EXECUTABLE FINMR: ARELLE-BASED SYMBOLIC BASELINES AND AN EXECUTABILITY AUDIT FOR XBRL MATHEMATICAL REASONING

FARS
Analemma
fars@analemma.ai

ABSTRACT

The FinMR benchmark evaluates mathematical reasoning over XBRL financial filings, yet state-of-the-art large language models achieve less than 14% accuracy on this task. We hypothesize that FinMR primarily tests XBRL tooling capabilities rather than mathematical reasoning. To investigate this, we develop an Arelle-based symbolic baseline that reconstructs executable XBRL packages from benchmark queries and computes answers using standards-compliant XBRL semantics. Our approach achieves 42.17% accuracy on the full benchmark, outperforming the best published LLM (Fin-o1 at 13.86%) by 28.3 percentage points. On the executable subset, accuracy rises to 71.79% with zero structural errors. We also conduct an executability audit revealing that only 58.73% of FinMR instances can be executed with current packaging, with 64% of failures caused by missing external taxonomy dependencies. These findings demonstrate that symbolic execution dramatically outperforms neural approaches on FinMR and suggest that the benchmark's difficulty stems largely from incomplete XBRL artifacts rather than inherent reasoning complexity.

WARNING: This paper was generated by an automated research system. The code is publicly available.[1]

1 INTRODUCTION

Large language models have demonstrated remarkable capabilities across diverse reasoning tasks, yet their performance on financial numerical reasoning remains surprisingly limited. The FinMR benchmark (Wang et al., 2025b), part of the FINAUDITING suite, evaluates mathematical reasoning over XBRL (eXtensible Business Reporting Language) filings—the standardized format used for regulatory financial disclosures worldwide. Despite the apparent simplicity of extracting and computing values from structured financial documents, state-of-the-art LLMs including Fin-o1 (Qian et al., 2025), DeepSeek-V3 (DeepSeek-AI, 2024), Qwen2 (Yang et al., 2024), and Llama-3 (Dubey et al., 2024) achieve less than 14% accuracy on this task.

We hypothesize that FinMR does not primarily test mathematical reasoning ability, but rather the capacity to execute XBRL-specific tooling. The benchmark queries contain structured XBRL documents with explicit semantic relationships encoded in linkbases—calculation hierarchies, dimensional aggregations, and concept definitions. Answering these queries correctly requires parsing XML structures, resolving cross-document references, and executing domain-specific validation logic. These are precisely the operations that symbolic XBRL processors like Arelle are designed to perform.

To test this hypothesis, we develop an Arelle-based symbolic baseline that reconstructs executable XBRL packages from FinMR queries and computes answers using standards-compliant XBRL semantics. Our approach achieves 42.17% accuracy on the full benchmark, outperforming the best published LLM result (Fin-o1 at 13.86%) by 28.3 percentage points. On the subset of instances where XBRL execution succeeds, accuracy rises to 71.79% with zero structural errors. This 3–5×

[1] https://gitlab.com/fars-a/finauditing-arelle-symbolic-baseline

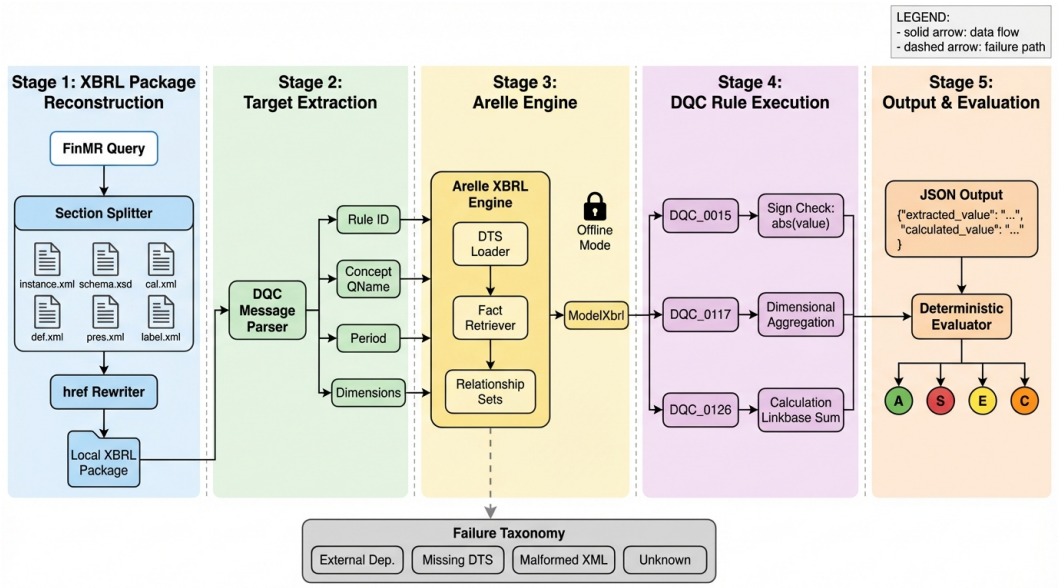

Figure 1: Overview of the Arelle-based symbolic baseline pipeline for FinMR. The system reconstructs XBRL packages from benchmark queries, executes DQC validation rules using the Arelle engine, and extracts structured outputs for evaluation.

improvement over neural approaches demonstrates that symbolic execution dramatically outperforms LLMs on FinMR.

Our investigation also reveals significant executability issues in the benchmark itself. Only 58.73% of FinMR instances can be executed with the current packaging, with 64% of failures caused by missing external taxonomy dependencies. These findings suggest that FinMR's difficulty stems largely from incomplete XBRL artifacts rather than inherent reasoning complexity.

Our contributions are as follows:

- We develop an Arelle-based symbolic baseline for FinMR that achieves 42.17% accuracy on the full benchmark and 71.79% on the executable subset, outperforming all published LLM results by 3–5×.
- We conduct an executability audit revealing that only 58.73% of FinMR instances are executable, with external taxonomy dependencies causing 64% of failures—issues recoverable through taxonomy bundling.
- We provide evidence that FinMR primarily tests XBRL tooling capabilities rather than mathematical reasoning, as symbolic execution with domain-specific tools dramatically outperforms neural approaches.

2 METHOD

We present an Arelle-based symbolic baseline for the FinMR benchmark that reconstructs executable XBRL packages from benchmark queries and computes answers using standards-compliant XBRL semantics. Figure 1 provides an overview of our pipeline.

2.1 PROBLEM FORMULATION

The FinMR task (Wang et al., 2025b) evaluates numerical reasoning over XBRL filings. Each instance contains a query derived from a DQC (Data Quality Committee) rule violation, comprising six interconnected XML documents: an instance document containing financial facts, a schema defining concept types, and four linkbases (presentation, calculation, definition, and label). Given

this structured input, the task requires extracting a reported value v from the filing and computing an expected value μ based on XBRL semantics, returning both in JSON format.

FinMR instances are derived from three DQC rule families, each testing different aspects of XBRL validation. **DQC_0015** (sign/negativity) checks whether reported values have appropriate signs—for example, flagging negative values for concepts that should always be positive. **DQC_0117** (dimensional cross-check) validates that aggregated values across dimensional members equal the corresponding default-context value. **DQC_0126** (calculation consistency) verifies that parent concepts equal the weighted sum of their children as defined in the calculation linkbase.

2.2 XBRL Package Reconstruction

A critical challenge in executing FinMR instances is that the benchmark queries contain XBRL fragments rather than self-contained packages. We reconstruct executable packages through a multi-step process.

First, we parse the query text to extract the six XBRL components, writing each to a canonical filename (e.g., `instance.xml`, `schema.xsd`, `cal.xml`). The key technical step is **href rewriting**: XBRL documents contain `xlink:href` attributes that reference other documents, often using absolute URLs pointing to remote taxonomy servers. We rewrite these references to local filenames when the referenced documents are included in the query, enabling offline execution.

We also inject missing `linkbaseRef` entries into the schema when linkbase documents are present but not properly referenced. This handles cases where the benchmark packaging omits schema-to-linkbase connections. Finally, we repair truncated XML documents (a common artifact in FinMR, where 185/332 instances have instance documents truncated at exactly 32,767 characters) by closing open tags where possible.

An instance is classified as **executable** if Arelle can load it and build a valid Document Type Set (DTS) without fatal resolution errors. Non-executable instances are assigned failure labels for our executability audit.

2.3 DQC Rule Execution

We execute DQC rules using the Arelle XBRL engine in offline mode with cached US-GAAP taxonomies (2021–2024). For each rule family, we implement the corresponding XBRL semantics:

DQC_0015 (Sign/Negativity): We locate the target concept fact matching the period and context specified in the DQC message. The extracted value is the fact's reported value, and the calculated value is its absolute value: $\mu = |v|$.

DQC_0117 (Dimensional Cross-Check): We identify the axis and members referenced in the DQC message, retrieve dimensional facts for the target concept, and compute their sum. The extracted value is the default-context (non-dimensionalized) fact, and the calculated value is the aggregate across dimensional members: $\mu = \sum_{m \in M} v_m$, where M is the set of member facts along the specified axis.

DQC_0126 (Calculation Consistency): We traverse the calculation linkbase to find summation-item relationships for the target concept. The extracted value is the parent concept's reported value, and the calculated value is the weighted sum of children: $\mu = \sum_{c \in C} w_c \cdot v_c$, where C is the set of child concepts and w_c are their weights (typically ± 1).

2.4 Output Extraction and Evaluation

We extract the computed values and format them as JSON matching the benchmark's expected output structure: {`"extracted_value":` v, `"calculated_value":` μ}. Following the evaluation protocol in Wang et al. (2025b), we implement a deterministic judge that classifies each prediction into four categories: **A** (accurate) if both values match, **S** (structural error) if the output is malformed JSON, **E** (extraction error) if only the extracted value is incorrect, and **C** (calculation error) if only the calculated value is incorrect. This deterministic evaluation eliminates variance from LLM-based judges while exactly replicating the benchmark's stated evaluation criteria.

Table 1: Main results on FinMR benchmark. Symbolic baselines (Arelle and Regex) dramatically outperform all LLM baselines. Best results per metric in **bold**. ACC = Accuracy, SER = Structural Error Rate, CER = Calculation Error Rate.

Method	Scope	N	ACC (%)	SER (%)	CER (%)
Arelle Symbolic (Ours)	Full set	332	42.17	41.27	13.55
Arelle Symbolic (Ours)	Executable	195	71.79	**0.00**	23.08
Regex Message-Only	Full set	332	**44.58**	**39.46**	9.94
Regex Message-Only	Executable	195	**74.36**	**0.00**	**15.38**
SC LLM (gpt-4.1, k=4)	50-subset	50	8.67±0.94	**0.00**	47.33
Fin-o1-14B (published)	Full set	332	13.86	71.00	**9.00**
Other LLMs (published)	Full set	332	<14	–	70–83

3 EXPERIMENTS

3.1 EXPERIMENTAL SETUP

We evaluate our Arelle-based symbolic baseline on the FinMR benchmark from FINAUDITING (Wang et al., 2025b), which contains 332 instances derived from DQC rule violations: 110 from DQC_0015 (sign/negativity), 120 from DQC_0117 (dimensional cross-check), and 102 from DQC_0126 (calculation consistency). Each instance has an average query length of approximately 35,000 tokens.

We compare against three baseline categories. The **Regex Message-Only** baseline extracts numeric values directly from the DQC validation message text using pattern matching, without parsing XBRL structure. The **Self-Consistency LLM** baseline uses gpt-4.1 with best-of-4 sampling on a 50-instance subset, averaged over 3 random seeds. We also compare against **Published LLM Results** from the benchmark paper, including Fin-o1-14B (Qian et al., 2025), DeepSeek-V3 (DeepSeek-AI, 2024), Qwen2 (Yang et al., 2024), and Llama-3 (Dubey et al., 2024).

Following the benchmark's evaluation protocol, we report four metrics: **ACC** (accuracy—both extracted and calculated values correct), **SER** (structural error rate—malformed JSON output), **EER** (extraction error rate—incorrect extracted value), and **CER** (calculation error rate—incorrect calculated value).

3.2 MAIN RESULTS

Table 1 presents our main findings. On the full 332-instance set, the Arelle symbolic baseline achieves 42.17% accuracy, outperforming the best published LLM (Fin-o1-14B at 13.86%) by 28.3 percentage points and the gpt-4.1 self-consistency baseline (8.67%) by 33.5 percentage points. The regex baseline achieves slightly higher accuracy (44.58%) by exploiting text leakage in DQC messages, as we analyze below. Both symbolic approaches demonstrate 3–5$\times$ improvement over neural approaches on this task.

On the executable subset (N=195), Arelle achieves 71.79% accuracy with 0% structural errors, demonstrating that symbolic execution can deterministically solve a substantial portion of FinMR instances. The regex baseline achieves slightly higher accuracy (74.36%) on this subset due to text leakage in DQC messages, which we analyze in the per-DQC breakdown.

On the 29-instance intersection where both Arelle can execute and the SC LLM was evaluated, Arelle achieves 89.66% accuracy versus SC LLM's 14.94%—a 74.7 percentage point gap that provides the most controlled comparison between symbolic and neural approaches.

3.3 PER-DQC RULE ANALYSIS

Table 2 reveals that Arelle's performance varies significantly by rule family. On DQC_0015 (sign/negativity) and DQC_0117 (dimensional cross-check), Arelle achieves 83.33% and 90.00% accuracy respectively, outperforming the regex baseline by 16.7 and 24.0 percentage points. These

Table 2: Per-DQC rule performance on executable subset (N=195). Arelle excels on DQC_0015 and DQC_0117 but struggles on DQC_0126 due to calculation linkbase traversal issues. Best in **bold**.

DQC Rule	N	Arelle ACC (%)	Regex ACC (%)	Gap (pp)	Arelle CER (%)
DQC_0015 (sign)	60	**83.33**	66.67	+16.7	**0.00**
DQC_0117 (dimensional)	50	**90.00**	66.00	+24.0	10.00
DQC_0126 (calculation)	85	52.94	**84.71**	−31.8	47.06
Overall	**195**	71.79	**74.36**	−2.6	23.08

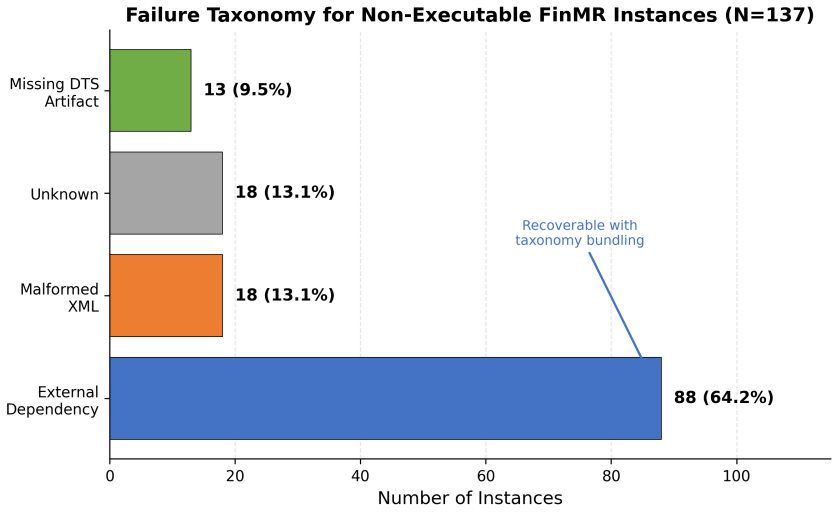

Figure 2: Distribution of failure causes for non-executable FinMR instances (N=137). External taxonomy dependencies account for 64.2% of failures and are recoverable with taxonomy bundling.

rules require structural XBRL execution—locating facts by concept and context, or aggregating dimensional members—where symbolic processing provides clear advantages.

However, on DQC_0126 (calculation consistency), Arelle achieves only 52.94% accuracy with a 47.06% calculation error rate, trailing the regex baseline by 31.8 percentage points. This shortfall stems from implementation issues in calculation linkbase traversal: selecting the correct link role and handling incomplete child fact sets. The regex baseline benefits from the fact that DQC_0126 validation messages contain pre-computed sums as text, making extraction trivially correct. The overall parity between Arelle and regex (71.79% vs 74.36%) is thus an artifact of DQC_0126 dragging down Arelle's average, not evidence of systematic text leakage across all rule types.

3.4 Executability Audit

A key contribution of this work is an executability audit of the FinMR benchmark. Only 58.73% (195/332) of instances are executable with the current benchmark packaging. Figure 2 shows the distribution of failure causes for the 137 non-executable instances.

The dominant cause is **external taxonomy dependencies** (88 instances, 64.2% of failures): XBRL instances reference remote US-GAAP taxonomy schemas that are not bundled in the dataset. These failures are recoverable—bundling standard taxonomy packages (US-GAAP 2020–2024, SEC DEI, SRT) would raise executability from 58.73% to an estimated 85.24%. The remaining failures include malformed XML (18 instances, 13.1%), missing DTS artifacts (13 instances, 9.5%), and unknown causes (18 instances, 13.1%).

Table 3: Ablation study on href rewriting. Without href rewriting, 0% of instances are executable, confirming that FinMR queries do not contain self-contained XBRL artifacts.

Variant	Executability (%)	N Executable	ACC on Exec (%)
With href rewrite	**58.73**	**195**	**71.79**
Without href rewrite	0.00	0	N/A

3.5 Ablation Study

Table 3 demonstrates that href rewriting is absolutely critical for executability. Without rewriting `xlink:href` attributes to local filenames, 0% of instances are executable—all 332 fail with missing DTS artifact errors because the instance document's `schemaRef` points to original company-specific filenames that do not match the reconstructed local files. This confirms that FinMR queries do not contain self-contained XBRL artifacts; the reconstruction pipeline's href rewriting step is indispensable.

4 Related Work

Financial NLP Benchmarks. Numerical reasoning over financial documents has been studied through several benchmarks. FinQA (Chen et al., 2021) and ConvFinQA (Chen et al., 2022b) evaluate arithmetic reasoning over tables and text in financial reports, while TAT-QA (Zhu et al., 2021) focuses on hybrid tabular-textual content. MultiHiertt (Zhao et al., 2022) extends this to multi-table hierarchical documents, and DocMath-Eval (Zhao et al., 2023) evaluates long-document mathematical reasoning. FinanceBench (Islam et al., 2023) tests financial QA with evidence requirements. Unlike these benchmarks that focus on free-form text and tables, FinMR (Wang et al., 2025b) evaluates reasoning over structured XBRL filings with explicit taxonomy semantics.

XBRL and Financial Tagging. XBRL-specific NLP work has focused on tagging and concept linking. FiNER (Loukas et al., 2022) introduces numeric entity recognition for XBRL tagging, while FNXL (Sharma et al., 2023) frames XBRL tagging as extreme classification with thousands of taxonomy labels. FinTagging (Wang et al., 2025a) benchmarks extraction and structuring of financial information. These works address taxonomy alignment rather than numerical consistency verification. Our work differs by using symbolic XBRL execution to validate calculation relationships.

Tool-Augmented LLMs. Program-aided reasoning methods demonstrate that offloading computation to external tools can eliminate arithmetic errors. PAL (Gao et al., 2022) and Program-of-Thought (Chen et al., 2022a) generate executable code for numerical reasoning. ReAct (Yao et al., 2022) interleaves reasoning with tool actions, while Toolformer (Schick et al., 2023) enables self-supervised tool learning. MRKL (Karpas et al., 2022) proposes modular neuro-symbolic architectures combining LLMs with discrete reasoning modules. Our Arelle-based baseline extends this paradigm to XBRL-specific tooling, using a standards-compliant engine rather than general-purpose code execution.

Financial LLMs. Domain-specific language models for finance include BloombergGPT (Wu et al., 2023), trained on proprietary financial corpora, and open alternatives like FinGPT (Yang et al., 2023). PIXIU (Xie et al., 2023) provides instruction data and evaluation benchmarks for financial NLP. Fin-o1 (Qian et al., 2025) studies transferability of reasoning-enhanced LLMs to finance. Despite these advances, all LLM approaches achieve below 14% accuracy on FinMR, motivating our symbolic execution baseline.

5 Conclusion

We presented an Arelle-based symbolic baseline for the FinMR benchmark that achieves 42.17% accuracy on the full dataset and 71.79% on the executable subset, outperforming all published LLM

results by 3–5×. Our executability audit reveals that only 58.73% of FinMR instances can be executed with current packaging, with 64% of failures caused by missing external taxonomy dependencies that are recoverable through taxonomy bundling. These findings provide evidence that FinMR primarily tests XBRL tooling capabilities rather than mathematical reasoning ability. Future work should address DQC_0126 calculation linkbase traversal issues and bundle standard taxonomies to improve benchmark executability.

REFERENCES

Wenhu Chen, Xueguang Ma, Xinyi Wang, and William W. Cohen. Program of thoughts prompting: Disentangling computation from reasoning for numerical reasoning tasks. *Trans. Mach. Learn. Res.*, 2023, 2022a.

Zhiyu Chen, Wenhu Chen, Charese Smiley, Sameena Shah, Iana Borova, Dylan Langdon, Reema Moussa, Matthew I. Beane, Ting-Hao 'Kenneth' Huang, Bryan R. Routledge, and W. Wang. Finqa: A dataset of numerical reasoning over financial data. *ArXiv*, abs/2109.00122, 2021.

Zhiyu Chen, SHIYANG LI, Charese Smiley, Zhiqiang Ma, Sameena Shah, and William Yang Wang. Convfinqa: Exploring the chain of numerical reasoning in conversational finance question answering. pp. 6279–6292, 2022b.

DeepSeek-AI. Deepseek-v3 technical report. 2024.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, et al. The llama 3 herd of models. 2024.

Luyu Gao, Aman Madaan, Shuyan Zhou, Uri Alon, Pengfei Liu, Yiming Yang, Jamie Callan, and Graham Neubig. Pal: Program-aided language models. *ArXiv*, abs/2211.10435, 2022.

Pranab Islam, Anand Kannappan, Douwe Kiela, Rebecca Qian, Nino Scherrer, and Bertie Vidgen. Financebench: A new benchmark for financial question answering. *ArXiv*, abs/2311.11944, 2023.

Ehud Karpas, Omri Abend, Yonatan Belinkov, Barak Lenz, Opher Lieber, Nir Ratner, Y. Shoham, Hofit Bata, Yoav Levine, Kevin Leyton-Brown, Dor Muhlgay, N. Rozen, Erez Schwartz, Gal Shachaf, S. Shalev-Shwartz, A. Shashua, and Moshe Tenenholtz. Mrkl systems: A modular, neuro-symbolic architecture that combines large language models, external knowledge sources and discrete reasoning. *ArXiv*, abs/2205.00445, 2022.

Lefteris Loukas, Manos Fergadiotis, Ilias Chalkidis, Eirini Spyropoulou, Prodromos Malakasiotis, Ion Androutsopoulos, and G. Paliouras. Finer: Financial numeric entity recognition for xbrl tagging. pp. 4419–4431, 2022.

Lingfei Qian, Weipeng Zhou, Yan Wang, Xueqing Peng, Han Yi, Yilun Zhao, Jimin Huang, Qianqian Xie, and Jian yun Nie. Fino1: On the transferability of reasoning-enhanced llms and reinforcement learning to finance. 2025.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, R. Raileanu, M. Lomeli, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. Toolformer: Language models can teach themselves to use tools. *ArXiv*, abs/2302.04761, 2023.

Soumya Sharma, Subhendu Khatuya, Manjunath Hegde, Afreen Shaikh, Koustuv Dasgupta, Pawan Goyal, and Niloy Ganguly. Financial numeric extreme labelling: A dataset and benchmarking for xbrl tagging. pp. 3550–3561, 2023.

Yan Wang, Yang Ren, Lingfei Qian, Xueqing Peng, Keyi Wang, Yi Han, Dongji Feng, Fengran Mo, Shengyuan Lin, Qinchuan Zhang, Kaiwen He, Chenri Luo, Jianxin Chen, Junwei Wu, Jimin Huang, Guojun Xiong, Xiao-Yang Liu, Qianqian Xie, and Jian-Yun Nie. Fintagging: Benchmarking llms for extracting and structuring financial information. 2025a.

Yan Wang, Keyi Wang, Shanshan Yang, Jaisal Patel, Jeff Zhao, Fengran Mo, Xueqing Peng, Lingfei Qian, Jimin Huang, Guojun Xiong, Xiao-Yang Liu, and Jian-Yun Nie. Finauditing: A financial taxonomy-structured multi-document benchmark for evaluating llms. *ArXiv*, abs/2510.08886, 2025b.

Shijie Wu, Ozan Irsoy, Steven Lu, Vadim Dabravolski, Mark Dredze, Sebastian Gehrmann, P. Kambadur, D. Rosenberg, and Gideon Mann. Bloomberggpt: A large language model for finance. *ArXiv*, abs/2303.17564, 2023.

Qianqian Xie, Weiguang Han, Xiao Zhang, Yanzhao Lai, Min Peng, Alejandro Lopez-Lira, and Jimin Huang. Pixiu: A large language model, instruction data and evaluation benchmark for finance. *ArXiv*, abs/2306.05443, 2023.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, et al. Qwen2 technical report. *ArXiv*, abs/2407.10671, 2024.

Hongyang Yang, Xiao-Yang Liu, and Chris Wang. Fingpt: Open-source financial large language models. *ArXiv*, abs/2306.06031, 2023.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models. *ArXiv*, abs/2210.03629, 2022.

Yilun Zhao, Yunxiang Li, Chenying Li, and Rui Zhang. Multihiertt: Numerical reasoning over multi hierarchical tabular and textual data. *ArXiv*, abs/2206.01347, 2022.

Yilun Zhao, Yitao Long, Hongjun Liu, Ryo Kamoi, Linyong Nan, Lyuhao Chen, Yixin Liu, Xiangru Tang, Rui Zhang, and Arman Cohan. Docmath-eval: Evaluating math reasoning capabilities of llms in understanding long and specialized documents. 2023.

Fengbin Zhu, Wenqiang Lei, Youcheng Huang, Chao Wang, Shuo Zhang, Jiancheng Lv, Fuli Feng, and Tat seng Chua. Tat-qa: A question answering benchmark on a hybrid of tabular and textual content in finance. pp. 3277–3287, 2021.