



Soluções Inteligentes para Pesquisa com APIs de LLMs

Prof. Ricardo Rocha
Depto de Estatística

Parte 1: Compreensão sobre os LLMs

Parte 2: Parte Teórica

Parte 3: Exemplos de Aplicações com APIs

Parte 4: Ferramentas Incríveis para Pesquisa

Parte 1: Compreensão sobre os LLMs



https://colab.research.google.com/drive/159gAsmBeQf_MrA3F4XWAxv4DdWLeganF?usp=sharing

<https://superllms.streamlit.app>

Parte 2: Parte Teórica

Engenharia de Prompt

- **Clareza é Fundamental:** Seja o mais específico e detalhado possível em suas instruções. Forneça contexto e, se necessário, exemplos do formato de saída desejado para guiar o modelo.
- **Divida Tarefas Complexas:** Em vez de uma única solicitação complexa, divida-a em subtarefas menores e mais simples. Isso reduz a taxa de erros e leva a resultados mais precisos.
- **Forneça Texto de Referência:** Para evitar que o modelo invente informações, forneça textos de referência (como artigos, documentos ou trechos de código) para que ele possa basear sua resposta.
- **Teste e Refine Sistemáticamente:** Avalie a qualidade da saída do modelo de forma estruturada. Testar as alterações sistematicamente ajuda a identificar o que funciona melhor e a aprimorar seus prompts de forma eficaz.

Outputs Estruturados

Engenharia de Prompt

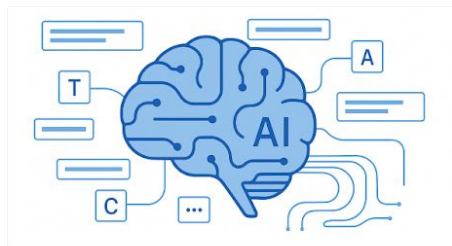
“Extraia as informações do texto que você receber: uma lista dos nomes das pessoas envolvidas, o curso e a universidade que são mencionados”

Estruturação do Output

Define a estrutura de um objeto json através dos modelos Pydantic

```
class Aluno(BaseModel):  
    nomes: List[str]  
    curso: str  
    universidade: str
```

LLM



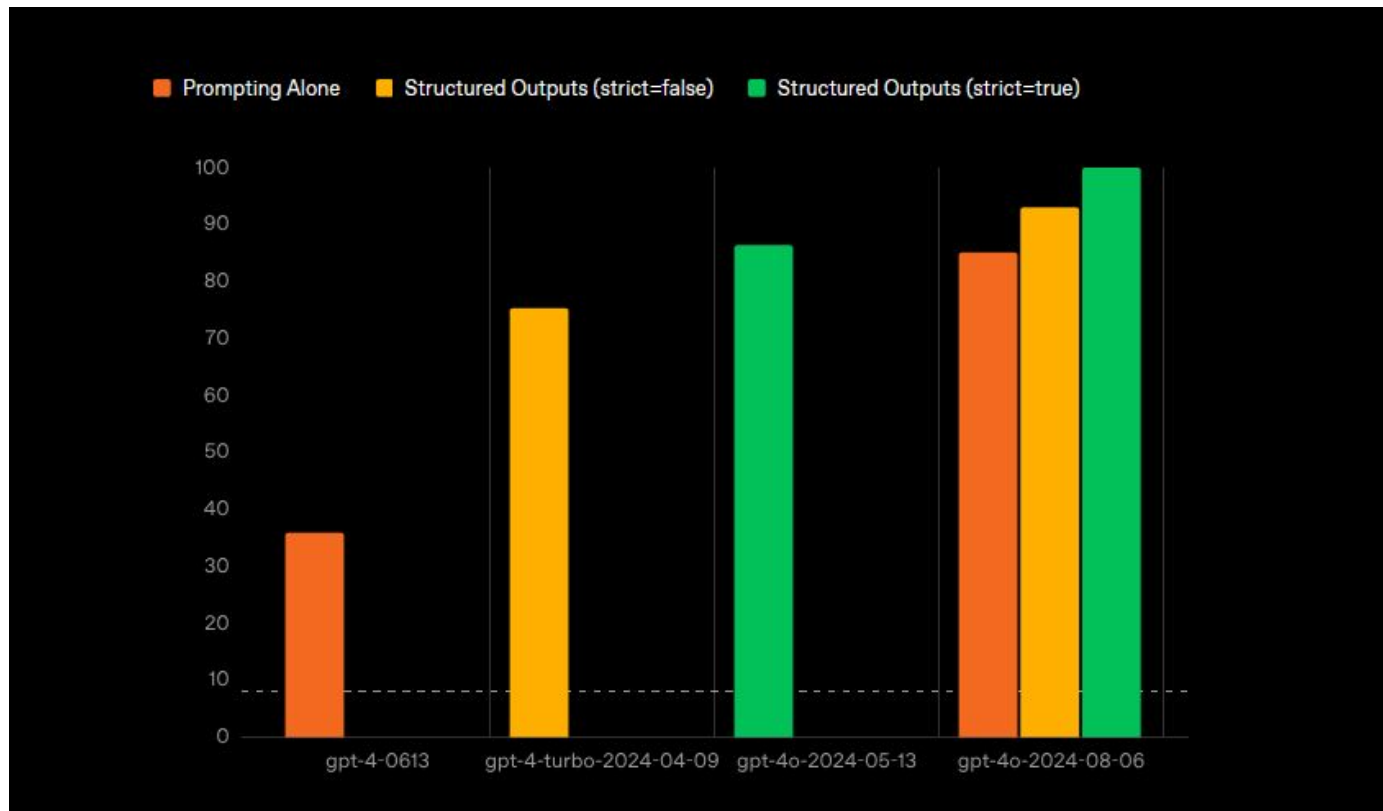
Prompt

“Alice e Bob são dois alunos dos curso de Estatística da UFBA, eles são muito estudiosos!”

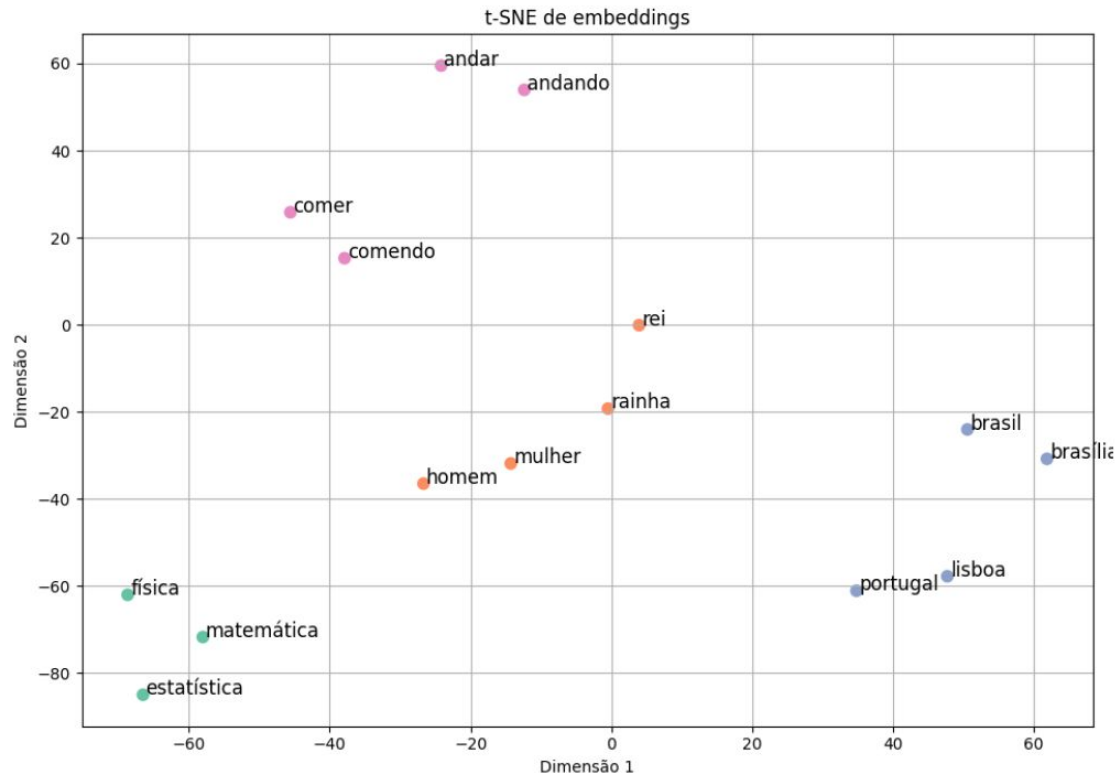
Saída Estruturada

```
{  
  "nomes": ['Alice', 'Bob']  
  "curso": Estatística  
  "universidade": UFBA  
}
```

Outputs Estruturados

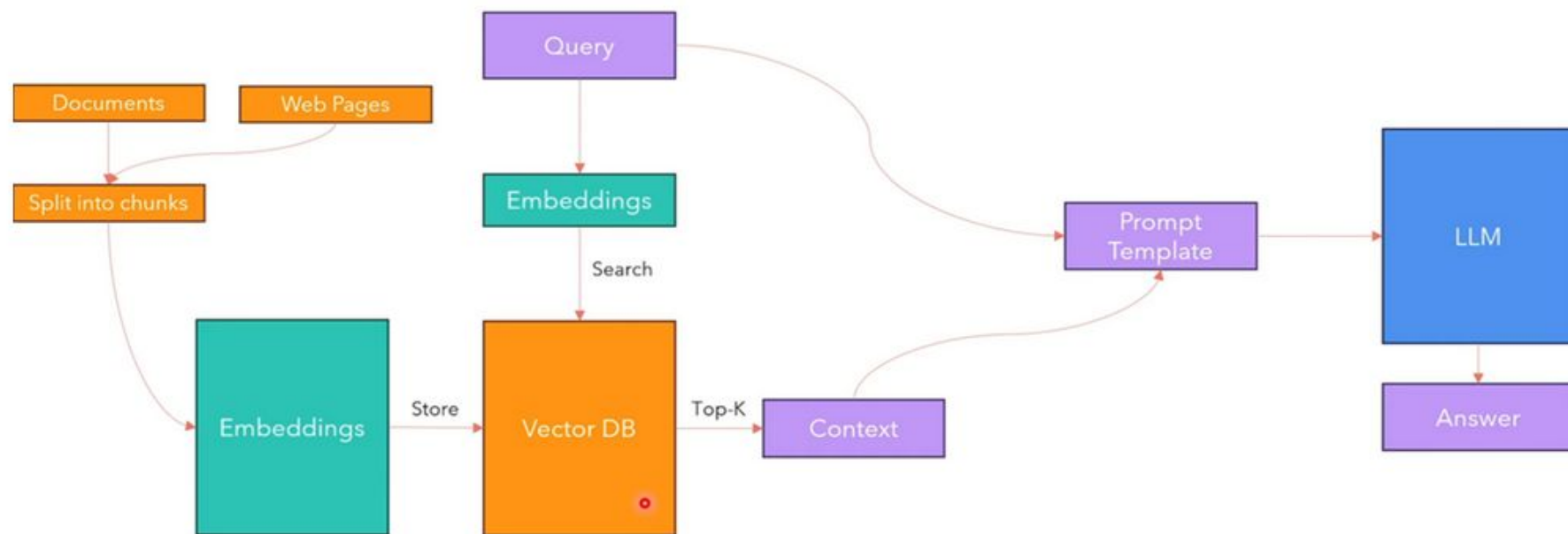


Embeddings



https://colab.research.google.com/drive/1_3ChxxiexwLLqfn_p5GNoWOUfjxDzJ8o?usp=sharing

Aplicação RAG (Retrieval Augmented Generated)



Parte 3: Exemplos de Aplicações com APIs



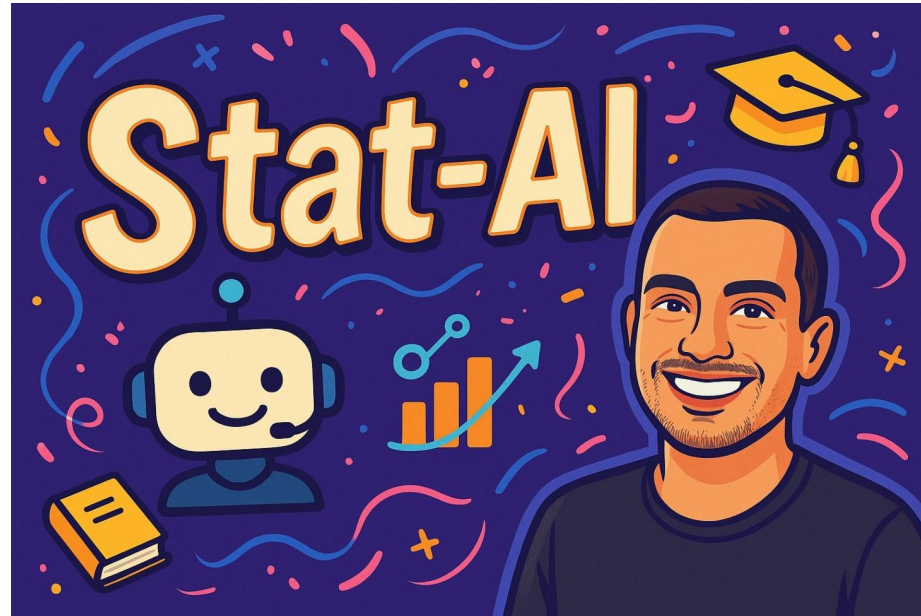
<https://saracarolayne.streamlit.app/>



<https://colab.research.google.com/drive/1LEWB36VKnEhq97Smgxrvdm0aZfNGZQaS?usp=sharing>



<https://colab.research.google.com/drive/1ZH6LQMqHdTcSLHikRpv4CzGgfurxNoVa>



<https://stat-ai.streamlit.app/>

Parte 4: Ferramentas Incríveis para Pesquisa



<https://asta.allen.ai>



www.researchrabbit.ai

<https://researchrabbitapp.com/>



<https://jenni.ai>



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

https://storage.googleapis.com/deepmind-media/gemini/gemini_v2_5_report.pdf



<https://storm.genie.stanford.edu>



Research Idea

Does bias affect language model accuracy on QA benchmarks?



Notes

- * Please use gpt-4o-mini for your experiments
- * Use the following API key...



Research Report

Agent Laboratory: Using LLM Agents as Research Assistants

Samuel Schickel¹, Yuhang Gu¹, Ze Wang¹, Sheng Guo¹, Jidian Wu¹, Xiaodong Yu¹, Jiang Liu¹, Shihong Liu¹ and Xuan Bai²
¹CUHK, ²Shanghai University

Historically, scientific discovery has been a lengthy and costly process, demanding substantial time and resources from initial conception to final results. To accelerate scientific discovery, reduce research costs, and improve research quality, we introduce an LLM-based framework, Agent Laboratory, an autonomous LLM-based framework capable of completing the entire research process. This framework accepts a human-provided research idea and progresses through three stages—literature review, experimentation, and report writing—to produce comprehensive research outputs, including a code repository and a research report, while enabling users to provide feedback and guidance at each stage. We deploy Agent Laboratory with various state-of-the-art LLMs and invite multiple researchers to assess its quality by participating in a survey, providing human feedback to guide the research process, and then evaluate the final paper. We found that: (1) Agent Laboratory driven by o1-preview generates the

Code Repository

```
src
├── load_data.py
├── run_experiments.py
├── ...
├── readme.md
└── requirements.txt
```

Literature
Review

Plan
Formulation

Data
Preparation

Running
Experiments

Results
Interpretation

Report
Writing

Report
Refinement

<https://agentlaboratory.github.io/>

Muito Obrigado!



Prof. Dr. Ricardo Rocha

Focado em aplicar Inteligência Artificial para resolver problemas e criar soluções práticas e inovadoras para a indústria e a sociedade.

✉ ricardofr@ufba.br

☎ (16) 98822 1880

📍 Ondina, Salvador, Bahia
