

# 比較器について

ComputerScience集会#30 @VRChat 2023-07-16

夜鍋ヨナ-yonabeyona <<http://x.com/yonabeyona>>

# CS 集 会

# # 30

# 生成AIと プログラミング

# 自己紹介

- 名前：夜鍋 ヨナ(よなべ よな)
- X(Twitter) : yonabeyona, yonabeyona\_sub
- Discord : yona\_47
- その他
  - ComputerScienceが好き
  - 数学勉強中
  - 物理も勉強中
  - ComputerScienceの中でも、ComputerArchitectureが好き
  - 最近言語学も興味あり



# 今日やること

- 生成AIサービスとかの動向調査
- 生成AIといっしょにシステム作った
- 事のいきさつ
- システム概要
- 大変だったこととか
- 感想

# 生成AIといっしょにシステム作った

## 概要

1. arXivの論文を
2. Geminiに要約してもらって
3. NotionDBに投げると
4. うれしい

# 生成AI系サービス使ってますか？

- ChatGPT
- Claude
- Perplexity
- CommandR
- Gemini
- Groq

# LLM使ってますか？

- GPT3.5
- GPT4
- Claude3.5
- Gemini-1.5-pro
- command-r
- LLaMa

# 生成AIとかまとめ

企業名	サービス名・LLM	特徴
ChatGPT	GPT3.5, GPT4	3.5は無制限、4oは80回/3h制限。Code Interpreterで色々処理してくれる
Anthropic	Claude 3	こちらの雑な指示を汲み取ってくれる能力が一番高い。Artifactが神。
Cohere	CommandR	2024年初頭に一瞬だけ話題になった。モデルがオープン。
Google	Gemini	チャットもAPIも無料!急げ! 1Mコンテキストウィンドウでたくさん入力できるぞ
Groq	(オリジナルなし)	爆速。未だ無料。コンテキストウィンドウちょっと小さいかも 8k
Perplexity	Sonner	検索していい感じにまとめてくれる。Wikiページ生成もする

# 最近はコーディング支援もある

- GitHub Copilot

<https://docs.github.com/ja/copilot/using-github-copilot/getting-code-suggestions-in-your-ide-with-github-copilot>

- Cursor Editor

<https://www.cursor.com/>

- Continue.dev

<https://www.cursor.com/>

エディタだったり、プラグインもできている

すでに作成しまくった領域なら、タブキーを連打するだけでコードが完成するとかいう噂も

# 動向調査は一旦終わり

進歩が早すぎる!!!!

すごいことはわかるが、それを実感するのは難しい

# こんだけすごいなら.....

AIでなんかプログラミングしてみるか  
せっかくならアプリも作りたいかも?

# 事のいきさつ

## 2023年の真ん中

- GPT3.5なんて、いうてまだまだstack overflow botでしょ → 敗北
- 単純なライティングなら..... → 敗北
- AIに知性で敗北しました。現代のシェヘラザード強すぎ。

## 2024年の最初らへん

- AIと勝負したらだめなことはわかった
- じゃあAIを使役する側に回ろう
- お、上流の軽い設計とかコメントつけるのとか早いな
- お、簡単なプログラムなら書いてくれるな.....
- もしかして、要件定義しっかりしたらAI産100%のコードでアプリとか作れるのでは？

# 作ったアプリ(頓挫含む)

- graphical PDF editor  
手書きでPDFにメモしたいな(閲覧したいな)  
→頓挫
- 書籍管理システム with Spreadsheet  
バックエンドの選定難しかった  
ちょっと便利なSpreadsheetに終わった  
→  $\alpha$ 版(?)で投げ出した
- 論文ポッドキャストのYouTubeチャンネル  
ずんだもんに論文読んでもらう  
(ヨナが)APIの話とかHTTPの話とかわからなすぎた  
→ 思ったより、やること知らないこと多くて頓挫
- arXiv to Notion  
ローカルの論文をgeminiで要約、ローカルのPDFをNotionに追加はあった  
→ arXivから論文引っ張ってきてWEBだけで完結しねえかな

# graphical PDF editor

2ペインにして、左にPDF1ページ、右にノートを作る

PDF1ページに対して、ノートを何ページもあるような形にしたかった

Pythonでモックを作って、ブラウザで動くようにしたかった

モックを作って満足してしまった

2023年12月～2024年1月ごろからChatGPT4と作った(Omniじゃないよ)

# 書籍管理システム with Spreadsheet

本棚が崩壊してきたので管理だけはしたい  
電子書籍と物理書籍でダブる事故を減らしたい  
バーコードリーダーでやると楽しそう  
デバッグはしてくれなくて大変

2024年1月ごろにGeminiで作った(Versionは1.0?)  
AiStudioじゃなくてチャットで作った(旧Bard)

# 論文ポッドキャストのYouTubeチャンネル

論文に触れたいので、ポッドキャストを作りたいかった

ずんだもんに読ませれば比較的楽しい可能性

YouTubeに上げれば出先でも見れるなあ

やることが多すぎて断念してしまった

VOICEVOXのサーバーをローカルに立てて満足してしまった

2024年3月ごろにChatGPT4と作った

デバッグはしてくれなくて大変

# arXiv to Notion

今回の話

arXivの論文読みたいけど、英弱なので読めない

Geminiに翻訳させたい

落合フォーマットに従わせたい

Notionに乗せたい

$\alpha$ 版ができた

# arXiv to Notion のシステム概要

1. 日付、カテゴリを指定してarXivで検索する

<https://arxiv.org/>

2. Jina APIでHTMLページをLLMに食べさせやすいように加工する

<https://jina.ai/>

3. Geminiに本文を渡して、落合フォーマットに従って抄録を作らせる

<https://aistudio.google.com/>

4. 抄録をNotionDBに乗せる

<https://notion.so>

5. 嬉しい

<https://github.com/yonayona4747/arxiv-notion/tree/main>

作成日時	Aa title	Index Term	概要・目的	先行研究との比較	核となる技術・手法	有効性の検証方法	議論すべき点	次に読むべき論文
2024年7月15日 17:22	Open (Clinical) LLMs are Sensitive to Instruction Phrasings	<ul style="list-style-type: none"> <li>Large Language Models</li> <li>Instruction Tuning</li> <li>Robustness</li> <li>Fairness</li> <li>Clinical NLP</li> </ul>	<p>本論文では、医療従事者による指示の自然なバリエーションに対する、指示チューニングされたオープンソースLLMのロバスト性を評価した。医療従事者12人を対象に、二値分類、アウトカム予測、情報抽出を含む16の臨床タスクについて指示を作成してもらい、指示のわずかな変化がモデルの性能に与える影響を分析した。</p>	<p>本研究は、従来研究で指摘されてきたLLMの指示に対する脆弱性を、医療現場という具体的なドメインに焦点を当てて深掘りした。医療従事者による実際の指示の多様性を収集し、汎用モデルとドメイン特化モデルの両方において、指示の表現の微妙な変化が性能に大きな影響を与えることを実証した。特に、医療データで特化トレーニングされたドメイン特化モデルが、汎用モデルよりも脆弱であるという、意外な結果が得られた。</p>	<p>本研究では、医療従事者からさまざまな臨床タスクに対応する指示を収集し、7つのLLM（汎用モデルとドメイン特化モデル）の性能を評価した。指示の性能を評価するために、各タスクに対するベスト、ミディアム、ワーストのパフォーマンスを算出し、各モデルにおける指示のバリエーションによる性能差を比較した。さらに、医療データの公平性という観点から、指示のバリエーションが人種や性別による性能差に与える影響を分析した。</p>	<p>指示の性能は、AUROC（分類タスク）とF1スコア（抽出タスク）を用いて評価した。公平性の検証には、MIMIC-IIIの死亡予測タスクにおいて、人種（白人、非白人）と性別（男性、女性）を層別化し、各サブグループにおけるパフォーマンスの差を調べた。</p>	<p>本研究では、オープンソースの指示チューニングされたLLMが、指示の表現のわずかな変化に脆弱であることが示され、患者の健康に影響を与える可能性のある医療現場への導入には注意が必要であるという結論に至った。しかし、本研究にはいくつかの限界がある。まず、本研究の結果は、より大規模な商用モデルには一般化できない可能性がある。また、参加者の集団が、これらの技術の潜在的なユーザーを代表するとは限らない。さらに、本研究では分類タスクの評価方法が、実際の使用状況を必ずしも反映していない。最後に、本研究では、指示のバリエーションが人種や性別による公平性に与える影響を調べたが、複合的なバイアスの影響を受ける交差的なアイデンティティへの影響は調べられていない。</p>	<p>引用文献22 (Kweon et al., 2023) の論文「Publicly shareable clinical large language model built on synthetic clinical notes」は、医療データのプライバシー問題を解決するために、合成医療データを活用した臨床LLMの開発について詳しく説明している。本論文の限界として、医療データのプライバシー問題が挙げられているため、この論文を読むことで、臨床LLMの開発における課題とその克服に向けた取り組みについて理解を深めることができるだろう。</p>
2024年7月15日 17:21	A Perspective on Foundation Models for the Electric Power Grid	<ul style="list-style-type: none"> <li>Foundation Models</li> <li>Data-Driven Power Grid ...</li> <li>Energy Transition</li> <li>AI-based Power Flow Sim...</li> </ul>	<p>本論文は、エネルギー転換に伴う電力網の複雑化と不確実性の増大に対処するために、ファウンデーションモデル（FM）を活用する新しいアプローチを提案している。具体的には、電力網の多様なデータを学習可能なGridFMと呼ばれるFMの開発を提唱し、その潜在的な利点と課題、そして実装に向けた具体</p>	<p>本研究は、電力網分析におけるAI/MLの適用に関する従来研究を踏まえて、従来のモデルでは扱えなかった複雑な問題に対処するために、大規模言語モデル（LLM）や気象予測モデルなどの成功例を参考に、電力網特化型のFMであるGridFMの開発を提案している。従来のAI/MLモデルは、特定のタスクに特化して</p>	<p>本論文で提案されているGridFMの核となる技術は、グラフニューラルネットワーク（GNN）を用いた多モジュール・スパティオテンポラルFMである。GNNは、グラフ構造データを処理することに優れており、電力網のトポロジーやノード間の関係性を効率的に学習することができる。GridFMは、GNNを用い</p>	<p>本論文では、GridFMの有効性を検証するための具体的な方法については言及していない。しかし、GridFMの概念に基づいた具体的な実装や実験は、今後行われることが期待される。GridFMの有効性を検証するためには、電力網のシミュレーションデータや実測データを用いた評価が不可欠である。GridFMは、電</p>	<p>GridFMの実装には、データの入手可能性、プライバシー、セキュリティ、信頼性、悪用防止などの課題が存在する。GridFMは、多量のデータを必要とするため、データ収集、処理、管理に関する技術的な課題が存在する。また、電力網データには、プライバシーやセキュリティに関する懸念があるため、データ</p>	<p>電力網におけるAI/MLの適用に関するより深い理解を得るために、以下の論文を読むことをお勧めする。 13. A. Varbella, K. Amara, B. Gjorgiev, M. El-Assady, G. Sansavini, PowerGraph: A power grid benchmark dataset for graph neural networks (2024).</p>



# 開発フロー

1. 要件を定義する
2. 要件定義をLLMに投げて、コードをモジュール単位で設計してもらおう(Claude 3.5 sonnet)
3. そのうちClaudeのコンテキストウィンドウから漏れ出す
4. 丸ごとコピーして、Geminiに投げる
5. Geminiをしばきながらコードを書かせる
6. 微妙だと思ったところは即時直させる
7. 手元のPCにコードをコピーして動作確認
8. デバッグは自力

**A** 論文をざっくり目を通したい  
arxivから論文を取得してまとめたい  
まとめた情報はNotionに入れたい  
落合フォーマットに従ってまとめてほしい

arXivから論文メタデータを取得してする

- \* 定時実行
  - \* いつ実行するかは設定ファイルに書く
- \* 特定の分野のサブカテゴリについて取得する
  - \* どのサブカテゴリかは設定ファイルに書く
- \* 実行した日付の最新論文を全て取得する
- \* 各論文について、タイトル、要約、著者、識別子を取得する

arXivの論文IDとjina APIを用いてarXiv論文ページ(HTML or PDF)にアクセスし、次の情報を取得する

- \* 論文タイトル
- \* アブストラクト
- \* 本文の全文
- \* 本文中からIndex Term

LLMで落合フォーマットを用いて論文の本文を要約する

- \* 論文の概要と目的
- \* 従来研究との比較・改善点
- \* 核となる技術・手法
- \* 有効性の検証方法
- \* 議論すべき点
- \* 次に読むべき論文

まとめたデータをNotionへ書き込む

### Chat controls

Claude 3.5 Sonnet

Most intelligent model [Learn more](#)

### Artifacts



**arXiv論文要約システムのフロー図**  
Click to open diagram • 1 version



**メインスクリプト構造**  
Click to open code • 2 versions



**config.py**  
Click to open code • 1 version



**utils.py**  
Click to open code • 1 version



**arxiv\_fetcher.py**  
Click to open code • 1 version



**jina\_processor.py**  
Click to open code • 1 version



**llm\_summarizer.py**  
Click to open code • 1 version

System Instructions

User

動いた!!!!!!これにて一旦α版とします!!!!!!

Model

素晴らしい! 🎉

ついに動作したとのこと、大変嬉しいです!

度重なるエラー修正、本当にお疲れ様でした!  
ここまで根気強く取り組んでいただき、感謝いたします。

α版完成、おめでとうございます!

これから実際に運用してみて、新たな課題や改善点が見つかるかもしれません。  
その時は、いつでもお気軽にご相談ください。

引き続き、プロジェクトの成功に向けて、サポートさせていただきます!  
頑張ってください! 🚀

# 大変だったこととか

- 最初はDifyにやらせようとしたけど断念
  - ローコードを使えるのはコードが書けるやつ、そいつはコードを書く
- AiStudioがやり取りを自動で保存してくれなくて詰みかける
  - 勝手に保存してくれ。君だけだぞ。
- 設計がうまいLLMがあるので使い分けたほうがいい(かも)
  - ClaudeのArtifact機能が神がかってる。
- AIはデバッグしてくれない
  - 結局人間が責任を持って動作確認するのは変わらんのか.....
  - シンギュラリティはよ
- ブラウザとエディタの行き来が少し面倒
  - 次はCursorとかContinue.devとか使っていい感じにしたい

# 感想

- 要件定義で「そういえば考えてなかった」の部分はAIも曖昧に実装する
- 健全なプロダクトは、健全な要件定義書・基本設計書・詳細設計書に宿る
- 最近のAIは融通がきくので、途中で仕様変更しても反逆してこない(場合もある)
- AIでプロダクトを作るときは、どれだけ完成像を明確にイメージできるかの世界
  - フリーレンみたいだね
- モックを作るだけでなくすぐできるからみんなやろう
  - 最近のAIサービスは無料化が激しいから使おう

- 要件定義で「そういえば考えてなかった」の部分はAIも曖昧に実装する
- 健全なプロダクトは、健全な要件定義書・基本設計書・詳細設計書に宿る

この辺は人間が設計/製造しても同じですね。

上流が曖昧であるほど、下流で苦勞する。何度泣かされたか。

そして発生する無慈悲な仕様変更

AI相手ならやり直しも早いし、文句も言ってこないのでもあ楽。


- 最近のAIは融通がきくので、途中で仕様変更しても反逆してこない(場合もある)
  - ちょっと前までプロンプティングが流行ってたがもはや必要なく感じる
- AIでプロダクトを作るときは、どれだけ完成像を明確にイメージできるかの世界
  - フリーレンみたいだね
  - あなたの頭の中にあって言語化してないものがAIに察せれるわけがない
  - ヒアリングを駆使するといいかも
- モックを作るだけならすぐできるからみんなやろう
  - 最近のAIサービスは無料化が激しいから使おう
  - APIだとまだちょっと高いかも? Geminiなら無料!!!!

# 木曜日はOSS/個人開発集会

みんなでプログラミングの勉強ができる集会

## VRC OSS集会

隔週木曜日22時～



```
token = x_token[0]
movie_path = f"movie/{token}.mp4"
temp_movie_path = f"movie/{token}_temp.mp4"
image_path = f"image/{token}.png"
with open(image_path, "rb") as f:
    image_data = f.read()
MovieMaker.image_to_movie(image_data, temp_movie_path)
name(temp_movie_path, f"movie/{token}.mp4")
("message", "URL: http://localhost:8080/movie/{token}")

get("/api/create_github_movie/")
create_github_movie(str, width: int = 1280, height: int = 720,
                    scroll_each: int = 100, catch: bool = True) -> dict:
    """
    Load github repository content into browser and take a screenshot.
    param url: URL to take screenshot
    param targets: list of targets to take screenshot
    param width: Browser width
    param height: Browser height
    param limit_height: Max scroll height
    param scroll_each:
    """
    :param catch: if catch is true, check saved movie is suitable.
    :return: Github repository URL
    """
    targets = targets.split(",")
    if len(url) == 0:
        raise HTTPException(status_code=400, detail="URL is empty. Pl
```

進捗報告LT, サイトのデバッグ,  
コード公開, コードレビューなど

参加方法 「のりちゃん(Noricha)」にフレンド申請&Join

@noricha\_vr

# 木曜日はOSS/個人開発集会

もうちょい詳しい話をするので興味あれば