論文要約

LLM関連

LLM-Powered Knowledge Graphs for Enterprise Intelligence and Analytics エンタープライズ・インテリジェンスとアナリティクスのための　LLMパワー・ナレッジグラフ

タグ: Knowledge Graph, LLM, RAG

リンク: https://arxiv.org/abs/2503.07993

論文公開年 : 2025

月: Mar

公開日 : 11

概要: LLMを用いてメール、カレンダー、チャット、ドキュメント、ログなど複数のエンタープライズデータソースを統合し、ユーザー活動を中心としたナレッジグラフを構築する枠組みを提案。

エンティティ抽出、関係推論、意味的拡張を自動化することで、高度な検索、推論、分析を可能にする。企業における専門性発見、タスク優先度の決定、意思決定支援などのアプリケーションにおいて、提案システムの有効性を実証するため、実環境での　6か月間のパイロット実験を行い、高精度な推論とユーザー満足度の向上が確認

LLMを活用してさまざまなデータから自動的に情報を取り出し、それらを統合して柔軟で拡張性の高いナレッジグラフを作成するための仕組みづくりに取り組みました

組織の規模を問わず、データを効果的に統合し、柔軟で動的なナレッジグラフを作ることで、業務上のさまざまな課題の解決を助け、意思決定の質を高められる可能性があります。

方法論

企業や組織では、多種多様なデータが日常業務の中で蓄積されていますが、これらを適切に結びつけ、利用しやすい形に整えることが課題となっています。ここでは、研究者らが開発した、分散したデータをひとつのまとまった知識構造（ナレッジグラフ）へと統合するための具体的な方法について説明します。

MDocAgent: A Multi-Modal Multi-Agent Framework for Document Understanding MDocAgent：文書理解のためのマルチモーダル・マルチエージェントフレームワーク

タグ: Agent, Document Question Answering, LLM, Multi-modal, RAG

リンク: https://github.com/aiming-lab/MDocAgent

https://arxiv.org/abs/2503.13964

論文公開年 : 2025

月: Mar

公開日 : 18

概要: 複雑な文書理解における質問応答（ DocQA）の精度向上を目指し、テキストと画像の両方を扱うマルチモーダル・マルチエージェント型フレームワーク「 MDocAgent」を提案する。RAG（検索拡張生成）を用いたテキスト・画像それぞれの情報抽出を行い、 5つの専門エージェント（汎用、重要情報抽出、テキスト処理、画像処理、統合要約）を使って回答を生成する。文書のテキストと視覚情報を横断的に活用することで、複雑な質問に対して高精度な回答を可能にする。 DocQAベンチマーク 5種において、最新手法に比べて平均 12.1%の精度向上を示した。

複数の専門エージェントが文書の「テキスト情報」と「画像情報」を個別に処理し、統合的に理解することを目指した新しい文書理解フレームワークの開発。

従来の文書質問応答（ DocQA）の手法と提案手法を比較し、それぞれの課題と特徴を示した図

背景

LLMs' Opinions Also Matter: Mixture of Opinions Enhances LLM's Mathematical Reasoning

タグ: CoT, LLM

リンク: https://arxiv.org/html/2502.19622v1

論文公開年 : 2025

月: Feb

公開日 : 26

概要: 数学的推論タスクにおける LLM（大規模言語モデル）の性能向上を目指し、弱い（もしくは中規模の） LLM群が生成する多様な「意見」（各々の推論過程を含む）を活用する手法「 Mixture of Opinions（MoO）」を提案しています。具体的には、各トレーニングサンプルに対し、補助的な LLM群からChain-of-Thought（CoT）形式の推論過程と回答を収集し、それを 1つのデータセットとして統合します。その後、メインとなるLLMをこのデータセットを用いて後処理学習（ポストトレーニング）し、最終的な推論精度を向上させるという流れです。

2. 先行研究との比較と本論文の革新点

従来の研究では、以下のような課題がありました。

- **大規模で閉じたモデル依存 **: GPT-4などの大規模モデルに頼るため、モデルの利用環境が限定される。

- **単一視点の推論 **: 1つのモデルによる推論では、入力のわずかな変化で大きく結果が変わるなどの不安定性が指摘されていました。

本論文の革新点は、 **弱いLLMの多様な意見を統合 **することで、各モデルが持つ異なる論理展開を学習データとして活用し、平均約 5%の精度向上を実現した点にあります。これにより、より小規模なモデル環境でも安定かつ高精度な数学的推論が可能になると示唆されています。

LLM Agents Making Agent Tools LLMエージェントがエージェントツールを作成する

タグ: Agent, LLM

リンク: https://arxiv.org/abs/2502.11705

論文公開年 : 2025

月: Feb

公開日 : 17

概要: TOOLMAKERは、論文に付属するコードをインストールし、エラーを解析・修正するしくみを使い、自動でツールを作成する。ツール作成の過程で　　LLMエージェントが外部ライブラリを整備して、生成したコードを実行する。エラーがある場合は再度エージェントがログを読み、問題点を見つけて再実装する。このように閉ループで何度も自己修正を行い、最終的にユーザのタスクを実行できるツールを完成させる。ツールを作るため Dockerなどを用い、依存関係を自動的に導入する。

2. 先行研究と比べてどこがすごいのか

- **自動化の範囲が広い **: 単なるツール使用ではなく、「既存の研究コードを　LLMエージェントが自ら読み、環境を構築して実行可能なツールへ変換する」点で画期的。

- **複雑な依存関係への対応 **: 医療や生命科学分野の研究コードは多種多様なライブラリ・モデルが必要になるが、　Dockerを用いた自動インストールとエラー修正機構を組み合わせることで、事前の人手設定を極力排除。

- **クローズドループのエラー修正 **: エラーが発生するたび、エージェントがログを解析し、不具合を修正する方針を自動で立案しコードを再生成するため、実験的な再試行の負担を人間が被らない。

---

3. 論文で説明している技術や手法

Leveraging Domain Knowledge at Inference Time for LLM Translation: Retrieval versus Generation

タグ: LLM, RAG

リンク: https://doi.org/10.48550/arXiv.2503.05010

論文公開年 : 2025

月: Mar

公開日 : 6

概要: LLMを専門分野の翻訳に適応させるために、外部リソースやモデル自身が生成した専門知識を活用する方法を使い、翻訳精度を改善する。専門用語や翻訳例を示して精度を比較するため、複数の手法を評価する。翻訳精度向上のため、翻訳例を示す手法が最も効果的であることを明らかにする。

従来は専門的な翻訳精度を向上させるために、専門的なデータを追加学習させるか、事前に外部で翻訳例や専門用語集を用意していました。しかし、本研究では、モデル自身が事前学習で得た膨大な知識を活用して翻訳例や専門用語を生成させる方法を新たに導入しています。さらに、「翻訳例」と「専門用語集」の効果を分離して検証したことで、専門的な文体・表現が翻訳品質により重要であるという新しい知見を提示しています。

**専門分野の知識を翻訳に取り入れる方法　**

専門的な分野の翻訳では、単に意味が伝わるだけでなく、その分野特有の言い回しや専門用語を正確に訳すことが求められます。このような専門知識を翻訳に取り入れるためには、「翻訳例」を活かす方法と「専門用語集」活かす方法の二つがあります。

「翻訳例」とは、すでに正しく翻訳された文章のペアのことです。たとえば、医薬品に関するドイツ語の説明文とその正しい英語訳をセットで示します。モデルはこうした実際の翻訳例を参考にして、専門分野の文体や表現方法を具体的に学ぶことができます。

DP-GTR: Differentially Private Prompt Protection via Group Text Rewriting DP-GTR：グループテキスト書き換えを用いた差分プライバシーによるプロンプト保護手法

タグ: LLM, Privacy Protection, Prompt Engineerring

リンク: https://github.com/FatShion-FTD/DP-GTR

https://arxiv.org/abs/2503.04990

論文公開年 : 2025

月: Mar

公開日 : 6

概要: DP-GTRは、差分プライバシー（ Differential Privacy; DP）を用いたグループテキスト書き換え技術を使い、プロンプトに含まれる個人情報の漏洩を防止する新たなフレームワーク。

既存手法は文書レベルの書き換えに限定されており、細かな単語レベルのプライバシー保護には不十分。

このため、DP-GTRは文書レベルと単語レベルの両方の情報を統合し、局所差分プライバシー (Local Differential Privacy; LDP)とグローバル DPの利点を組み合わせてプロンプトのプライバシー保護と実用性を両立している。具体的には、次の 3段階の処理を行う:

1. グループテキスト書き換えによる複数のプロンプトパラフレーズ生成

2. 単語の出現頻度分析に基づいたプライバシーと実用性の細粒度制御（重要な単語を特定して選択的に保護）

3. 識別されたプライベート単語を抑制しつつ、最適なパラフレーズをプロンプトに用いて LLMに入力し、最終的なプライバシー保護済みプロンプトを生成

CommonSense QAやDocVQAといった質問応答タスクにおける評価により、従来手法より優れたプライバシーと実用性のバランスを示した。

LLMのオンライン利用のために企業や個人がプライバシーを保護するために

これまでにも差分プライバシーという技術が情報漏洩対策として研究されてきました。差分プライバシーとは、データを加工する際に、元のデータに関する個々の情報が特定されないよう統計的にノイズを加

LLMs Can Generate a Better Answer by Aggregating Their Own Responses

タグ: LLM, Self-Aggregation

リンク: https://doi.org/10.48550/arXiv.2503.04104

論文公開年 : 2025

月: Mar

公開日 : 6

概要: 一般的には一度の質問に一つの回答を生成するといった使い方がされています。しかしそれでは複雑な問題に対して不十分な場合があります。

そこで今回は、LLMが複数の回答を生成し、それらを活用してより質の高い回答を生み出す方法と、その効果について既存の手法の中でもう一つ有望視されてきたのが「自己整合性」という手法です。これは、LLMが生成した複数の回答から多数決で最も多い回答を選ぶことで　https://ai-data-base.com/archives/25930を向上させる方法です。しかし回答が明確に定まらない自由記述型の問題には適用が難しく、汎用性に欠けるという問題（限界）があります。

そこで外部からのフィードバックや多数決に頼らずに、　LLM自身が生成する多様な回答を活用し、それらを統合することで新たな回答を導き出す　Generative Self-Aggregation（GSA）を提案しました

「Generative Self-Aggregation（GSA）」

通常、言語モデルは質問に対して一度に一つの回答を生成しますが、それでは複雑な問題に対して誤った答えを出すことがあります。そこで今回、「一度に一つ」ではなく、「複数の回答」をモデル自身が生成し、その多様な回答群から新たな回答を導き出すことで、品質の高い回答を得る手法が考案されました。

生成された回答をモデル自身が評価したり選択したりする「識別的な判断」を一切行わず、モデルが本来得意としている「文脈を参照して新しい文章を生成する能力」だけを用いて回答品質を改善しする方法です。言い換えると、モデルが苦手な自己評価を避け、得意な生成能力を最大限に活用しようというモデです。

Chatbots for Data Collection in Surveys: A Comparison of Four Theory-Based Interview Probes アンケートにおけるデータ収集のためのチャットボット：　4つの理論に基づくインタビュープローブの比較

タグ: Interview, LLM

リンク: https://arxiv.org/abs/2503.08582

論文公開年 : 2025

月: Mar

公開日 : 11

概要: チャットボットがアンケート内での自由回答を深めるために使えるかを調べるため、　4種類の理論に基づくインタビュープローブを使い、異なる　HCI研究段階において応答の質とユーザー体験を評価する。LLMを使ったチャットボットを用い、探索・要件定義・評価という　3つの場面で、記述的、個人的、明確化、説明的な　4種のプローブを評価する。効果を測るため、　64名の参加者にチャットボットベースのアンケートを実施し、Griceの協調原理に基づいた応答品質指標とユーザー体験を計測

チャットボットの質問方法によって回答の質や参加者の体験に差が生じることがわかりました。

中でも「個別的プローブ（ Idiographic）」を使った質問は、回答者が自分の経験を具体的かつ自然に話せるため効果的でした。一方、「説明的プローブ（　Explanatory）」のように理由を繰り返し問う質問では、回答者が疲れたり会話が停滞したりする可能性があることがわかりました

これまでにインタビュー研究で使われてきた質問テクニックを参考に、チャットボットでのアンケートに最適な質問方法を探る研究に取り組みました。参加者が楽しみながらも、深く考えて答えられるような、新しいアンケート手法の開発を目指したのです。

**課題の整理 **

**オンラインアンケートの課題 **

API Agents vs. GUI Agents: Divergence and Convergence API vs GUIでのエージェントの使い分け

タグ: API, Agent, GUI, LLM

リンク: https://arxiv.org/abs/2503.11069

論文公開年 : 2025

月: Mar

公開日 : 14

概要: LLMは自然言語入力を用いた自動化タスクを実行するソフトウェアエージェントとして進化し、 APIベースのエージェントは堅牢な自動化とプログラムとの統合で台頭し、 GUIベースのエージェントは人間のようなインターフェース操作を可能にする。両者はタスク自動化を目的とするが、アーキテクチャ、開発手法、ユーザーとの対話方式が異なるため本論文では両者の違いと共通点を系統的に分析する。両者の長所を活かすハイブリッドアプローチも提案し、現実の用途に応じた選定指針を提供する。

LLMは具体的な指示を実行可能な行動へと直接変換するソフトウェアエージェントとして活用されるようになっています。

そんないわゆる LLMエージェントには、これまで大きく二つのタイプが登場しています。

一つはAPIベースのエージェントです。これは、プログラムが提供する明確なインターフェース(API)を通じて外部のシステムやサービスと連携し、自動で処理を進めます。 APIエージェントは効率的で信頼性が高く、MicrosoftのCopilotのように、すでに広く産業界でも採用されています。

もう一つは、最近注目を集めている GUIベースのエージェントです。こちらは画面上の視覚的な情報を人間のように見て理解し、マウスクリックやキーボード入力など、実際のユーザー操作を模倣する形でソフトウェアとやり取りをします。 GUI(グラフィカルユーザーインターフェース)上での作業を自動化できるため、 APIが用意されていない環境でも柔軟に活用できます。

これら二つのエージェントにはそれぞれ長所と短所があります。 APIエージェントは高速で信頼性が高い反面、利用可能な APIがなければ機能できません。対して、GUIエージェントは、多様な画面に対応できる

Beyond Prompt Content: Enhancing LLM Performance via Content-Format Integrated Prompt Optimization プロンプト内容を超えて：コンテンツとフォーマットを統合した最適化による　LLM性能の向上

タグ: LLM, Prompt Engineerring

リンク: https://doi.org/10.48550/arXiv.2502.04295

https://github.com/HenryLau7/CFPO

論文公開年 : 2025

月: Feb

公開日 : 6

概要:

CFPOはプロンプトの内容とフォーマットを統合的に最適化する手法であり、性能向上のために自然言語変異による内容の改良とフォーマット探索による最適構造の決定を行う。　LLMのフォーマット感度に対応するため動的評価と生成に基づくフォーマット選定戦略を採用し、最適なプロンプトを得るために反復的な最適化を実施する。

LLMが実際の業務やタスクで真に効果を発揮するかどうかは、モデルへの「問いかけ方」、つまりプロンプトの設計に大きく左右されます。良いプロンプトはモデルの性能を引き出し、ユーザーが望む出力を効果的に得るための鍵となりますが、一方で、プロンプトを人手で最適化することには課題があります。

その課題とは、LLMがプロンプトの微妙な変化に非常に敏感であるという性質に起因しています。たとえば、同じ内容の質問であっても、言葉遣いや文の構成をわずかに変えるだけで、　LLMが返す回答が大きく異なることがあります。

また、モデルごとに「好むプロンプト形式」が異なるという性質があり、あるモデルで高性能なプロンプトが別のモデルではうまく機能しないこともあります。そのため、人が事前に万能なプロンプトの形式を決めることは難しいです。

これまでの研究では、そのような課題を解決するために、自動的にプロンプトを最適化する手法が提案されてきました。しかし、多くはプロンプトの「内容」だけに焦点を当てており、プロンプトなどのような構造

An LLM’s Attempts to Adapt to Diverse Software Engineers’ Problem-Solving Styles: More Inclusive & Equitable? LLMは多様なソフトウェアエンジニアの問題解決スタイルに適応できるか？ : より包摂的で公平な支援を目指して

タグ: LLM, Prompt Engineerring

リンク: https://arxiv.org/abs/2503.11018

論文公開年 : 2025

月: Mar

公開日 : 14

概要: LMがソフトウェアエンジニアの多様な問題解決スタイルに適応してコードの説明を行うことで、理解を支援できるかを調べる。　GenderMagの5種類の問題解決スタイルを使い、それぞれに対応する説明を生成する。効果を調べるため、 53名のCOBOL未経験エンジニアに、同一コードに対してスタイルに合わせた・合わせない・非適応の　3種類のLLM出力を提示し、評価を収集

多様なSEが、LLMによるコード説明を受け取る際に、どのような支援が公平で包括的であるかを調査。

同じようにコードを理解したり、問題を解決したりするわけではありません。例えば、新しいコードに触れるとき、じっくりマニュアルを読んで理解を深める人もいれば、実際にコードを動かしながら直感的に学ぶ人もいます。

このような個人差を整理し、デザインに生かす方法として「　GenderMag（ジェンダーマグ）」という手法があります。 GenderMagは、特に性別と関連が深い 5つの「問題解決スタイル」を使って、製品やサービスが偏りなく、多様な人に使いやすいかどうかを評価します。その　5つとは、学び方（学習スタイル）、自信（自己効力感）、リスクを好むかどうか（リスク態度）、情報収集の仕方（情報処理スタイル）、技術を使う理由（技術への動機付け）です。

本研究はその GenderMagに着目しています。その理由は、この手法は「性別」という直接的な属性だけでなく、「個人が問題を解決するときの考え方や行動」に注目することに応用できるためです。

Appendix