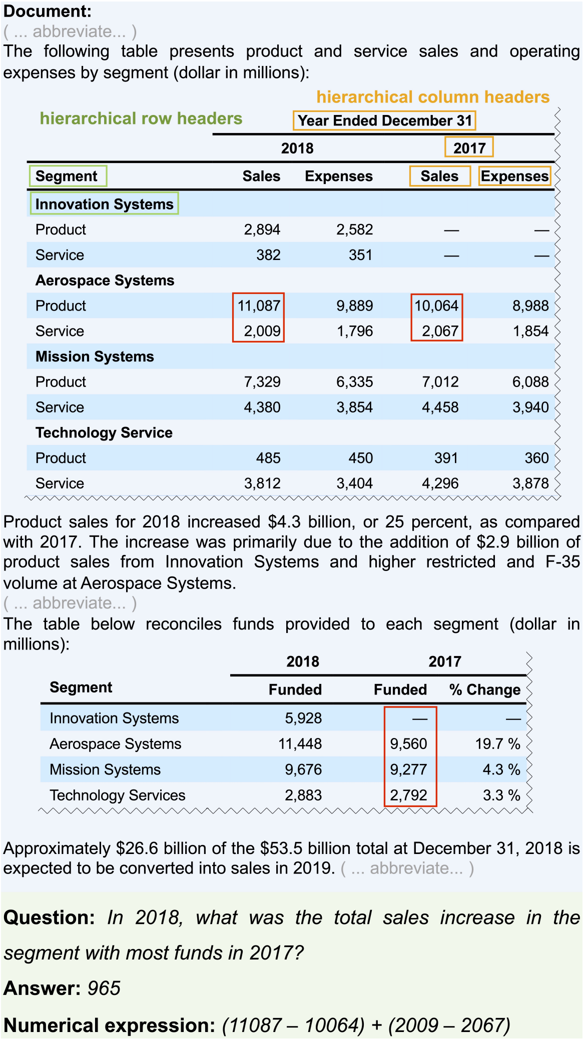
MultiHiertt: Numerical Reasoning over Multi Hierarchical Tabular and Textual Data

Yilun Zhao[1] **Yunxiang Li**[2] **Chenying Li**[3] **Rui Zhang**[4]
[1]Yale University [2]The Chinese University of Hong Kong
[3]Northeastern University [4]Penn State University
yilun.zhao@yale.edu 1155124348@link.cuhk.edu.hk
li.chenyin@northeastern.edu rmz5227@psu.edu

Abstract

Numerical reasoning over hybrid data containing both textual and tabular content (e.g., financial reports) has recently attracted much attention in the NLP community. However, existing question answering (QA) benchmarks over hybrid data only include a single flat table in each document and thus lack examples of multi-step numerical reasoning across multiple hierarchical tables. To facilitate data analytical progress, we construct a new large-scale benchmark, MultiHiertt, with QA pairs over **Multi Hier**archical **T**abular and **T**extual data. MultiHiertt is built from a wealth of financial reports and has the following unique characteristics: 1) each document contain multiple tables and longer unstructured texts; 2) most of tables contained are hierarchical; 3) the reasoning process required for each question is more complex and challenging than existing benchmarks; and 4) fine-grained annotations of reasoning processes and supporting facts are provided to reveal complex numerical reasoning. We further introduce a novel QA model termed MT2Net, which first applies facts retrieving to extract relevant supporting facts from both tables and text and then uses a reasoning module to perform symbolic reasoning over retrieved facts. We conduct comprehensive experiments on various baselines. The experimental results show that MultiHiertt presents a strong challenge for existing baselines whose results lag far behind the performance of human experts. The dataset and code are publicly available at https://github.com/psunlpgroup/MultiHiertt.



Document:
(... abbreviate...)
The following table presents product and service sales and operating expenses by segment (dollar in millions):

hierarchical column headers
hierarchical row headers

	Year Ended December 31			
	2018		2017	
Segment	Sales	Expenses	Sales	Expenses
Innovation Systems				
Product	2,894	2,582	—	—
Service	382	351	—	—
Aerospace Systems				
Product	11,087	9,889	10,064	8,988
Service	2,009	1,796	2,067	1,854
Mission Systems				
Product	7,329	6,335	7,012	6,088
Service	4,380	3,854	4,458	3,940
Technology Service				
Product	485	450	391	360
Service	3,812	3,404	4,296	3,878

Product sales for 2018 increased $4.3 billion, or 25 percent, as compared with 2017. The increase was primarily due to the addition of $2.9 billion of product sales from Innovation Systems and higher restricted and F-35 volume at Aerospace Systems.
(... abbreviate...)
The table below reconciles funds provided to each segment (dollar in millions):

	2018	2017	
Segment	Funded	Funded	% Change
Innovation Systems	5,928	—	—
Aerospace Systems	11,448	9,560	19.7 %
Mission Systems	9,676	9,277	4.3 %
Technology Services	2,883	2,792	3.3 %

Approximately $26.6 billion of the $53.5 billion total at December 31, 2018 is expected to be converted into sales in 2019. (... abbreviate...)

Question: *In 2018, what was the total sales increase in the segment with most funds in 2017?*
Answer: *965*
Numerical expression: *(11087 – 10064) + (2009 – 2067)*

Figure 1: An example of MultiHiertt: The system needs to first locate which segment got the most funds in 2017 in the second hierarchical table, then select relevant numbers from the first hierarchical table and generate the correct reasoning program to get the answer. The annotated supporting facts are highlighted in red, and the hierarchical column and row headers are highlighted in orange and green, respectively.

1 Introduction

In recent years, as key to many NLP tasks such as QA, there is a flurry of works on numerical reasoning over various types of data including textual data (Dua et al., 2019; Amini et al., 2019; Xie and Sun, 2019) and tabular data (Moosavi et al., 2021; Suadaa et al., 2021). More recently, numerical reasoning over hybrid data containing both textual and tabular content (Zhu et al., 2021; Chen et al., 2021) has attracted much attention. For example,

Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics
Volume 1: Long Papers, pages 6588 - 6600

the FinQA dataset (Chen et al., 2021) focuses on questions that require numerical reasoning over financial report pages, e.g., "What portion of the total identifiable net assets is in cash?". Such questions need the system to locate relevant cells in the tabular content and then perform a division operation to get the final answer.

However, existing QA datasets over hybrid data only contain a single flat table in each document (Zhu et al., 2021; Chen et al., 2021). Therefore, they lack examples that require multi-step reasoning processes across multiple paragraphs and hierarchical tables. Hierarchical tables are widely used in scientific or business documents. A hierarchical table usually contains multi-level headers, which makes cell selection much more challenging because it requires multi-level and bi-dimensional indexing techniques. For instance, consider the example of our proposed dataset MULTIHIERTT in Figure 1, each table contains both column headers and row headers, which are hierarchical in nature. And ignoring the row / column headers or not reasoning on the entire header hierarchy may lead to the wrong result. For instance, in the given example, if the system simply searched for cells with a flat row header containing "Product" and "Service" and column header containing "2018", it may mistakenly return the value 2,894 and 382 appearing in the beginning of the first table. Additionally, in real life, when analyzing financial reports, professionals such as analysts or investors often refer to multiple hierarchical tables and multiple paragraphs to obtain conclusions. For instance, finding "the segments with most funds in 2017" requires the system to locate and perform numerical reasoning on the second hierarchical table. Then the system should use the results gained from the second table to reason on the first table. However, existing QA datasets lack such examples of reasoning across multiple tables.

To address these shortcomings, we present MULTIHIERTT: an expert-annotated dataset that contains 10,440 QA pairs, along with annotations of reasoning processes and supporting facts. To the best of our knowledge, MULTIHIERTT is the first dataset for solving complicated QA tasks over documents containing multiple hierarchical tables and paragraphs. In addition, to address the challenge of MULTIHIERTT, we propose MT2Net to first retrieve supporting facts from financial reports then generate executable reasoning programs to answer the questions. Our experiments show that MT2Net outperforms all other baselines and achieves 38.43% F1 score. However, all models still lag far behind the performance of human experts with 87.03% in F1. It demonstrates MULTIHIERTT presents a strong challenge for existing baseline models and is a valuable benchmark for future research.

The main contribution of this work can be summarized as follows:

- We propose a new large-scale dataset MULTIHIERTT. It contains 10,440 examples along with fully annotated numerical reasoning processes and supporting facts. A strict quality control procedure is applied to ensure the meaningfulness, diversity, and correctness of each annotated QA example.

- Compared with existing datasets, each document in MULTIHIERTT contains multiple hierarchical tables and longer unstructured text. A more complex reasoning process across multiple tables and paragraphs is required to correctly answer the question.

- We propose a novel QA model, MT2Net. The model first applies facts retrieving to extract relevant supporting facts from both hierarchical tables and text. And it then uses a reasoning module to reason over retrieved facts.

- Comprehensive experiments are conducted on various baselines. The experimental results demonstrate that the current QA models still lag far behind the human expert performance, and further research is needed.

2 Related Work

Question Answering Benchmark There are numerous QA datasets focusing on text, table/knowledge base (KB), and hybrid data. SQuAD (Rajpurkar et al., 2016) and CNN/Daily Mail (Hermann et al., 2015) are classic datasets for textual data. Table/KB QA datasets mainly focus on structured tables (Pasupat and Liang, 2015; Zhong et al., 2017; Yu et al., 2018; Nan et al., 2022) and knowledge bases (Berant et al., 2013; Yih et al., 2015; Talmor and Berant, 2018; Xie et al., 2022). And some recent works focus on reasoning over more complex tables including hierarchical tables (Cheng et al., 2021b; Katsis et al.,

QA Dataset	Textual & Tabular Data / Doc (DB)			Numerical Reasoning	# Doc (DB)	# Questions
	Avg. # words	Table types	Avg. # tables			
Textual QA Dataset						
DROP (Dua et al., 2019)	210.0	✗	✗	✓	6,735	45,959
MathQA (Amini et al., 2019)	37.9	✗	✗	✓	37,259	37,259
Math23K (Xie and Sun, 2019)	35.4	✗	✗	✓	23,161	23,161
Tabular QA Dataset						
WTQ (Pasupat and Liang, 2015)	✗	Flat	1	✗	2,108	22,033
Spider (Yu et al., 2018)	✗	Relational	5.13	✗	200	10,181
AIT-QA (Katsis et al., 2021)	✗	Hierarchical	1	✗	116	515
HiTab (Cheng et al., 2021b)	✗	Hierarchical	1	few	3,597	10,686
Hybrid QA Dataset						
HybridQA (Chen et al., 2020)	2,326.0	Flat	1	✗	13,000	69,611
MMQA (Talmor et al., 2021)	240.7	Flat	1	✗	29,918	29,918
GeoTSQA (Li et al., 2021)	52.4	Flat	1.58	few	556	1,012
TAT-QA (Zhu et al., 2021)	43.6	Mostly Flat	1	✓	2,757	16,552
FINQA (Chen et al., 2021)	628.1	Flat	1	✓	2,789	8,281
MULTIHIERTT (Ours)	1,645.9	Hierarchical	3.89	✓	2,513	10,440

Table 1: Comparison of MULTIHIERTT with other QA datasets (Doc, DB denote Document and DataBase).

2021). More recently, there are also some pioneering studies working on QA over hybrid data. Specifically, HybridQA (Chen et al., 2020), TAT-QA (Zhu et al., 2021), and FinQA (Chen et al., 2021) focus on both textual and tabular data, while MMQA (Talmor et al., 2021) focus on QA over text, tables, and images. In addition, reasoning including numerical reasoning and multi-hop reasoning has gained attention lately. For example, DROP (Dua et al., 2019) is a machine reading comprehension benchmark that requires numerical reasoning on text data. HotpotQA (Yang et al., 2018) and HybridQA (Chen et al., 2020) are datasets requiring multi-hop reasoning.

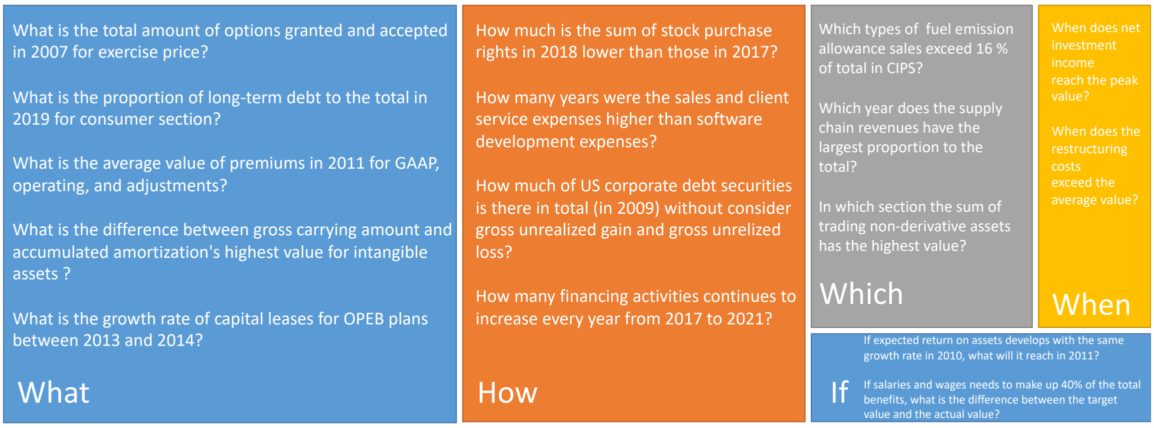
Numerical Reasoning Numerical reasoning plays an important role in different NLP tasks (Dua et al., 2019; Zhang et al., 2021; Chen et al., 2021; Zhu et al., 2021). To enhance the model's numerical reasoning ability, some work adapt standard extractive QA models with specialized modules to perform numerical reasoning (Ran et al., 2019; Hu et al., 2019). Recent work also focus on probing and injecting numerical reasoning skills to pre-trained language models (Geva et al., 2020; Lin et al., 2020; Zhang et al., 2020; Berg-Kirkpatrick and Spokoyny, 2020). Meanwhile, various benchmarks and models are proposed to solve math word problems (Koncel-Kedziorski et al., 2016; Xie and Sun, 2019; Amini et al., 2019; Hendrycks et al., 2021; Hong et al., 2021; Cobbe et al., 2021). The most recent numerical reasoning QA benchmark over hybrid data are FinQA (Chen et al., 2021) and TAT-QA (Zhu et al., 2021).

Financial NLP Financial NLP has attracted much attention recently. There have been various application in different tasks like risk management (Han et al., 2018; Theil et al., 2018; Nourbakhsh and Bang, 2019; Mai et al., 2019; Wang et al., 2019), asset management (Filgueiras et al., 2019; Blumenthal and Graf, 2019), market sentiment analysis (Daudert et al., 2018; Tabari et al., 2018; Buechel et al., 2019), financial event extraction (Ein-Dor et al., 2019; Zhai and Zhang, 2019) and financial question answering (Lai et al., 2018; Maia et al., 2018). More recently, pre-trained language models are presented for finance text mining (Araci, 2019; Yang et al., 2020). The most relevant work to us is FinQA (Chen et al., 2021) and TAT-QA (Zhu et al., 2021), which both construct a QA dataset acquiring numerical reasoning skills on financial reports with tabular data.

3 MULTIHIERTT Dataset

3.1 Data Collection and Preprocessing

MULTIHIERTT are deployed based on the FinTabNet dataset (Zheng et al., 2021), which contains 89,646 pages with table annotations extracted from the annual reports of S&P 500 companies. For each table contained, the FinTabNet dataset provides a detailed HTML format annotation, in which table hierarchies and cell information such as text and



What is the total amount of options granted and accepted in 2007 for exercise price?

What is the proportion of long-term debt to the total in 2019 for consumer section?

What is the average value of premiums in 2011 for GAAP, operating, and adjustments?

What is the difference between gross carrying amount and accumulated amortization's highest value for intangible assets ?

What is the growth rate of capital leases for OPEB plans between 2013 and 2014?

How much is the sum of stock purchase rights in 2018 lower than those in 2017?

How many years were the sales and client service expenses higher than software development expenses?

How much of US corporate debt securities is there in total (in 2009) without consider gross unrealized gain and gross unrelized loss?

How many financing activities continues to increase every year from 2017 to 2021?

Which types of fuel emission allowance sales exceed 16 % of total in CIPS?

Which year does the supply chain revenues have the largest proportion to the total?

In which section the sum of trading non-derivative assets has the highest value?

When does net investment income reach the peak value?

When does the restructuring costs exceed the average value?

If expected return on assets develops with the same growth rate in 2010, what will it reach in 2011?

If salaries and wages needs to make up 40% of the total benefits, what is the difference between the target value and the actual value?

Figure 2: Examples of question by top-5 most frequent starting words, where box size represents frequency.

formats can be extracted and post-processed according to HTML tags.

The raw data is filtered as follows: First, we extract documents with 1 to 4 pages and 2 to 6 tables from FinTabNet. Second, we filter out the documents with limited textual contents. Third, as we aim for the numerical reasoning ability, we also exclude documents with tables containing little numerical information. Then, we use a pre-processing script to extract the hierarchical structure of each HTML-format table. And we ignore those tables that cannot be handled by the pre-processing script. As a result, a total of 4,791 documents were selected for further annotation.

3.2 Question-Answer Pair Annotation

For each document selected in §3.1, the annotators are required to compose one or two QA examples along with detailed annotation. The process of annotating each QA example is as follows: 1) The annotators are first asked to compose a complex question that requires numerical reasoning and is meaningful for helping novices understand the annual reports. The annotators are encouraged to compose questions that require the information from both the textual and tabular content or from multiple tables. 2) For those questions requiring numerical expression, the annotators are then asked to write down the reasoning programs to answer the question. In detail, the annotators are asked to elaborate on the operation steps to answer the question. The definitions of all operations are shown in Table 7 in Appendix. 3) They are also required to mark all the supporting facts from tabular and textual content for each question.

3.3 Quality Control

Strict quality control procedures are designed to ensure the quality of dataset annotation, especially the diversity and meaningfulness of proposed questions. The human evaluation scores and inter-evaluator agreements are reported in Table 2.

Annotation Quality	**%S $\geq$ 4**	**Agree**	**Kappa / 95% CI**
Question Complexity	76.8	0.77	0.72 / [0.65, 0.79]
Question Correctness	93.2	0.91	0.83 / [0.77, 0.89]
Question Meaningfulness	91.4	0.87	0.81 / [0.74, 0.88]
Reasoning Correctness	92.4	0.92	0.89 / [0.84, 0.94]
Support Facts Correctness	84.9	0.81	0.77 / [0.72, 0.82]
Answer Correctness	94.0	0.93	0.90 / [0.87, 0.93]

Table 2: Human evaluation over 100 samples of MULTIHIERTT. Four internal evaluators are asked to rate the samples on a scale of 1 to 5. We report 1) percent of samples that have average score $\geq$ 4 to show high quality of MULTIHIERTT; and 2) percent of agreement and Randolph's Kappa with 95% CI (Randolph, 2005) to show high inter-annotator agreement of MULTIHIERTT.

Expert Annotators To help improve the annotation process, we first enroll five experts with professional experience in finance. During annotation, they are asked to provide feedback regarding the task instructions and the user experience of the annotation interface, based on which we iteratively modify the annotation guideline and interface design. In the stage of crowd-sourced annotation, we hire 23 graduate students (14 females and 9 males) majoring in finance or similar discipline. Before starting the official annotation process, each annotator is given a two-hour training session to learn the requirements and the annotation interface.

Annotation De-Biasing As suggested in previous research (Kaushik and Lipton, 2018; Clark et al., 2019; Jiang and Bansal, 2019; Yang et al., 2022), consider annotation bias of QA benchmarks is of great significance. During the pilot annotation period, we found that when generating question-answering pairs, annotators may prefer simpler ones. To solve this issue, we use thresholds to restrict the proportions of questions with different numbers of numerical reasoning steps. Meanwhile, the proportions of questions with span selection answer types are set to $\leq 20\%$. To further increase the diversity of question-answer pair annotation, we also select and include 2,119 QA examples from FinQA (Chen et al., 2021).

Multi-Round Validation To further ensure the diversity and correctness of proposed question-reasoning pairs, each document is assigned to three annotators and one verifier in order. For annotators, each is required to first validate the previous annotator's annotation and fix the mistakes if there are. Then, they are asked to create one or two more question-reasoning pairs that are different from the existing ones. After each annotator finishes tasks, we assign another verifier with good performance on this project to validate all the annotations.

3.4 Dataset Analysis

Core statistics of MULTIHIERTT are reported in Table 3. Table 1 shows a comprehensive comparison of related datasets. MULTIHIERTT is the first dataset to study numerical reasoning questions over hybrid data containing multiple hierarchical tables. Compared with TAT-QA and FinQA, documents in MULTIHIERTT contain longer unstructured input text and multiple tables, making the evidence retrieving and reasoning more challenging. And MULTIHIERTT has diverse and complex questions, as illustrated in Figure 2.

We also analyze supporting facts coverage for each question. In MULTIHIERTT, 1) 10.24% of the questions only require the information in the paragraphs to answer; 2) 33.09% of the questions only require the information in one table to answer; 3) 7.93% require the information in more than one table but without paragraphs to answer; 4) 48.74% require both the text and table information to answer, and among them, 23.20% required the information in more than one table. The average number of annotated supporting facts are 7.02. Meanwhile, among those questions with annotated numerical reasoning programs, 28.94% of them have 1 step; 37.76% of them have 2 steps; 15.21% of them have 3 steps; and 18.10% of them have more than 3 steps. As a result, the average number of numerical reasoning steps is 2.47.

Property	**Value**
# Examples (Q&A pairs with annotation)	10,440
# Documents	2,513
Vocabulary	24,193
Avg. # Sentences in input text	68.06
Avg. # Words in input text	1,645.9
Avg. # Tables per Document	3.89
Avg. # Rows per Table	10.78
Avg. # Columns per Table	4.97
Avg. # Question Length	16.78
Training Set Size	7,830 (75%)
Development Set Size	1,044 (10%)
Test Set Size	1,566 (15%)

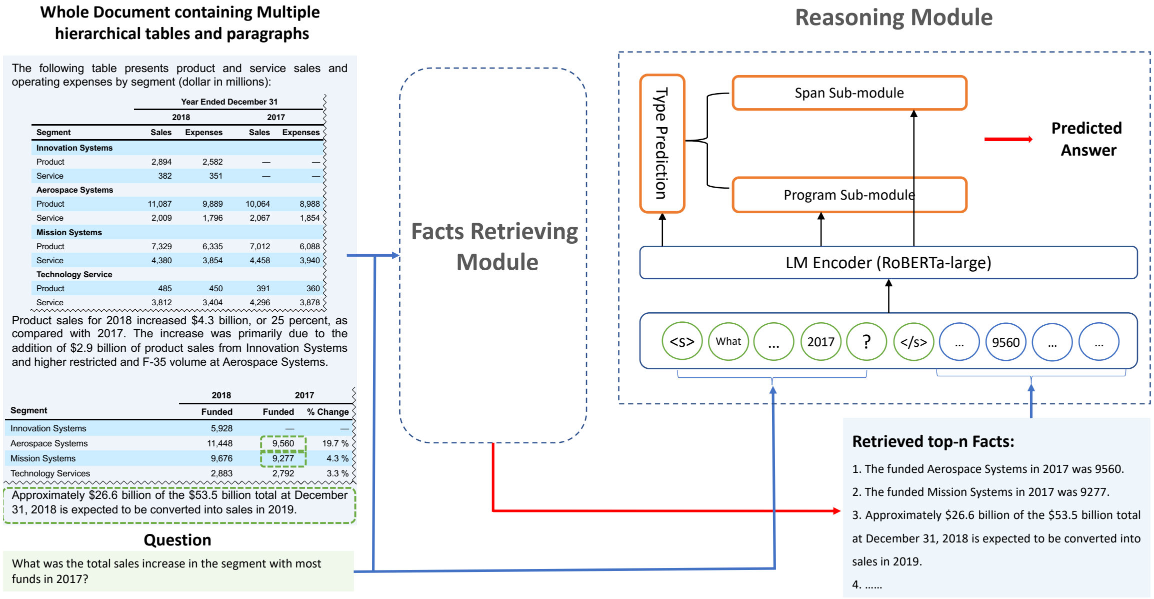
Table 3: Core Statistics of MULTIHIERTT.

4 MT2Net Model

To address the challenge of MULTIHIERTT, we propose a framework named MT2Net. Figure 3 gives an overview of our proposed model. MT2Net first applies fact retrieving module to extract relevant supporting facts from the hierarchical tables and paragraphs. Then, a reasoning module is adapted to perform reasoning over retrieved facts and get the final answer.

Fact Retrieving Module The whole input text in each document of MULTIHIERTT can exceed 3,000 tokens and contain many numbers, which is beyond the capability of the current popular QA models (Devlin et al., 2019; Liu et al., 2019). Therefore, we employ a fact retrieving module to first retrieve the supporting facts from the documents. Previous works on hybrid datasets (Zhu et al., 2021; Chen et al., 2021; Li et al., 2021) use templates to flatten each row of the table into sentences. And our facts retrieving module applies similar ideas. However, different from other hybrid datasets, most tables in MULTIHIERTT are hierarchical. Therefore, we turn each cell into a sentence, along with its hierarchical row and column headers. For example, the first data cell in the first table in Figure 1 is translated as "For Innovation Systems of Segment, sales of product in 2018, Year Ended December 31 is 2,894".

We concatenate each annotated supporting fact with the question as input to train a BERT-based



The following table presents product and service sales and operating expenses by segment (dollar in millions):

	Year Ended December 31			
	2018		2017	
Segment	Sales	Expenses	Sales	Expenses
Innovation Systems				
Product	2,894	2,582	—	—
Service	382	351	—	—
Aerospace Systems				
Product	11,087	9,889	10,064	8,988
Service	2,009	1,796	2,067	1,854
Mission Systems				
Product	7,329	6,335	7,012	6,088
Service	4,380	3,854	4,458	3,940
Technology Service				
Product	485	450	391	360
Service	3,812	3,404	4,296	3,878

Product sales for 2018 increased $4.3 billion, or 25 percent, as compared with 2017. The increase was primarily due to the addition of $2.9 billion of product sales from Innovation Systems and higher restricted and F-35 volume at Aerospace Systems.

	2018	2017	
Segment	Funded	Funded	% Change
Innovation Systems	5,928	—	—
Aerospace Systems	11,448	9,560	19.7 %
Mission Systems	9,676	9,277	4.3 %
Technology Services	2,883	2,792	3.3 %

Approximately $26.6 billion of the $53.5 billion total at December 31, 2018 is expected to be converted into sales in 2019.

What was the total sales increase in the segment with most funds in 2017?

1. The funded Aerospace Systems in 2017 was 9560.
2. The funded Mission Systems in 2017 was 9277.
3. Approximately $26.6 billion of the $53.5 billion total at December 31, 2018 is expected to be converted into sales in 2019.
4. ......

Figure 3: The framework of MT2Net. The model consists of a facts retrieving module and a reasoning module.

bi-classifier (Devlin et al., 2019). During the inference stage, the top-n sentences are retrieved as supporting facts. They are reordered according to the order of appearance in the original document. Then they will serve as input to reasoning module.

Reasoning Module We first use pre-trained LMs to encode the retrieved sentences from the facts retrieving module. Then, we divide the answers into two types: arithmetic program and span. For each answer type, we use a unique sub-module to calculate the conditional answer probability $P(\text{answer}|\text{type})$:

Program sub-module: The structure is similar with the program generator of FinQANet (Chen et al., 2021). The sub-module aims to generate the executable program to answer the question. Specifically, an LSTM is used for decoding. At each decoding step T, the LSTM can generate one token from 1) the numbers from the retrieved, 2) pre-defined operators, and 3) the tokens already generated in the previous steps. After the completion of generation, the sub-module will execute the generated programs and get the predicted answer.

Span sub-module: The span sub-module aims to select the predicted span candidate, which is a span of retrieved sentences. The answer probability is defined as the product of the probabilities of the start and end positions in the retrieved evidence.

Meanwhile, an extra output layer is used to predict the probability $P(\text{type})$ of each answer type. In particular, we take the output vector [CLS] from LMs as input to compute the probability. In the training stage, the final answer probability is defined as the joint probability over all feasible answer types, i.e., $\sum_{\text{type}} P(\text{type}) \times P(\text{answer}|\text{type})$. Here, both $P(\text{type})$ and $P(\text{answer}|\text{type})$ is learned by the model. In the inference stage, the model first selects the most probable answer type and then uses corresponding sub-modules to predict the answer.

5 Experiments

5.1 Baseline Systems

TAGOP TAGOP[1] is the baseline model for TAT-QA dataset (Zhu et al., 2021). It first uses sequence tagging with the Inside–Outside tagging (IO) approach to extract supporting facts. Then an operator classifier is applied to decide which operator is used to infer the final answer via extracted facts. Different from ours, TAGOP can only perform symbolic reasoning with a single type of pre-defined aggregation operators (e.g. change Ratio, division), and might fail to answer complex questions requiring multi-step reasoning.

FinQANet FinQANet[2] is the baseline model for FinQA dataset (Chen et al., 2021). It first uses a BERT-based retriever to take the top-n supporting facts. Then a program generator is applied to generate the reasoning programs to get the final answers.

[1] https://github.com/NExTplusplus/tat-qa

[2] https://github.com/czyssrs/FinQA

Different from ours, FinQANet ignores the hierarchical structure of tables when linearizing each row of a table. And it is not designed to answer span selection questions.

Longformer + Reasoning module To demonstrate the necessity of breaking up models into facts retrieving and reasoning modules, we directly use the pre-trained Longformer-base[3] (Beltagy et al., 2020) as the input encoder in the reasoning module, and encode the whole document.

Fact Retrieving Module + TAPAS We employ TAPAS (MASKLM-base)[4] (Herzig et al., 2020; Eisenschlos et al., 2020) as a baseline over tabular data. TaPas is pretrained over large-scale tables and associated text from Wikipedia jointly. To finetune it, we use the table with most supporting facts along with the answer as input for each example. For the inference stage, the table with most portion of top-15 retrieved facts is used as input.

Fact Retrieving + NumNet NumNet+[5] (Ran et al., 2019) has demonstrated its effectiveness on the DROP dataset (Dua et al., 2019). It designs a NumGNN between the encoding and prediction module to perform numerical comparison and numerical reasoning. However, NumNet+ only supports addition and subtraction when performing symbolic reasoning, thus cannot handle those complex questions requiring operators such as division.

Fact Retrieving Module + Seq2Prog A Seq2Prog architecture adopted from baseline of MathQA dataset (Amini et al., 2019) is used as the reasoning module. Specifically, we use a biLSTM encoder and an LSTM decoder with attention.

5.2 Implementation Details

For the fact retrieving module, we use BERT-base as the classifier. Since most of the examples in our dataset have less than 7 supporting facts (89.3%), and we find that longer inputs might lower the performance of the reasoning module, we take the top-10 retrieving facts as the retriever results. For the reasoning module, we experiment on using BERT (Devlin et al., 2019) and RoBERTa (Liu et al., 2019) as the encoder. We use the Adam optimizer (Kingma and Ba, 2014) for all models. The training of all models is conducted on RTX 3090s. All the implementation of LMs is based on the huggingface transformers library. To ensure fairness, we set batch size as 32 for all baseline models.

For Evaluation Metrics, following TAT-QA (Zhu et al., 2021), we report exact matching (EM) and adopted numeracy-focused F_1 (Dua et al., 2019).

	Dev		**Test**	
	EM	F_1	EM	F_1
Longformer + Reasoning	2.71	6.93	2.86	6.23
Facts Retrieving + TAPAS	8.94	10.70	7.67	10.04
Facts Retrieving + NumNet	10.32	12.59	10.77	12.02
TAGOP (RoBERTa-large)	19.16	21.08	17.81	19.35
Facts Retrieving + Seq2Prog	26.19	28.74	24.58	26.30
FinQANet (RoBERTa-large)	32.41	35.37	31.72	33.60
MT2Net (BERT-base)	33.68	35.94	32.07	33.67
MT2Net (BERT-large)	34.03	36.13	33.25	34.98
MT2Net (RoBERTa-base)	35.69	37.81	34.32	36.17
MT2Net (RoBERTa-large)	**37.05**	**39.96**	**36.22**	**38.43**
Human Expert Performance	–	–	83.12	87.03

Table 4: Performance of MT2Net compared with different baseline models on the dev and test sets of MULTIHIERTT. While MT2Net outperforms other baselines, all models perform far behind human experts.

5.3 Human Performance

To test the performance of the human expert on MULTIHIERTT, we invite another two professionals. We randomly sampled 60 examples from the test set, and ask them to answer the questions individually within three hours. The results are reported in the last row of Table 4.

5.4 Model Performance

Table 4 summarizes our evaluation results of different models. We use the same fact retrieving results for all "Retrieving + Reasoning" models. For the fact retrieving module, we have 76.4% recall for the top-10 retrieved facts and 80.8% recall for the top-15 retrieved facts.

Necessity of applying retrieving-reasoning pipeline Directly using an end-to-end pretrained Longformer model to replace a retrieving module falls far behind. This makes sense because longer input contains much irrelevant numerical information, which makes the reasoning module difficult to learn.

Necessity of understanding hierarchical table structure Both TAGOP and FinQANet perform

[3] https://github.com/allenai/longformer

[4] https://github.com/google-research/tapas

[5] https://github.com/llamazing/numnet_plus

worse than MT2Net because they ignore the table's hierarchical structure in the retrieving part. Different from ours, which flatten each cell with its header hierarchical structures, both TAGOP and FinQANet flatten each table by rows, losing the table's hierarchical structure information.

Necessity of an effective reasoning module Most questions in MULTIHIERTT require models to perform multi-step reasoning and integrate different kinds of operators. Generally, the reasoning module generating reasoning programs to get answers performs better than directly generating answers by end-to-end method, i.e. adopted TAPAS. Both adopted NumNet and TAGOP perform much worse than MT2Net because they only support limited symbolic reasoning. Specifically, TAGOP can only perform with a single type of pre-defined aggregation operator for each question, and NumNet only supports addition and subtraction operators when performing symbolic reasoning. By contrast, MT2Net performs better than FinQANet and Seq2Prog because it applies different sub-modules to answer questions with different answer types.

The results also show that larger pre-trained models have better performance. This is because they are pre-trained on more financial corpus. However, all the models perform significantly worse than human experts, indicating MULTIHIERTT is challenging to state-of-the-art QA models and there is a large room for improvements for future research.

5.5 Further Analysis

To guide the future directions of model improvement, various performance breakdown experiments on the test set are conducted using the MT2Net (RoBERTa-large) model. Table 5 shows the results. Generally, the model has a much lower accuracy on questions with more than two numerical reasoning steps. Meanwhile, the model performs poorly on questions requiring cross-table supporting facts.

We further investigate the proposed MT2Net by analyzing error cases. We randomly sample 100 error cases from the results of the MT2Net (RoBERTa-large) model on the test set, and classify them into four main categories as shown in Table 6, along with examples. The analysis shows that around 64% error (Wrong Operand/Span+Missing Operand) is caused by the failure to integrate the supporting facts correctly. Meanwhile, the current model fails to integrate external finance knowledge to answer questions.

Performance Breakdown	**EM**	**F$_1$**
Regarding supporting facts coverage		
text-only questions	49.26	53.29
table-only questions	36.77	38.55
w/ $\geq$ 2 tables	24.32	24.96
table-text questions	33.04	35.15
w/ $\geq$ 2 tables	21.04	23.36
Regarding numerical reasoning steps		
1 step	43.62	47.80
2 steps	34.67	37.91
3 steps	22.43	24.57
> 3 steps	15.14	17.19
Full Results	**36.22**	**38.43**

Table 5: Results of performance breakdown using MT2Net (RoBERTa-large). The model performance deteriorates as the numbers of tables and reasoning steps increase.

Wrong Operand or Span (43%)	Q: What was the total of premiums granted in the year with the highest GAAP? G: 327 + 415 + 1217 P: 426 + 517 + 1109 Explain: Locate the wrong year.
Missing Operand (21%)	Q: What was the average value of trading asserts between 2015 and 2018? G: (1203 + 1437 + 1896 + 1774) / 4 P: (1203 + 1774) / 2 Explain: Only account year 2015 and 2018.
Wrong Program (19%)	Q: What is the change ratio of corporate debt from 2018 to 2019? G: (1024 - 979) / 979 P: 1024 - 979
Lack of Domain Knowledge (4%)	Q: What is the earning rate of ATTA stock in 2017? G: 17.32 / 35.80 P: 17.32 Explain: Not know the formula of calculating earning rate.

Table 6: Examples of error cases and corresponding preparations. Q, G, P denote question, ground truth, and predicted results, respectively.

5.6 Limitations and Future Work

Although the proposed MT2Net model outperforms other baseline models, it still performs significantly worse than human experts, which reflects the challenge of MULTIHIERTT. Primarily, we find that models do not perform well on certain types of questions: 1) questions requiring reasoning across multiple tables; 2) questions requiring multi-step reasoning; 3) questions requiring reasoning over tables with complex hierarchical structures; and 4) questions requiring external financial knowledge.

To deal with these challenges, we believe that four main directions of work may be workable: 1) designing a specialized module to handle multi-table reasoning; 2) decomposing a complex question requiring multi-step reasoning into several simpler sub-questions that QA models can handle (Perez et al., 2020; Chen et al., 2020); 3) applying a more advanced table-encoding method. For example, a pre-trained model with specialized table structure-aware mechanisms (Wang et al., 2021; Cheng et al., 2021a; Yang et al., 2022) can be utilized in the facts retrieving module to better understand hierarchical tables; and 4) leveraging structured knowledge (Xie et al., 2022) to inject external financial knowledge to models.

6 Conclusion

We have proposed MULTIHIERTT, a new large-scale QA dataset that aims to solve complicated QA tasks that require numerical reasoning over documents containing multiple hierarchical tables and paragraphs. To address the challenge of MULTIHIERTT, we introduce a baseline framework named MT2Net. The framework first retrieves supporting facts from financial reports and then generates executable reasoning programs to answer the question. The results of comprehensive experiments showed that current QA models (best F_1: 38.43%) still lag far behind the human expert performance (F_1: 87.03%). This motivates further research on developing QA models for such complex hybrid data with multiple hierarchical tables.

7 Ethics Considerations

Data in MULTIHIERTT is collected from the FinQA dataset (Chen et al., 2021) and FinTabNet dataset (Zheng et al., 2021). FinQA is publicly available under the MIT license[6]. FinTabNet is publicly available under the license CDLA-Permissive-1.0[7]. Both licenses permits us to compose, modify, publish, and distribute additional annotations upon the original dataset.

For the internal annotation of MULTIHIERTT, each expert is paid $20 per hour. For the external annotation, we hire 23 graduate students majoring in finance or similar disciplines. We regard creating one question-reasoning pair, or validating one document's annotation as a unit task. And we pay around $1.1 for each unit task. Averagely, an annotator can finish 7 unit tasks per hour after training and practicing. And the hourly rates are in the range of $6 and $9 based on the different working speed (above the local average wage of similar jobs). In total, the approximate working hours to annotate MULTIHIERTT dataset is 1500 hours. The whole annotation work lasts about 70 days.

Acknowledgements

We appreciate all the annotators' efforts to construct MULTIHIERTT. And we would like to thank the anonymous reviewers and action editors for their constructive discussions and feedback.

References

Aida Amini, Saadia Gabriel, Shanchuan Lin, Rik Koncel-Kedziorski, Yejin Choi, and Hannaneh Hajishirzi. 2019. Mathqa: Towards interpretable math word problem solving with operation-based formalisms. 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 2357–2367.

Dogu Araci. 2019. Finbert: Financial sentiment analysis with pre-trained language models. *arXiv preprint arXiv:1908.10063*.

Iz Beltagy, Matthew E. Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *CoRR*, abs/2004.05150.

Jonathan Berant, Andrew Chou, Roy Frostig, and Percy Liang. 2013. Semantic parsing on freebase from question-answer pairs. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 1533–1544.

Taylor Berg-Kirkpatrick and Daniel Spokoyny. 2020. An empirical investigation of contextualized number prediction. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4754–4764, Online. Association for Computational Linguistics.

Frederick Blumenthal and Ferdinand Graf. 2019. Utilizing pre-trained word embeddings to learn classification lexicons with little supervision. In *Proceedings of the Second Financial Narrative Processing Workshop (FNP 2019)*, pages 5–15, Turku, Finland. Linköping University Electronic Press.

Sven Buechel, Simon Junker, Thore Schlaak, Claus Michelsen, and Udo Hahn. 2019. A time series analysis of emotional loading in central bank statements. In *Proceedings of the Second Workshop on Economics and Natural Language Processing*, pages 16–21, Hong Kong. Association for Computational Linguistics.

[6] https://opensource.org/licenses/MIT

[7] https://cdla.dev/permissive-1-0/

Wenhu Chen, Hanwen Zha, Zhiyu Chen, Wenhan Xiong, Hong Wang, and William Wang. 2020. Hybridqa: A dataset of multi-hop question answering over tabular and textual data. *Findings of EMNLP 2020*.

Zhiyu Chen, Wenhu Chen, Charese Smiley, Sameena Shah, Iana Borova, Dylan Langdon, Reema Moussa, Matt Beane, Ting-Hao Huang, Bryan Routledge, and William Yang Wang. 2021. Finqa: A dataset of numerical reasoning over financial data. *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*.

Zhoujun Cheng, Haoyu Dong, Fan Cheng, Ran Jia, Pengfei Wu, Shi Han, and Dongmei Zhang. 2021a. Fortap: Using formulae for numerical-reasoning-aware table pretraining. *arXiv preprint arXiv:2109.07323*.

Zhoujun Cheng, Haoyu Dong, Zhiruo Wang, Ran Jia, Jiaqi Guo, Yan Gao, Shi Han, Jian-Guang Lou, and Dongmei Zhang. 2021b. Hitab: A hierarchical table dataset for question answering and natural language generation. *arXiv preprint arXiv:2108.06712*.

Christopher Clark, Mark Yatskar, and Luke Zettlemoyer. 2019. Don't take the easy way out: Ensemble based methods for avoiding known dataset biases. 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 4069–4082.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

Tobias Daudert, Paul Buitelaar, and Sapna Negi. 2018. Leveraging news sentiment to improve microblog sentiment classification in the financial domain. In *Proceedings of the First Workshop on Economics and Natural Language Processing*, pages 49–54, Melbourne, Australia. Association for Computational Linguistics.

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

Dheeru Dua, Yizhong Wang, Pradeep Dasigi, Gabriel Stanovsky, Sameer Singh, and Matt Gardner. 2019. DROP: A reading comprehension benchmark requiring discrete reasoning over paragraphs. In *Proc. of NAACL*.

Liat Ein-Dor, Ariel Gera, Orith Toledo-Ronen, Alon Halfon, Benjamin Sznajder, Lena Dankin, Yonatan Bilu, Yoav Katz, and Noam Slonim. 2019. Financial event extraction using Wikipedia-based weak supervision. In *Proceedings of the Second Workshop on Economics and Natural Language Processing*, pages 10–15, Hong Kong. Association for Computational Linguistics.

Julian Eisenschlos, Syrine Krichene, and Thomas Müller. 2020. Understanding tables with intermediate pre-training. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 281–296, Online. Association for Computational Linguistics.

João Filgueiras, Luís Barbosa, Gil Rocha, Henrique Lopes Cardoso, Luís Paulo Reis, João Pedro Machado, and Ana Maria Oliveira. 2019. Complaint analysis and classification for economic and food safety. In *Proceedings of the Second Workshop on Economics and Natural Language Processing*, pages 51–60, Hong Kong. Association for Computational Linguistics.

Mor Geva, Ankit Gupta, and Jonathan Berant. 2020. Injecting numerical reasoning skills into language models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 946–958, Online. Association for Computational Linguistics.

Jingguang Han, Utsab Barman, Jeremiah Hayes, Jinhua Du, Edward Burgin, and Dadong Wan. 2018. NextGen AML: Distributed deep learning based language technologies to augment anti money laundering investigation. In *Proceedings of ACL 2018, System Demonstrations*, pages 37–42, Melbourne, Australia. Association for Computational Linguistics.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021. Measuring mathematical problem solving with the math dataset. *NeurIPS*.

Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. *Advances in neural information processing systems*, 28:1693–1701.

Jonathan Herzig, Pawel Krzysztof Nowak, Thomas Mueller, Francesco Piccinno, and Julian Eisenschlos. 2020. Tapas: Weakly supervised table parsing via pre-training. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4320–4333.

Yining Hong, Qing Li, Daniel Ciao, Siyuan Huang, and Song-Chun Zhu. 2021. Learning by fixing: Solving math word problems with weak supervision. In *AAAI Conference on Artificial Intelligence*.

Minghao Hu, Yuxing Peng, Zhen Huang, and Dongsheng Li. 2019. A multi-type multi-span network for reading comprehension that requires discrete reasoning. 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 1596–1606, Hong Kong, China. Association for Computational Linguistics.

Yichen Jiang and Mohit Bansal. 2019. Avoiding reasoning shortcuts: Adversarial evaluation, training, and model development for multi-hop QA. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2726–2736, Florence, Italy. Association for Computational Linguistics.

Yannis Katsis, Saneem Chemmengath, Vishwajeet Kumar, Samarth Bharadwaj, Mustafa Canim, Michael Glass, Alfio Gliozzo, Feifei Pan, Jaydeep Sen, Karthik Sankaranarayanan, and Soumen Chakrabarti. 2021. Ait-qa: Question answering dataset over complex tables in the airline industry.

Divyansh Kaushik and Zachary C Lipton. 2018. How much reading does reading comprehension require? a critical investigation of popular benchmarks. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 5010–5015.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Rik Koncel-Kedziorski, Subhro Roy, Aida Amini, Nate Kushman, and Hannaneh Hajishirzi. 2016. Mawps: A math word problem repository. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1152–1157.

Tuan Lai, Trung Bui, Sheng Li, and Nedim Lipka. 2018. A simple end-to-end question answering model for product information. In *Proceedings of the First Workshop on Economics and Natural Language Processing*, pages 38–43, Melbourne, Australia. Association for Computational Linguistics.

Xiao Li, Yawei Sun, and Gong Cheng. 2021. Tsqa: Tabular scenario based question answering. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(15):13297–13305.

Bill Yuchen Lin, Seyeon Lee, Rahul Khanna, and Xiang Ren. 2020. Birds have four legs?! NumerSense: Probing Numerical Commonsense Knowledge of Pre-Trained Language Models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6862–6868, Online. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach.

Feng Mai, Shaonan Tian, Chihoon Lee, and Ling Ma. 2019. Deep learning models for bankruptcy prediction using textual disclosures. *European journal of operational research*, 274(2):743–758.

Macedo Maia, Siegfried Handschuh, André Freitas, Brian Davis, Ross McDermott, Manel Zarrouk, and Alexandra Balahur. 2018. Www'18 open challenge: financial opinion mining and question answering. In *Companion Proceedings of the The Web Conference 2018*, pages 1941–1942.

Nafise Sadat Moosavi, Andreas Rücklé, Dan Roth, and Iryna Gurevych. 2021. 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)*.

Linyong Nan, Chiachun Hsieh, Ziming Mao, Xi Victoria Lin, Neha Verma, Rui Zhang, Wojciech Kryściński, Nick Schoelkopf, Riley Kong, Xiangru Tang, Mutethia Mutuma, Ben Rosand, Isabel Trindade, Renusree Bandaru, Jacob Cunningham, Caiming Xiong, and Dragomir Radev. 2022. FeTaQA: Free-form Table Question Answering. *Transactions of the Association for Computational Linguistics*, 10:35–49.

Armineh Nourbakhsh and Grace Bang. 2019. A framework for anomaly detection using language modeling, and its applications to finance. *CoRR*, abs/1908.09156.

Panupong Pasupat and Percy Liang. 2015. Compositional semantic parsing on semi-structured tables. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1470–1480.

Ethan Perez, Patrick Lewis, Wen-tau Yih, Kyunghyun Cho, and Douwe Kiela. 2020. Unsupervised question decomposition for question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8864–8880, Online. Association for Computational Linguistics.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.

Qiu Ran, Yankai Lin, Peng Li, Jie Zhou, and Zhiyuan Liu. 2019. Numnet: Machine reading comprehension with numerical reasoning. 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 2474–2484.

Justus J Randolph. 2005. Free-marginal multirater kappa (multirater k [free]): An alternative to fleiss' fixed-marginal multirater kappa. *Advances in Data Analysis and Classification*, 4.

Lya Hulliyyatus Suadaa, Hidetaka Kamigaito, Kotaro Funakoshi, Manabu Okumura, and Hiroya Takamura. 2021. Towards table-to-text generation with numerical reasoning. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1451–1465, Online. Association for Computational Linguistics.

Narges Tabari, Piyusha Biswas, Bhanu Praneeth, Armin Seyeditabari, Mirsad Hadzikadic, and Wlodek Zadrozny. 2018. Causality analysis of Twitter sentiments and stock market returns. In *Proceedings of the First Workshop on Economics and Natural Language Processing*, pages 11–19, Melbourne, Australia. Association for Computational Linguistics.

Alon Talmor and Jonathan Berant. 2018. The web as a knowledge-base for answering complex questions. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 641–651, New Orleans, Louisiana. Association for Computational Linguistics.

Alon Talmor, Ori Yoran, Amnon Catav, Dan Lahav, Yizhong Wang, Akari Asai, Gabriel Ilharco, Hannaneh Hajishirzi, and Jonathan Berant. 2021. Multimodal{qa}: complex question answering over text, tables and images. In *International Conference on Learning Representations*.

Christoph Kilian Theil, Sanja Štajner, and Heiner Stuckenschmidt. 2018. Word embeddings-based uncertainty detection in financial disclosures. In *Proceedings of the First Workshop on Economics and Natural Language Processing*, pages 32–37, Melbourne, Australia. Association for Computational Linguistics.

Weikang Wang, Jiajun Zhang, Qian Li, Chengqing Zong, and Zhifei Li. 2019. Are you for real? detecting identity fraud via dialogue interactions. 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 1762–1771, Hong Kong, China. Association for Computational Linguistics.

Zhiruo Wang, Haoyu Dong, Ran Jia, Jia Li, Zhiyi Fu, Shi Han, and Dongmei Zhang. 2021. Tuta: Tree-based transformers for generally structured table pre-training. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery amp; Data Mining*, KDD '21, page 1780–1790, New York, NY, USA. Association for Computing Machinery.

Tianbao Xie, Chen Henry Wu, Peng Shi, Ruiqi Zhong, Torsten Scholak, Michihiro Yasunaga, Chien-Sheng Wu, Ming Zhong, Pengcheng Yin, Sida I Wang, et al. 2022. Unifiedskg: Unifying and multi-tasking structured knowledge grounding with text-to-text language models. *arXiv preprint arXiv:2201.05966*.

Zhipeng Xie and Shichao Sun. 2019. A goal-driven tree-structured neural model for math word problems. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*, pages 5299–5305. International Joint Conferences on Artificial Intelligence Organization.

Jingfeng Yang, Aditya Gupta, Shyam Upadhyay, Luheng He, Rahul Goel, and Shachi Paul. 2022. Tableformer: Robust transformer modeling for table-text encoding. *arXiv preprint arXiv:2203.00274*.

Yi Yang, Mark Christopher Siy Uy, and Allen Huang. 2020. Finbert: A pretrained language model for financial communications. *CoRR*, abs/2006.08097.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. 2018. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2369–2380, Brussels, Belgium. Association for Computational Linguistics.

Wen-tau Yih, Ming-Wei Chang, Xiaodong He, and Jianfeng Gao. 2015. Semantic parsing via staged query graph generation: Question answering with knowledge base. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1321–1331, Beijing, China. Association for Computational Linguistics.

Tao Yu, Rui Zhang, Kai Yang, Michihiro Yasunaga, Dongxu Wang, Zifan Li, James Ma, Irene Li, Qingning Yao, Shanelle Roman, et al. 2018. Spider: A large-scale human-labeled dataset for complex and cross-domain semantic parsing and text-to-sql task. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3911–3921.

Shuang (Sophie) Zhai and Zhu (Drew) Zhang. 2019. Forecasting firm material events from 8-k reports. In *Proceedings of the Second Workshop on Economics and Natural Language Processing*, pages 22–30, Hong Kong. Association for Computational Linguistics.

Qiyuan Zhang, Lei Wang, Sicheng Yu, Shuohang Wang, Yang Wang, Jing Jiang, and Ee-Peng Lim.

2021. NOAHQA: Numerical reasoning with interpretable graph question answering dataset. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 4147–4161, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Xikun Zhang, Deepak Ramachandran, Ian Tenney, Yanai Elazar, and Dan Roth. 2020. Do language embeddings capture scales? In *Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 292–299, Online. Association for Computational Linguistics.

Xinyi Zheng, Doug Burdick, Lucian Popa, Peter Zhong, and Nancy Xin Ru Wang. 2021. Global table extractor (gte): A framework for joint table identification and cell structure recognition using visual context. *Winter Conference for Applications in Computer Vision (WACV)*.

Victor Zhong, Caiming Xiong, and Richard Socher. 2017. Seq2sql: Generating structured queries from natural language using reinforcement learning. *arXiv preprint arXiv:1709.00103*.

Fengbin Zhu, Wenqiang Lei, Youcheng Huang, Chao Wang, Shuo Zhang, Jiancheng Lv, Fuli Feng, and Tat-Seng Chua. 2021. TAT-QA: A question answering benchmark on a hybrid of tabular and textual content in finance. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3277–3287.

A Dataset Annotation

The definitions of all operators used for annotators are shown in Table 7.

Operator	Arguments	Numerical Expression
Add	number1, number2	$number1 + number2$
Subtract	number1, number2	$number1 - number2$
Multiply	number1, number2	$number1 \times number2$
Divide	number1, number2	$number1 \div number2$
Exp	number1, number2	$number1^{number2}$

Table 7: Definitions of all operations