修士論文

日本語を対象とした大規模言語モデル生成テキストの識別
－データセットの構築と特定のモデルに依存しない分類器の実現－

森川周

主指導教員 白井　清昭

北陸先端科学技術大学院大学
先端科学技術研究科
（情報科学）

令和８年３月

Abstract

In recent years, the technology of Large Language Models (LLMs) has advanced rapidly, making it possible to generate natural and fluent text that is indistinguishable from human-written text. LLMs have the potential to support humans in a wide range of fields, such as education, law, and medicine. On the other hand, their advanced generative capabilities are giving rise to increasingly serious social issues, including the dissemination of fake news and the illicit use of these models in student reports. Against this background, the demand for methods to detect whether a text is LLM-generated text or human-written text is increasing. Most existing research targets English, and research on LLM-generated text detection for Japanese has not been paid much attention. Furthermore, it has been reported that in existing methods, the detection performance decreases when the domain or the LLM of the text differs between the training data and the test data. Under this background, this study aims to establish a general-purpose LLM-generated text detection method for Japanese.

First, we construct a large-scale dataset that does not depend on specific LLMs or domains. We targeted research paper abstracts as the primary data. We collected 9,343 research paper abstracts from 11 academic journals indexed in J-STAGE and regarded them as human-written text. Next, we used LLMs to generate abstracts for the same papers. In this process, we extracted text corresponding to the "Introduction" and "Conclusion" from the collected papers and provided them in prompts. In addition, we performed processes such as generating abstracts whose length is close to the actual abstracts and normalizing punctuation to generate text similar to human-written text. For the LLMs, in addition to the latest commercial models such as GPT-4o and Gemini 1.5 Flash, we used models specialized for Japanese: Llama-3-ELYZA-JP-8B (Llama3) and Llama-3.1-Swallow-8B (Swallow-8B). Furthermore, to verify the generalization performance of the classifier against different domains, we constructed datasets for three domains: Yahoo! Answers, Wikinews, and Wikipedia. Finally, we constructed a dataset consisting of approximately 37,000 pairs of human-written texts and LLM-generated texts for research paper abstracts, and approximately 10,000 pairs for the other domains. This dataset enables the establishment of a general-purpose LLM-generated text detection model and the comprehensive verification of its robustness against different domains.

Using the constructed dataset, we analyze the differences in linguistic features between human-written text and LLM-generated text. First, we investigate statistics such as the average number of characters and the diversity of the vocabulary used. As a result, it was confirmed that human-written text uses a more diverse vocabulary compared to LLM-generated text. Next, we calculated the perplexity

(PPL) of each text using Llama3, Swallow-8B, and Japanese GPT-2, and compared their distributions. As a result, in many domains, LLM-generated text had a lower PPL compared to human-written text, and this tendency was particularly strong when evaluated by the model that generated the text itself. However, in the domain of research paper abstracts, the difference in PPL between them was small, and it was found that it is difficult to perform LLM-generated text detection using only PPL. Furthermore, we conducted analyses from various perspectives, such as part-of-speech distribution, dependency structure, sentiment polarity, and sentence embeddings using SBERT, but no feature that could clearly distinguish between human-written and LLM-generated texts was confirmed.

To detect LLM-generated text with high precision, we construct an LLM-generated text detection model using a Japanese pre-trained RoBERTa model. Specifically, we fine-tune RoBERTa using the constructed dataset to build a classifier. We also conduct experiments to evaluate the performance of the LLM-generated text detection model using our dataset. We used Accuracy, Recall, Precision, F1, and AUROC as evaluation metrics. Furthermore, we created the training data and test data with various combinations of domains and LLMs used for text generation to evaluate the robustness of the classifier.

As a result of the experiments, the classifier trained using all subsets achieved extremely high detection performance, with an Accuracy of approximately 0.96 and an AUROC of 0.99 or higher. On the other hand, the accuracy of LLM-generated text detection by humans was approximately 62%, which was close to random prediction. It implies that the machine learning model captures features that are imperceptible to humans and can detect with high precision. In the evaluation of the robustness of the classifiers, we confirmed that a classifier trained with text from a single domain and a single LLM as training data experienced a decrease in detection performance against test data of unknown domains or LLMs. For example, a classifier trained only on the "Yahoo! Answers" subset had an Accuracy of approximately 0.52 against other domains. In contrast, when one subset was excluded and all the remaining data was used as training data, an Accuracy of 0.91 or higher was maintained even for the test data of the excluded unknown subset. This result demonstrates that the robustness of LLM-generated text detection increases by training a classifier on a dataset that mixes texts from various domains. Furthermore, even against attacks that paraphrased part of the input text using a T5 model, the decrease in the detection accuracy of the classifier was small, confirming that it possesses robustness against detection evasion.

Finally, to explore the factors behind why the classifier demonstrated high performance, we performed visualization of feature words using Integrated Gradients (IG), evaluation of the effectiveness of the layers of the RoBERTa model, and anal-

ysis using linear classifiers. In the visualization by IG, it was found that the model focuses on typical phrases in academic papers that have high frequency in LLM-generated texts. In the analysis of the effectiveness of each layer of RoBERTa, it was found that abstract representations capable of performing LLM-generated text detection with sufficiently high precision are obtained at the stage of the middle layers (6th and 7th layers), which are said to reflect syntactic information. As a result of training linear classifiers whose features are only surface features, syntactic features, or semantic features and evaluating their performance, an Accuracy comparable to the RoBERTa classifier (approximately 0.93) was obtained using only Bag-of-words (BoW) features. These results suggest that machine-learned classifiers do not necessarily detect LLM-generated text through advanced contextual understanding or semantic judgment, but rather use features such as "lexical bias" and "formulaic expression patterns", which are difficult for humans to recognize, as primary clues.

概要

近年，大規模言語モデル（LLM）の技術は飛躍的に進歩し，人間が書いたテキストと区別がつかないほど自然で流暢なテキストを生成することが可能となった．LLM は教育，法務，医療など多岐にわたる分野で人間を支える技術となりうる一方で，フェイクニュースの発信や学生のレポート作成における LLM の不正利用など，その高度な生成能力に起因する社会的な課題も深刻化している．こうした背景から，テキストが LLM で生成されたか人間によって書かれたかを識別する手法の需要が増している．既存研究の多くは英語を対象としており，日本語を対象とした LLM 生成テキスト識別に関する研究は大きく遅れている．さらに，既存手法では訓練データとテストデータとでテキストのドメインや LLM が異なると識別の性能が低下することが報告されている．以上の背景の下，本研究は日本語テキストを対象とした汎用的な LLM 生成テキスト識別手法を確立することを目的とする．

まず，特定の LLM やドメインに依存しない大規模なデータセットを構築をする．主要なデータとして論文の概要を対象とした．J-STAGE に収録された論文の概要を人間生成テキストとみなし，11 個の学術ジャーナルから 9,343 件の論文の概要を収集した．次に，同じ論文に対して LLM を用いて概要を生成する．この際，収集した論文から「はじめに（Introduction）」と「おわりに（Conclusion）」に相当するテキストを抽出し，これらをプロンプトとして与える．また，実際の概要に近い文字数の概要を生成したり，句読点を統一したりするなどの処理を行い，人間が書いた概要に近いテキストを生成する．LLM には，GPT-4o および Gemini 1.5 Flash といった最新の商用モデルに加え，日本語に特化したモデルである Llama-3-ELYZA-JP-8B（Llama3）および Llama-3.1-Swallow-8B (Swallow-8B) を用いる．加えて，異なるドメインに対する分類器の汎化性能を検証するために，Yahoo!知恵袋，Wikinews，Wikipedia の 3 つのドメインについてもデータセットを構築する．最終的に論文の概要については人間生成テキストと LLM 生成テキストの組がおよそ 37,000 件，その他のドメインについてはおよそ 10,000 件からなるデータセットを構築した．本データセットは，特定のドメインやモデルに依存しない汎用的な LLM 生成テキスト識別モデルの確立と，その汎化性能の包括的な検証を可能にする．

構築したデータセットを用いて人間生成テキストと LLM 生成テキストの言語的特徴の違いを分析する．まず，平均文字数や使用される語彙の多様性などの統計量を調査する．その結果，人間生成テキストは LLM 生成テキストと比較して，より多様な語彙を使用していることが確認された．次に，Llama3，Swallow-8B，および日本語 GPT-2 を用いて各テキストのパープレキシティ（PPL）を算出し，その分布を比較する．その結果，多くのドメインで LLM 生成テキストの方が人間生成テキストに比べて PPL が低く，特にテキストを生成したモデル自身で評価した場合にその傾向が強かった．しかし，論文の概要ドメインにおいては両者の PPL の差は小さく，PPL のみを用いて LLM 生成テキストを識別するのは困難であることがわかった．さらに，品詞分布，依存構造，感情極性，および SBERT を用い

た文埋め込みなど多様な観点から分析を行ったが，人間生成テキストと LLM 生成テキストを明確に識別できるような特徴は確認できなかった．

LLM 生成テキストを高精度に識別するため，日本語事前学習済み言語モデル RoBERTa を用いた LLM 生成テキスト識別モデルを構築する．具体的には，構築したデータセットを用いて RoBERTa をファインチューニングし，分類器を構築する．また，本研究のデータセットを用いて LLM 生成テキスト識別モデルの性能を評価する実験を行う．評価指標として，Accuracy，Recall，Precision，F1，AUROC を採用する．さらに，ドメインやテキスト生成に用いた LLM の様々な組み合わせで訓練データとテストデータを作成し，識別モデル (分類器) の汎化性能を評価する．

実験の結果，全てのサブセットを用いて学習した分類器は，Accuracy 0.96 程度，AUROC 0.99 以上という非常に高い識別性能を達成した．一方，人間による LLM 生成テキスト識別の正解率は約 62%であり，ランダムな予測に近かった．このことから，機械学習モデルが人間には知覚できない特徴を捉えて，高精度で識別できることが確認された．汎化性能の検証において，単一のドメインと単一の LLM のテキストを訓練データとして学習された分類器は，未知のドメイン・LLM のテストデータに対する識別性能が低下することを確認した．例えば「Yahoo!知恵袋」のみのサブセットで学習した分類器は，他ドメインに対する Accuracy は 0.52 程度となった．対照的に，1 つのサブセットを除外して残りの全てのデータを訓練データとした場合，除外された未知のサブセットのテストデータに対しても Accuracy は 0.91 以上となり，様々なドメインのテキストを混合したデータセットから分類器を学習することで LLM 生成テキスト識別の汎化性能が増すことを確認した．さらに，T5 モデルを用いて入力テキストの一部をパラフレーズ（言い換え）した攻撃に対しても，分類器の識別精度の低下は小さく，識別回避に対する頑健性を有していることが確認された．

最後に，分類器が高い性能を発揮した要因を探るため，Integrated Gradients (IG) による特徴語の可視化，RoBERTa モデルの層の有効性の評価，および線形分類器による分析を行う．IG による可視化では，LLM 生成テキストにおいて頻度が高く，学術論文における典型的なフレーズにモデルが注目していることがわかった．RoBERTa の各層の有効性の分析では，構文情報を反映すると言われている中間層 (第 6,7 層) の段階で十分に高い精度で LLM 生成テキストを識別できる抽象表現が得られていることがわかった．表層的特徴，構文的特徴，意味的特徴のみを素性とする線形分類器を学習してその性能を評価した結果，トークンの出現頻度 (Bag-of-words) の特徴のみで RoBERTa 分類器に匹敵する Accuracy(0.93 程度) が得られた．これらの結果は，機械学習された分類器が必ずしも高度な文脈理解や意味的な判断によって LLM 生成テキストを識別しているわけではなく，人間には認識困難な「単語の偏り」や「定型的な表現パターン」といった特徴を主な手がかりとしていることを示唆する．

目 次

図 目 次

表 目 次

第1章　序論

1.1　背景

1.1.1　生成 AI の技術的進展と社会的リスク

近年，自然言語処理（Natural Language Processing; NLP）分野は飛躍的に進歩した．特に，大規模言語モデル（Large Language Model; LLM）は急速な発展を遂げており，人間に近いレベルで文章が生成できるようになった．2022 年 11 月に OpenAI が ChatGPT を公開したことは，世界的な「生成 AI ブーム」を引き起こす決定的な転換点となった．これを皮切りに，Google による Gemini や Anthropic による Claude など，極めて高性能なモデルが次々と開発され，激しい開発競争が繰り広げられている．これらの最新のモデルは，インターネット上の膨大なテキストデータを学習することで，人間が記述した文章と見分けがつかないほど自然で流暢なテキストを生成する能力を獲得した．その応用範囲は，従来の翻訳や要約といった言語タスクにとどまらず，複雑なプログラミングコードの生成，高度なデータ分析，論理的な推論支援など多岐にわたる．教育，法務，医療といった専門性の高い領域から，日々の業務効率化に至るまで，LLM は現代社会において不可欠な知的インフラとしての地位を確立しつつある．

しかし，LLM の普及は社会に多くの利便性をもたらすと同時に，その高い文章生成能力に起因する様々な問題を顕在化させている．最も懸念される問題の一つが，フェイクニュースの生成と拡散である．最近の調査では，2022 年 1 月から 2023 年 5 月の間に公開された記事において，主要なニュースサイト上の AI 生成記事が 57.3%増加し，誤情報サイトにおいては 474%という驚異的な増加を示したことが明らかになっている [11]．LLM を用いれば，事実に基づかないもっともらしい文章を大量かつ安価に生成することが可能であるためである．

また，サイバーセキュリティの観点からは，フィッシングメールや詐欺メッセージの文面が高度化し，従来のスパムフィルターをすり抜ける事例も報告されている．米国のセキュリティ企業 SlashNext が発表した調査報告書『The State of Phishing 2023』によれば，同社が 2022 年第 4 四半期から 2023 年第 3 四半期までの 12ヶ月間にわたり，メール，モバイル，およびブラウザ上でスキャンされた数十億件の脅威データを分析した結果，ChatGPT の公開以降，悪意あるフィッシングメールの検知数は 1,265%の増加した [42]．同報告書では，この急激な増加の背景として，攻撃者が生成 AI を利用することで，高度な標的型攻撃やビジネスメール詐欺を迅

速かつ大量に展開できるようになった点を指摘している．こうした脅威は日本国内においても顕在化している．警察庁が公開した『令和 6 年におけるサイバー空間をめぐる脅威の情勢等について』[14] によれば，生成 AI の普及に伴い,「フィッシングメール」や「偽情報」の作成に AI が悪用されるリスクが公的に指摘されており，専門知識を持たない者でも高度な攻撃が可能になることへの懸念が示されている．

1.1.2 学術・教育分野における課題と規制

LLM や生成 AI が文章作成支援ツールとして社会に浸透しつつある一方で，教育および学術機関において，その適切な運用と規律の維持が課題となっている．本来，LLM は人間の知的生産活動を補助する有用な技術であるが，教育の場面など自身の思考力や表現力の育成が求められる場面や，使用が明示的に禁止されている状況下で利用された場合，学習効果の毀損や評価の公平性が損なわれる懸念がある．現状，LLM が生成したテキストは人間が記述したものと見分けがつかないほど自然であり，教育現場において読書感想文の課題や大学のレポートなどで使用が禁止されているにも関わらず使用された場合，識別が困難であることが，教育現場における深刻な課題となっている．例えば，文部科学省は 2024 年 12 月，生成 AI の普及や技術進展を踏まえた「初等中等教育段階における生成 AI の利活用に関するガイドライン（Ver. 2.0）」を発表した [31]．同ガイドラインでは,「不適切と考えられる例」として,「各種コンクールの作品やレポート・小論文等について，生成 AI による生成物をほぼそのまま自己の成果物として応募・提出する」行為を挙げており，こうした行為は自分のためにならず，評価基準や応募規約によっては不正行為に当たると明記している．

デジタル教育評議会が 2024 年に実施した世界 AI 学生調査 [4] では，調査対象 16 か国の学生の 86%が学習に AI を使用しており，最も利用されているツールは ChatGPT であった．一方で，急速な普及にもかかわらず 58%の学生が AI リテラシーの不足を自覚しており，大学側の対応に不満を抱く学生も多く，AI ガイドラインを十分に理解している学生は 5%にとどまると報告されている．さらに，学生の 60%は AI 利用時の評価の公平性に懸念を示しており，教育機関における AI 利用の課題が浮き彫りとなっている．

こうした生成 AI の影響は初等中等教育にとどまらず，高度な専門性が求められる学術出版の分野においても顕在化している．Gray は，学術データベース Dimensions に収録された論文を対象に，LLM が好んで使用する特定の形容詞や副詞の出現頻度を分析した [9]．その結果，ChatGPT の公開翌年である 2023 年に出版された論文において,「intricate」の使用頻度が前年比 117%増,「commendable」が 83%増,「meticulous」が 59%増と，異常な急増を示していることが明らかになった．さらに，これらの「強い」指標となる単語を含む論文は，2023 年だけで前年比 87.4%

も増加しており，Gray は少なくとも 6 万本（全論文の 1.15%）の学術論文が，適切な開示なしに LLM によって生成・修正された可能性があると推定している．

こうした状況を受け，主要な学術ジャーナルは生成 AI の利用に関する厳格なガイドラインを相次いで策定している．Nature 誌は 2023 年の社説において，著者の資格には研究に対する説明責任（accountability）が伴うことを前提とし，「いかなる LLM ツールも研究論文の著者として認められない（no LLM tool will be accepted as a credited author on a research paper）」と明言した [32]．同様に，Science 誌の編集長 Thorp は，同誌が求める「オリジナル（original）」な著作物という要件に照らし，ChatGPT が生成したテキストの使用は許容されないと述べている．Thorp は，AI による生成物は「結局のところ，ChatGPT からの盗用（plagiarized from ChatGPT）」に当たると指摘し，AI ツールの安易な利用に警鐘を鳴らしている [49]．

こうした背景から，LLM 生成テキストを識別する技術への需要は急速に高まっている．しかし，現在の LLM のテキスト生成能力は，非常に高く，テキストが人間によって書かれたか LLM によって生成されたかを人間が判定することは困難である．一方で，LLM 生成テキスト識別手法の研究も進められているが，その多くは英語圏を中心としたものであり，日本語を対象とした研究は遅れているのが現状である．加えて，既存の研究で構築されたデータセットでは特定の LLM のみを用いて構築された場合が多く，多様な LLM から生成されたテキストに対応できる汎用的な識別手法は未だ確立されていない．

1.2 目的

前述した背景を踏まえ，本研究では，特定の言語モデルに依存しない，汎用的な日本語 LLM 生成テキスト識別手法の開発を目的とする．具体的には，以下の 3 つの課題に取り組む．

- 第一に，複数の最新の LLM を用いたデータセットの構築である．
  既存のデータセットは，単一のモデル（ChatGPT が多い）によって生成されたものが多く，多様な LLM を網羅できていない．そこで本研究では，日本語の学術論文データベースである J-STAGE から収集した人間生成テキスト（論文の概要）と，それに対応する形で生成された LLM テキストの対からなるデータセットを新たに構築する．生成には，OpenAI の ChatGPT に加え，Google の Gemini，Meta の Llama3 など，アーキテクチャや学習データの異なる複数のモデルを採用する．これにより，LLM 生成テキストに共通する普遍的な特徴を学習できるデータセットの構築を目指す．

- 第二に，日本語における人間と LLM の言語的特徴の違いの解明である．
  構築したデータセットを，形態素解析による品詞分布，係り受け解析による

構文構造，感情分析，パープレキシティ（Perplexity）などの指標を用いて多角的に分析する．英語圏の研究では，LLM 生成テキストは人間と比較して単調であり，感情表現が平坦であるといったような報告があるが，文法構造の異なる日本語においても同様の傾向が見られるかは明らかでない．本研究では，人間生成テキストと LLM が生成したテキストとでこれらの言語的特徴を定量的に比較することで，識別モデルがどのような特徴に着目すべきかの指針を得ることを目指す．

- 第三に，事前学習済みモデルを用いた識別モデルの構築と評価である．
  日本語の分類タスクにおいて高い性能を示す RoBERTa モデルをベースとし，構築したデータセットを用いてファインチューニングを行うことで，入力テキストが LLM によって生成されたか否かを判定する二値分類モデルを構築する．評価においては，Accuracy，F1，AUROC（Area Under the Receiver Operating Characteristic Curve）を主要な指標とし，実用的な水準である Accuracy で 0.9 以上の精度達成を目標とする．さらに，特定のモデルで学習した分類器が，未知のモデルや異なるドメインのテキストに対しても有効であるか（汎化性能）を検証し，特定の LLM を想定しない汎用的な識別手法の構築を目指す．

1.3 本論文の構成

本論文の構成は以下の通りである．2 章では，本研究に関連する研究について述べる．さらに，既存の LLM 生成テキスト識別手法や評価手法と比較し，本研究の特徴を示す．3 章では，日本語 LLM 生成テキストのデータセットの構築方法について説明する．データ収集の手順，複数の LLM，複数のドメインにおける LLM 生成テキストの生成方法，テキストの後処理，構築したデータセットの統計を示す．4 章では，構築したデータセットに対する言語的特徴の分析結果について説明する．人間と LLM 生成テキストの文長，語彙，品詞，係り受け構造，感情分析，およびパープレキシティ（PPL）の分析を行う．5 章では，LLM 生成テキスト識別器の学習および評価実験について述べる．まず構築したデータセットを用いた RoBERTa による分類器の構築方法，ゼロショット手法や人間による識別実験の方法について説明する．そして，構築した分類器の評価や考察を行う．さらに，IG(Integrated Gradients) や Attention の可視化，層別評価などを用い，モデルの判断根拠となる内部挙動を分析した結果を示す．6 章では，本研究のまとめと今後の課題を述べる．

第2章　関連研究

本章では，LLM 生成テキスト識別に関する先行研究を整理し，本研究の位置づけを明確にする．2.1 節では，先行研究で構築されたデータセット（HC3，CHEAT など）の特徴と課題をまとめる．2.2 節では，言語モデルの統計的性質に基づくゼロショット識別手法について，基本指標（対数尤度，Rank／Log-rank，Entropy）と，近年提案された手法（DetectGPT，DetectLLM，DNA-GPT）を概説する．2.3 節では，事前学習済みモデルのファインチューニングに基づく教師あり分類器と，言語的特徴量を用いる識別手法を整理する．2.4 節では，識別回避の攻撃と識別性能への影響を述べる．2.5 節では，生成側で識別信号を埋め込むウォーターマーク方式を紹介する．2.6 節では，日本語を対象とした既存研究の特徴と課題をまとめる．最後に 2.7 節では，以上を踏まえた本研究の特徴を述べる．

2.1　LLM 生成テキスト識別データセット

LLM 生成テキスト識別タスクにおいて，大規模で高品質なデータセットは必要不可欠な要素となっている．LLM の急速な発展に伴い，識別手法の研究開発を目的としたデータセットが構築されてきた．

【HC3 (Human ChatGPT Comparison Corpus)】

Guo らが構築した Human ChatGPT Comparison Corpus (HC3) データセットは ChatGPT(GPT-3.5) が生成したテキストと人間生成テキストを収集した大規模で先駆的なデータセットである [10]．本データセットは，同一の質問に対して人間と ChatGPT がそれぞれ回答したテキストをペアとして含めている．人間の回答データは，(1)WikiQA や Reddit ELI5（Explain Like I’m Five）といった既存の QA データセット，(2) 金融，医療分野の専門家に対する質問応答，(3)Wikipedia から抽出された概要の説明から構成されている．データセットは英語と中国語の 2 言語で構成され，英語版では 58,546 件の人間の回答と 26,903 件の ChatGPT の回答，中国語版では 22,259 件の人間の回答と 17,522 件の ChatGPT の回答が含まれている．

また，作成したデータセットに対し，品詞分析，依存関係解析，感情分析，Perplexity (PPL) などの言語的特徴の分析を行っている．品詞分析では ChatGPT 生成テキストは名詞の割合が高いこと，感情分析では ChatGPT 生成テキストが中

立的な感情のテキストを出力しやすいことを発見している．また，ChatGPT 生成テキストは，人間生成テキストに対して PPL が有意に低いことを発見している．しかし，HC3 データセットには，データ作成に使用される LLM やプロンプトの多様性がないなど，いくつかの限界がある．

【HC3 Plus】

Su らは HC3 の課題であったタスクの偏りに対処するため，拡張版である HC3 Plus を構築した [46]．HC3 が主に QA タスクを対象としていたのに対し，HC3 Plus は要約，翻訳，言い換えといったテキスト変換タスク，すなわちテキストをその意味内容を変えずに表層表現を変えるタスクに焦点を当てている．本データセットは，CNN/DailyMail，XSum，WMT などの既存のコーパスを人間のテキストとし，それらに対応する生成テキストを GPT-3.5-Turbo を用いて生成することで構築した．

Su らは，これらのタスクにおいては生成テキストが変換元文章の意味的制約を強く受けるため，QA タスクと比較して人間と AI の差異が縮まり，識別難易度が高くなることを報告している．しかし，HC3 Plus も依然として GPT-3.5 系列の LLM で構築されており，多様な LLM に対する汎化性能には課題が残されている．

【CHEAT (CHatGPT-writtEn AbsTract dataset)】

Yu らはコンピュータサイエンス分野の学術論文の概要をドメインとするデータセット CHatGPT-writtEn AbsTract dataset (CHEAT) を構築した [54]．本データセットは，IEEE Xplore から収集したコンピュータサイエンス分野の 15,395 件の人間の概要と，ChatGPT によって生成された 35,304 件の概要から構成される．CHEAT は単純な概要生成（Generation）に加え，人間が書いた概要を推敲させたもの（Polish），人間と AI のテキストを混合したもの（Mix）を含んでいる．CHEAT は，コンピュータサイエンス分野のテキストのみを対象としており，領域横断的な検証ができない点に課題がある．

【GROVER】

Zellers らは，ニューラルネットワークによるフェイクニュースの識別を目的として，GROVER データセットを構築した [55]．本データセットでは，人間生成テキストは，Common Crawl から収集した Google News の 5,000 ドメインの記事群（RealNews）から取得されている．LLM 生成テキストはこれらの記事を含むテキストから学習された 15 億のパラメータを持つ Grover-Mega モデルによって生成されている．ChatGPT 出現以前の研究であり，対象がニュース記事に特化している点が特徴である．

【MAGE ベンチマーク】
特定のモデルに依存しない汎用的な識別手法の確立に向けて，Li らは MAGE ベンチマークを構築した [24]．このデータセットは人間生成テキストと LLM が書いたテキスト 447,674 件で構成されており，ニュース，物語，質問応答など 10 種類の異なるタスクに対し，Llama，GPT-4，GLM など 27 種類もの多様な LLM を用いて生成されたテキストを含んでいる．
Li らはこのデータセットを用いた様々な評価を行い，特定のモデルやドメインで学習された分類器は，未知のモデルやドメインに対して著しく性能が低下する脆弱性を指摘している．これは，特定の LLM に依存しない識別手法を開発するためには，学習データのモデルやドメインの多様性が不可欠であることを裏付けている．

このように英語圏では多様なモデルやドメインを網羅したデータセットによる検証が進んでいるが，日本語を対象とした LLM 生成テキスト識別の研究は未だ少なく，既存の日本語データセットは単一のモデルに依存しているか，小規模な検証に留まっているのが現状である．日本語環境において特定の LLM やドメインに依存しない汎用的な識別手法を確立するためには，複数の最新の LLM と複数のドメインを包括した大規模な日本語データセットの構築が必要不可欠である．

2.2 ゼロショット識別手法

前節では，LLM 生成テキストの識別におけるデータセットの重要性と現状について述べた．これらのデータセットを用いた LLM 生成テキスト識別手法の研究も急速に進展している．識別手法は大きく分けて，大量のラベル付きデータを用いてモデルを訓練する「教師あり学習」と，追加の訓練を必要とせずモデルの性質そのものを利用する「ゼロショット (統計的手法)」に分類される．本節では，後者である言語モデルの統計的性質や確率分布を利用した手法について概説する．

2.2.1 基本的な識別手法

2019 年の GPT-2 の公開は，LLM の生成能力が飛躍的に向上した転換点であり，それと同時に，生成されたテキストをどのように識別するかという試みが本格的に開始された時期でもある．この時期に言語モデルが持つ統計的性質のみを利用して識別を行う「ゼロショット」の手法が複数提案され，現在のベースラインとして使用されている．

【対数尤度 (Log-likelihood)】

Solaiman らは，GPT-2 の公開に伴うレポートにおいて，モデルの出力分布を利用したいくつかの「ゼロショットのベースライン」を提案している [43]．対数尤度（Log-likelihood）は，与えられたテキスト内の各トークンごとの対数確率の平均を測定する方法である．LLM は，自身の出力確率分布に基づき出現確率の高い語彙を優先的に選択してテキストを生成する．この生成されたテキストに対し，各トークンの出現確率を算出すると，それらはモデルにとって予測が容易なトークンであるため，算出される確率は高くなる．こうした LLM の特性により，LLM が生成したテキストは人間生成テキストに比べて対数尤度が高くなる傾向があることが知られている．この指標は，LLM 生成テキスト識別タスクにおける最も一般的なベースライン指標の一つとして広く認識されている．レポートでは，15 億（1.5B）パラメータの GPT-2 モデルを用い，Top-K 40 のサンプリング設定で生成されたテキストに対し，対数尤度に基づく識別手法で 83%から 85%の正解率（Accuracy）を達成したことを報告している．

【GLTR】

Gehrmann らは，LLM 生成テキスト識別に利用できる複数の統計量（Rank，Log-rank，Entropy）を定義し，テキストの内，どの箇所が LLM で生成された可能性が高いか表示するフレームワークである GLTR を提案した [7]．

まず，生成確率の順位（Rank）は，モデルの出力確率分布において，実際に出現した単語が何番目に高い確率を持っていたか（順位）によって識別する指標である．Rank が小さいほどモデルにとって予測しやすい語であり，LLM 生成文ではこのような高確率の語が選択されやすいことから，平均 Rank は人間生成テキストよりも小さくなる傾向が報告されている．また，生成確率の対数順位 (Log-rank) は，この Rank に対して対数を取った指標であり，非常に大きな順位値の影響を緩和することを目的としている．

エントロピー（Entropy）は，次トークンの確率分布 $p(w \mid w_{1:i-1})$ の分布の違いにより識別する指標である．生成確率が比較的高い単語が特定の少数の語に集中しているか，あるいは多くの語に分散しているかを定量化する．LLM が生成する文では，高確率語が選ばれやすい傾向があるため分布が尖りやすく，結果としてエントロピーが小さくなる．この特徴を利用し，エントロピーの値が低いほど，そのテキストが LLM によって生成されたと判断し識別する．

GLTR は，テキストの各トークンについて Rank，Log-rank，Entropy といった指標を計算し，その結果を色分けやヒストグラムとして可視化することで，LLM 生成テキストの特徴を人間が把握できるように設計したシステムである．例えば，確率上位 10 語以内のトークンを緑，上位 100 語以内を黄，上位 1000 語以内を赤，それ以外を紫で表示することで，テキストが「どの程度モデルにとって予測しやすい語で占められているか」を一目で確認できるようになっている．

これらの指標は現在の LLM 識別手法のもっとも単純なベースラインとして広く

使用されており，本研究でも比較対象として採用する．なお，本研究で実際に使用する各指標（対数尤度，Rank，Log-rank，Entropy）の具体的な定義式については，5.1.2 項にて述べる．

2.2.2 最新のゼロショット識別手法

近年は，対数尤度や Rank といった単純な統計量だけではなく，言語モデルの確率分布そのものの構造に踏み込んだゼロショット識別手法が提案されている．その代表的な例として以下の 3 つの論文を紹介する．

【DetectGPT】

Mitchell らは，LLM が自ら生成した文は，人間が記述した文に比べて，モデルの対数確率関数において負の曲率が強い領域に現れやすいという性質に着目し，DetectGPT という手法を提案した [30]．識別対象のテキストに対して，小さなマスク付き言語モデル（例：T5）を用いて意味的に類似した摂動テキストを多数生成し，元のモデルにおける対数確率 $f(x) = \log p(x)$ と摂動後テキスト x' に対する対数確率 $f(x')$ の差分を平均することでスコアを計算する．このスコアに閾値を設定することで，人間生成テキストと LLM 生成テキストを追加の学習なしのゼロショットで判別する．

実験では，ニュース記事を対象としたデータセット上で GPT-NeoX-20B など複数の生成モデルを評価し，従来の最良ゼロショット手法が達成していた AUROC 0.81 に対して，DetectGPT は 0.95 まで性能が向上することを示している．また，この枠組みは特定のモデルやデータセットに依存せず，多様なソースモデル・サンプリング条件に対して安定して高い性能を示すことが報告されており，現在の LLM 生成テキスト識別研究における代表的なゼロショット手法として広く参照されている．

【DetectLLM】

Su らは LLM の出力分布に内在する「順位（Rank）」に着目したゼロショット識別手法 DetectLLM を提案した [45]．DetectLLM は，LLM が生成するトークンは人間生成テキストに比べて，その LLM の出力確率分布では高い順位を取りやすいという経験的事実に基づき，テキスト全体の対数尤度と対数順位の両方を利用する点が特徴である．論文では，対数尤度と対数順位を組み合わせて識別する LRR（Log-Likelihood Log-Rank Ratio），およびテキストに摂動を加えた際の対数順位の変動により識別する NPR（Normalized Log-Rank Perturbation）という二つのゼロショット分類器を新たに定義している．

実験では，LRR と NPR のいずれも既存のゼロショット手法を上回る性能を示し，中でも NPR は比較的少ない摂動で安定した AUROC を達成していることが報告されている．また，NPR は DetectGPT と同様に摂動ベースの手法であるが，

DetectGPT に比べて摂動の種類や数が少量で済み，計算コストも小さいという利点を持つ．実験では，LRR と NPR は従来の Log-likelihood，Rank，Log-rank といったベースライン手法を上回り，平均で 3〜4 ポイント程度 AUROC を改善する結果が報告されている．

【DNA-GPT】

Yang らは，追加学習を必要としないゼロショット識別手法として DNA-GPT（Divergent N-Gram Analysis） を提案した [53]．この手法は，識別するテキストが LLM によって生成されたテキストだった場合，前半テキストを与えられた場合，その LLM にその続きを再生成させると類似した後半テキストを生成しやすいという性質を利用して識別を行う手法である．DNA-GPT では，まず入力テキストを前半と後半に分割し，後半部分を対象モデルにより複数回再生成する．そして，元の後半テキストと再生成された後半テキストとの間で n-gram 一致率を計算し，この一致度をもとに「人間生成テキスト」か「LLM 生成テキスト」かを判別する．

実験では，GPT-3.5，GPT-4，GPT-NeoX，LLaMA 系など複数のモデルを対象に評価が行われ，AUROC が 0.99 に近い高い識別性能を示した．また，パラフレーズなどの軽微な編集攻撃に対しても性能劣化が小さいことが報告されており，既存のゼロショット識別手法よりも頑健性が高い点が特徴である．

2.3 教師あり学習による識別

2.3.1 事前学習済みモデルのファインチューニングによる識別手法

現在，LLM 生成テキストの識別において事実上の標準となっているのが，BERT や RoBERTa などの事前学習済み言語モデルをファインチューニングする手法である．この手法は LLM 生成テキスト識別の最も標準的なアプローチの一つであるが，その判断根拠はブラックボックスであり，未知のデータに対する脆弱性が懸念されている．このアプローチの基礎を築いたのは，GPT-2 の公開に伴う OpenAI のレポートである．

【GPT-2 レポートにおける RoBERTa 分類器の報告】

Solaiman らは，GPT-2 のレポートの中で生成されたテキストの悪用リスクを軽減するための防御策として，ゼロショットの手法に加えて RoBERTa を用いた分類器を開発・評価した [43]．彼らは，RoBERTa-Large モデルを「人間生成テキスト」と「GPT-2 が生成したテキスト」の二値分類タスクでファインチューニングすることで，最大モデル（1.5B パラメータ）が生成したテキストに対しても約 95%という高い正解率を達成できることを報告している．これは，同報告内で示されている TF-IDF unigram/bigram を用いた線形分類器の精度（124M モデルで約 88%，1.5B モデルで約 74%）を大きく上回る性能であり，ファインチューニング手法が

単純な統計的手法を凌駕することを示している．この研究は，単純な統計的手法よりも，事前学習済みモデルを教師あり学習でファインチューニングする手法が高性能であることを示した．

【大規模データセット（HC3 ／ CHEAT）における教師あり分類器の性能報告】

このアプローチの有効性は，その後構築された大規模データセットを用いた検証でも確認されている．Guo らは HC3 データセットにおいて，RoBERTa ベースの分類器が F1 スコアが英語で 99.82%, 中国語で 98.79%という高い値に達したと報告している [10]．また，学習時と異なるドメインのデータに対しては従来の統計的手法（GLTR）よりも堅牢であるものの，F1 スコアが低下する点についても議論されている．同様に，Yu らも CHEAT データセットにおいて，RoBERTa,BERT などを使用した分類器を学習し,「Generation」データに対しては AUROC が 100%に達したが，人間が関与する「Polish」データでは精度が低下し,「Mix」データに対してはさらに AUROC が低下することを報告している [54]．

【学習ドメイン外への汎化性能検証】

Rodriguez らは，科学技術論文（物理学および生物医学分野）を対象に，GPT-2 生成テキストの識別実験を行った [38]．彼らは，ある分野（例：物理学）で学習したモデルを異なる分野（例：生物医学）に適応させるクロスドメイン設定において，ターゲット領域の学習データが 100 件程度と極めて少量であっても，RoBERTa を用いることで約 0.89〜0.91 の高い正解率が達成可能であることを報告している．

【教師あり分類器の頑健化 (J-Guard)】

事前学習済みモデルの識別能力を補強するために，テキスト固有の「構造的特徴」をモデルに統合する方法も提案されている．Kumarage らは，ニュース記事の識別において，プロのジャーナリストが遵守する執筆ルール（AP スタイルブック等）に基づいた「ジャーナリズム特徴（Journalism Features)」に着目し，これを RoBERTa 等の分類器に組み込むフレームワーク「J-Guard」を提案した [22]．彼らは，導入文の長さ，受動態の使用頻度，特定の句読法（オックスフォードカンマの有無など）といった，LLM 生成文の文体や構造上の特徴を定義・抽出し，これを深層学習モデルの学習プロセスにガイドとして導入した．

実験の結果，この手法は ChatGPT を含む多様な生成モデルに対して高い識別精度を示し，特にパラフレーズ（言い換え）攻撃に対して高い堅牢性を示した．RoBERTa ベースの OpenAI Detector が AUROC で約 15 ポイント低下する一方で，構造的特徴を組み込んだ J-Guard は性能低下を軽微（約 7 ポイント減）に抑えることに成功している．これは，深層学習モデルに構造的・文体的特徴を明示的に与えることで，識別の安定性が向上することを示唆している．

2.3.2　言語的特徴量を用いた識別手法

ニューラルネットワークによるブラックボックスなアプローチとは対照的に，テキストから明示的な言語的特徴（Linguistic Features）を抽出し，それらを従来の機械学習アルゴリズムに入力して分類を行う手法も研究されている．このアプローチの最大の利点は，モデルが何に基づいて判断を下したかが明確であり，高い解釈性（Interpretability）を持つ点にある．

【文体的特徴による識別】

Shah ら（2023）は，説明可能な AI（XAI）を用いて，文体的特徴のみで LLM 生成テキストを高精度に識別できることを示した [41]．彼らは Wikipedia 導入部を人間生成テキストとし，GPT-J および Orca に同内容のテキストを生成させて 2 種類の 2 万件データセットを構築した．各文に対し，平均単語長や句読点の数といった語彙的特徴，Flesch Reading Ease などの読みやすさ指標，Herdan's C・Maas・Simpson's Index といった語彙多様性指標を付与し，ロジスティック回帰や SVM 等で分類を行ったところ，最も単純なロジスティック回帰でも Orca データセットで 92 %の正解率を得た．さらに LIME および SHAP を用いた分析により，特に Herdan's C や Simpson ' s Index など語彙多様性を示す指標が識別に強く寄与することが示され，深層モデルを用いずとも適切な特徴量設計により説明可能な高精度識別が可能であることが明らかとなった．

【生成プロンプトの変化に対する脆弱性の報告】

一方で，こうした特徴量ベースの手法には，生成条件の変化に対する脆弱性も指摘されている．Mindner らは，ChatGPT を用いて生成された教育関連のテキストを対象に，多角的な特徴量（Perplexity，意味的特徴，エラー率など）を用いた識別実験を行った [29]．彼らの実験では，単純に「テキストを生成せよ」と指示した（Basic）場合，XGBoost 等の分類器は F1 スコア 98%という極めて高い性能を示し，既存の識別ツール（GPTZero）を上回った．

しかし，プロンプトで「人間が書いたように見えるように書き直せ」と指示した場合（Rephrased），同一の特徴量を用いても F1 スコアは約 79%まで大幅に低下することが確認された．この結果は，手作業で作成した特徴量が特定の生成パターンには極めて有効であるものの，意図的な隠蔽やプロンプトエンジニアリングによって文体が操作された場合，その識別能力が著しく損なわれる可能性を示唆している．

2.4　パラフレーズ攻撃

LLM 生成テキスト識別手法の評価においては，識別器そのものの性能だけでなく，利用者や攻撃者による改変に対してどの程度頑健であるかを明らかにするこ

とが重要である．特に，生成された文章を意味を保ったまま書き換えるパラフレーズ攻撃は，現実の利用場面でも想定される代表的な回避手段の一つである．

【DIPPER によるパラフレーズ攻撃】

Krishna ら (2023) は長文向けのパラフレーズモデル DIPPER を設計し，既存分類器に対する攻撃実験を行った [20]．DIPPER は T5 系モデルを基盤とし，小説コーパスなどから構築した段落レベルのパラフレーズでファインチューニングすることで，入力段落と意味的には類似しつつも表現の異なるテキストを出力できるようにしたモデルである．さらに，語彙の置き換え量を制御するパラメータ L と，文順や構成の変化量を制御するパラメータ O を導入することで，言い換えの強度を連続的に調整できるよう設計されている．攻撃時には，LLM が生成した段落を Dipper に入力し，L と O を変化させながら複数のパラフレーズを生成し，それらを分類器に入力して性能の変化を評価する．

実験では，GPT2 や OPT，GPT-3.5 など複数の LLM が生成したニュース記事や長文 QA を対象に，DetectGPT や OpenAI Detector などを含む多様な分類器に対する影響を検証した．その結果，DIPPER による強いパラフレーズを適用すると，多くの識別モデルで AUROC が大きく低下し，場合によってはほぼランダム予測に近い水準まで性能が劣化することが報告されている．一方で，意味類似度モデルにより元文とパラフレーズ文の意味的近さを評価したところ，多くの設定で高い類似度が維持されており，意味内容をほとんど変えずに分類器だけを無力化できる強力な攻撃であることが示されている．

2.5　ウォーターマークの埋め込みによる識別

LLM 生成テキストの識別手法において，生成側のモデルであらかじめ「ウォーターマーク」を埋め込むことで，後から識別を容易にする方法も提案されている．その代表的な例として，Kirchenbauer らの手法を紹介する [16]．

この手法では，まず各生成ステップで語彙集合を擬似乱数により green list と red list に分割し，green list に属するトークンのロジットに一定のシフトを加えることで，それらが選択されやすいように出力分布をわずかに偏らせる．識別時には，同じ擬似乱数系列を用いて各トークンが green list に属するかどうかを再計算し，シーケンス中の green list トークンの割合に基づく統計量が有意に大きいかどうかを検定することで，ウォーターマーク付きテキストか否かを判定する．

実験では，OPT 系列のモデルを用いてニュース記事コーパス上で評価を行い，感度評価の例としてトークン長がおよそ 200 程度のシーケンスを仮定した上で，one-proportion z-test の検定統計量 z に対して閾値を $z > 4$ とすることで，片側 p 値として偽陽性率 3×10^{-5} に相当する水準を保ちつつ，ウォーターマーク付きテキストを高い感度で識別できることが示されている．また，バイアス強度 δ や green list の割合といったパラメータを適切に設定すれば，テキストの PPL や読

みやすさをほとんど損なうことなく，識別性能を確保できることも報告されている．一方で，低エントロピーな出力（訓練データの暗唱を含む）では出力分布を十分に偏らせにくく識別性能が低下しやすいことや，パラフレーズ攻撃を加えることで green list 側への偏り（信号）が弱まるといった制約も議論されている．

ウォーターマークの埋め込みは，生成モデル側が識別信号を埋め込むことを前提とするが，現状の主要な LLM ではウォーターマークの付与がされておらず，また既に述べたようにパラフレーズ攻撃によって信号が弱まる可能性も指摘されている．以上を踏まえ，本研究では生成元 LLM がウォーターマークを埋め込むことを考慮せず,「与えられたテキストのみを入力としてそれが LLM 生成テキストかを識別する」という前提の下で日本語 LLM 生成テキスト識別タスクに取り組む．

2.6 日本語を対象とした LLM 識別手法

ここまでの節で説明した通り，英語では多数の識別手法が提案されている一方で，日本語を対象に行った研究はきわめて限られている．先行研究を調査した限りでは，丸井らの Yahoo!知恵袋コーパスを用いた日本語 LLM テキスト識別の研究 [28] 以外に，日本語を対象とした識別の論文は管見の限り見当たらない．

丸井らの論文では，Yahoo!知恵袋第 3 版コーパスのベストアンサーを人間生成テキストとして用い，質問文をプロンプトとして三つの LLM に回答を生成させることでデータセットを構築している．生成に用いられるモデルは，gpt-3.5-turbo，llm-jp-13b-instruct-full-jaster-dolly-oasst-v1.0，Swallow-7b-instruct-hf の三種であり，各モデルについて LLM 生成テキストと人間生成テキストを 7,500 件ずつ，合計 30,000 件のデータからなるデータセットを構築している．

識別手法としては，まず Waseda RoBERTa 日本語モデルをベースとした教師あり分類器を学習している．9 割を訓練データ，1 割をテストデータとし，人間 と GPT-3.5，人間 と llm-jp，人間 と Swallow の三種類の分類器を個別に学習している．併せて，日本語 GPT-2 を用いて各文に対する対数尤度，生成確率の順位，生成確率の対数順位，エントロピーを計算し，それぞれの AUROC を算出することで，ベースラインとなるゼロショット識別性能も評価している．

評価の結果，同一の LLM に対しては，gpt-3.5 では AUROC が 0.998, llm-jp では 0.946 前後の高い性能を示す一方で，swallow に対しては 0.733 と低い数値となっている．また，異なる LLM では AUROC が 0.4〜0.7 程度まで低下し，訓練データとテストデータで使用する LLM が異なる場合に課題があることが示されている．また，ゼロショット検出手法では gpt-3.5-turbo をテストデータとした場合には，これらの指標はいずれも AUROC 0.79 〜 0.89 の性能を示し，生成確率の対数順位の精度が最も良かった．別の LLM である llm-jp や Swallow をテストとした場合には 0.56〜0.78 程度にとどまる．また，gpt-3.5 を Yahoo!知恵袋データでファインチューニングし，人間生成テキストにより近い分布を持つ LLM 生成テキストを作成も行っている．RoBERTa 分類器の学習に用いられているの

はファインチューニング前の gpt-3.5 の生成テキストであり，ファインチューニング後の gpt-3.5 生成文テキストは検証用テストデータとして用いられている点である用いる実験も行っている．ファインチューニング前のモデルで生成したテキストをテストデータとしたときと比べて，統計量ベースの手法（対数尤度，Rank，Log-rank，Entropy）はほぼランダムに近い性能まで劣化する一方，RoBERTa 分類器は AUROC 0.92 程度を維持することが報告されている．

2.7 本研究の特徴

既存の LLM 生成テキスト識別に関する研究は，その大部分が英語圏で行われており，日本語を対象とした研究は非常に少ない．2.6 節で述べた先行研究は日本語を対象としていたが，Yahoo!知恵袋のテキストに限定されていた．これに対し本研究では，論文の概要，Yahoo!知恵袋，ニュース記事，Wikipedia という文体が異なる 4 つのドメインを対象とし，さらに GPT-4o や Llama3 といった最新の 4 種類の LLM から生成されたデータセットを構築する．これにより，学術不正やフェイクニュース，また様々な LLM から生成されたテキストといった多様な識別状況を想定した検証を可能としている．

また，従来の教師あり手法である RoBERTa 分類器の構築だけでなく，ゼロショット識別手法，人間による識別の結果との比較を行う．従来の研究の課題として，学習に使用していないドメインやモデルを対象とした場合，精度が低下することが報告されている．本研究では，教師あり手法については，単なる精度の報告に留まらず，学習時と異なるドメインやモデルに対する性能の評価を行い，分類器の汎化性能についての調査する．さらに，学習データ量の変化や 2.4 節で述べたパラフレーズ攻撃に対する堅牢性の実験など，識別精度に影響を与えうる外的要因についても検証を行う．

2.3 節で述べたように，RoBERTa などの事前学習済みモデルを用いたファインチューニングは，現在の識別における標準手法の 1 つであり高い精度を誇るが，モデルがテキストのどのような特徴に基づいて判断しているのかといった判断根拠の解明には踏み込んでいなかった．これに対し本研究では，分類器の判断根拠の解明に取り組む．Integrated Gradients (IG) や Attention の可視化，線形分類器による分析を組み合わせることで，分類器が意味を理解して分類に取り組んでいるのか，あるいは単なる単語の偏りなどに基づいて分類しているのかを検証する．

以上をまとめると，本研究はデータの網羅性，手法の比較，堅牢性，判断根拠の解明などの多角的な視点から，日本語の LLM 生成テキスト識別手法の有効性と限界を明らかにすることが特徴である．

第3章　データセット構築

本章では，特定の言語モデルに依存しない日本語 LLM 生成テキスト識別手法の開発，およびその評価に用いるデータセットの構築について述べる．3.1 節では，本研究の主となる学術論文の概要を用いたデータセットの構築方法について述べる．3.2 節では，Yahoo!知恵袋，Wikipedia，ニュース記事から構成されるデータセットの構築について述べる．これらのデータセットは主に識別モデルの汎化性能を検証するために用いる，3.3 節では，本章で構築した各データセットのデータ数などの統計情報について述べる．

3.1　論文の概要データセットの構築

1.1 節で述べた通り，現状，LLM によって生成されたテキストは人間が記述したものと見分けがつかないため，教育分野や学術分野における LLM 生成テキストの識別は重要な課題となっている．不正利用を抑止し，評価の公平性や研究の信頼性を担保するために，本研究では学術分野に対応した LLM 生成テキスト識別タスクのデータセットとして，人間の書いた論文の概要と，LLM が生成した概要の対から構成されるデータセットの構築する．なお，LLM による概要生成にあたっては，論文全文ではなく「はじめに（Introduction）」と「おわりに（Conclusion）」に相当するセクションのみを入力として用いる手法を採用する．

本節の構成は以下の通りである．3.1.1 項では，J-STAGE を用いた論文データの収集および抽出方法について述べる．3.1.2 項では，収集したテキストを用いた LLM による概要生成の設定，および生成したテキストの後処理について述べる．3.1.3 項では，実際に構築されたデータセットの具体例を示す．

3.1.1　論文データの収集と抽出

3.1.1.1　収集する論文の選定

ここでは人間生成テキストの論文の概要の収集方法について述べる．日本語の学術論文を収集するため，国立研究開発法人科学技術振興機構（JST）が運営する電子ジャーナルプラットフォームである J-STAGE を利用した [19].

データの収集には J-STAGE Web API（検索用 API）を使用した [18]．収集の対象とする論文として，タイトル，論文 PDF，概要が取得できること．論文，概

要が日本語で書かれていることを条件として収集した．まず，J-STAGE Web API を用いて条件に合致する論文のメタデータを取得し，論文タイトルと記事リンク（URL）を抽出してリスト化した．その後，このリストに基づき各論文の PDF ファイルをダウンロードした．

3.1.1.2　テキストの抽出

次に，収集した論文の PDF ファイルからテキストを抽出する過程について説明する．PDF ファイルの解析およびテキストデータの抽出には，Python のライブラリである PyMuPDF（fitz）を使用した．PyMuPDF は PDF の内部構造にアクセスし，テキストを単なる文字列としてだけでなく，ページ上の座標情報を含むブロック単位で取得することが可能である．本研究では，この機能を用いてレイアウト情報を保持したままテキストを抽出し，後述するフッター除去などのクリーニング処理に活用した．

本研究では，LLM によって論文の概要を生成する際，前述のとおり論文の全文ではなく「はじめに（Introduction）」と「おわりに（Conclusion）」に相当する 2 つのセクションのみを用いる．この方法を採用した理由は，予備実験の結果に基づくものである．当初は CHEAT [54] の設定を参考に，論文タイトルとキーワードから概要を生成する手法を検討した．しかし，入力情報が限定されすぎることが原因で，生成テキストが同じような表現となり，人手でも容易に判別できるテキストしか出力されないという問題が確認された．そこで論文全文を入力としたところ，出力されるテキストの品質が不安定になる傾向が見られた．一方で,「Introduction」に相当するテキストのみを用いた場合は，生成される概要の情報量が不足し，文章の質が十分ではなかった．検証の結果,「Introduction」と「Conclusion」を組み合わせた入力が，人間の概要と同程度の長さを保ちつつ，質の高い概要を安定して生成できることが確認された．

「Introduction」と「Conclusion」に相当するセクションを抽出するために，正規表現を用いた処理を行った．単に「1.」や「I.」で始まる行を機械的に抽出した場合，図表のキャプション（例：「図 1. 構成図」）や箇条書きの番号をセクション見出しとして誤認識する問題が発生した．そこで本研究では，対象とする論文の集合から「1.」等の後に続く文字列を収集し，その統計をとり，実際にセクション見出しとして機能している文字列パターンを目視で確認し,「Introduction」のセクション見出しのリストを構築した．この構築したリストに基づき，図表番号やノイズを除外し，セクション見出しのみを抽出するフィルタを構築した．なお，セクション見出しに「謝辞」が含まれる場合は，概要生成において不要な情報であるため，抽出対象から除外する処理を加えた．また，Conclusion については，抽出された有効なセクション (テキストが存在するセクション) のうち，最後のセクションを採用する手法を取った．

3.1.1.3　テキストデータのクリーニング

PDF から抽出されたテキストには，本文とは無関係なノイズが多く含まれるため，以下の基準で不要な文字列の削除処理を行った．

1. **ページ周縁部のノイズ除去**: PyMuPDF を用いて PDF のレイアウト座標を解析し，ページ左下などの領域に存在するテキストブロックを，ページ番号や著作権情報など本文ではないテキストとみなして抽出対象から除外した．
2. **図表キャプションの削除**: 正規表現を用い,「図 1」「Table 2」「Figure 3」などの図表番号を含むキャプション行を特定し削除した．
3. **引用および参考文献の削除**: 本文中の [1] などの引用，および「参考文献（References）」セクション以降の全てのテキストを一律で削除した．
4. **書誌情報の削除**: 著者の所属や連絡先などの個人情報を LLM に与えてしまわないように，郵便番号（〒...），電話番号，メールアドレス，URL，住所のパターンに一致する文字列を削除した．
5. **特定文字列パターンによるノイズ削除**: 座標指定では除去しきれない論文誌特有のヘッダー情報（例：「Annual Conference...」）や，論文 ID などの特定の文字列パターンについては，正規表現を用いて論文誌別に定義し削除した．
6. **ページ番号の削除**:「1」や「2」などの数字のみで構成される行は，ページ番号の残りとみなし削除した．

3.1.2　概要生成実験の設定とテキストの加工

本項では，3.1.1 項で収集および抽出を行った論文テキストを入力とし，LLM を用いて概要を生成するプロセスについて述べる．まず，使用した LLM とその選定理由について述べ，次に生成に用いたプロンプトの設計について説明する．最後に，生成したテキストのノイズ除去について説明する．

3.1.2.1　使用した LLM と選定理由

本研究では，特定の言語モデルに依存しない汎用的な LLM 生成テキスト識別手法の構築を目指している．そのため，単一のモデルのみを用いるのではなく，開発元，アーキテクチャ，パラメータ数，および学習データが異なる複数の LLM を選定した．具体的には，以下の 4 つのモデルを採用した．

- **GPT-4o**: OpenAIが提供するGPT-4系列の商用LLMである[33]．本研究では，実運用上広く利用される商用モデルを評価対象に含める目的でGPT-4oを採用した．GPT-4oは，音声・視覚・テキストを処理できるマルチモーダルモデルとなっており，GPT-4と比較し対応速度や低コスト化などが図られている．概要生成にはAPIで指定可能なスナップショットのうちgpt-4o-2024-08-06を使用した[34]．

- **Gemini 1.5 Flash**: Googleによって開発されたGemini 1.5系列の商用LLMである．Gemini 1.5は，テキスト・画像・音声などを統合的に扱うマルチモーダルモデルとして設計されており，特に長大なコンテキスト入力を効率的に処理できる点を特徴とする[15, 48]．Google Generative AI APIにおいて提供されていたGemini 1.5 Flashを採用した．なお，Gemini 1.5 Flashは2025年9月29日をもって提供終了となっている．

- **Llama-3-ELYZA-JP-8B**: 株式会社ELYZAによって開発された日本語特化のオープンLLMである．Meta社のLlama 3をベースモデルとし，日本語テキストによる追加事前学習および指示追従学習が施されている[6, 12]．公式に報告されているJapanese MT-Benchによる評価では，80億パラメータ規模の軽量モデルでありながら，「GPT-3.5 Turbo」や「Claude 3 Haiku」などに匹敵する性能を示している．

- **Llama-3.1-Swallow-8B-Instruct-v0.3**: 東京工業大学（現：東京科学大学）の研究チームによって開発された日本語対応オープンLLMである．Meta社のLlama 3.1に，日本語データによる追加学習および指示追従学習を行うことで，日本語対話能力の強化が図られている[50, 2]．公式に公開されているMT-Bench JAによる評価では，80億パラメータ規模のオープンモデルとしては高い性能を示しており，特にv0.3では過去バージョンと比較して対話品質の向上が確認されている．

3.1.2.2　概要生成プロンプトの設計

前述の通り本研究では，論文本文から抽出したIntroductionとConclusionを入力プロンプトとして与える方式を取った．LLMに与えるプロンプトでは，その論文本文の情報に加えて，人間の書く概要に近づけるために以下の工夫を行った．

- **文体の統一**：指定を行わなかった場合，LLMは「です・ます調」で生成する傾向があった．文末表現が識別の手掛かりにならないよう，生成条件として「である調」を明示した．

- **文字数の制御と再生成**：テキストの文字数を手掛かりにLLM生成テキストであると識別されることを避けるため，論文誌ごとに人間が書いた概要の文

字数分布を調査し，最頻値に近い文字数を目標文字数としてプロンプトに含めた．ただし，指定文字数から外れる出力が多かったため，生成結果が最頻値± 50 字の範囲を外れた場合は再生成を行った．再生成は最大 5 回までとし，生成品質と計算コストのバランスを考慮した．なお，Gemini 1.5 Flash は指定文字数を大幅に超過する傾向が強かったため，他の 3 モデルでは最頻値に近い文字数を目標として与える一方で，Gemini のみ目標文字数を最頻値より小さく設定し，その上で最頻値± 50 字の範囲を満たすまで再生成する手法を採用した．

- **生成パラメータの設定**：生成文の多様性を確保する目的で，temperature=0.6, top_p=0.9 を採用した．

実際に生成に用いたプロンプトを以下に示す．

> 以下に論文の Introduction と Conclusion を記述します．この文章から ［ここに文字数が入る］ 文字数程度の論文の概要を生成してください．また，概要はです・ます調ではなく，である調で生成してください．
>
> 論文の Introduction： ［ここに Introduction が入る］
>
> 論文の Conclusion： ［ここに Conclusion が入る］

3.1.2.3 生成テキストの後処理

LLM により生成された概要テキストについて，人間書いた概要と形式を揃えるため，以下の後処理を行った．本処理は内容の改変を目的とするものではなく，表記および構造の統一を目的としている．

- **短文・空行の除去**: 生成結果には，概要本文とは別に,「以下は概要である」等の前置きのような記述が混入する場合がある．これらは概要本文としては不要であり，識別の手掛かりにもなり得る．そこで，生成テキストを改行単位で分割し，文頭から 50 字以下の短い文を除去することで，概要本文と無関係な前置きや注記を排除した．
- **句読点の統一**: 句読点で簡単に識別できることを避けるため,「、」「。」をそれぞれ「,」「.」に統一した．
- **文頭空白の削除**: LLM が文頭に不要な空白を生成する傾向が見られたため，文頭に含まれる空白文字を削除した．
- **空行の削除**: 人間が書いた論文の概要の形式に合わせるため，空行をすべて削除した．

3.1.3　生成した概要の例

本項では，前述の手法を用いて実際に生成されたLLM生成テキストの例を示す．以下は，Llama3を用いて生成された概要の例である．

- **出典**：人工知能学会全国大会論文集
- **論文タイトル**：BERTベース分類器とSHAPによる地方議会会議録における特徴表現抽出の試み
- **生成された概要**：地方議会会議録の分析は，地方議会の議論の内容を可視化し，地域特有の課題や発言者の政治思想を明らかにするために重要な研究分野である．近年，地方議会会議録コーパスを構築し，発言中のキーワードや特徴的表現の抽出を試みる研究が行われている．しかしながら，従来の分析では単語レベルでの抽出が主流であり，複数の単語で構成されるキーフレーズや文脈に依存する特徴的表現を考慮することができなかった．本研究では，事前学習済みのBERTを用いた分類器を構築し，分類器の判断根拠を数値化することで発言文中の特徴的な表現を抽出する手掛かりを分析した．実験の結果，分類精度は約5割となったが，分類に成功した事例を可視化した結果，都道府県や発言者を特徴付ける表現がハイライトされることがわかった．特に，文脈に依存する複数単語の連なりも含まれることが確認できた．本研究の成果は，地方議会会議録の分析に新たな視点を提供し，議会の議論をより深く理解するための手がかりを与えることが期待される．

3.2　汎化性能検証用データセットの構築

特定のドメインやモデルに依存しない汎用的な識別手法を実現するためには，学習時とは異なる性質を持つテキストに対する識別性能，すなわち汎化性能の検証が不可欠である．そのため，論文の概要とは異なるドメインのテキストを用いた汎化性能検証用のデータセットを構築した．検証対象として，Yahoo!知恵袋，Wikinews，Wikipediaの3つのドメインを選定した．

本節の構成は以下の通りである．3.2.1項では，各ドメインのデータの収集方法と前処理について述べる．3.2.2項では，LLMによるテキストの生成設定について述べる．3.2.3項では，生成テキストの加工とクリーニングについて述べる．3.2.4項では，最終的な検証用データセットの構成について述べる．

3.2.1　汎化性能検証用データの収集と前処理

3.2.1.1　収集対象ドメインの特徴

検証対象として，以下の3つのドメインを選定した．

- **Yahoo!知恵袋**: Yahoo!知恵袋[1]は，ユーザが質問を投稿し，別のユーザがその質問に対する回答を投稿するウェブ上の QA 型掲示板である．データセット構築にあたり，国立国語研究所が現代日本語の語彙や文法の使用実態を体系的に把握することを目的に構築した「現代日本語書き言葉均衡コーパス（BCCWJ）」[27] にサブコーパスとして含まれている「Yahoo!知恵袋」のデータを用いた．このデータは，2004 年 10 月から 2005 年 10 月にかけて投稿された約 312 万件の質問と回答のペアを収集したものである．また，複数の回答が投稿されていた場合，ベストアンサーの回答が収集されている．他ドメインと異なり口語的表現が多く含まれる．

- **Wikinews**: ウィキメディア財団が，中立的な視点に基づいた自由なニュースソースを世界中の人々に提供することを目的に運営する「日本語版 Wikinews」[51] の公式ダンプデータを用いた．ダンプデータは 2025 年 9 月 16 日に取得し，取得時点の最新版を使用した．

- **Wikipedia**: ウィキメディア財団が，人類の知識の総和をすべての人に提供することを目的に運営するオンライン百科事典「日本語版 Wikipedia」[52] の公式ダンプデータを用いた．ダンプデータは 2025 年 9 月 2 日に取得し，取得時点の最新版を使用した．

3.2.1.2 原文データの抽出と前処理

上記 3 つのドメインのコーパスから人間生成テキストを抽出した．この際，強調文字やリンク，注釈などのノイズが含まれていたため，これらを除去する前処理を行った．以下，ドメイン毎に原文データの抽出と前処理の詳細を述べる．

Yahoo!知恵袋においては，BCCWJ 収録の Yahoo!知恵袋データから質問文と回答文の組をランダムに 1,000 件抽出した．

Wikinews においては，公式ダンプより取得した最新版のデータから，タイトル，ニュース記事本文のテキストを抽出した．テキストのクリーニングとして，正規表現を用いたパターンマッチングにより，記事末尾に付与される「出典」「注釈」「関連記事」といった本文以外のテキストや，カテゴリ行，画像タグ（Category:, frame—等），言語間リンクを除去した．また，文頭の空白削除および空行の除去を行った．

Wikipedia においては，まずアルゴリズム，計算機科学，自然言語処理など 25 種類の情報工学に関連するカテゴリを人手で選定した．また，記事全文では LLM 生成テキスト検出の実験的検証に用いるには長すぎるため，記事の導入部（リード文）を人間生成テキストとした．また，前処理として，パターンマッチングでファイル，画像，各種テンプレート，タグ，リンク，強調等のメタ情報の削除を行った．また，文頭の空白削除および空行の除去も行った．

[1] https://chiebukuro.yahoo.co.jp/

3.2.2 LLM によるテキストの生成と後処理

3.2.1 項で収集した人間生成テキスト（原文データ）に対応する LLM 生成テキストを構築するため，Llama-3-ELYZA-JP-8B および Llama-3.1-Swallow-8B-Instruct-v0.3 の 2 モデルを用いて LLM 生成テキストを作成した．

3.2.2.1 プロンプトの設計

生成に使用したプロンプトは，各ドメインの特性に合わせて設計した．表 3.1 に，各ドメインにおける具体的なプロンプトの構成を示す．Yahoo!知恵袋では，人間による回答と文脈を揃えるため，BCCWJ から抽出した質問文をそのまま入力として与えた．Wikinews および Wikipedia では，表 3.1 に示す通り記事タイトル等の情報を与えることで，トピックに沿った内容を生成させた．生成におけるパラメータ設定は，論文の概要データセットの構築（3.1.2 項）と同一の条件を採用した．

表 3.1: 汎化性能検証データにおける生成プロンプト

ドメイン	プロンプトの構成
Yahoo!知恵袋	質問文を直接入力
Wikinews	記事タイトル: [タイトル] 公開日: [日付] 上記のニュース記事の本文を書いてください。
Wikipedia	[タイトル] について、Wikipedia 記事の冒頭の概要部分を作成してください。必ず、で・ある調で作成してください。

3.2.2.2 生成テキストの後処理

LLM によって生成されたテキストには，指示に対する承諾や解説といった本文以外のノイズ，および識別タスクにおいて不適切なヒントとなる情報が含まれていた．これらを適切に除去し，評価の妥当性を確保するため，以下の手順で後処理を実施した．

- **前置きの除去**: 生成結果には，本文とは別に「以下は，〜の内容です.」といった前置きや補足の説明が混入する場合がある．これらを排除するため，Wikinews では「ニュース記事」，Wikipedia では「Wikipedia」,「ウィキペディア」という語句が含まれる場合，それらを本文とは異なるノイズとみなして除去した．

- **Wikinews のタイトル除去**: Wikinews において，プロンプトで与えたタイトルが生成文の冒頭にそのまま出力されることがあった．これを防ぐため，テキスト冒頭の文字列がタイトルと完全一致する場合や，最初の改行の前にタイトルと同じ文字列が存在する場合これ除去する処理を行った．

3.2.3 検証用データの LLM 生成テキストの例

本項では，論文の概要以外のドメイン（Yahoo!知恵袋，Wikinews，Wikipedia）においての LLM によるテキストの生成例を示す．以下に，Llama-3-ELYZA-JP-8B（Llama3）を用いて生成された，Yahoo!知恵袋における「花粉」に関する質問への回答，Wikinews における「脱線事故」に関する記事，および Wikipedia における「スプーフィング攻撃」の記事をそれぞれ示す．

- **Yahoo!知恵袋**
  花粉は、主に樹木や草花の花粉が風や虫などによって運ばれます。緑が少ないところでも、周囲に花粉を生産する植物があれば、花粉は飛来する可能性があります。以下に、花粉が飛来する可能性のある例を挙げます。

  1. 樹木の花粉: 樹木の花粉は、風によって数キロから数十キロも運ばれます。例えば、スギやヒノキの花粉は、風に乗って数十キロも飛来することが知られています。緑が少ない都会でも、周囲に樹木があれば、花粉が飛来する可能性があります。
  2. 草花の花粉: 草花の花粉は、風や虫によって運ばれます。例えば、タンポポやセイタカアワダチソウなどの草花の花粉は、風や虫によって数十メートルから数百メートルも運ばれます。緑が少ないところでも、周囲に草花があれば、花粉が飛来する可能性があります。
  3. 遠くから飛来する花粉: 花粉は、風や気象条件によって、数十キロから数百キロも運ばれることがあります。例えば、遠方の山や森林で大量に花粉を生産する樹木があれば、風に乗って花粉が飛来する可能性があります。

  以上のように、緑が少ないところでも、周囲に花粉を生産する植物があれば、花粉は飛来する可能性があります。

- **Wikinews**
  新潟県内を通る JR 羽越線で発生した脱線事故で、国土交通省は 30 日、線路の安全性を確認し、12 月から速度規制を解除することを発表した。2007 年 8 月 24 日、羽越線の新津ー阿賀野川間で貨物列車が脱線し、線路が大破する事故が発生。原因は、線路の老朽化と保守不十分が指摘され、国土交通省は事故後、線路の全面改修工事を実施。工事は順調に進み、安全性が確認されたため、速度規制を解除することになった。速度規制解除後は、列車の運行

速度が通常の速度に戻り、所要時間も短縮される見通し。JR 東日本は、12月 1 日から通常の運行体制に移行する予定で、通勤や通学に影響が出る可能性は低いとみている。一方、事故原因の究明と再発防止策の策定は、引き続き行われる。国土交通省は、安全性を確保するため、定期的な点検やメンテナンスの実施を JR 東日本に指導している。

- **Wikipedia**
  スプーフィング攻撃（spoofing attack）は、ネットワーク通信のセキュリティを侵害する手口の 1 つで、攻撃者が他人の IP アドレスやドメイン名を詐称して、通信を送信する行為を指す。詐称された IP アドレスやドメイン名は、実際には攻撃者が所有しているものであるため、通信の送信元を偽装することが可能である。

3.3 データセットの統計

本章で構築した論文の概要データセットおよび汎化性能検証データセットのデータ数などの統計量について述べる．各ドメイン，LLM 別のデータ内訳を表 3.2 に示す．また，論文の概要データセットの引用元論文集および抽出件数を表 3.3 に示す．

表 3.2: データセットの統計

ドメイン	人間	GPT-4o	Gemini	Llama3	Swallow
論文の概要	9,343	9,343	9,343	9,343	9,343
Yahoo!知恵袋	904	—	—	904	904
Wikinews	1,828	—	—	1,828	1,828
Wikipedia	2,249	—	—	2,249	2,249

論文の概要ドメインでは，4 種類の LLM（ChatGPT，Gemini，Llama3，Swallow-8B）を用いて生成を行っている．本データセットは，各データに対し LLM 生成テキストが 1 つずつ対応するように構築しているため，すべてのモデルにおいて人間生成データと LLM 生成データは同数となっている．汎化性能検証データ（Wikinews，Wikipedia，Yahoo!知恵袋）についても同様に，Llama3 および Swallow-8B を対象として，人間と LLM のサンプルが対になるようデータセットとして構築した．

表 3.3: 論文の概要データセットの引用元論文集と抽出件数

引用元論文集名	抽出件数（人間）
土木学会論文集（A1, A2, F1–F6 分冊）	2,087
人工知能学会全国大会論文集	1,555
精密工学会学術講演会講演論文集	1,467
自動車技術会論文集	826
日本森林学会誌	544
都市計画論文集	534
日本教育工学会論文誌	477
農業農村工学会論文集	393
映像情報メディア学会技術報告	366
マーケティングジャーナル	253
日本科学教育学会年会論文集	154
合計	9,343

論文概要データセットは，合計 11 の学会・論文集から収集された 9,343 件の原文データで構成されている．最も大きな割合を占める土木学会論文集については，構造・地震工学（A1），応用力学（A2），トンネル工学（F1），地下空間研究（F2），土木情報学（F3），建設マネジメント（F4），土木技術者実践（F5），安全問題（F6）といった研究分野の論文の概要が含まれている．このように，論文誌が特定分野に偏らないように，分野の異なる複数の学会・論文集から論文の概要を収集し，文章の多様性を確保した．

第4章　データセットの分析

本章では，構築したデータセットの言語的特徴を分析し，人間生成テキストとLLM生成テキストの統計的な差異を明らかにすることで，識別に有用な特徴量を特定することを目指す．

2.1節で述べた先行研究（HC3やCHEATなど）では，英語や中国語のデータセットに対して言語的分析が行われている．例えばHC3では，品詞分布や依存構造を比較し，ChatGPTによる生成文は名詞や形容詞の出現頻度がわずかに高い点や，依存距離の分布に人間との差異が生じる点などを報告している．本研究ではこれらの報告を参考に，3章で作成した日本語LLM生成テキスト識別データセットに対して言語的特徴の分析を行い，人間生成テキストとLLM生成テキスト性質の違いを発見することを目指す．ここで得られた知見を整理し，5章における分類器の設計に活用することを最終的な目標とする．

本章の構成は以下の通りである．4.1節では，データセットの基礎統計として，平均文長，語彙豊富さなどの指標を算出して分析する．4.2節では，MeCabを用いた品詞の出現頻度の比較の分析について述べる．4.3節では，GiNZA（spaCy）を用いた文の構造的特徴の分析について述べる．4.4節では，2つの感情分析モデルを用いたテキストの感情分析結果について述べる．4.5節では，文埋め込みモデルを用いた人間生成テキストとLLM生成テキストに対する意味の差を分析した結果について述べる．最後に4.6節では，言語モデルによるパープレキシティ（PPL）を用いたテキストの予測容易性の観点からの分析について述べる．

4.1　文長と語彙の分析

本節では，構築したデータセットに含まれる人間生成テキストとLLM生成テキストについて，平均文字数，語彙の豊富さなどの両者の表層的特徴を確認する．指標は，平均文字数，平均文数，1文あたりの平均文字数，そしてType/Token比（以下，T/T比）の4つである．

平均文字数は，各テキストの文字数を合計し，ドメイン，LLM毎に平均を計算した指標である．平均文数は，1テキストに含まれる文の数の平均を集計した指標である．分割に用いる文末記号は,「。」「.」「.」「!」「?」「！」「？」とした．ただし,「。。。」「...」などの連続した文末記号に関しては，文末として扱わないようにした．1文あたりの平均文字数は，全テキストの文字数を文数で除算して算出した

指標である．T/T 比は語彙多様性を捉える指標として用いる．MeCab で形態素解析を行い，得られたトークンの種類数（異なり語数）を V，総トークン数を N_{tok} とし，T/T 比（%）を次式で定義する．この値が大きいほど，同じ長さのテキスト内で多様な語彙が使用されていることを示す．

$$\text{T/T 比 (\%)} = \frac{100 \times V}{N_{\text{tok}}} \tag{4.1}$$

表 4.1: 構築したデータセットの統計量（長さ・語彙多様性）

ドメイン	LLM	種別	平均文字数	平均文数	平均文字数/文	T/T 比 (%)
論文の概要	–	人間	321.47	5.01	64.07	1.52
	GPT-4o	LLM	373.38	6.41	58.19	1.24
	Gemini	LLM	374.42	6.43	58.16	1.30
	Llama3-8B	LLM	318.12	5.27	60.34	1.48
	Swallow-8B	LLM	371.25	6.48	57.25	1.27
Yahoo!知恵袋	–	人間	123.17	3.84	32.03	11.7
	Llama3-8B	LLM	473.11	13.60	34.78	4.84
	Swallow-8B	LLM	439.07	11.60	37.84	4.80
Wikinews	–	人間	541.13	9.87	54.81	3.96
	Llama3-8B	LLM	315.47	7.20	43.75	3.88
	Swallow-8B	LLM	368.30	8.34	44.12	3.27
Wikipedia	–	人間	319.47	5.12	62.30	5.44
	Llama3-8B	LLM	196.92	3.46	56.94	4.54
	Swallow-8B	LLM	333.27	6.42	51.87	2.66

表 4.1 に，人間生成テキストと LLM 生成テキストの基礎統計量（平均文字数，平均文数，1 文あたりの平均文字数，T/T 比）を示す．

まず平均文字数について述べる．論文の概要では，生成時に文字数を指定するプロンプト制御を導入しているが，LLM 生成テキストの平均文字数が人間テキストより多い傾向が見られた．これに対し，Wikinews および Wikipedia では，Wikipedia の Swallow-8B 以外では人間生成テキストの平均文字数が LLM 生成テキストを上回る傾向が観察される．Yahoo!知恵袋では，LLM 生成テキストの平均文字数が人間テキストを大きく上回っている．これは人間の回答では 20 文字程度の短文でもベストアンサーに選ばれる場合があるのに対し，LLM では必ず長文で返答を行うため，両者に大きな差が生じたと考えられる．

次に文の統計（平均文数と 1 文あたりの平均文字数）について述べる．論文の概要では，LLM 生成テキストは文数が多く，1 文あたりの平均文字数は人間生成テキストが大きい傾向が見られる．また Wikinews，Wikipedia では人間生成テキストの方が 1 文あたりの文字数がやや長く，Yahoo!知恵袋では逆に人間の方がやや短い傾向が見られた．全体的に多くのドメインでは人間がやや長く，Yahoo!知恵袋では LLM がやや長い傾向が見られた．

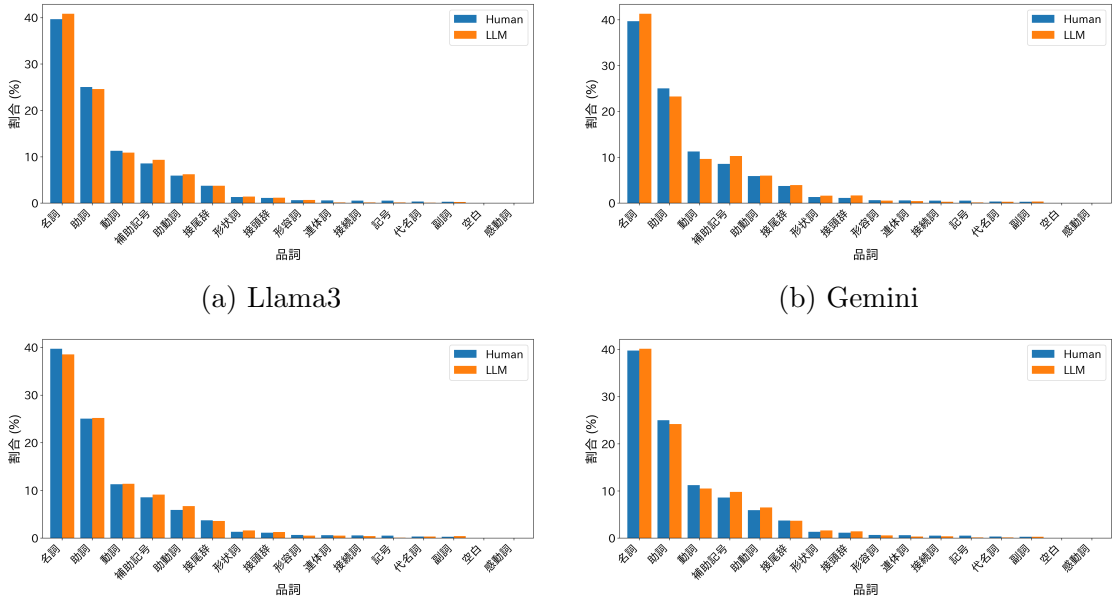
最後に T/T 比について述べる．論文の概要については，人間と LLM の間で数値に大きな差は見られなかった．これは，人間も LLM も同様に，入力元である論文の「Introduction」や「Conclusion」に含まれる単語を多く引用・使用して概要を作成するためであると考えられる．一方，他のドメイン（Yahoo!知恵袋，Wikinews，Wikipedia）については人間の方が T/T 比が高く，LLM と比較してより多様な単語を用いてテキストを作成していることがわかる．

以上より，平均文字数および文の統計に関する差異はドメインおよび LLM の種類によって大きく変動する一方で，T/T 比では人間生成テキストの方が語彙多様性が高い傾向が一貫して確認された．

4.2 品詞の分析

本節では，人間生成テキストと LLM 生成テキストの品詞の分布を比較する．

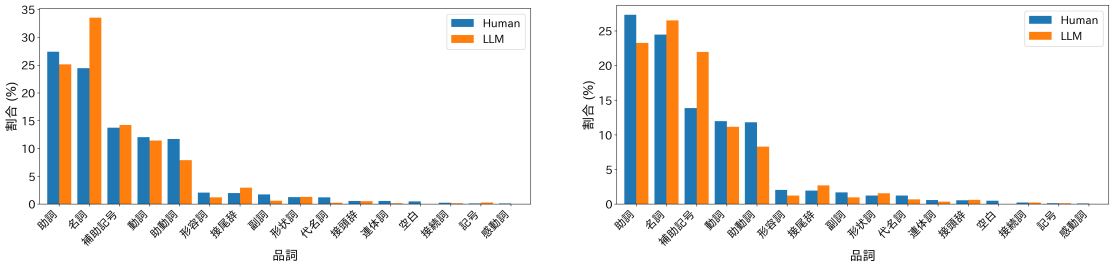
各テキストを形態素解析し，単語への分割と品詞付けを行う．形態素解析ツールとして MeCab[21] を用い，辞書として UniDic (CWJ) を使用する．MeCab の結果得られる品詞としては，(例：名詞，動詞，助詞，形容詞，助動詞，補助記号など）がある．そして，各 LLM・ドメイン毎に人間生成テキスト (Human) と LLM 生成テキスト (LLM) の各集合について，全テキストに含まれる総単語数に対する各品詞の出現頻度の割合を算出する．可視化では，品詞を横軸，割合（%）を縦軸とし，Human と LLM を並列に表示する．また，人間と LLM の差を分析しやすくするため，Human の割合が大きい品詞から順に並べて描画する．



(a) Llama3　(b) Gemini

(c) ChatGPT　(d) Swallow-8B

図 4.1: 論文の概要における品詞分布



(a) Llama3　　(b) Swallow-8B

図 4.2: Yahoo!知恵袋における品詞分布

図 4.1 および図 4.2 に，論文の概要と Yahoo!知恵袋における品詞分布（Human/LLM の比較）を示す．なお，Wikinews および Wikipedia における解析結果については付録 A.1 に示す．

まず論文の概要（図 4.1）では，4 つの LLM すべてにおいて，主要品詞の構成比が Human/LLM でほとんど同じである．このことから，論文の概要を対象としたときは，品詞の出現割合を LLM 生成テキストの識別指標として用いることは難しいことが分かる．次に Yahoo!知恵袋（図 4.2）では，Human と LLM の分布傾向は概ね一致しているものの，名詞に関しては LLM 生成テキストの方が Human と比較して出現頻度が高い傾向が確認できる．一方，Wikinews および Wikipedia（付録 A.1）については，論文の概要と同様に Human と LLM の分布が類似しており，識別の指標として用いることは難しいことが確認された．

以上の結果から，今回の品詞分布の分析では，大きな差は確認されなかった．これは，品詞の統計量は複数ドメイン・複数 LLM に共通する識別の有力な手がかりにはならないことを示している．

4.3　依存構造の分析

本節では，人間生成テキストと LLM 生成テキストの係り受け（依存構造のラベル）の分布を比較する．

係り受け解析には，spaCy 上で動作する日本語 NLP パイプラインである GiNZA を用いる [23]．GiNZA は Universal Dependencies（UD）に基づく解析器を spaCy のパイプラインとして提供しており，入力テキストを形態素解析した上で，各語に品詞および係り受け（依存関係）を付与できる．

依存構造解析では，各トークンが文中のどの語（係り先）に依存するかを推定し，係り先（head）と，その依存関係の種類を表す依存関係ラベル（dependency label）を付与する．依存関係ラベルは UD で定義される関係（例：`nsubj`, `obj`, `compound`）に対応する．各単語の依存関係の集合が文全体の依存構造（依存木）を構成する [5]．分析では，各 LLM・ドメイン毎に人間生成テキスト (Human) と LLM 生成テ

キスト (LLM) の各集合について，全テキストに含まれる総依存関係ラベル数に対する各ラベルの出現頻度の割合を算出する．

GiNZA による解析例を図 4.3 に示す．図中の矢印は語と語の修飾関係（依存関係）を表しており，矢印の上に記載されたラベルがその関係の種類を示している．例えば,「技術」から「進歩」への矢印にある `nsubj` は主語関係を表し,「の」から「自然言語処理」への矢印にある `case` は格表示（格助詞等）を表している．なお,各単語の下部に記載されているトークン（`PROPN`, `ADP`, `NOUN` 等）は品詞タグであり，本節で分析対象とする依存関係ラベルとは異なる．

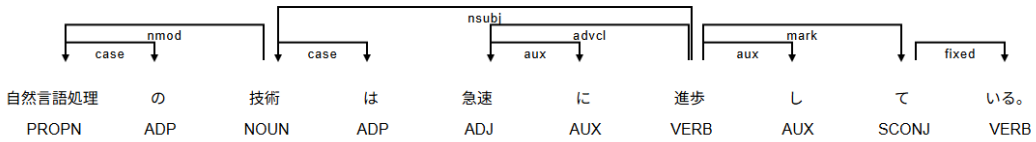
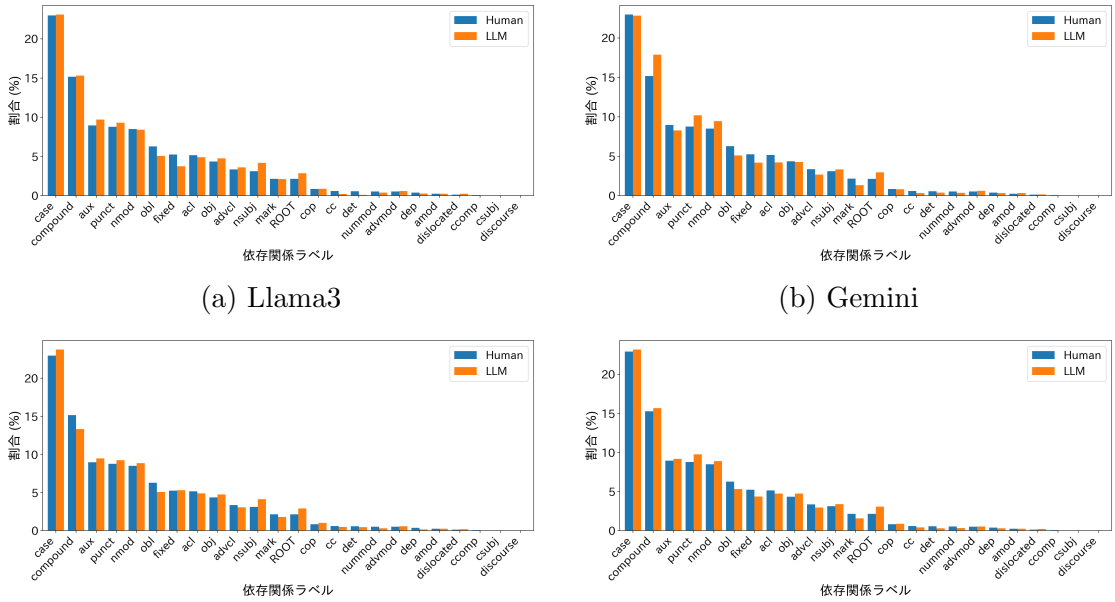


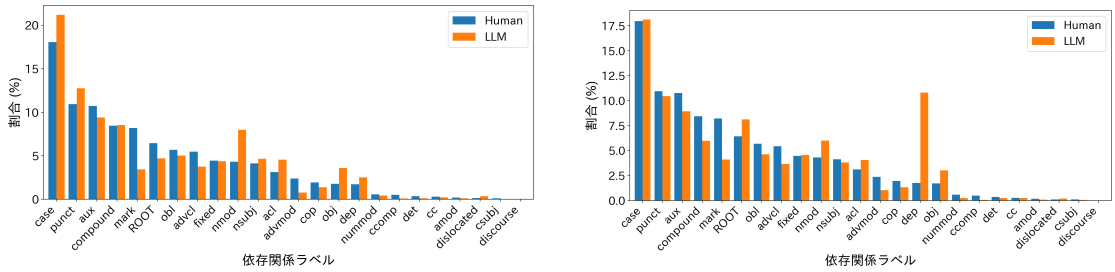
図 4.3: GiNZA による依存構造解析の例（UD 準拠）



(a) Llama3　(b) Gemini

(c) ChatGPT　(d) Swallow-8B

図 4.4: 論文の概要における依存関係ラベル分布



(a) Llama3　　(b) Swallow-8B

図 4.5: Yahoo!知恵袋における依存関係ラベル分布

図 4.4 および図 4.5 に，論文の概要と Yahoo!知恵袋における依存関係のラベルの分布を示す．なお，Wikinews および Wikipedia における解析結果については付録 A.2 に示す．これらの図では，横軸を依存関係ラベル，縦軸を出現割合（%）とし，人間生成テキスト (Human) と LLM 生成テキスト (LLM) を並列に表示した．また，各図は Human の割合が大きいラベルから順に並べ替えて描画した．

まず論文の概要（図 4.4）では，4 つの LLM すべてにおいて依存関係ラベルの構成比が人間と LLM とでほとんど差がなく，分布はほぼ重なっている．主要なラベル（例：`compound`，`nsubj`，`obj`，`nmod` 等）の順位と割合が Human と LLM で概ね一致して依存関係ラベルの統計量については，Human と LLM を明確に分離できるほどの差は確認できない．

一方で，Yahoo!知恵袋（図 4.5），Wikinews，Wikipedia（付録 A.2）では，論文の概要と比較すると一部の依存関係ラベルで Human と LLM との間に差が見られる．Yahoo!知恵袋では，`obj`，`dep` などが LLM の方が高く，`mark`，`advmod` などは Human の方が高い．Wikinews では `aux`，`obj` などが LLM の方で高く，`compound`，`nummod` などは Human の方で高い．Wikipedia では `nmod`，`aux` などが LLM の方で高く，`compound`，`nummod` などは Human の方で高い．

しかし，多くのラベルでは依然として近い割合を示している．また，差が見られたラベルも，ドメインや LLM によって一致せず，差の現れ方に一貫性がない．したがって，依存関係ラベルの統計量は品詞分布と同様に複数ドメイン・複数 LLM に共通する識別の有力な手がかりにはならない事を示している．

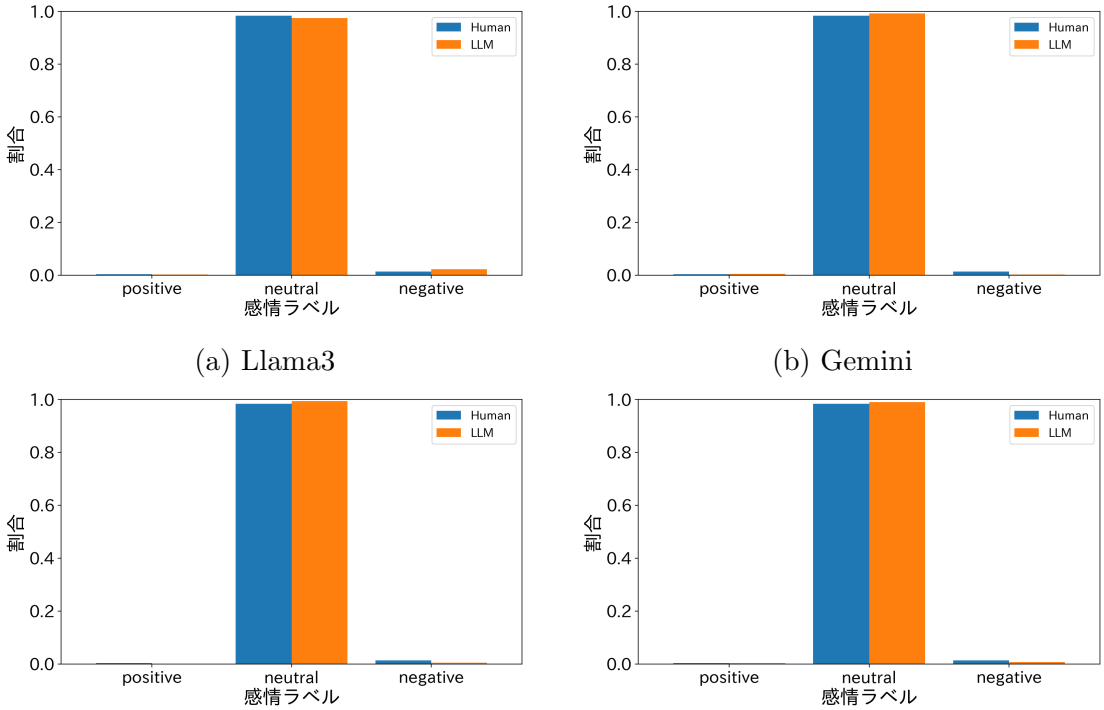
4.4　感情極性の分析

本節では，人間生成テキストと LLM 生成テキストに含まれる感情極性の分布を比較し，LLM 生成テキストが人間生成テキストと比較して特定の感情極性（肯定・中立・否定）へ偏る傾向があるかを確認する．感情ラベルの付与には，推論モデルの違いによる差も確認するため，以下の 2 種類の事前学習済み感情分類器を使用する．

- **cardiffnlp/twitter-xlm-roberta-base-sentiment**[3]（以下，XLM-R）
  XLM-R（XLM-RoBERTa）をベースとし，約 1.98 億件のツイートで学習された多言語モデルを感情分析向けにファインチューニングしたモデルである．出力は `Positive/Neutral/Negative` の 3 値である．このモデルは HC3 データセットで感情分析モデルとして使用されている．また，感情タスクのファインチューニングは 8 言語で実施されているが，このモデルの開発者はこれら以外の言語にも適用可能である旨が説明されているため，本研究でも日本語テキストに対する感情極性の推定に使用する．

- **lxyuan/distilbert-base-multilingual-cased-sentiments-student**[26]（以下，DistilBERT）
  多言語入力に対応した BERT 系の感情分類器である．本モデルは多言語の一般テキストを想定しており，日本語に対して安定した極性判定が期待できるため，1 つ目のモデルとは異なる感情分析モデルとして採用する．

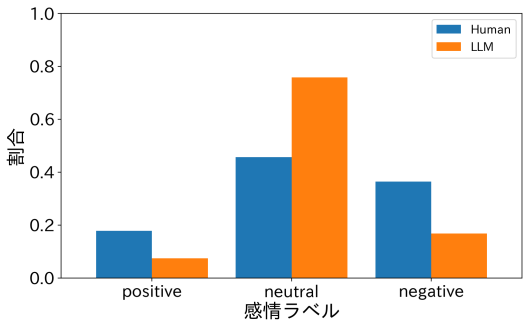
分析では，各ドメイン× LLM のテキストに対し，それぞれのモデルで感情分析を実行し，`positive/neutral/negative` の 3 値のいずれかに分類する．そして，人間生成テキスト (Human) と LLM 生成テキスト (LLM) の各集合について，全テキスト数に対する各感情ラベルの出現頻度の割合を算出する．



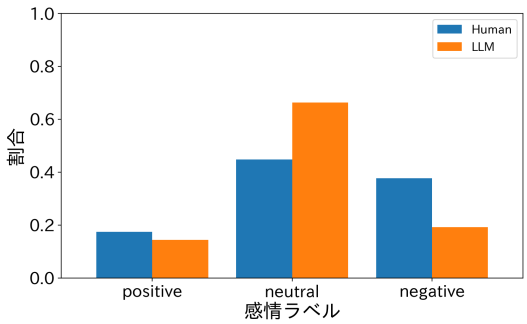
(a) Llama3　(b) Gemini

(c) ChatGPT　(d) Swallow-8B

図 4.6: 論文の概要における感情極性の分布（XLM-R）

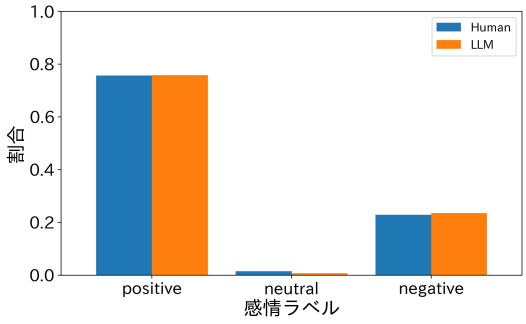


(a) Llama3

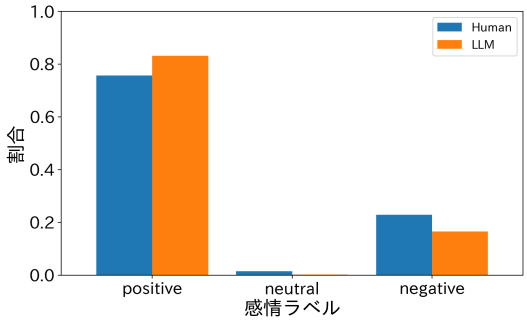


(b) Swallow-8B

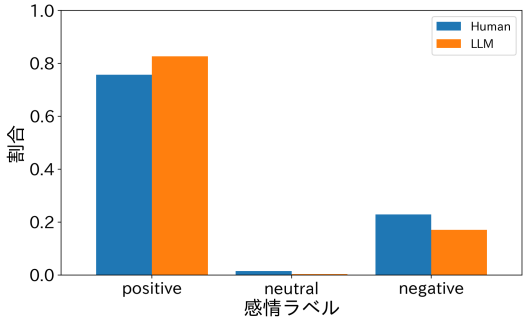
図 4.7: Yahoo!知恵袋における感情極性の分布（XLM-R）



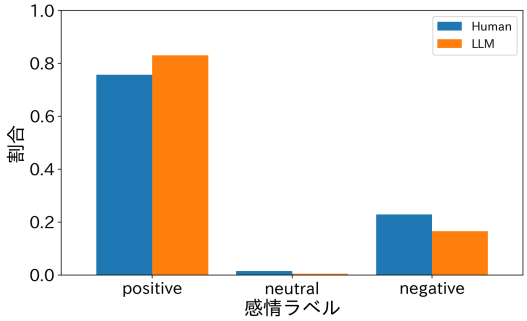
(a) Llama3



(b) Gemini

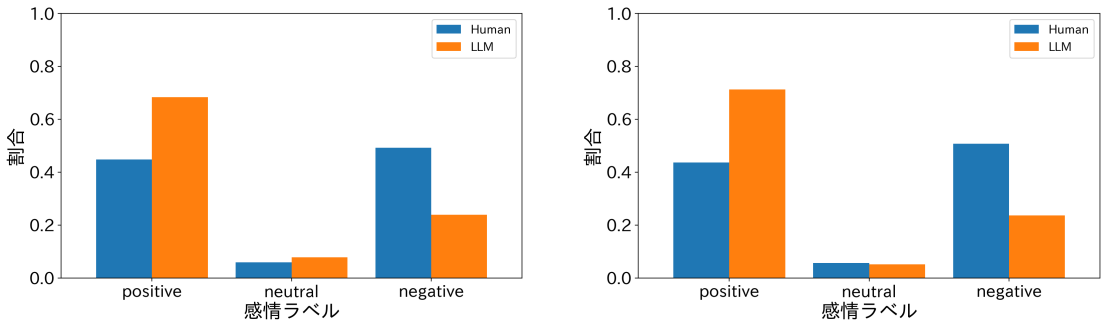


(c) ChatGPT



(d) Swallow-8B

図 4.8: 論文の概要における感情極性の分布（DistilBERT）



(a) Llama3　　(b) Swallow-8B

図 4.9: Yahoo!知恵袋における感情極性の分布（DistilBERT）

図 4.6 および図 4.7 に，XLM-R による感情分析の結果得られた感情ラベルの分布を示す．同様に，図 4.8 および図 4.9 に，DistilBERT による分析結果を示す．なお，Wikinews および Wikipedia における解析結果については付録 A.3 に示す．

まず XLM-R の結果を見ると，論文の概要（図 4.6）では，いずれの LLM においても Human も LLM も `neutral` のテキストが突出して多く，極性分布の差は視覚的にほとんど確認できない．一方で，Yahoo!知恵袋（図 4.7）では，LLM が `neutral` の割合が相対的に高く，Human は `negative` が相対的に高い傾向が見られる．付録 A.3 を参照すると，Wikinews では LLM の `neutral` の割合が Human と比較して低く，逆に `positive` や `negative` の割合が高くなる傾向が見られる．一方，Wikipedia では論文の概要と同様に `neutral` の割合が非常に高かった．

次に DistilBERT の結果では，全体を通して `neutral` がほとんど出現せず，`positive` と `negative` を中心とした分布になっている．論文の概要（図 4.8）では，いずれの生成モデルでも `positive` の割合が大きく，Human/LLM の差は相対的に小さい．これに対し，Yahoo!知恵袋（図 4.9），Wikinews，Wikipedia（付録 A.3）では，LLM は `positive` の割合が高く，Human は `negative` の割合が高い傾向が一貫して見られる．

以上より，データセットに対する感情極性分布の推定結果は感情分類モデルに強く依存しており，同一データに対しても分布形状が大きく異なることが分かる．その上で共通して言えるのは，論文の概要ドメインでは両推論器において Human と LLM の分布差が小さい点である．一方，Yahoo!知恵袋や Wikinews では，Human と LLM の分布差が確認できるものの，差の方向（`neutral` への偏りか，`positive` への偏りか）はモデルにより一致しない．

したがって，感情分布はドメインによっては Human と LLM の差が観察されるため LLM 生成テキスト識別の特徴量として利用できる可能性がある一方，差の現れ方がドメイン依存かつモデルで大きく変動するため，汎用的な識別指標として用いることは難しいことが確認できた．

4.5　文埋め込みによる分析

本節では，文埋め込みモデルを用いて各テキストをベクトル化し，人間生成テキストと LLM 生成テキストが埋め込み空間上でどの程度重なり，あるいは分離するかを確認する．

文埋め込みには，Sentence-BERT（SBERT）[35] に基づく日本語モデルである **sonoisa/sentence-bert-base-ja-mean-tokens-v2**[44] を用いる．SBERT は，意味的に類似した文同士を近くに，異なる文を遠くに配置するように学習されている．したがって，もし人間生成テキストと LLM 生成テキストに内容や意味的な傾向の差異があれば，埋め込み空間上でも両者の分布は分離すると期待される．なお，本節の分析においては，各データセットからランダムに抽出した 1/10 のデータを分析対象とした．

まず可視化として，得られた埋め込みベクトルを L2 正規化し，コサイン距離に整合した表現へ変換する．その上で次元削減（UMAP）により 2 次元へ射影し，散布図として描画する．散布図では Human と LLM を色分けして表示し，両クラスの重心も併記することで，全体的な配置の偏りや分離傾向を視覚的に確認できるようにする．

次に定量評価として，埋め込み空間における Human と LLM の分離度をシルエット係数 S により測定する．本分析では,『Human』および『LLM』の各クラスをそれぞれ一つのクラスタとして定義する．シルエット係数 S は，クラスタリングの評価指標であり，各サンプルが「自分のクラスタにどれだけ適切に属しているか」を，クラスタ内の凝集度とクラスタ間の分離度の両面から評価する指標である．今回の実験においては， S が大きいほど人間生成テキストと LLM 生成テキストが埋め込み空間上で別クラスタとして分かれやすいことを表し，S が 0 付近であれば両者の意味分布が重なりやすく，埋め込みだけでは分離が難しいことを表す．以下にシルエット係数の定義と本実験における設定を示す．

本研究では，各テキストの埋め込みベクトル x_i に対し，まず式 (4.2) に示す L2 正規化を施す．

$$\tilde{x}_i = \frac{x_i}{\|x_i\|_2} \tag{4.2}$$

シルエット係数は，この正規化後ベクトル $\tilde{x}_i$ からなる高次元の埋め込み空間（次元削減前）において算出する．正規化後ベクトル間のコサイン距離 $d_{\cos}(\tilde{x}_i, \tilde{x}_j)$ は式 (4.3) で与えられる．

$$d_{\cos}(\tilde{x}_i, \tilde{x}_j) = 1 - \tilde{x}_i^{\top} \tilde{x}_j \tag{4.3}$$

この距離に基づき，以下の手順でシルエット係数を算出する．

まず，サンプル i が属するクラスタを $C(i)$ とするとき，同一クラスタ内の平均距離 $a(i)$ は式 (4.4) で定義される．

$$a(i) = \frac{1}{|C(i)| - 1} \sum_{j \in C(i),\, j \neq i} d_{\cos}(\tilde{x}_i, \tilde{x}_j) \tag{4.4}$$

また，異なるクラス C に対する平均距離 $d(i, C)$ を式 (4.5) のように表す．

$$d(i, C) = \frac{1}{|C|} \sum_{j \in C} d_{\cos}(\tilde{x}_i, \tilde{x}_j) \tag{4.5}$$

これを用い，最近傍の他クラスタへの平均距離 $b(i)$ を式 (4.6) のように定義する．

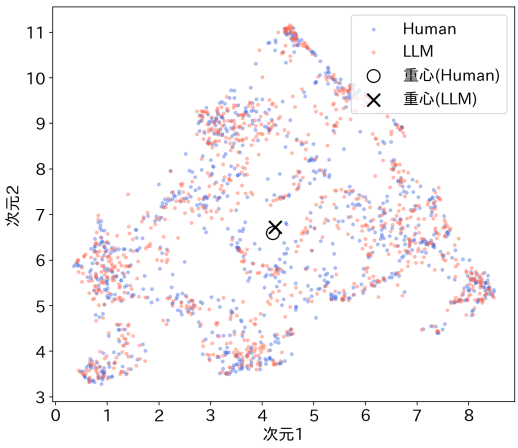
$$b(i) = \min_{C \neq C(i)} d(i, C) \tag{4.6}$$

以上の定義より，各サンプルのシルエット値 $s(i)$ は式 (4.7) で算出される．

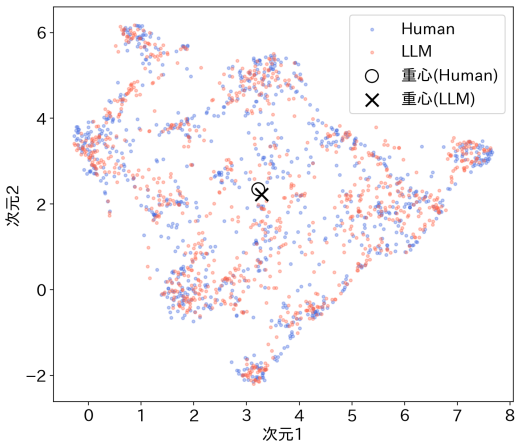
$$s(i) = \frac{b(i) - a(i)}{\max\{a(i),\, b(i)\}} \tag{4.7}$$

ここで，$s(i)$ は $-1 \leq s(i) \leq 1$ を満たす．$s(i)$ が大きいほど，同一クラスタ内で近く（$a(i)$ が小さい），異クラスから遠い（$b(i)$ が大きい）ことを意味する．最後に，式 (4.8) に示す全サンプルの平均 S がシルエット係数となる．本研究ではこの S をデータセットにおける人間生成テキストと LLM 生成テキストの意味的分離度指標として用いる．

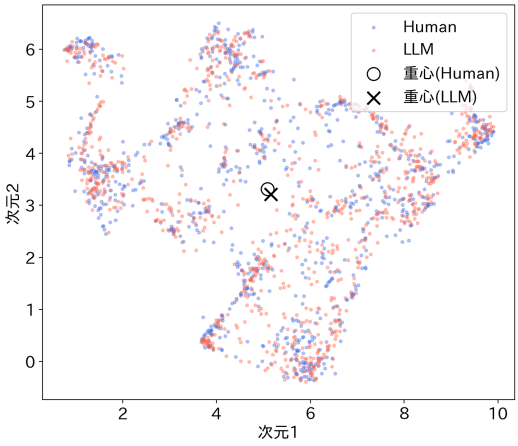
$$S = \frac{1}{N} \sum_{i=1}^{N} s(i) \tag{4.8}$$



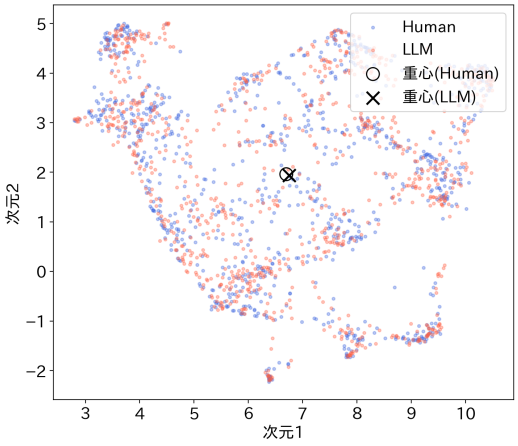
(a) Llama3



(b) Gemini

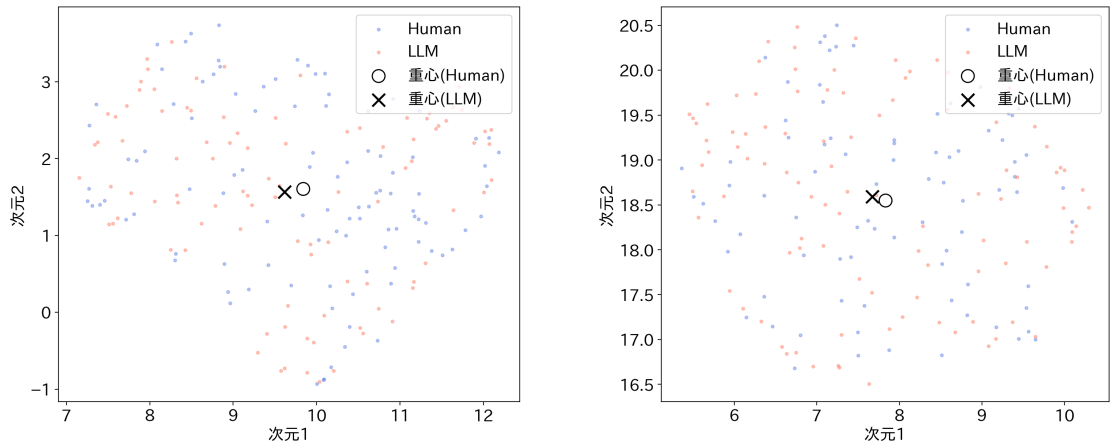


(c) ChatGPT



(d) Swallow-8B

図 4.10: 論文の概要における意味埋め込みの 2 次元可視化



(a) Llama3　　(b) Swallow-8B

図 4.11: Yahoo!知恵袋における意味埋め込みの 2 次元可視化

表 4.2: 文埋め込み（SBERT）に基づく Human/LLM の分離度（シルエット係数）

ドメイン	LLM	シルエット係数
論文の概要	ChatGPT	0.0195
	Gemini	0.0220
	Llama3	0.0053
	Swallow-8B	0.0129
Yahoo!知恵袋	Llama3	0.0283
	Swallow-8B	0.0145
Wikinews	Llama3	0.0075
	Swallow-8B	0.0103
Wikipedia	Llama3	0.0062
	Swallow-8B	0.0104

図 4.10 および図 4.11 に論文概要のデータセットと Yahoo!知恵袋のデータセットのサンプルを可視化した結果を示す．なお，Wikinews および Wikipedia における解析結果については付録 A.4 に示す．また，表 4.2 に，ドメインと LLM 毎のシルエット係数を示す．

可視化の結果を見ると，いずれのケースでも SBERT による埋め込み空間上で人間生成テキストと LLM 生成テキストの分布は大きく重なっていることが確認できる．また定量的にもシルエット係数は 0.0053〜0.0283 の範囲に収まっており，0 に非常に近い値であることから，人間生成テキストと LLM 生成テキストの意味分布は大きくは分離していないことが確認できる．

これは，文埋め込みモデルが捉える意味空間において両者の分布がほとんど一

致していることを示しており，LLM 生成テキストが人間のテキストと極めて類似した意味を持つことが確認できた．

4.6 PPL 分析

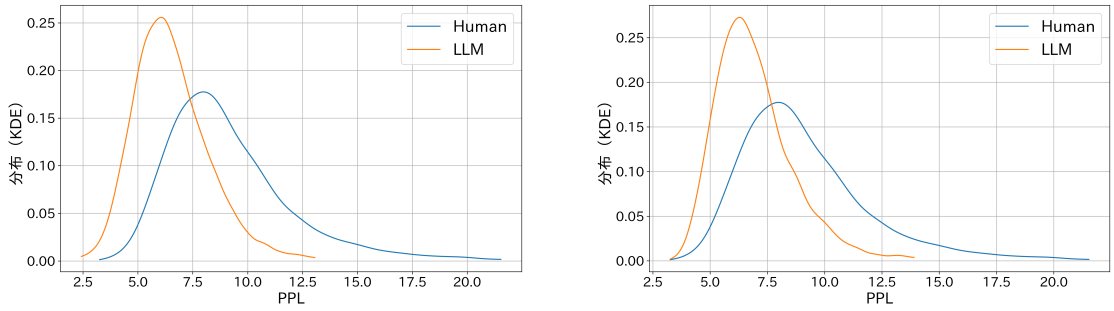
本節では，人間生成テキストと LLM 生成テキストに対して言語モデルの Perplexity（PPL）を算出し，両者で PPL に大きな差があるかを分析する．PPL は，対数尤度の負の平均を指数関数として表現した指標であり，次トークンの予測のしやすさを表すものである．LLM が次に出現するトークンを予測 PPL の値は低くなる．本節では，次式で定義される PPL を計算し，これを人間生成テキストと LLM 生成テキストで比較する．

$$\mathrm{PPL}(\mathbf{w}) = \exp\left(-\frac{1}{T}\sum_{i=1}^{T}\log p(w_i \mid w_{1:i-1})\right) \tag{4.9}$$

式 (4.9) に示すように，LLM が次トークンに対して高い確率を付与するほど PPL は小さな値となる．LLM は一般に確率の高いトークンを選択して文章生成を行うため，生成に用いたモデル自身（あるいは同等のモデル）で評価した場合，その PPL は低くなる傾向にある．実際，英語および中国語のデータセットである HC3 における検証でも，LLM 生成テキストの PPL は人間生成テキストと比較して顕著に小さく，両者の間に明確な分布差が存在することが報告されている．また，2.2 節で触れた DetectGPT などの強力なゼロショット手法は，この「生成モデル自身が自身の生成テキストに対して高い対数尤度（低い PPL）を与える」という確率的な性質を，利用して LLM 生成テキストを識別している．したがって，本節の分析において Human と LLM 間で明確な PPL 分布の差（Human より LLM が低 PPL になる傾向）が確認できれば，日本語の本データセットに対しても，PPL やトークン確率に基づくゼロショット識別手法が有効に機能することが期待できる．

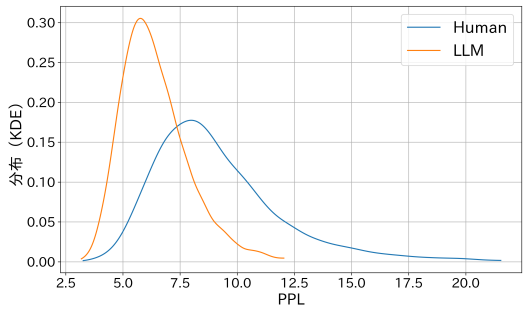
PPL の計算に用いる LLM としてはオープンモデルを用いる．GPT-4o や Gemini のような商用モデルはトークン確率の出力に API 利用コストを要するためである．まず，Llama3 と Swallow を PPL 計算モデルとして用いる．これらを Llama3 あるいは Swallow が生成したテキストに適用したときは，テキストを生成した LLM 自身によって PPT を測ることになる．一方，テキストを生成した LLM 以外の LLM を用いて PPL を計算する設定での検証を行うため，オープンで利用可能な日本語 GPT-2（rinna/japanesegpt2-medium[37]）も用いる．

各ドメイン・LLM のデータセットに対し，それぞれの計算モデルでテキストの PPL を算出し，Human と LLM で分布を比較する．可視化では，PPL の分布をカーネル密度推定（Kernel Density Estimation; KDE）により平滑化し，Human/LLM を同一図上に重ねて表示する．

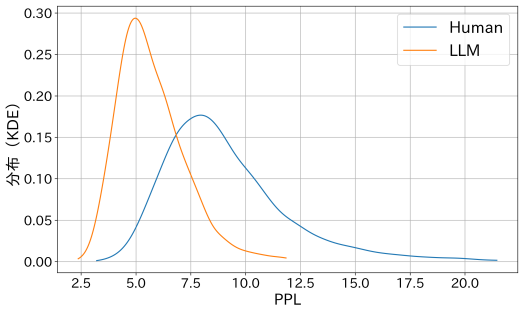


(a) Llama3

(b) Gemini

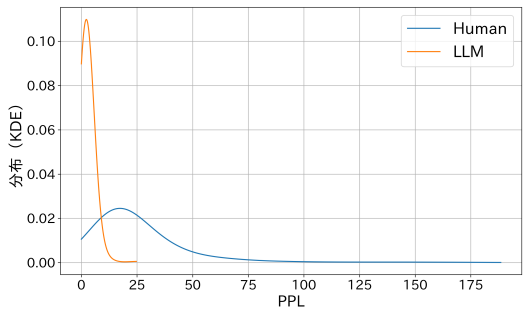


(c) ChatGPT

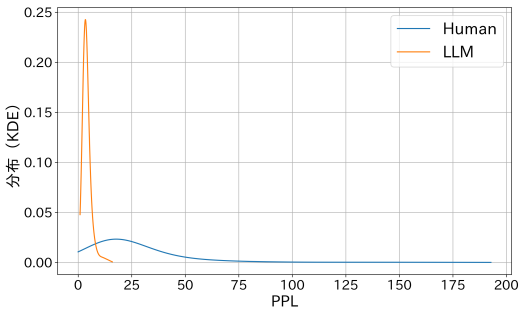


(d) Swallow-8B

図 4.12: 論文の概要の PPL 分布 (PPL は Llama3 で算出)

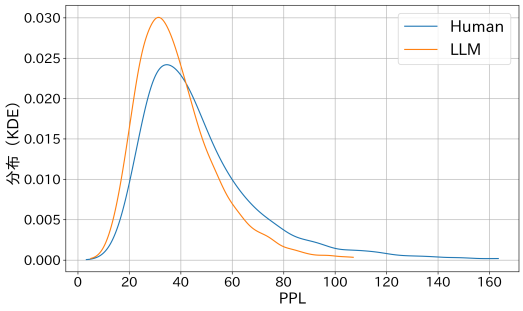


(a) Llama3

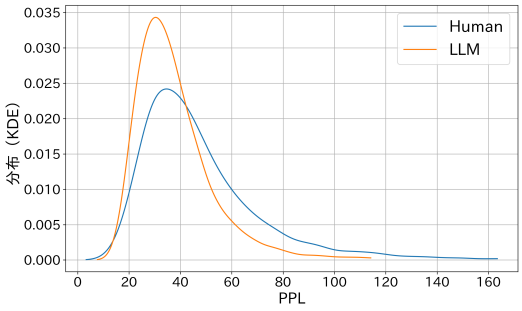


(b) Swallow-8B

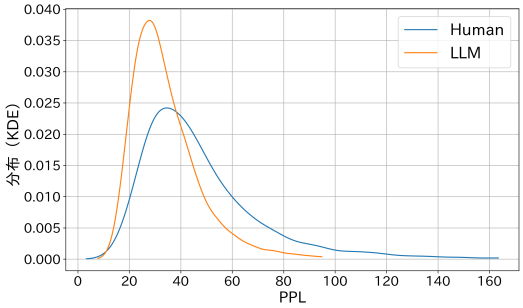
図 4.13: Yahoo!知恵袋の PPL 分布 (PPL は Llama3 で算出)



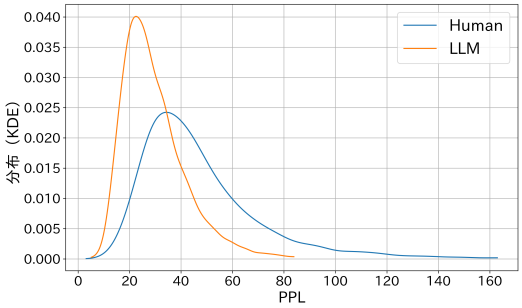
(a) Llama3



(b) Gemini

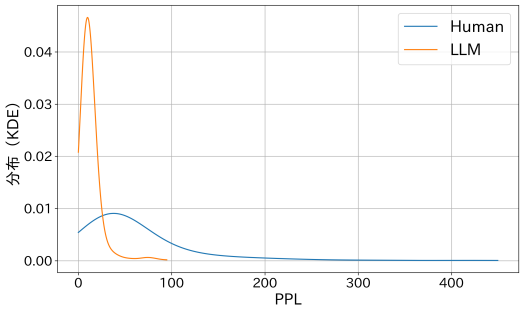


(c) ChatGPT

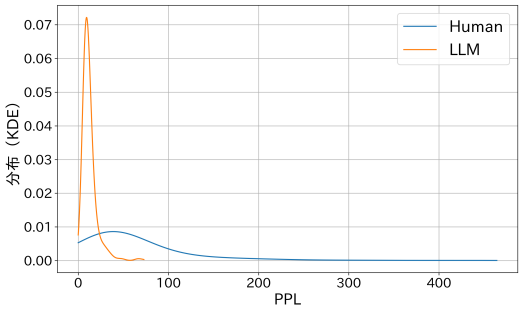


(d) Swallow-8B

図 4.14: 論文の概要の PPL 分布 (PPL は GPT-2 で算出)



(a) Llama3



(b) Swallow-8B

図 4.15: Yahoo!知恵袋の PPL 分布 (PPL は GPT-2 で算出)

図 4.12 および図 4.13 は，PPL 計算モデルを Llama3 としたときの論文の概要と Yahoo!知恵袋における PPL 分布（KDE）を示す．なお，Wikinews および Wikipedia における解析結果については付録 A.5 に示す．また，図 4.14 および図 4.15 は，PPL 計算モデルを日本語 GPT-2 としたときの論文の概要と Yahoo!知恵袋における PPL 分布を示す．なお，Wikinews および Wikipedia における解析結果については付録 A.5 に示す．また，PPL 計算モデルを Swallow-8B としたときの解析結果についても付録 A.5 に示す．

まず全体傾向として，いずれの PPL 計算モデルでも，LLM の分布が Human よ

り低い傾向が確認できる．特に Yahoo!知恵袋，Wikinews，Wikipedia では，LLM が低 PPL 付近に鋭く集中し，Human は高 PPL に裾が長い（右裾のロングテール）形が繰り返し現れている．一方，論文の概要では 3 つの PPL 計算モデルすべてにおいて Human と LLM の分布が相対的に近く，重なりが大きい．このことから，PPL は Human と LLM の識別に有効である一方，論文の概要のように両者の差が小さいドメインも存在することが分かる．

加えて，PPL 計算モデルが Llama3 または Swallow-8B の場合，テキストを生成した LLM と PPL 計算モデルが一致する条件では，論文の概要以外で Human と LLM の違いがより明瞭になる傾向が見られる（例：図 4.13(a) の Llama3 生成文）．これは，生成時にモデルが相対的に高確率なトークンを選びやすく，その結果として同一モデルで PPL を計算すると低 PPL になりやすいという PPL が低くなりやすいという一般的な傾向が，本分析においても確認されたといえる．

論文の概要で Human と LLM の差が小さい点については，プロンプト設計が関係していると考えられる．論文の概要では，Introduction と Conclusion をプロンプトに与えたが，他のドメインと比較して与えた情報が多く，文字数や文体まで指定している．この条件により入力プロンプトに含まれる単語や文体の影響を受け，結果として人間が書いた概要と PPL の分布が近くなったと推察される．これに対し，Yahoo!知恵袋，Wikinews，Wikipedia では，論文の概要と比べて生成時に与える入力情報が相対的に少なく，生成テキストの内容が LLM 自身が内在する知識に依存しやすい．その結果として LLM 側の PPL が低い領域に集中し，分布がよりはっきり分かれたと考察される．

第5章　LLM生成テキスト識別モデルの学習・評価

本章では，本研究で構築したデータセットを用いてLLM生成テキスト識別のための分類器を学習し，その性能を評価し，考察を行う．また，分類器がLLM生成テキストのどこに注目しているのかについても様々な実験を通じて分析する．

5.1節では，分類器の学習方法および評価方法を述べる．5.2節では，学習した分類器による評価結果とその考察を行う．5.3節では，分類器の内部に対する分析を通じて，分類器が何に注目して分類を行っているのかを考察する．

5.1　分類器の構築

本節では，3章で構築したデータセットを用いてLLM生成テキスト識別用分類器を学習・評価するための実験について説明する．2章の関連研究で紹介したように，既存の強力な識別方法として，事前学習済み言語モデルをLLM生成テキストと人間生成テキストのペアで構成されるデータセットを用いてファインチューニングし，分類器を構築する方法がある．本研究でも，それに倣い，日本語の事前学習済みRoBERTaモデルを本研究で構築したデータセットでファインチューニングすることで，分類器を構築する．本節では，データセットの分割方法，分類器の設定，評価指標の定義など，学習から評価までの実験設定について説明する．また，本研究では主に教師あり機械学習による分類手法について扱うが，ゼロショットのベースライン手法が本研究の日本語データセットに対してどの程度有効であるかについても議論を行う．さらに，人間がどの程度の精度でLLM生成テキストを識別可能かを確認するため，構築したデータセットの一部を人手でLLMが生成したテキストか否かを判定する実験を行う．本節では，その実験設定についても説明する．

本節の構成は以下の通りである．5.1.1項では，RoBERTaを用いた分類器の設定と学習・評価手順について述べる．5.1.2項では，既存のベースライン手法などRoBERTa分類器以外の分類器や評価方法について述べる．5.1.3項では，人手によるLLM生成テキスト識別の実験について説明する．

5.1.1 RoBERTa 分類器の構築

5.1.1.1 使用モデルと学習条件

本研究では，LLM 生成テキスト分類器のベースモデルとして，日本語コーパス（CC-100 および Wikipedia）を用いて事前学習された **rinna/japanese-roberta-base**[37, 40] を用いる．本モデルは標準的な RoBERTa-base[25] のアーキテクチャ（12 層，隠れ層次元数 768，Attention ヘッド数 12，パラメータ数約 1.1 億）に基づき構築されており，トークナイザに SentencePiece（語彙サイズ 32,000）が採用されている．

分類器の構築にあたり，本モデルの最終層に 2 値分類用の全結合層を付与し，LLM 生成テキスト（ラベル 1）と人間生成テキスト（ラベル 0）を識別するようファインチューニングを行う．入力データに関しては，RoBERTa モデルの最大長に満たないサンプルについては，パディング処理を施して入力を揃えた．また，学習サンプルの順序の影響を排除するため，各分割内でデータをシャッフルして学習に供する．また，欠損や最大入力トークン数超過により学習できないサンプルは事前に除外する．学習時の最適化手法およびハイパーパラメータの設定を表 5.1 に示す．学習は最大 3 エポック実施し，各エポック終了時に検証データを用いた正解率（Accuracy）による評価を行う．全エポックの中で最も正解率が高かったモデルを最終的な分類器として採用する．

表 5.1: RoBERTa 分類器の学習ハイパーパラメータ

項目	設定値
ベースモデル	`rinna/japanese-roberta-base`
タスク	2 値分類（Binary Classification）
ラベル定義	0: Human, 1: LLM
最大入力トークン長	512 token
最適化手法	AdamW
学習率 (Learning Rate)	2×10^{-5}
重み減衰 (Weight Decay)	0.01
バッチサイズ (Train/Eval)	4
学習エポック数	3
混合精度学習 (Mixed Precision)	fp16
モデル選択基準	Validation Accuracy

5.1.1.2 学習データの構成

本実験では，学習に使用するデータの組み合わせ（構成条件）を変化させることで，ドメインや LLM の違いが分類器の性能に与える影響を検証する．表 3.2 に

おける各行（ドメインと使用 LLM の組み合わせ）を本研究におけるデータ構成の最小単位と定義し，これを「サブセット」と呼称する．すなわち，本実験におけるサブセットの総数は 10 である．各サブセットには，対応付けされた人間生成テキストと LLM 生成テキストの両方が含まれる．

各サブセットを訓練・開発・テストに 8:1:1 の割合で分割する．いずれの条件でも訓練データで学習し，開発データでモデル選択を行った上で，テストデータで評価を行う．

実験条件として，様々な観点から以下の 5 種類を設定した．

(1) 全データセット総合（条件：All）　全てのサブセットを結合して学習データとする条件である．分類器は最も汎用的な識別性能を有すると期待されるため，これは本研究の基準となる．なお，本条件については学習時の初期値などを考慮し，異なるランダムシードで 3 回の試行を行い，その平均値で評価する．

(2) 1 種類のドメインのみ（条件：Domain）　特定のドメインに属するサブセット群のみを結合して学習する条件である．例えばドメインが「論文の概要」のときには該当する 4 サブセットを,「Wikinews」のときには該当する 2 サブセットを結合して用いる．これにより，訓練・開発データとテストデータとでドメインが異なるときのモデルの汎化性能を検証する．ドメインによりデータ数に差があることに注意する必要がある．

(3) 1 種類の LLM のみ（条件：LLM）　テキスト生成に用いた LLM を固定し，ドメインを横断して該当するサブセットを結合して学習する条件である．例えば「Llama3」条件では，4 つのドメインすべてから Llama3 を用いて生成されたサブセット（計 4 サブセット）を結合して用いる．ただし，ChatGPT と Gemini は論文の概要にしか使用していないため，この条件下では実験の対象としない．これにより特定の LLM に共通する特徴への依存度を検証する．

(4) 単一サブセットのみ（条件：Single）　10 個のサブセットのうち，単一のサブセットのみを用いて学習する条件である．

(5) 1 サブセット除外（条件：LOFO）　10 のサブセットから任意の 1 サブセットのみを除外し，残りの 9 サブセットを結合して学習する条件である（Leave-One-File-Out）．基準となる条件（All と比較することで，除外されたサブセットが識別性能の向上にどれだけ貢献するかを評価する．

5.1.1.3　評価方法と指標

本研究におけるタスクは，入力されたテキストが LLM によって生成されたもの（正例：ラベル 1）か，人間によって書かれたもの（負例：ラベル 0）かを判定する 2 値分類問題である．

モデルの識別性能を定量的かつ多角的に評価するため，以下の指標を採用する．なお，以降の定義において，TP（True Positive）は LLM 生成テキストを正しく LLM と判定した数，TN（True Negative）は人間生成テキストを正しく人間と判定した数，FP（False Positive）は人間生成テキストを誤って LLM と判定した数，FN（False Negative）は LLM 生成テキストを誤って人間と判定した数を表す．

また，各サンプルの予測ラベルの決定においては，モデルの出力ロジットにソフトマックス関数を適用して事後確率を算出し，識別閾値を 0.5 として判定を行う．すなわち，LLM 生成である確率が 0.5 を超える場合を予測ラベル 1，それ以外を予測ラベル 0 とする．

- **正解率 (Accuracy)**
  全データの中で，モデルの予測結果と正解ラベルが一致した割合である．

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}$$

- **精度 (Precision)**
  モデルが LLM 生成テキスト（陽性）と予測したデータのうち，実際に LLM 生成テキストであった割合である．誤検知の少なさを示す．

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

- **再現率 (Recall)**
  実際の LLM 生成テキスト（陽性）のうち，モデルが正しく識別できた割合である．見逃しの少なさを示す．

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

- **F1 値 (F1-Score)**
  精度と再現率の調和平均であり，両者のバランスを総合的に評価する指標である．

$$\text{F1} = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

- **AUROC (Area Under the ROC Curve)**
  AUROC は，閾値を変動させた際の ROC 曲線（Receiver Operating Characteristic curve）の下面積である．ROC 曲線は，識別閾値を連続的に変化させ

たときの真陽性率（TPR）と偽陽性率（FPR）の関係を表す曲線であり，横軸に $\mathrm{FPR} = \frac{\mathrm{FP}}{\mathrm{FP+TN}}$，縦軸に $\mathrm{TPR} = \frac{\mathrm{TP}}{\mathrm{TP+FN}}$ をとる．値は 0.5 から 1.0 の範囲をとり，1.0 は完全に分類できる理想的な状態を，0.5 はランダムな予測と同等の性能であることを示す．

なお，本研究において AUROC を採用する理由は，特定の識別閾値の設定に依存せず，モデルの識別能力を評価するためである．正解率などの指標は，モデルの出力確率が閾値（0.5）を超えたか否かで判定を行うため，識別に必要な特徴を学習できていたとしても，出力確率の偏りによって低い値となる可能性がある．一方で，AUROC は閾値を変動させて評価を行うため，正例に対し負例よりも高いスコアを与えられているかという「順序付けの性能」で評価できる．そのため，仮に正解率が低い場合であっても，AUROC が高い値を示していれば，モデルは識別に有効な特徴を捉えている可能性が高いと客観的に判断できる．

さらに，本研究では全 10 種類のサブセットを対象に評価を行う．各サブセットのテストデータに対する性能検証に加えて，全体の総合的な傾向を把握するために，以下の 2 種類の平均値を算出する．

- **マイクロ平均 (Micro Average)**
  全てのサブセットに対する予測結果と正解ラベルを統合し，全体を一つのデータセットとみなして評価した値である．データ数が多い論文の概要サブセットの影響を強く受ける．

- **マクロ平均 (Macro Average)**
  各サブセットに対して個別に指標を算出し，それらの数値の平均をとった値である．データ数の偏りに影響されず，各ドメインや LLM に対する性能を均等に評価できる．

5.1.2 ゼロショット識別手法

本項では，教師あり機械学習モデルとの比較対象として，追加の学習を必要としない「ゼロショット」の識別手法について述べる．本研究では，近年のゼロショット手法のベースラインとして以下の 4 手法を採用する．

5.1.2.1 基本的なゼロショット指標

本研究では，単純なゼロショットの手法として，以下の 4 つの統計的指標を採用する．これらの指標の算出には，すべて PPL 計算にも利用した GPT-2（**rinna/japanese-gpt2-medium**） を使用する．

これらの指標に基づく識別性能の評価には，閾値設定に依存しない指標である AUROC を用いる．Accuracy や F1 スコアなどの指標は判定のための閾値を決定

する必要があるが，ゼロショット手法において最適な閾値を事前に決定することは困難であることがその理由である．この指標でどの程度 LLM 生成テキストと人間生成テキストを識別する能力があるのか，定量的に評価する．

LLM 生成テキストか否かを判別する指標の定義は以下の通りである．ここで，$\mathbf{w} = (w_1, w_2, \ldots, w_T)$ はトークン列，$p(w_i \mid w_{1:i-1})$ はモデルが出力する条件付き確率，V は語彙集合，$I(\cdot)$ は条件が真のときに 1 となる指示関数を表す．

(1) 対数尤度 (Log-Likelihood)　テキストを構成する各トークンの，直前の文脈を与えられた条件下で出現する確率（条件付き確率）の対数を計算し，それをテキスト全体で平均した値である．この値は PPL と単調変換の関係にあり，数学的には PPL の対数をとり符号を反転させた量と等価である．したがって，AUROC による評価では PPL を用いた場合と同じ結果となる．一般に，LLM は自身の学習分布に近いテキストに対して高い確率を割り当てるため，LLM 生成テキストの対数尤度は人間生成テキストよりも高くなる傾向がある．

$$\text{Log-Likelihood} = \frac{1}{T} \sum_{i=1}^{T} \log p(w_i \mid w_{1:i-1}) \tag{5.1}$$

(2) 生成確率の順位 (Rank)　各トークン i において，モデルが予測した全語彙の確率分布の中で，実際に出現したトークン w_i が何番目に高い確率であったか（順位）を計算し，その平均をとったものである．一般に，LLM は確率の高い単語を優先して選択する傾向が強いため，生成されたテキストにおける平均順位は低くなる (1 位に近くなる) 傾向にある．

$$\text{Rank} = \frac{1}{T} \sum_{i=1}^{T} \sum_{w \in V} I(p(w \mid w_{1:i-1}) < p(w_i \mid w_{1:i-1})) \tag{5.2}$$

(3) 生成確率の対数順位 (Log-Rank)　Rank 指標に対して対数を適用したものである．順位の数値が極端に大きい場合の影響を緩和し，上位（1 位に近い）領域での順位差を捉えやすくなる．

$$\text{Log-Rank} = \frac{1}{T} \sum_{i=1}^{T} \log \left[\sum_{w \in V} I(p(w \mid w_{1:i-1}) < p(w_i \mid w_{1:i-1})) + 1 \right] \tag{5.3}$$

(4) エントロピー (Entropy)　各トークン位置 i においてモデルが出力する，次トークンの予測確率分布のエントロピー（平均情報量）をテキスト全体で平均した値である．モデルが予測した語彙 V 全体に対する確率分布の形状を評価する．分布が特定の単語に集中している（確信度が高い）場合はエントロピーが低くなり，逆に多くの単語に確率が分散している場合はエントロピーが高くなる．LLM

は一般に人間よりも確信を持って次トークンを予測する傾向があるため，LLM 生成テキストでは低い値を示すと報告されている．

$$\text{Entropy} = -\frac{1}{T}\sum_{i=1}^{T}\sum_{w\in V} p(w \mid w_{1:i-1}) \log p(w \mid w_{1:i-1}) \tag{5.4}$$

5.1.3 人手によるデータセットの評価

本研究で構築したデータセットにおいて，人間がどの程度正確に LLM 生成テキストと人間生成テキストを識別できるかを明らかにするため，人手による識別実験を実施した．機械学習モデルだけでなく，人間による評価を行うことで，データセットの難易度や，人間にとって識別が困難なドメインやモデルの傾向を把握することを目的とする．

5.1.3.1 評価用データの選定

評価対象とするデータは，前述の表 3.2 の 10 種類のサブセットである．教師あり学習において評価に用いた各サブセットのテストデータから，LLM 生成テキスト（正例）と人間生成テキスト（負例）がそれぞれ 25 件ずつとなるように，ランダムに 50 サンプルずつ抽出した．これにより，人手評価に使用したサンプルの総数は 500 件（10 サブセット × 50 サンプル）となる．

5.1.3.2 評価手順と評価者

評価実験は，本論文の著者を含む大学院生 3 名によって行われた．以下，これら 3 名の評価者を「評価者 A」,「評価者 B」,「評価者 C」と記す．なお，評価者 A（著者）はデータセット構築の過程でプロンプト設計や一部の生成結果を確認しているが，テストデータの内容を記憶しているわけではないため，他の評価者と同様に判定を行った．

評価にあたっては，各評価者にテキストの本文と，記事や論文のタイトルを提示した．評価者は，提示されたテキストに基づき，そのテキストが「人間生成テキスト」か「LLM 生成テキスト」かを 2 値で判定した．

5.1.3.3 評価指標

人間の識別能力をに評価するため，以下の 2 種類の指標を用いる．

- **正解率 (Accuracy)**
  各評価者ごとの正解率，および 3 名の平均正解率を算出する．

- **Fleiss' Kappa 係数**
  3 名の評価者間における判定の一致度を評価するために，Fleiss' Kappa 係数を用いる．Kappa 係数は -1 から 1 の値をとり，偶然による一致の影響を除いた上で評価者間の判定がどの程度一致しているかを示す指標である．1 は完全な一致を，0 は評価者間の一致が完全に偶然であることを表し，負の値は偶然よりも一致度が低いことを表す．この値が 1 に近いほど，評価者間の一致度が高いと解釈する．本実験では，この指標を用いることで，LLM 生成テキストの特徴が人間にとって共通して認識しやすいものか，あるいは評価者によって判断が分かれる曖昧なものかを分析する．

  Fleiss' Kappa 係数は以下の手順で算出される．ここで，N はサンプルの総数，n は評価者の人数，k はカテゴリ数（本実験では 2）を表す．また，n_{ij} は i 番目のサンプルをカテゴリ j と判定した評価者の数とする．

  まず，全判定においてカテゴリ j が選択された割合 p_j は次式で表される．

$$p_j = \frac{1}{Nn} \sum_{i=1}^{N} n_{ij} \tag{5.5}$$

  次に，評価者がランダムに判定を行った場合に偶然一致する確率（期待一致率）P_e は以下のように計算される．

$$P_e = \sum_{j=1}^{k} p_j^2 \tag{5.6}$$

  一方，実際に観測された一致率 P_o は，各サンプル i における評価者ペアの一致割合 P_i の全体平均として求められる．

$$P_i = \frac{1}{n(n-1)} \sum_{j=1}^{k} n_{ij}(n_{ij} - 1) \tag{5.7}$$

$$P_o = \frac{1}{N} \sum_{i=1}^{N} P_i \tag{5.8}$$

  最終的に，Fleiss' Kappa 係数 κ は，観測された一致率 P_o から偶然による一致率 P_e を補正した値として，以下のように定義される．

$$\kappa = \frac{P_o - P_e}{1 - P_e} \tag{5.9}$$

5.2 結果と考察

本節では，前節で述べた実験設定に基づき，構築した RoBERTa 分類器の評価結果を示すとともに，その結果から読み取れる特性について考察する．

本節の構成は以下の通りである．5.2.1 項では，全てのサブセットで分類器を学習する実験「条件:All」の結果について述べる．5.2.2 項では，学習データを特定のドメインや LLM に限定した場合や，特定のサブセットを除外した場合の評価結果を示し，モデルの汎化性能やドメイン依存性について検証する．5.2.3 項では，教師あり学習を行わないゼロショット識別手法による評価結果について述べ，5.2.4 項では自動識別と人間による識別を比較する．5.2.5 項では，RoBERTa 分類器に対する追加検証として，学習データ量の変化やドメイン間のデータ数不均衡の影響，および入力テキストへの摂動（パラフレーズ）に対する頑健性について述べる．最後に 5.2.6 項では，これら一連の実験結果を総括し，本研究で構築した分類器の特性について考察する．

5.2.1 全データセット学習分類器の評価

本研究で構築した 10 個のサブセットすべてを学習に用いた場合「条件:All」の結果を示す．表 5.2 に，各サブセットをテストデータとしたときの評価指標および全体の平均スコアを示す．

表 5.2: 全サブセット学習分類器の評価

ドメイン	LLM	Accuracy	Precision	Recall	F1	AUROC
論文の概要	Llama3	0.9543	0.9190	0.9964	0.9562	0.9957
	Gemini	0.9561	0.9193	1.0000	0.9579	0.9973
	ChatGPT	0.9561	0.9193	1.0000	0.9579	0.9974
	Swallow-8B	0.9551	0.9191	0.9979	0.9569	0.9963
Yahoo!知恵袋	Llama3	0.9967	0.9933	1.0000	0.9967	1.0000
	Swallow-8B	0.9871	0.9748	1.0000	0.9872	0.9999
Wikinews	Llama3	0.9690	0.9417	1.0000	0.9699	0.9998
	Swallow-8B	0.9774	0.9568	1.0000	0.9779	1.0000
Wikipedia	Llama3	0.9756	0.9547	0.9985	0.9761	0.9985
	Swallow-8B	0.9602	0.9263	1.0000	0.9617	0.9977
マイクロ平均		0.9594	0.9259	0.9988	0.9609	0.9969
マクロ平均		0.9688	0.9424	0.9993	0.9698	0.9982

まず全体を概観すると，すべてのドメインおよび LLM のサブセットにおいて，Accuracy と F1 スコアはいずれも 0.96 前後と非常に高い値を示した．これは，事

前学習済み RoBERTa モデルをファインチューニングする手法が，今回構築した日本語データセットでも有効に機能することを示している．

次に，それぞれの評価指標の関係に着目する．本結果では，すべてのサブセットにおいて再現率（Recall）が 1.0000 に近い値を示しているのに対し，精度（Precision）は若干低い傾向が見られた．これは，本モデルが LLM 生成テキストの見逃しをほぼ完全に防いでいる一方で，人間生成テキストの一部を誤って LLM 生成と判定していることを意味する．この傾向は先行研究の報告と同じであり，分類器がテキストを LLM 生成と判断しやすくなる傾向が，今回の日本語データセットの検証においても確認された．なお，AUROC はいずれの条件でも 0.99 以上と極めて高い値を示しており，判定閾値をどのように設定しても非常に高い識別能力を有していることが分かる．

さらに，ドメインおよび生成 LLM ごとの結果の違いを確認する．「論文の概要」は他のドメインと比較して Accuracy，F1，AUROC がいずれも相対的に低い値となった．一方，「Yahoo!知恵袋」では Accuracy がほぼ 1.0 に達しており，識別が容易であったことがうかがえる．また，生成 LLM の違いによる結果の差異は，ドメイン間の差異と比較して小さいことも確認できる．

5.2.2 様々な条件で学習した分類器の評価

本節では，「条件:All」以外の条件で学習データを構成した場合の評価結果について述べる．これにより，ドメインや LLM の違いが識別性能や汎化能力に与える影響について検証する．

5.2.2.1 特定のドメインまたは LLM のみを用いた場合（条件：Domain, LLM）

ドメイン（ジャンル）の違いと生成 LLM の違いが，それぞれ分類器の汎化性能にどのような影響を与えるかを検証する．ここでは，「論文の概要」ドメインのみで学習した場合（表 5.3）と，代表的な LLM である「Llama3」で生成したテキストのみで学習した場合（表 5.4）の結果を示す．なお，本項で扱う条件以外のドメインおよび LLM を用いた実験結果については付録 B.1 に掲載する．

表 5.3: 論文の概要のデータセットのみで学習した分類器の評価

ドメイン	LLM	Accuracy	Precision	Recall	F1	AUROC
論文の概要	Llama3	0.9706	0.9536	0.9893	0.9711	0.9967
	Gemini	0.9754	0.9540	0.9989	0.9759	0.9992
	ChatGPT	**0.9759**	0.9540	**1.0000**	**0.9765**	**0.9997**
	Swallow-8B	0.9711	0.9536	0.9904	0.9716	0.9980
Yahoo!知恵袋	Llama3	0.9397	**0.9780**	0.8990	0.9368	0.9865
	Swallow-8B	0.8785	0.9595	0.7889	0.8659	0.9718
Wikinews	Llama3	0.9126	0.8947	0.9341	0.9140	0.9717
	Swallow-8B	0.9429	0.9137	0.9783	0.9449	0.9904
Wikipedia	Llama3	0.8289	0.8627	0.7822	0.8205	0.9064
	Swallow-8B	0.9004	0.8577	0.9602	0.9061	0.9753
マイクロ平均		0.9569	0.9417	0.9741	0.9576	0.9933
マクロ平均		0.9296	0.9282	0.9321	0.9283	0.9796

表 5.4: Llama3 で生成したデータのみで学習した分類器の評価

ドメイン	LLM	Accuracy	Precision	Recall	F1	AUROC
論文の概要	Llama3	0.9647	0.9366	0.9968	0.9658	0.9991
	Gemini	0.7705	0.9002	0.6081	0.7259	0.9155
	ChatGPT	0.8737	0.9235	0.8148	0.8658	0.9593
	Swallow-8B	0.8181	0.9125	0.7034	0.7944	0.9363
Yahoo!知恵袋	Llama3	0.9698	**0.9429**	**1.0000**	0.9706	0.9988
	Swallow-8B	0.9558	0.9184	**1.0000**	0.9574	0.9984
Wikinews	Llama3	0.9645	0.9333	**1.0000**	0.9655	**0.9998**
	Swallow-8B	**0.9728**	**0.9485**	**1.0000**	**0.9735**	**0.9998**
Wikipedia	Llama3	0.9556	0.9184	**1.0000**	0.9574	0.9993
	Swallow-8B	0.9690	0.9417	**1.0000**	0.9700	0.9992
マイクロ平均		0.8797	0.9239	0.8273	0.8729	0.9630
マクロ平均		0.9215	0.9276	0.9123	0.9146	0.9806

表 5.3 から，同一ドメイン (テストデータが論文の概要のとき) に対しては極めて高い性能（Accuracy で 0.97 など）を示す一方で，それ以外のドメインに対しては全体的に性能が低下していることが確認できる．特に Wikipedia の Llama3 では最も性能低下が大きく,「条件:All」と比較して，Accuracy で 0.1305 ポイント，F1 で 0.1556 ポイント，AUROC で 0.0921 ポイントと大きく低下している事が確認できる．また，付録 B.1 に示す他のドメインのデータセットのみで学習した場合（表 B.1, B.2, B.3）においては，論文の概要のみで学習した分類器よりも，学習に使用

していないドメインに対してさらに大きく性能が低下する傾向が確認された．この要因として，ドメイン間の性質の違いに加え，学習データ量に大きな差があることが汎化性能に影響している可能性もあると考えられる．論文の概要のデータセットのサンプル数は約 75,000 件で最も大きいのに対し，Yahoo!知恵袋データセットでは約 3,500 件，Wikinews データセットでは約 7,300 件，Wikipedia データセットでは約 9,000 件となっており，Wikinews，Wikipedia のサンプル数は論文の概要の 10%程度，Yahoo!知恵袋のサンプル数が 5%程度となっている．

次に，表 5.4 を見ると，学習に使用していない Swallow-8B の生成テキストをテストデータとしたとき，Yahoo!知恵袋,Wikinews,Wikipedia において Accuracy 0.95 以上，F1 でも 0.95 以上と高い数値を維持している．特に Wikinews・Swallow-8B が最も高い性能を示すという現象も確認できる．一方で論文の概要をテストデータとしたとき，Gemini で Accuracy が 0.7705 ポイントと,「条件:All」と比較して 0.1856 ポイント低下するなど，Llama3 以外の LLM で生成したテキストのテストデータでは大きく低下していることが確認できる．ただし，付録 B の表 B.4 に示す Swallow-8B で生成したデータセットのみで学習した場合の結果を見ると，論文の概要ドメインの Llama3 サブセットに対しては Accuracy が 0.8898 と大きく低下しているものの，ChatGPT や Gemini に対しては Accuracy 0.97 以上を維持しており，性能の低下は起こっていない．このことから，学習に使用していない未知の LLM であっても，学習に使用した LLM によっては，性能低下がほとんど生じなかったり，向上するケースがあることが確認された．

以上の結果から，LLM 生成テキストの識別においては，テキストを生成した LLM およびドメインの両方が分類器の性能に影響を与えていることが確認できる．したがって，事前学習済み言語モデルをファインチューニングして汎用的な分類器を構築するためには，多様な LLM だけでなく，多様なドメインのデータを学習データに含めることが重要であるといえる．

5.2.2.2　単一サブセットのみを用いた場合（条件:Single）

次に,「条件:Single」の評価結果について述べる．ここでは，サブセット「論文の概要・Llama」のみで学習した場合（表 5.5）と，サブセット「Yahoo!知恵袋・Llama3」のみで学習した場合（表 5.6）の結果を示す．なお，これら以外の単一サブセットを用いた場合の結果については付録 B.2 に掲載する．

表 5.5: サブセット「論文の概要・Llama3」のみで学習した分類器の評価

ドメイン	LLM	Accuracy	Precision	Recall	F1	AUROC
論文の概要	Swallow-8B	0.7731	0.9337	0.5878	0.7214	0.9286
	Llama3	**0.9770**	0.9598	**0.9957**	**0.9774**	**0.9988**
	ChatGPT	0.8678	0.9490	0.7773	0.8546	0.9663
	Gemini	0.7442	0.9270	0.5300	0.6744	0.8957
Yahoo!知恵袋	Swallow-8B	0.6464	0.8824	0.3333	0.4839	0.8618
	Llama3	0.8744	**0.9744**	0.7677	0.8588	0.9694
Wikinews	Swallow-8B	0.7989	0.8873	0.6848	0.7730	0.9075
	Llama3	0.9262	0.9235	0.9286	0.9260	0.9777
Wikipedia	Swallow-8B	0.8894	0.8894	0.8894	0.8894	0.9579
	Llama3	0.9156	0.9269	0.9022	0.9144	0.9588
マイクロ平均		0.8451	0.9377	0.7391	0.8267	0.9338
マクロ平均		0.8413	0.9253	0.7397	0.8073	0.9422

表 5.6: サブセット「Yahoo!知恵袋・Llama3」のみで学習した分類器

ドメイン	LLM	Accuracy	Precision	Recall	F1	AUROC
論文の概要	Swallow-8B	0.5179	0.5090	**1.0000**	0.6746	0.6660
	Llama3	0.5179	0.5090	**1.0000**	0.6746	0.6091
	ChatGPT	0.5179	0.5090	**1.0000**	0.6746	0.6887
	Gemini	0.5179	0.5090	**1.0000**	0.6746	0.6427
Yahoo!知恵袋	Swallow-8B	0.9834	0.9677	**1.0000**	0.9836	**0.9988**
	Llama3	**0.9849**	**0.9800**	0.9899	**0.9849**	0.9984
Wikinews	Swallow-8B	0.5082	0.5041	**1.0000**	0.6703	0.6093
	Llama3	0.5109	0.5042	**1.0000**	0.6704	0.6752
Wikipedia	Swallow-8B	0.5022	0.5011	**1.0000**	0.6677	0.8771
	Llama3	0.5022	0.5011	**1.0000**	0.6677	0.6378
マイクロ平均		0.5345	0.5176	0.9998	0.6821	0.6799
マクロ平均		0.6063	0.5994	0.9990	0.7343	0.7403

表5.5の結果を見ると，学習データと同一のサブセット「論文の概要・Llama3」に対してはAccuracy 0.9770，F1 0.9774，AUROC 0.9988と極めて高い性能を示している．一方で，他のサブセットに対しては，同一ドメイン（論文の概要）であってもAccuracyが0.74〜0.87と性能が大きく低下していることが確認できる．テストデータのドメインが異なる場合 (論文の概要以外の場合) でも性能低下が見られる．さらに，またドメインとLLMの両方が異なる場合には性能低下の度合が大きいことも確認できる．

次に，表 5.6（Yahoo!知恵袋・Llama3 のみ）の結果を見ると，学習データと同一のドメインである Yahoo!知恵袋に対しては Accuracy 0.98 以上と極めて高い性能を示している．また，同じドメインで LLM が異なる Yahoo!知恵袋・Swallow-8B をテストデータとしたときも非常に高い指標が得られていることが特徴的である．しかし，他ドメイン（論文の概要，Wikinews，Wikipedia）に対しては Accuracy が 0.51 程度と，ほぼランダムな予測に近い値まで低下した．この性能低下の原因としては，Yahoo!知恵袋の文章特性を過剰に学習した可能性が考えられるが，それ以外にも学習データ量が「論文の概要・Llama3」と比較して約 1/10 と少ない点も無視できない要因であると考えられる．

さらに，付録 B.2 に掲載したその他の単一サブセットの結果（表 B.5〜B.12）も考慮し，単独のサブセットで学習した場合には共通するいくつかの傾向が確認できた．

- 学習に使用したサブセットと同じサブセットのテストデータに対しては，いずれの条件においても非常に高い性能を示した．
- 同一ドメインであれば，LLM が異なっても性能低下は比較的小さく，ある程度の識別能力が維持される傾向が見られた．
- 他ドメインに対しては，多くの条件で Accuracy が 0.5 程度まで低下し，識別が困難になるケースが散見された．

以上の結果から，単一サブセットのみでの学習では汎用的な分類器の構築は困難であり，多様なドメインのデータを訓練データに含めることが重要であるといえる．

5.2.2.3　特定のサブセットを除外した場合（条件：LOFO）

次に,「条件:LOFO」（Leave-One-File-Out）に基づき，特定のサブセットを除外して分類器を学習した場合の評価結果について述べる．この実験ではでは，除外されたサブセットに対する性能低下を確認し，汎化性能を検証することが主眼であるため，表中の太字は最高値ではなく，学習データから除外されたサブセットを示している．ここでは，サブセット「論文の概要・Llama3」を除外した場合（表 5.7）と，サブセット「Yahoo!知恵袋・Llama3」を除外した場合（表 5.8）の結果を示す．なお，それ以外のサブセットを除外した場合の結果については付録 B.3 に掲載する．

表 5.7: サブセット「論文の概要・Llama3」を除外して学習した分類器

ドメイン	LLM	Accuracy	Precision	Recall	F1	AUROC
論文の概要	**Llama3**	**0.9160**	**0.9440**	**0.8844**	**0.9132**	**0.9747**
	Gemini	0.9738	0.9502	1.0000	0.9744	0.9997
	ChatGPT	0.9738	0.9502	1.0000	0.9744	0.9997
	Swallow-8B	0.9722	0.9500	0.9968	0.9728	0.9987
Yahoo!知恵袋	Llama3	0.9749	0.9519	1.0000	0.9754	1.0000
	Swallow-8B	0.9834	0.9677	1.0000	0.9836	1.0000
Wikinews	Llama3	0.9699	0.9430	1.0000	0.9707	1.0000
	Swallow-8B	0.9810	0.9634	1.0000	0.9813	1.0000
Wikipedia	Llama3	0.9889	0.9825	0.9956	0.9890	0.9998
	Swallow-8B	0.9712	0.9456	1.0000	0.9720	0.9996
マイクロ平均		0.9630	0.9509	0.9764	0.9635	0.9948
マクロ平均		0.9705	0.9548	0.9877	0.9707	0.9972

表 5.8: サブセット「Yahoo!知恵袋・Llama3」を除外して学習した分類器

ドメイン	LLM	Accuracy	Precision	Recall	F1	AUROC
論文の概要	Llama3	0.9770	0.9685	0.9861	0.9772	0.9973
	Gemini	0.9818	0.9688	0.9957	0.9820	0.9995
	ChatGPT	0.9839	0.9689	1.0000	0.9842	0.9999
	Swallow-8B	0.9791	0.9686	0.9904	0.9794	0.9985
Yahoo!知恵袋	**Llama3**	**0.9849**	**0.9706**	**1.0000**	**0.9851**	**0.9996**
	Swallow-8B	0.9724	0.9474	1.0000	0.9730	0.9995
Wikinews	Llama3	0.9863	0.9733	1.0000	0.9864	0.9999
	Swallow-8B	0.9864	0.9735	1.0000	0.9866	1.0000
Wikipedia	Llama3	0.9867	0.9782	0.9956	0.9868	0.9983
	Swallow-8B	0.9801	0.9617	1.0000	0.9805	0.9999
マイクロ平均		0.9811	0.9688	0.9943	0.9814	0.9990
マクロ平均		0.9819	0.9679	0.9968	0.9821	0.9992

まず，表 5.7（論文の概要・Llama3 除外）の結果を見ると，除外された「論文の概要・Llama3」のテストデータに対する Accuracy は 0.9160 となり，全データを使用した場合（条件:All）の 0.9543 と比較して，約 0.04 ポイントの低下が見られた．一方で，その他のサブセット（Gemini や ChatGPT など）に対しては，Accuracy 0.97 以上と極めて高い性能を維持している．

次に，表 5.8（Yahoo!知恵袋・Llama3 除外）の結果を見ると，除外された「Yahoo!知恵袋・Llama3」のテストデータに対する Accuracy は 0.9849 と極めて高く，「条

件:All」と比較しても低下は約 0.01 ポイントとわずかであった．さらに，付録 B.3 に掲載したその他のサブセットを除外した場合も含め，LOFO 実験全体からいずれのサブセットを除外した場合でも,「条件:All」と比較して，除外されたサブセットに対する評価の低下はわずかである傾向が確認できた．つまり複数の LLM，ドメインのテキストを組み合わせることで未知のデータに対する汎化性能が維持される．

一方，平均の結果を確認した場合,「条件:LOFO」の方が平均の性能が「条件:All」と比較して良い場合も存在することが確認できた．これは主に学習時の初期乱数やドロップアウトといったランダム要素に起因する結果であると考えられる．実際に,「条件:All」で学習した場合であっても，初期乱数を変更するだけで Accuracy に ±0.02 ポイント程度の差異が生じることが確認されている．一方で，特定のサブセットが学習過程において一種のノイズとして働き，識別性能の最適化を妨げていた可能性も考えられる．より厳密な要因分析を行うためには，分類器の学習と評価を何回か行い，それらの試行の指標の平均値で評価する必要がある．

特定のサブセットを除外した場合（条件:LOFO）の評価結果を以下にまとめる．

- 除外したサブセットのテストセットに対する識別性能の低下は小さい，つまり複数の LLM，ドメインのテキストを組み合わせて学習を行う事で未知のデータに対する汎化性能が維持される．
- 評価指標が全データ学習時（条件:All）を上回るケースも見られたが，これは初期乱数等のランダム要素や，一部のデータがノイズとして機能していた可能性が考えられる．

5.2.3 ゼロショット識別手法の評価

本項では，教師あり学習を行わないゼロショット手法による LLM 生成テキスト識別実験の結果について述べる．RoBERTa をベースとした教師あり学習と比較して，学習データを必要としないこれらの手法が，日本語テキストに対してどの程度の識別性能を示すかを検証する．

まず，言語モデルの出力確率に基づく 4 つの基本的なゼロショット指標（対数尤度，Rank，Log-Rank，Entropy）に基づく LLM 生成テキスト識別手法について，それぞれの AUROC を測り，評価する．本実験では，スコア算出に用いる言語モデルとして，GPT-2(**rinna/japanese-gpt2-medium**) と，データセットの生成モデルの一つである Llama3(**elyza/Llama-3-ELYZA-JP-8B**) を用いる．一般に，識別対象となるテキストを生成したモデルと，スコア算出に用いるモデルが一致している場合，モデルは生成時の確率分布を他モデルよりも正確に推定可能だと考えられる．そのため，他モデルを算出に用いる場合と比較して識別精度が向上する傾向にある．本実験はこのことを検証する．

表 5.9: 基本的なゼロショット指標による LLM 生成テキスト識別手法の評価（算出モデル：GPT-2）

ドメイン	LLM	対数尤度	Rank	LogRank	Entropy
論文の概要	Llama3	0.6281	0.5716	0.5706	0.5336
	Gemini	0.6470	0.6194	0.6205	0.5039
	ChatGPT	0.7085	0.6357	0.6345	0.5472
	Swallow-8B	0.7833	0.6624	0.6575	0.6126
Yahoo!知恵袋	Llama3	0.9149	0.8149	0.7864	**0.8838**
	Swallow-8B	0.9077	0.8169	**0.7944**	0.8460
Wikinews	Llama3	0.9357	0.7768	0.7496	0.5761
	Swallow-8B	**0.9831**	**0.8185**	0.7770	0.6948
Wikipedia	Llama3	0.7680	0.6442	0.6370	0.4316
	Swallow-8B	0.9341	0.7601	0.7167	0.6491
マイクロ平均		0.7128	0.6514	0.6448	0.5611
マクロ平均		0.8210	0.7121	0.6944	0.6279

表 5.10: 基本的なゼロショット指標による LLM 生成テキスト識別手法の評価（算出モデル：Llama3）

ドメイン	LLM	対数尤度	Rank	LogRank	Entropy
論文の概要	Llama3	0.7658	0.6313	0.6311	0.6228
	Gemini	0.7669	0.6159	0.6154	0.7019
	ChatGPT	0.8352	0.6851	0.6845	0.7391
	Swallow-8B	0.8762	0.6859	0.6852	0.7938
Yahoo!知恵袋	Llama3	0.9782	**0.9346**	**0.9339**	0.9614
	Swallow-8B	0.9779	0.8852	0.8849	**0.9626**
Wikinews	Llama3	0.9887	0.8788	0.8774	0.8092
	Swallow-8B	**0.9996**	0.8719	0.8704	0.8715
Wikipedia	Llama3	0.9887	0.7090	0.7081	0.6799
	Swallow-8B	0.9934	0.8458	0.8441	0.8748
マイクロ平均		0.8344	0.6907	0.6900	0.7351
マクロ平均		0.9171	0.7744	0.7735	0.8017

スコア算出モデルとして GPT-2 を用いたときの結果を表 5.9 に，Llama3 の結果を表 5.10 に示す．これらの結果から以下の傾向が確認できる．

まず，スコア計算モデルとしてどちらのモデルを使用した場合でも，RoBERTa をファインチューニングする教師あり手法と比較すると性能が低いことが確認された．一方で、Llama3 をスコアの計算に使用したとき，論文の概要以外では AUROC

で 0.97 ポイント以上と，RoBERTa 分類器と比較するとやや劣るものの十分に高い識別性能があることが確認できる．以上をまとめると，これら基本的なゼロショット手法は，RoBERTa 分類器には劣るものの，ドメインやスコア算出モデルによってはそれに近い識別性能を有している可能性が確認できる．

また，スコア計算モデルの違いによる影響を確認すると，GPT-2 より Llama3 を計算に使用した方が，全データセットのテストデータにおいて識別精度が大幅に高い傾向が確認された．この要因として，スコア計算モデルの性能が高いほどテキストの確率分布をより正確に近似可能となり，LLM 生成テキスト特有の統計的な偏りをより敏感に検知できるようになった可能性が考えられる．また，Llama3 でスコアを計算した場合，生成モデルが Llama3 のサブセットに対して性能が向上することが期待されたが，Llama3 より Swallow-8B で生成されたサブセットに対して高い性能を示す傾向が確認された．これは 4.6 節の PPL 分布から分かるように，Llama3 生成テキストは PPL 計算モデルに Llama3,Swallow-8B のどちらを使用した場合でも，他 LLM で生成されたデータセットと比較すると人間と LLM で分布傾向が近くなる．これは，テキストの識別難易度が高いことを意味しており，このことが Llama3 サブセットに対する性能低下の原因となった可能性が考えられる．

次に，ゼロショット評価指標の優劣については，モデルによらず対数尤度（Log-Likelihood）が一貫して最も高い性能を示した．また，GPT-2 では Entropy 指標の AUROC が低いが，Llama-3 ではマクロ平均で 0.8017 ポイントまで向上し，4 つの指標のうち 2 番目に良い成績であった．したがって，ゼロショット評価指標の算出に用いるモデルが異なると，評価指標の優劣も異なることが確認された．

また，ゼロショット識別手法の性能はドメインによって明確な差が見られる．論文の概要以外のドメインについて，指標を Llama3 で計算した場合，AUROC 0.97 から 0.99 という極めて高い値を示した．一方，論文の概要についてはも AUROC は 0.76 から 0.87 程度に留まり，他のドメインと比較して識別難易度が高いことが分かる．この要因の一つとして，論文の概要はプロンプトで人間の書いた文章の一部を与えることで，生成テキストが人間の記述と類似し，他のドメインよりも LLM 生成テキストと人間生成テキストの差が小さくなった可能性が推察される．

以上のゼロショット手法による評価結果のまとめを以下に示す．

- RoBERTa をファインチューニングする教師あり学習手法よりも性能が低いが，ドメインによってはそれと同程度の高い識別性能を示す．
- ゼロショット指標の中では対数尤度が一貫して高い性能を示した．
- ドメインによって識別難易度が大きく異なり，それはプロンプトの設計が影響している可能性が考えられる．

5.2.4 人手による LLM 生成テキスト識別の評価

本項では，人間がテキストを読んで LLM が書いたか人間が書いたかを判定したときの結果について述べる．表 5.11 に，3 名の評価者による正解率（Accuracy）および評価者間一致度（Fleiss' Kappa）を示す．なお，評価者 A は本論文の著者であり，評価者 B および C はデータセット構築に関与していない大学院生である．

表 5.11: 人手による LLM 生成テキスト識別の結果

ドメイン	LLM	Acc(A)	Acc(B)	Acc(C)	Avg Acc	Fleiss' κ
論文の概要	ChatGPT	0.78	0.66	0.54	0.6600	0.1202
	Gemini	0.62	0.62	0.56	0.6000	-0.0250
	Llama3	0.66	0.56	0.48	0.5667	-0.1398
	Swallow-8B	0.66	0.62	0.54	0.6067	0.2027
Yahoo!知恵袋	Llama3	0.94	0.80	0.94	0.8933	0.6199
	Swallow-8B	0.90	0.70	0.84	0.8133	0.4117
Wikinews	Llama3	0.58	0.48	0.42	0.4933	-0.0398
	Swallow-8B	0.70	0.48	0.64	0.6067	-0.0417
Wikipedia	Llama3	0.64	0.30	0.36	0.4333	0.1115
	Swallow-8B	0.84	0.36	0.56	0.5867	-0.0775
平均		0.732	0.558	0.588	0.6260	0.1142

表 5.11 の結果より，全体的な正解率および Fleiss' κ 係数の低さが確認できる．全体の平均正解率は 0.626 であり，ランダムな予測（0.50）をわずかに上回る程度に留まった．特に，データセット構築に関する知識を持たない評価者（B, C）においては，Wikinews や Wikipedia などのドメインで正解率が 0.5 を下回るケースも散見される．また，Fleiss' Kappa 係数は平均で 0.114 と極めて低く，Yahoo!知恵袋以外のドメインでは 0 近辺あるいは負の値を示しており，人間の判断がほとんど一致していないことを意味している．

次に，ドメインによる正解率の差が大きい点が確認できる．論文の概要，Wikinews，Wikipedia のドメインでは，正解率が低く，Kappa 係数が 0 近辺あるいは負の値を示しているのに対し，「Yahoo!知恵袋」に関しては，全ての評価者で 0.7〜0.94 の高い正解率を示しており，Kappa 係数も 0.41〜0.62（中程度〜かなりの一致）と，評価者間で判断が一致しやすい傾向が見られた．

このような結果から，人間による LLM 生成テキストの識別は非常に難しいことが確認できた．それにもかかわらず，本研究で構築した RoBERTa 分類器は，同一のテストデータに対して「条件:All」で Accuracy 0.96 以上，AUROC 0.99 以上という極めて高い性能を達成している．これは人間には難しい特徴を RoBERTa 分類器が学習して LLM 生成テキストを識別していること意味しており，分類器が何に注目しているのか，検証する必要性がある．

5.2.5 RoBERTa 分類器に対する追加検証

本節では，本研究の主な提案手法である RoBERTa を用いた教師あり分類器が，学習データの量，特定のドメインの偏りに依存しているかを明らかにするために，追加の検証実験を行う．具体的には，学習データ数を段階的に削減した場合の挙動や，サブセット間のデータ数を統一した場合の性能変化，そして入力テキストに対する摂動（パラフレーズ）への耐性について分析する．

5.2.5.1 学習データ量が識別性能に与える影響の調査

学習データの量とモデル性能の関係を検証する．本実験では，全学習データ「条件:All」からランダムにサンプリングを行い，各サブセットの訓練セットのデータ量を 50%,25%, 10%, 5%, 1%, に減少させて分類器を学習した．評価指標として，全サブセットのテストセットを結合し，Accuracy，F1 スコア，AUROC のマイクロ平均を算出した．学習データ量に対するこれら 3 つの評価指標の推移を図 5.1 に示す．

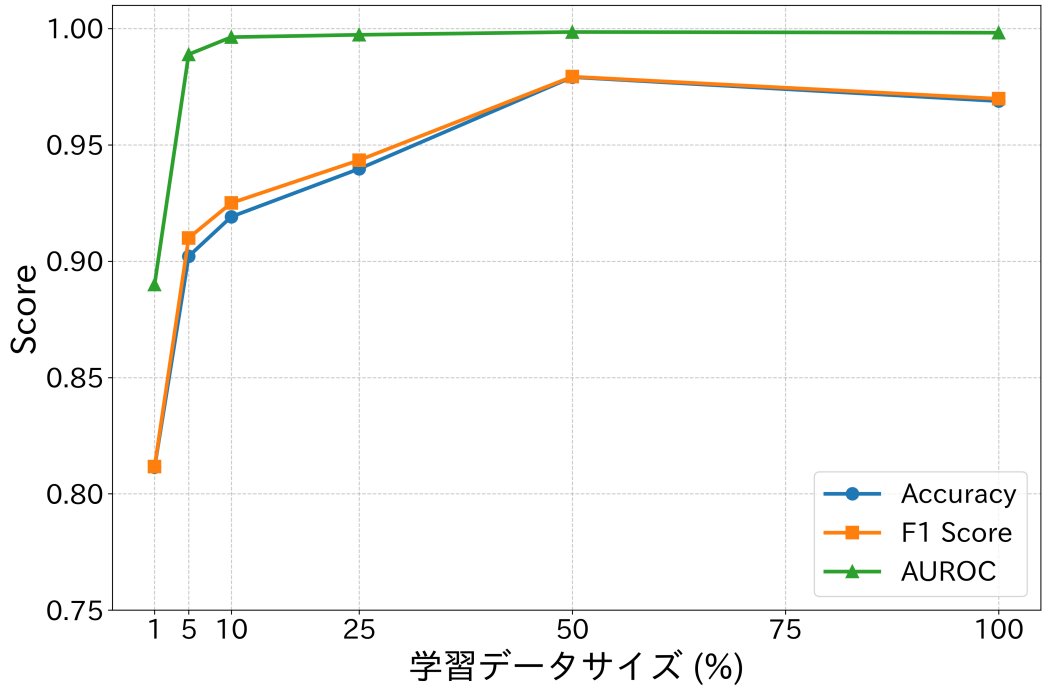


図 5.1: 学習データサイズと識別性能の推移（1%〜100%）

図 5.1 の結果より，全体的な傾向として，学習データ数が減少するにつれて識別性能は低下しており，LLM 生成テキストを正確に識別するには十分なデータ量が必要であることが確認できる．データ量が 1%（訓練データが約 750 件）の場合，Accuracy は約 0.81，AUROC は約 0.89 であり，一定の識別能力は有しているもの

の，実用には不十分な精度である．ここからデータ量を 5%，10%と増加させるにつれて，Accuracy は 0.90，0.92 と順調に向上しており，データ量の増加がモデルの性能向上と関係していることが確認できる．

データ量を 25%まで増加させると Accuracy は約 0.94 に達し，100%使用した場合（0.9688）との差は 0.03 ポイント以内に縮まる．さらに 50%の時点では Accuracy 0.9791 となり，訓練データ量が 100%のときの結果を上回るケースすら見られた．これは学習時のランダム性による影響だと考えられる．このことから，本実験の条件下においては，全データの 25%〜50%程度のデータ量があれば主要な識別能力は獲得され，それ以上のデータ追加による性能向上の幅は限定的であることが確認された．

5.2.5.2　サブセット間のデータ数統一実験

表 3.2 に示したように，本研究で構築したデータセットでは「論文の概要」のデータ数が他のドメインと比較して多い．これまでの実験でサブセットによる違い (ドメインまたは LLM による違い) を考察したが，サブセットのデータ数の違いが結果に影響を与えた可能性もある．そのため，全てのサブセットにおけるデータ数が同じという条件下での実験を行う．全 10 サブセットについて，学習データとして 1,400 件，テストデータとして 180 件をランダムに抽出し，すべてのサブセットで同じデータ量となるよう調整した．

このデータ数を統一したサブセットを用いて，以下の 2 つの条件で学習および評価を行った．

1. **全サブセット混合学習**: 全サブセットの学習データをすべて使用して学習する．
2. **単一ドメイン学習**: 特定の 1 ドメインのデータのみを用いて学習する．本項では,「論文の概要」,「Yahoo!知恵袋」を使用した場合を比較する．

表 5.12: サブセット間のデータ数統一実験 - 全サブセット混合学習による評価

ドメイン	LLM	Accuracy	Precision	Recall	F1	AUROC
論文の概要	Swallow-8B	0.8667	0.7895	**1.0000**	0.8824	0.9947
	Llama3	0.8667	0.7895	**1.0000**	0.8824	0.9970
	ChatGPT	0.8667	0.7895	**1.0000**	0.8824	0.9998
	Gemini	0.8667	0.7895	**1.0000**	0.8824	0.9993
Yahoo!知恵袋	Swallow-8B	0.9056	0.8411	**1.0000**	0.9137	**1.0000**
	Llama3	0.9278	0.8738	**1.0000**	0.9326	0.9996
Wikinews	Swallow-8B	**0.9500**	0.9091	**1.0000**	**0.9524**	**1.0000**
	Llama3	0.9389	0.8911	**1.0000**	0.9424	0.9998
Wikipedia	Swallow-8B	0.9444	0.9000	**1.0000**	0.9474	0.9998
	Llama3	**0.9500**	**0.9175**	0.9889	0.9519	0.9979
平均		0.9083	0.8491	0.9989	0.9170	0.9988

表 5.13: サブセット間のデータ数統一実験 – 「論文の概要」ドメインのみ学習による評価

ドメイン	LLM	Accuracy	Precision	Recall	F1	AUROC
論文の概要	Swallow-8B	0.8667	0.7895	**1.0000**	0.8824	0.9917
	Llama3	0.8667	0.7895	**1.0000**	0.8824	0.9937
	ChatGPT	0.8667	0.7895	**1.0000**	0.8824	**1.0000**
	Gemini	0.8667	0.7895	**1.0000**	0.8824	**1.0000**
Yahoo!知恵袋	Swallow-8B	0.7722	**0.9804**	0.5556	0.7092	0.9349
	Llama3	**0.8833**	0.9726	0.7889	0.8712	0.9746
Wikinews	Swallow-8B	**0.8833**	0.8165	0.9889	**0.8945**	0.9694
	Llama3	0.8278	0.7658	0.9444	0.8458	0.9485
Wikipedia	Swallow-8B	0.8500	0.7692	**1.0000**	0.8696	0.9941
	Llama3	0.7611	0.7156	0.8667	0.7839	0.9153
平均		0.8444	0.8178	0.9144	0.8504	0.9722

表 5.14: サブセット間のデータ数統一実験 – 「Yahoo!知恵袋」ドメインのみ学習による評価

ドメイン	LLM	Accuracy	Precision	Recall	F1	AUROC
論文の概要	Swallow-8B	0.5667	0.5357	**1.0000**	0.6977	0.7652
	Llama3	0.5611	0.5329	0.9889	0.6926	0.6336
	ChatGPT	0.5667	0.5357	**1.0000**	0.6977	0.7832
	Gemini	0.5667	0.5357	**1.0000**	0.6977	0.7225
Yahoo!知恵袋	Swallow-8B	**0.9667**	**0.9375**	**1.0000**	**0.9677**	0.9991
	Llama3	**0.9667**	**0.9375**	**1.0000**	**0.9677**	**0.9993**
Wikinews	Swallow-8B	0.5000	0.5000	**1.0000**	0.6667	0.7794
	Llama3	0.5278	0.5143	**1.0000**	0.6792	0.7891
Wikipedia	Swallow-8B	0.5000	0.5000	**1.0000**	0.6667	0.9305
	Llama3	0.5000	0.5000	**1.0000**	0.6667	0.7417
平均		0.6222	0.6029	0.9989	0.7400	0.8144

表 5.12 に全サブセット混合学習の結果，表 5.13，表 5.14 に論文の概要ドメインのみ，Yahoo!知恵袋ドメインのみの結果を示す．なお，Wikinews および Wikipedia のみを用いて学習した場合の結果については付録 C に掲載する．

まず，表 5.12 と，表 5.2「条件:All」の比較から，今回のデータ量を統一した場合で大きく性能が低下している．今回のデータ量統一実験では,「Yahoo!知恵袋」などの小規模ドメインは元データに近いデータ量が維持されているのに対し，元データの大半を占めていた「論文の概要」ドメインは 1/10 以下まで大幅にデータ量が減少している．このことが性能に影響していると考えられる.「論文の概要」ドメインのスコアは,「Yahoo!知恵袋」や「Wikinews」と比較して相対的に低くなっており,「論文の概要」ドメインの予測難易度が相対的に高いことが確認された．

次に，表 5.13（論文の概要のみ）と表 5.14（Yahoo!知恵袋のみ）の結果から，学習と同じドメインに対しては高い性能を示すが，他ドメインに対しては非常に低い性能を示している.「条件:All」や「条件:Domain」の実験では，特に Yahoo!知恵袋ドメインの汎化性能の低さがデータ不足によるものかドメイン固有の難しさに起因するのかの判別が困難であったが，データ数を揃えた今回の実験において，論文の概要ドメインではある程度の汎化性能が維持されている．このことから，Yahoo!知恵袋でのみ学習した場合に他ドメインを識別できない現象は，データ量に依存するものではなく，ドメインの性質に起因するということが確認できた．

5.2.5.3 パラフレーズ攻撃に対する頑健性

LLM 生成テキストの自動検知を回避するため，テキストの一部を書き換える（パラフレーズする）などの処理が行われることがある．特に英語圏では，Undetectable

AI[1] のように，AI 生成文を人間らしい文へ自動書き換えすることで GPTZero[8] などの検出器を回避することを目的としたサービスが登場している．そのため，LLM 生成テキスト識別モデルの検出回避策に対する頑健性の検証が重要である．そこで本研究では，代表的な識別回避手法である，文章中の一部単語を意味を変えずに書き換える (パラフレーズ) 攻撃に対する頑健性を検証する．

軽微な書き換え (摂動) を加えるモデルには，日本語 T5 モデルである retrieva-jp/t5-large-long[36] を使用した．具体的な摂動生成の手順は以下の通りである．まず，T5 のトークナイザを用いてテキストをトークン ID 列に変換した後，全トークン数の 15%に相当する箇所をランダムに選択し，スパン長 2 のマスク（<extra_id_x>）に置き換える．その後，T5 モデルによってマスク箇所を生成（穴埋め）することで，元の文の意味を維持したまま単語が変更された摂動テキストを作成した．なお，生成時のサンプリング温度（Temperature）は 1.0 に設定した．

評価に使用したテストデータの件数およびその内訳は，5.2.1 項の「条件:All」，すなわち全データセットを用いてモデルを学習する条件と同一である．摂動を加えたテストセットに対する評価結果を表 5.15 に示す．

表 5.15: パラフレーズ攻撃に対する頑健性評価

ドメイン	LLM	Accuracy	Precision	Recall	F1	AUROC
論文の概要	Swallow-8B	0.9486	0.9190	0.9839	0.9504	0.9883
	Llama3	0.9320	0.9164	0.9507	0.9333	0.9802
	ChatGPT	0.9556	0.9200	0.9979	0.9573	0.9937
	Gemini	0.9545	0.9199	0.9957	0.9563	0.9926
Yahoo!知恵袋	Swallow-8B	0.9669	0.9667	0.9667	0.9667	0.9960
	Llama3	**0.9899**	**0.9898**	0.9898	0.9898	0.9997
Wikinews	Swallow-8B	0.9701	0.9436	**1.0000**	0.9710	**1.0000**
	Llama3	0.9536	0.9188	0.9945	0.9551	0.9975
Wikipedia	Swallow-8B	0.9602	0.9298	0.9956	0.9615	0.9946
	Llama3	0.9556	0.9556	0.9556	0.9556	0.9923
マイクロ平均		0.9587	0.9380	0.9830	0.9597	0.9935
マクロ平均		0.9510	0.9242	0.9825	0.9901	

表 5.15 の結果を，摂動なしの条件（表 5.2）と比較すると，全体として性能低下は小さいことが確認できる．

全体のマイクロ平均 Accuracy は 0.9594 から 0.9510 へ，マクロ平均 Accuracy は 0.9688 から 0.9587 へと微減するに留まっている．ドメイン別に見ても，Llama3 の論文の概要においては Accuracy が 0.9543 から 0.9320 へと約 0.02 ポイント低下しているものの，依然として 0.93 以上の高い水準を維持している．また，Yahoo!知恵袋など，摂動後もほぼ同等の識別性能を維持しているケースも見られた．これ

らの結果から，今回学習した RoBERTa 分類器は摂動を加えるパラフレーズ攻撃に対して一定の頑健性を有しているといえる．

ただし，本実験における摂動は特定のマスク率（15%）および 1 つのモデルによるパラフレーズ攻撃に限定されており，マスク率や摂動モデルを変えた場合や，異なる攻撃手法に対する頑健性は検証していない．モデルの頑健性をより詳細に評価するためには，さらなる検証が必要となる．

5.2.6 結果のまとめと考察

本節では，5.1 節で述べた実験設定に基づき，RoBERTa による教師あり学習，ゼロショット識別実験，および人間による識別実験を行い，その性能を検証した．また，教師あり分類器については，学習データの構成（ドメイン・LLM）やデータ量を変化させた場合の汎化性能，さらには入力テキストへの摂動に対する耐性など，多角的な検証を行った．本項では，これらの実験結果から得られた知見を総括し，本研究で構築した分類器の特性について考察する．

まず，本研究で構築した教師あり分類器と，比較対象であるゼロショット手法および人間による評価の結果について総括する．人間による評価では，正解率が平均で 62%程度に留まるなど，人間が LLM 生成テキストを見抜くことは難しいことが明らかとなった．このように人間には識別が困難なタスクであるにもかかわらず，RoBERTa モデルをファインチューニングした分類器が,「条件:All」において Accuracy 0.96 以上という極めて高い精度を達成した．一方，ゼロショット手法に関しては対数尤度が最も性能が良かったものの，AUROC で平均 0.91 程度と RoBERTa 分類器よりも劣った．以上のことから，今回の実験設定において，教師あり学習に基づくアプローチが強力であるということが確認できた．

次に，分類器の汎化性能について，分類器は学習データに含まれない「未知のドメインと LLM」に対しては識別性能が低下しやすいという問題が確認できた．しかし，この問題は特定のドメインに依存せず，性質の異なる複数のドメインのテキストから学習データを構成することで劇的に改善され，高い汎化性能が獲得できることが確認された．また，その際に必要なデータ量についても，データ量が増えるほど性能の向上が確認できたものの，本研究で構築したデータセットの全データの 25%〜50%程度で性能が飽和する傾向が見られた．また，T5 によるパラフレーズ（言い換え）攻撃に対しても性能低下は小さく，構築したモデルは単純な攻撃には一定の頑健性を有しているといえる．

以上の結果より，本研究で提案した RoBERTa 分類器は，人間には分からない特徴を捉え，高精度で LLM 生成テキストを識別できることが確認できた．しかし，「なぜ人間には識別ができないのに，モデルはこれほど高精度に識別できるのか」，モデルが何を根拠に識別を行っているのかが明らかになっていない．実社会での運用を想定した場合，人間生成テキストを誤って LLM 生成と判定する（誤検知）リスクや，その逆のリスクは避けられない．そのため，単に精度が高いだけでな

く，その判定根拠を提示できる説明性が重要であり，これはLLM生成テキスト識別タスクにおける重要な課題の一つである．そこで次節では，モデルが具体的にテキストのどのような特徴に着目しているのかを解明し，LLM生成テキストを人間以上の正確さで識別できる理由を明らかにすることを目指す．

5.3 LLM生成テキスト識別モデルの内部の分析

本節では，RoBERTaをベースモデルとしたLLM生成テキスト識別モデルの内部を分析し，同モデルが高い識別性能を持つ理由や，識別モデルがテキスト内のどのような特徴に着目しているかを考察する．

5.3.1 AttentionならびにIGによる可視化

RoBERTa分類器が入力テキストのどの部分に着目して識別を行ったかを明らかにするため，Attentionの可視化とIntegrated Gradients (IG) [47]の計算による可視化を行う．

本分析では,「生成テキストの後処理（句読点の統一など）」を適用する前の段階のデータセットの「条件:All」によって構築された分類器を解析対象とした．次に，同一の分類器を用いて，各入力についてモデルが最終的に出力した予測クラス（LLM／人間）のスコアを対象とし，そのスコアに対するAttention重みおよびIntegrated Gradients（IG）を算出した．これにより，モデルが当該クラスを選択した際に根拠として寄与したトークンを分析する．以下に，それぞれの手法の定義，分析における概念，スコアの意味，および本実験における具体的な算出方法について述べる．

5.3.1.1 AttentionならびにIGの定義と計算方法

Attention可視化の定義　Attention Mechanism（注意機構）とは，Transformerモデルにおいて入力系列内の各トークンが互いにどの程度関連しているか（重み）を計算する仕組みである．BERTやRoBERTaを用いた分類タスクでは，入力の先頭に付与される特殊トークン `[CLS]` が入力全体の情報を集約し，最終的にそのベクトル表現が分類層へ入力されることでラベルが予測される．モデルは多層のAttention層を経て，入力テキスト全体の文脈情報をこの `[CLS]` トークンのベクトル表現へと集約する．

Attentionの可視化は，この情報の集約過程において，この `[CLS]` トークンが入力テキスト中の「どのトークンに強く注意を向けたか」可視化する方法である．最終層における `[CLS]` から各トークンへのAttention重みを抽出し，モデルが分類判断を行う直前に参照されたトークンの傾向を調べる．

本実験における Attention の算出方法　本実験では，RoBERTa モデルの最終層（第 12 層）における Attention Weight を使用した．最終層の [CLS] トークンから全入力トークンへの Attention 重みを取得し，トークンごとに平均化した．さらに，算出された重みに対して 0 から 1 の範囲に収まるよう Min-Max 正規化を適用し，これを最終的な Attention スコアとした．

IG の定義と概念　Integrated Gradients (IG) は，入力の変化がモデル出力をどう変動させたかという関係を勾配積分により寄与度として定量化する手法である．IG は，情報を持たないベースラインから実際の入力ベクトルに至るまで，予測確率が入力に対してどれだけ変化するか（勾配）を経路に沿って積分し，その累積量として評価する．この累積プロセスは,「予測確率の総変化量」を各トークンの動いた方向へ分解して割り振ることに相当する．これにより，各トークンが予測スコアをどれだけ押し上げ／押し下げたかを寄与度として集計できる．本研究のような 2 クラス分類においては，この集計値が正負の符号を持つ寄与度として算出されるため,「LLM 生成」と「人間生成」のどちらの予測根拠として機能したかを定量的に評価することが可能である．

第 i 番目の特徴量 x_i に対する IG の値は以下の式で定義される．

$$IG_i(x) = (x_i - x_i') \times \int_{\alpha=0}^{1} \frac{\partial F(x' + \alpha(x - x'))}{\partial x_i} d\alpha \tag{5.10}$$

ここで，F はモデルの出力関数，$\alpha \in [0, 1]$ は経路の補間係数を表す．右辺の積分項は，ベースラインから入力までの変化に伴う勾配を計算する．第一項の $(x_i - x_i')$ はベースラインと入力の差分である．この「変化の幅」を「総合的な勾配」に乗じることで，特定の入力次元が予測値の変動に果たした総量が導かれ，これが各トークンの寄与度となる．なお，実際の計算においては積分を解析的に求めることは困難であるため，経路を有限個のステップに離散化し，その総和によって近似的に IG を算出する．

本実験における IG の算出方法　本実験では，解釈可能性ライブラリ Captum[17] を用いて IG スコアを算出した．ベースラインにはゼロ埋め込みベクトル（Zero Embedding Vector）を設定し，ベースラインから入力までの積分経路を 100 ステップ（$n\_steps = 100$）で計算した．RoBERTa の入力は各トークンに対して 768 次元の埋め込みベクトルを持つため，算出された各次元の寄与度を合算（Sum）し，トークンごとのスカラー値を IG スコアとした．

5.3.1.2　Attention ならびに IG による可視化の例

図 5.2 および図 5.3 に，論文の概要ドメインの LLM 生成テキストに対して Attention と IG のスコアを算出し可視化した例を示す．Attention 可視化（図 5.2）で

は色の濃さがモデルの注目度を表し，IG の可視化（図 5.3）では，赤色が LLM 生成への寄与，青色が人間生成への寄与を表している．

図 5.2 では「本研究では」や「と考えられた」といった箇所に注目していることが確認でき，図 5.3 では「研究では」,「組み合わせる」,「可能である」といった箇所の「LLM 生成テキスト」クラスへの寄与スコアが高いことが確認できる．しかし，これらを全テキストに対して個別に目視確認することは現実的ではなく，統計的な分析が必要である．そこで本実験では，ドメイン単位でトークンのスコアを平均し，そのスコアが上位となるトークンを確認することで，モデルがどういったトークンを判断の根拠にしていることが多いのかを確認する．

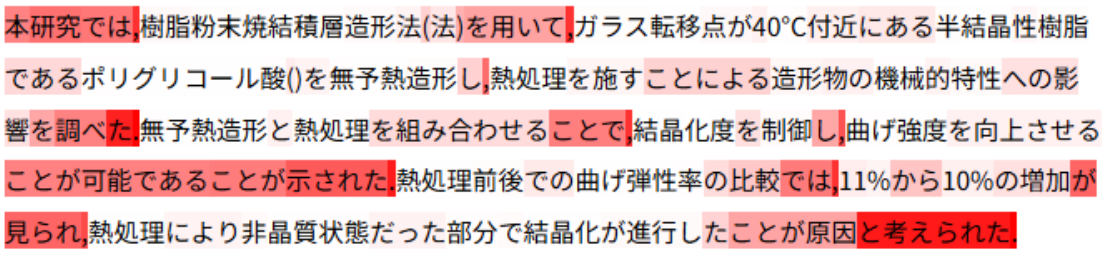


図 5.2: Attention の可視化例

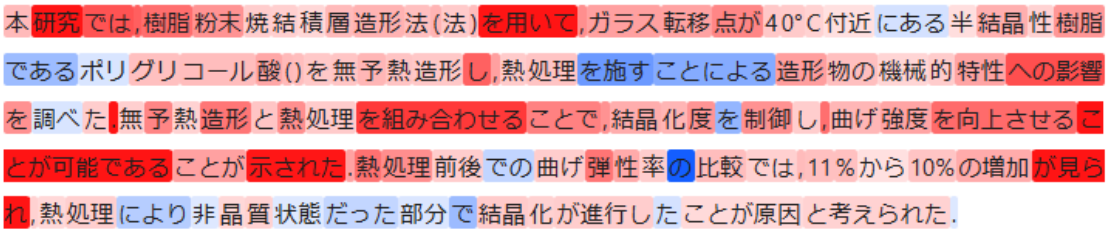


図 5.3: IG の可視化例

5.3.1.3　上位スコアのトークンの表示と分析

前項で述べた統計的な検証を行うにあたり，本実験では全テストデータのスコアに対し，以下の前処理および正規化を適用した上で集計を行った．

前処理と正規化　まず，Attention 可視化では本文中のどのトークンに注意が向いたかという分析目的と対応しないため，[CLS] を含む特殊トークンを集計対象から除外した．次に，算出された Attention スコアおよび IG スコアに対し，サンプルごとの L1 正規化を適用した．生のスコアは入力長や予測確信度によりサンプル間でスケールが異なるため，各サンプル内でスコアの絶対値の総和が 1 となるよう正規化し，テキスト内での相対的な重要度を示すようにした．

$$\hat{S}_w = \frac{S_w}{\sum_{w' \in T} |S_{w'}| + \epsilon} \tag{5.11}$$

ここで，S_w はトークン w のスコア，T はそのテキスト内の全トークン集合である．

集計方法　正規化されたスコアをドメインごとにトークン単位で平均化し，これを最終的なスコアとした．なお，ごく少数の出現回数のサンプルが上位を占めることを避けるため，各ドメイン内で出現回数が 10 回未満のトークンはランキングから除外した．また，IG は寄与の方向（正負）によって，Attention は予測されたラベル（LLM/Human）によって集計対象を分けた．

なお，ここでの分析結果は前述したとおり,「生成テキストの後処理（句読点の統一など)」を適用する前の段階のデータを用いて算出されている．そのため，以下の結果表には，後処理によって除去・統一される対象である「、」や「。」，あるいは空白などが上位に含まれている．これらは，前処理適用前のデータにおいてモデルが強く着目していた特徴を示している．

以上の条件に基づき，各ドメインにおいて算出された Attention および IG の上位 15 トークンを表 5.16 から表 5.19 に示す．

表 5.16: 各ドメインの LLM 予測 Attention スコア上位トークン（Top 15）

Rank	論文の概要	Yahoo!知恵袋	Wikinews	Wikipedia
1	のことである	月	、	ことが可能である
2	となっているが	、	、2007	で使用される
3	述べている	私は	、2006	用いられる
4	した後	、「	を決定した	を有する
5	を確立した	などで	カル	のことである
6	を搭載し	なので	、2005	調
7	旅行者	年齢	を提供する	、
8	たものである	い	発達	に使用される
9	を提供した	問い合わせ	進んで	年
10	にした	ファン	レース	表示
11	,	!	に達し	の分野で
12	ことなど	「	海上	とされる
13	を取り入れ	ありません	y	することが可能である
14	が広く	魅力	騎手	を使用して
15	ことにより	います	番組	つである

表 5.17: 各ドメインの人間予測 Attention スコア上位トークン（Top 15）

Rank	論文の概要	Yahoo!知恵袋	Wikinews	Wikipedia
1	、	.....	T	、「
2	,100	なん	、	、
3	江戸	!!!	削除	日本の
4	が増える	になり	避難	カテゴリ
5	で行われた	、	フィリピン	情報処理
6	,	かな	上陸	の分野で
7	を行っていた	感じ	ページ	パルス
8	と考えられ	ん	島の	株式会社
9	まとめ	な	事件で	ウェブ
10	があるため	んで	北西	開発された
11	しようとする	「	進	電気
12	復元	んだ	ハリケーン	関連
13	とは言えない	ご	可能性がある	などを
14	が出る	まで	、「	試験
15	を引き起こす	さ	見込み	事

表 5.18: 各ドメインの LLM 寄与 IG スコア上位トークン（Top 15）

Rank	論文の概要	Yahoo!知恵袋	Wikinews	Wikipedia
1	論文	.....	を決定した	ことが可能である
2	_日本では	私は	事故が発生	ことを目的としている
3	近年	名	ことが明らかになった	近年
4	今後の	用の	フジテレビ	することが可能である
5	従来の	メーカー	、2006	法は
6	コール	参考	大相撲	設計された
7	将来的に	があれば	特に	が開発した
8	研究成果	あ	今後の	ことを目的として
9	ことを目的としている	多	関東地方	問題は
10	を目指し	!!!	に貢献	を活用し
11	ただし	彼は	号は	設立され
12	今後	ですか	することを発表	を有する
13	に焦点を当て	個人の	は同	特に
14	ことが明らかになった	特に	は不明	で使用される
15	に貢献	レンタル	ことを決めた	に貢献

表 5.19: 各ドメインの Human 寄与 IG スコア上位トークン（Top 15）

Rank	論文の概要	Yahoo!知恵袋	Wikinews	Wikipedia
1	そのために	_その	。	それらの
2	、	…	以後	。
3	そこから	感じ	に対して	ながら
4	の目的は	じゃない	『	そのため
5	することによって	ではない	によれば	それら
6	した上で	^	」(	もしくは
7	その中で	!!	_また	その後
8	それらの	くらい	ページ	げん
9	た上で	_また	であった	「
10	に関して	よ	伝えている	その
11	パートナーシップ	いま	_今	ly
12	トータル	ボール	において	現実
13	に対して	。	貴	それ
14	と言われている	しない	その	に対して
15	にて	その	その後	の

表 5.16 から表 5.19 の結果について考察を行う．LLM 生成テキストの Attention 分析の結果（表 5.16）では「、」や「のことである」といった数値・文末表現や記号が上位に確認される傾向にあった．一方で，IG 分析の結果（表 5.18) では，論文の概要における「論文」「今後の」，Wikinews や Wikipedia における「ことが明らかになった」「ことが可能である」のように，各ドメインにおける書き出しや文章の締めの定型表現として用いられる語句が多い傾向が見られ，Attention 分析の結果とは異なる傾向にあった．また，それぞれのドメインの上位トークンの比較では「に貢献」など一部共通のトークンは確認できるものの，ほとんどは各ドメインで共通していない．そして，これらの Attention スコアと IG スコアの上位トークンは，人間生成テキストでも多用されている表現や単語に感じ，なぜこれらトークンが LLM 寄与スコアで上位なのか分からなかった．

一方，人間生成テキストに対する IG 上位（表 5.19）では，論文の概要の「そのために」や Wikipedia の「それらの」といった接続詞が上位に含まれる傾向があり，Attention（表 5.17）においても「、」や「,」といった句読点が上位に位置するなど，LLM 生成テキストとは異なる特徴が確認された．

次に，上位トークンについてより深く議論するため，論文の概要ドメインの順位とトークンに追加して，各スコア，出現回数，および出現比率（Ratio）を併記した結果を表 5.20 から表 5.23 に示す．

表 5.20: 論文の概要ドメイン（LLM 生成予測）の Attention スコア分析

順位	トークン	スコア	出現回数		比率 (H/LLM)
			LLM	Human	
1	のことである	0.0167	10	0	∞ 倍
2	となっているが	0.0165	13	15	0.87 倍
3	述べている	0.0163	18	5	3.60 倍
4	した後	0.0161	15	36	0.42 倍
5	を確立した	0.0158	16	0	∞ 倍
6	を搭載し	0.0150	10	0	∞ 倍
7	旅行者	0.0148	12	18	0.67 倍
8	たものである	0.0145	145	0	∞ 倍
9	を提供した	0.0141	14	0	∞ 倍
10	にした	0.0141	11	5	2.20 倍
11	,	0.0140	63,226	39,981	1.58 倍
12	ことなど	0.0137	14	10	1.40 倍
13	を取り入れ	0.0133	14	6	2.33 倍
14	が広く	0.0131	21	26	0.81 倍
15	ことにより	0.0128	11	95	0.12 倍

表 5.21: 論文の概要ドメイン（人間生成予測）の Attention スコア分析

順位	トークン	スコア	出現回数		比率 (H/LLM)
			LLM	Human	
1	、	0.0416	0	15	∞
2	,100	0.0360	19	10	0.53
3	江戸	0.0318	0	10	∞
4	ウォーク	0.0267	16	5	0.31
5	が増える	0.0255	8	10	1.25
6	▁日本では	0.0246	7	4	0.57
7	ことを示している	0.0245	18	5	0.28
8	科学技術	0.0234	16	5	0.31
9	を報告した	0.0230	5	5	1.00
10	問題が	0.0226	37	1	0.03
11	,	0.0222	63,226	39,981	0.63
12	で行われた	0.0216	7	14	2.00
13	が強く	0.0215	13	5	0.38
14	の良い	0.0211	18	5	0.28
15	ことが原因	0.0208	6	5	0.83

表 5.22: 論文の概要ドメイン（LLM 寄与）の IG スコア分析

順位	トークン	スコア	出現回数		比率 (L/H)
			LLM	Human	
1	論文	0.0573	1,275	230	5.54
2	_日本では	0.0417	7	4	1.75
3	近年	0.0337	317	84	3.77
4	今後の	0.0334	973	72	13.51
5	従来の	0.0317	920	145	6.34
6	将来的に	0.0289	99	24	4.12
7	研究成果	0.0282	27	0	∞
8	コール	0.0281	6	4	1.50
9	ことを目的としている	0.0281	346	28	12.36
10	を目指し	0.0277	304	28	10.86
11	ただし	0.0274	211	28	7.54
12	今後	0.0270	693	112	6.19
13	に焦点を当て	0.0269	192	24	8.00
14	ことが明らかになった	0.0267	508	93	5.46
15	に貢献	0.0264	414	24	17.25

表 5.23: 論文の概要ドメイン（Human 寄与）の IG スコア分析

順位	トークン	スコア	出現回数		比率 (H/L)
			LLM	Human	
1	そのために	0.0395	3	20	6.67
2	、	0.0345	0	12	∞
3	そこから	0.0333	4	12	3.00
4	の目的は	0.0267	12	128	10.67
5	することによって	0.0249	9	60	6.67
6	その中で	0.0231	3	20	6.67
7	た上で	0.0228	3	20	6.67
8	した上で	0.0228	7	28	4.00
9	それらの	0.0223	9	96	10.67
10	に関して	0.0206	7	72	10.29
11	に対して	0.0191	202	436	2.16
12	トータル	0.0181	4	8	2.00
13	にて	0.0181	0	68	∞
14	いわゆる	0.0180	2	12	6.00
15	加えて	0.0178	7	48	6.86

表5.20から表5.23の追加の分析結果から，分類器の判断傾向について考察する．

まずLLM寄与を示すIGスコア（表5.22）では,「論文」（1,275回/Ratio 5.54）や「今後の」（973回/Ratio 13.51）など，人間生成テキストより出現回数が多く，比率も高いトークンとなっている．この結果は，LLMで多用される表現が，分類器におけるLLM判定の根拠として強く機能していることを示す．つまり，IGスコアの上位トークンは単体で見ると人間が書いた文章でも一般的に用いられる表現に思えるが，実際はそれぞれの生成テキストにおける典型的な表現であることが分かった．Human寄与を示すIGスコア（表5.23）でも出現頻度に差がある傾向が確認でき，1位の「そのために」では6.67倍の出現差があるなど，全てのトークンで人間生成テキストのほうが出現頻度が高い．しかし，LLM寄与と比較すると，全体的な出現回数が少なく，また寄与スコアも小さいという違いも確認できた．

一方で，Attentionスコアの上位（表5.20）については，IGとは異なる挙動が見られた．IGでは見られなかった特徴として，出現比率が1.0を下回る（すなわち人間生成テキストの方が多い）トークンが上位に含まれている点が挙げられる．加えて，出現回数が10回程度の比較的低頻度なトークンが多くランクインしており，IGで見られたような出現頻度との明確な相関は確認できない．この要因については不明であるが，少なくともIGとは異なる箇所に着目していたことが確認できる．

これらの結果から，IGスコアの上位トークンに関しては出現頻度や出現比率とある程度の相関が確認できた．つまり，分類器はそれぞれのテキストの典型的な表現を識別の根拠として学習し，それらが識別の根拠として機能して，高精度の識別へと結びついていた可能性が考えられる．

5.3.1.4 N-gramによるIGの分析

前節のトークン単位の分析では，特定の語彙が識別に寄与していることは確認できたが，それらが文脈の中でどのように用いられているかが確認できなかった．そこで，モデルがどういった単語の並びに注目しているのか分析するため，連続するトークン（N-gram）を単位としたフレーズ分析を行った．

本分析では，文脈の最小単位として意味を成しやすい3-gram（連続する3トークン）を抽出対象とした．各フレーズのスコア算出およびランキングの手順は以下の通りである．

- **フレーズスコアの算出**: あるサンプル内に出現した3-gramフレーズ $P = (w_1, w_2, w_3)$ に対し，構成する各トークンのIGスコアの総和を，そのフレーズのスコア S_P とした．

$$S_P = \sum_{i=1}^{3} S_{w_i} \tag{5.12}$$

- **集計とランキング**: ドメイン内の全データにおける各フレーズの出現回数 C_P と，スコアの累計を算出した．最終的な評価指標として，先ほどの単語単位の分析と同様に，出現 1 回あたりの平均スコア（$AverageScore = \sum S_P / C_P$）を採用した．

- **フィルタリング**: 統計的な信頼性を確保するため，ドメイン内での出現回数が 10 回未満のフレーズは除外した．

表 5.24 および表 5.25 に，論文の概要ドメインにおける IG スコア上位の 3-gram フレーズを示す．

表 5.24: 論文の概要ドメインにおける LLM 寄与 IG スコア上位フレーズ（3-gram）

Rank	Phrase (3-gram)	Count	AvgScore
1	_本 論文 は	567	0.1068
2	論文 は ,	683	0.1018
3	を開発した . 従来の	12	0.1001
4	を実施した . 結果	28	0.0954
5	ことが明らかになった . 今後の	19	0.0889
6	を示唆する . 今後の	12	0.0883
7	砥 石 の	13	0.0880
8	_この 論文 は	68	0.0878
9	. 今後の 研究	93	0.0848
10	_本 論文 では	363	0.0817
11	ことを目的としている . 近年	12	0.0814
12	ことを目的とした . 従来の	14	0.0812
13	を目指した . 従来の	13	0.0810
14	_ 著者 らは	12	0.0801
15	. 結果 ,	451	0.0799

表 5.25: 論文の概要ドメインにおける Human 寄与 IG スコア上位フレーズ（3-gram）

Rank	Phrase (3-gram)	Count	AvgScore
1	▁本 報 では	12	0.0799
2	報告されている . 本	13	0.0432
3	々 行 車	12	0.0413
4	オク タン 価	12	0.0400
5	本 システムの 有効性	10	0.0391
6	, 基地 局	16	0.0378
7	国土交通省 では ,	12	0.0370
8	リガ ネ ムシ	12	0.0356
9	ワイ ヤ 放電	12	0.0349
10	本 研究 ではこの	16	0.0338
11	を用いた 漏れ 検査	12	0.0337
12	生物 多様性 保全	11	0.0333
13	火 花 点火	14	0.0332
14	電波 暗 室	12	0.0330
15	走 査 型	22	0.0324

LLM 生成テキストにおける定型フレーズの検知　LLM 寄与（表 5.24）の上位フレーズを確認すると,「本論文は（▁本 論文 は)」（Rank1, 567 回）や「論文は,（論文 は ,)」（Rank2, 683 回）といった書き出しの定型句が高いスコアとなっている．それ以外でも,「ことが明らかになった．今後の」（Rank 5）のように，文末から次の文の文頭へ跨るフレーズなど，文章のつなぎとして使用される表現が高いスコアとなっていることが確認できる．

これらの結果から，モデルは「論文」などの単語単体だけでなく，周辺トークンを含む語の並びにも注目していることが分かる．実際に，トークン単位の上位スコア（例：「論文」0.0573）に対し，それを含む 3-gram の平均スコア（例：「▁本 論文 は」0.1068）が高く，3-gram が 3 トークンの総和である点を踏まえても中心語のみで寄与が決まっているとは言い難い．そのため前後のトークンも含めたフレーズも識別根拠として機能している可能性が高いと解釈できる．

人間生成テキストとの対比　一方，人間生成テキスト（表 5.25）は，LLM と比較して上位フレーズが出現回数の少ない．LLM の一部上位フレーズが数百回規模出現していたのに対し，Human の上位フレーズは 10 回から 20 回程度にとどまっている．この理由については 4.1 節でも説明したように人間生成テキストは文章表現が多様であることが理由であると考えられる．人間は様々な表現で文章を記述するため，LLM のように偏りがなく，その結果上位フレーズでも出現回数が少ない

可能性が考えられる．

フレーズの内容に着目すると,「オクタン価」「ワイヤ放電」「生物多様性保全」といった専門的な技術用語を含むフレーズが多くランクインしている点が特徴的である．また，書き出しや文頭表現においても，LLM が一様に「本論文は」を使用するのに対し，人間は「本報では」「本システムの実効性」「本研究ではこの」のような表現が見られた．これらのことから，モデルが人間生成テキストを識別する際には，定型的なテンプレートへの依存度は低く，LLM では使用されない多様な表現や専門用語を，人間による記述の特徴として捉えている可能性があると考えられる．

5.3.2 モデルの層別評価による識別根拠の分析

前項までの IG およびフレーズ分析により，モデルが特定の単語や定型的な表現（書き出しや接続パターン）を識別の手掛かりとしている可能性が示唆された．しかし，Attention や IG による分析では，モデルがこれらの特徴を「表層的な記号」として捉えているのか，あるいは「文法的な構造」や「意味」として学習しているのか不明である．そこで本項では，モデル内部の表現がどの段階で識別に十分な情報を獲得しているかを確認するため，RoBERTa モデルの Transformer 層の深さに着目した分析を行う．この分析は「生成テキストの後処理（句読点の統一など)」を行った後の「条件:All」の分類器を対象とする．

層ごとに保持される言語情報　BERT 系列の内部表現が層に応じて異なる性質を持つことは，いわゆる BERTology の文脈で広く検討されてきた．Rogers らは，BERT の内部解析に関する研究を体系的に整理した調査論文であり，層ごとに保持されやすい情報（語順・構文など）に関する知見を総括している [39]．また，Jawahar らは，BERT の各層から得られる表現を用いた層別比較を行い，層の深さに応じて抽出可能な言語情報が変化することを実証的に示している [13]．これらの先行研究による知見は以下のようにまとめられる．

- **下位層（Lower Layers；Layer 1-4 付近）**：単語レベルの局所的な情報や，線形的な語順など，表層的な情報を保持する傾向がある．
- **中間層（Middle Layers；Layer 5-8 付近）**：依存関係や句構造など，構文に関わる情報が強く現れる傾向がある．
- **上位層（Upper Layers；Layer 9-12 付近）**：文脈に依存した高次の情報が増え，意味的な情報が含まれる傾向がある．

すなわち，層が浅い段階では単語や語順といった表層的情報が強く，層が進むにつれて構文的・意味的な情報がより顕在化する傾向がある．本項の層別読み出

し評価は，このような層ごとの表現差を背景として，LLM 生成テキストの識別に有効な情報がどの層から読み出し可能になるかを確認するための実験となる．

層別読み出し評価の実験方法　本実験では,「条件:All」の学習済み RoBERTa 分類器のパラメータを固定したまま，層ごとの隠れ表現を分類ヘッドへの入力として LLM 生成テキストか否かを分類する．入力文を学習時と同一の設定でトークナイズし，各層 ℓ $(0 \leq \ell \leq 12)$ の隠れ表現 $\mathbf{H}^{(\ell)}$ を取得する（$\ell = 0$ は埋め込み層に相当）．各層の隠れ表現から [CLS] トークンに対応するベクトル $\mathbf{h}_{\mathrm{CLS}}^{(\ell)}$ を抽出し，これを共通の（学習済み）分類ヘッドへ入力して予測を行う．

なお，本検証における評価データには,「条件:All」，すなわち全てのサブセットのテストセットを結合したデータを用いた．この操作を層ごとに繰り返し，性能（Accuracy，Precision，Recall，F1，AUROC）の推移を比較することで，どの層から LLM 生成テキストの識別に有効な抽象表現が獲得できているかを確認する．

表 5.26: 層ごとの分類性能の検証結果

Layer	Accuracy	Precision	Recall	F1	AUROC
0	0.6996	0.7289	0.6350	0.6787	0.7200
1	0.7107	0.7486	0.6337	0.6864	0.7403
2	0.7228	0.7612	0.6487	0.7004	0.7817
3	0.6838	0.8537	0.4431	0.5834	0.7701
4	0.7724	0.7977	0.7294	0.7621	0.8679
5	0.8327	0.7878	0.9104	0.8446	0.9251
6	0.9010	0.8346	**1.0000**	0.9098	0.9771
7	0.9474	0.9051	0.9996	0.9500	0.9681
8	0.7630	0.6785	0.9989	0.8081	0.9673
9	0.9604	0.9274	0.9989	0.9618	0.9784
10	**0.9606**	**0.9277**	0.9989	**0.9620**	0.9789
11	**0.9606**	**0.9277**	0.9989	**0.9620**	0.9842
12	0.9592	0.9254	0.9989	0.9608	**0.9947**

結果と考察　実験結果を表 5.26 に示す．結果を見ると，下位層（Layer 0-3）では Accuracy が 0.7 前後と比較的低く推移しているが，中間層にあたる Layer 4 から Layer 7 にかけて急激な性能向上が確認された．特に Layer 6 および Layer 7 の時点で，Accuracy で 0.9474 という高い値に達している．BERTology の知見に基づけば，この中間層（Layer 5-9 付近）は構文的な情報の処理を最も得意とする領域である．一方で，意味理解を担うとされる上位層（Layer 10-12）においては，Accuracy は高止まりしており，中間層からの改善はほとんど見られない．

本結果から確認できるのは，分類ヘッドを固定した層別読み出しという条件の下で，最終層で学習された判定規則が中間層（Layer 4–7 付近）の [CLS] 表現に対

しても成立し，既に高い識別性能が得られるという点である．そのため，本分類器が高い識別性能を得るために必要な情報は，上位層に到達する前に，下位層と中間層の [CLS] 表現の段階で既に十分に獲得されていることが確認された．この結果は，表層的特徴や構文的手掛かりが強く反映される層の表現だけで LLM 生成テキストを正確に識別できる可能性を示唆する．

5.3.3 線形分類器を用いた特徴の分析

前項の層別評価の結果，表層的あるいは構文的な情報を主に反映する中間層までの情報で，既に十分な識別性能を獲得していることが確認された．この結果は，LLM 生成テキストの識別において，高度な文脈理解や意味的な整合性の判断は必ずしも必要なく，単語の出現分布や局所的な構文パターンといった，より低次な特徴の組み合わせでタスクが解決可能であることを示唆している．

そこで本項では，この仮説を検証するため，テキストから「表層」「構文」「意味」の 3 つの階層の特徴量を個別に抽出し，それぞれを用いて単純な線形分類器（ロジスティック回帰）を構築し，これらを比較する実験を行う．RoBERTa のような巨大なモデルではなく，解釈容易な特徴量と線形モデルによる近似でどの程度の精度が達成できるかを確認することで，LLM 生成テキストがどのような特徴で識別できるのか検証を行う．

5.3.3.1 各階層の特徴量の定義

検証にあたり，3 種類の特徴量を以下のように定義した．

表層的特徴　「どのようなトークンが使用されているか」という表層的な情報である．もし RoBERTa が主に「特定の単語の有無」や「定型的な言い回し」に反応しているならば，これらの特徴のみを用いたモデルでも高い精度が出るはずである．本実験では，以下の 2 つを特徴量とした．

- **Unigram (BoW)**：文脈を考慮しない，トークン単位の出現頻度．LLM 特有の生成単語の偏りを捉える．
- **N-gram (2-gram, 3-gram)**：連続するトークン．「_本 論文 は」のようなフレーズを捉える．

これらの特徴量は入力テキストを RoBERTa のトークナイザによってサブワード分割し抽出する．「条件:All」の訓練データにおける頻度上位 5,000 件の特徴量をベクトルの次元とし，単純な出現回数をベクトルの値として，特徴ベクトルを作成した．

構文的特徴　文法的な情報だけでどれぐらいの精度で識別が可能かを検証する．本実験では，係り受け解析器（GiNZA）を用いて各テキストの依存構造木を構築し，根（Root）から各トークンへ至る依存関係パスと，その経路上の途中ノードから始まる部分パスを特徴とした．パスの構成要素により，以下の 2 種類の特徴量を定義した．

- **依存関係ラベルパス**：パス上のノードを「依存関係ラベル」のみで表現したもの（例：`ROOT` → `obl` → `case`，および Root から始まらない部分パスの例：`advmod` → `xcomp` → `xcomp`）．
- **依存関係ラベル+POS 付与パス**：パス上のノードに品詞（POS）情報を付与し，親 `POS` → `dep` → ... → 子 `POS` の形式で表現したもの「`NOUN` → `acl` → `AUX`」）はこの特徴量の例である．「名詞（NOUN）が節修飾（acl）によって修飾され，その修飾した語が補助動詞（AUX）」といった構造パターンを表す．

意味的特徴　表層や構文の情報ではなく，テキスト全体が持つ意味の内容によって人間生成テキストと LLM 生成テキストが識別可能か検証する．本実験では，4.5 節と同様に，日本語文埋め込みモデルとして SBERT`sonoisa/sentence-bert-base-ja-mean-tokens-v2` を採用し，テキストを 768 次元のベクトルに変換したものを特徴量として定義した．

5.3.3.2　実験設定

実験は全てのサブセットを用いる「条件:All」の設定で行った．分類器にはすべてロジスティック回帰で学習した．ハイパーパラメータである正則化強度の逆数 C は，係数の大きさ（正則化の強さ）を制御するパラメータである．本実験では，全ての特徴量において候補 $C \in \{0.01, 0.1, 1.0, 10.0\}$ から検証データ（Valid）を用いて探索を行い，最も正解率（Accuracy）の高かった値を採用した．

表 5.27: ロジスティック回帰分類器の結果 (特徴量はトークン Unigram)

ドメイン	LLM	Accuracy	Precision	Recall	F1	AUROC
論文の概要	Swallow-8B	0.9310	0.9333	0.9283	0.9308	0.9812
	Llama3	0.9197	0.9317	0.9058	0.9186	0.9720
	ChatGPT	0.9604	0.9370	**0.9872**	0.9614	**0.9957**
	Gemini	0.9551	0.9363	0.9764	0.9560	0.9925
Yahoo!知恵袋	Swallow-8B	0.9006	0.9500	0.8444	0.8941	0.9552
	Llama3	0.9497	**0.9785**	0.9192	0.9479	0.9709
Wikinews	Swallow-8B	**0.9620**	0.9521	0.9728	**0.9624**	0.9913
	Llama3	0.9180	0.9419	0.8901	0.9153	0.9763
Wikipedia	Swallow-8B	0.9314	0.9046	0.9646	0.9336	0.9759
	Llama3	0.8778	0.9337	0.8133	0.8694	0.9560
マイクロ平均		0.9373	0.9352	0.9397	0.9374	0.9825
マクロ平均		0.9306	0.9399	0.9202	0.9290	0.9767

表 5.28: ロジスティック回帰分類器の結果 (特徴量はトークン 2-gram)

ドメイン	LLM	Accuracy	Precision	Recall	F1	AUROC
論文の概要	Swallow-8B	0.9449	0.9434	0.9465	0.9449	0.9864
	Llama3	0.9369	0.9425	0.9304	0.9364	0.9826
	ChatGPT	0.9674	0.9459	**0.9914**	0.9681	**0.9976**
	Gemini	0.9625	0.9454	0.9818	0.9632	0.9951
Yahoo!知恵袋	Swallow-8B	0.9061	0.9506	0.8556	0.9006	0.9775
	Llama3	**0.9799**	**1.0000**	0.9596	**0.9794**	0.9954
Wikinews	Swallow-8B	0.9647	0.9572	0.9728	0.9650	0.9970
	Llama3	0.9071	0.9512	0.8571	0.9017	0.9861
Wikipedia	Swallow-8B	0.9624	0.9447	0.9823	0.9631	0.9939
	Llama3	0.9044	0.9789	0.8267	0.8964	0.9788
マイクロ平均		0.9494	0.9477	0.9513	0.9495	0.9894
マクロ平均		0.9436	0.9560	0.9304	0.9419	0.9890

表 5.29: ロジスティック回帰分類器の結果 (特徴量はトークン 3-gram)

ドメイン	LLM	Accuracy	Precision	Recall	F1	AUROC
論文の概要	Swallow-8B	0.9224	0.9330	0.9101	0.9214	0.9782
	Llama3	0.8909	0.9284	0.8469	0.8858	0.9626
	ChatGPT	**0.9625**	0.9381	**0.9904**	**0.9635**	**0.9968**
	Gemini	0.9572	0.9375	0.9797	0.9581	0.9934
Yahoo!知恵袋	Swallow-8B	0.9171	0.9412	0.8889	0.9143	0.9694
	Llama3	0.9045	0.9255	0.8788	0.9016	0.9742
Wikinews	Swallow-8B	0.9511	**0.9770**	0.9239	0.9497	0.9927
	Llama3	0.7514	0.9417	0.5330	0.6807	0.9466
Wikipedia	Swallow-8B	0.8827	0.9303	0.8274	0.8759	0.9540
	Llama3	0.7444	0.9508	0.5156	0.6686	0.8704
マイクロ平均		0.9147	0.9365	0.8895	0.9124	0.9749
マクロ平均		0.8884	0.9403	0.8295	0.8720	0.9638

表 5.30: ロジスティック回帰分類器の結果 (特徴量は依存関係ラベルパス)

ドメイン	LLM	Accuracy	Precision	Recall	F1	AUROC
論文の概要	Llama3	0.8609	0.8616	0.8597	0.8607	0.9279
	Gemini	0.8609	0.8616	0.8597	0.8607	0.9272
	ChatGPT	0.8780	0.8662	0.8940	0.8799	0.9406
	Swallow-8B	0.8497	0.8584	0.8373	0.8477	0.9169
Yahoo!知恵袋	Llama3	0.8894	0.9053	0.8687	0.8866	0.9383
	Swallow-8B	0.8564	**0.9706**	0.7333	0.8354	0.8971
Wikinews	Llama3	0.8907	0.9438	0.8297	0.8830	0.9650
	Swallow-8B	**0.9293**	0.9389	**0.9185**	**0.9286**	**0.9730**
Wikipedia	Llama3	0.7800	0.8088	0.7333	0.7692	0.8733
	Swallow-8B	0.8385	0.8228	0.8628	0.8423	0.9096
マイクロ平均		0.8615	0.8659	0.8551	0.8605	0.9283
マクロ平均		0.8634	0.8838	0.8397	0.8594	0.9269

表 5.31: ロジスティック回帰分類器の結果 (特徴量は依存関係ラベル+POS 付与パス)

ドメイン	LLM	Accuracy	Precision	Recall	F1	AUROC
論文の概要	Llama3	0.8796	0.8720	0.8897	0.8808	0.9412
	Gemini	0.8812	0.8724	0.8929	0.8825	0.9457
	ChatGPT	0.8994	0.8768	**0.9293**	0.9023	0.9620
	Swallow-8B	0.8732	0.8704	0.8769	0.8736	0.9371
Yahoo!知恵袋	Llama3	0.9095	**0.9263**	0.8889	0.9072	0.9358
	Swallow-8B	0.8453	0.8974	0.7778	0.8333	0.8890
Wikinews	Llama3	0.8907	0.9128	0.8626	0.8870	0.9641
	Swallow-8B	**0.9239**	0.9239	0.9239	**0.9239**	**0.9750**
Wikipedia	Llama3	0.8044	0.8442	0.7467	0.7925	0.9128
	Swallow-8B	0.8761	0.8632	0.8938	0.8783	0.9442
マイクロ平均		0.8810	0.8761	0.8872	0.8816	0.9458
マクロ平均		0.8783	0.8859	0.8682	0.8761	0.9407

表 5.32: ロジスティック回帰分類器の結果 (特徴量は SBERT)

ドメイン	LLM	Accuracy	Precision	Recall	F1	AUROC
論文の概要	Swallow-8B	0.8261	0.8155	0.8426	0.8289	0.9077
	Llama3	0.7175	0.7664	0.6253	0.6887	0.7976
	ChatGPT	**0.8727**	0.8308	**0.9358**	**0.8802**	**0.9520**
	Gemini	0.8609	0.8272	0.9122	0.8676	0.9428
Yahoo!知恵袋	Swallow-8B	0.7459	**0.9231**	0.5333	0.6761	0.8957
	Llama3	0.7035	0.8448	0.4949	0.6242	0.8221
Wikinews	Swallow-8B	0.7663	0.7475	0.8043	0.7749	0.8349
	Llama3	0.7568	0.7460	0.7747	0.7601	0.8418
Wikipedia	Swallow-8B	0.8009	0.8301	0.7566	0.7917	0.8919
	Llama3	0.7378	0.8092	0.6222	0.7035	0.8323
マイクロ平均		0.8063	0.8098	0.8001	0.8049	0.8869
マクロ平均		0.7788	0.8141	0.7302	0.7596	0.8719

5.3.3.3 結果と考察

表 5.27 から表 5.32 に，各特徴量を用いたロジスティック回帰による分類結果を示す.

表層的特徴による識別性能　表 5.27 の結果が示すように，文脈情報を考慮しない単語の出現頻度（Unigram）のみを用いた場合であっても，全テストセットのマイクロ平均 Accuracy は 0.9373，AUROC は 0.9825 という非常に高い数値を示した．このことから単純な単語の情報だけでも，今回対象とした LLM 生成テキストの多くを識別可能であることが確認された．このことは，本データセットにおいて，人間生成テキストと LLM 生成テキストの間に単語の選択や単語の頻度分布の明確な差が存在し，それ自体が強い識別手掛かりとして機能することを示している．

さらに，隣接するトークンの並びを考慮した 2-gram（表 5.28）においては，Unigram と比較して平均 Accuracy が約 1.2 ポイント向上し，0.9494 となった．これは，5.3.1.4 のフレーズ分析で観測されたような「定型的な言い回し」や「局所的な接続パターン」が，識別において有効な特徴量として寄与していることを示唆している．一方で，3-gram（表 5.29）では，一部の条件で精度の低下が見られた．これは，N-gram の次数を上げたことで特徴空間が疎（スパース）になり，性能に悪影響を及ぼした可能性があると考えられる．

構文的・意味的特徴との比較　構文的特徴（表 5.31）に着目すると，POS 付与パスを用いた場合のマイクロ平均 Accuracy は 0.8810 となり，表層的特徴には及ばないものの，一定の識別精度を示した．これは，LLM 生成テキストにおける品詞や係り受け構造のパターンにある程度の特徴が存在する可能性を示している．

一方で，意味的特徴である SBERT を用いた場合（表 5.32）のマイクロ平均 Accuracy は 0.7788 にとどまり，今回比較した 3 つの特徴の中では最も低い結果となった．この結果は，SBERT のベクトル表現と線形分類器の組み合わせという本実験の設定下においては，表層や構文と比較して，意味内容による人間生成テキストと LLM 生成テキストの区別が相対的に困難であることを示している．

5.3.4　分析結果のまとめと考察

本節では，Attention や IG の可視化，層別評価，および線形分類器による分析を通じ，RoBERTa 分類器の判断根拠について検証を行った．一連の分析から得られた主要な知見は，モデルが「高度な意味理解」ではなく，単語の出現頻度や定型的なフレーズといった「表層的な統計的特徴」を強力な手掛かりとしている可能性が高いことである．

この知見が，5.2 節で確認された「人間による識別精度の低さ（約 62%）」と「機械学習モデルの極めて高い識別精度（95%以上）」という精度の差異を説明する重要な手掛かりであると考えられる．すなわち，人間が識別を行う際，現在の最新の LLM は流暢な日本語を生成できるため，人間にとっての手掛かり（違和感）は極めて少ない．特に，本研究の「論文の概要」ドメインのように，論文本文の一部をプロンプトとして与えて生成させた場合，意味的な破綻や文章的な不自然さはさらに抑制されるため，人間による識別は難しい．一方で，機械学習モデルは大

量のテキストデータに潜在する「単語の偏り」や「定型パターン」といった，人間には知覚し得ない統計的な特徴を学習することができる．すなわち，RoBERTa 分類器は，主に訓練データに存在する単語やパターンの偏りを識別のための根拠として学習することで，人間よりも高精度な識別を実現したと考えられる．この「表層的なパターンへの依存」という仮説は，未知のドメインや LLM に対して識別精度が大きく低下するという実験結果（5.2.2 項）とも整合する．すなわち，学習データに含まれないドメインやモデルでは，分類器が学習した特定の語彙や定型パターンが存在しない，あるいは異なる分布となるため，識別性能が損なわれたと考えられる．

一方で，5.2.3 のパラフレーズ攻撃に対する結果と比較すると，単純な単語置換によって主要な手掛かりが容易に失われるわけではないことも示している．この現象は，分類器が元々高い確信度（大きなマージン）をもって判定しているサンプルが多いことに起因する可能性がある．実際に softmax 後の LLM クラス確率を確認すると，0 または 1 に非常に近い値をとるサンプルが多く，高い確信度で判定されていることが分かる（付録 D）．パラフレーズによって分類の信頼度は低くなる (正解クラスの確率が小さくなる) が，最終的な判定が変わるほどには変化せす，結果として性能低下が小さく抑えられたと考えられる．

第6章　結論

6.1　本論文のまとめ

本研究は日本語のLLM生成テキスト識別に取り組んだ．既存研究の多くが英語を対象としており，日本語を対象とした研究やデータセットが不足しているという現状を踏まえ，大規模な日本語LLM生成テキストデータセットの構築，分類器の開発，評価を行った．

まず，日本語LLM生成テキスト識別タスクのためのデータセットを構築した．論文の概要を対象に，4種類の最新LLM（GPT-4o，Gemini 1.5，Llama 3，Swallow-8B）を用い，人間生成テキストとLLM生成テキストの組を収録した大規模なデータセットを構築した．また，Yahoo!知恵袋，Wikinews，Wikipediaといった異なるドメインを対象としたデータセットも構築し，多様な性質を持つLLM生成テキストの評価環境を整備した．これにより，文章構造や使用される単語の傾向が異なるテキストでの検証や，既存の研究の課題である異なるLLMから生成されたテキストの識別を検証できる基盤を構築した．

次に，構築したデータセットに対する言語的特徴分析を行った．人間生成テキストはLLM生成テキストと比較してType/Token比が一貫して高く，より多様な語彙を使用していることが統計的に確認された．また，パープレキシティ（PPL）の分析では，LLM生成テキストが人間生成テキストよりもPPLが低い傾向が見られ，既存のPPLを手がかりとするゼロショット手法が日本語に対しても適用可能である可能性が高いことを示した．しかし，論文の概要のドメインでは人間生成テキストとLLM生成テキストとでPPLの分布の重なりが大きく，ドメインによってはゼロショット手法による完全な識別は難しい可能性が高いことも示した．

さらに，構築したデータセットを用いてLLM生成テキストを識別する分類器を教師あり学習した．事前学習済み日本語RoBERTaを基盤とする2値分類器を学習し，複数テストセットに対する包括的な性能評価を行った．実験の結果，全ドメインのデータを学習させた分類器は，Accuracy 0.96以上，AUROC 0.99以上という極めて高い識別性能を達成した．人間による識別実験において，平均正解率が約62%とランダムな予測（50%）をわずかに上回る程度に留まった事実と比較すると，RoBERTa分類器は人間には知覚できないテキストの特徴を捉え，LLM生成テキストを高精度に識別可能であることが実証された．また，未知のLLMやドメインに対して分類精度が低下する問題も確認できた．この問題に対する対策として，複数のLLMやドメインを組み合わせたデータセットから分類器を学習する

ことで，未知のテキストに対しても，汎化性能がある程度保たれることも示した．

また，本研究では分類器が高い性能を発揮できた要因を探るため，Integrated Gradients (IG) や Attention の可視化，モデルの層別評価，特徴量を限定した線形分類器による分析を試みた．その結果，単純な単語の出現頻度（BoW）や短いフレーズ（N-gram）を用いた線形分類器であっても RoBERTa に近い精度が達成できることや，RoBERTa の層別評価において構文情報が処理される中間層の段階で識別性能が飽和する傾向が確認された．これらの分析結果は，今回構築した分類器が，必ずしも高度な文脈理解や意味的な判断によって識別を行っているわけではなく,「使用される単語の偏り」や「定型的なパターン」といった表層的特徴を主要な手掛かりとしている可能性が高いことを示唆するものであった．

6.2 今後の課題

本研究の限界と今後の課題について述べる．

- **文章の一部に LLM 生成テキストが使用されている場合の識別**
  本研究の識別対象となるテキストは，テキスト全体を人間もしくは LLM が生成したものであることを前提としている．しかし，実際の運用環境では，LLM 生成テキストに対して人間が加筆・編集を行う，文章全体でなく一部分のみ LLM 生成文が使用される，といった状況も想定される．本研究の手法は，文書全体を入力として LLM 生成か人間生成か判定をするため，LLM 生成テキストと人間生成テキストが混在しているテキストに対して対応することが困難である．今後は，混在・編集を含むテキストを対象とし，文単位や段落単位で LLM 生成テキストか否かを識別手法の開発が必要である．

- **データセットの拡張**
  本研究では，識別の汎用性を高めるべく 4 種の LLM および 4 種のドメインを対象にデータセットを構築したが，この組み合わせだけで日本語の生成テキスト全体を網羅できているとは言えない．本研究のデータセットに含まれない全く異なる文体や専門性を持つドメイン，あるいは今後登場する新たな LLM に対して，本研究で構築した LLM 生成テキスト識別モデルが現状のような十分に高い性能を維持するとは保証されていない．そのため，教師あり分類器の汎化性能向上やより多様な検証を行うためにも，ドメインの追加や最新の LLM を含むデータを収集することが必要であると考えている．

- **教師あり手法以外の既存手法との比較**
  本研究では，事前学習済み RoBERTa モデルをファインチューニングする手法の有効性を確認した．しかし，2 章で述べたように，LLM 生成テキスト識別の手法としては，教師あり機械学習以外にも，PPL 等の統計的指標に基づくゼロショット手法や，生成過程に特定の情報を埋め込むウォーターマーキ

ング手法など，複数の有力なアプローチが存在する．本研究では，基本的なゼロショット手法の検証を実施したものの，最新のゼロショット検出法との比較など他の有力な手法の検証はしていない．

これらの先行研究の手法は，識別精度や汎化性能，運用コストにおいて異なる特性を持つことが報告されている．そのため，既存の有力な手法を本研究で構築した日本語データセットに適用して比較実験を行い，日本語を対象としたLLM生成テキスト識別タスクにおける各手法の有効性と限界，および本手法との優劣を明らかにすることが今後の課題である．

参考文献

[1] Undetectable AI. Undetectable ai. *Official website*, 2026. accessed 2026-01-23.

[2] AI@Meta. Llama 3 model card. *Official website*, 2024.

[3] cardiffnlp. twitter-xlm-roberta-base-sentiment. *Hugging Face model card*, 2025. accessed 2025-12-22.

[4] Digital Education Council. *What Students Want: Key Results from DEC Global AI Student Survey 2024*. Digital Education Council, 2024.

[5] Universal Dependencies. Universal dependencies: Dependency relations. *Official website*, 2025. accessed 2025-12-22.

[6] Inc. ELYZA. Llama-3-elyza-jp-8b. *Hugging Face Model Card*, 2026. accessed 2026-01-05.

[7] Sebastian Gehrmann, Hendrik Strobelt, and Alexander M. Rush. Gltr: Statistical detection and visualization of generated text. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pp. 111–116, 2019.

[8] GPTZero. Gptzero. *Official website*, 2026. accessed 2026-01-23.

[9] Andrew Gray. Chatgpt "contamination": estimating the prevalence of llms in the scholarly literature. *arXiv preprint arXiv:2403.16887*, 2024.

[10] Biyang Guo, Xin Zhang, Ziyuan Wang, Minqi Jiang, Jinran Nie, Yuxuan Ding, Jianwei Yue, and Yupeng Wu. How close is chatgpt to human experts? comparison corpus, evaluation, and detection. *arXiv preprint arXiv:2301.07597*, 2023.

[11] Hans W. A. Hanley and Zakir Durumeric. Machine-made media: Monitoring the mobilization of machine-generated articles on misinformation and mainstream news websites. *arXiv preprint arXiv:2305.09820*, 2023.

[12] Masato Hirakawa, Shintaro Horie, Tomoaki Nakamura, Daisuke Oba, Sam Passaglia, and Akira Sasaki. elyza/llama-3-elyza-jp-8b. *Hugging Face Model Card*, 2024.

[13] Ganesh Jawahar, Benoît Sagot, and Djame Seddah. What does bert learn about the structure of language? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL 2019)*, pp. 3651–3657, 2019.

[14] 警察庁. 令和 6 年におけるサイバー空間をめぐる脅威の情勢等について. 警察庁, 2025.

[15] Logan Kilpatrick and Shrestha Basu Mallick. Gemini 1.5 pro および 1.5 flash の一般公開，1.5 flash のチューニング サポートとレート制限の緩和，その他の api アップデート. *Google Developers Blog*, 2024.

[16] John Kirchenbauer, Jonas Geiping, Yuxin Wen, Jonathan Katz, Ian Miers, and Tom Goldstein. A watermark for large language models. In *Proceedings of the 40th International Conference on Machine Learning (ICML)*, pp. 17061–17084, 2023.

[17] Narine Kokhlikyan, Vivek Miglani, Miguel Martin, Edward Wang, Bilal Alsallakh, Jonathan Reynolds, Alexander Melnikov, Natalia Kliushkina, Carlos Araya, Siqi Yan, and Orion Reblitz-Richardson. Captum: A unified and generic model interpretability library for pytorch. *arXiv preprint arXiv:2009.07896*, 2020.

[18] 国立研究開発法人科学技術振興機構（JST）. J-STAGE WebAPI ご利用マニュアル（Ver.1.2）. J-STAGE, 2024.

[19] 国立研究開発法人科学技術振興機構（JST）. J-stage（科学技術情報発信・流通総合システム）. *Official website*, 2026. accessed 2026-01-21.

[20] Kalpesh Krishna, Yixiao Song, Marzena Karpinska, John Wieting, and Mohit Iyyer. Paraphrasing evades detectors of ai-generated text, but retrieval is an effective defense. In *Advances in Neural Information Processing Systems (NeurIPS) 36*, pp. 27469–27500, 2023.

[21] Taku Kudo, Kaoru Yamamoto, and Yuji Matsumoto. Applying conditional random fields to japanese morphological analysis. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 230–237, 2004.

[22] Tharindu Kumarage, Amrita Bhattacharjee, Djordje Pijetlovic, Tyler R. Giallanza, Sheeraz Rahim, Joshua Garland, and Huan Liu. J-guard: Journalism guided adversarially robust detection of ai-generated news. In *Proceedings of the 13th International Joint Conference on Natural Language Processing and the 3rd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 484–497, 2023.

[23] Megagon Labs. Ginza: 日本語自然言語処理オープンソースライブラリ. *Official website*, 2025. accessed 2025-12-22.

[24] Yafu Li, Qintong Li, Leyang Cui, Wei Bi, Zhilin Wang, Longyue Wang, Linyi Yang, Shuming Shi, and Yue Zhang. Mage: Machine-generated text detection in the wild. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 36–53, 2024.

[25] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

[26] lxyuan. distilbert-base-multilingual-cased-sentiments-student. *Hugging Face model card*, 2025. accessed 2025-12-22.

[27] 前川喜久雄, 山崎誠. 書き言葉コーパス—設計と構築—（講座日本語コーパス 2）. 朝倉書店, 2014.

[28] 丸井渚生, 曹洋, 中村篤祥. 日本語における大規模言語モデルの生成文検出. DEIM Forum 2024 論文集, 2024.

[29] Lorenz Mindner, Tim Schlippe, and Kristina Schaaff. Classification of human- and ai-generated texts: Investigating features for chatgpt. *arXiv preprint arXiv:2308.05341*, 2023.

[30] Eric Mitchell, Yoonho Lee, Alexander Khazatsky, Christopher D. Manning, and Chelsea Finn. Detectgpt: Zero-shot machine-generated text detection using probability curvature. In *Proceedings of the 40th International Conference on Machine Learning (ICML)*, pp. 24950–24962, 2023.

[31] 文部科学省. 初等中等教育段階における生成 AI の利活用に関するガイドライン (Ver. 2.0). 文部科学省, 2024.

[32] Nature. Tools such as chatgpt threaten transparent science; here are our ground rules for their use. *Nature*, Vol. 613, No. 7945, p. 612, 2023.

[33] OpenAI. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024.

[34] OpenAI. Gpt-4o model (snapshots: gpt-4o-2024-08-06). *OpenAI API Documentation*, 2026. accessed 2026-01-05.

[35] Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 3982–3992, 2019.

[36] retrieva jp. t5-large-long. *Hugging Face model card*, 2026. accessed 2026-01-15.

[37] rinna. japanese-roberta-base. *Hugging Face model card*, 2026. accessed 2026-01-13.

[38] Juan Diego Rodriguez, Todd Hay, David Gros, Zain Shamsi, and Ravi Srinivasan. Cross-domain detection of gpt-2-generated technical text. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 1213–1233, 2022.

[39] Anna Rogers, Olga Kovaleva, and Anna Rumshisky. A primer in bertology: What we know about how bert works. *Transactions of the Association for Computational Linguistics*, Vol. 8, pp. 842–866, 2020.

[40] Kei Sawada, Tianyu Zhao, Makoto Shing, Kentaro Mitsui, Akio Kaga, Yukiya Hono, Toshiaki Wakatsuki, and Koh Mitsuda. Release of pre-trained models for the japanese language. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pp. 13898–13905, 2024.

[41] Aditya Shah, Prateek Ranka, Urmi Dedhia, Shruti Prasad, Siddhi Muni, and Kiran Bhowmick. Detecting and unmasking ai-generated texts through explainable artificial intelligence using stylistic features. *International Journal of Advanced Computer Science and Applications (IJACSA)*, Vol. 14, No. 10, pp. 1043–1048, 2023.

[42] SlashNext. *The State of Phishing 2023*. SlashNext Threat Labs, 2023.

[43] Irene Solaiman, Miles Brundage, Jack Clark, Amanda Askell, Ariel Herbert-Voss, Jeff Wu, Alec Radford, Gretchen Krueger, Jong Wook Kim, Sarah Kreps, Miles McCain, Alex Newhouse, Jason Blazakis, Kris McGuffie, and

Jasmine Wang. Release strategies and the social impacts of language models. *arXiv preprint arXiv:1908.09203*, 2019.

[44] sonoisa. sentence-bert-base-ja-mean-tokens-v2. *Hugging Face model card*, 2025. accessed 2025-12-21.

[45] Jinyan Su, Terry Yue Zhuo, Di Wang, and Preslav Nakov. Detectllm: Leveraging log rank information for zero-shot detection of machine-generated text. In *Findings of the Association for Computational Linguistics: EMNLP*, pp. 12395–12412, 2023.

[46] Zhenpeng Su, Xing Wu, Wei Zhou, Guangyuan Ma, and Songlin Hu. Hc3 plus: A semantic-invariant human chatgpt comparison corpus. *arXiv preprint arXiv:2309.02731*, 2023.

[47] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *Proceedings of the 34th International Conference on Machine Learning (ICML 2017), Proceedings of Machine Learning Research*, Vol. 70, pp. 3319–3328, 2017.

[48] Gemini Team. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv preprint arXiv:2403.05530*, 2024.

[49] H. Holden Thorp. Chatgpt is fun, but not an author. *Science*, Vol. 379, No. 6630, p. 313, 2023.

[50] TokyoTech-LLM. Llama-3.1-swallow-8b-instruct-v0.3. *Hugging Face Model Card*, 2026. accessed 2026-01-05.

[51] Wikinews. ウィキニュースについて. *Official website*, 2025. accessed 2025-12-19.

[52] Wikipedia. ウィキペディアについて. *Official website*, 2025. accessed 2025-12-19.

[53] Xianjun Yang, Wei Cheng, Yue Wu, Linda Ruth Petzold, William Yang Wang, and Haifeng Chen. Dna-gpt: Divergent n-gram analysis for training-free detection of gpt-generated text. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2024.

[54] Peipeng Yu, Jiahan Chen, Xuan Feng, and Zhihua Xia. Cheat: A large-scale dataset for detecting chatgpt-written abstracts. *arXiv preprint arXiv:2304.12008*, 2023.

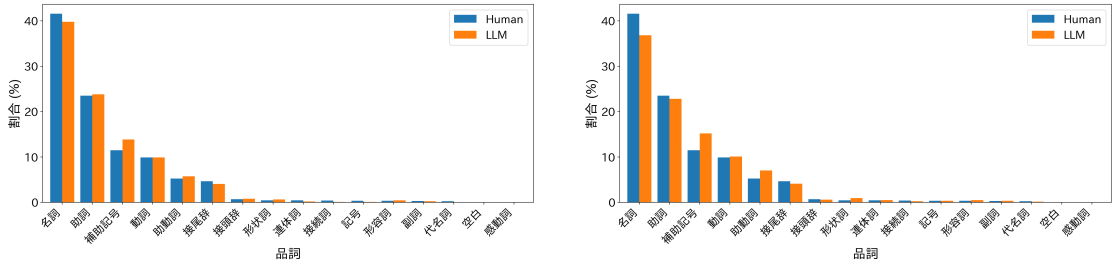
[55] Rowan Zellers, Ari Holtzman, Hannah Rashkin, Yonatan Bisk, Ali Farhadi, Franziska Roesner, and Yejin Choi. Defending against neural fake news. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 9051–9062, 2019.

付 録A　データセットの分析結果の補足

本章では4 章で述べたデータセットの分析結果を補足する

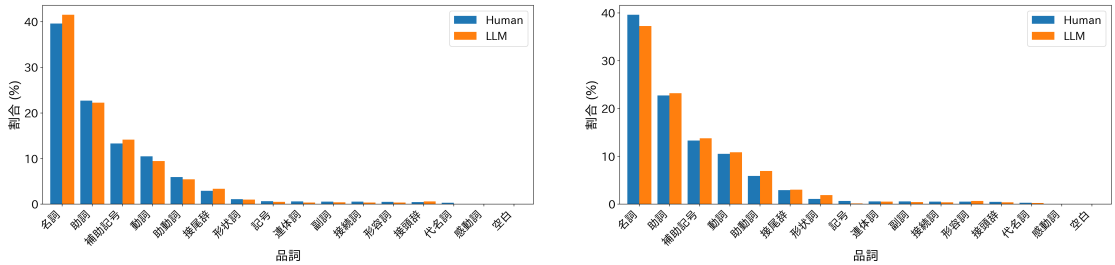
A.1　品詞の分析

4.2 節で述べた品詞の分析結果を補足する．図 A.1 および図 A.2 は，それぞれ Wikinews と Wikipedia における品詞の分布である．



(a) Llama3　　(b) Swallow-8B

図 A.1: Wikinews における品詞分布

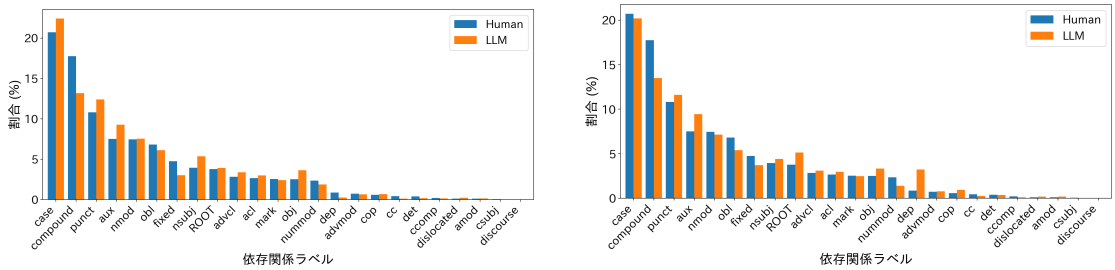


(a) Llama3　　(b) Swallow-8B

図 A.2: Wikipedia における品詞分布

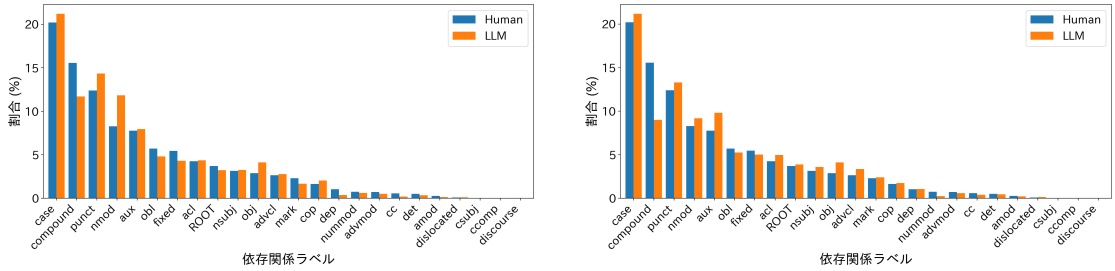
A.2　依存構造解析の結果

4.3 節で述べた品詞の依存構造解析結果を補足する．図 A.3 および図 A.4 は，それぞれ Wikinews と Wikipedia における依存関係ラベルの分布である．



(a) Llama3　　(b) Swallow-8B

図 A.3: Wikinews における依存関係ラベル分布

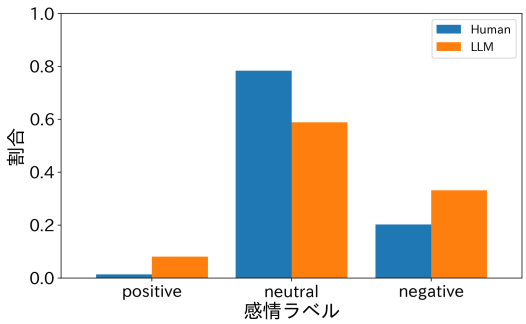


(a) Llama3　　(b) Swallow-8B

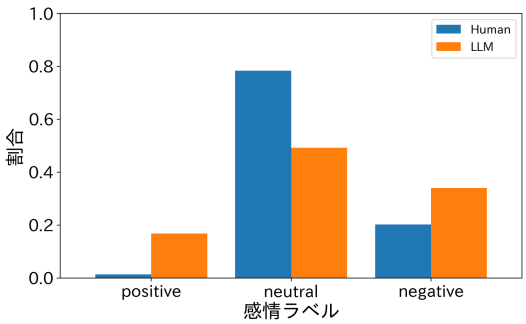
図 A.4: Wikipedia における依存関係ラベル分布

A.3　感情分析

4.4 節で述べた感情分析の結果を補足する．図 A.5 および図 A.6 は，それぞれ Wikinews と Wikipedia の XLM-R を用いた際の感情ラベルの分布である．また，図 A.7 および図 A.8 は，それぞれ Wikinews と Wikipedia の DistilBERT を用いた際の感情ラベルの分布である．

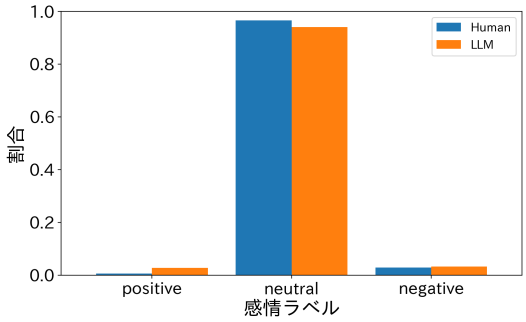


(a) Llama3

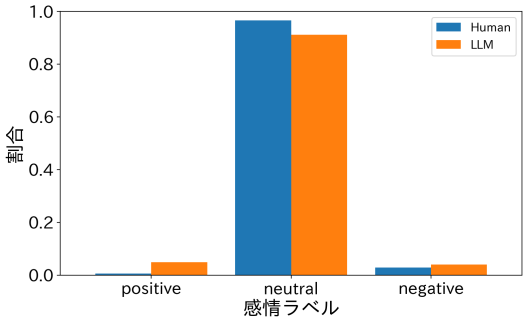


(b) Swallow-8B

図 A.5: Wikinews における感情極性の分布（XLM-R）

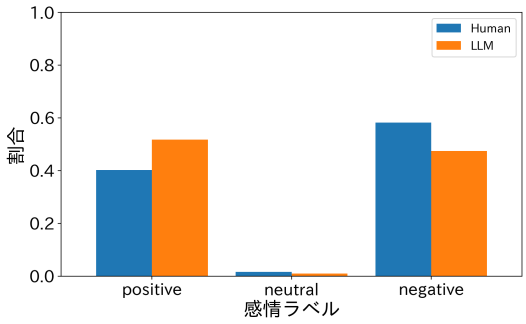


(a) Llama3

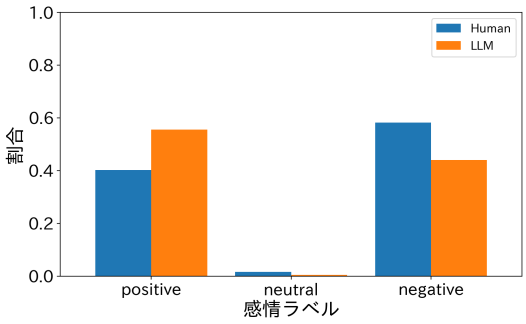


(b) Swallow-8B

図 A.6: Wikipedia における感情極性の分布（XLM-R）

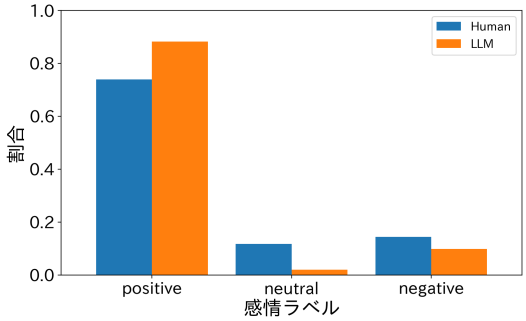


(a) Llama3

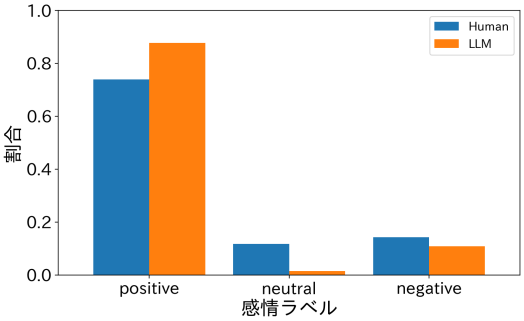


(b) Swallow-8B

図 A.7: Wikinews における感情極性の分布（DistilBERT）



(a) Llama3

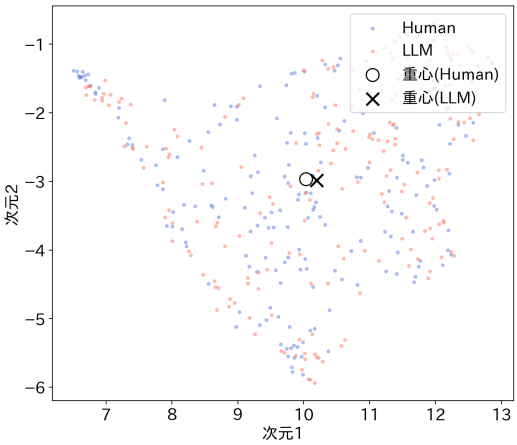


(b) Swallow-8B

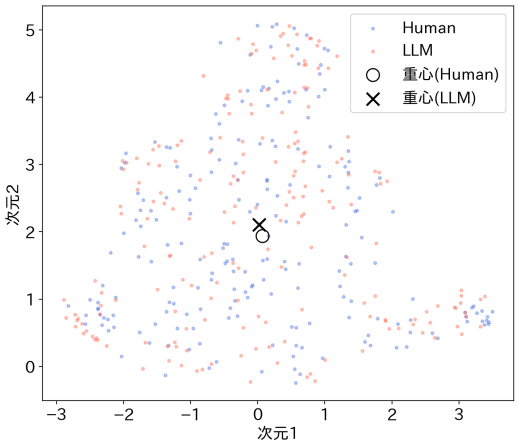
図 A.8: Wikipedia における感情極性の分布（DistilBERT）

A.4　文埋め込みの可視化結果

4.5 節で述べた文埋め込み分析の結果を補足する．図 A.9 および図 A.10 は，それぞれ Wikinews と Wikipedia の文埋め込みの可視化結果である．

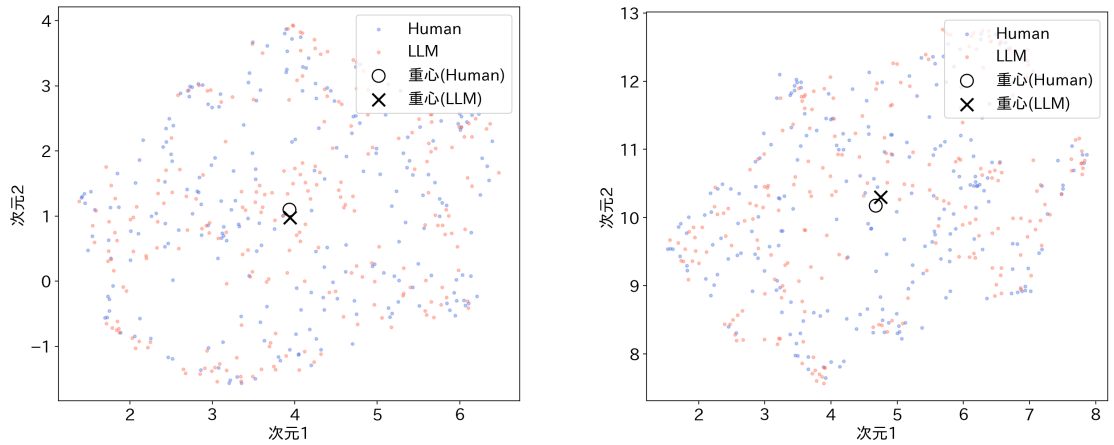


(a) Llama3



(b) Swallow-8B

図 A.9: Wikinews における意味埋め込みの 2 次元可視化

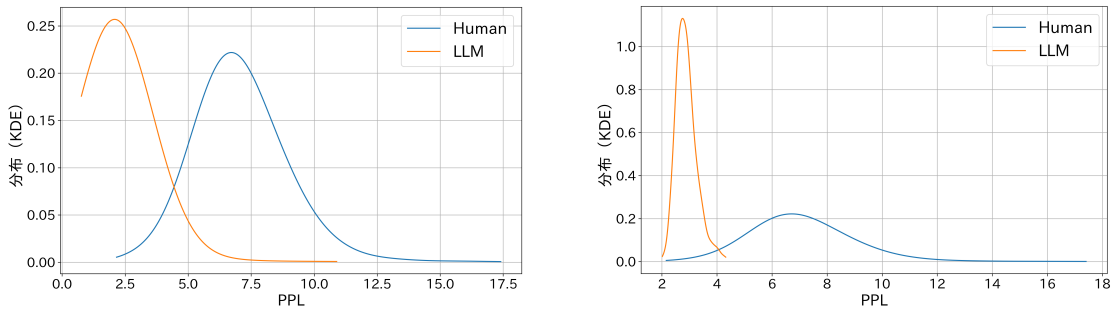


(a) Llama3　　(b) Swallow-8B

図 A.10: Wikipedia における意味埋め込みの 2 次元可視化

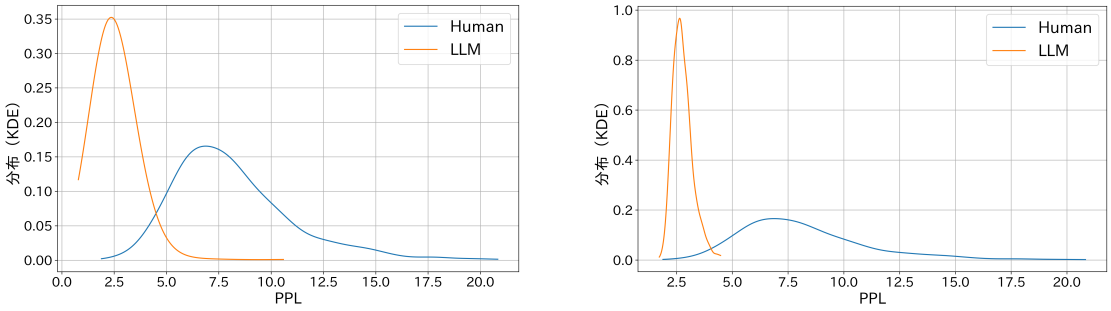
A.5　PPL 分析

4.6 節で述べた PPL の分析結果を補足する．図 A.11 および図 A.12 は，それぞれ Llama3 を計算モデルとしたときの Wikinews と Wikipedia における PPL の分布である．図 A.13 および図 A.14 は，それぞれ GPT-2 を計算モデルとしたときの Wikinews と Wikipedia における PPL の分布である．図 A.15 および図 A.18 は，それぞれ Swallow-8B を計算モデルとしたときの全ドメインにおける PPL の分布である．



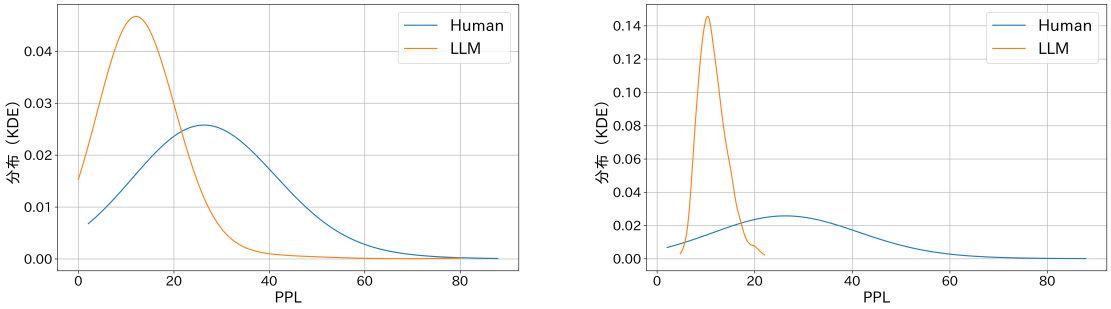
(a) Llama3　　(b) Swallow-8B

図 A.11: Wikinews の PPL 分布 (PPL は Llama3 で算出)



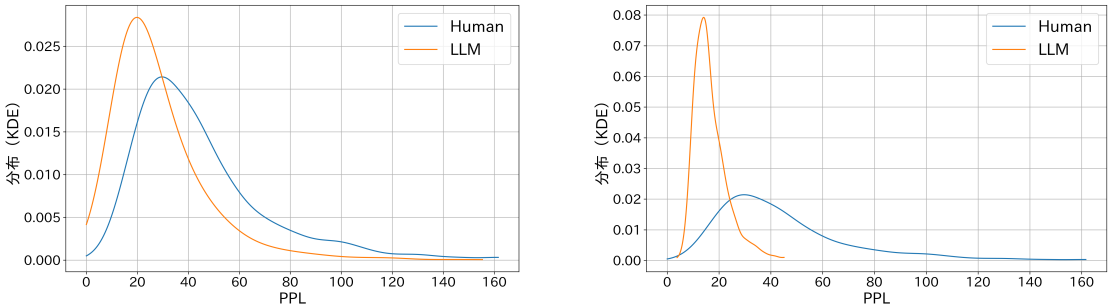
(a) Llama3　　(b) Swallow-8B

図 A.12: Wikipedia の PPL 分布 (PPL は Llama3 で算出)



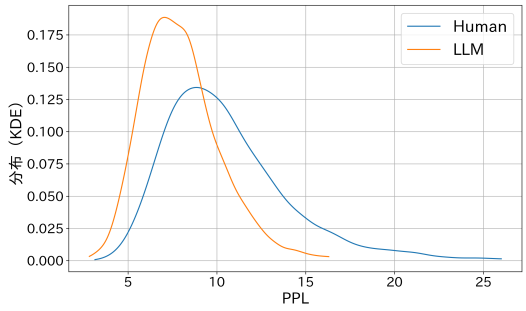
(a) Llama3　　(b) Swallow-8B

図 A.13: Wikinews の PPL 分布 (PPL は GPT-2 で算出)

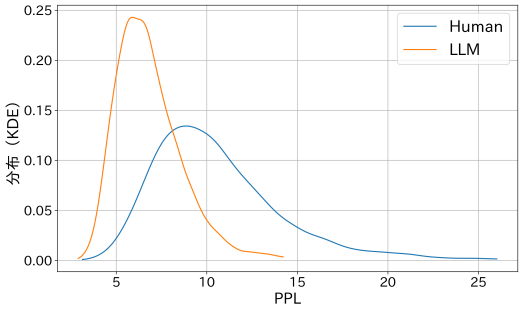


(a) Llama3　　(b) Swallow-8B

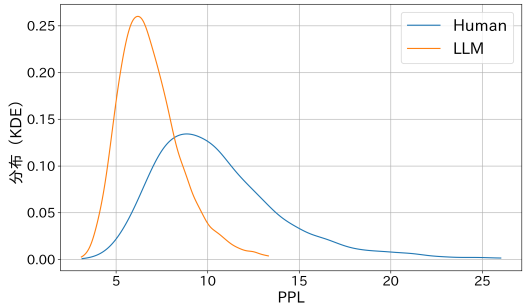
図 A.14: Wikipedia の PPL 分布 (PPL は GPT-2 で算出)



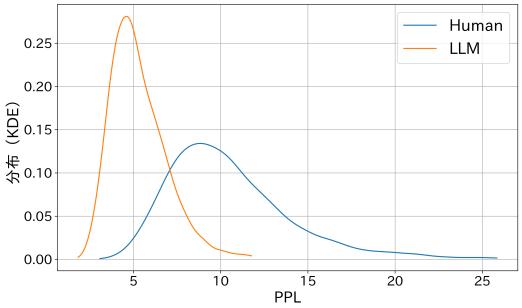
(a) Llama3



(b) Gemini

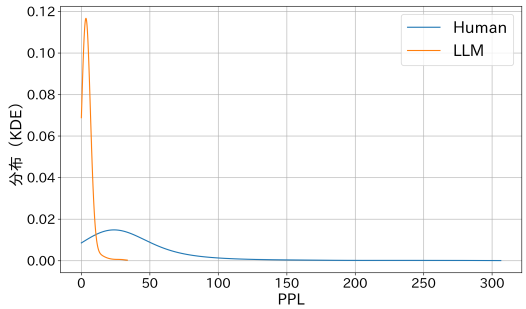


(c) ChatGPT

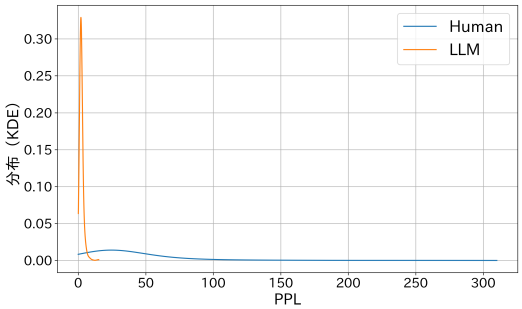


(d) Swallow-8B

図 A.15: 論文の概要の PPL 分布 (PPL は Swallow で算出)

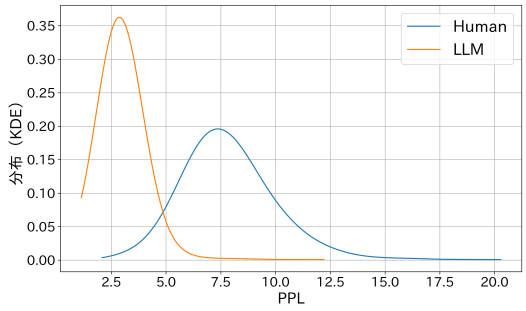


(a) Llama3

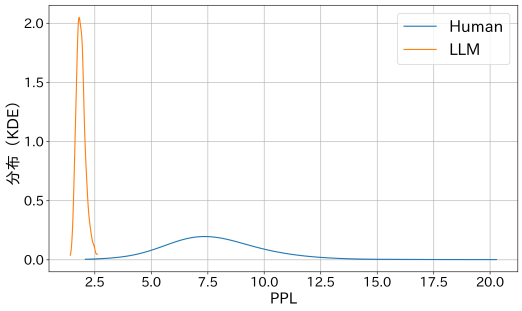


(b) Swallow-8B

図 A.16: Yahoo!知恵袋の PPL 分布 (PPL は Swallow で算出)

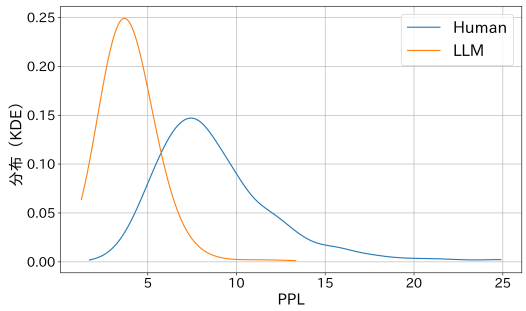


(a) Llama3

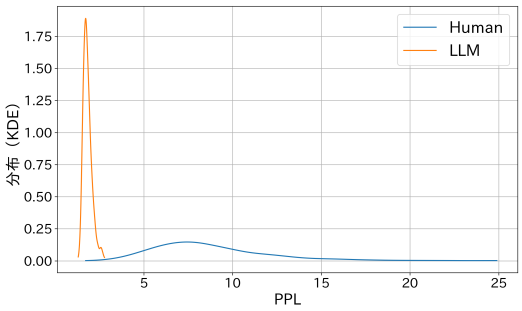


(b) Swallow-8B

図 A.17: Wikinews の PPL 分布 (PPL は Swallow で算出)



(a) Llama3



(b) Swallow-8B

図 A.18: Wikipedia の PPL 分布 (PPL は Swallow で算出)

付 録B　LLM生成テキスト識別モデルの実験結果の補足

本章では，5.2.2 項における RoBERTa 分類器の評価実験のうち，本文で代表例として示したもの以外の分類器の結果を記載する．

B.1　特定のドメインまたはLLMのみを用いた場合（条件：Domain, LLM）

表 B.1 から表 B.3 に，それぞれ「Yahoo!知恵袋」「Wikinews」「Wikipedia」サブセットのみを学習に用いた場合の結果を示す．また，表 B.4 に Swallow-8B を用いて生成したサブセットのみを学習に用いた場合の結果を示す．

表 B.1: Yahoo!知恵袋のデータセットのみで学習した分類器の評価

ドメイン	LLM	Accuracy	Precision	Recall	F1	AUROC
論文の概要	Llama3	0.6025	0.5646	0.8940	0.6921	0.6728
	Gemini	0.6169	0.5724	0.9229	0.7066	0.7443
	ChatGPT	0.6463	0.5874	0.9818	0.7351	0.8425
	Swallow-8B	0.6388	0.5837	0.9668	0.7279	0.8115
Yahoo!知恵袋	Llama3	**0.9900**	**0.9802**	**1.0000**	**0.9900**	**0.9992**
	Swallow-8B	0.9613	0.9278	**1.0000**	0.9626	0.9983
Wikinews	Llama3	0.5710	0.5371	0.9945	0.6975	0.8599
	Swallow-8B	0.5679	0.5364	**1.0000**	0.6983	0.8217
Wikipedia	Llama3	0.5178	0.5091	0.9956	0.6737	0.7134
	Swallow-8B	0.5089	0.5045	**1.0000**	0.6706	0.9583
マイクロ平均		0.6251	0.5753	0.9534	0.7176	0.7889
マクロ平均		0.6521	0.6303	0.9756	0.7554	0.8422

表 B.2: Wikinews のデータセットのみで学習した分類器の評価

ドメイン	LLM	Accuracy	Precision	Recall	F1	AUROC
論文の概要	Llama3	0.5950	0.5527	0.9936	0.7103	0.8225
	Gemini	0.5971	0.5538	0.9979	0.7123	0.8564
	ChatGPT	0.5976	0.5540	0.9989	0.7128	0.9118
	Swallow-8B	0.5955	0.5530	0.9946	0.7108	0.8898
Yahoo!知恵袋	Llama3	0.6633	0.5964	**1.0000**	0.7472	0.9866
	Swallow-8B	0.6961	0.6207	**1.0000**	0.7660	0.9883
Wikinews	Llama3	0.9617	0.9286	**1.0000**	0.9630	0.9991
	Swallow-8B	**0.9728**	**0.9485**	**1.0000**	**0.9735**	**1.0000**
Wikipedia	Llama3	0.6778	0.6093	0.9911	0.7547	0.9431
	Swallow-8B	0.6637	0.5979	**1.0000**	0.7483	0.9911
マイクロ平均		0.6354	0.5784	0.9966	0.7320	0.9055
マクロ平均		0.7021	0.6515	0.9976	0.7799	0.9389

表 B.3: Wikipedia のデータセットのみで学習した分類器の評価

ドメイン	LLM	Accuracy	Precision	Recall	F1	AUROC
論文の概要	Llama3	0.7014	0.6306	0.9722	0.7650	0.8557
	Gemini	0.6961	0.6280	0.9615	0.7597	0.8166
	ChatGPT	0.7127	0.6359	0.9946	0.7758	0.9153
	Swallow-8B	0.7047	0.6321	0.9786	0.7681	0.9065
Yahoo!知恵袋	Llama3	0.8744	0.8083	0.9798	0.8858	0.9597
	Swallow-8B	0.8398	0.7850	0.9333	0.8528	0.9147
Wikinews	Llama3	0.9563	0.9278	0.9890	0.9574	0.9819
	Swallow-8B	0.9484	0.9146	0.9891	0.9504	0.9811
Wikipedia	Llama3	**0.9733**	**0.9532**	0.9956	**0.9739**	0.9991
	Swallow-8B	0.9624	0.9300	**1.0000**	0.9638	**0.9998**
マイクロ平均		0.7542	0.6752	0.9789	0.7992	0.9001
マクロ平均		0.8369	0.7846	0.9794	0.8653	0.9330

表 B.4: Swallow-8B で生成したデータセットのみで学習した分類器の評価

ドメイン	LLM	Accuracy	Precision	Recall	F1	AUROC
論文の概要	Llama3	0.8898	0.9450	0.8276	0.8824	0.9616
	Gemini	0.9706	0.9536	0.9893	0.9711	0.9976
	ChatGPT	0.9754	0.9540	0.9989	0.9759	0.9992
	Swallow-8B	0.9732	0.9538	0.9946	0.9738	0.9992
Yahoo!知恵袋	Llama3	0.9698	0.9429	**1.0000**	0.9706	0.9992
	Swallow-8B	0.9558	0.9184	**1.0000**	0.9574	0.9989
Wikinews	Llama3	0.9836	**0.9835**	0.9835	0.9835	0.9993
	Swallow-8B	**0.9891**	0.9787	**1.0000**	**0.9892**	**1.0000**
Wikipedia	Llama3	0.9267	0.9800	0.8711	0.9224	0.9744
	Swallow-8B	0.9735	0.9496	**1.0000**	0.9741	0.9997
マイクロ平均		0.9551	0.9543	0.9559	0.9551	0.9899
マクロ平均		0.9608	0.9560	0.9665	0.9600	0.9929

B.2 単一のサブセットのみを用いた場合（条件:Single）

表 B.5 から表 B.7 に，論文の概要ドメインにおいて，それぞれ「Swallow-8B」「ChatGPT」「Gemini」のデータのみを学習に用いた場合の結果を示す．また，表 B.8 に，Yahoo!知恵袋ドメインの「Swallow-8B」データのみを用いた場合の結果を示す．

続いて，Wikinews ドメインの単一サブセットを用いた結果を表 B.9（Llama3）および表 B.10（Swallow-8B）に，Wikipedia ドメインの結果を表 B.11（Llama3）および表 B.12（Swallow-8B）にそれぞれ示す．

表 B.5: サブセット「論文の概要・Swallow-8B」のみで学習した分類器

ドメイン	LLM	Accuracy	Precision	Recall	F1	AUROC
論文の概要	Llama3	0.8785	0.9274	0.8212	0.8711	0.9506
	Gemini	0.9663	0.9395	0.9968	0.9673	0.9990
	ChatGPT	**0.9674**	**0.9396**	**0.9989**	**0.9683**	**0.9995**
	Swallow-8B	0.9668	0.9395	0.9979	0.9678	0.9990
Yahoo!知恵袋	Llama3	0.8492	0.8485	0.8485	0.8485	0.9179
	Swallow-8B	0.8122	0.8256	0.7889	0.8068	0.9344
Wikinews	Llama3	0.8525	0.8902	0.8022	0.8439	0.9385
	Swallow-8B	0.9484	0.9365	0.9620	0.9491	0.9872
Wikipedia	Llama3	0.7733	0.8122	0.7111	0.7583	0.8468
	Swallow-8B	0.8982	0.8488	0.9690	0.9050	0.9602
マイクロ平均		0.9265	0.9216	0.9321	0.9268	0.9783
マクロ平均		0.8913	0.8908	0.8897	0.8886	0.9533

表 B.6: サブセット「論文の概要・ChatGPT」のみで学習した分類器

ドメイン	LLM	Accuracy	Precision	Recall	F1	AUROC
論文の概要	Llama3	0.6966	0.9553	0.4122	0.5759	0.9418
	Gemini	0.9765	0.9806	0.9722	0.9763	0.9978
	ChatGPT	**0.9888**	0.9810	**0.9968**	**0.9888**	**0.9993**
	Swallow-8B	0.8652	0.9749	0.7495	0.8475	0.9830
Yahoo!知恵袋	Llama3	0.4975	0.0000	0.0000	0.0000	0.9421
	Swallow-8B	0.5028	0.0000	0.0000	0.0000	0.9215
Wikinews	Llama3	0.5082	**1.0000**	0.0110	0.0217	0.9156
	Swallow-8B	0.5136	**1.0000**	0.0272	0.0529	0.9640
Wikipedia	Llama3	0.5044	**1.0000**	0.0089	0.0176	0.7874
	Swallow-8B	0.5155	**1.0000**	0.0310	0.0601	0.9339
マイクロ平均		0.8025	0.9758	0.6200	0.7582	0.9555
マクロ平均		0.6569	0.7892	0.3209	0.3541	0.9386

表 B.7: サブセット「論文の概要・Gemini」のみで学習した分類器

ドメイン	LLM	Accuracy	Precision	Recall	F1	AUROC
論文の概要	Llama3	0.6683	0.7940	0.4540	0.5777	0.8150
	Gemini	0.9272	0.8919	0.9722	0.9303	0.9827
	ChatGPT	**0.9395**	0.8943	**0.9968**	**0.9428**	**0.9936**
	Swallow-8B	0.8256	0.8671	0.7687	0.8150	0.9241
Yahoo!知恵袋	Llama3	0.5075	0.6667	0.0202	0.0392	0.7626
	Swallow-8B	0.5193	**1.0000**	0.0333	0.0645	0.7414
Wikinews	Llama3	0.5383	**1.0000**	0.0714	0.1333	0.8872
	Swallow-8B	0.5082	**1.0000**	0.0163	0.0321	0.8362
Wikipedia	Llama3	0.5089	**1.0000**	0.0178	0.0349	0.8665
	Swallow-8B	0.5575	0.9643	0.1195	0.2126	0.9238
マイクロ平均		0.8182	0.9790	0.6499	0.7812	0.9348
マクロ平均		0.6500	0.9078	0.3470	0.3782	0.8733

表 B.8: サブセット「Yahoo!知恵袋・Swallow-8B」のみで学習した分類器

ドメイン	LLM	Accuracy	Precision	Recall	F1	AUROC
論文の概要	Llama3	0.5661	0.5385	0.9218	0.6798	0.5984
	Gemini	0.5886	0.5503	0.9668	0.7014	0.6284
	ChatGPT	0.6014	0.5568	0.9925	0.7134	0.7139
	Swallow-8B	0.5934	0.5527	0.9764	0.7059	0.6708
Yahoo!知恵袋	Llama3	**0.9598**	**0.9505**	0.9697	**0.9600**	**0.9958**
	Swallow-8B	0.9558	0.9362	0.9778	0.9565	0.9954
Wikinews	Llama3	0.5383	0.5186	0.9945	0.6817	0.7090
	Swallow-8B	0.5380	0.5198	**1.0000**	0.6840	0.7024
Wikipedia	Llama3	0.5244	0.5127	0.9867	0.6748	0.6668
	Swallow-8B	0.5487	0.5256	**1.0000**	0.6890	0.9126
マイクロ平均		0.5936	0.5532	0.9701	0.7045	0.6821
マクロ平均		0.6415	0.6162	0.9786	0.7447	0.7594

表 B.9: サブセット「Wikinews・Llama3」のみで学習した分類器

ドメイン	使 LLM	Accuracy	Precision	Recall	F1	AUROC
論文の概要	Llama3	0.7673	0.7478	0.8062	0.7759	0.8550
	Gemini	0.7571	0.7429	0.7859	0.7638	0.8549
	ChatGPT	0.8208	0.7706	0.9133	0.8359	0.9166
	Swallow-8B	0.7892	0.7576	0.8501	0.8012	0.8878
Yahoo!知恵袋	Llama3	0.7688	0.6853	0.9899	0.8099	0.9778
	Swallow-8B	0.7624	0.6767	**1.0000**	0.8072	0.9801
Wikinews	Llama3	0.9809	0.9679	0.9945	0.9810	**0.9992**
	Swallow-8B	**0.9810**	**0.9733**	0.9891	**0.9811**	0.9991
Wikipedia	Llama3	0.8378	0.7774	0.9467	0.8537	0.9407
	Swallow-8B	0.8518	0.7751	0.9912	0.8699	0.9914
マイクロ平均		0.8039	0.7686	0.8693	0.8158	0.9047
マクロ平均		0.8317	0.7875	0.9267	0.8480	0.9403

表 B.10: サブセット「Wikinews・Swallow-8B」のみで学習した分類器

ドメイン	LLM	Accuracy	Precision	Recall	F1	AUROC
論文の概要	Llama3	0.5227	0.7763	0.0632	0.1168	0.6757
	Gemini	0.5211	0.7671	0.0600	0.1112	0.7137
	ChatGPT	0.5554	0.8759	0.1285	0.2241	0.8248
	Swallow-8B	0.5773	0.9045	0.1724	0.2896	0.7967
Yahoo!知恵袋	Llama3	0.8442	0.7881	0.9394	0.8571	0.9679
	Swallow-8B	0.8674	0.7946	0.9889	0.8812	0.9855
Wikinews	Llama3	0.9672	0.9830	0.9505	0.9665	0.9979
	Swallow-8B	**0.9973**	**0.9946**	**1.0000**	**0.9973**	**1.0000**
Wikipedia	Llama3	0.6956	0.8793	0.4533	0.5982	0.8687
	Swallow-8B	0.9447	0.9548	0.9336	0.9441	0.9895
マイクロ平均		0.6167	0.8966	0.2632	0.4069	0.7760
マクロ平均		0.7493	0.8718	0.5690	0.5986	0.8820

.

表 B.11: サブセット「Wikipedia・Llama3」のみで学習した分類器

ドメイン	LLM	Accuracy	Precision	Recall	F1	AUROC
論文の概要	Llama3	0.7731	0.7736	0.7719	0.7728	0.8603
	Gemini	0.6083	0.6619	0.4422	0.5302	0.7047
	ChatGPT	0.7699	0.7721	0.7655	0.7688	0.8452
	Swallow-8B	0.7651	0.7699	0.7559	0.7628	0.8458
Yahoo!知恵袋	Llama3	0.8744	0.8627	0.8889	0.8756	0.9229
	Swallow-8B	0.6796	0.7353	0.5556	0.6329	0.7955
Wikinews	Llama3	0.9344	0.9489	0.9176	0.9330	0.9786
	Swallow-8B	0.8832	0.9490	0.8098	0.8739	0.9649
Wikipedia	Llama3	0.9689	0.9451	0.9956	0.9697	**0.9995**
	Swallow-8B	**0.9735**	**0.9496**	**1.0000**	**0.9741**	0.9976
マイクロ平均		0.7681	0.7903	0.7294	0.7586	0.8571
マクロ平均		0.8230	0.8368	0.7903	0.8094	0.8915

表 B.12: サブセット「Wikipedia・Swallow-8B」のみで学習した分類器

ドメイン	LLM	Accuracy	Precision	Recall	F1	AUROC
論文の概要	Llama3	0.7154	0.6525	0.9208	0.7638	0.8244
	Gemini	0.7154	0.6525	0.9208	0.7638	0.8181
	ChatGPT	0.7512	0.6693	0.9925	0.7995	0.9327
	Swallow-8B	0.7357	0.6622	0.9615	0.7843	0.8999
Yahoo!知恵袋	Llama3	0.9196	0.9029	0.9394	0.9208	0.9703
	Swallow-8B	0.9227	0.8958	0.9556	0.9247	0.9705
Wikinews	Llama3	0.9262	0.8818	0.9835	0.9299	0.9784
	Swallow-8B	0.9266	0.8720	**1.0000**	0.9316	0.9907
Wikipedia	Llama3	0.9289	**0.9801**	0.8756	0.9249	0.9738
	Swallow-8B	**0.9801**	0.9617	**1.0000**	**0.9805**	**0.9999**
マイクロ平均		0.7737	0.7018	0.9511	0.8077	0.9005
マクロ平均		0.8522	0.8131	0.9550	0.8724	0.9359

B.3　特定のサブセットを除外した場合（条件：LOFO）

表 B.13 から表 B.15 に，論文の概要ドメインにおいてそれぞれ Gemini，ChatGPT，Swallow-8B のデータを除外した場合の結果を示す．また，表 B.16 には Yahoo!知恵袋ドメインの Swallow-8B データを，表 B.17 および B.18 には Wikinews ドメインデータを，表 B.19 および B.20 には Wikipedia ドメインのデータをそれ

ぞれ除外した場合の結果を掲載する．これらの表における太字は学習データから除外されたサブセットをテストデータとしている場合を表す.

表 B.13: サブセット「論文の概要・Gemini」を除外して学習した分類器

ドメイン	LLM	Accuracy	Precision	Recall	F1	AUROC
論文の概要	Llama3	0.9513	0.9128	0.9979	0.9535	0.9938
	Gemini	**0.9497**	**0.9126**	**0.9946**	**0.9518**	**0.9934**
	ChatGPT	0.9524	0.9130	1.0000	0.9545	0.9936
	Swallow-8B	0.9513	0.9128	0.9979	0.9535	0.9935
Yahoo!知恵袋	Llama3	0.9749	0.9519	1.0000	0.9754	1.0000
	Swallow-8B	0.9558	0.9184	1.0000	0.9574	1.0000
Wikinews	Llama3	0.9699	0.9430	1.0000	0.9707	0.9999
	Swallow-8B	0.9701	0.9436	1.0000	0.9710	1.0000
Wikipedia	Llama3	0.9600	0.9259	1.0000	0.9615	0.9996
	Swallow-8B	0.9447	0.9004	1.0000	0.9476	0.9949
マイクロ平均		0.9533	0.9160	0.9981	0.9553	0.9947
マクロ平均		0.9580	0.9234	0.9990	0.9597	0.9969

表 B.14: サブセット「論文の概要・ChatGPT」を除外して学習した分類器

ドメイン	LLM	Accuracy	Precision	Recall	F1	AUROC
論文の概要	Llama3	0.9695	0.9470	0.9946	0.9702	0.9966
	Gemini	0.9722	0.9473	1.0000	0.9729	0.9990
	ChatGPT	**0.9722**	**0.9473**	**1.0000**	**0.9729**	**0.9988**
	Swallow-8B	0.9711	0.9472	0.9979	0.9718	0.9982
Yahoo!知恵袋	Llama3	1.0000	1.0000	1.0000	1.0000	1.0000
	Swallow-8B	1.0000	1.0000	1.0000	1.0000	1.0000
Wikinews	Llama3	0.9863	0.9733	1.0000	0.9864	0.9999
	Swallow-8B	0.9837	0.9684	1.0000	0.9840	1.0000
Wikipedia	Llama3	0.9867	0.9782	0.9956	0.9868	0.9996
	Swallow-8B	0.9845	0.9700	1.0000	0.9847	0.9996
マイクロ平均		0.9748	0.9535	0.9983	0.9754	0.9985
マクロ平均		0.9826	0.9678	0.9988	0.9830	0.9992

表 B.15: サブセット「論文の概要・Swallow-8B」を除外して学習した分類器

ドメイン	LLM	Accuracy	Precision	Recall	F1	AUROC
論文の概要	Llama3	0.9674	0.9422	0.9957	0.9682	0.9947
	Gemini	0.9695	0.9425	1.0000	0.9704	0.9972
	ChatGPT	0.9695	0.9425	1.0000	0.9704	0.9971
	Swallow-8B	**0.9609**	**0.9415**	**0.9829**	**0.9618**	**0.9931**
Yahoo!知恵袋	Llama3	0.9899	0.9802	1.0000	0.9900	1.0000
	Swallow-8B	0.9945	0.9890	1.0000	0.9945	1.0000
Wikinews	Llama3	0.9727	0.9479	1.0000	0.9733	0.9996
	Swallow-8B	0.9810	0.9634	1.0000	0.9813	1.0000
Wikipedia	Llama3	0.9711	0.9454	1.0000	0.9719	0.9997
	Swallow-8B	0.9580	0.9224	1.0000	0.9597	0.9998
マイクロ平均		0.9684	0.9440	0.9958	0.9692	0.9965
マクロ平均		0.9734	0.9517	0.9979	0.9741	0.9981

表 B.16: サブセット「Yahoo!知恵袋・Swallow-8B」を除外して学習した分類器

ドメイン	LLM	Accuracy	Precision	Recall	F1	AUROC
論文の概要	Llama3	0.9556	0.9200	0.9979	0.9574	0.9975
	Gemini	0.9567	0.9202	1.0000	0.9584	0.9995
	ChatGPT	0.9567	0.9202	1.0000	0.9584	0.9996
	Swallow-8B	0.9551	0.9200	0.9968	0.9568	0.9983
Yahoo!知恵袋	Llama3	0.9799	0.9612	1.0000	0.9802	0.9996
	Swallow-8B	**0.9779**	**0.9674**	**0.9889**	**0.9780**	**0.9995**
Wikinews	Llama3	0.9836	0.9681	1.0000	0.9838	1.0000
	Swallow-8B	0.9864	0.9735	1.0000	0.9866	1.0000
Wikipedia	Llama3	0.9822	0.9657	1.0000	0.9825	0.9999
	Swallow-8B	0.9735	0.9496	1.0000	0.9741	0.9999
マイクロ平均		0.9612	0.9290	0.9987	0.9626	0.9990
マクロ平均		0.9707	0.9466	0.9984	0.9716	0.9994

表 B.17: サブセット「Wikinews・Llama3」を除外して学習した分類器

ドメイン	LLM	Accuracy	Precision	Recall	F1	AUROC
論文の概要	Llama3	0.9492	0.9085	0.9989	0.9516	0.9924
	Gemini	0.9497	0.9086	1.0000	0.9521	0.9982
	ChatGPT	0.9497	0.9086	1.0000	0.9521	0.9970
	Swallow-8B	0.9497	0.9086	1.0000	0.9521	0.9944
Yahoo!知恵袋	Llama3	0.9749	0.9519	1.0000	0.9754	1.0000
	Swallow-8B	0.9834	0.9677	1.0000	0.9836	1.0000
Wikinews	**Llama3**	**0.9836**	**0.9731**	**0.9945**	**0.9837**	**0.9996**
	Swallow-8B	0.9864	0.9735	1.0000	0.9866	1.0000
Wikipedia	Llama3	0.9844	0.9698	1.0000	0.9847	0.9997
	Swallow-8B	0.9779	0.9576	1.0000	0.9784	0.9999
マイクロ平均		0.9565	0.9202	0.9996	0.9583	0.9964
マクロ平均		0.9689	0.9428	0.9993	0.9700	0.9981

表 B.18: サブセット「Wikinews・Swallow-8B」を除外して学習した分類器

ドメイン	LLM	Accuracy	Precision	Recall	F1	AUROC
論文の概要	Llama3	0.9652	0.9402	0.9936	0.9662	0.9966
	Gemini	0.9674	0.9405	0.9979	0.9683	0.9989
	ChatGPT	0.9684	0.9406	1.0000	0.9694	0.9995
	Swallow-8B	0.9642	0.9401	0.9914	0.9651	0.9973
Yahoo!知恵袋	Llama3	0.9749	0.9519	1.0000	0.9754	0.9988
	Swallow-8B	0.9669	0.9375	1.0000	0.9677	0.9996
Wikinews	Llama3	0.9781	0.9579	1.0000	0.9785	0.9999
	Swallow-8B	**0.9810**	**0.9634**	**1.0000**	**0.9813**	**1.0000**
Wikipedia	Llama3	0.9733	0.9494	1.0000	0.9740	0.9984
	Swallow-8B	0.9491	0.9076	1.0000	0.9516	0.9996
マイクロ平均		0.9670	0.9409	0.9966	0.9679	0.9983
マクロ平均		0.9688	0.9429	0.9983	0.9697	0.9989

表 B.19: サブセット「Wikipedia・Llama3」を除外して学習した分類器

ドメイン	LLM	Accuracy	Precision	Recall	F1	AUROC
論文の概要	Llama3	0.9401	0.8960	0.9957	0.9432	0.9961
	Gemini	0.9422	0.8964	1.0000	0.9453	0.9979
	ChatGPT	0.9422	0.8964	1.0000	0.9453	0.9984
	Swallow-8B	0.9422	0.8964	1.0000	0.9453	0.9974
Yahoo!知恵袋	Llama3	0.9698	0.9429	1.0000	0.9706	1.0000
	Swallow-8B	0.9724	0.9474	1.0000	0.9730	0.9994
Wikinews	Llama3	0.9645	0.9333	1.0000	0.9655	0.9996
	Swallow-8B	0.9755	0.9534	1.0000	0.9761	1.0000
Wikipedia	**Llama3**	**0.9467**	**0.9718**	**0.9200**	**0.9452**	**0.9885**
	Swallow-8B	0.9668	0.9378	1.0000	0.9679	0.9998
マイクロ平均		0.9465	0.9066	0.9954	0.9489	0.9974
マクロ平均		0.9562	0.9272	0.9916	0.9578	0.9977

表 B.20: サブセット「Wikipedia・Swallow-8B」を除外して学習した分類器

ドメイン	LLM	Accuracy	Precision	Recall	F1	AUROC
論文の概要	Llama3	0.9609	0.9301	0.9968	0.9623	0.9970
	Gemini	0.9625	0.9303	1.0000	0.9639	0.9987
	ChatGPT	0.9625	0.9303	1.0000	0.9639	0.9988
	Swallow-8B	0.9593	0.9299	0.9936	0.9607	0.9977
Yahoo!知恵袋	Llama3	0.9950	0.9900	1.0000	0.9950	1.0000
	Swallow-8B	0.9834	0.9677	1.0000	0.9836	1.0000
Wikinews	Llama3	0.9754	0.9529	1.0000	0.9759	1.0000
	Swallow-8B	0.9810	0.9634	1.0000	0.9813	1.0000
Wikipedia	Llama3	0.9756	0.9534	1.0000	0.9761	0.9952
	Swallow-8B	**0.9757**	**0.9536**	**1.0000**	**0.9762**	**0.9993**
マイクロ平均		0.9651	0.9363	0.9981	0.9662	0.9982
マクロ平均		0.9731	0.9501	0.9990	0.9739	0.9987

付 録C　サブセット間のデータ数統一実験の補足

本章では，5.2.5.2 で述べたドメイン間のデータ数統一実験のうち，本文で割愛した「Wikinews」および「Wikipedia」の各ドメインのみを用いて学習した場合の評価結果を記載する．表 C.1 に，データ数を統一した条件下で「Wikinews」ドメインのみを学習に用いた場合の結果を示す．また，表 C.2 に，同様の条件下で「Wikipedia」ドメインのみを学習に用いた場合の結果を示す.

表 C.1: サブセット間のデータ数統一実験 - Wikinews のみの分類器による評価

ドメイン	LLM	Accuracy	Precision	Recall	F1	AUROC
論文の概要	Swallow-8B	0.7556	0.7018	0.8889	0.7843	0.8621
	Llama3	0.7278	0.6881	0.8333	0.7538	0.7889
	ChatGPT	0.7667	0.7069	0.9111	0.7961	0.8723
	Gemini	0.7333	0.6909	0.8444	0.7600	0.8275
Yahoo!知恵袋	Swallow-8B	0.6722	0.6040	**1.0000**	0.7531	0.9842
	Llama3	0.6722	0.6054	0.9889	0.7511	0.9731
Wikinews	Swallow-8B	**0.9833**	**0.9677**	**1.0000**	**0.9836**	**0.9991**
	Llama3	0.9556	0.9184	**1.0000**	0.9574	0.9970
Wikipedia	Swallow-8B	0.7222	0.6429	**1.0000**	0.7826	0.9757
	Llama3	0.6556	0.5933	0.9889	0.7417	0.8794
Average		0.7644	0.6941	0.9456	0.8006	0.9221

表 C.2: サブセット間のデータ数統一実験 – Wikipedia のみの分類器による評価

ドメイン	LLM	Accuracy	Precision	Recall	F1	AUROC
論文の概要	Swallow-8B	0.8111	0.7500	0.9333	0.8317	0.9005
	Llama3	0.7833	0.7383	0.8778	0.8020	0.8543
	ChatGPT	0.8278	0.7565	0.9667	0.8488	0.9002
	Gemini	0.7722	0.7333	0.8556	0.7897	0.8168
Yahoo!知恵袋	Swallow-8B	0.8444	0.7870	0.9444	0.8586	0.9206
	Llama3	0.8722	0.8131	0.9667	0.8832	0.9696
Wikinews	Swallow-8B	0.9667	0.9565	0.9778	0.9670	0.9863
	Llama3	0.9333	0.9063	0.9667	0.9355	0.9798
Wikipedia	Swallow-8B	0.9667	0.9375	**1.0000**	0.9677	**0.9995**
	Llama3	**0.9722**	**0.9570**	0.9889	**0.9727**	0.9975
Average		0.8750	0.8274	0.9478	0.8835	0.9438

付 録D　分類器の確信度の可視化

本付録では,「条件:All」で学習した RoBERTa 分類器について，全サブセットの結合テストセット内のサンプルに対する予測の確信度の分布を報告する．モデル出力として，クラス 1（LLM）に対応する確率を $p_1 = P(\mathrm{LLM} = 1)$ とする．p_1 は分類器による予測の確信度を表すとみなせる．縦軸を p_1，横軸を各サンプルを p_1 の昇順に並べ替えたときのインデックス（sample sorted by p_1）とし，サンプル毎の p_1 をプロットすることでテストデータ全体に対する予測確率の分布を可視化する．結果を図 D.1 に示す.

この図から，多くのサンプルで p_1 が 0 付近または 1 付近に強く張り付いており，中間的な確率（例えば 0.2〜0.8 程度）を取るサンプルが極めて少ないことが確認できる．この傾向は，分類器が多数のサンプルに対して非常に強い確信度をもって予測していることを示している.

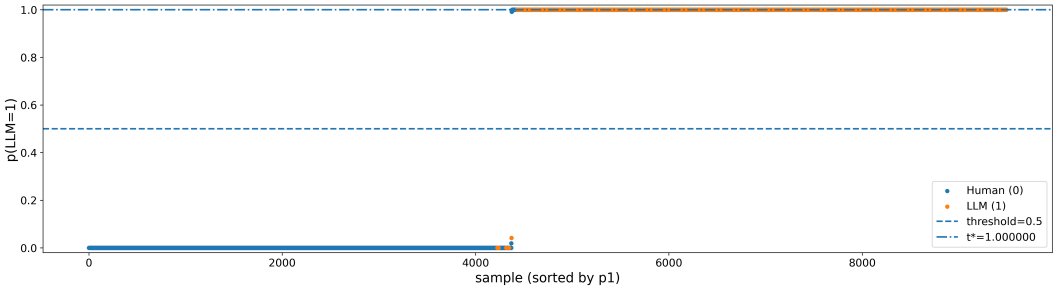


図 D.1: 分類器の確信度の可視化