Relátorio de pesquisa

Victor Carneiro dos Santos Angelo - 122110725

Geração do Dataset

Metodologia de Geração

O processo de construção do dataset seguiu o fluxo abaixo:

1. **Extração de Dados:** Coleta de informações contextuais da Wikipedia sobre os clubes da Série A e o histórico da competição.
2. **Geração Sintética:** Utilização do modelo **Gemini 3 Flash** para gerar 1.000 instruções, seguindo o formato de dados proposto no repositório *LLMs from Scratch* (Sebastian Raschka).
3. **Auditoria e Curadoria:** O modelo **Gemini 3 Pro** foi empregado para analisar o lote gerado, com o objetivo de excluir:
    - **Vazamento de Resposta (Data Leakage):** Instruções onde a resposta já estava contida ou sugerida no campo de entrada (`input`).
    - **Instruções Inválidas:** Exemplos com erros lógicos ou falta de utilidade prática.

Primeira tentativa

1. **Base de Conhecimento:** Foram utilizadas informações extraídas de páginas da Wikipédia abrangendo todos os clubes da Série A do Brasileirão, além de dados sobre competições sul-americanas e seleções nacionais.

2. **Disponibilidade:** O dataset completo pode ser acessado através do seguinte link: [Google Drive - Dataset]().

3. **Estatísticas e Filtragem:** Inicialmente, foram geradas **1.000 instruções**. Após uma etapa de filtragem (para remoção de inconsistências e vazamento de dados), restaram **930 instruções** válidas. Exemplos de entradas descartadas: Foram removidos casos de *data leakage* (onde a resposta estava no input) ou contradições factuais no enunciado.


```
[
  {
    "instruction": "Name the stadium that hosted the opening match of
    "input": "Arena Corinthians",
```

```
    "output": "The match was held at the Arena Corinthians (now Neo (
  },
  {
    "instruction": "Identify the city where the 'Arena Pantanal' is l
    "input": "Curitiba",
    "output": "The stadium is located in Cuiabá."
  }
]
```


4. **Problemas:** durante a fase de testes, identificou-se uma acentuada dificuldade do modelo GPT-2 em responder corretamente às perguntas. Hipótese Principal: A baixa performance e as alucinações sugerem que o modelo original não foi exposto a esses dados específicos durante o seu pré-treino. Como o GPT-2 carece dessa base de conhecimento prévia sobre o nicho do futebol sul-americano, o fine-tuning de instruções não foi suficiente para garantir a precisão factual.

Segunda tentativa

1. **Base de Conhecimento:** Com o objetivo de mitigar as dificuldades de convergência e alucinação da primeira tentativa, o foco da extração de dados foi alterado. Substituímos o nicho específico do futebol por páginas da Wikipédia com temas de conhecimento geral, incluindo literatura clássica, obras populares e fatos históricos amplamente documentados.

2. **Disponibilidade:** O dataset completo pode ser acessado através do seguinte link: [Google Drive - Dataset].

3. **Estatísticas e Filtragem:** Inicialmente, foram geradas **1.100 instruções**. Após uma etapa de filtragem (para remoção de inconsistências e vazamento de dados), restaram **1010 instruções** válidas.

Prompts utilizados

1. **Geração de Dados**

```
===
Tarefa: Agir como um gerador de datasets sintéticos para fine-tuning

===
Objetivo: Gerar pares de Instrução, Entrada e Saída seguindo rigorosa

===
Regras de Geração
```

```
    Idioma: Todo o conteúdo deve ser escrito em Inglês.
    Base de Conhecimento: Utilize apenas informações de 2022 ou ante
    Uso do Campo 'Input': Utilize o campo input apenas se a instruçã
    Formato de Saída: Retorne os dados estritamente em formato JSON

===
Exemplo:
    {
        "instruction": "Evaluate the following phrase by transformin
        "input": "freind --> friend",
        "output": "The spelling of the given phrase \"freind\" is co
    }
```

2. **Filtragem de Dados**

```
===
Você é um Especialista em Curadoria de Dados (Data Curator) para mo

===
Critérios de Avaliação (Filtros)

Corte Temporal (Crucial): Remova qualquer exemplo que mencione event
Idioma: Todas as instruções, entradas e saídas devem estar obrigato
Integridade do Formato JSON: Verifique se o campo input está vazio
Veracidade e Fonte: O conteúdo deve ser baseado em conhecimento púb
Qualidade da Resposta: A resposta no campo output deve ser útil, di

===
DADOS
Arquivo Json
```

Treinamento

Dataset

Para melhorar o dataset criado, foi realizado a concatenção com o dataset original do repositório do Sebastian Raschka. Foram utilizadas 1000 instruções de cada, totalizando 2000 instruções.

Configuração do Experimento (Hiperparâmetros)

O treinamento foi executado utilizando o script oficial do repositório *LLMs from Scratch*, configurado com os seguintes hiperparâmetros:

- **Batch Size:** 8
- **Learning Rate (LR):** 5e-5 (0.00005)
- **Weight Decay:** 0.1
- **Épocas:** 1

Modelos e Divisão de Dados

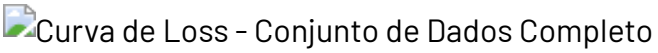
Foram conduzidos dois experimentos distintos utilizando a arquitetura **GPT-2 (pré-treinado)**:

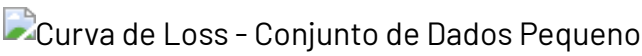
1. **Modelo GPT-2 Full:** Treinado com o dataset completo de 2.000 instruções.
    - *Split:* 1.600 (Treino) | 200 (Validação) | 200 (Teste).
2. **Modelo GPT-2 Small:** Treinado com um subconjunto reduzido de 500 instruções.
    - *Split:* 400 (Treino) | 50 (Validação) | 50 (Teste).

Resultados e Avaliação

Os modelos foram avaliados com base na perda de validação (*Validation Loss*), apresentando os seguintes indicadores:

Modelo	Val Loss
GPT-2_Full	0.504
GPT-2_Small	0.596

1. Modelo GPT-2 Full (2.000 instruções)

Curva de Loss - Conjunto de Dados Completo

2. Modelo GPT-2 Small (500 instruções)

Curva de Loss - Conjunto de Dados Pequeno

Testes e Avaliação Qualitativa

Geração de Respostas Estruturadas (JSON)

Para comparar o desempenho dos modelos, foram gerados dois arquivos JSON contendo as inferências baseadas em instruções de teste. O objetivo foi contrastar a capacidade de resposta entre o modelo original e as variantes ajustadas.

1. **Comparativo Base vs. Full:** Contém as respostas do modelo original (sem *fine-tuning*) e do modelo ajustado com o dataset de 2.000 instruções.
2. **Modelo Small:** Contém as inferências do modelo ajustado com apenas 500 instruções.

Exemplo de Estrutura de Saída

O exemplo abaixo ilustra o comportamento típico de cada modelo. Note que o modelo sem o ajuste adequado tende a entrar em um *loop* de repetição do prompt (regressão infinita), enquanto o modelo ajustado busca formular uma definição direta.


```
{
  "instruction": "Define the term 'kinetic energy'.",
  "input": "",
  "output": "Kinetic energy is the energy that an object possesses due to
  "model_1_response": "Kinetic energy is the energy that is released by a
  "model_2_response": "Write a response that appropriately completes the
}
```


Avaliação: LLM as a Judge

Para uma análise comparativa e quantitativa das inferências, implementamos a metodologia **LLM as a Judge**. Utilizou-se uma instância local do modelo **Llama 7B** como juiz imparcial para avaliar as respostas geradas pelos modelos ajustados.

Metodologia de Avaliação

O juiz avaliou cada par de respostas (Modelo A vs. Modelo B) de forma cega, baseando-se em três pilares fundamentais, com pontuações de 0 a 5:

1. **Acurácia Factual (*Factual Correctness*):** Veracidade das informações fornecidas.
2. **Aderência à Instrução (*Instruction Adherence*):** Capacidade do modelo de seguir exatamente o que foi solicitado.
3. **Clareza e Utilidade : *Qualidade*** da escrita e útil a resposta é para o usuário final.

Configuração do Prompt de Avaliação

O modelo avaliado foi instruído a retornar um objeto JSON, garantindo que os dados obtidos sejam processados programaticamente para a geração de métricas finais.

```
# Estrutura de saída esperada (Strict JSON)
example_output = {
```

```
    "model_a": {
        "factual_correctness": 0,
        "instruction_adherence": 0,
        "clarity_and_usefulness": 0,
        "total": 0
    },
    "model_b": {
        "factual_correctness": 0,
        "instruction_adherence": 0,
        "clarity_and_usefulness": 0,
        "total": 0
    },
    "winner": "A | B | Tie",
    "justification": "Breve comparação objetiva justificando a decisão."
}
```

Análise Consolidada de Resultados

Os resultados obtidos através da avaliação do LLM como Juiz demonstram um salto qualitativo substancial após o processo de ajuste fino. O modelo GPT-2 Fine-tuned (Full) superou consistentemente a versão base em todas as métricas de desempenho e praticidade.

Comparativo de Desempenho (Pontuações Médias)

Abaixo, apresentamos a mídia detalhada das contribuições atribuídas pelo avaliador (escala 0–5 por prêmios):

Métrica	GPT-2 Base (Modelo B)	GPT-2 Ajustado (Modelo A)
Precisão Factual	1,37	**3,79**
Aderência à Instrução	1,54	**4.12**
Clareza e Utilidade	1,48	**4.15**
Pontuação Total	4,39	**12.06**

Taxa de Vitória (*Win Rate*)

Na comparação direta (*head-to-head*), o modelo ajustado com o conjunto de dados completo dominou as avaliações:

- **Vitórias do Modelo A (Completo):** 75,00%
- **Vitórias do Modelo B (Base):** 17,50%
- **Empates:** 7,50%


Distribuição de Pontuações Totais

Impacto do Volume de Dados

A comparação entre as variantes Full (2.000 instruções) e Small (500 instruções) revela que o volume de dados foi determinante para a estabilidade do modelo:

- **GPT-2 Completo (Total Médio):** 12.06
- **GPT-2 Pequeno (Total Médio):** 8,46

O ganho médio de desempenho do modelo Full em relação à base foi de 7,67 pontos, atingindo picos de melhoria de até 15 pontos em instruções complexas onde a base falhava completamente.

Resumo Estatístico Percentual

A tabela abaixo normaliza os resultados para uma escala percentual, permitindo visualizar a amplitude de desempenho de cada configuração:

Modelo	Pontuação (%)	Melhor Caso (%)	Pior Caso (%)
Base GPT-2	29,25%	100,0%	0,00%
GPT-2 Ajustado (Pequeno)	56,40%	100,0%	0,00%
GPT-2 Ajustado (Completo)	**80,40%**	100,0%	6,67%
