

DeepLearning, AI & UPSTAGE AI with Andrew Ng
* Multi-stage Continual Training

Pre-trained Model → Fine-Tuned Model - Aligned Model

In usual case : Pre-trained → Fine-Tuned Model

↳ use in specific domain

or

use their own data in
Fine-Tuned Model & Aligned Model

exceptional : Fine-Tuned & Aligned Model changing
(but quite often)

↓

Not enough.

↳ should change or give some dataset in
pretrained Model

↳ usually occur in new domain.

(not just changing topic, contexts)

↙

but : Same weights (parameters) : Cost & Time & Iterations : Pre-trained Model
was most expensive.

< Dataset for Pre-Training >

Kind of dataset : Unstructured text → ex) series of books, articles, whatever

↳ keep predicting next word

↳ while this phase : weights are updated

< Dataset for fine-tuning >

Kind of dataset : highly structured text

ex) question - answer pairs, instruction - response pairs etc.

↳ Pre-Training : Reading books)

Fine-Tuning : Testing, exams

↳ data should be very specific and precise.

Traditionally, performed by human. but right now, trying LLM to generate dataset.

***** Quality of data.

↳ correlates the performance of LLM.

↓

Data cleaning → Required. → Method :

↙

From UPSTAGE AI - Dataverse - open-source
AI

↙

Data-cleaning - SW.

1. Deduplication.

duplicate data could bias the model
in specific pattern.

2. Quality Filters

ex) Focusing on specific language.

3. Content Filters

→ No more bad words (don't let bias to toxic contents)

4. Privacy Protection & Rule-based cleaning

↳ Protect data leakage

↓
rule-based
cleaning
reverse poorly formatted
text

[Data - cleaning]

Form of

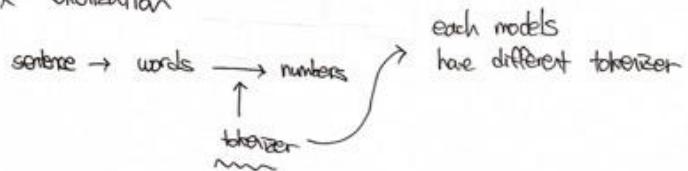
1. web crawling → Pre Training dataset.
2. Structured data → unstructured data (∵ Pre - Training Model)
3. Cut out rows which is too short (Under 3 words)
4. Paragraphs - duplicates / $\text{len}(\text{paragraph}) > 0.3$: remove
↳ deduplicating.
 - a) Using HuggingFace : We can use quite fine datasets.
↳ have done Reprocessing a lot.
5. language filter → specify to specific language.
 - a) Parquet : columnar storage file format
used in big data, big data analysis
(kind of csv)

[Data Packaging : Tokenizing + Packing]

Tokenizing : breaking ^{text} each into smaller meaningful units, which are called "tokens"

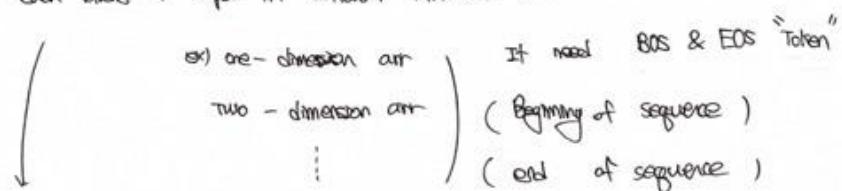
Packing : "Pack" tokens into the maximum sequence length to improve training efficiency

*** Tokenization



*** Packing

each blocks : shaped in different kind of DS



Training LLM: same length of data
is much efficient.

~~Repayage~~ model for training

This section was
about the way to customize, generate
the one LLM models.

c) In this course, only use(focus) on "decoder-only - Transformer"

Using this model, we'll predict the text and
let the model operate autoregressive model property.

autoregressive model: making output depends^{linearly} on its own
previous value (dataset)

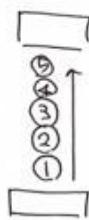
** decoder only Transformer model

: text → vector → weights layer

Making & Training Model: with random weights : Too much time & cost

↳ let's just use existing open-source SW's weights.

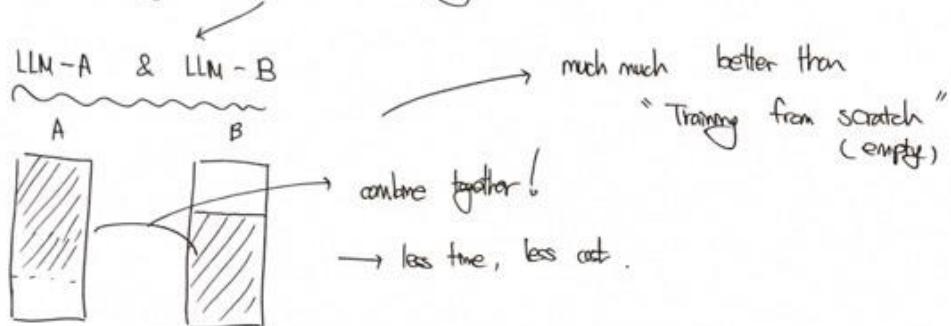
~~customizing~~ "Downscaling"
LLMs with



: removing some layers from

Big LLM. But in small models, it doesn't
work well.

~~Customizing~~ Customizing LLMs with Upgrading



cf) About "Model Size"

1. Number of Parameters = weight + bias

↳ trained Parameters.

2. Architecture = number of layers, number of nodes, and the way

they connected.

↳ deep neural network > shallow neural network.

3. Precision \Rightarrow model size.

The number of bits floating point.

4. Compression : retain the model's performance

Techniques. but minimize the model size

ex) pruning, quantization, knowledge distillation.

mobile phone, embedded system: need smaller model.

↳ model size is important component.

<Training Cycle>

- * 1. Data Prep
 - 2. Hyper parameter configuration
 - 3. Training
 - 4. Monitoring
- a lot of time would be take.
if you can use
Tranning Cluster to check
how It would be take.

Common Benchmark Dataset (Evaluating ML/DL model)

- ARC, MMLU, Hellaswag, TruthfulQA
 - Wingrade, GSM8k, MT Bench, EQ Bench
 - IIEval
- They evaluate specific part of Model
all ex) Language understanding, True or false, Mathematical Reasoning --