論文要約

LLM関連

Is Conversational XAI All You Need? Human-AI Decision Making With a Conversational XAI Assistant 会話型XAIは全てを解決するのか？会話型 XAIアシスタントによる人間と AIの意思決定

https://github.com/delftcrowd/IUI2025_ConvXAI

概要: 説明可能な人工知能（ XAI）は、AIシステムの予測をユーザーが理解しやすくするための手法であり、これに会話型ユーザーインターフェース（ Conversational UI）を統合することで、より直感的でインタラクティブな説明を提供できる可能性がある。本研究では、従来の XAIダッシュボードと比較し、会話型 XAIインターフェース がユーザーの AI理解、信頼、依存に与える影響を分析した。

結果として、会話型 XAIはユーザーのエンゲージメントを向上させる一方で、過信（ over-reliance）を引き起こしやすくなることが判明した。特に、大規模言語モデル（ LLM）を組み込んだ会話型 XAIは、説明の説得力が増す ものの、実際の理解や適切な依存にはつながらないリスクがあることが分かった。本研究は、今後の XAI設計において、ユーザーの適切な信頼と理解を促進するデザインが必要であることを示唆する。

**研究の背景 **

- **XAIの重要性 **: AIシステムの予測はブラックボックス化しやすく、説明がないとユーザーは適切に活用できない。

- **既存の課題 **:

  - 多くのXAI手法は静的な説明しか提供できず、 **ユーザーごとに異なる理解度や情報ニーズに対応できない **。

  - **「説明の過信」** によって、ユーザーが AIの出力を盲信し、適切な判断を下せなくなるリスクがある。

- **会話型XAIの可能性 **:

  - **対話的に説明をカスタマイズ ** することで、ユーザーの疑問に即座に対応し、より直感的な理解を促進できる。

  - しかし、**会話型XAIが必ずしも適切な理解を促すわけではない ** 可能性がある。

ScoreFlow: Mastering LLM Agent Workflows via Score-based Preference Optimization ScoreFlow: スコアベースの選好最適化による LLMエージェントワークフローの最適化

https://github.com/Gen-Verse/ScoreFlow

概要: ScoreFlowは、大規模言語モデル（ LLM）を用いたマルチエージェントワークフローの自動最適化を目指すフレームワークである。従来の最適化手法では、ワークフロー構造が固定的であり、適応性や拡張性に制約があった。本研究では、勾配ベースの最適化手法（ Score-DPO）を導入し、評価スコアを直接考慮することでワークフローの柔軟な最適化を実現する。

本研究の主な成果は以下の通り ：

• ワークフロー最適化の自動化 ：従来の手作業による設計から、タスクに応じた適応的なワークフロー生成へ移行。

• スコアを考慮した選好最適化 ：評価スコアを利用することで、学習の安定性と収束速度を向上。

• 高いパフォーマンス：6つのベンチマークにおいて、既存手法を平均 8.2%上回る性能を達成。

• 小型モデルの有効活用 ：小型モデル（ Llama-3.1-8B）でも、大型モデル（ GPT-4o）と同等以上の性能を実現。

**先行研究との比較：本研究の優位性 **

| 手法 | ワークフローの柔軟性 | 最適化手法 | スコア考慮 | パフォーマンス向上 |

| --- | --- | --- | --- | --- |

| AFlow | 一部適応可能 | Monte Carlo Tree Search | なし | 中程度 |

| ADAS | コードベースのワークフロー探索 | 離散最適化 | なし | 低 |

| GPTSwarm | グラフベース | 強化学習 | なし | 低 |

| DyLAN | LLMエージェントの通信構造最適化 | 限定的 | なし | 低 |

| **ScoreFlow (本研究)** | **完全適応型** | **勾配ベース最適化 ** | **あり** | **高（8.2%向上）** |

GENERAL SCENE ADAPTATION FOR VISION-AND-LANGUAGE NAVIGATION 視覚と言語を用いたナビゲーションのための一般的なシーン適応

概要: Vision-and-Language Navigation (VLN) は、エージェントが自然言語の指示をもとに環境を移動するタスクであり、従来の研究では　　ゼロショット能力（未知の環境での一般化）が主に評価されていた。しかし、現実のナビゲーションシステムでは、エージェントが同じ環境で繰り返し動作し、学習しながら適応すること　　が求められる。

本研究では、新たに　General Scene Adaptation for VLN（GSA-VLN） を提案し、エージェントが継続的にナビゲーション環境へ適応できる仕組みを構築する。これを実現するため、以下の要素を導入:

1. GSA-R2Rデータセットの構築:既存の　VLN データセットの課題（環境と指示の多様性不足）を克服

2. 三段階の指示生成パイプライン:　LLM を活用し、指示の多様性を向上

3. 新手法「 Graph-Retained DUET（GR-DUET）」の開発:エージェントの長期適応を実現するメモリベースのナビゲーショングラフを活用

**2. 既存研究との比較と本研究の貢献　**

(1) 従来の　VLN の課題

🔹 **環境適応の欠如 **

- **既存の　VLN は、1回限りの指示実行を前提としており、継続的な学習ができない　**

- **実世界ではロボットが同じ環境で繰り返し動作するため、継続的な適応が必要　**

🔹 **ナビゲーション環境の多様性不足　**

AgentRec: Agent Recommendation Using Sentence Embeddings Aligned to Human Feedback AgentRec: 文埋め込みを人間のフィードバックに基づいて調整したエージェント推奨

https://github.com/joshprk/agentrec

概要: 多エージェントシステムがタスクに最適なエージェントを推奨する問題に対処するため、 Sentence-BERT (SBERT) を拡張した新しいアーキテクチャを提案しています。この方法では、自然言語プロンプトを文埋め込みに変換し、コサイン類似度を使用してタスクに適したエージェントを選択します。本システムは、以下の特徴を持ちます。

計算コストが低い。

新しいクラスに適応可能。

解釈可能かつ任意の評価基準を使用して制御可能。

テストデータで 92.2%のトップ1精度を達成し、各分類に 300ミリ秒以下を要する。

さらに、合成データセットを使用してエージェント推奨を実現し、これを公開しています。

技術や手法

データセット

- **データ生成 :** 合成データセットを使用(10,000プロンプト、8エージェント)。各エージェントごとに 1,250プロンプトを含む。

- **非重複化 :** MinHash を用いたデータの重複削除。

- **データ分割 :** 訓練用 (N=8000)、テスト用 (N=2000) に分割。

RAMQA: A Unified Framework for Retrieval-Augmented Multi-Modal Question Answering RAMQA: 情報検索を補助するマルチモーダル質問応答のための統一フレームワーク

https://github.com/TonyBY/RAMQA

概要: RAMQAは、テキストと画像を統合したマルチモーダル情報検索補助質問応答（ MRAQA）のためのフレームワークを提案しています。従来のランク付け手法がエンコーダーモデルに依存している点を改善し、LLMを利用可能にします。具体的には以下のアプローチが採用されています:

• マルチモーダルデータのランク付け: LLaVA（視覚・言語統合モデル）を用いてポイントワイズランカーを訓練。

• 生成型ランク付け: LLaMAモデルを用いて、指示調整および生成的学習で文書を再ランク付け。

• 性能向上の工夫: 文書候補の順列生成を通じて、ランキングモデルのバイアスを軽減。

**1. RankLLaA: マルチモーダルポイントワイズランカー **

概要

RankLLaVAは、マルチモーダルデータ（テキスト＋画像）の初段階ランク付けを行うモデルです。 LLaVA（マルチモーダル大規模言語モデル）をベースとして、ポイントワイズ（ 1つのクエリに対し 1つの文書の関 連性を評価）手法を採用しています。

**具体的な手順 **

1. **入力データの準備 **:

- クエリ Qi と文書 di （テキスト部分 dtext と画像部分 dimage）を準備。

Understanding and Supporting Formal Email Exchange by Answering AI-Generated Questions AI生成の質問に回答することでフォーマルなメールのやり取りを理解し支援する

概要: フォーマルなメールの返信は、丁寧な表現・適切な言葉遣い・送信者の意図を理解する必要があり、時間と認知的負担が大きい。

• LLMを活用し、受信メールを解析して質問を生成。

• ユーザーが簡潔な質問に答えるだけでフォーマルな返信文を自動生成する　QA（質問応答）アプローチを提案。

• プロトタイプシステム　ResQ を開発し、制御実験（ N=12）とフィールド実験（ N=8）を実施。

• 従来のプロンプトベース手法と比較し、 QAアプローチは返信の効率を向上させ、認知負担を軽減しつつ、高品質な返信を維持できることを示した。

• メール返信プロセスや対人関係への影響、 QAアプローチの課題について考察。

**手法: QAアプローチの実行プロセス **

本研究では、**GPT-4o** を活用した **ResQ** を開発し、フォーマルメールの返信支援を行う。

**1. システムの実行プロセス **

1. **受信メールの解析 **

   - LLMがメール本文を解析し、**必要な回答要素（日時、意図、行動）を抽出 **。

   - **情報分類（依頼、質問、確認など） ** を実施。

ATMOSSCI-BENCH: Evaluating the Recent Advance of Large Language Model for Atmospheric Science ATMOSSCI-BENCH: 大規模言語モデルの大気科学における最近の進展を評価する

https://github.com/Relaxed-System-Lab/AtmosSci-Bench\

概要: 大気科学は、複雑な物理現象と異種データ(例　: 温度、風速、気圧、放射データなど)で使用できる　ATMOSSCI-BENCH という新しい LLMベンチマークを提案し、 5つの主要カテゴリ(水文学、大気力学、大気物理学、地球物理学、物理海洋学)にわたる　LLMの性能を体系的に評価　する。特に、以下の要素を重視して設計された。

大学院レベルの選択式問題(　MCQ)を用いた、精度の高い評価

テンプレートベースの問題生成フレームワークによる、多様かつスケーラブルな設計

物理・数学的推論、数値計算、誤答選択肢の精巧な設計

代表的なLLMを4つのカテゴリ(指示チューニング、数学特化、推論特化、気候特化)に分類し詳細な評価

モデルの「推論能力」「数値計算精度」「シンボリック変動耐性」の評価を通じて、強みと弱点を明確化

**2. ATMOSSCI-BENCH の設計と技術的特徴 **

**(1) 質問設計 **

- **5つの主要カテゴリに分類 **
    - **水文学(Hydrology)** - 水の循環、降水量、地表水・地下水の動態
    - **大気力学(Atmospheric Dynamics)** - 大気の運動、気象パターン、循環システム
    - **大気物理学(Atmospheric Physics)** - 放射、雲形成、熱力学

Agentic Bug Reproduction for Effective Automated Program Repair at Google Google における効果的な自動プログラム修正のためのエージェント型バグ再現

概要: バグ再現と修正の課題

ソフトウェア開発では、バグの修正が不可欠であり、その第一歩として　バグの再現 が重要となる。しかし、多くのバグレポートは不完全で、開発者が手動でバグを再現するのは非常に困難である。そのため、以下の課題が存在する:

• バグ再現テスト(BRT: Bug Reproduction Test) がバグレポートに含まれないことが多い。

• BRT の作成には バグの原因特定・適切なテストの記述 という時間のかかる作業が必要。

• 自動プログラム修正(APR: Automated Program Repair)において、バグを適切に再現できなければ正しい修正が適用できない。

本研究では、LLM(大規模言語モデル)を活用した BRT の自動生成 に焦点を当て、Google の大規模プロプライエタリコードベースに適用可能な新しい手法　BRT Agent を提案する。

**既存技術と課題 **

**2.1 LIBRO(従来手法)の限界 **

LIBRO は、LLM を用いた BRT 生成手法として提案されている。しかし、 Google の環境で適用すると以下の問題が発生した。

1. **プロプライエタリコードへの適用が困難 **

    - Google 特有の API やライブラリを適切に扱えない。

2. **エラーハンドリングの不足 **

    - 生成された BRT の多くがビルドエラーで実行できない。

JRE-L: Journalist, Reader, and Editor LLMs in the Loop for Science Journalism for the General Audience JRE-L: 科学ジャーナリズムにおけるジャーナリスト、読者、編集者　LLM のループ

概要: JRE-Lは、科学ジャーナリズムを自動化する新しいフレームワークであり、ジャーナリスト・読者・編集者の役割を持つ　3つのLLMが協力して記事を改善する。ジャーナリスト　LLMが科学論文を一般向けに要約し、読者 LLMがその記事を読み、理解度に基づいてフィードバックを提供。編集者　LLMがそのフィードバックを評価し、改訂の提案を行う。このループを繰り返すことで、既存手法よりも高い可読性を持つ記事を生成できることを実験で確認した。

- **LLMを3つの役割（ジャーナリスト、読者、編集者）に分け、協調して記事を改善するプロセスを構築。　**

- **読者LLMによるフィードバックを用いることで、可読性を向上させる新しいアプローチ。　**

- **GPT-4などの高性能モデルと比較しても、読みやすさ（　readability）が向上することを実験で確認。 **

**技術・手法 **

**JRE-Lの構成要素 **

1. **ジャーナリスト　LLM**

   - 科学論文を基に一般向けの記事を生成する。

   - LLMのプロンプトにより、単に論文を要約するのではなく、一般人向けのストーリーテリングを意識した書き方を促す。

2. **読者 LLM**

AUTOMATED ASSIGNMENT GRADING WITH LARGE LANGUAGE MODELS: INSIGHTS FROM A BIOINFORMATICS COURSE 大規模言語モデルによる課題の自動採点 : バイオインフォマティクスコースからの洞察

概要: この研究では、大規模言語モデル(LLMs)を用いた課題採点の実践的評価を行い、その有効性を検証した。

目的: 多数の学生が受講する授業で、高品質なフィードバックを効率的に提供すること。

手法: 2024/25年度のバイオインフォマティクス導入コースにて、 36の文章ベースの課題に対して 6つのLLMを用い、学生が受け取るフィードバックの品質と採点精度を比較。

結果: 適切なプロンプト設計により、 LLMは人間の採点者と同等の採点精度とフィードバック品質を実現できる。また、オープンソース LLMは商用LLMと同様の性能を発揮。

技術や手法

1. **プロンプト構造 **:
    - **System Prompt**: 採点とフィードバックの基本的な指針を提供。
    - **User Prompt**: 課題固有の質問、模範解答、採点基準、過去の採点例を含む。
    - **出力形式 **: JSON形式で得点、基準の満足度、フィードバックを記述。
2. **採点基準の準備 **:
    - 授業担当者と教員アシスタント(TA)が採点基準を作成し、 LLMとの一致度を検証。
    - 修正は最小限に留め、過剰な最適化を防止。
3. **モデル評価 **:
    - 商用モデル(GPT-4o)とオープンソースモデル(Llamaシリーズ)の性能を比較。

Non-literal Understanding of Number Words by Language Models 言語モデルによる数詞の非字義的理解

概要: 背景

人間は数値を単なる数量情報としてではなく、文脈や話し手の意図を考慮して解釈する。例えば:

- 「このコーヒー、10000ドルしたよ！」
  - ➡ 実際に10000ドルではなく、「とても高い」という **誇張表現（Hyperbole）**と理解
- 「この時計は 50ドルだったよ」
  - ➡「約50ドル」という **語用論的ハロー効果（ Pragmatic Halo）**が働く

近年、大規模言語モデル（ LLMs）は急速に進化しているが、非字義的な数詞の解釈を人間のように行えているのか？

この問いを解明するために、以下の 3点を検証:

1. LLMsは人間と同じように数詞を非字義的に解釈するのか？

2. 数詞の解釈において、 LLMsと人間の違いはどこにあるのか？

3. 語用論モデル（ Rational Speech Act: RSA）を用いた推論で、LLMsの解釈を改善できるか？

研究の主な結論

1. LLMsは数値の知識を持っているが、話し手の意図を推論できない
   - LLMsは「10000ドルは高い」という知識を持つが、「 10000ドルした」は誇張表現だと理解できない
2. LLMsは語用論的ハロー効果を適切に適用できない
   - 人間:「50ドル」➡ 約50ドル、「48ドル」➡ 正確に48ドル
   - LLMs:「50ドル」も「48ドル」も正確な値として扱う

LP-LM: No Hallucinations in Question Answering with Logic Programming LP-LM：論理プログラミングによる質問応答での幻覚（ハルシネーション）を起こさない仕組み

概要: LP-LMは英語文を論理的に解析し、知識ベースに含まれる事実のみを照合する　Prologを使い、幻覚を防ぐ。幻覚を防ぐため、知識ベースの事実と単一化して答えを得る。

- 「英語文を論理プログラミング（DCG)で解析する」
- 「解析結果を知識ベース（KB)に照合する」
- 「KBに無い情報は答えられないため、幻覚が起こらない」

という仕組みであり、大規模言語モデル（　LLM）が抱える“幻覚”の問題を抑制できる。

先行研究と比べて

1. **LLMでの幻覚問題を根本的に排除　**
    - GPTなどのLLMは膨大なテキストを統計的に学習する結果、真偽不確かな内容を　“もっともらしく”生成してしまう（幻覚）という欠点がある。
    - RAG（Retrieval-Augmented Generation）でも外部知識を参照するが、最終的に文を生成する段階で統計的推定に依拠する以上、幻覚を完全に防ぐのは難しい。
    - **LP-LMは** ロジックをベースにした事実照合により、 KBにない内容を「作り出す」ステップそのものを排除している。
2. **PCFG＋DCG＋タブリングによる効率的解析　**
    - NLPで広く知られる CYKやEarleyと同様に文法に基づく解析を行うが、 DCGを使うことで Prologに自然言語文法を直接埋め込み、推論エンジンのタブリング機能を活用できる。
    - 大規模文法を用いた実験では、 NLTKに実装された Viterbiパーサなどの従来型よりも高速に動作する結果が示されている。
3. **LLMとの比較実験でも幻覚の抑制を実証　**
    - 論文中の例では、「 Furosemideが一時的な難聴を引き起こす」「　Fir treesが人間の肺で育つ」など本来あり得ない　or稀な事実をわざと入力したケースで、 GPT系モデルが“それらしい回答 ”を生成する一方、LP-LMはKBに存在しなければ「答えがない」もしくは「　KBにない」と返すため、誤情報が出ないことを示している。

LLM-powered Multi-agent Framework for Goal-oriented Learning in Intelligent Tutoring System 学習目標達成を指向する LLMマルチエージェントフレームワークによるインテリジェント・チュータリング・システム

概要: 学習者の具体的な目標達成を支援するインテリジェント・チュータリング・システム (ITS)を提案しています。LLMを複数の役割で協調させる「マルチエージェント」構成を採用し、 (1)学習者の目標から必要スキルを抽出する、(2)学習者の進捗や嗜好をモデル化して継続的に更新する、 (3)学習パスを最適化し、(4)検索などの外部データを活用して正確な教材を生成する、といった機能を一貫して実現します。従来の対話型 LLMが受動的に応答するだけであったのに対し、本システムでは学習者を能動的にゴールへ誘導できる点が特徴です。

2. 従来研究との比較・新規性

1. **従来の ITSとの比較 **

   - 多くの ITSは学習者モデルや教材管理などを別々の機械学習モジュールで構成しており、拡張性と統合性に制約がありました。

   - 本研究では LLMを「ゴール分析・スキル抽出・教材生成」などの複数タスクに振り分けるマルチエージェント方式を導入しているため、 **一元化されたフレームワーク **と**柔軟な対応 **が可能になります。

2. **対話型 LLMとの比較 **

   - 既存の対話型 LLMは「質問に答える」形が中心で、学習者を目標達成まで導く流れを設計する機能が弱い傾向にありました。

   - 本システムでは学習パスの自動編成や学習者のモデル更新など、 **目標指向型 (Goal-Oriented)のプロアクティブなアプローチ **を実装し、従来の受動的なやりとりから大きく進化しています。

3. **新規性と利点 **

   - 学習者の行動ログから推定したプロファイルをもとに、 **学習パスの進行・難易度・教材形式を最適化 **する機能を備えています。

   - 外部検索を取り入れた RAG(Retrieval-Augmented Generation)により、**最新かつ正確な学習教材 **を自動生成できます。

   - 企業向けプラットフォームへの実装も行われており、現場レベルでの有効性を検証済みです。

QualityFlow: An Agentic Workflow for Program Synthesis Controlled by LLM Quality Checks QualityFlow：LLMの品質チェックによって制御されるプログラム合成のためのエージェント的ワークフロー

概要：QualityFlowはLLMエージェントを協調させる仕組みを使い、高精度のプログラム合成を行う。テストを想定実行することでコードの正しさを判定し、適宜デバッグや問題の再解釈を組み合わせて精度を向上させる。コードが合格基準に達しなかった場合は、原因を推定し再度生成やテストを行うため、無駄な修正を繰り返さずに最適な解を得られる。

先行研究と比べて優れている点

1. **ユニットテストの想定実行（ Imagined Execution）**
   - 評価用テストを「実行」せずに、 LLM自身がテストの入力から出力を段階的に推論し合否を判断する。
   - ベンチマークによっては評価テストをそのまま実行することが不正確な評価やラベルリークの原因になるが、この想定実行により正当性を高精度にチェックできる。
2. **Quality Checkerによるワークフロー全体の制御 **
   - コードを生成・修正しても正解に近づかない場合は、明示的に「不合格」と判定して次のエージェントを呼び出す、あるいは再生成や巻き戻しが可能。
   - 途中で既に正しいプログラムを得られた場合は即座に確定するため、過度なデバッグで正解コードを壊してしまうリスクを下げられる。
3. **テスト自体の品質評価（ Test Quality Checker）**
   - 新たに生成した追加テストが間違っていると、誤ったバグ指摘で正しいコードを破壊しかねない。
   - QualityFlowではテスト自体の信頼性も検査するため、誤ったテストによるデバッグ崩壊を防ぐ設計が施されている。
4. **多様なプロンプト活用（ Diversified Prompting）**
   - 同じLLMでも入力プロンプトを少し変えるだけで出力結果が多様化し、それらを候補として比較・採択する。
   - 多数決や一発生成に比べ「どれかひとつが正解にヒットする」可能性が高まり、さらに Quality Checkerが正しい解を選別することで合成精度を一段と高める。

STP: Self-play LLM Theorem Provers with Iterative Conjecturing and Proving STP：反復的な定理予想と定理証明を組み合わせた自己対戦型　LLM定理証明手法

概要: STP で 自己対戦型の定理証明　を行い、LLMを使い、定理証明の精度を向上する方法を提案する。

定理を証明するため **Conjecturer（予想者）とProver（証明者）**を切り替え、未証明定理の学習効率を高める。

STPを実現するため Conjecturerが難しすぎず簡単すぎない定理を生成する。

その結果 Proverが証明に成功したら再学習し、お互いを強化する。

4. 先行研究と比べてどこがすごいのか

1. **既存手法は成功報酬が希薄　**
   - Expert Iterationや強化学習ベースのアプローチでは「証明が成功したときにのみ学習信号が得られる」ため、証明が難しい問題だと正解証明が極端に少なくなり学習が進みにくい。
   - さらに、元データセットの定理の難度や数が固定されているので、高度な問題への拡張やより高い精度の達成が停滞しがちであった。
2. **新規の定理（ Conjecture）を動的に生成 **
   - 本研究では、モデル自身が「手頃な難易度」の定理を　**自動生成 **し、それを自分で証明する「自己対戦　(self-play)」を実行する。
   - これにより、証明できる定理を単に増やすだけでなく、徐々に難易度を上げながら「証明者　　(Prover)」の能力を継続的に引き上げられる。
3. **実際の定理証明器 (Lean, Isabelle)を用いた大規模検証 **
   - 数学定理の正式言語化に定評のある　Lean や Isabelle で大規模実験を行い、既存手法より高い成功率・スケーリング性能を達成している。
   - LeanWorkbook (約8.9万定理) に対して従来の倍となる **26.3%**の定理を証明でき、 miniF2F, ProofNet, PutnamBenchなどでもSOTA級の精度を示した。

A Comparison of DeepSeek and Other LLMs DeepSeek とその他の LLMの比較

概要: DeepSeek は、短いテキストを使い、分類や予測を行う。統計学論文の要旨や引用周辺の短文を解析し、その文章が AI によるものか人間によるものかを高精度で見分ける。大量の実験結果を得るため、MadStatAI と CitaStat という 2 つのデータセットを使い、 DeepSeek と他の LLM（Claude、Gemini、GPT、Llama）を比較する。各モデルの分類精度や実行時間、コストを詳細に評価するため、多角的に実験を行う。

技術や手法

本論文では、「 **著者識別 (Authorship Classification)**」と「**引用分類 (Citation Classification)**」という 2 つの具体的タスクを用い、 DeepSeek と他モデルの性能を比較・検証している点が大きな特色です。

4.1 著者識別 (Authorship Classification)

(1) タスク概要

- **短文 (ここでは学術論文のアブストラクト)** を対象に、「人間が書いたもの（ hum）」と「AI が書いたもの（ AI）」、あるいは「人間執筆を AI が編集したもの（ humAI）」のいずれかを判定。

- 比較的シンプルな「 hum vs. AI」か、より微妙な違いを問う「 hum vs. humAI」かの 2 種類の分類を行う。

(2) データセット (MadStatAI)

- **MADStat** は、統計学関連の 83,000 超の論文要旨を含む大規模データベース。

Agentic Verification for Ambiguous Query Disambiguation あいまいなクエリの意味を明確化するためのエージェンティック検証

概要: Retrieval-Augmented Generation（RAG）のあいまいなクエリを解消し、正確で多様な解釈を生成するエージェンティックなフレームワークを使い、高い精度を実現する。 RetrieverとGenerator双方のフィードバックを組み合わせるため、多様化と検証を同時に行う。あいまいな質問を処理するため、 Retrieverで得た文書とLLMの回答可否を活用する。

既存研究の一般的アプローチ

- **Diversify-then-Verify（DtV）**

既存の多くの研究（例：DIVA, 2024など）は、最初にあいまいクエリを多様な書き換え候補（サブクエリ）に分岐し、後段でそれらが正当な解釈かを検証しようとする「 DtV」パイプラインを採用していました。

しかしこの方法では、書き換え候補の段階で不必要・根拠のない解釈が大量発生するリスクがあり、それを後から取り除く検証ステップにも限界があるため、雑音（ノイズ）が蓄積し精度や効率が低下してしまいます。

- **RAC（Retrieval-Augmented Clarification）手法**

Retrieverが返した文書集合に対して一括でサブクエリを書き出す手法も提案されていますが、長文コンテキスト内で多数の文書を一度に処理するため、特にパラメータ数の少ない LLMでは解釈が少なくなったり重複が多かったりといった問題が生じやすいです。

4.2 本研究の新規性・優位点

Locally-Deployed Chain-of-Thought (CoT) Reasoning Model in Chemical Engineering: Starting from 30 Experimental Data ローカル展開型チェーン・オブ・ソート（ CoT）による化学工学の推論モデル： 30件の実験データから

概要: 化学工学における分子性質（例：溶解度）予測を実施。

30件の実験データを CoTを使い精度向上を実現しました。

計的または単一の機械学習手法では捉えきれなかった非線形関係や少数データ問題を、反復的なエラー解析と「再考（ rethinking）」プロセスで補正することで、より堅牢な予測システムを構築しました

2. 先行研究との比較と革新性

従来手法の課題 :

- **実験の手間とコスト： ** 長時間・高コストな実験プロセスに依存し、データ数が限られる。

- **機械学習の限界： ** 少数データでは過学習や予測の不安定性、または解釈性の低さが問題となる。

- **大規模言語モデルの限界： ** テキストマッチングやパターン認識は可能でも、因果関係や深層的な推論が不足。

本研究の革新性 :

- **LLM-CoT手法：** DeepSeek-R1やQwen2といった大規模言語モデルを、ローカル環境で Ollamaなどのツールと連携させ、エラー解析を伴う反復プロセスにより予測精度を向上。

  - 予測結果が規定の誤差範囲（例：誤差 100%以下）に達しない場合、再度「再考」することで精度を改善。

- **ML-LLM-CoT手法：** 事前学習済みの Gaussianモデルで初期予測を行い、 LLMによる補正プロセスを組み合わせることで、特に分子構造が類似している場合に極めて安定した予測を実現。

UltraGen: Extremely Fine-grained Controllable Generation via Attribute Reconstruction and Global Preference Optimization UltraGen：属性再構築とグローバルプリファレンス最適化による極細粒度制御生成

概要：従来の制御生成手法が 3〜5個程度の制約に留まる中、実際のアプリケーションで必要とされる 30個以上、場合によっては 45個近い細かい属性を同時に扱うための新しいアプローチです。

• 概要説明の形式：「 UltraGenは、自然文から抽出した柔軟なソフト属性とプログラム的に得られる厳格なハード属性を用いて、テキストの再生成を行うことで、複数の制約を同時に満たす制御生成を実現する。これにより、位置バイアスや注意の分散といった課題を克服し、従来モデルに比べて高い制約遵守率(CSR)とテキスト品質を達成する。」

4. 先行研究との比較および優位性

- **従来の制御生成手法： ** 主に3〜5個の属性しか扱えず、実世界の複雑な要求（例：旅行プランニングなど）には適用が困難。

- **UltraGenの革新点：** 1. **属性再構築(AR):** ・LLMを用いて、原文からソフト属性（例：文章のトーン、感情、文体）を抽出し、さらにプログラム的に導出されるハード属性（例：文章長、キーワード頻度、構造制約）と統合。 ・テキスト再生成の学習により、モデルが細かい制約を内部に取り込み、自然な文章生成を実現。 2. **グローバルプリファレンス最適化(GPO):** ・大規模な属性プールから有効な属性セットを選定し、直接の好み最適化(DPO)を行う。 ・属性間の相関性や冗長性を考慮した戦略を導入し、属性の位置バイアスや注意散漫問題を軽減。

この結果、UltraGenは従来手法と比較して、極端な属性数が必要な状況下でも制約遵守率(CSR)が大幅に向上し、生成されるテキストの品質も保たれる点が非常に優れていると評価されています。

---

5. 論文で説明されている技術や手法の詳細

5.1 属性再構築(Auto-Reconstruction, AR)

Native Sparse Attention: Hardware-Aligned and Natively Trainable Sparse Attention ネイティブ・スパース・アテンション：ハードウェアに適合した、かつネイティブに訓練可能なスパース・アテンション

概要：長い文脈を扱う際に発生する計算負荷を、従来の全結合型アテンションではなく、スパース・アテンション手法を用いることで大幅に削減することを目的としている。概要は「本手法は、入力シーケンスをブロック単位に分割し、各ブロックの情報を　MLPを使い圧縮することでグローバルな文脈を低コストで把握し、さらに注意スコアに基づいて重要なブロックを選択することで局所情報を保持する。局所パターンはスライディングウィンドウで補完するため、計算資源を効率的に活用しつつ、長文処理の精度と速度の両立を実現する。」という形式で説明できる。

従来のスパース・アテンション手法は主に推論時の　KVキャッシュ削減や固定パターンによる計算削減に留まり、トレーニング時のエンドツーエンドな最適化が困難であった。これに対し、　NSAは以下の点で優れている：

- **ハードウェア最適化 **：Tensor Coreなど最新 GPUの特性に合わせたブロックワイズなメモリアクセスとループスケジューリングを実現し、理論上の計算削減を実際の速度向上に結びつけている。

- **エンドツーエンド学習の実現 **：非微分可能な操作を排除し、動的かつ連続的なトークン選択機構を導入することで、トレーニング中にもスパースパターンを最適化可能とし、全体の学習効率を向上させている。

- **グローバルとローカルの両立 **：圧縮によるグローバルな情報把握と、選択およびスライディングウィンドウによる局所的な情報補完を組み合わせることで、どちらか一方に偏らないバランスの取れたアテンション機構を構築している。

技術や手法

1. **トークン圧縮（ Token Compression）**

    - 入力シーケンスを一定のブロック（例：長さ　l）に分割し、各ブロック内のトークン群を　MLPと位置エンコーディングを用いて1つの圧縮表現に変換する。

    - 数式では、$\tilde{k}_{cmp}(t) = \phi(k_{i+1:i+\square})$　(式7)と表され、これにより計算対象となるトークン数を大幅に削減する。

2. **トークン選択（ Token Selection）**

    - 圧縮トークンから得られる中間の注意スコアを利用し、各ブロックの重要度を算出。

Large Language Diffusion Models 大規模言語拡散モデル

概要: 従来の自己回帰モデル(ARM)に依存する大規模言語モデルのアプローチとは一線を画し、拡散モデルを利用した新しい生成手法「 LLaDA」を提案

テキスト全体に対してランダムにトークンをマスクし、そのマスク状態から同時に全トークンを予測する拡散モデルを使い、従来の逐次生成では実現困難であった逆方向推論や多ターン対話の課題を克服するために、従来の自己回帰的な生成手法が抱える計算コストや左から右への依存性の問題を解決し、双方向的な情報の活用と柔軟な生成を実現する

先行研究との比較

従来の大規模言語モデルは、トークンを逐次生成する自己回帰モデル(ARM)に依存していました。しかし、 ARMは以下の問題点を有していました。

- **逐次生成による計算負荷の増大 **トークンを一つずつ生成するため、長文生成時の計算コストが大きくなる。

- **左から右への生成の制約 **逆向き推論(例:前の行を生成するタスク)において性能が低下する。

一方、LLaDAは、マスク処理を通じた拡散モデルにより、全トークンを同時に予測することでこれらの問題を解決し、指示追従や多ターン対話など、より複雑なタスクに対しても柔軟に対応できる優位性を持っています。

論文で説明している技術・手法

SWE-Lancer: Can Frontier LLMs Earn $1 Million from Real-World Freelance Software Engineering? SWE-Lancer：最先端LLMは実世界のフリーランスソフトウェアエンジニアリングで　100万ドルを稼ぐことができるか？

概要：実世界のフリーランスソフトウェアエンジニアリングタスクを対象に、実際の報酬（総額　　100万ドル）を基準とした新しい評価ベンチマーク「　SWE-Lancer」を提案する。

概要は以下の形式で説明できる：

「本論文は、実世界のフリーランス案件を対象とし、実際の報酬とエンドツーエンドテストを用いて、最新の大規模言語モデル（　　LLM）がソフトウェアエンジニアリングタスクをどの程度解決できるかを定量的に評価するため、タスクの解決率と経済的成果を指標としてマッピングする評価手法を採用する。」

具体的には、個々のバグ修正や機能追加（　IC SWE Tasks）と、複数の提案から最良の解決策を選択する管理タスク（　SWE Manager Tasks）の両方を評価対象とし、各タスクには実際の報酬が設定されている。さらに、各タスクの評価は　Playwrightなどのブラウザ自動化ツールを活用したエンドツーエンドテストにより実施され、現役のソフトウェアエンジニアによる三重の検証を経ている。

先行研究と比較しての優位性

- **実世界性の追求 **

　従来のベンチマークは、ユニットテストやプログラム合成といった限定されたタスクに依存していたが、　　SWE-Lancerは実際にUpworkで提示されたフリーランス案件から収集したタスクを対象としており、現実のソフトウェア開発環境を忠実に再現している。

- **経済的指標の導入 **

　各タスクには実際に支払われた報酬が紐づいており、モデルが解決できたタスクの報酬合計をもってその性能を経済的な価値に直結させる点が新しい。これにより、単なる精度評価ではなく、実際の市場価値に基づく評価が可能となっている。

Optimizing Model Selection for Compound AI Systems 複合AIシステムのためのモデル選択の最適化

概要: 複数のLLMを組み合わせる複合 AIシステムにおいて、各モジュールに対して最適な LLMを選択する問題に焦点を当てています。従来のシステムは、全てのモジュールで同一の LLMを利用するケースが多かったのに対し、本論文では各モジュールごとに異なる LLMを割り当てることで、システム全体の性能を大幅に向上させる手法を提案しています。提案手法「 LLMSelector」は、各モジュールの性能を LLM診断器によって推定し、その結果をもとに逐次的な更新を行うことで、効率的に最適なモデル割り当てを見つけ出します。実験結果では、最適なモデル選択により、従来手法と比較して 5%〜70%の精度向上が報告されています。

先行研究と比べてどこがすごいのか

- **従来手法との違い **
  - 従来はプロンプト最適化やモジュール間の相互作用調整に重点を置き、全モジュールで同一の LLMを使用する手法が主流でした。
  - 本論文では、各モジュールに異なる LLMを割り当てるという新たなアプローチを導入し、各 LLMの得意分野を活かして全体性能を向上させています。
- **技術的革新点 **
  - **LLM診断器の利用 **
    - 各モジュールの出力や最終結果から、 LLM診断器が個々のモジュールの性能を定量的に評価します。これにより、単に全体の正誤だけではなく、各モジュールの寄与度を正確に把握可能となりました。
  - **単調性仮定と最適解への収束 **
    - 各モジュールの性能がモジュール間・モジュール内ともに単調性を持つという仮定のもと、最適な LLMの割り当てが有限の反復で見つかることを理論的に保証しています（定理 4.1）。
  - **効率的な最適化アルゴリズム **
    - 組み合わせ爆発を回避するため、初期のランダム割り当てから出発し、各反復で一つのモジュールのみを更新することで、計算コストを大幅に削減しながら最適解に近づけます。

TOWARDS AUTO-REGRESSIVE NEXT-TOKEN PREDICTION: IN-CONTEXT LEARNING EMERGES FROM GENERALIZATION 自己回帰的次トークン予測に向けて：文脈内学習は一般化から出現する

概要：

従来のi.i.d.（独立同分布）仮定に依拠した解析では捉えきれなかった現実の言語生成プロセスに焦点を当て、自己回帰的次トークン予測（ AR-NTP）の枠組みの下で文脈内学習（ In-Context Learning, ICL）の発現メカニズムを明らかにすることを目的としています。具体的には、事前学習フェーズと ICLフェーズを統一的に扱い、トークン間の依存性を厳密にモデル化することで、シーケンスとトピックの二段階の一般化（Two-level Expectation）を通じた最適化解析を行っています。この理論的枠組みでは、 PAC-Bayesian解析や連続時間確率微分方程式（ SDE）を駆使して、データ依存型・トピック依存型の事前分布を導入し、後部分布との KLダイバージェンスを評価することで、未知のトピックや新規シーケンスに対する予測誤差（人口損失）の上界を導出しています。さらに、シンセティックな言語データセット（ GINC）や実世界の多様な言語データセットを用いた実験により、理論の実証と実用上の意義を検証しています。

先行研究と比べた優れている点

従来の文脈内学習に関する研究は、以下の 2点で制約がありました。

1. **i.i.d.仮定の限界 ** ・多くの研究は、入力と出力が独立同分布であると仮定し、実際の言語生成におけるトークン間の依存性を無視していました。

2. **最適化解釈のみの説明 ** ・ICLの発現メカニズムを暗黙の最適化プロセス（例：単一ステップの勾配降下）として捉えるだけで、なぜ事前学習済みモデルが未知のタスクに対しても高い ICL能力を発揮するのか、その理論的根拠を十分に説明できていませんでした。

本論文はこれらの問題点に対し、以下の革新点を提示しています。

・**AR-NTPパラダイムの採用 **

実際の言語生成プロセスに即して、トークン間の依存関係を反映する自己回帰的生成モデルを採用。これにより、より現実的なモデル解析が可能となりました。

Appendix