Kaggle - LLM Prompt Recovery 竞赛银牌算法概览

竞赛概览：

竞赛要求参赛者利用算法恢复用于改写给定文本的 LLM 提示。竞赛使用 1300 多个原始文本及其经 Gemma（Google 新开放的大语言模型）改写的版本组成的数据集进行测试。评估标准基于 sentence-t5-base 模型生成的嵌入向量，通过“锐化余弦相似度”（Sharpened Cosine Similarity）计算得分。

所用算法：

1. 生成训练样本数据：
   a) 生成 prompt：利用 ChatGPT 大量生成改写文本的 prompt；
   b) 获取原始文本：从 https://huggingface.co/datasets/Skylion007/openwebtext 获取开源网络文本，过滤较长文本；
   c) 重写原始文本：使用步骤 a 与 b 的文本数据输入到谷歌开源 LLM 模型 gemma-7b-it，生成新文本数据，用此方法大量生成训练样本。
2. Seq2Seq 模型：
   a) 预处理：将生成样本的 prompt 转化为 embedding 向量化存储，以便加速训练；
   b) 构建训练 pipeline：将原始文本和重写文本一起输入到 deberta-v3-large 模型，拼接两者的特征输出，与实际 prompt 的 embedding（即步骤 1 的向量数据）进行相似度训练；
   c) 检索库：构建一个拥有大量 prompt 的检索库，以便推理检索；
   d) 推理：将步骤 b 训练完成的模型线上推理，预测得到 prompt 的 embedding 向量，从步骤 c 的检索库中寻找最相似的 prompt 文本，作为 seq2seq 模型的预测输出。
3. Phi2-微调模型：使用开源 phi2 微调模型推理预测，截取关键文本作为模型预测输出。
4. zero-shot 的 LLM 模型：使用开源模型 Mistral-7B-Instruct-v0.2，输入数条范例，直接使用相关指令让模型预测。对模型预测结果修剪，去除多余的符号或文本。
5. 集成预测：将三种模型的预测文本进行字符串拼接，作为最终的预测结果。

数据说明：

prompts_df.csv：改写文本的 prompt

train_clean.parquet：训练样本数据

validation826.csv：验证集

Silver Medal Algorithm Overview for **LLM Prompt Recovery** Competition

Competition Overview:

In recent years, the development of Large Language Models (LLMs) is becoming matured, making the text they generate increasingly difficult to distinguish from human writing. The competition required participants to develop a machine learning model capable of accurately detecting whether an essay was written by a student or an LLM. The competition dataset included essays written by students and articles generated by various LLMs. This competition was a typical binary classification problem, with the evaluation metric being AUC.

Algorithm Descriptions:

1. **Training Sample Generation**:
    a) **Prompts Creation**: Extensively generate rewriting prompts using ChatGPT.
    b) **Original Texts**: Source open web texts from Hugging Face, filtering longer texts.
    c) **Text Rewriting**: Input texts from steps above into Google's open LLM, gemma-7b-it, to create rewritten text data for training samples.
2. **Seq2Seq Model**:
    a) **Preprocessing**: Convert prompts from generated samples into embedding vectors for faster training.
    b) **Training Pipeline**: Input original and rewritten texts into deberta-v3-large model, concatenate feature outputs, and train on similarity with actual prompt embeddings.
    c) **Retrieval Database**: Create a large database of prompt embeddings for inference retrieval.
    d) **Inference**: Use the trained model for online inference to predict prompt embeddings and retrieve the most similar prompt text from the database.
3. **Phi2 Fine-Tuning Model**: Employ an open-source Phi2 fine-tuning model for prediction, focusing on key text segments.
4. **Zero-Shot LLM Model**: Use the open-source model Mistral-7B-Instruct-v0.2, inputting examples to predict directly.
5. **Ensemble Prediction**: Combine predictions from the three models by string concatenation for the final result.

Data Files:

- **prompts_df.csv**: Prompts for rewritten texts.
- **train_clean.parquet**: Training data samples.
- **validation826.csv**: Validation set.