地方議会の予算表を対象とした LLM による表形式変換を用いた RAG の提案

前多陸玖[1]　木村泰知[1]
[1] 小樽商科大学
g2021344@edu.otaru-uc.ac.jp　kimura@res.otaru-uc.ac.jp

概要

本研究では，MBLink（Mineutes-to-Budget-Linking）で提供されている小樽市の令和 4 年度の議会会議録と予算表に対して，表の含まれる文書の前処理において，マルチモーダル LLM(M-LLM) を使用することによる，検索における精度への影響を検証する．また，embedding model 間の性能差や Chunk Size，Overlap Size の値の影響を検証する．

1　はじめに

近年，大規模言語モデルにおける追加学習なしに外部知識を扱う手法として，RAG (Retrieval-Augmented Generation) が用いられている [1]．RAG では外部文書をベクトルに変換し，データベースに保存することで，検索を可能にしている．これによりハルシネーションのリスクを軽減させ，信頼性の高い出力を期待できる．一方で，RAG で使用する文書に HTML や表・画像等が含まれている場合，出力の精度が低下する可能性があり，適切な前処理が必要である [2, 3]．またベクトルインデックス作成の際に使用する embedding model や Chunk Size，Overlap Size が出力に影響を与える [4]．

RAG の活用が見込まれる議会会議録や有価証券報告書・学術論文には，その文書内に多くの表が含まれており，それらに対する適切な前処理が求められる．本研究では，MBLink（Mineutes-to-Budget-Linking）で提供されている小樽市の令和 4 年度の議会会議録と予算表に対して，表の含まれる文書の前処理において，マルチモーダル LLM(M-LLM) を使用することによる，検索における精度への影響を検証する．また，embedding model 間の性能差や Chunk Size，Overlap Size の値の影響を検証する．

本研究の貢献は，以下の 3 点である．

- M-LLM を用いた表構造理解の手法を提案した．
- HTML 形式で記述された表と，M-LLM を通してマークダウン形式に変換した表のそれぞれをデータベースとして，RAG を適用した場合にどのような差があるのかを検証した．
- 複数の embedding model や chunck Size, Overlap Size において，それぞれの差を比較した．

2　関連研究

機械判読を目的とした表の分類の関連研究には，有価証券報告書を対象とした分類がある [5, 6]．この研究では，有価証券報告書に含まれる機械判読が困難な表のセル分類を目的として行った．その中で，有価証券報告書内の機械判読が困難な表を「小見出し行」「複数の Header（セル結合を含む）」「空白セル」「非スカラー値」「特殊な形」の 5 つのタイプに分類した．それらを含んだデータセットに対して一般的な機械学習手法を用いたアプローチを行うことで，有価証券報告書における表を対象としたセル分類の困難性を実証した．

3　対象データ

3.1　MBLink とは

MBLink は NTCIR-17 QA-Lab PoliInfo-4[1] のサブタスクである．議会の予算審議において，ある予算に関する発言に対して，関連する予算表のセルを紐づけるタスクである [7][8]．NTCIR-17 は，NII が主催する評価型ワークショップであり，2022 年 7 月から 2023 年 12 月まで開催された[2]．MBLink で使用されるデータセットは，議会会議録中の市長の発言文と予算説明書などに含まれる表，それぞれの HTML からなり，GitHub 上で公開されている[3]．図 1 の発言

1） https://sites.google.com/view/poliinfo4
2） https://research.nii.ac.jp/ntcir/ntcir-17/
3） https://github.com/poliinfo4/PoliInfo4-FormalRun-Minutes-to-Budget-Linking

This work is licensed by the author(s) under CC BY 4.0
(https://creativecommons.org/licenses/by/4.0/).

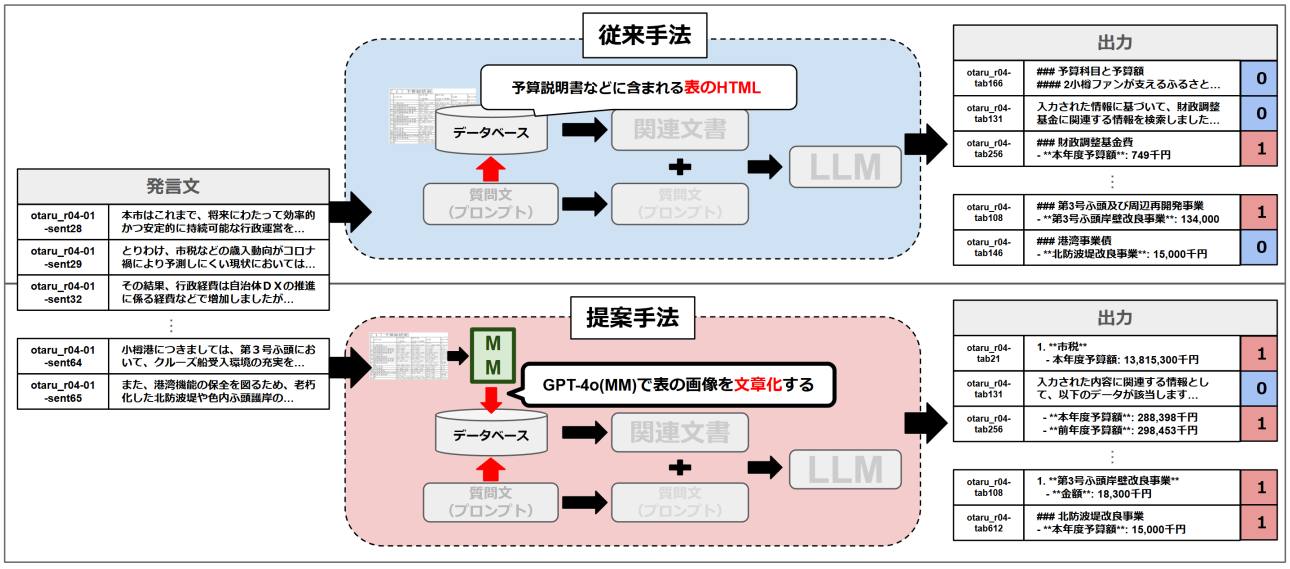

図 1　マルチモーダル LLM（MM）用いた RAG の流れ

文からわかるように，MBLink において HTML にはそれぞれ識別用の ID が割り振られており，入力されたテキストに対して関連した複数の表を紐づけている．本研究では，MBLink のデータセットの中でも，小樽市の令和 4 年度の議会会議録から作成された HTML を使用する．Sentence ID の付与された議会会議録のテキストと，そのテキストに対して正解の Table ID の付与されている表の HTML をのそれぞれ 46 個を対象とする．

3.2　GPT-4o を用いた表の前処理

本研究では，表の前処理において OpenAI が開発した M-LLM である GPT-4o（gpt-4o-2024-05-13）を使用する．図 2 は表の前処理の具体例である．GPT-4o を用いて表をマークダウン形式に変換する上では，各 Table ID ごとに対象となる表の画像が必要となる．まず，MBLink に含まれる令和 4 年度の予算説明書は一つの HTML に収められているため，table タグごとに HTML を分割する．その後対象となる表かどうかを判別し，使用する 46 個の HTML を作成した．

また表の HTML を分割する作業と並行して，HTML のスタイルシート（CSS）を一部書き換えた．CSS を変更したことにより，表の罫線が二重線から単純な直線へと置換される．これにより M-LLM による画像認識の精度を大きく向上した．

Table ID ごとに分割された HTML に対して，Selenium を用いて HTML をブラウザ上で開いたのち，表のスクリーンショットを撮影した．また HTML の分割に際して，表がページを跨ぐ場合，表のヘッダー情報が欠損することがある．このような場合には，対応するヘッダーを持つ表からヘッダー部分をキャプチャし，ヘッダー画像と表画像の両方を GPT-4o に入力した．

表の画像を GPT-4o に入力し，マークダウンに近い形式に変換する際に GPT-4o の出力トークンの問題から，表のすべての要素を出力することができなかった場合には，プロンプトに前回の出力を含めて入力することで，表の続きを生成した．本研究の目的は，表の HTML に対して，M-LLM を用いて前処理を行ったマークダウン形式のテキストデータと，未処理の HTML データのどちらが検索の精度が高いかを明らかにする．またデータベースをベクトルに変換する際に，どのような embedding model，Chunck Size，Overlap Size が適切なのか，比較・検証を行う．RAG は Python のライブラリである LangChain を用いて作成する．

4　実験設定

4.1　提案手法の検証実験の設定

提案手法の検証においては，embedding model として OpenAI embedding（text-embedding-ada-002）を用いる．未処理の表の HTML を OpenAI embedding を通したベクトルデータベースと，マークダウン形式に変換した表のテキストを OpenAI embedding を通したベクトルデータベースを作成する．作成したデータベースに対して，議会会議録のテキストを

This work is licensed by the author(s) under CC BY 4.0
(https://creativecommons.org/licenses/by/4.0/).

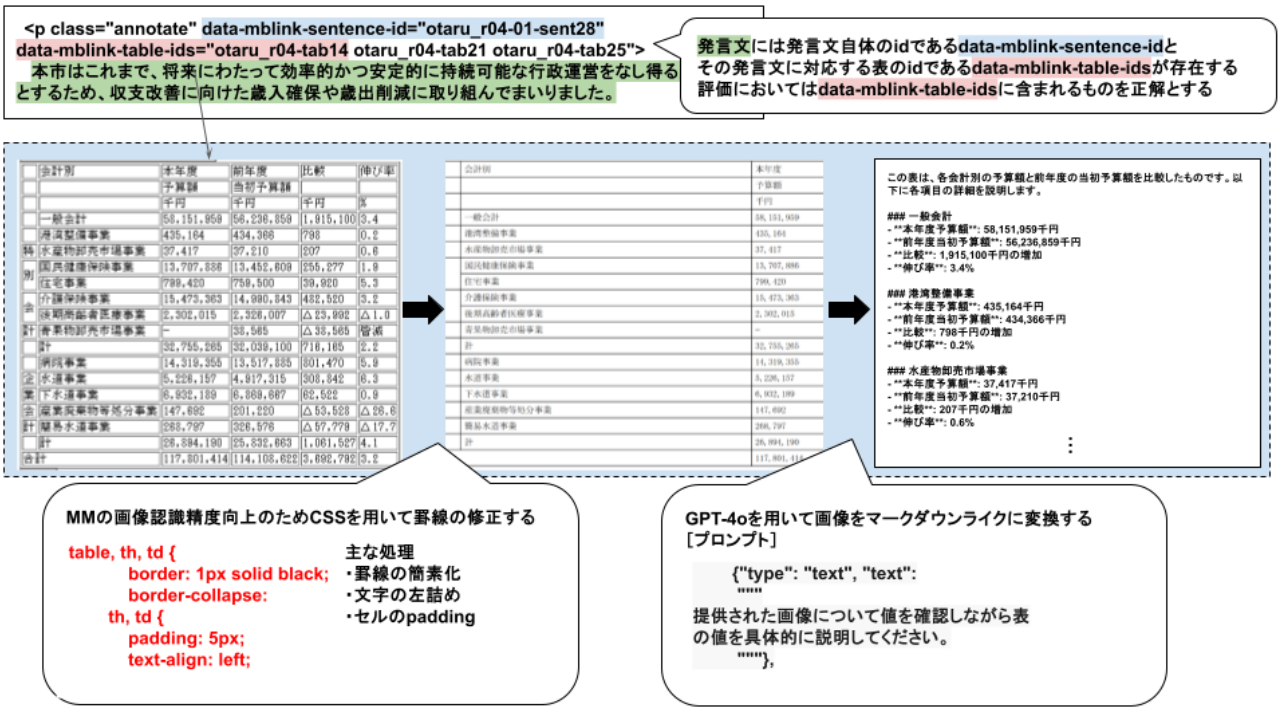

図 2　表の前処理の流れ

入力として，ベクトル検索を行い，類似度の高い上位 3 件の表を LLM に渡し回答を生成する．この際 chunk Size は 200 から 1000 までを 200 区切りで求め，Overlap Size は設定しない．回答を生成する LLM には gpt-4o-2024-05-13 を用いる．

4.2　モデル・パラメータ検証実験の設定

embedding model・パラメータ検証には, text-embedding-ada002, text-embedding-3-large, text-embedding-3-small を用いる．M-LLM を用いて前処理を行った表のテキストデータを，各 embedding model に通してベクトルデータベースを作成する．作成したデータベースに対して，議会会議録のテキストを入力として，ベクトル検索を行い，類似度の高い上位 3 件の表を LLM に渡し回答を生成する．この際 chunk Size は 200 から 1000 までを 200 区切りで求め，Overlap Size は chunk Size に対して 0%, 25%，50%で求める．

4.3　評価方法

どちらの実験においても，正解率を用いる．文書検索時に参照した上位 3 件の表 id の中に，正解となる表 id が正解とする．参照した表 id と正解の表 id に一致しているものがある場合は正解，ない場合は不正解として扱われる．

5　実験結果

5.1　提案手法の実験結果

表 1 は従来手法と提案手法の正解率を比較した表である．提案手法ではどの chunk Size においても 55% 近い正解率を算出したのに対して，従来手法ではどの chunk Size においても正解率は 41.35% であった．また chunk Size が 600，800 の場合において正解率が 56.52% と最大で 15% 以上の性能改善が見られた．

表 1　提案手法の検証実験

chunk Size	提案手法	従来手法
200	54.35%	41.30%
400	52.17%	41.30%
600	56.52%	41.30%
800	56.52%	41.30%
1000	54.35%	41.30%

5.2　モデル・パラメータ検証の実験結果

各 chunk Size，Overlap Size ごとの text-embedding-ada002, text-embedding-3-small, text-embedding-3-large の結果を表 2 に示す．正解率が最も高かったのは，embedding model が text-embedding-3-small で overlap

This work is licensed by the author(s) under CC BY 4.0
(https://creativecommons.org/licenses/by/4.0/).

表 2　各モデル・パラメータ間の比較

model	text-embedding-ada-002			text-embedding-3-large			text-embedding-3-small		
chunkSize	overlap Size0%	overlap Size25%	overlap Size50%	overlap Size0%	overlap Size25%	overlap Size50%	overlap Size0%	overlap Size25%	overlap Size50%
200	54.35%	**56.52%**	47.83%	45.63%	45.65%	41.30%	58.70%	**63.04%**	56.52%
400	52.17%	50.00%	50.00%	47.83%	43.48%	**52.17%**	52.17%	56.52%	58.70%
600	**56.52%**	43.38%	47.83%	41.30%	36.29%	32.61%	56.52%	50.00%	50.00%
800	**56.52%**	50.00%	47.83%	45.65%	43.48%	34.78%	54.35%	50.00%	50.00%
1000	54.35%	50.00%	50.00%	45.65%	47.83%	39.13%	52.17%	54.35%	56.52%

Size が 25%，chunk Size が 200 の時で 63.04% であった．

6　考察

表 3 は，提案手法と従来手法における発言文ごとの正解・不正解の分類である．提案手法と従来手法の中でどの chunk Size においても，正解を検索出来ていた場合には「すべてのケースで正解」のラベルを，正解を検索できなかった場合には「すべてのケースで不正解」のラベルを付けた．．ラベル付けにおいては重複を含めずに行った．

提案手法のみに正解があったケースは 13 件あり，従来手法のみに正解があったケースは 1 件あることがわかる．そのことから，ケースごとに見ても提案手法が従来手法と比較して優れていることがわかる．

表 3　提案手法と従来手法における正解の分類

分類	合計
全てのケースで正解	12
全てのケースで不正解	14
提案手法のみ全て正解	5
従来手法のみ全て正解	1
提案手法のみ一部正解	8
従来手法のみ一部正解	0
どちらのケースでも正解	6

図 3 は提案手法と従来手法における chunk の違いである．提案手法では header とそれに対応する値が横並びになっているのに対し，従来では header とそれに対応する値が離れていることがわかる．これにより，検索・生成の精度に違いが生じていると考える．

表 4 は，embedding model によるケースごとの正解の分類である．chunk Size が 200，Overlap Size が 0% 時における，text-embedding-ada002，text-embedding-3-small，text-embedding-3-large ごとの正解をまとめた．それぞれのケースにおいて，各 embedding model のみが正解であった場合と，不正解であった場合を計測した．結果，text-embedding-3-small が最も正解が多く，不正解が少ないことがわかった．

表 4　embedding model における正解の分類

分類	合計
全てのケースで正解	13
全てのケースで不正解	10
ada-002 のみ正解	3
ada-002 のみ不正解	2
3-small のみ正解	4
3-small のみ不正解	2
3-large のみ正解	4
3-large のみ不正解	8

7　おわりに

本研究では，表の前処理に MM を用いてマークダウン形式に変換することによる精度向上について検証した．従来手法と提案手法では，正解率が最大で 15%以上向上することがわかった．また embedding model が text-embedding-3-small の際に，最も正解が多く，chunk size は 200 が適切であることがわかった．Overlap Size は検索に大きな影響を与えないことがわかった．

謝辞

本研究は JSPS 科研費 21H03769 の助成を受けたものである．

This work is licensed by the author(s) under CC BY 4.0
(https://creativecommons.org/licenses/by/4.0/).

参考文献

[1] Patrick S. H. Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive NLP tasks. **CoRR**, Vol. abs/2005.11401, , 2020.

[2] Yunhu Ye, Binyuan Hui, Min Yang, Binhua Li, Fei Huang, and Yongbin Li. Large language models are versatile decomposers: Decompose evidence and questions for table-based reasoning, 2023.

[3] Fei Wang, Kexuan Sun, Muhao Chen, Jay Pujara, and Pedro Szekely. Retrieving complex tables with multi-granular graph representation learning. In **Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval**, SIGIR '21, p. 1472–1482. ACM, July 2021.

[4] 阿部晃弥, 新納浩幸. Rag における小説データベースの chunk size と overlap size と embedding モデルの効果. 言語処理学会 第 30 回年次大会 発表論文集, Vol. 2024, pp. 3215–3220, 2024.

[5] 前多陸玖, 奥山和樹, 佐藤栄作, 木村泰知. 有価証券報告書を対象とした機械判読が困難な表のセル分類に向けて. 人工知能学会全国大会論文集, Vol. JSAI2024, pp. 3M5OS12b03–3M5OS12b03, 2024.

[6] 奥山和樹, 木村泰知. 有価証券報告書を対象とした機械判読が困難な表構造の分析. 言語処理学会第 30 回年次大会 (NLP2024), pp. P3–20–, 3 2024.

[7] 木村泰知, 梶縁, 乙武北斗, 門脇一真, 佐々木稔, 小林暁雄. 議会会議録と予算表を紐づける minutes-to-budget linking タスクの提案. 言語処理学会 第 29 回年次大会 発表論文集, Vol. 2023, pp. 2427–2431, 2023.

[8] 小川泰弘, 木村泰知, 渋木英潔, 乙武北斗, 内田ゆず, 高丸圭一, 門脇一真, 秋葉友良, 佐々木稔, 小林暁雄. Ntcir-17 qa lab-poliinfo-4 のタスク設計. 言語処理学会 第 29 回年次大会 発表論文集, Vol. 2023, pp. 611–615, 2023.

This work is licensed by the author(s) under CC BY 4.0
(https://creativecommons.org/licenses/by/4.0/).

A　付録

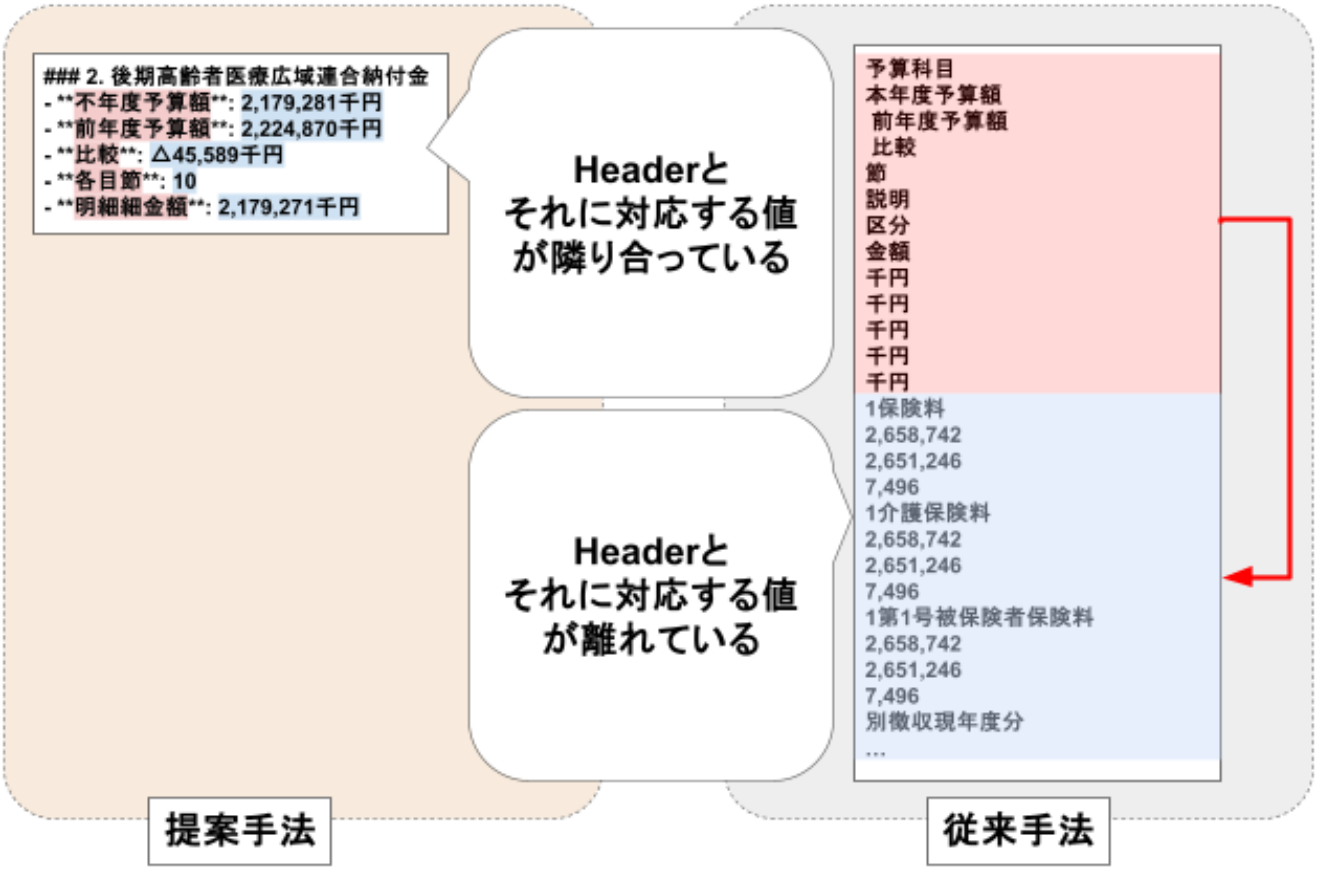

図 3　提案手法と従来手法における chunk の違い

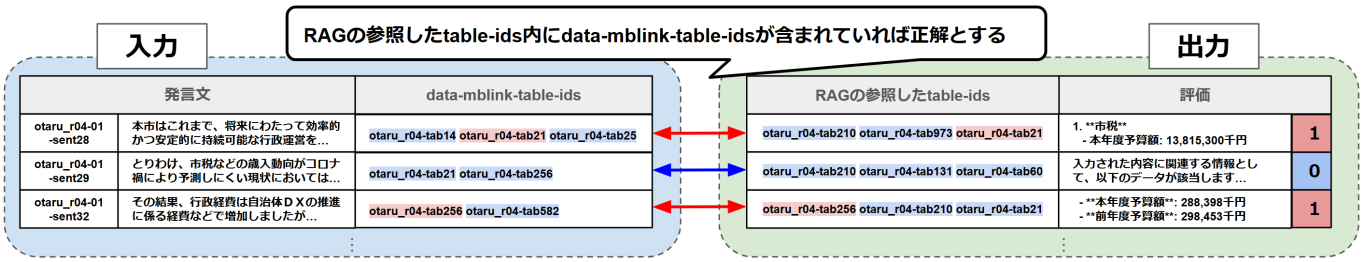

図 4　正解率算出の例

This work is licensed by the author(s) under CC BY 4.0
(https://creativecommons.org/licenses/by/4.0/).