論文要約

LLM関連

LONG2RAG: Evaluating Long-Context & Long-Form Retrieval-Augmented Generation with Key Point Recall LONG2RAG: 長文コンテキストおよび長文形式の検索強化生成の評価とキーポイントリコールによる評価

概要: LONG2RAGは、長文コンテキストでの LLMのRAG性能を評価するためのベンチマークです。 280の質問に5つの関連文書を設定し、検索された文書から抽出されたキーポイントをどれだけ含んでいるかを測定するKPRで評価。GPT-4oが最高スコアの 0.579を記録

RAG評価で主に長文コンテキストに対応する LLM評価ベンチマーク LONG2RAGを提案。280の質問が10の領域で設定され各質問に対して 5つの関連文書を設定。評価指標には KPR（Key Point Recall）を設定し検索された文書から抽出されたキーポイントをどれだけ含んでいるかを測定する方法（各スコアは 0から1の範囲で、高いほど良い性能を示します）を使用。質問は 8つのカテゴリ（事実、説明、比較、主観、因果関係、仮定、予測、方法論）に分類して評価、 GPT-4oのKPRは 0.579、Claude-3-SonnetのKPRは 0.477、Qwen2-72B（オープンソースモデルの大規模版） : KPRは、0.449、Phi-3-mini-128K: KPRは 0.434と商用モデルである GPT-4oが最も優れた結果を示しました。

また、KPRは長文生成を好む傾向もあるため、生成の質と長さのバランスが重要であることもわかっています

技術や手法

- **LONG2RAGベンチマーク **: 280の質問を使用し、各質問に対して平均 2444語の5つの検索文書が関連付けられています。これにより、モデルが長文の検索情報を取り込む能力を評価します。

- **キーポイントリコール（ KPR）**: 検索された文書から抽出されたキーポイントが生成された回答にどの程度含まれているかを評価する手法です。この評価を通じて、モデルが検索情報を活用しているかどうかを測定します。

- **データセットの構築方法 **: 自動パイプラインを用いて質問を生成し、関連する文書を検索してキーポイントを抽出。その後、 LLMと人間の協力によりキーとなるポイントの検証を行い、データセットを構築しました。

評価手法とパフォーマンス指標

論文では、LONG2RAGベンチマークを用いて 9つの最新の LLM（大規模言語モデル）を評価しました。評価に用いられた指標は以下の通りです。

VERBOSITY ≠ VERACITY: DEMYSTIFY VERBOSITY COMPENSATION BEHAVIOR OF LARGE LANGUAGE MODELS 冗長性 ≠ 真実性: 大規模言語モデルの冗長性補償行動の解明

概要: LLMは応答に自信がないほど冗長な回答をする傾向冗長性補償（ Verbosity Compensation, VC）があり、対応方法として

冗長性補償の定義と定量化 : 冗長性補償行動は、本来不要な冗長な情報を過剰に生成する行動を指します。この行動を検出するための「冗長性検出器」を提案し、応答が情報の圧縮なしに保持できるかを評価しています。

パフォーマンスと冗長性の関係の分析 : 冗長な応答と簡潔な応答のパフォーマンスを比較する評価指標を導入し、冗長な応答がしばしばパフォーマンス低下を招くことを示しています。

冗長性と不確実性の関係の分析 : 冗長な応答はモデルの不確実性の高さと関連しており、データセットを用いた実験によりこの関連性を明らかにしています。

冗長性軽減のためのカスケードアルゴリズム : 冗長な応答を強いモデルによる応答で置き換える「カスケードアルゴリズム」により、冗長性を効果的に軽減しています。

詳細説明

1. **冗長性補償の定義と定量化 **

**冗長性補償行動 **とは、LLMが確信がない際に余計な情報を多く生成する行動のことです。この行動はしばしば無駄なトークンの生成を伴い、情報をユーザーに効果的に伝達する上で障害となります。冗長性補償行動は、重要な情報にたどり着くまでに時間がかかることでユーザーの混乱を招き、生成にかかるリソースを無駄にする原因ともなります。

研究者らはこの冗長性補償行動を検出するために「冗長性検出器（ Verbosity Detector）」を開発しました。これは、生成された応答が冗長でないかどうかを判断し、圧縮しても情報が失われない場合にはその応答を冗長と見なします。具体的には、与えられた応答が簡潔な形で同じ内容を保持できるかを基準に評価されます。検出器は、入力されたデータと生成された応答を比較し、冗長であると判断された場合は「1」、そうでなければ「 0」を出力します。この仕組みにより、冗長性補償の頻度を定量的に測定することが可能となりました。

2. **パフォーマンスと冗長性の関係の分析 **

Cross-Domain Recommendation Meets Large Language Models クロスドメイン推薦と大規模言語モデルの出会い

概要: LLMとクロスドメイン推薦（ CDR）を使い、プロンプト設計を情報の明確化とタスク定義を工夫し、データ不足のシナリオに対してよい結果に

- **新しいプロンプト設計 **: CDRに特化した 2種類のプロンプト（ターゲットドメイン行動の注入あり /なし）を設計し、それにより LLMsが異なるドメイン間の関係を学習できることを確認しました。

- **LLMsの適応力 **: LLMsが異なるドメインペアでのランキングと評価予測タスクにおいて、既存の最先端 CDRモデルを上回るパフォーマンスを示しました。

使用している技術や手法

1. **プロンプト設計 **:

   - **ターゲットドメイン行動注入プロンプト **: ソースおよびターゲットドメインの両方のユーザーインタラクション履歴を含むプロンプトを使用して、ウォームスタートシナリオをシミュレートします。これにより、LLMsがターゲットドメインに対するユーザーの嗜好をより効果的に学習できることが示されています。

   - **ターゲットドメイン行動注入なしプロンプト **: ソースドメインのみのユーザーインタラクション履歴を含むプロンプトを使用し、コールドスタートシナリオをシミュレートします。このプロンプトは、 LLMsがターゲットドメインのデータなしでどの程度適応できるかを評価するためのものです。

2. **プロンプトの重要な要素 **:

   - **LLMsの役割定義 **: 「クロスドメイン推薦システムとして振る舞う」という明確な役割を設定し、 LLMの推論フレームワークを構築します。

   - **情報の明確な分離 **: 入力データとタスク定義を明確に分け、各ドメインの項目とユーザーの評価をリスト化します。

   - **タスクの明確な定義 **: 推薦タスクの具体的な説明を行い、出力形式を指定することで一貫性を持たせます。

3. **評価指標 **:

   - **ランキングタスク **: 各ユーザーのターゲットドメインにおけるアイテムのランキングを実施し、 NDCGとMRRを使用して結果を評価します。

   - **評価予測タスク **: 可能性のある評価をテキストラベルとして与え、それを数値にマッピングする形で MAEおよびRMSEを用いて評価を行います。

AIDetx: a compression-based method for identification of machine-learning generated text AIDetx: 機械学習で生成されたテキストを識別するための圧縮ベースの手法

概要: AIDetxは圧縮技術を用いて AI生成テキストを検出する手法で ,人間とAIそれぞれの圧縮モデルを作成し、圧縮率に基づき分類し、高精度に

技術・手法

AIDetxは、情報理論の一部であるデータ圧縮技術を用いてテキストを分類する手法である。主に以下の手順で動作する:

1. **モデル構築 **: 人間が書いたテキストと AI生成テキスト、それぞれに対し圧縮モデルを構築する。この圧縮プロセスは、分類器の「トレーニング」に相当し、各クラスを代表する文書を圧縮することで、そのクラスの統計的な特徴を学習する。

2. **分類プロセス **:

   - 新しいテキストが与えられると、人間が書いたテキストと AI生成テキストの両方に対応する圧縮モデルを使ってそのテキストを圧縮する。

   - 圧縮比が高い方のモデルにテキストを分類する。

3. **有限コンテキストモデル (FCM)**: 圧縮モデルとして、マルコフモデルの一種である有限コンテキストモデル(FCM)を使用する。FCMは、コンテキストの深さに基づき次に来るシンボルの確率を予測する確率モデルであり、これにより効率的な圧縮が可能となる。

   - 圧縮に必要なビット数は次の式で表される:

   $$i=1\sum n-log2P(xi|xi-1,xi-2,...,xi-k)$$

   ここで、xiはテキストの位置 iにあるシンボルであり、 nはテキストの長さを示す。

$$\sum_{i=1}^{n} -\log_2 P(x_i | x_{i-1}, x_{i-2}, \ldots, x_{i-k})$$

Multi-Facet Blending for Faceted Query-by-Example Retrieval 多面的クエリによるサンプル検索のための多面ブレンディング

概要: FaBleは文書をファセット（背景、手法、結果など）に分解し、ファセットごとに類似・非類似ペアを生成。この手法により、特定のファセットに基づく検索性能が向上

技術や手法

1. ファセット分解 (Facet Decomposition)

まず、文書を「ファセット」（特定の側面）に分解します。ファセットとは、文書内の特定の要素（例：背景、手法、結果）を指します。 FaBleは公開されている LLM（LLaMA2-13Bモデル）を用いて、文書をこれらのファセットに分解するプロンプトを使用し、それぞれのファセットの要約を生成します。分解のプロセスはゼロショットプロンプトを使って行われます。このゼロショットプロンプトにより、モデルが指定されたファセットについての要約を生成し、その要約を通して文書の構成をモジュール化します。

**プロンプト例 **:

- 文書のファセットを特定して分解するためのプロンプトは以下のような形式です： `{facet}`の部分には「背景」「手法」「結果」などのファセット名が挿入され、モデルはその部分を要約します。この要約が、次のファセット生成プロセスの基準となります。

```csharp
csharp
コードをコピーする
This is the paper abstract: {document}
```

Long Context RAG Performance of Large Language Models 長いコンテキストにおける RAG性能の大規模言語モデル

概要: 20の主要なLLMを使用して、コンテキスト長と RAG性能を検証

多くのモデルは 16k-32kトークン付近でピークを迎え、その後性能が低下するものの、 o1、GPT-4o、Claude 3.5などの商用モデルは 64k以上のコンテキスト長でも一貫して高く 100kトークンまでは性能の向上を確認

検証結果

1. 長い文脈の利用が RAGの性能を一様に向上させるわけではない

- **長い文脈が全てのモデルに対して有益ではない **長いコンテキストを使用した RAGの性能について、全てのモデルで一様に向上するわけではないことが明らかになりました。多くのモデルでは、文脈長を増やすと最初は性能が向上するものの、あるポイント（一般的に 16k〜32kトークン付近）で頭打ちとなり、さらに文脈長を増やすと逆に性能が低下することが観察されています。例えば、 Llama 3.1 405Bモデルでは、32kトークンを超えると性能が徐々に低下することが確認されました。

- **一部の最新商用モデルのみが高いコンテキスト長で一貫した性能を発揮 **一方で、特に性能の良いモデル、例えば OpenAIのo1シリーズや Claude 3.5、Gemini 1.5 Proといった最新の商用モデルは、128,000トークンやさらに長いコンテキストに対しても比較的高い性能を維持しました。これらのモデルは、特に最大 100kトークン程度までは一貫して性能が向上する傾向にありました。

2. モデルの失敗モードとその特定

- **モデルによって異なる失敗の傾向 **各モデルは長いコンテキストを使用する際に異なる失敗モードを示しました。この研究では、次のような失敗パターンが特定されています。

  - **誤回答**: 一部のモデルは長い文脈を扱う際に間違った回答を出すことが多くなります。これは、適切な文脈情報を選択できずに誤った文脈から回答を生成してしまうためです。

  - **指示の無視**: Claude 3 Sonnetのようなモデルでは、文脈が長くなるにつれて指示を正確に理解せずに無視したり、正しく実行しないケースが増加しました。このケースでは、指示に従わずに要約を行ったり、回答を拒否したりするパターンが確認されています。

Leveraging Large Language Models to Generate Course-specific Semantically Annotated Learning Objects 大規模言語モデルを活用したコース固有の意味的注釈付き学習オブジェクトの生成

概要: 自動採点可能なクイズ生成に LLMのRAGの利用を調査

結果、構造的な注釈の生成は効果的であった一方で、関係的注釈には限界があり、生成された問題の品質は大学レベルのコンピュータサイエンス教育的基準を満たすには不十分

1. **ターゲットの具体性 **: 以前の研究では、記憶に関する問題の生成が主流でしたが、本研究では「理解」や「適用」を対象とした問題の生成を目指しています。

2. **RAG技術の利用 **: 一般的なプロンプトエンジニアリングに依存せず、リトリーバル強化生成を活用してコース固有の文脈を加味しています。

3. **意味的注釈の導入 **: 問題の内容と学習者モデルをリンクするための意味的注釈の生成を目指している点で、従来の AQG（自動問題生成）研究を超えています。

技術や手法

1. **リトリーバル強化生成（ RAG）**:
   - 外部知識ベースや学習資料をクエリし、その結果をモデルの入力プロンプトに追加することで、文脈に合ったクイズ問題を生成。
   - コース資料（講義ノート、スライドなど）を活用し、トピックに関連するテキスト断片を抽出。
2. **意味的注釈の生成 **:
   - 学習オブジェクトに対して、概念、認知次元（ Bloomの分類に基づく）、事前知識を示す注釈を付与。
   - STEXパッケージを使用して、注釈を LaTeXコード内で表現。

Retrieval-Augmented Machine Translation with Unstructured Knowledge 非構造化知識を用いた検索強化型機械翻訳

概要: 非構造化文書を活用した検索強化型機械翻訳（ RAGtrans）を提案

GPT-4oや人間の翻訳者によって収集された Wikipediaのリード文と関連文書を使用した 79,000件の多言語形式の翻訳サンプル非構造化データセット RAGtransを作成

マルチタスクトレーニングに基づき、 LLMが多言語の文書を利用して翻訳する方法を学習。この方法により、 BLEUスコアが1.58〜3.09、COMETスコアが1.00〜2.03向上。

独自性:

1. **非構造化知識の活用 **:
   - これまでの研究は、構造化された対訳文や知識グラフに依存していたのに対し、本研究は非構造化文書を活用。
   - 多言語文書間の不完全な整合性を考慮し、異なる言語間での情報補完を実現。
2. **データセット「 RAGtrans」**:
   - Wikipediaのリード文と関連文書を使用。
   - 多言語対応を実現し、中国語、ドイツ語、フランス語、チェコ語も含む。

技術や手法:

1. **データセット作成（ RAGtrans）**:
   - Wikipediaのリード段落を翻訳のソースとして選択。

Evaluating and Aligning CodeLLMs on Human Preference CodeLLMsの評価と人間の好みに基づく調整

概要: コード生成における人間の好みへの適合性を測定するための新しいベンチマーク　CodeArenaを提案。SynCode-Instructを使用した大規模データセットとともに、 40以上のLLMを評価した結果、クローズドモデルが依然として高い結果

技術や手法

1. **実験設定 **
   - **対象モデル :** 7B〜72Bパラメータを持つ 23のLLMを評価。これには、一般的な LLM、コード専用の LLM、オープンソースモデル、クローズドソースモデルが含まれる。
   - **データセット :** 大規模合成コーパス「 SynCode-Instruct」は、19Bトークンの合成データと 1Bトークンの高品質データで構成。
2. **評価ベンチマーク **
   - **EvalPlus:** HumanEvalとMBPPの拡張版であり、基本および追加のテストケースを使用してコード生成能力を評価。
   - **MultiPL-E:** Python、Java、C++、Javascript、Typescriptなどの多言語コード生成能力をテスト。
   - **CodeArena:** 実世界の Q&Aに基づき、アルゴリズムではなく人間の好みに基づく評価を重視。質問ごとに　2回のスコアリングを行い、勝率と引き分け率を算出。
3. **評価指標 **
   - **Pass@k:** モデルが生成したコードの正確性をテストケースを用いて検証。　EvalPlusとMultiPL-Eで使用。
   - **LLMによる判断 :** GPT-4oをジャッジとして、2つの回答を比較し、どちらが優れているかを判断。　CodeArenaでは勝率と引き分け率をレポート。
4. **実装詳細 **
   - **モデル微調整 :** Qwen2.5-Coder-32BをSynCode-Instructで微調整。256台のNVIDIA A100-80GB GPUを使用。
   - **学習設定 :** Adamオプティマイザを使用し、学習率は 8×10^-5で100ステップのウォームアップ後、コサイン減衰スケジューラを採用。

Generating Knowledge Graphs from Large Language Models: A Comparative Study of GPT-4, LLaMA 2, and BERT 大規模言語モデルからの知識グラフ生成： GPT-4、LLaMA 2、BERTの比較研究

概要: LLMを使用して知識グラフ (KGs)を生成するために、Wikipediaからの抜粋を前処理（トークナイズ、クレンジング）し、 GPT-4、LLaMA 2、BERTでエンティティと関係を抽出してグラフ化。基準グラフと比較し、精度、再現率、F1スコア、編集距離、意味的類似度で評価

1. **データ選定 (Data Selection)**

KGを生成するための適切なデータを選びます。

- **選定基準 **: 技術的に多様で、十分な情報が含まれるが、計算負荷を抑える小規模なデータセット。
- **今回のデータ **: Wikipediaの「Cプログラミング言語」ページからの抜粋。
    - 特徴、特性、歴史的背景などの説明が含まれ、エンティティと関係をテストするのに適しています。

---

2. **データ前処理 (Data Preprocessing)**

未構造化データをモデルが処理可能な形式に整えます。

手順:

AutoDCWorkflow: LLM-based Data Cleaning Workflow Auto-Generation and Benchmark AutoDCWorkflow: LLMベースのデータクリーニングワークフロー自動生成とベンチマーク

概要: LLMを使用したデータクリーニングワークフロー　AutoDCWorkflowを提案

目的変数の選定、各列の品質レポートの作成

レポートに基づいた適切なクリーニング操作の選択を　OpenRefineのAPIを使用して実行 Llama 3よりも良い結果が得られました

技術や手法

本論文で説明されている「 AutoDCWorkflow」は、大規模言語モデル（ LLMs）を利用してデータクリーニングを自動化するためのパイプラインです。その技術や手法について順を追って詳細に説明します。

**1. AutoDCWorkflowのアーキテクチャ **

AutoDCWorkflowは、以下の 2つの主要なセッションから構成されています:

1. **LLMエージェントによるプランニングセッション　**

2. **OpenRefine APIを使用したデータクリーニング実行セッション　**

LLMs as Research Tools: A Large Scale Survey of Researchers' Usage and Perceptions 研究ツールとしての LLM: 研究者の利用状況と認識に関する大規模調査

概要: 研究者 816名を対象に LLMの実際の活用状況と、その影響について調査　研究者の 81%が情報収集や文章の編集のために LLMを使用。 非白人は有用性を感じ頻繁に使用し非英語話者は編集で使用、特に若いほど利点を評価する一方、高年次の研究者や女性は倫理的懸念を抱く傾向になった

技術や手法

調査設計

1. **調査項目 **:

   - LLMの使用頻度と目的を 6カテゴリ（例 : 情報収集、データ生成）で分類。

   - 回答者の背景情報（人種、ジェンダー、経歴）を収集。

   - 倫理的懸念や利益認識に関する自由記述も含む。

2. **データ分析 **:

   - リッカート尺度（ 1-5）を用いた評価。

   - **線形混合モデル **: 個人ごとのバイアスを除外し、背景要因と LLM利用パターンの関連を解析。

   - **質的分析 **: 自由記述データをテーマごとに分類。

主な技術成果

Densing Law of LLMs LLMの密度化法則

概要: LLMのコストやリソースを考慮した「能力密度」を評価指標として提案

モデルの有効なパラメータ数を実際のパラメータ数で割ることで算出し、単なる性能に加えてコストやリソースを呼応慮した効率性の評価ができるようにし、これまでのモデルは約 3.3か月ごとに能力密度が倍増していることが示されています

1. **Scaling Lawとの関係性 :** 従来のScaling Lawはモデルの性能がパラメータ数やデータ量の増加により向上することを示していたが、本研究は効率性を重視した新たな視点を提供。

2. **圧縮技術の評価 :** 既存のプルーニングや蒸留などのモデル圧縮技術では、必ずしも能力密度が向上しないことを示唆。

技術や手法

1. **能力密度の定義と計算方法 :**

- 有効パラメータ数: 基準モデルが同等の性能を達成するのに必要なパラメータ数。

- 密度計算式 :

ρ(M)=NMN^(SM)N^(SM): 有効パラメータ数、 NM: 実際のパラメータ数。

ρ(M)=N^(SM)NM\rho(M) = \frac{\hat{N}(S_M)}{N_M}

C3oT: Generating Shorter Chain-of-Thought without Compromising Effectiveness C3oT: 効果を損なうことなく短い Chain-of-Thoughtの生成

概要: Chain-of-Thought（CoT）は生成結果が長すぎ、短くすると結果が悪くなるため CoTを圧縮して短く長い CoTと短い CoTを同時に学習し短い CoTを生成する C3oTフレームワークを提案

CoTの長さを最大 50%以上削減しながら性能を維持できることを確認

技術や手法

問題設定

- **データセット構成 **: `(xi, r_long, y)`で表され、`xi`は指示文、`r_long`は長いCoT、`y`は解答。
- **目標**: 圧縮後の短い CoT (`r_short`) でも解答の正確性を維持するモデルを学習。

C3oTフレームワークの構成

1. **圧縮器（Compressor）**:
    - GPT-4を使用して長い CoTを圧縮し、冗長な部分を削減。
2. **条件付き学習（ Conditioned Training）**:
    - 長いCoTと短いCoTを条件付きトークンで区別し、それらの関係を学習。
3. **条件付き推論（ Conditioned Inference）**:

C3oT: Generating Shorter Chain-of-Thought without Compromising Effectiveness C3oT: 効果を損なうことなく短い Chain-of-Thoughtの生成

概要: Chain-of-Thought（CoT）は生成結果が長すぎ、短くすると結果が悪くなるため CoTを圧縮して短く長い CoTと短い CoTを同時に学習し短い CoTを生成する C3oTフレームワークを提案

CoTの長さを最大 50%以上削減しながら性能を維持できることを確認

技術や手法

問題設定

- **データセット構成 **: `(xi, r_long, y)`で表され、`xi`は指示文、`r_long`は長いCoT、`y`は解答。
- **目標**: 圧縮後の短い CoT (`r_short`) でも解答の正確性を維持するモデルを学習。

C3oTフレームワークの構成

1. **圧縮器（Compressor）**:
    - GPT-4を使用して長い CoTを圧縮し、冗長な部分を削減。
2. **条件付き学習（ Conditioned Training）**:
    - 長いCoTと短いCoTを条件付きトークンで区別し、それらの関係を学習。
3. **条件付き推論（ Conditioned Inference）**:

Apollo: An Exploration of Video Understanding in Large Multimodal Models Apollo: 大規模マルチモーダルモデルにおける動画理解の探究

概要: Apolloは、動画理解のための LLM。この設計の過程でスケーリングの一貫性という小規模モデルで得られた知見が　LLMでも有効である性質を発見

Apolloを設計中に学習時に fps（フレーム毎秒）サンプリングが有効であり適切なエンコーダを選択しトークン処理をっ最適化することで設計の最適化ができることがわかりました

**技術・手法の詳細 **

1. **スケーリングの一貫性 **

小規模モデル（ ~2-4Bパラメータ）での設計決定が大規模モデル（ 7B以上）にも転移する現象を発見。これにより、少ない計算資源で効率的に設計を検証可能に。

2. **動画サンプリング **

- fpsサンプリングが従来の均一サンプリングより性能を向上。

- トークン数と fpsのトレードオフを最適化することで、長短動画の双方に対応。

3. **エンコーダ選択 **

- SigLIP-SO400Mが最適な単体エンコーダ。

- InternVideo2との組み合わせでさらに精度向上。

4. **トークンリサンプリング **

- Perceiver Resamplerを利用することで、情報損失を最小化。

Nano-ESG: Extracting Corporate Sustainability Information from News Articles Nano-ESG: ニュース記事から企業の持続可能性情報を抽出する

概要: Nano-ESGは、企業の持続可能性(ESG)情報をニュースから抽出するアプローチ　約84万から重要な記事を LLM(特にGPT-4o)で要約、ESGの感情、環境、社会、ガバナンスのどれに関するかを抽出、要約や分類の正確性を評価した結果は感情分類の正確性　79.9%、ESG要素分類 78.5%でした

独自性:

1. **ESGデータの収集方法 :**

　従来のESGスコアリングは外部の評価機関に依存しており、再現性が難しいとされてきました。一方、この研究はニュース記事を情報源として活用し、リアルタイムでの　ESGイベント追跡を可能にしています。

2. **データの詳細度 :**

　公開されるデータセットには、記事の要約や　ESG感情、関連要素が含まれており、タイムスタンプ付きで時系列分析が可能です。

3. **技術的アプローチ :**

　最新のNLP技術やLLM(特にGPT-4o)を使用し、記事の関連性、要約、感情、　ESG要素を高精度で抽出しています。

LongBench v2: Towards Deeper Understanding and Reasoning on Realistic Long-context Multitasks LongBench v2: 実際の長文マルチタスクにおける深い理解と推論を目指して

概要: LongBench v2は、長文のマルチタスクを評価するためのベンチマークで以下の　6つの主要タスクカテゴリーに分類された選択問題があります

単一文書 QA

複数文書 QA

長いコンテキスト学習

長い対話履歴の理解

コードリポジトリの理解

長い構造化データの理解

これは人間の専門家が 15分以内に回答した場合でも平均 53.7%の正答率なのに対して最良だった o1-preview では57.7%の正答率を達成。しかし、現在の LLMが長い文脈での深い理解や推論においてまだ課題を抱えていることを強調しており、推論能力をさらに強化することが今後の重要な方向性であるとしています

The Only Way is Ethics: A Guide to Ethical Research with Large Language Models 唯一の道は倫理 : 大規模言語モデルを用いた倫理的研究のためのガイド

概要: LLMの倫理的考慮に関して NLPの研究者や LLMアプリケーション開発者が倫理的影響を評価するための実践的なリソースとして　LLM ETHICS WHITEPAPERを導入「簡潔な Do's and Don'ts」に基づいて「やるべきこと」と「やるべきでないこと」を明確にしつつプロジェクトの進行段階ごとに実践的な倫理ガイドが作成できます

技術や手法

1. **倫理シート :** AIプロジェクト開始前に考慮すべき質問（例　:「なぜこのタスクを自動化する必要があるのか？」）を提供。

2. **ALTAI（Assessment List for Trustworthy AI）:** AIシステムの信頼性を自己評価するためのリスト。

3. **内部監査フレームワーク :** アルゴリズム開発のプロセスを評価するフレームワーク。

4. **モデルカード :** モデルの使用目的や制限を明確にするためのドキュメントテンプレート。

5. **参加型設計 :** 関係者（特に影響を受けるコミュニティ）との協力を促進するためのツール。

1. **プロジェクトのライフサイクルに基づいた構造　**

論文では、プロジェクトの進行を以下の段階に分けています　:

1. プロジェクトの開始

2. データ収集

3. データ準備

Appendix