Kaggle - LLM - Detect AI Generated Text竞赛银牌算法概览

竞赛概览：

在近年来，大型语言模型（LLM）的发展日益成熟，它们生成的文本越来越难以与人类的写作相区分。竞赛要求参赛者开发一个能准确检测出一篇文章是由学生还是LLM写成的机器学习模型。竞赛数据集包含了学生写的论文和由各种LLM生成的文章。该竞赛为典型的二分类问题，评价指标为AUC。

所用算法：

本次竞赛使用了三种模型进行推理：

1. 基于与比赛数据集分布相似构成特定议论文数据集（DAIGT V2 Train Dataset）的线性模型：
   1. 依据相似度过滤重复性数据；
   2. 使用测试集文本预训练分词器，将所得分词器tokenizer分词训练集与测试集所有文本，得到一致词汇表的统计特征；
   3. 分词完成后使用TFIDF获取Ngram（3,5）文本统计特征向量；
   4. 将上述特征输入MultinomialNB与SGDClassifier构成的集成分类器训练，然后预测结果。
2. 基于大规模数据集（Pile and Ultra、Human vs. LLM Text Corpus）的深度学习模型：
   1. 采集网络开源数据，分别来自于人工写作与LLM大模型对话；
   2. 将大规模数据简单处理后输入到文本二分类模型deberta-v3-small微调，得到训练完成的权重，然后在Kaggle进行推理。
3. 开源语言模型（其他参赛者的开源结果，主要用于集成）：利用第三方数据集进行语言模型微调训练完成，然后在Kaggle推理预测结果。
4. 集成预测：将三种建模方法的预测结果进行rank scale后加权融合得到最终预测。

数据与模型链接：

Pile and Ultra：https://www.kaggle.com/datasets/canming/piles-and-ultra-data

Human vs. LLM Text Corpus：https://www.kaggle.com/datasets/starblasters8/human-vs-llm-

text-corpus

DAIGT V2 Train Dataset：https://www.kaggle.com/datasets/thedrcat/daigt-v2-train-dataset

开源语言模型: https://www.kaggle.com/code/mustafakeser4/train-detectai-distilroberta-0-927

代码说明：

深度学习模型训练代码：deberta\_train\_exp5.py

嵌入可视化代码：embed\_vis.py

完整推理代码：llm-detect-code.ipynb

Silver Medal Algorithm Overview for

**Kaggle - LLM - Detect AI Generated Text** 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:**

This competition used three models for inference:

1. A linear model based on a specific argumentative essay dataset (DAIGT V2 Train Dataset) with a distribution similar to the competition dataset:
   1. Filtering out repetitive data based on similarity;
   2. Pre-training a tokenizer with test set texts, then tokenizing training and test set texts to obtain statistical features with a consistent vocabulary;
   3. After tokenization, using TFIDF to obtain Ngram (3,5) text statistical feature vectors;
   4. Inputting the above features into an ensemble classifier composed of MultinomialNB and SGDClassifier for training, then predicting the results.
2. A deep learning model based on large datasets (Pile and Ultra, Human vs. LLM Text Corpus):
   1. Collecting open-source data from the internet, coming from both human writings and LLM dialogues;
   2. After simple processing of the large-scale data, inputting it into the text binary classification model deberta-v3-small for fine-tuning, obtaining the trained weights, and then performing inference on Kaggle.
3. Open-source language models (open-source results from other participants, mainly used for ensemble): Using third-party datasets for language model fine-tuning training completion, then predicting results on Kaggle.
4. Ensemble Prediction: Weighted fusion of the prediction results of the three modeling methods after rank scaling to obtain the final prediction.

**Data & Model Links:**

* Pile and Ultra:

https://www.kaggle.com/datasets/canming/piles-and-ultra-data

* Human vs. LLM Text Corpus:

https://www.kaggle.com/datasets/starblasters8/human-vs-llm-text-corpus

* DAIGT V2 Train Dataset:

https://www.kaggle.com/datasets/thedrcat/daigt-v2-train-dataset

* Open-source Language Model:

https://www.kaggle.com/code/mustafakeser4/train-detectai-distilroberta-0-927

**Code Description:**

* Deep learning model training code: deberta\_train\_exp5.py
* Embedding visualization code: embed\_vis.py
* Complete inference code: llm-detect-code.ipynb