Large Vision-Language Model を用いた非構造ドキュメント画像向け情報抽出

江上 尚志[1] 中田 百科[1] 福地 鈴佳[2] 久保田 茉莉花[2] 山根 大輝[1] 薬師寺 政和[1]
[1]株式会社 リクルート [2]株式会社 ビーンズラボ
{takashi_egami, hyakka_nakada, daiki_yamane, masakazu_yakushiji}@r.recruit.co.jp
{fukuji, kubota}@beanslabo.co.jp

概要

本研究では，非構造ドキュメント画像の情報抽出に取り組む．光学文字認識（OCR）の結果に対して大規模言語モデルを適用することで高精度な情報抽出が行えることが報告されているが，文書画像は一般に文字だけでなく，図形や文字の色などを含んだレイアウトも用いて情報を表現している．そのため OCR によって文字情報に劣化させる方法に比べ，画像を直接扱う Large Vision-Language Model の方がより高精度に抽出できる可能性がある．そこで，非構造ドキュメントとしてメニュー画像を対象とし，この方法で情報抽出を行い従来手法と比較した．

1 はじめに

光学文字認識（Optical Character Recognition, OCR）は画像から文字を抽出する技術であり，文書のデジタル化や請求書の読み取り[1]などの効率化に寄与している．例えば，飲食店のメニュー画像を元にメニュー情報を掲載サイトへ入稿する作業は多くの工数がかかるが，OCR を適用することで効率化できうる．しかし，請求書の読み取りや入稿では OCR の結果の羅列だけでは不十分であり，検出した単語の関係性を整理し情報を抽出することが必要である．以下では，OCR 結果を OCR テキストと呼び，OCR 結果に含まれる単語の座標を OCR 座標と呼ぶ．

請求書の読み取りにおいては OCR と大規模言語モデル（Large Language Model, LLM）を組み合わせると精度良く情報抽出できることが報告されている[1]．また，一般的に表構造の乏しいメニュー画像においても LLM を活用することで高精度な抽出ができることが報告されている[2,3]が，これらの研究は料理名と価格のみを抽出していた．一方で，実際のメニュー画像には図 1 のようにカテゴリや料理に付随する説明文も含まれる．そこで本研究では，料理名と価格に加えてカテゴリと料理の説明文も抽出する．例えば図 1 では「自慢の逸品」カテゴリに対して「もつ煮:900 円」，「揚出し豆腐:600 円」，「唐揚げ小:600 円:揚げたてサクサク！」,「唐揚げ大:200 円:揚げたてサクサク！」を抽出する．つまり，従来手法に比べて抽出する項目が増えるため，生成すべき構造が複雑になる．多くのメニュー画像ではカテゴリと料理の対応関係はボックスなどの図形や文字の色で区別されることが多く，OCR テキストに含まれる文字情報のみでは高精度な抽出が難しい．

従来の LLM は言語のみを入力としていたが，視覚基盤モデル[4]と統合することで，言語と視覚を統合して扱う Large Vision-Language Model（LVLM）が台頭し，これにより画像とテキストの両方を高精度に認識できるようになった[5]．そのため，LVLM を用いることで OCR を介さずに直接メニュー画像を入力でき，従来の LLM では難しい図形や文字の色で区別されるカテゴリを高精度に抽出できると期待される．そこで，本研究では，カテゴリや説明文を持ったメニュー画像を含むデータを対象に，OCR を用いない LVLM による抽出に取り組んだ．OCR と LLM を用いる従来手法[2]と比較することで，LVLM の有効性を検証する．なお，従来手法では横書きのメニュー画像のみを対象にしていたが，実際のメニ

図 1 カテゴリを含んだメニュー画像例

ュー画像には縦書きも多く存在する．そのため本研究では縦書き画像も含めて検証する．

2 従来手法

従来手法として，中田らの手法[2]を説明する．この手法は，OCR 座標と単語を用いてプロンプトを生成し，LLM で料理名と価格を抽出する．座標をプロンプトに加えることで位置関係を踏まえた抽出を可能としている．具体的には，まず，メニュー画像に OCR を適用し OCR テキストを得る．このうち OCR 座標を量子化し，単語ごとの行数・列数を求める．座標が近い単語同士は結合し，数字のみからなる単語についてはその座標を用いて税抜・税込の判定を行い，その情報を OCR テキストに付与する．以上の結果をまとめてプロンプトを作成し，LLM による推論を行う．このとき出力の形式は<料理名>:<税抜価格>:<税込価格>であり，カテゴリと説明文が含まれていないことに注意されたい．また，図 1 のようにメニュー画像には一部情報の省略が多く，料理名と価格に多対一あるいは一対多の関係がある場合があるため，そのような省略を適切に補った教師データにより LLM をファインチューニングしている．

3 手法

3.1 問題設定

本研究では，メニュー画像における情報抽出を，カテゴリ・料理名・税抜価格・税込価格・説明文の出力とし，その出力形式を以下のように定める．

出力形式

<カテゴリ名>
<料理名>:<税抜価格>:<税込価格>:<説明文>
<料理名>:<税抜価格>:<税込価格>:<説明文>
<カテゴリ名>
…

次に，表 1 に示す評価指標の概要を説明する．従来手法[2]では料理名と価格の対応関係に着目していたため，これらのペアの一致率を指標としていた．しかし，本研究ではカテゴリ・料理名・価格・説明文とより多くの項目が存在するため，それぞれに対して単語一致率を定義する．ここで，抽出された単語が正解したかどうかの判定を正誤判定とする．OCR の誤字を考慮し，完全一致ではなく類似度が閾値以上で正解とした．実際の入稿においては，誤字脱字を人の手で修正する必要があるため，単語一致率とは別に，誤字脱字の割合を評価する文字一致率を定義した．また，記載順に結果を出力することが望ましいため，カテゴリ・料理名については並び順の良さを評価する並び順距離という指標を定義した．

次に，正誤判定と各指標の詳細を述べる．正誤判定では，正解文字列に対して予測文字列群のうちジャロウィンクラー類似度が最も高い組み合わせを探し，その類似度が 0.8 以上の場合に正解と判定する．ここで，出力形式は文字列であり，カテゴリを示す行と料理名や価格，説明文を示す行に分けられる．カテゴリの正誤判定では，出力形式からカテゴリを示す行を取り出し，それらの各行を正解文字列群および予測文字列群として用いて判定する．料理名の正誤判定では，カテゴリとは独立して判定するために，各料理名が属するカテゴリ情報は無視した上で，出力形式から料理名を示す部分のみを取り出して判定する．価格の正誤判定では，料理名が正解した行同士から価格を取り出し判定する．このとき従来手法と同様に，税込価格の記載があればそれを税込価格，無ければ税抜価格を 1.1 倍したものを税込価格として判定に用いる．説明文の正誤判定では，価格と同様に，料理名が正解した行同士から説明文を取り出し判定する．正誤判定の結果を式(1)に代入することで平均値を算出し，単語一致率を求める．

$$\text{単語一致率} = \frac{\text{正解した文字列数}}{\text{正解文字列群の数}} \tag{1}$$

次に，並び順距離について説明する．カテゴリに対する並び順距離は，カテゴリ名をそれぞれユニークな文字に置き換えたときの文字列同士の編集距離により求める．具体的には，まず正解した予測カテゴリ群と対応する正解カテゴリ群を求め，それぞれのカテゴリ名にユニークな文字を割り当てる．これにより予測カテゴリの並び順を表す文字列と正解カテゴリの並び順を表す文字列を得る．この文字列同士の編集距離を求め正解カテゴリ数で割り平均値を算出することで，カテゴリに対する並び順距離を求める．例えば，図 1 では正解カテゴリ群内の並び順は「とりあえず」→「定番」である．ここで，予測カテゴリ群内の順番が「定番」→「とりあえず」だったとすると，それぞれの並び順は AB,BA とできる．よって，これらの編集距離を求め，並び順距離は 1 と算出される．料理名に対する並び順距離は，同一カテゴリ内での並び替えの量を指標としたいため，同一カテゴリに含まれる料理名を元に計算する．つまり，同一カテゴリに含まれる正解した料理名か

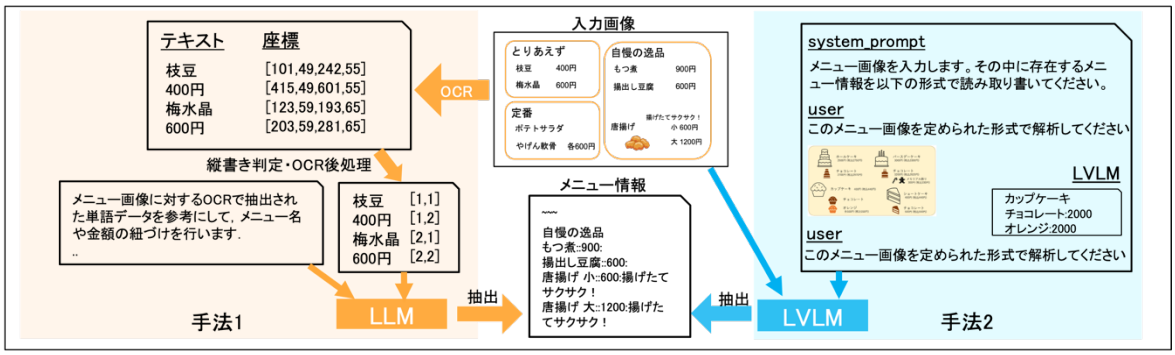

図 1　OCR と LLM 用いた抽出方法（左）と LVLM を用いた抽出方法（右）

ら，予測料理名群と対応する正解料理名を求め，これを用いて上記と同様に並び順距離を計算する．

最後に，文字一致率について説明する．それぞれの項目に対して，正解したときの予測文字列と正解文字列の間の編集距離を両者の文字列長の最大値で割ることで，標準化した編集距離を求める．正解した文字列数でこの平均値を算出することで以下の式(2)により文字一致率を求める．

$$\text{文字一致率} = 1 - \frac{\text{標準化した編集距離の総和}}{\text{正解した文字列数}} \quad (2)$$

表 1　評価指標に対する平均操作の対象

	単語一致率	並び順距離	文字一致率
カテゴリ	全カテゴリ	正解したカテゴリ	正解したカテゴリ
料理名	全料理名	同一カテゴリの正解した料理名	正解した料理名
価格	正解した料理名の価格	定義なし	定義なし
説明文	正解した料理名の説明文		正解した説明文

3.2 提案手法

3.2.1 手法 1:OCR と LLM を用いた抽出

3.1 節の抽出を既存の LLM で行うため，中田らの従来手法を拡張した．図 2（左）に示したように，従来手法に従い OCR テキストからプロンプトを作成する．加えて，縦書きへの対応及びカテゴリと説明文を読み取るために以下の拡張を行った．まず，OCR の文字単位の座標を用いて単語の向きを判定し，画像全体で多数決をとることで画像の向きを分類した．次に，横書きの場合は従来手法を適用し，縦書きの場合は縦横を交換した OCR 座標に適用した．また，カテゴリと説明文の抽出のため，3.1 節に定義した出力形式でアノテーションした教師データを用いて LLM をファインチューニングした．

3.2.1 手法 2:LVLM を用いた抽出

図 2（右）に示した，OCR を用いない LVLM による抽出方法について説明する．まず，タスクの説明を含めたシステムプロンプトを作成する．このときメニュー画像から読み取るべき内容と出力形式の説明をプロンプトに含めた．これに加えて，メニュー画像を base64 でエンコードしたものをプロンプトとして入力した．また，推論精度を高めるために Few-shot と呼ばれる例を与えた．具体的には，図 1 にある唐揚げ小と大のように一対多の関係を持つ横書き画像やカテゴリが多い複雑な横書き画像とそれらに対応する出力を与えた．

4 実験

4.1 使用データ

メニュー画像 492 件を 3.1 節で定義した出力形式に従いアノテーションした．このうち，横書き画像 50 枚と縦書き画像 20 枚を評価データとし，残りの横書き画像 342 枚と縦書き画像 80 枚の計 422 件を手法 1 の教師データとして使用した．手法 2 における Few-shot に用いる例は，教師データから選択した．

4.2 実験条件

手法 1 において，従来手法 [1]と同様に OCR には Google の Cloud Vision API[6]を用いた．LLM には Google の gemini-1.0-pro-002[7]を用いた．また Google Vertex AI [8]を用いて 4.1 節の教師データでファインチューニングを行った．エポック数は 4，学習率の乗数は 1，アダプタサイズは 1 とし，いずれもデフォルト値を用いた．推論時のパラメータのうち

temperature は出力形式からの揺らぎを抑えるために 0 として，max_output_tokens は 4192，top_p は 0.8，top_k は 40，candidate_count は 1 とした．手法 2 では LVLM に OpenAI の gpt-4o-2024-05-13[5]を使用し，max_tokens を 4095 に，top_p，temperature をそれぞれ事前検証で定性的な精度が良かった値に設定し，他はデフォルト値とした．具体的には，temperature を 0.2，top_p は 0.95 とした．

4.3 結果と考察

4.1 節の条件のもと，手法 1,2 をそれぞれ評価した．その結果を表 2 に示す．まず，横書きの結果について述べる．カテゴリにおいて，単語一致率は手法 2 が 0.875 と手法 1 より 1.5 倍高い．並び順距離と文字一致率においても，手法 2 が優れていた．これは，カテゴリが色や図などの文字以外の情報で表現されることが多く，LVLM は画像特徴量を使えるため高精度になったと考えられる．一方，料理名の単語一致率は手法 2 が手法 1 に比べて 0.79 倍と低く，実際に，手法 2 の出力には認識漏れが多く見られた．料理名は一般にメニュー画像に多く含まれる傾向があり，網羅は困難であるため，手法 2 では精度が低くとどまったと考えられる．対照的に，手法 1 では料理名を直接含んだ OCR テキストを利用し，さらにファインチューニングされているため，料理名を高精度に抽出できたと推察される．並び順距離と文字一致率は両手法とも高い精度を示した．価格の単語一致率に関しては，手法 2 が高い値を示したものの，前提となる料理名の単語一致率が低い．そのため，料理名と価格がともに正解した割合を計算すると，手法 1 が 0.49，手法 2 が 0.46 と同程度であった．説明文では，単語一致率は両手法とも 0.21～0.26 と低い．文字一致率では手法 1 が高く，手法 2 は 0.83 と低い値を示した．観察の結果，手法 2 では説明文に対して本文中に記載のない内容を出力するハルシネーションが多く見受けられた．これは，説明文に小さな文字が多く，手法 2 ではそれらを正確に認識できず，周辺の情報から補完しようとした結果，ハルシネーションが発生したと考えられる．

次に，縦書きの結果について述べる．全体的に，両手法とも横書きより精度が悪化している．横書きと同様に，手法 1 に比べ手法 2 の単語一致率は，カテゴリでは高いが料理名では低い．両手法ともに，並び順距離は横書きに比べて大きく劣化した．これは，教師データおよび Few-shot に横書きのデータが多く含まれており，縦書きの場合の順序を適切に捉えられていないためと考えられる．縦書きの教師データを増やすことや，画像に応じて縦書きの例を Few-shot に用いることで，精度の向上が期待される．価格についても，両手法ともに精度が劣化した．説明文に関して，単語一致率は手法 1 が大きく精度を下げた一方で，手法 2 は向上している．手法 2 の説明文の文字一致率は 0.64 と横書きと同様に低い値となった．

表 2 各手法の評価結果

	評価指標	横書き			縦書き		
手法 1 LLM (gemini)		単語一致率	並び順距離	文字一致率	単語一致率	並び順距離	文字一致率
	カテゴリ	0.584	0.280	0.875	0.443	**0.370**	0.971
	料理名	**0.788**	0.073	**0.930**	**0.707**	**0.209**	0.913
	価格	0.621	--	--	**0.525**	--	--
	説明文	0.216	--	**0.917**	0.075	--	**0.750**
手法 2 LVLM (gpt-4o)	カテゴリ	**0.875**	**0.09**	**0.965**	**0.607**	0.486	**0.975**
	料理名	0.625	**0.043**	0.916	0.425	0.230	**0.928**
	価格	**0.734**	--	--	0.450	--	--
	説明文	**0.260**	--	0.833	**0.644**	--	0.643

5 おわりに

本研究では，非構造ドキュメント画像としてメニュー画像の情報抽出を対象に，従来手法を拡張し料理名と価格に加えてカテゴリと料理の説明文の抽出を行った．文字情報のみでは高品質な抽出が難しいカテゴリを含む非構造ドキュメント画像に対して，OCR を用いない LVLM による抽出方法を構築した．従来手法と比較して，LVLM はカテゴリの理解に優れることを示したが，一方で料理名の抽出では従来手法が高精度であることがわかった．

今後の課題として，縦書きのメニュー画像に対する精度改善と，LVLM に OCR テキストと画像を与える抽出方法を構築することが挙げられる．また，本研究では LVLM を用いた方法においては， Few-shot を用いているが，ファインチューニングは行っていない．現段階でもいくつかの指標では従来手法に匹敵する精度を達成しているため，今後さらにファインチューニングを行い精度改善に取り組んでいきたい．

参考文献

[1] Vincent Perot, et al., “LMDX: Language Model-based Document Information Extraction and Localization.”, arXiv preprint, arXiv: 2309.10952, 2023.

[2] 中田 百科ほか, “非構造ドキュメント画像向け OCR テキスト解析のための進化計算に基づく自動プロンプトエンジニアリング”, 人工知能学会第 38 回全国大会, 2024.

[3] 江上 尚志ほか, “大規模言語モデルチューニングによる非構造化ドキュメント画像向け OCR テキスト解析”, 人工知能学会第 38 回全国大会, 2024.

[4] Alec Radford, et al., Learning transferable visual models from natural language supervision. In International conference on machine learning, pp. 8748–8763. PMLR, 2021.

[5] OpenAI, “GPT-4o System Card.”, https://cdn.openai.com/gpt-4o-system-card.pdf, 2024, （閲覧日 2024-09-20）.

[6] “Cloud Vision API”, https://cloud.google.com/vision/, （閲覧日 2024-09-20）

[7] Team, Gemini, et al., “Gemini: a family of highly capable multimodal models.”, arXiv preprint arXiv: 2312.11805, 2023.

[8] “Train custom ML models”, https://cloud.google.com/vertex-ai?hl=en#train-custom-ml-models, （閲覧日 2024-09-20）