Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

SPM-RAG: Structure-Preserving Multimodal RAG for Question Answering over Turkish Integrated Reports

Erdem Nezihi Aksu
Gebze Technical University
Saliha Yesilyurt
s.yesilyurt@gtu.edu.tr

Gebze Technical University
Sibel Aydemir
Fatih Sultan Mehmet Waqf University
Yakup Genc
Gebze Technical University
Yusuf Sinan Akgul
Gebze Technical University

Research Article

Keywords:

Posted Date: November 23rd, 2026

DOI: https://doi.org/10.21203/rs.3.rs-11730482/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

SPM-RAG: Structure-Preserving Multimodal RAG for Question Answering over Turkish Integrated Reports

Erdem Nezihi Aksu[1], Saliha Yesilyurt[1,3*], Sibel Aydemir[2], Yakup Genc[1], Yusuf Sinan Akgul[1]

[1*]Department of Computer Engineering, Gebze Technical University, Kocaeli, Turkiye.
[2]Department of Management Information Systems, Fatih Sultan Mehmet Vakıf University, Istanbul, Turkiye.
[3]Department of Computer Engineering, Sivas Cumhuriyet University, Sivas, Turkiye.

*Corresponding author(s). E-mail(s): s.yesilyurt@gtu.edu.tr
Contributing authors: e.aksu2021@gtu.edu.tr; sdaydemir@fsm.edu.tr; yakup.genc@gtu.edu.tr; akgul@gtu.edu.tr

Abstract

Integrated reports combine financial and sustainability disclosures in long, unstandardized documents, and much of their content sits in high-density tables, charts and multi-column layouts. The International Integrated Reporting Framework, which sets out the guiding principles for these reports worldwide, expects them to be balanced and free from material error, a reliability strengthened by internal audit and independent external assurance. In Turkiye, this expectation carries regulatory weight: sustainability reporting is mandatory for large entities and subject to assurance audits. Auditors, standard setters and investors therefore need to find specific facts in these reports reliably. Conventional retrieval-augmented generation (RAG) falls short, because it flattens tables and layouts into plain text, cutting rows off from their headers and passages off from their sections. We present a structure-preserving multimodal RAG (SPM-RAG) method for question answering over Turkish integrated reports. SPM-RAG splits each report into a structured text stream and a stream of extracted tables and charts, and queries both in parallel. A three-phase chunking algorithm keeps every table intact with its column headers and every chunk inside its section hierarchy, while the visual stream encodes only the extracted tables and charts,

not whole pages, as multi-vector embeddings. On 46 Turkish integrated reports and 1,380 question–answer pairs, scored by a large language model acting as judge, SPM-RAG reaches 0.82 Context Recall, 0.82 Context Precision and 0.71 Answer Accuracy, 14 percentage points above the strongest baseline. Because every answer is traceable to its evidence, SPM-RAG offers a practical basis for verifying sustainability disclosures at scale.

Keywords: Retrieval-Augmented Generation, Structure-Preserving Chunking, Multimodal Retrieval, Integrated Reports, Document Question Answering, Turkish Natural Language Processing

1 Introduction

Companies today are asked to account for more than their financial results. They are also expected to report on their environmental, social and governance activities. The Global Reporting Initiative (GRI), whose standards are the most widely used for sustainability reporting [1], goes so far as to state that profitability and shareholder returns do not capture an organization's true economic impact [2]. Integrated reports answer this expectation by bringing financial and sustainability information together in a single document, and the International <IR> Framework asks that they be balanced, free from material error and not manipulated in any way [3, paras. 3.39, 3.44]. In Turkiye, the matter has also moved into regulation: since 2024, sustainability reporting has been mandatory for large entities and banks, integrated reports fall explicitly within its scope [4], and the reports are subject to assurance audits [5, 6]. Standard setters and auditors must therefore monitor these reports, track how firms progress over the years and answer audit questions from the reports themselves. Doing this manually is challenging and, across many reports and years, nearly impossible. Integrated reports have no standard format, since the Framework is principles-based and prescribes none [3, para. 4.2]. They often run beyond 150 pages, and much of their information sits in tables, charts and multi-column layouts. The Framework itself notes that information and communication technology can be used to improve the ability to search, access, combine, connect and analyze report content [3, para. 3.9].

Retrieval-Augmented Generation (RAG) grounds the answers of Large Language Models (LLMs) in external, verifiable sources [7, 8]. Text-based RAG, however, struggles with complex documents that contain tables, charts, multi-column layouts and content spanning page boundaries [9]: linearizing a two-dimensional layout into one-dimensional text destroys row–column relationships and separates content from its section context. ColPali [10] instead encodes page images directly as multi-vector visual representations; MinerU [11] converts PDFs into structured Markdown with extracted visual elements; and RAG-Anything [12] builds a dual-layer knowledge graph at a substantial construction cost. Work on corporate reports has mainly addressed financial filings and English documents, for example indicator extraction with GPT-4 [13] and text-only RAG over 10-K filings [14]. Turkish RAG studies have used plain-text collections such as Wikipedia [15], Turkish document-parsing benchmarks focus on OCR

accuracy [16], and Turkish retrieval embeddings are evaluated on plain-text pairs [17]. To our knowledge, question answering over Turkish integrated reports has not been addressed.

We first applied the established RAG designs to this problem and found that none of them solves it. Fixed-size chunking (Naive RAG) splits tables from their headers; hierarchy-aware recursive splitting (Dynamic RAG) keeps sections together but still works on text in which the parser has already flattened the tables; and full-page visual retrieval (Vision RAG) loses small tables and figures in the visual noise of dense pages. This paper presents a structure-preserving multimodal RAG (SPM-RAG) method designed to remove these failures. SPM-RAG rests on two ideas: *structure-preserving text chunking*, a three-phase algorithm that keeps tables intact with their headers and keeps every chunk within its section hierarchy, and *targeted visual encoding*, which encodes only the tables and charts extracted from the report instead of whole pages. The two streams are queried in parallel and merged into one context. Our contributions are (1) SPM-RAG, a method for question answering over Turkish integrated reports, with a formal description of its components; (2) an evaluation on 46 Turkish integrated reports and 1,380 question–answer pairs with an LLM-as-a-Judge protocol [18]; and (3) a comparison with Naive, Dynamic and Vision RAG, supported by a per-document analysis and a case study, that shows which failure each component of SPM-RAG removes.

2 Why Existing RAG Designs Fail on Integrated Reports

Naive RAG converts the PDF into a linear text stream with a standard parser and cuts it into fixed-size chunks of 1000 characters. This causes *syntactic fragmentation* (P1): a cut leaves one chunk with the column headers of a table and the next with "headerless" data rows that the LLM cannot interpret. It also causes *contextual dissociation* (P2): a section header such as "2.1 Financial Risks" is separated from its body text, so that a chunk is coherent on its own but its topic is unknown.

Dynamic RAG replaces fixed-size cutting with recursive segmentation along the document hierarchy. With T the input text, L_{max} the target chunk size and $\mathcal{S} = \{s_1, \ldots, s_n\}$ an ordered set of separators from coarse to fine (section headers, paragraphs, sentences), the segmentation is

$$\Phi(T, i) = \begin{cases} \{T\} & \text{if } |T| \leq L_{max}, \\ \bigcup_{t \in \text{split}(T, s_i)} \Phi(t, i+1) & \text{otherwise,} \end{cases} \tag{1}$$

started from $\Phi(T, 1)$, where $\text{split}(T, s_i)$ divides T at every occurrence of s_i. This reduces P2, but Dynamic RAG still operates on the linearized stream: once the parser has flattened a table, no splitting rule can restore it, so P1 remains.

Vision RAG avoids parsing altogether. Each page is rendered as an image and encoded by ColPali [10], which splits the image into patches and produces one embedding per patch, so that a page is represented as a *multi-vector* set. A query $Q = \{q_1, \ldots, q_m\}$ is scored against the patch embeddings $V = \{v_1, \ldots, v_p\}$ of a page

with MaxSim:

$$S_{vis}(Q, V) = \sum_{i=1}^{m} \max_{j=1}^{p} (q_i \cdot v_j). \tag{2}$$

A page of an integrated report, however, typically mixes narrative text, photographs and several tables. When the whole page is encoded, the small table or data point that holds the answer is lost among unrelated content. We call this *granularity mismatch* (P3).

3 The Proposed Method: SPM-RAG

SPM-RAG is designed to solve P1–P3 together (Figure 1). MinerU [11] first decomposes each report into a Markdown text stream D_{md}, in which narrative text is kept as Markdown and tables as raw HTML, and an image stream $\mathcal{I}_{ext}$ of extracted visual elements such as tables and charts. The text stream is processed by a three-phase chunking algorithm (Section 3.1) and the image stream by a targeted visual encoder (Section 3.2); at query time both are searched in parallel (Section 3.3).

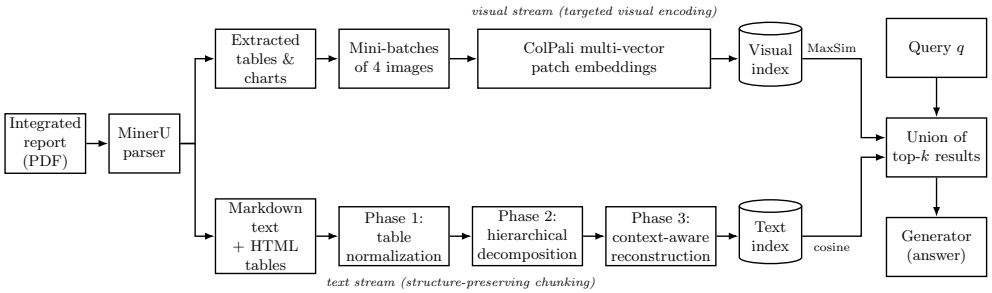


Fig. 1 Overview of SPM-RAG. MinerU splits the report into a text stream and a stream of extracted tables and charts. The text stream is chunked by the three-phase structure-preserving algorithm; the visual elements are encoded as ColPali multi-vector embeddings. Both indexes are searched in parallel and their top-k results are merged. *TikZ code generated by Claude (Anthropic), author-verified.*

3.1 Structure-Preserving Text Chunking

Phase 1: Table normalization (solves P1).

The algorithm scans the stream S for HTML tables. A table that exceeds a token threshold μ is decomposed into sub-tables $T_i = H_{table} \cup R_i$, $i \in \{1, \ldots, k\}$, where R_i is a subset of data rows and the column headers H_{table} are replicated in every sub-table, so that no data row is ever separated from its headers. A protection function then replaces each table by an atomic placeholder token and stores its HTML in a key–value store $\mathcal{M}$:

$$\Psi(s_i) = \begin{cases} \phi_i, \quad \mathcal{M}[\phi_i] \leftarrow s_i & \text{if } s_i \text{ is a table,} \\ s_i & \text{otherwise.} \end{cases} \tag{3}$$

Because a placeholder is a single token, the later phases cannot cut through a table.

Phase 2: Hierarchical decomposition (solves P2).

The normalized stream is segmented in three steps. A *topological split* cuts it at Markdown headers and attaches to each segment its hierarchical path (e.g., "Section 3 > Subsection 3.2"). An *isolation split* cuts at placeholder boundaries, so that every block is either pure narrative text or a single table token ϕ; tables and prose are never mixed in one block. Finally, a *recursive refinement* applies the recursive splitter of Equation (1) to purely textual blocks that still exceed the length limit. The hierarchy-aware splitting idea of Dynamic RAG is thus used inside SPM-RAG, but only after tables have been protected.

Phase 3: Context-aware reconstruction (keeps P2 solved without redundancy).

To avoid fragmented context, segments $\{s_1, \ldots, s_n\}$ are greedily merged into retrieval chunks. Repeating the section header before every segment would waste the context window, so the header $H(s_i)$ is injected only when it differs from the header of the last added segment $H(s_{last})$ or when a new chunk C starts. The cost of adding s_i is

$$\Delta(s_i) = \begin{cases} \text{len}^*(s_i) + |H(s_i)| & \text{if } H(s_i) \neq H(s_{last}) \text{ or } C = \emptyset, \\ \text{len}^*(s_i) & \text{otherwise,} \end{cases} \tag{4}$$

where $\text{len}^*(s_i)$ is the length after placeholders are resolved through $\mathcal{M}$. Segment s_i is merged into C if $|C| + \Delta(s_i) \leq L_{max}$; otherwise C is closed and a new chunk starts with s_i. Placeholders are then replaced by their HTML. Every chunk therefore carries its section context, and every table appears complete and labeled.

3.2 Targeted Visual Encoding (solves P3)

Some content of integrated reports, such as charts, infographics and tables with complex layouts, is not fully captured even by careful text extraction. SPM-RAG therefore keeps a visual stream, but changes its unit of encoding: instead of whole pages, it encodes only the tables and charts that MinerU has extracted. Each extracted element is encoded by ColPali into patch embeddings and scored with MaxSim (Equation (2)). Because the encoded image contains little besides the element itself, the query is matched against the region that holds the answer rather than against a whole page. The ColPali visual retrieval idea of Vision RAG is thus used inside SPM-RAG, but at the granularity of individual elements.

3.3 Parallel Retrieval and Fusion

The two streams are queried in parallel. With $\mathcal{R}_{text}(q, k_{text})$ the top text chunks and $\mathcal{R}_{vis}(q, k_{vis})$ the top visual elements, the context passed to the generator is

$$C_{\text{SPM-RAG}} = \mathcal{R}_{text}(q, k_{text}) \cup \mathcal{R}_{vis}(q, k_{vis}). \tag{5}$$

The generator thus receives both the hierarchically intact text, including complete HTML tables, and the visual element most relevant to the query.

4 Experimental Evaluation

4.1 Setup

A synthetic "Golden Test Set" of 1,380 question–answer pairs was generated with Gemini 2.5 Pro from 46 Turkish integrated reports that present financial, environmental, social and governance indicators through a mix of text, tables, charts and photographs. 70% of the questions require data from tables or charts and 30% are text-based. All experiments ran on a single NVIDIA A100 GPU; the 4-bit quantized Qwen3-VL-32B-Instruct model served as both generator and judge, and `ytu-ce-cosmos/turkish-e5-large` provided the text embeddings. SPM-RAG retrieved $k_{text} = 3$ text chunks and $k_{vis} = 1$ visual element per query. The baselines of Section 2 used the same generator, judge and embedding model: Naive RAG used 1000-character chunks with a 200-character overlap, Dynamic RAG chunks of up to 1000 tokens, both with the top three chunks, and Vision RAG the top page. Following the LLM-as-a-Judge paradigm [18], which reaches over 80% agreement with human preferences, the judge scored each answer from 0 to 1, with a written justification, on **Context Recall** (the retrieved context contains the correct answer), **Context Precision** (the context is relevant and free of noise) and **Answer Accuracy** (the final answer is correct).

4.2 Results

Table 1 reports the results. SPM-RAG achieves the highest score on all three metrics: it outperforms the strongest baseline, Dynamic RAG, by 5 percentage points in Context Recall, 4 percentage points in Context Precision and 14 percentage points in Answer Accuracy. A per-document breakdown shows that the advantage holds across the 46 reports: the accuracy of SPM-RAG lies mostly between 0.6 and 0.9, whereas Vision RAG varies strongly, with several reports below 0.2.

Table 1 Results on the Golden Test Set

Method	Context Recall	Context Precision	Answer Accuracy
Naive RAG	0.75	0.78	0.54
Dynamic RAG	0.77	0.78	0.57
Vision RAG	0.54	0.70	0.47
SPM-RAG (ours)	**0.82**	**0.82**	**0.71**

Naive and Dynamic RAG reach competitive Context Recall (0.75–0.77) but much lower Answer Accuracy (0.54–0.57): they retrieve the right words, but in a stream where columns are merged, headers and footers are interleaved and tables have lost their row and column context (P1). Hierarchy-aware splitting alone improves accuracy

by only 3 percentage points, because it cannot repair a table that the parser has already flattened. Vision RAG scores lowest on every metric, as expected from the granularity mismatch (P3). SPM-RAG removes both failures: table normalization and context-aware reconstruction give the generator complete, labeled tables, and targeted visual encoding removes the page noise, which largely explains the 24-percentage-point gap in Answer Accuracy (0.47 → 0.71) between the two visually grounded methods.

4.3 Case Study

For the *Adana Çimento 2019 Integrated Report*, the query *"What proportion of the employees is covered by the collective bargaining agreement?"* has the ground truth "70%, i.e., 225 employees". Naive and Dynamic RAG retrieved the sentence that states the 70% ratio and answered with this ratio, which the judge accepted. Their contexts, however, show the effect of linearization: in the Dynamic RAG context the headcount appears only as an unlabeled number sequence ("2019 70 225 225") mixed with chart residue. Vision RAG retrieved an unrelated page of occupational health and safety photographs and could not answer (scores 0.00, 0.20 and 0.00). SPM-RAG retrieved the normalized HTML table on the distribution of personnel covered by the agreement, in which the ratio and the headcount appear with their row labels, and gave the complete answer with full scores.

5 Discussion

Much of the information in integrated reports is given in tables and charts rather than in plain text. Our results show that this information can be reached systematically, even when it sits in a table on page 150 of a long report.

These results can benefit national and international standard setters. GRI may want to search sustainability reports for the answers to questions related to its standards, for example whether and how organizations report the disclosures a standard requires. Turkiye's Public Oversight, Accounting and Auditing Standards Authority (KGK) may want to do the same during sustainability audits, now that reports prepared under the Turkish Sustainability Reporting Standards (TSRS) are subject to assurance [5, 6]. Instead of reading long reports one by one, SPM-RAG makes it possible to ask such questions across many reports and years and to check quickly whether the required information is present.

SPM-RAG may also support the reliability of integrated reports. The International <IR> Framework requires reports to include all material matters, positive and negative, in a balanced way and without material error, and notes that reliability is enhanced by internal audit and external assurance [3, paras. 3.39, 3.40]. Verification requires that each piece of information can be traced to its source. In SPM-RAG, every answer is produced from retrieved evidence, that is, the text chunks, structured tables and table or chart images on which it is based, so a reviewer sees not only the answer but also where in the report it comes from. This makes SPM-RAG useful for preparers checking their reports before publication, as well as for auditors and regulators.

5.1 Limitations and future research

First, the test questions were generated automatically by an LLM from the reports themselves. They suit a controlled comparison but may not reflect the questions of standard setters, auditors or investors. Future research can define questions together with standard setters, for example based on GRI Standards or the requirements KGK checks in sustainability audits, and can use the evidence-based nature of RAG to compare claims with the evidence in the same report and in the organization's earlier reports, in line with the Framework's call to report against previously reported targets [3, para. 3.45]. Second, the advantage of SPM-RAG would be shown more conclusively under equal retrieval budgets: in our setup it received four context items, the text baselines three and Vision RAG one. Its highest Context Precision, despite the larger context, nevertheless suggests that the gain does not stem from additional context alone. The Dynamic and Vision RAG baselines already indicate how each idea performs on its own, but a full ablation of the two streams of SPM-RAG, a matched-budget comparison, other languages and adaptive k_{text}/k_{vis} are left to future work.

6 Conclusion

Established RAG designs fail on Turkish integrated reports: linearized chunking separates tables from their headers, and full-page retrieval loses the answer region in visual noise. SPM-RAG solves these problems with a three-phase chunking algorithm that keeps tables intact and every chunk within its section hierarchy, and a visual stream that encodes only the extracted tables and charts. On 46 reports and 1,380 question–answer pairs it reached an Answer Accuracy of 0.71, 14 percentage points above the strongest baseline. Because its answers are traceable to the retrieved evidence, SPM-RAG offers standard setters and auditors a practical path to verifying sustainability disclosures at scale.

Acknowledgements. This paper is an extended version of the four-page study presented at the 34th IEEE Signal Processing and Communications Applications Conference (SIU 2026). E.N.A. completed his graduation project on this study under Y.S.A.'s supervision.

Declarations

Funding: TÜBİTAK 3005 Programme, Grant No. 324K183 ("Do Integrated Reports Disclosed in Turkiye Display the Signals of Greenwashing? A Deep Learning-Based Visual and Textual Analysis"). **Conflict of interest:** None declared. **Ethics approval, consent to participate/publish, and materials availability:** Not applicable. **Data and code availability:** Source reports are publicly available corporate disclosures; curated QA pairs and scripts are available from the corresponding author on reasonable request. **Author contribution:** E.N.A. designed and implemented the pipelines and conducted the experiments. Y.S.A. and S.Y. supervised the research. Y.G. contributed to the evaluation framework. S.A. contributed the business and sustainability-reporting perspective and drafted the Discussion section. All authors approved the final manuscript. **Use of AI assistance:** During the

preparation of this work, the authors used Claude (Anthropic) to improve the language of the manuscript and to generate the TikZ code of Figure 1, and Gemini (Google) and ChatGPT (OpenAI) to check reference details. After using these tools, the authors reviewed and edited the content as needed and take full responsibility for the content of the publication.

References

[1] Global Reporting Initiative (GRI): GRI is top choice for sustainability reporting worldwide. https://www.globalreporting.org/news/news-center/gri-is-top-choice-for-sustainability-reporting-worldwide/. Accessed 2 October 2026

[2] Global Reporting Initiative (GRI): Topic Standards Project for Economic Impact. https://www.globalreporting.org/standards/standards-development/topic-standards-project-for-economic-impact/. Accessed 8 October 2026

[3] International Integrated Reporting Council (IIRC): International <IR> Framework (January 2021). IIRC, London (2021)

[4] Public Oversight, Accounting and Auditing Standards Authority (KGK): Türkiye Sürdürülebilirlik Raporlama Standartları (TSRS) Uygulama Kapsamına İlişkin Kurul Kararı [Board Decision on the Scope of Application of the Turkish Sustainability Reporting Standards]. Decision No. 75935942-050.01.04-[01/21634], 27 December 2023. Official Gazette of the Republic of Turkiye, No. 32414 (Duplicate), 29 December 2023, pp. 419–423 (2023)

[5] Public Oversight, Accounting and Auditing Standards Authority (KGK): Sürdürülebilirlik Denetimi Yönetmeliği [Regulation on Sustainability Audit]. Official Gazette of the Republic of Turkiye, No. 32785, 17 January 2025 (2025)

[6] Public Oversight, Accounting and Auditing Standards Authority (KGK): Sürdürülebilirlik Denetimi Yönetmeliğindeki Bazı Muafiyet Tarihlerine İlişkin Değişiklik [Amendment on certain exemption dates in the Regulation on Sustainability Audit]. https://www.kgk.gov.tr/Portalv2Uploads/files/Duyurular/v2/Diger/Duyuru%20-%2057_.pdf. Announcement No. 2025-57, 30 September 2025. Accessed 8 October 2026 (2025)

[7] Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W.-t., Rocktäschel, T., Riedel, S., Kiela, D.: Retrieval-augmented generation for knowledge-intensive NLP tasks. In: Advances in Neural Information Processing Systems (NeurIPS), vol. 33, pp. 9459–9474 (2020)

[8] Gao, Y., Xiong, Y., Gao, X., Jia, K., Pan, J., Bi, Y., Dai, Y., Sun, J., Wang, M., Wang, H.: Retrieval-augmented generation for large language models: A survey. arXiv preprint arXiv:2312.10997 (2024)

[9] Tripathi, V., Odapally, T., Das, I., Allu, U., Ahmed, B.: Vision-guided chunking

is all you need: Enhancing RAG with multimodal document understanding. arXiv preprint arXiv:2506.16035 (2025)

[10] Faysse, M., Sibille, H., Wu, T., Omrani, B., Viaud, G., Hudelot, C., Colombo, P.: ColPali: Efficient document retrieval with vision language models. In: Proceedings of the International Conference on Learning Representations (ICLR) (2025)

[11] Wang, B., Xu, C., Zhao, X., Ouyang, L., Wu, F., Zhao, Z., Xu, R., Liu, K., Qu, Y., Shang, F., Zhang, B., Wei, L., Sui, Z., Li, W., Shi, B., Qiao, Y., Lin, D., He, C.: MinerU: An open-source solution for precise document content extraction. arXiv preprint arXiv:2409.18839 (2024)

[12] Guo, Z., Ren, X., Xu, L., Zhang, J., Huang, C.: RAG-Anything: All-in-one RAG framework. arXiv preprint arXiv:2510.12323 (2025)

[13] Gomes Ziegler, G.: Automating information extraction from financial reports using LLMs: A comparative study of text, image, and multimodal approaches. Master's thesis, Aalto University, Espoo (2024)

[14] Cheng, Z., Lai, L., Liu, Y., Cheng, K., Qi, X.: Enhancing financial report question-answering: A retrieval-augmented generation system with reranking analysis. arXiv preprint arXiv:2603.16877. Accepted at ICECET 2026 (2026)

[15] Köse, S.K., Baytekin, M.C., Aktaş, B., Görür, B.K., Munis, E.A., Yılmaz, D., Kartal, M.Y., Toraman, Ç.: RAGTurk: Best practices for retrieval augmented generation in Turkish. In: Proceedings of the Second Workshop on Natural Language Processing for Turkic Languages (SIGTURK 2026), pp. 179–196 (2026). https://doi.org/10.18653/v1/2026.sigturk-1.15

[16] Yılmaz, D., Munis, E.A., Toraman, Ç., Köse, S.K., Aktaş, B., Baytekin, M.C., Görür, B.K.: OCRTurk: A comprehensive OCR benchmark for Turkish. In: Proceedings of the Second Workshop on Natural Language Processing for Turkic Languages (SIGTURK 2026), pp. 197–208 (2026). https://doi.org/10.18653/v1/2026.sigturk-1.16

[17] Ezerceli, Ö., Gümüşçekiçci, G., Erkoç, T., Özenç, B.: TurkEmbed4Retrieval: Turkish embedding model for retrieval task. In: Proceedings of the 33rd IEEE Signal Processing and Communications Applications Conference (SIU), pp. 1–4 (2025). https://doi.org/10.1109/SIU66497.2025.11112338

[18] Zheng, L., Chiang, W.-L., Sheng, Y., Zhuang, S., Wu, Z., Zhuang, Y., Lin, Z., Li, Z., Li, D., Xing, E.P., Zhang, H., Gonzalez, J.E., Stoica, I.: Judging LLM-as-a-judge with MT-Bench and Chatbot Arena. In: Advances in Neural Information Processing Systems (NeurIPS), vol. 36, pp. 46595–46623 (2023)