Fine-Tuning and Data Evaluation

NCHC LLM DevOps
Engineering Assistant
Yenyun Chen

6/18/2025

Outline

- Introduction of Pipeline Services
- Usage of Pipelines
- Hand-on Practice

Introduction of Pipeline Services

- Web UI
- Operating data and model on Git server

01 Data Automation

02 LLM Training

03 Model Evaluation

Data Automation

01

Data Generation

Generate dataset from seed dataset.

02

Data Evaluation

Score the quality of dataset.

Data Generation

Expand on minimal data

Pre Processing

Convert EXCEL, JSON, JSONL to compatible format.

Data Generation

Generate new data based on seed data with LLM.

Post Processing

Convert format, remove some metadata.

Data Distillation

Remove bad data, such as misspelling, not answering the question, broken grammar etc.

Data Convert

Convert to training ready format.

Data Evaluation

Typo-Free Score

- Detect errors like misspelled or misused words, mixing different languages, terms not used in Taiwan. Higher when less errors detected.

Perplexity (PPL) Score

- Lower when sentences are less fluent, with issues like wrong homophones or word order errors.

Diversity Score＆Redundant Score

- Greater global diversity and lower repetition in the dataset scored higher.
- Calculated with cosine distance.

Fine Tuning Pipeline

- Parameters optimized for hardwares to avoid OOM or other problems.
- Automatically converting model format to safetensors.
- Model files saved to Git repository.

Model Evaluation

Question Bank

- Get Response from Model

Scored by other Model

- Compare to Ground Truth
- Generate Text Report and Scores
- Format / Context

Post Processing

- Overall Score
- Other Details

Usage of Pipelines

Login with iService account

https://jenkins.genai.nchc.org.tw/

Select pipeline

- 01-data-automation:
  - 01-data-generation-NCHC
  - 02-data-evaluation
- 02-llm-training
- 03-model-evaluation

Run pipeline

- Click “Build with Parameters”
- Fill parameters
- Click “Build” and wait

Preparation

01

Git Repository:

- GitLab (Recommended)
- HuggingFace (Each file under 10 MiB)
- GitHub (Each file under 2GiB/5GiB)

02

LLM Access:

- Portal
- OpenAI (except data-evaluation)
- OpenAI Compatible (except data-evaluation)

Hand-on Practice

Data Automation

Path: Dashboard > 01-data-automation > **01-data-generation-NCHC**

Git_REPO_URL: GitLab/GitHub/HuggingFace repository url, with “http(s)://”

GIT_REPO_TOKEN: Token with R/W permission to the content of repository

- GitLab: Personal access tokens / Project access token
- GitHub: Settings > Developer Settings > Personal access tokens > Fine-grained tokens / Tokens (classic)
- HuggingFace: Access Tokens

DATA_FILE: Path to data, relative to the root directory of the repository

SHEET_NAME: If using EXCEL, specify which sheet contains the data

Data Automation

Path: Dashboard > 01-data-automation > **01-data-generation-NCHC**

GEN_MODEL: Model name that you want to use to generate data, depends on your LLM provider.

- *Usually, can get from $API_BASE/models endpoint*.

GEN_API_URL: URL of your LLM provider, can be any OpenAI compatible.

GEN_API_KEY: API key that has permission to use your GEN_MODEL

DISTILLATION_MODEL: Model name that you want to use for distillation

DISTILLATION_API_URL, DISTILLATION_API_KEY: As **GEN**

Data Automation

Path: Dashboard > 01-data-automation > **01-data-generation-NCHC**

TASK: Choose what kind of data you need

TOPIC: If the options in **TASK** do not meet your need, assign other topic

Q_COL: The key or column name of question/user input field in your data

A_COL: The key or column name of answer/assistant output field in you data

DEFAULT_COUNT: How many pairs you want to generate from a seed pair

SAMPLE: How many data you want to sample as seed data, set to 0 for use all

SYSTEM_MSG: Instruction for data generation

Data Automation

Path: Dashboard > 01-data-automation > **01-data-generation-NCHC**

DO_DEDUP: Do deduplication or not

SIMILARITY_THRESHOLD: Remove data if the similarity higher than this number

DO_DISTILLATION: Do distillation or not

Data Automation

Path: Dashboard > 01-data-automation > **02-data-evaluation**

REDUNDANCY_THRESHOLD: The value for calculating redundancy

LLM Training

Path: Dashboard > **02-llm training**

BASE_MODEL: Base model to train

MAX_EPOCHS: How many round to train

DEEPSPEED_ZERO_STAGE: Level of optimization, speed vs memory consuming

MAX_MODEL_LENGTH: Max content lenght, not longer than base model

MODEL_CONFIG_TORCH_DTYPE: Percision of trained model, depends on GPU,

- v100 supports fp16/32

HARDWARE_TYPE: Model of GPU, recently only v100

GPU_COUNTS: How many GPU to use in this training session

EMAIL_NOTIFY: True for email notification on pipeline finishing

Model Evaluation

Path: Dashboard > **03-model-evaluation**

GEN_MODEL_SOURCE: Choose the source of the model you want to evaluate, can be OpenAI, OpenAI compatible or NCHC provided.

GEN_MODEL, GEN_BASE_URL, GEN_API_KEY: Info to use model

JUDGE_MODEL_SOURCE: Choose the source of the model as the judge

JUDGE_MODEL, JUDGE_MODEL_URL, JUDGE_MODEL_KEY: Info to use model

EVAL_ITER: How many round to score

TASKS: The tasks to evaluate, depends on the purpose of using model

Model Evaluation

Path: Dashboard > **03-model-evaluation**

MAX_NEW_TOKENS: Max length for model's answer

BATCH_SIZE: Max number of async client interaction with LLM

NUM_ROWS: How many rows to evaluate

Example Data Format

https://gitlab.td.nchc.org.tw/baronhsu/llm-bootcamp-0618

THANK YOU

國家實驗研究院
NATIONAL INSTITUTES OF APPLIED RESEARCH