Beyond Pure Text: Summarizing Financial Reports Based on Both Textual and Tabular Data

Ziao Wang[1], **Zelin Jiang**[1], **Xiaofeng Zhang**[1*], **Jaehyeon Soon**[1], **Jialu Zhang**[2], **Wang Xiaoyao**[1] and **Hongwei Du**[1]

[1]School of Computer Science and Technology, Harbin Institute of Technology, Shenzhen, China
[2]Beijing University of Posts and Telecommunications
{20b351002, jiangzelin, jaehyeon_soon}@stu.hit.edu.cn, {zhangxiaofeng, hwdu}@hit.edu.cn, byzhangjl@163.com, wxy_mic@outlook.com

Abstract

Abstractive text summarization is to generate concise summaries that well preserve both salient information and the overall semantic meanings of the given documents. However, real-world documents, e.g., financial reports, generally contain rich data such as charts and tabular data which invalidates most existing text summarization approaches. This paper is thus motivated to propose this novel approach to simultaneously summarize both textual and tabular data. Particularly, we first manually construct a "table+text → summary" dataset. Then, the tabular data is respectively embedded in a row-wise and column-wise manner, and the textual data is encoded at the sentence-level via an employed pre-trained model. We propose a salient detector gate respectively performed between each pair of row/column and sentence embeddings. The highly correlated content is considered as salient information that must be summarized. Extensive experiments have been performed on our constructed dataset and the promising results demonstrate the effectiveness of the proposed approach w.r.t. a number of both automatic and human evaluation criteria.

1 Introduction

The general purpose of abstractive text summarization is to generate concise summaries that well preserve both salient information and the overall semantic meanings of the given documents. Different from the extractive text summarization, which directly extracts the key sentences from the input documents, the abstractive text summarization is generated by abstracting the textual content at a semantic level.

Prior work has demonstrated that the abstractive summary generators could be trained in either supervised [See *et al.*, 2017; Xu *et al.*, 2021] or unsupervised [Nayeem *et al.*, 2018; Chu and Liu, 2019; Tampe *et al.*, 2022] manner. The merit of unsupervised abstractive summaries is that no domain experts are needed to craft the summaries to train the model, and thus could largely save the human annotation cost. To this end, a good number of pre-trained models, e.g., [Zhang *et al.*, 2020a; Liu and Lapata, 2019; Lewis *et al.*, 2020], have been proposed by taking the self-supervised information to guide the unsupervised model learning process. Not surprisingly, a very large corpus set is usually collected in their work to pre-train the generative language model (LM). On the contrary, the supervised models require human annotators to summarize a small portion of their data as the ground truth to supervise the later model learning process. Although the human cost is high, the quality of the generated summaries are reasonably higher than those generated in the unsupervised manner.

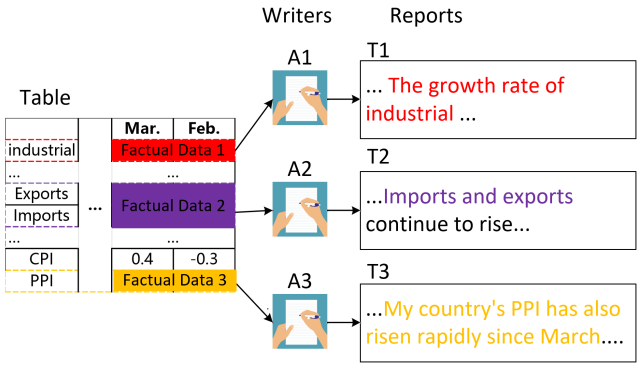

Figure 1: An illustrating example about how the "biased" financial reports are generated.

Different from the pure text summarization task, our proposed financial report summarization task contains two unique research challenges: (1) our task is to abstract both tabular data and textual data, whereas the conventional approaches only abstract textual data; and (2) our adopted textual data is assumed to contain *selective bias issue* induced by the diversified domain knowledge of human writers. For the first challenge, our financial reports usually contain rich data such as charts and tables to provide the factual data. As aforementioned, most existing approaches are proposed to abstract pure textual data [Zhang *et al.*, 2020a; Lewis *et al.*, 2020; Liu and Lapata, 2019] and could not be extended to our task. We also noticed that there exist a few related approaches generating text according to tabular data [Moosavi *et al.*, 2021; Chen *et al.*, 2022]. However, none of them are delicately

*Corresponding author. Email: zhangxiaofeng@hit.edu.cn

designed to directly abstract both tabular and textual data [Zhang *et al.*, 2020a; Lewis *et al.*, 2020; Liu and Lapata, 2019]. For the second challenge, the textual content of our financial report data is assumed to contain *selective bias issue*, as illustrated in Figure 1. To write financial reports for the same table, writer A1 may focus on factual data 1 as highlighted in red color according to his/her domain knowledge as well as the past expertise. Meanwhile, A2 and A3 might focus on the purple and orange tabular data. We model this phenomenon as *selective bias issue*, e.g. the human writer may only choose part of the tabular data to generate the corresponding financial report. Thus, both the original tabular data and the textual data should be utilized to generate a unbiased summary, and which motivates our work.

To the best of our knowledge, we are the first to raise this research problem, i.e., summarizing both tabular and textual data. A novel approach is proposed to generate Unbiased financial report Summaries using both Tabular and Textual data, abbreviated as USTT. Without loss of generality, our proposed "table+text $\rightarrow$ summary" (TTS) problem has three core sub tasks: (1) preserving salient information; (2) covering the overall semantic meanings; and (3) alleviating the *selective bias* issue. To preserve the salient information, if tabular data and textual data are coupled with each other, then such data should be considered as the salient information. For those textual data that are not highly correlated with the tabular data, we consider them as external knowledge-driven content written by the human writers based on their background domain knowledge. To preserve the overall semantic meanings, these external knowledge-driven content should be utilized to guarantee the coverage. To alleviate the *selective bias issue*, the original tabular data is assumed to contain complete factual data which is not fulled addressed by the given textual data. Thus, we use all historical textual and tabular data to approximately build an external unbiased knowledge base. For each queried tabular data, the retrieved sub set of documents from this base are used to form the unbiased set of documents to train the model.

The major contributions of this work are summarized as follows.

- To the best of our knowledge, we are the first to propose the "table+text$\rightarrow$summary" task to directly summarize financial reports using both tabular and textual data.
- We proposes a novel approach to simultaneously abstract financial reports using both tabular and textual data. The proposed approach consists of several components designed to detect salient information, and the selective bias issue is partially alleviated using an external unbiased knowledge base.
- We have constructed a financial report summarization dataset and made it publicly available [1]. Extensive experiments are performed and the promising experimental results demonstrate the effectiveness of the proposed approach w.r.t. a number of automatic and human evaluation metrics.

[1]https://huggingface.co/datasets/wza/USTT

2 Related Work

In the literature, text summarization has long been investigated which could be roughly classified into extractive [Kupiec *et al.*, 1995; Saini *et al.*, 2018] and abstractive [Gui *et al.*, 2019; Gui *et al.*, 2018; Nallapati *et al.*, 2016; See *et al.*, 2017] summarization techniques. A typical solution for extractive text summarization task is to extract the best matched sentences from source articles. Numerous works have been proposed for this task, such as integer linear programming [Galanis *et al.*, 2012] and various graph-based techniques [Erkan and Radev, 2004; Litvak *et al.*, 2010; Mihalcea and Tarau, 2004]. Alternatively, abstractive text summarization task is to generate concise text preserving the semantic meanings of the given documents [See *et al.*, 2017; Dong *et al.*, 2021], and thus has attracted increasing research efforts. The dominating techniques for abstractive text summarization are various sequence-to-sequence (seq2seq) based models [Fu *et al.*, 2020; Jangra *et al.*, 2020a; Jangra *et al.*, 2020b; Klein *et al.*, 2014; Krizhevsky *et al.*, 2017]. Generally, seq2seq-based approaches employ a LSTM model to encode input text into a fixed length embeddings and decode the summaries from the learnt embeddings in a generative manner. The attention mechanism [Bahdanau *et al.*, 2015] plays a key role in preserving salient information as well as guaranteeing the coverage, and thus is widely adopted in related approaches [Gui *et al.*, 2019; Fu *et al.*, 2020]. Recently, a number of graph-based approaches have been proposed for text summarization by treating sentences as graph nodes, and their edges denote their interaction behaviors. The hidden interactive patterns are assumed to contain both salient information as well as global semantic meanings. Beyond text summarization, recent approaches have focused on multi-modal summarization tasks, e.g., image-text to summary [Chen and Zhuge, 2018] and image-video-text to summary [Jangra *et al.*, 2020b] problems. Although there exist various summarization techniques for different types of data, none of them are proposed to simultaneously abstract both tabular and textual data similar to our financial report summarization problem.

3 Formulating the TTS Problem

Let $\mathcal{D} = \{\langle t_l, d_l, s_l \rangle\}_{l=1}^{N}$ denote the annotated "table+text $\rightarrow$ summary", where t_l and d_l respectively denote a table and its corresponding text, s_l denote the human-written summary containing m_l words denoted as $w_1, w_2, ..., w_{m_l}$. The purpose of the proposed TTS problem is to learn a model with parameters θ^* to maximize the probability P of generating summary s_l given t_l and d_l. Accordingly, our proposed TTS problem could be formulated as,

$$\theta^* = \arg\max_{\theta} \prod_{l=1}^{N} \prod_{q=1}^{m_l} P(w_q | w_{<q}, t_l, d_l, \theta), \qquad (1)$$

where $w_{<q}$ denotes the words before w_q.

4 The Proposed Approach

The proposed approach is detailed in this section which consists of four components: (1) an encoder component to encode both tabular and textual data; (2) a salient content detection component; (3) an external knowledge base; and (4)

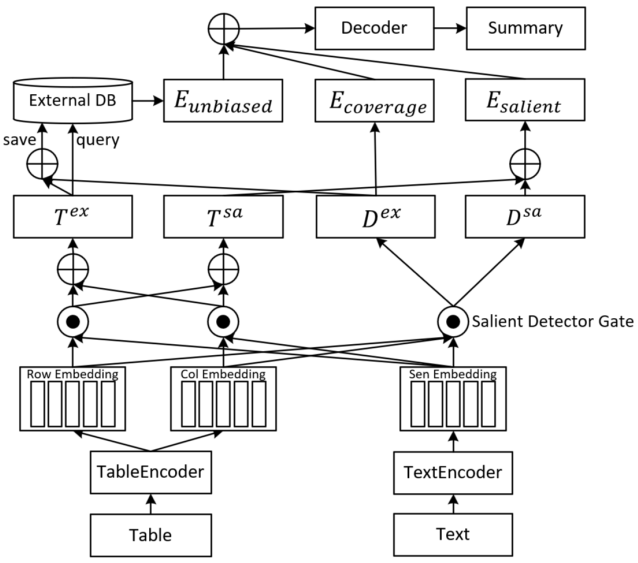

Figure 2: The architecture of the proposed approach.

a novel unbiased summarization generator. The proposed architecture is plotted in Figure 2, with each component discussed in individual subsections for clarity.

4.1 Encoding Tabular and Textual Data

To encode tabular data, it is a natural choice to perform the encoding processing along the column-wise and row-wise. Our employed table encoder contains two sub encoders: *row encoder* and *column encoder*.

Row encoder and column encoder. Similar to [Yin *et al.*, 2020], our row encoder is built as follows. First, we linearize each cell into 'row name|column name|cell value' and concatenate each cell representation with '[SEP]'. Then, the linearization results are fed into an encoder to acquire the embedding of a cell $c_{i,j}$, where i and j respectively denote the row and column index. To differentiate the cell importance in a row, a row-wise attention is performed [Gong *et al.*, 2019] with the cell score calculated as $\alpha_{i,j,j'} \propto exp(c_{i,j}^T W_a c_{i,j'})$, where W_a is a trainable parameter matrix. Thus, the weighted cell representations are computed as $v_{i,j}^{row} = \sum_{j',j' \neq j} \alpha_{i,j,j'} c_{i,j'}$. By applying a MLP layer, the embedding of a cell $c_{i,j}$ is finalized to $c_{i,j}^{row} = \tanh(W_b[c_{i,j}; v_{i,j}^{row}])$. After acquiring the embeddings of each cell, a row Row_i is embedded using a mean pooling operation over all its cells, given as

$$e(Row_i) = MeanPooling(c_{i,1}^{row}, c_{i,2}^{row}, ..., c_{i,j}^{row}). \quad (2)$$

Similarly, we could build the column encoder and the corresponding column embeddings are directly given as

$$e(Col_j) = MeanPooling(c_{1,j}^{col}, c_{2,j}^{col}, ..., c_{i,j}^{col}),$$

where W_c and W_d are trainable parameters.

Encoding textual data. We employ a BERT based encoder to encode text d_l and the mean embeddings of all tokens are considered as the text embeddings $e(d_l)$.

4.2 Detecting Salient Content

To extract salient tabular or textual data, we treat a row or a column as a basic unit to represent certain semantic meanings. Similarly, each sentence in the text d_l is treated as a basic textual unit. If the semantic meanings of a table unit are aligned with those of certain textual units, these aligned units represent highly correlated (coupled) content and therefore they are considered as salient content.

However, such alignment should be processed in an asymmetric way, i.e., "from table to text" and "from text to table". The perspective "from table to text" could capture which part of document is relatively important w.r.t. certain tabular data. The sentences with a higher score indicates a higher probability to become salient textual data. Accordingly, the perspective "from text to table" could capture the salient tabular data. Let matrix $R \in \mathcal{R}^{d \times O}$, $C \in \mathcal{R}^{d \times U}$ and $K \in \mathcal{R}^{d \times L}$ respectively denote the row, column and sentence embeddings. To calculate the pair-wise importance between a row/column and a sentence, we propose a salient detector gate which is computed as follows.

Salient detector gate between row and text. We first calculate the similarity matrix $A \in \mathcal{R}^{O \times L}$ measuring the distance between a row and a sentence, computed as

$$A = \tanh(R^T W_e K), \quad (3)$$

where $W_e \in \mathcal{R}^{d \times d}$ is the weight matrix. The gate from row to text is then calculated as

$$\begin{aligned} H_{r \to k} &= \tanh(W_f K + (W_g R)A), \\ \alpha_{r \to k} &= softmax(\omega_a^T H_{r \to k}), \end{aligned}$$

where parameters $W_f, W_g \in \mathcal{R}^{O \times d}$ and $\omega_a \in \mathcal{R}^O$, $\alpha_{r \to k}$ is the score associated with each sentence by setting a row as the anchor. Similarly, the score from text to row could be written as

$$\begin{aligned} H_{k \to r} &= \tanh(W_g R + (W_f K)A^T), \\ \alpha_{k \to r} &= softmax(\omega_b^T H_{k \to r}). \end{aligned}$$

Salient detector gate between column and text. Similar calculations could be performed to compute the salient detector gate between column embeddings and sentence embeddings, and the score $\alpha_{c \to k}$ (from column to text) and $\alpha_{k \to c}$ (from text to column) are directly given as

$$\begin{aligned} \alpha_{c \to k} &= softmax(\omega_c^T H_{c \to k}), \\ \alpha_{k \to c} &= softmax(\omega_d^T H_{k \to c}). \end{aligned}$$

Filtering salient and external content. We empirically set a threshold γ to decouple tabular and textual data into salient content and external content. It will separate the salient information shared between the text and tabular data from the external knowledge-driven content. This process enables our model to focus on the most relevant information when generating summaries. The salient tabular data could then be represented as

$$T^{sa} = [I(\alpha_{k \to r} \geq \gamma) \odot R; I(\alpha_{k \to c} \geq \gamma) \odot C], \quad (4)$$

where $I()$ denotes a mask function setting the corresponding position to 1 if $\alpha_{k \to r} \geq \gamma$ holds, ';' denotes concatenation operation and $\odot$ denotes an element-wise multiplication.

Eq. 4 means that if the score of the gate is higher than the threshold, its embeddings will be kept as salient content. On the contrary, the tabular data with its score is lower than the threshold will be filtered as external content, formulated as

$$T^{ex} = [I(\alpha_{k\to r} < \gamma) \odot R; I(\alpha_{k\to c} < \gamma) \odot C]. \quad (5)$$

Similarly, the salient and external textual data are respectively computed as

$$D^{sa} = I(\alpha_{r\to k} \geq \gamma) \odot K + I(\alpha_{c\to k} \geq \gamma) \odot K \quad (6)$$

$$D^{ex} = I(\alpha_{r\to k} < \gamma) \odot K + I(\alpha_{c\to k} < \gamma) \odot K \quad (7)$$

4.3 Building Unbiased Knowledge Base

Building knowledge base. To partially alleviate the *selective bias issue*, an external domain knowledge base (KB) is built in this section. Generally, experienced domain experts are desired to build such domain KB which inevitably involves a high annotation cost. To save such human cost, it is naturally to assume that all the historical data, which contains documents generated by different experts to interpret similar tabular data, could be used to build such unbiased KB. This unbiased KB $\mathcal{G} = \{\langle g_u \rangle\}_{u=1}^{U}$ is built as follows. First, we concatenate the external table embeddings with their corresponding external text embeddings, written as

$$g_u = [T_u^{ex}; D_u^{ex}], \quad (8)$$

where u denotes the index of the unbiased data. Note that we dynamically update this KB during the model training. Empirically, we set its capacity to 5,000 pieces of data, and a FIFO (first in first out) strategy is adopted to update the KB.

Retrieving knowledge base. To query this KB, T^{ex} in Eq.5 is used to probe this base. The best matched results in the KB are considered as the similar historical data and their associated text embeddings are retrieved to form a sub set $\mathcal{H} = \{\langle h_v \rangle\}_{v=1}^{V} \in \mathcal{G}$ which is considered as an unbiased set. We adopt cosine similarity to measure the relevance between a query variable and an existing record, written as

$$h_v = Sim(T^{ex}, g_u) = \frac{T^{ex} \times g_u}{|T^{ex}||g_u|} > \delta, \quad (9)$$

where σ is empirically set to 0.9 to ensure the results are highly correlated.

Detecting unbiased content. The set $\mathcal{H}$ is assumed to contain complementary textual content w.r.t the queried tabular data. To extract the unbiased content, we first clustered this set using the conventional K-means algorithm as

$$\sum_{v=0}^{V} \min_{\mu_j \in C} (||h_v - \mu_j||^2), \quad (10)$$

where μ_j is the centroid of each cluster $C \in \mathcal{H}$. For each queried table, we could find its nearest centroid μ_j. Then, documents belong to that cluster (with its centroid as μ_j) are assumed to contain documents written by different experts. This cluster of documents is assumed to be unbiased and the corresponding embeddings of document are fed into the decoder to generate summaries.

4.4 Generating Unbiased Summaries

The optimization goal is to maximize the saliency and coverage and minimize the selective bias simultaneously. To this end, the input of the employed decoder consists of three parts: salient embeddings, coverage embeddings and unbiased embeddings. The process for constructing these embeddings, as well as the design of the model loss, are detailed below.

Building salient embeddings. Both salient tabular T_{sa} and textual data D_{sa} are used to decode the summaries, and thus we concatenate these embeddings to form the salient embeddings, written as

$$E_{sa} = [T^{sa}; D^{sa}]. \quad (11)$$

Building coverage embeddings. Constructing the coverage embeddings is not as straightforward as building the salient embeddings. Typically, the source embeddings of both the table and text are used. However, both of them contain redundant content, especially in tabular data, which are equivalent to noisy data to some extent. To address this issue, we use the embeddings of external text D^{ex}. As introduced in Section 4.2, D^{ex} is a weighted vector by assigning lower weights to less important content, and higher weights to more important content, and its original contextual order keep unchanged. Thus, the coverage embeddings is set as $E_{coverage} = D^{ex}$.

Building unbiased embeddings. The extracted unbiased content from external KB are assumed to provide background knowledge to the similar tabular data, Therefore, the corresponding unbiased embeddings could be computed as

$$E_{unbiased} = \sum_{\arg\min_v (||h_v - \mu_j||^2), \mu_j \in C} h_v \quad (12)$$

Model loss. These three embeddings will be concatenated together to decode the summaries. Intuitively, these embeddings should not be equally weighted, and the weighted embeddings are given as

$$E_{final} = [\alpha \odot E_{salient}; \beta \odot E_{coverage}; \gamma \odot E_{unbiased}]. \quad (13)$$

These coefficients, i.e., α, β, γ are used to balance the contributions of the text encoder, tabular data encoder, and external KB embeddings in the final summary. Our designed summarizer is required to generate the most possible tokens as well as minimizing the loss between generated summary $\hat{s}$ and the ground truth s, and thus the overall model loss is defined as

$$\mathcal{L} = -\sum_{q=1}^{Len(s)} \log p(\hat{s}_{q+1} | s_q, E_{final}). \quad (14)$$

5 Experiments

5.1 Constructing Dataset

As there is no public dataset for our proposed problem, we manually construct this financial report summarization dataset containing both tabular and textual data. Each triplet

in this dataset consists of a table, a long text and a short human-generated summary recapitulating both tabular and textual data. To build such dataset, we first crawled 25,228 financial reports. Then, we automatically filter out reports without tabular data and 19,752 raw documents are left. To generate the dataset, we hire five experienced experts to annotate these financial reports and details of annotation process are described as follows.

Extracting tabular and textual data. As each financial report (pdf file) might contain more than one table and the annotators are required to read the report carefully and manually retrieve each table and its corresponding text. Note that the table structures vary dramatically for different stock brokers, the annotators are required to keep their original structures and convert these tabular data into editable format. At last, the retrieved each pair of data is numbered in sequence order and stored in the dataset. To process each financial report, one annotator usually takes fifteen minutes, and the overall working hour for this step is about 700 hours in total.

Writing summaries. Having tabular and textual data, these annotators are required to write a piece of concise summary based on both tabular and textual data. Each annotator independently writes a summary based on his own understandings. The criterion of each qualified triplet of data is given as follows.

- The long text should be highly correlated to the corresponding table, and must have at least 200 words.
- The human-written summaries should recapitulate both the tabular and the textual data.

Quality control. Since the summary writing is a creative task for the annotators, quality control is thus very important. We hire two additional annotators with more profound experiences to score the annotated data ranging from 1 to 5 based on the following two aspects:

- **Coverage:** does the summary contain sufficient information from both tabular and textual data?
- **Salience:** does the summary contain key information conveyed by both tabular and textual data?

To measure the inter-annotator agreement, we calculate the pair-wise cohen kappa where the annotators achieved 0.7057 in the coverage score and 0.7111 in the salience score. A triplet of data whose average score is greater than 3 will be kept to construct the dataset. At last, 13,897 pieces of data triplets are collected and are randomly split into training, validating and test sets consisting of 11117, 1390 and 1390 data pairs, respectively. The statistics of our constructed dataset are reported in Table 1.

	Mean	Q-5%	Q-95%
# rows per table	21.0	4	32
# columns per table	9.3	4	13
# words per text	358.1	215	789
# words per summary	47.3	16	93

Table 1: Statistics of our collected dataset, 'Q' refers to quantile

5.2 Evaluation Criteria and Experiment Settings

To evaluate the summarization results, both automatic evaluation and human evaluation criteria are adopted in the experiments. For automatic evaluation criteria, we chose the widely adopted ROUGE (R-1, R-2, R-L) [Lin, 2004] and BERTScore (BS) [Zhang* *et al.*, 2020b], where ROUGE criteria measures the recall performance on generating n-gram terms and the BERTScore measures the semantic similarity between the generated summary and the ground truth summary. To evaluate whether the generated summary is factually consistent with the input table or not, we adopt a triple-based factual consistency metric [Huang *et al.*, 2021] (FactScore). We respectively extract triple set Γ_{pred} and Γ_{tab} from the generated summary and the input table, and then the FactScore (FS) is calculated as

$$FS = (\Gamma_{pred} \cap \Gamma_{tab})/Count(\Gamma_{pred}),$$

where $Count()$ calculates the size of the set. For human evaluation criterion, we randomly select 100 pieces of data from each model and hire two groups of annotators, and each group contains 3 independent annotators. Annotators of the two group have similar backgrounds to alleviate human bias. The generated results together with their input tables and texts are anonymized and sent to another group for evaluation. The average results of these two groups are reported as our final results. The annotators are asked to give score ranges from 1 to 5 based on the following aspects: Coverage and Salience.

5.3 Compared Methods

As there are no related work for our proposed problem, we choose a number of state-of-the-art pre-trained models in the experiments to evaluate the model performance, i.e., BART [Lewis *et al.*, 2020], Bert2Bert [Rothe *et al.*, 2020], T5 [Raffel *et al.*, 2020], PEGASUS [Zhang *et al.*, 2020a]. As these compared methods were not proposed to summarize tabular data, we use the same method to pre-process the tabular data and concatenate them with the textual data as the input of each model for fair comparison.

5.4 Automatic Evaluation Results

We respectively train all models using only textual data, tabular data and both data, the corresponding results are reported in Table 2.

Similarity evaluation results. Both ROUGE-Typed criteria and BertScore measure the similarities between the generated summary and the ground truth. From this table, we have following observations. First, it is noticed that all models achieve better model performance when fed with textual data rather than tabular data. This is consistent with the expectation that text already provides sufficient information for summary generation. Second, if further fed with both table and text, the model performance of all baselines is significantly improved which verifies the necessity to generate summaries using both text and table data simultaneously. Finally, our proposed USTT achieves the best model performance as highlighted in bold when compared with all baselines. These observations verify that our designed model could well summarize both table and text data simultaneously.

	R-1	R-2	R-L	BS	FS
Input: text					
Bert2Bert_text	19.62	6.28	18.67	60.49	0.85
BART_text	23.20	7.11	22.04	62.86	2.31
T5_text	26.47	10.40	25.17	64.34	3.76
PEGASUS_text	24.11	7.61	22.94	63.53	2.78
Input: table					
Bert2Bert_table	16.14	4.67	15.25	59.70	1.81
BART_table	20.67	5.26	18.83	57.35	4.79
T5_table	21.34	5.62	19.32	57.26	5.10
PEGASUS_table	20.71	5.54	19.01	57.21	4.77
Input: table + text					
Bert2Bert_text+table	21.78	7.50	20.78	61.04	2.35
BART_text+table	27.51	9.52	26.09	63.66	5.65
T5_text+table	30.13	12.68	28.40	64.70	6.37
PEGASUS_text+table	29.41	10.99	28.05	64.65	6.77
Proposed					
USTT	**32.28**	**14.08**	**30.62**	**65.61**	**9.73**
USTT w/o table	26.59	9.75	25.02	64.17	4.75
USTT w/o db	29.86	12.59	28.30	64.63	6.97
USTT w/o coverage	29.35	11.81	27.47	65.01	9.13
USTT w/o salient	27.55	10.47	25.82	63.61	8.61

Table 2: Automatic evaluation results, 'BS' refers to the 'BERTScore' and 'FS' refers to the FactScore.

Factual consistency results. We report the factual consistency results in the last column of Table 2. It is obvious that the model performance of the second group (input:table) is much better than that of the first group (input:text). If feed the model using "table+text", the model performance could be further enhanced. This strongly verifies that it could improve the quality of the generated summaries if using the original table as input. The best compared model for this criterion is the PEGASUS, and it is clear that our proposed USTT could improve the model performance by 43.7% (9.73 vs 6.77). This result demonstrates the superiority of our designed model to directly summarize table and text simultaneously.

5.5 Ablation Study

To evaluate the effectiveness of each proposed component, we respectively remove the table, the external database, the coverage embedding and the salient embedding component, denoted as 'w/o table', 'w/o db', 'w/o coverage' and 'w/o salient', respectively . The corresponding results are reported in Table 2. Notably, the model's performance drops significantly after removing the table from its input, indicating the importance of tabular data in generating summaries. If removes the salient information, the model performance also drops a lot and we can infer that the coupled part between text and table could well preserve the discriminative content to generate summaries. Similarly, we could find that both coverage and external db components are helpful for our task. Moreover, it is also clear that the FactScore of the "USTT w/o table" drops by 51.2% which strongly shows the effectiveness of our designed model. We also observe that the coverage and salient information does not seriously affect the factual consistency, and this indicates that the pure text is not sufficient to summarize the "Text+Table" data.

	Coverage	Salience
Bert2Bert	2.35	2.84
BART	2.57	2.89
T5	2.63	3.12
PEGASUS	2.67	3.19
USTT	**2.98**	**3.41**
USTT w/o table	2.51	2.73
USTT w/o db	2.77	3.18
USTT w/o coverage	2.68	3.08
USTT w/o salient	2.71	2.97
Golden	2.96	3.48

Table 3: Human evaluation results.

5.6 Human Evaluation Results

To perform human evaluation, we first randomly choose a hundred of generated summaries and distribute them to independent annotators after anonymizing any identification information for human evaluation, and the corresponding results are reported in Table 3. First, the human evaluation results of our approach are better than all baselines w.r.t. coverage and salience criterion, and our model could achieve comparably results, i.e., 2.98 vs 2.96 and 3.41 vs 3.48, with the golden results. This verifies the superiority of our proposed approach. Second, the model performance significantly drops if removes table information as shown in the "USTT w/o table". This also verifies the importance of the table information which is consistent with our previous results. Similar observations could be found for coverage, salient and external db.

5.7 Effect of the External Database

To investigate the effect of the external unbiased knowledge base, we visualize the word cloud of the queried results returned from the database in Figure 3. The bigger the word, the more important the word. We have following two observations. First, the summary contains a good number of words that are not contained in the original text, and this shows that the external knowledge base could well provide more informative information. Second, the query results in Figure 3 (c)

(a) Text (b) Summary (c) DB

Figure 3: Word cloud of the original text, original summary and external knowledge base.

industry	Value	change
electron	1692.88	95.46
Computer	773.81	28.64
synthesis	148.8	9.22
Car	702.31	5.22
communication	158.4	3.74
Textiles	45.13	2.92
...	...	...
Electrical	2262.25	-0.45
real estate	404.02	-20.6
steel	284.86	-23.45
chemical industry	1391.83	-29.17
Non-bank	1410.82	-39.98
building material	564.12	-47.62
Leisure services	631.55	-52
nonferrous metal	442.78	-52.31
Bank	2110.38	-86.39
Pharmaceutical	3030.86	-119.77
Food	4178.07	-141.08
total	25597.6	-629.87

Text	Among them, the market value of food and beverage, pharmaceutical biology and banking fell more, down 14.108 billion yuan, 11.977 billion yuan and 8.639 billion yuan respectively. Electronic, computer and mechanical equipment rebounded more, with a rebound of 9.546 billion yuan, 2.864 billion yuan and 1.404 billion yuan respectively. Compared with June 11, ratio of TOP20 stocks was reduced by more than half on June 18. Among them, Hengrui Medicine, reduced their holdings by 2.54%, 0.56% and 0.46% respectively; Luxshare Precision, Oriental Fortune , Changjiang Power were reduced by 0.32%, 0.18% and 0.11%.
Summary	In terms of industries, the market value of most industries has declined.
USTT	In terms of different industries, compared with the previous month, the balance of two-finance and financing in most industries has rebounded. Equities and commodities were mixed. Domestic electrical equipment, chemicals and pharmaceutical and biological consumer goods fell back
w/o table	Food and beverages rose in the north, and small and medium-sized enterprises rebounded. Corporate profits continued to decline, turning into net outflows, and the total amount of financing balance reached a record high, but the growth rate contracted. Net capital from
w/o db	Food items decreased year-on-year, from the profit of industrial enterprises in May last month to a net outflow, mainly because the balance of food and beverages and medicine and biology rebounded more, which were 4.704 billion yuan and 4.717 billion yuan respectively.
w/o coverage	In terms of different industries, compared with the previous month, the balance of two-finance and financing in most industries has rebounded. The two-year corporate debt repayment pressure continued to increase, but the increase in money market interest rate and yields fell.
w/o salient	In March, the total net inflow of funds from northbound capital fell, the scale of newly issued funds increased and turned into outflow. The amount of financing reached a record high, but the growth rate has converged.

Figure 4: A case study for subjective assessment.

contains a few words that are not contained in the original text but are contained in the summary. This indicates that the constructed external knowledge base could well provide supplementary information and thus is helpful to alleviate the *selective bias issue*.

5.8 Qualitative Results Analysis

Ability to generate novel tokens. Figure 5 presents the experimental results on the ability of an abstractive summarizer to generate novel tokens. Higher bars indicate a greater ability to generate novel tokens. First, we observe that machine-generated summaries contain more novel tokens than the golden one. Second, our proposed model could generate more novel terms than all the baselines which indicates that our proposed model have a better generation ability. Last, from previous experiment, we found that the external knowledge base helps to alleviate the selective bias issue, and thus we remove this component is this experiment to further investigate its effect. The results from the "USTT w/o db" experiment supports the claim that the external knowledge base plays a crucial role in improving summarization performance.

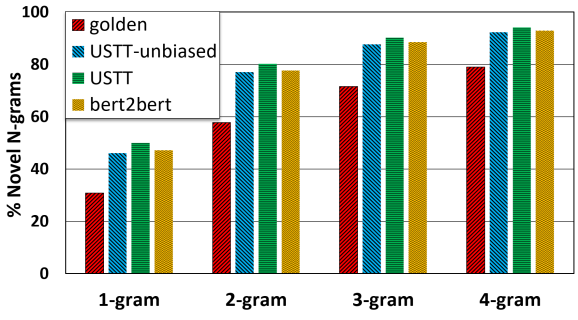

Figure 5: Evaluating the ability to generate novel n-grams compared to the input text.

A case study. We report a case in Figure 4 for readers to subjectively evaluate the quality of the generate summary. The left in this figure is the original tabular data, whereas the right part respectively reports the generated summaries by different versions of our proposed approach. First, it is well observed that the USTT well generates a comprehensive and unbiased summary. If removes the salient detection component, most generated textual data are hallucinatory (irrelevant to the tabular data). If removes the coverage component, the results become a bit better as it could generate textual content that are partially faithful to the original tabular data. However, the generated summary contains some content about "bond" which is not mentioned in the input table. If removes unbiased component, the corresponding summary only mentions content about "food" and "medical", whereas other industries are not mentioned. If removes the tabular data, similar observations to that of "-salient" could be found. The generated text contains many hallucination data. This indicates that both salient information and tabular data are crucial to generate a better summary.

6 Conclusion

Summarizing pure text has long been investigated in the literature. However, some domain-specific applications generally require to abstract both tabular and textual data. To tackle this issue, this paper proposed a novel "table+text $\rightarrow$ summary" problem and we manually constructed a novel "table+text $\rightarrow$ summary" summarization dataset. To resolve this challenging issue, we propose a novel unbiased financial report summarizer using both tabular and textual data. The proposed USTT model consists of several delicately designed modules and the model is trained to maximize the salience and coverage and minimize the unbiased loss simultaneously. Extensive experiments have been performed on the constructed dataset and the promising results demonstrate the superiority of the proposed approach over a number of SOTA baselines.

Acknowledgements

This work was supported in part by the Shenzhen Science and Technology Program under Grant No. JCYJ20200109113201726 and the National Natural Science Foundation of China under Grant No. 61872108.

References

[Bahdanau *et al.*, 2015] Dzmitry Bahdanau, Kyung Hyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. In *3rd International Conference on Learning Representations, ICLR 2015*, 2015.

[Chen and Zhuge, 2018] Jingqiang Chen and Hai Zhuge. Abstractive text-image summarization using multi-modal attentional hierarchical RNN. In *Proc. of EMNLP*, pages 4046–4056, Brussels, Belgium, October-November 2018. ACL.

[Chen *et al.*, 2022] Miao Chen, Xinjiang Lu, Tong Xu, Yanyan Li, Zhou Jingbo, Dejing Dou, and Hui Xiong. Towards table-to-text generation with pretrained language model: A table structure understanding and text deliberating approach. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, 2022.

[Chu and Liu, 2019] Eric Chu and Peter Liu. MeanSum: A neural model for unsupervised multi-document abstractive summarization. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 1223–1232. PMLR, 09–15 Jun 2019.

[Dong *et al.*, 2021] Luobing Dong, Meghana N Satpute, Weili Wu, and Ding-Zhu Du. Two-phase multidocument summarization through content-attention-based subtopic detection. *IEEE Transactions on Computational Social Systems*, 8(6):1379–1392, 2021.

[Erkan and Radev, 2004] Günes Erkan and Dragomir R Radev. Lexrank: Graph-based lexical centrality as salience in text summarization. *Journal of artificial intelligence research*, 22:457–479, 2004.

[Fu *et al.*, 2020] Xiyan Fu, Jun Wang, Jinghan Zhang, Jinmao Wei, and Zhenglu Yang. Document summarization with vhtm: Variational hierarchical topic-aware mechanism. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):7740–7747, Apr. 2020.

[Galanis *et al.*, 2012] Dimitrios Galanis, Gerasimos Lampouras, and Ion Androutsopoulos. Extractive multi-document summarization with integer linear programming and support vector regression. In *Proceedings of COLING 2012*, pages 911–926, 2012.

[Gong *et al.*, 2019] Heng Gong, Xiaocheng Feng, Bing Qin, and Ting Liu. Table-to-text generation with effective hierarchical encoder on three dimensions (row, column and time). 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 3143–3152, Hong Kong, China, November 2019. ACL.

[Gui *et al.*, 2018] Min Gui, Zhengkun Zhang, Zhenglu Yang, Yanhui Gu, and Guandong Xu. An effective joint framework for document summarization. In *Companion Proceedings of the The Web Conference 2018*, pages 121–122, 2018.

[Gui *et al.*, 2019] Min Gui, Junfeng Tian, Rui Wang, and Zhenglu Yang. Attention optimization for abstractive document summarization. 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 1222–1228, Hong Kong, China, November 2019. ACL.

[Huang *et al.*, 2021] Yi-Chong Huang, Xia-Chong Feng, Xiao-Cheng Feng, and Bing Qin. The factual inconsistency problem in abstractive text summarization: A survey. *CoRR*, 2021.

[Jangra *et al.*, 2020a] Anubhav Jangra, Raghav Jain, Vaibhav Mavi, Sriparna Saha, and Pushpak Bhattacharyya. Semantic extractor-paraphraser based abstractive summarization. In *Proceedings of the 17th International Conference on Natural Language Processing (ICON)*, pages 191–199, Indian Institute of Technology Patna, Patna, India, December 2020. NLP Association of India (NLPAI).

[Jangra *et al.*, 2020b] Anubhav Jangra, Adam Jatowt, Mohammad Hasanuzzaman, and Sriparna Saha. Text-image-video summary generation using joint integer linear programming. In Joemon M. Jose, Emine Yilmaz, João Magalhães, Pablo Castells, Nicola Ferro, Mário J. Silva, and Flávio Martins, editors, *Advances in Information Retrieval*, pages 190–198, Cham, 2020. Springer International Publishing.

[Klein *et al.*, 2014] Benjamin Klein, Guy Lev, Gil Sadeh, and Lior Wolf. Fisher vectors derived from hybrid gaussian-laplacian mixture models for image annotation. *arXiv preprint arXiv:1411.7399*, 2014.

[Krizhevsky *et al.*, 2017] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. *Commun. ACM*, 60(6):84–90, may 2017.

[Kupiec *et al.*, 1995] Julian Kupiec, Jan Pedersen, and Francine Chen. A trainable document summarizer. In *Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 68–73, 1995.

[Lewis *et al.*, 2020] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 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*, 2020.

[Lin, 2004] Chin-Yew Lin. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, 2004.

[Litvak *et al.*, 2010] Marina Litvak, Mark Last, and Menahem Friedman. A new approach to improving multilingual summarization using a genetic algorithm. In *Proceedings of the 48th annual meeting of the association for computational linguistics*, pages 927–936, 2010.

[Liu and Lapata, 2019] Yang Liu and Mirella Lapata. 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 2019. ACL.

[Mihalcea and Tarau, 2004] Rada Mihalcea and Paul Tarau. Textrank: Bringing order into text. In *Proceedings of the 2004 conference on empirical methods in natural language processing*, pages 404–411, 2004.

[Moosavi *et al.*, 2021] Nafise Sadat Moosavi, Andreas Rücklé, Dan Roth, and Iryna Gurevych. Scigen: a dataset for reasoning-aware text generation from scientific tables. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*, 2021.

[Nallapati *et al.*, 2016] Ramesh Nallapati, Bowen Zhou, Cicero dos Santos, Çağlar Gulçehre, and Bing Xiang. 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, 2016.

[Nayeem *et al.*, 2018] Mir Tafseer Nayeem, Tanvir Ahmed Fuad, and Yllias Chali. Abstractive unsupervised multi-document summarization using paraphrastic sentence fusion. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1191–1204, Santa Fe, New Mexico, USA, August 2018. ACL.

[Raffel *et al.*, 2020] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21:140:1–140:67, 2020.

[Rothe *et al.*, 2020] Sascha Rothe, Shashi Narayan, and Aliaksei Severyn. Leveraging pre-trained checkpoints for sequence generation tasks. *TACL*, 8:264–280, 2020.

[Saini *et al.*, 2018] Naveen Saini, Sriparna Saha, Anubhav Jangra, and Pushpak Bhattacharyya. Extractive single document summarization using multi-objective optimization: Exploring self-organized differential evolution, grey wolf optimizer and water cycle algorithm. *Knowledge-Based Systems*, 164, 11 2018.

[See *et al.*, 2017] Abigail See, Peter J Liu, and Christopher D Manning. Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1073–1083, 2017.

[Tampe *et al.*, 2022] Ignacio Tampe, Marcelo Mendoza, and Evangelos Milios. Neural abstractive unsupervised summarization of online news discussions. In Kohei Arai, editor, *Intelligent Systems and Applications*, pages 822–841, Cham, 2022. Springer International Publishing.

[Xu *et al.*, 2021] Hongyan Xu, Hongtao Liu, Pengfei Jiao, and Wenjun Wang. Transformer reasoning network for personalized review summarization. In *SIGIR '21: The 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2021.

[Yin *et al.*, 2020] Pengcheng Yin, Graham Neubig, Wen-tau Yih, and Sebastian Riedel. TaBERT: Pretraining for joint understanding of textual and tabular data. In *Proc. of ACL*, pages 8413–8426, Online, July 2020. ACL.

[Zhang *et al.*, 2020a] Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter Liu. PEGASUS: Pre-training with extracted gap-sentences for abstractive summarization. In Hal Daumé III and Aarti Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 11328–11339. PMLR, 13–18 Jul 2020.

[Zhang* *et al.*, 2020b] Tianyi Zhang*, Varsha Kishore*, Felix Wu*, Kilian Q. Weinberger, and Yoav Artzi. Bertscore: Evaluating text generation with bert. In *ICLR*, 2020.