting errors that arise in the PDF to plain text conversion. As a result, our preprocessing of the data is relatively straightforward: other than a few recurring encoding errors, we focused primarily on making sure that sentences remain intact to allow high-quality matching of gold-standard summary sentences to a candidate summary section during training. For example, we remove blank spaces within a sentence; and concatenate lines of the same sentence that might be mistakenly split due to punctuation or lack of punctuation.

3 System Description

Our approach to FNS summarization requires first that we split the document into coherent narrative sections (Section 3.2), which is accomplished by the heuristic parsing of the Table of Contents (ToC) (Section 3.1). We then train neural network models (Section 3.3) to identify which section(s) of a report to extract to comprise the summary.

3.1 Parsing the Table of Contents

Through observation, most ToCs contain titles and page numbers for each listed section of the report. However, some page numbers are left aligned while the others are right aligned. Some ToCs include both the start and the end page numbers for each section; others contain only the start page numbers. Our ToC parser attempts to consider all possible formatting situations with the goal of identifying a title and start page for each section. We clean the data by deleting the end page numbers from all ToCs. Special characters such as dots which are ornaments between page numbers and titles are also deleted. We search for continuous numbers and string combinations only if there exists the same alignment within one ToC, The search starts from the head of a document as we assume ToCs appear in the first few pages. Some exceptions are allowed since a few ToCs have subtitles.

3.2 Split Narrative Section

Because the given data is partially damaged when converted from PDF format to TXT format — even sometimes including the loss of page numbers and section titles — we experimented with both title-based and page-number-based heuristics to split each report into narrative sections according to the TOC. *Title-based search* is the process of scanning the report for titles in the order extracted from the ToC; we assume that these delimit one section from the next. Some spelling error tolerance is employed. If a title is not found, it is temporarily skipped. For *page-based search*, we look for the presence of a single, isolated number associated with the start of each section, and assume that these delimit the sections.

For a given report, we estimate which of the above approaches is more accurately identifying the sections. We consider both the number of sections found and the number of lines on each page. For example, we assume that the number of lines on each page should be similar; that identification of earlier sections from the TOC is important (since sentences extracted for the summary usually appear in the first few sections of the report); and reject the section that has an extreme number of lines to a number of page proportions. We prioritize the title-search method as it returns better results. If some sections cannot be found, we split the number of unassigned lines evenly into those missing sections bounded by neighboring found sections.

3.3 Models

SUMSUM-BASE: the Baseline Model

Our baseline summarization model extracts as the summary the concatenation of the first and second sections (referred as Section 0 and 1) from the full report, truncating extracted text to 1000 tokens.

SUMSUM-BERT

Our BERT-based summarization model is trained to make a binary classification for a given section — INCLUDE or DO NOT INCLUDE it in the summary. To do this, we require a training set that pairs

individual sections of a report with a positive (include) or negative (don't include) label depending on whether or not the gold-standard summary is derived from it. Given the existing FNS training data, there are a number of options for doing this, none of which is guaranteed to produce noise-free labels. But examination of the training and validation sets indicated that 2761 out of 3000 (training), and 351 out of 363 (validation) financial reports have more than $\frac{2}{3}$ of their summary sentences originating from the same section. As a result, we employ a sentence-overlap method to identify a single section as the "summary section" section, and label all other sections as negative. Specifically, we regard one sentence as overlapping a report section if most of its words appear in that section. When there are multiple gold-standard summaries for a financial report, we choose the summary with the highest sentence overlap rate as the gold standard. Finally, to prevent extremely unbalanced data, we only consider the first five sections, as the gold-standard summary for 2893 of out 3000 number of financial reports in the training set overlaps with text from the first five sections.

We considered the following BERT-based models:

Model 1: Truncate sections to 512 tokens. Apply BERT to get the embedding for the section input, and add a linear layer to produce a Boolean output[1] .

Model 2: Truncate each sentence to 100 tokens. Apply BERT[1] to obtain sentence embeddings, take the average of all sentence embeddings, and add a linear layer to produce a Boolean output.

Model 3: Apply Hierarchical BERT[2] (Zhang et al., 2019) as above with a maximum of 50 words per sentence and a maximum of 100 sentences.

Model	Precision	Recall	F1
Model 1	0.869	0.869	0.869
Model 2	0.847	0.847	0.847
Model 3	0.855	0.855	0.855

Table 2: Selecting sections using Bert models.

As shown in Table 2, the first model performs best at selecting summary sections (on the validation set). Therefore, SUMSUM-BERT uses sections predicted by Model 1 as the summary, truncating it to 1000 words. If more than one section is selected, their truncated versions are concatenated to produce the summary.

SUMSUM-01: BERT-based Model with Section 0 & 1 as Candidates:

These models use Section 0, Section 1, and both Section 0 & 1 as candidate summary sections for classification. For training, we labeled the one with the highest Rouge-2 F-1 score as the gold summary, and truncate each input to 512 tokens. (For input concatenated from section 0 & 1, we limit each section to 256 tokens.) At test time, we apply BERT with a maximum input length of 512 tokens to determine section(s) selected for summary.

4 Results

For the evaluation, we employed the three systems described above. We show our results using two different evaluation metrics: the *official* metrics, which take the average of the Rouge scores of the produced summary against all available gold summaries; and *alternative* metrics, which use the Rouge scores of the single gold summary that best matches the system-generated summary. The latter is the metric proposed in the FNS-2020 workshop call, and the metric that we optimized for when training the SUMSUM models.

4.1 Official Results: Average

The official results of the SUMSUM systems are given in Table 3. The rows in light gray are the official baselines, provided for comparison. The highest scores among the SUMSUM systems and the given baselines are in bold. We see that SUMSUM-BASE and SUMSUM-BERT perform similarly, and

[1]Scripts available at `https://github.com/castorini/hedwig/tree/master/models/bert`.

[2]Scripts available at `https://github.com/castorini/hedwig/tree/master/models/hbert`.

Model	Rouge-1			Rouge-2			Rouge-SU4			Rouge-L		
	F	P	R	F	P	R	F	P	R	F	P	R
SUMSUM-BASE	**.462**	.481	**.494**	.294	.259	**.398**	.288	.236	**.442**	.324	.350	.332
SUMSUM-BERT	.460	**.530**	.450	**.306**	**.295**	.365	**.302**	**.268**	.406	.322	**.389**	.304
SUMSUM-01	.442	.511	.447	.286	.277	.358	.282	.253	.398	.313	.375	.304
SUMM-TL-MUSE	.433	.413	.483	.234	.198	.311	.253	.201	.375	**.407**	.370	**.470**
LEXRANK-SUMMARY	.264	.269	.337	.120	.107	.193	.140	.117	.253	.218	.263	.210
TEXTRANK-SUMMARY	.172	.118	.414	.070	.044	.229	.079	.048	.302	.206	.197	.235
SUMM-BL-POLY	.274	.253	.324	.105	.088	.147	.135	.105	.213	.205	.177	.260

Table 3: Official scores: averages scores over all gold summaries corresponding to each annual report.

outperform the official baselines when measured by Rouge-1, Rouge-2 and Rouge-SU4. The official baseline SUMM-TL-MUSE beats the SUMSUM systems when measured by Rouge-L due to its much higher recall.

4.2 Alternative Results: Best Match

Model	Rouge-1			Rouge-2			Rouge-SU4			Rouge-L		
	F	P	R	F	P	R	F	P	R	F	P	R
SUMSUM-BASE	.674	.647	**.823**	.630	.562	**.778**	.635	.599	**.783**	.670	.684	.738
SUMSUM-BERT	**.727**	**.750**	.794	**.681**	**.691**	.748	**.686**	**.699**	.752	**.698**	.704	**.754**
SUMSUM-01	.691	.723	.775	.647	.662	.731	.652	.672	.736	.661	**.735**	.676

Table 4: Alternative metric scores: uses the highest-scoring gold summary corresponding to each annual report. Scores are computed for the FNS-2020 validation set.

Results on the validation set according to the alternative metric are shown in Table 4. With this metric, all scores are substantially higher, and SUMSUM-BERT is the best system among the three models.

5 Discussion of Results

As mentioned above, we relied on the alternative evaluation metric (that takes the highest score among all gold summaries of one financial report) for model selection. And as shown in Table 4, the SUMSUM-BERT system, which uses BERT to select the summary sections from among the first five sections as candidates for the summary, returns the best results under this metric by far. It reaches a 86.9% F1 score in selecting the correct summary section. Occasionally, the system does not return the correct section, but the alternative section chosen still seems to perform well. Presumably, this is the case because there are multiple summaries corresponding to one financial report.

However, when measured according to the official metrics, the results of the BASE and BERT model are quite close to one another (Table 3). We believe this is because the different gold summaries for each report have varying characteristics. Most of the time, there is one gold summary that is primarily constructed of extracted snippets from one section and another summary that is constructed of sentences taken from a few sections. This causes SUMSUM-BERT to outperform the other models when using the alternative metrics that use the gold summary with the highest Rouge score. Finally, almost all gold summaries are comprised of some sentences extracted from the first two sections. This makes the BASE method perform well according to the average-based official metrics.

As described in the Introduction, we initially hoped to employ neural network-based summarization methods directly for the FNS-2020 Shared Task. In particular, we experimented with neural network NLP models such as PreSumm (Liu and Lapata, 2019) and Transformer-XL (Dai et al., 2019), using them to generate summaries given the chosen sections as input (instead of outputting the first 1000 tokens). All, however, led to worse results. Specifically, the models performed poorly on Recall though the Precision scores were close to those of our submitted SUMSUM-BERT systems: the methods failed to select enough sentences compared to the gold summaries. A future direction of work could be improving sentence extraction from sections of different report components.

6 Conclusion

In this paper, we describe the SUMSUM systems for the Financial Narrative Summarisation Shared Task (FNS-2020.) Our best system parses the report Table of Contents (ToC), splits the report into narrative

sections based on the ToC, and applies BERT to each section to determine which section(s) to include for the summary. The best system achieves F1 scores of 72.66%, 68.12%, 68.58%, 69.75% with Rouge-1, Rouge-2, Rouge-SU4, and Rouge-L on the validation set using our alternative evaluation metric that the systems were optimized for. The F1 scores are 46.0%, 30.6%, 30.2%, 32.2% according to the official metrics.

Acknowledgments. We thank the reviewers for their comments and suggestions; and thank the organizers for their work to create this Shared Task.

References

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

Mahmoud El-Haj, Ahmed AbuRa'ed, Nikiforos Pittaras, and George Giannakopoulos. 2020. The Financial Narrative Summarisation Shared Task (FNS 2020). In *The 1st Joint Workshop on Financial Narrative Processing and MultiLing Financial Summarisation (FNP-FNS 2020)*, Barcelona, Spain.

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, November. Association for Computational Linguistics.

Ramesh Nallapati, Bowen Zhou, Cicero dos Santos, Çağlar GuÌ‡lçehre, 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, August. Association for Computational Linguistics.

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