読書支援システムにおける レイアウト解析及びLLMを用いた機能開発

小林 大晟[†‡]

†株式会社想隆社 〒169-0051 東京都新宿区西早稲田 1-22-3
早稲田大学 19-3 号館　アントレプレナーシップセンター内

‡上智大学理工学研究科〒102－8554　東京都千代田区紀尾井町 7－1

E-mail:　†‡t-kobayashi@eagle.sophia.ac.jp

あらまし　株式会社想隆社では視覚障害者のための読書支援システム「YourEyes」を開発・運用している．利用者がスキャンした書籍のレイアウト解析によって書籍を確かに所持していることを認証した後に，書籍の読み上げ情報を提供する．これまで書籍画像をルールベースで解析していたが AI を利用しより汎用なレイアウトへの対応を目指した．また利用者の UX 向上を目的とし，大規模言語モデルを用いた機能開発として OCR 読み取りミス整形，書籍画像内の挿絵・写真検出とキャプショニング，書籍内容要約機能を開発し，利用者のアンケートによる評価を行った．

キーワード　勾配ブースティング，形態素解析，t-SNE，レイアウト解析，大規模言語モデル，OCR，文章要約，物体検出，イメージキャプショニング

Development of Features using Layout Analysis and LLM in a Reading Support System

Taisei KOBAYASHI[†‡]

†Soryu-sha, 1-22-3 Nishi-Waseda, Shinjuku-ku, Tokyo 169-0051

Inside the Entrepreneurship Center, Waseda University Building No. 19-3

‡Faculty of Science and Technology, Sophia University, 7-1 Kioicho, Chiyoda-ku, Tokyo 102-8554

E-mail:　†‡t-kobayashi@eagle.sophia.ac.jp

Abstract Soryu-sha develops and operates the reading support system "YourEyes" for visually impaired individuals. After authenticating that the user indeed possesses the book through the analysis of the layout of the scanned book, the system provides the information for reading the book aloud. Previously, book images were analyzed based on rules, but now the aim is to support more general layouts by utilizing AI. In addition, with the goal of improving user experience (UX), features developed using Large Language Models (LLMs) include correction of OCR reading errors, detection and captioning of illustrations and photographs within book images, and a book content summarization function. User surveys have been conducted to evaluate these features.

Keyword　Gradient Boosting, Morphological Analysis, tf-idf, Layout Analysis, Large Language Models, OCR, Text Summarization, Object Detection, Image Captioning

1. はじめに

想隆社では，視覚障害者及びディスレクシア等，書籍を読むことが困難な方を支援するシステム「YourEyes」を提供している．著作権の都合上，このシステムは利用者が撮影した書籍の画像をルールベースで解析し，その解析結果から利用者が確かに書籍を所有していることを検証する．書籍を所有していることが確認できた場合のみデータベース内の書籍の読み上げ音声を提供する．本研究では大きく 2 つの機能提案と実装を行った．1 つ目が AI を用いた書籍のレイアウト解析，２つ目が大規模言語モデル(Large Language Model，以下 LLM)を用いた UX 向上である．

１つ目のレイアウト解析に関して，以前までのルールベースのシステムによるレイアウト解析ではマイナーなレイアウトの書籍や複雑なレイアウトの書籍に対して脆弱であった．本研究では機械学習を用いて書籍画像内のページ番号や柱，本文，脚注などを検出するシステムを実装し，より汎用的な書籍レイアウト解析

を可能にした.

2 つ目の LLM を用いた UX の向上に関して，利用者に対してアンケートを実施し，YourEyes の利用状況や普段の読書状況の調査を踏まえ，テキスト内ミスの修正機能，書籍画像内の挿絵・写真の検出及びキャプショニング機能，テキスト内容要約機能を実装した．また利用者によるこれら機能の評価を行った.

2. レイアウト解析

2.1.学習用データ

学習用データの用意するにあたってアノテーション作業を行った．書籍画像を OCR ソフトを通して画像内のテキストとテキストの位置を取得した．テキストがある一定の範囲に固まっているものを意味的なテキストのブロックとし，そのテキストごとに手動でラベル付けをするアノテーション作業を行った．アノテーション作業にあたっては，書籍画像を確認しながらラベル付けを行う Web アプリケーションを開発し，約 500 ページの画像データを用意した．ラベルは以下の 8 種類である.

- 本文
- ノンブル（記載されているページ番号）
- 柱（ページの上部や左右に配置されていることが多い．書籍名や章名などがある.）
- ノンブル＋柱（例: 岩波文庫の萩原朔太郎詩集「151 『月に吠える』抄」）
- 脚注
- 見出し
- 図のキャプション

2.2.勾配ブースティングでの実装

勾配ブースティングとは，弱学習器である決定木を複数用いてアンサンブル学習させる手法で，ある学習器の予測値の誤差を次の学習器に引き継ぐのが特徴である．勾配ブースティングの実装の一つである LightGBM を利用した. 学習データとして, テキストから以下の 8 つの特徴量を作成した．特徴量は「テキスト長」「テキスト内の非数字文字列の割合」「テキスト範囲の正規化座標」「テキストが『表』または『図』の文字列から始まるか」を用いた．用意したアノテーションデータのうち 7 割を学習データ，3 割をテストデータとして学習及び検証を行った.

結果は以下の通りである．ただしラベル別のデータ量の偏りを考慮した f1-score である重み付き macro-f1 の値は 0.921 であった.

表 1. ラベル別 勾配ブースティングのスコア.

ラベル名	F1-score
本文	0.95
ノンブル	0.99
柱	0.95
ノンブル+柱	1.00
脚注	0.73
見出し	0.80
図のキャプション	0.69

2.3. ベクトル化テキストを加えた勾配ブースティングでの実装

前述した勾配ブースティングで用いた特徴量に，前述した tf-idf によりベクトル化したテキストを新たな特徴量として追加して学習を行った．勾配ブースティングの実装の一つである LightGBM を利用し実装する. 学習用のテーブルデータとして，特徴量を作成した．特徴量は，「テキスト長」「テキスト内の非数字文字列の割合」「テキスト範囲の正規化座標」「テキストが『表』または『図』の文字列から始まるか」「形態素分解しベクトル化したテキスト」とした．学習結果は以下の通りである．ただしラベル別のデータ量の偏りを考慮した f1-score である重み付き macro-f1 の値は 0.934 であった.

表 2. ラベル別 ベクトル化テキストを加えた勾配ブースティングのスコア.

ラベル名	F1-score
本文	0.96
ノンブル	0.99
柱	0.97
ノンブル+柱	0.94
脚注	0.43
見出し	0.81
図のキャプション	0.82

2.4.CNN ベースモデルでの実装

まず初めに 2 層の畳み込み層, 2 層のプーリング層, 2 層の全結合層からなる畳み込みニューラルネットワーク（Convolutional Neural Network, CNN）に対して，バイアス項にテキストから生成した特徴量を追加して学習させた．このときのテストデータに対する Accuracy は 0.70 程度であった.

学習させた CNN に書籍画像を入力した際の中間層の値を画像の特徴量ベクトルとして勾配ブースティングでの推論に用いた．またこのとき中間層の次元数が約 43 万であったため，主成分分析を用いて 100 次元に圧縮したものを特徴量とした．学習結果は以下の通りである．ただしラベル別のデータ量を考慮した f1-score である重み付き macro-f1 の値は 0.920 であった.

表 3. ラベル別 CNN ベースモデルのスコア.

ラベル名	F1-score
本文	0.96
ノンブル	0.99
柱	0.97
ノンブル+柱	0.94
脚注	0.43
見出し	0.81
図のキャプション	0.82

2.1.YourEyes システムへの適用

検証した実装方法の中で最も高い精度を出した，テキスト化ベクトルを特徴量に加えた勾配ブースティングを使用した．以下が実行例である．

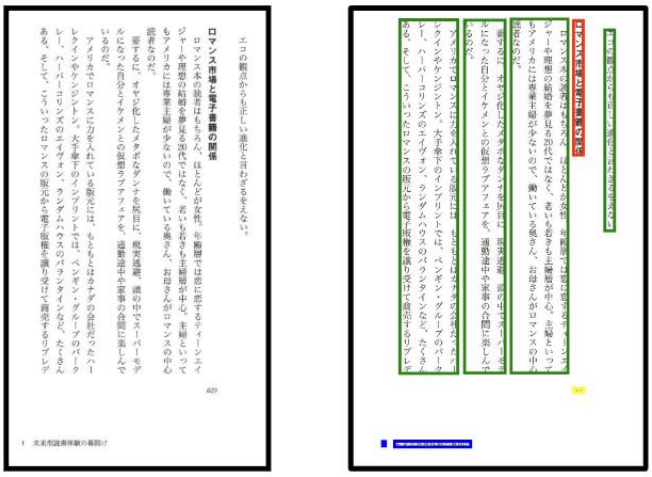

図 1. 新書形式での解析例（「ルポ 電子書籍大国アメリカ」 大原ケイ）.

図 2. 多段組での解析例（「矢沢久雄セレクション アルゴリズム&デザインパターン」矢沢久雄）.

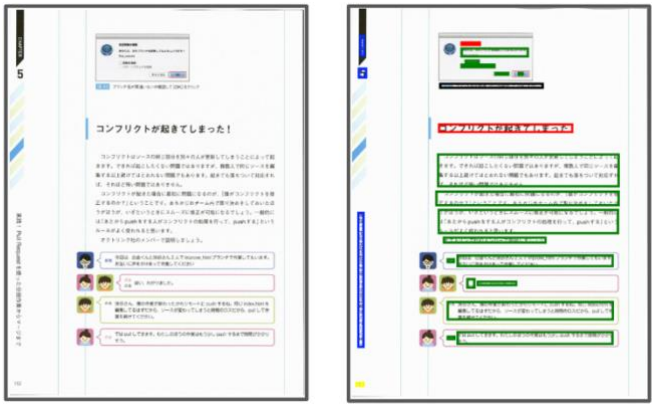

図 3. 技術書での解析例(「Web 製作者のための GitHub の教科書」塩谷 啓，紫竹 佑騎，原 一成，平木 聡).

3. LLM を用いた UX の向上

3.1.利用者アンケートと背景

視覚障害者やディスレクシア等，書籍形式での読書が困難な方の読書方法は限られている．独自に行ったアンケートによれば，テキスト情報を得ることができるデータであれば iPhone の VoiceOver や PC の有償の読み上げソフトを利用している．例えば Kindle などの電子書籍などの媒体や，テキストが埋め込まれた PDF であれば情報にアクセスすることができる．次に利用されていたのがデイジー図書と呼ばれる規格で，書籍の読み上げ音声とテキストやイラストを同時に見ることができるようになっている媒体である．テキスト情報にアクセスできない媒体では光学的文字認識（Optical Character Recognition, OCR）と呼ばれる文字起こしツールを利用する．例えば，テキストが埋め込まれていない PDF の場合は OCR を行ってテキスト化して読み上げを行うことで書籍やテキスト情報にアクセスしている．また紙媒体でしか手元にないもの（手紙や紙のチラシなど)はスキャンを行った上で OCR 処理を行い読み上げを行っている．これらに加えて同居家族や支援センターに支援してもらうという声も見られた．視覚障害者やディスレクシアの方がこうした状況で情報にアクセスしていることがアンケートから判明した．

YourEyes で提供する読み上げ情報はボランティアによって作成された合成音声の読み上げデータである．もしデータベースに読み上げに関するデータが存在しない場合には OCR テキストを提供する．したがって OCR の精度は YourEyes での読書体験に大きく影響すると考えられる．

視覚障害者の方の読書状況や YourEyes の利用状況の調査から，いずれの場合においても，OCR によってテキスト化されたものを読む機会が多くあることが判明した．またアンケートによれば，テキストの読み上

げを行う際，OCR による読み取り誤りや単純なタイプミスなど，テキストが間違いを含んでいると感じる場面によく遭遇していることも判明している．そうした際は，前後の文章の文脈から補完したり，読み上げソフトの「漢字詳細読み」という漢字の詳細情報を得る機能を利用したり，同居家族に読み上げてもらうなどの対処をしている．

また昨今，ChatGPT を始めとする LLM の発展が目覚ましい．LLM は，特定のタスクや学習データを調整することなく，数十億の単語や文からなる大規模なテキストデータベースを用いて訓練されたモデルである．現在は自動翻訳や要約，質問の応答や文章生成などに応用されている．Yang, J. (2023)によれば，LLM が有する能力として，自然言語理解，自然言語生成，知識集約型タスク，推論能力の 4 つが挙げられている．

ここまでの読書状況の調査や LLM の特性を踏まえ，視覚障害者及びディスレクシアの方が直面する課題を解決する機能を 3 つ考案した．

- OCR 誤りを含むテキストの整形
- 書籍画像内の挿絵や写真の検出とキャプションの生成
- テキスト内容の要約

これら 3 つの機能について順に簡単に説明する．詳しい機能や実装方法は次章で説明する．

OCR 誤り誤りを含むテキストの整形について．アンケートの実施により，ユーザーがテキストのタイプミスや OCR の読み取りミスに頻繁に直面している現状があった．これに対し，LLM の自然言語理解能力及び推論能力によってテキスト内の誤りを推定し，自然言語生成能力によって誤りを修正したテキストを生成する，という手段が有効であるのではないかと考えた．

書籍画像内の挿絵や写真の検出とキャプションの生成について．書籍内でテキストとして書き起こす事ができない情報に，書籍内の挿絵や写真がある．これらを物体検出によって検出し，マルチモーダルな LLM を用いてキャプションを生成することで，読者が書籍内の情報により多くアクセスできるようになるのではないかと考えた．マルチモーダルな LLM とは，通常の入力がテキストである LLM と異なり，入力に画像とテキスト，音声とテキストのように複数の形式の入力を受け付けた処理が可能な LLM のことである．

テキスト内容の要約について．上記 2 つの機能によって書籍内情報をテキストにすることができる．ここから LLM の知識集約型タスクによって書籍内容を要約することで，書籍を読むかどうかの判断や，書籍をざっと見通すような使い方ができるのではないかと考えた．

以降では各機能の実装と利用者の評価アンケートを実施した．またこれらの機能の実装については YourEyes のサーバから API として利用できる形で実装を行った．

3.2.テキスト内誤り修正

実装の前に，書籍のページの画像に OCR をかけ，どのような OCR での読み取り誤りがあるかを確認した．利用した OCR エンジンはオープンソースの Tesseract [2]である．統計的な分析は行っていないが，直感的に多かった誤りとしては,形が似ている文字がある文字，ふりがながあるテキスト，記号などが見られた．例えば,「待」と「侍」,「島」と「鳥」,「の」と「@」,「かいけつ解決」などである．これらを参考に，LLM によるテキストの修正機能は 2 つの実装を行い比較検討した．1 つ目がプロンプトでの修正，2 つ目が LangChain での修正である．

1 つ目のプロンプトでの修正は，LLM に特定のタスクを実行させる際の最も基本的な実装である．実装時点において日本語で利用可能なモデルの選択肢は以下を検討した．

- ChatGPT API [3]
- Google Cloud API PaLM 2 [4]
- rinna [5]
- Japanese StableLM Base Alpha 7B [6]

上記には API で利用するものとローカルで動作させるモデルがある．ローカルで動作させる場合，API をホストするサーバーに GPU を追加する必要があり,費用面で月間の維持費が増加する．今回は費用及びドキュメントの充実度，性能面いずれにおいても十分である ChatGPT の API を利用した．留意したい点として，今後より良い LLM モデルが利用できるようになった際はモデルの差し替えを行うことで，機能の性能を最新のものに維持する事ができる．

LLM にタスクを実行させる際，LLM の出力はプロンプトと呼ばれる入力テキストに大きく左右されることから, LLM における入力プロンプトは非常に重要である[7]. 良い出力を得られるプロンプトはブラックボックス最適化として分析されており，プロンプトエンジニアリングとして経験的に有効なプロンプトが蓄積されており，ChatGPT を開発した OpenAI も“良い回答を得られる”サンプルとなるテンプレートプロンプトを提供している [9]．これらを参考に試行錯誤を行い，今回は次のようなプロンプトを採用した．

> これから文章を渡します。それらの文章は OCR によって読み取られた文章でところどころ間違いがあります。 OCR の間違いの特徴として、形が似ている漢字が間違えられることが多いです。例えば「侍」と「待」や、「鳥」と「島」などです。逆に文

字数の変化などは起こりにくいです。以下の文章は、そのような間違いを含んでいる可能性があります。間違いがあれば修正してください。間違いがなければ、そのまま送信してください。:

またChatGPTでは入力テキストの上限トークン数が決まっている．ChatGPT API で呼び出すモデルによってその上限は変わるが，今回利用したモデル（ChatGPT-3.5）では約 4097 トークン，日本語で約 8000 字から 12000 字である．ここでは長文テキストを想定せず，単純な実装で実行可能な数百字程度のものを対象としてテストを実行した．

2 つ目の LangChain を用いた修正では，LangChain は LLM の操作を容易にし，モデルの変更などを行う操作を統一することで，様々なモデルを統合して使用することができるフレームワークである．この LangChain には長文を処理する関数が実装されている．長文の処理では以下のようなプロセスが行われる．まずテキストを LLM の入力トークン制限を超えない程度の大きさに分割してブロックにする．続いて分割したブロックの一つ目に対して与えたプロンプトによる処理を行い，その処理結果と次のブロックを繋げたものに対して同様にプロンプトによる処理を行う．これを繰り返していくことで長文に対する処理を行う事ができる．この長文テキスト処理機能が利用される場面として多いのが長文の要約である．あるブロックに対して要約を行い，その要約結果を踏まえて次のブロックの要約を行い，これを繰り返していくことで長文テキストの内容の要約が可能になる．今回はこの機能を長文テキストにおける誤りの修正に利用する．

長文テキストに対して行う処理にはプロンプトを指定できる．要約するのであれば「以下の文章をまとめてください」や「以下の文章をテーマごとにそれぞれ 150 字程度でまとめてください」のように指定して利用されている．ここでは 1.1 プロンプトによる修正と同様のプロンプトを採用してテストを行った．以下にそれぞれの実装での実行結果を示す．

修正例 1:　芥川龍之介「トロッコ」より

原文

往きと返りと変るせいか、景色の違うのも不安だった。すると今度は着物までも、汗の濡れ通ったのが気になったから、やはり必死に駈け続けたなり、羽織を路側へ脱いで捨てた。
蜜柑畑へ来る頃には、あたりは暗くなる一方だった。「命さえ助かれば――」良平はそう思いながら、辷ってもつまずいても走って行った。

誤りを含む OCR 結果

往きと返りと変るせいか、景色の違うのも不安だった。すると今度は着物までも、汗の濡れ通ったのか気になったか 5 やはり必死に甑け続けたなり羽織を路側へ脱いて捨てた。
蜜柑畑へ来る頃には、あたりは暗くなる一方だった。`命さえ助かれば一一」良平はそう思いながら、Lつてもつますいても走って行った。

ChatGPT-3.5 でのプロンプト指示による修正結果

行きと帰りと変わるせいか、景色の違うのも不安だった。すると今度は着物までも、汗の濡れ通ったのか気になったからか、やはり必死に汗を拭きつづけていらっしゃったなり羽織を路側へ脱いで捨てた。
みかん畑へ来る頃には、辺りは暗くなる一方だった。「命さえ助かれば一一」良平はそう思いながら、とてもつらいとも走って行った。

LangChain を導入し ChatGPT-3.5 でのプロンプト指示による修正結果

往きと返りと変わるせいか、景色の違うのも不安だった。すると今度は着物までも、汗の濡れ通ったのか気になったかもしれないが必死に甑け続けたなり羽織を路側へ脱いで捨てた。
蜜柑畑へ来る頃には、あたりは暗くなる一方だった。「命さえ助かればいい」と良平はそう思いながら、走って行った。

またこれらの実装での処理結果を用いて行ったユーザ評価アンケートの結果を以下に示す．ここでは芥川竜之介「トロッコ」，江戸川乱歩「怪人二十面相」の一部分を用いて，誤りを含む OCR テキスト，プロンプトでの修正，LangChain を用いた修正のそれぞれに対する「耳で聞いたときの理解のしやすさ」を 5 段階評価で行った．

表 4. 誤り修正テキストの評価．

テキスト種別	評価の平均値
OCR 誤りテキスト	2.75
プロンプト修正テキスト	3.5
LangChain を用いた修正テキスト	3.83

LnagChain を用いた実装ではテキストに依存せずユーザの評価の向上が見られた一方で著者特有の表現などが一部言い換えられてしまっている箇所が見受けられた．

3.3.書籍画像内挿絵写真の検出とキャプションの生成

YourEyes ではユーザーが撮影した書籍画像の補正

処理を行った書籍のページが画像として入力される．この画像の中から，書籍文章の内容の理解をサポートする挿絵や写真を検出して切り出しを行い，挿絵や写真の内容を説明するキャプションを生成する．これにより書籍情報の内容を全てテキストとして処理できると考えたためである．この機能では，書籍画像内からの挿絵や写真の物体検出，画像のキャプションの生成と，2 段階に分けて処理を行う．

物体検出タスクは，画像の中に検出対象を検出し，その位置を特定するタスクである．今回は高速かつ効率的な物体検出モデルである YOLO を用いて学習を行い，挿絵や画像，図表の検出を行う．そのために学習データとして文庫本の小説や新書のエッセイ，技術書やハードカバーなど様々な形式の書籍の画像から学習データを作成した．検出対象である，挿絵，写真，図表の 3 種類を対象としてアノテーション作業を行った．作業には VoTT というアノテーションツールを使用した．用意した学習データを用いて YOLOv8m モデルをファインチューニングした．

学習させたモデルによる書籍画像と書籍画像内から検出した物体の画像の例を図 4 に示す．画像の検出はできている一方，ラベル付けが間違っている事がわかる．学習では illustration, image, figure のラベルデータで行った．しかしイラストに対して image というラベル付けが行われている．その他異なる書籍のデータでも image ラベルをつけているものがよく見られた．これは用いた学習データが不均衡データであり, image データが多かったためであると考えられる．ラベルに応じてキャプショニングモデルを使い分けることを想定してこの 3 つのラベル付けを行っていたが今回学習させたモデルではそれが困難であると判断した．したがって一定程度の精度のもとで画像(写真,イラスト，図を問わない）部分で検出したものを利用する方針で進めた．改善のために考えられる手法として，より良いアノテーションデータの作成と利用や，より精度の高い物体検出モデルの利用などが考えられる．

キャプション生成は，画像や動画などの内容を説明するテキストを生成するタスクである．キャプションの生成を行えるモデルやサービスは複数あるが，ほとんど全てのモデルは Transformer[10]というモデルをベースとしたモデルである．Transformer は現在の LLM のベースにもなっており，これを画像処理にも適用させたモデルが現在主流である．ここでは採用を検討したモデルを紹介する．いずれも商用利用可能なライセンスで提供されているものである．

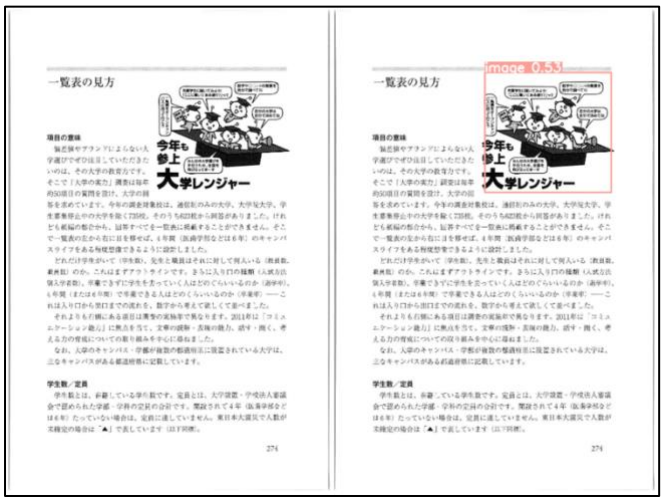

図 4．検出例 出典:「大学の実力 2012」読売新聞社

- Stability.ai,「Japanese InstructBLIP Alpha」[11]
  - 入力したテキストから画像を生成する stable diffusion を開発した企業の日本法人が開発した画像とテキストのマルチモーダルモデル．現時点で日本語に対応した唯一のマルチモーダルモデルであった．画像に写っているものの説明をするだけでなく，画像内の情報に関する質問に回答する事もできる．
- Google,「DePlot」[12]
  - 英語での図表の内容を構造化されたテキストで出力するモデル．画像のキャプションの生成というよりも，英語の図表をテキストで書き起こすモデル．このモデルで図表をテキスト化できれば，LLM は構造化されたテキストを処理できるため，物体検出で図表と検出されたものに対して使用することを想定．
- Salesforce,「blip-image-captioning-large」[13]
  - Salesforce が開発した画像キャプション生成モデル．ヴィジョン言語事前学習（Vision-Language Pre-training, VLP）において，ヴィジョン言語理解タスク（画像をベクトル化し，ベクトル化された自然言語と同様に扱える）とキャプションの生成タスクを柔軟に移行できる BLIP をベースとしたモデル．
- Salesforce,「blip2-opt-2.7b」[14]
  - Salesforce が開発した LLM モデル OPT-2.7b を画像処理タスクへと向上させた BLIP-2 モデル．マルチモーダルであり，Japanese InstructBLIP Alpha と同様に画像キャプションの生成だけでなく画像に対する質問に回答する事ができる．ただし対応言語は英語のみである．

- OpenGVLab,「LLaMA-Adapter v2 multimodal7b」[15]
  - Meta 社が開発したオープンソースの LLaMa 2 という LLM を適用したモデル．対応言語は英語のみではあるが，API が提供されており容易に利用が可能である．

最終的な実装では LLaMa-Adapter v2 multimodal7b を採用した．理由としては API が利用できる点である．blip2-apt-2.7b などは比較的軽量かつ手元の PC で動かして性能を確かめることはできたが，API を提供する場合に API サーバをホストするクラウドサービスで GPU を利用すると利用料が高額になる．オンプレミスでサーバを立てることも検討したが，機能に対する評価アンケートの実施を想定していたため，このアンケートの終了までは実装の早さを優先し，API を提供しているモデルを採用した．また対応言語が英語のみのモデルではあるため，イラストや写真内の日本語テキストを踏まえたキャプションは生成されないことに留意する．今回は出力される英語のテキストを Google の翻訳 API で日本語に翻訳する実装を行った．

以上実装した機能を用いて作成したデータで 6 人のユーザを対象に評価を実施した．2 冊の書籍のうち，1 冊からは挿絵を含むページ，もう 1 冊からは画像を含むページを用いて，それぞれキャプションを含まないテキスト，キャプションを挿入したテキストを用意した．使用した書籍は美馬のゆり『理系女子的生き方のススメ』，安部公房『壁』である．評価はキャプションを含むものと含まないもの，情景が理解しやすいと感じた方を選択してもらった．キャプション付きの文が選択された割合は 66.6%であった．

ユーザ評価の結果から，画像キャプションの必要性が示唆された．残された課題として，キャプショニングの対象となる画像の検出精度の向上，画像情報内の日本語の理解や図表への対応などが挙げられる．検出精度の向上にはトレーニングデータの質の向上が必要であると考える．キャプショニング対象をより多様にするためには，より抜本的に高性能なマルチモーダルモデルが必要である．またこれらが API として安価に提供されること若しくは手頃な価格帯のローカルマシンで動作することが，サービスとして提供するために重要であることも留意する必要がある．

3.3.テキスト内容の要約

テキストの内容の要約は，YourEyes のユーザが手元の書籍を読むかどうかの判断や書籍をざっと見通せることを目標とする機能である．テキストの要約の実装は，LangChain を利用して長文テキストに対して処理を可能にした．また LangChain から呼び出す LLM モデルは，価格及び性能の点から ChatGPT-3.5 を利用した．現在 ChatGPT の API として利用できるモデルの内，最も性能が高いとされているのが ChatGPT-4 であるが，実装及び評価アンケートを実施した時点では ChatGPT-3.5 が最新であった．与えたプロンプトは以下の通りである．

```
次の文章の 200 字程度の要約を作成してください:

{text}

要約:
```

以上の実装にて実行した例が以下である．
以下がテキスト要約のサンプルである．

要約例 3：星新一「表と裏」

男が強盗殺人を成功させ、大金を手に入れるが、神の目に逃れられず逮捕の恐怖におびえ逃げ回る。ある夜、トラックに跳ねられ死の寸前で「天罰を免れることはできなかった」とつぶやく。一方、裁判では容疑者が無実を主張し、真犯人の逮捕を期待するが、その可能性はないと検事は言う。天の助けもなく、恐ろしいことだと結論づけられる。

要約例 2：星新一「健康な犬」

エヌ氏は朝早く起きて犬の散歩をする日課があり、犬の元気さに少し苦労している。ある日、ペットの店に行き、犬の元気さについて相談するが、店の主人は犬がエヌ氏を運動させていることを指摘し、健康のためのペットを提案する。エヌ氏は小鳥を買い、タバコを吸うと小鳥がそれを叩き落とすことで健康を促す効果を実感する。

要約例 3：太宰治「走れメロス」

メロスは王を倒すために行動することを決意し、市に向かう途中で様々な困難に直面する。彼は友人のセリヌンティウスを救うために走り続け、最終的には間に合う。セリヌンティウスを解放し、二人は抱擁し喜び合う。暴君ディオニスも感銘を受け、メロスとセリヌンティウスを仲間に加えることを願う。群衆も彼らを称え、メロスには少女からマントが贈られる。メロスは恥ずかしさを感じながらも、マントを受け取る。

元のテキスト量は星新一「表と裏」が 500 字程度，「健康な犬」が 1300 字程度，太宰治『走れメロス』は 1 万字程度である．

テキストとその要約テキストを読んでもらい，要約の妥当性を 5 段階で評価してもらった．評価は，選択肢 5 本文を非常によく要約している，選択肢 4 本文をある程度要約できている，選択肢 3 良いとも悪いとも言えない要約である，選択肢 2 要約と本文の内容が

異なる，選択肢 1 要約と本文の内容が非常に異なる，の 5 つである．テキストはテキスト長の異なる 3 種類を用意した．使用したテキストは星新一「表と裏」「健康な犬」，太宰治『走れメロス』の全文である．

表 5. 要約テキスト評価結果.

書籍名	テキスト長	評価平均値
表と裏	496	3.83
健康な犬	1260	4.0
走れメロス	10073	2.67

概してテキストの要約の妥当性は平均以上なものが得られることが分かった．またテキスト長やテキスト内容が要約の妥当性に影響すると考えられる．特に著者が星新一で同一である「表と裏」と「健康な犬」を比較した際に,テキストを読んだ所感として「表と裏」は内容が暗喩や皮肉を含み複雑な構造になっており理解が難しい．このことから人間にとって要約しにくい文章内容であれば LLM による要約の妥当性も上がりにくいと考えられる．またテキスト長が長い場合に終盤の箇所を特に要約に含めやすく成る傾向があるように感じられる．これは要約文の長さを長くするなどの工夫による改善や，長いテキストを章ごとなどに区切り，内容に一定のまとまりを持った単位ごとに要約を生成するなどの工夫などが考えられる．

4.まとめ

本研究では，古典的手法と画像処理手法を用いたレイアウト解析において多様な形式のレイアウトに対して高い精度を達成し, LLM と物体検出を用いた機能開発においてはユーザの高い評価を得た．レイアウト解析の今後の課題として，今回使用したテキストから生成したベクトルに関して，例えば見出しや柱であれば体言止めが多用されるなど，品詞情報から特徴量を作成することも可能であると考える．今回は tf-idf で生成した単語の出現頻度のベクトルをそのままテキストの特徴として使用したが，自然言語処理による特徴量生成による精度向上の余地があると考える．また LLM では画像の検出モデルの学習データの質の改善が必要であると考えられる．

文　献

[1] Yang, Jingfeng, Hongye Jin, Ruixiang Tang, Xiaotian Han, Qizhang Feng, Haoming Jiang, Bing Yin, and Xia Hu. "Harnessing the power of llms in practice: A survey on chatgpt and beyond." arXiv preprint arXiv:2304.13712 (2023).

[2] Tesseract, https://github.com/tesseract-ocr/tesseract, 2023/12/15 アクセス.

[3] OpenAI,“Intoroducing ChatGPT and Whisper APIs”, https://openai.com/blog/introducing-chatgpt-and-whisper-apis , 2023/12/15 アクセス.

[4] Google Cloud, “Large Language Models (LLMs) with Google AI”, https://cloud.google.com/ai/llms?hl=ja , 2023/12/15 アクセス.

[5] rinna, https://rinna.co.jp/news/2023/07/20230731.html, 2023/7/31 アクセス.

[6] stability.ai, https://ja.stability.ai/blog/japanese-stablelm-alpha, 2023/8/10.

[7] Chen, Lichang, Jiuhai Chen, Tom Goldstein, Heng Huang, and Tianyi Zhou. "InstructZero: Efficient Instruction Optimization for Black-Box Large Language Models." arXiv preprint arXiv:2306.03082 (2023).

[8] Preferred Networks, “日本語 LLM ベンチマークと自動プロンプトエンジニアリング”, https://tech.preferred.jp/ja/blog/prompt-tuning/, 2023/10/13.

[9] OpenAI, “Prompt engineering”, https://platform.openai.com/docs/guides/prompt-engineering , 2023/12/19 アクセス.

[10] Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser Ł, Polosukhin I. Attention is all you need. Advances in neural information processing systems. 2017;30.

[11] stability.ai, “日本語画像言語モデル「Japanese InstructBLIP Alpha」をリリースしました”, https://ja.stability.ai/blog/japanese-instructblip-alpha , 2023/8/17.

[12] Google, “DePlot: One-shot visual language understanding by plot-to-text translation”, https://ja.stability.ai/blog/japanese-instructblip-alpha, 2022. 2023/12/19 アクセス.

[13] Li J, Li D, Xiong C, Hoi S. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. InInternational Conference on Machine Learning 2022 Jun 28 (pp. 12888-12900). PMLR.

[14] Li J, Li D, Savarese S, Hoi S. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. arXiv preprint arXiv:2301.12597. 2023 Jan 30.

[15] Gao P, Han J, Zhang R, Lin Z, Geng S, Zhou A, Zhang W, Lu P, He C, Yue X, Li H. Llama-adapter v2: Parameter-efficient visual instruction model. arXiv preprint arXiv:2304.15010. 2023 Apr 28.