Science domain Pre-train and Instruction Dataset for LLMs

Seminar – Fall 2023

Min-kyun Ko

UST

KiSTi www.kisti.re.kr
Korea Institute of Science and Technology Information

CONTENTS

❖ Motivation

❖ LLM models are most popular recently, and there are many researches.

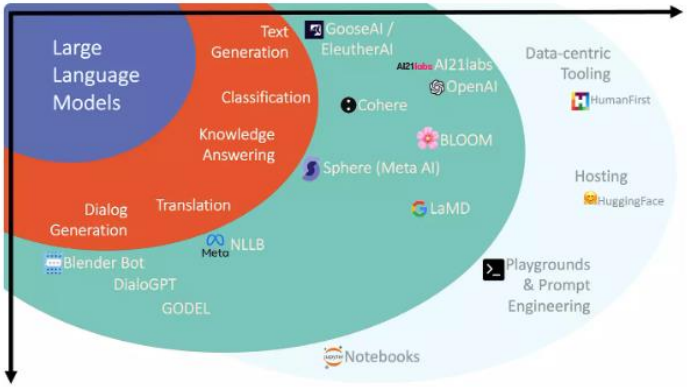


Figure1

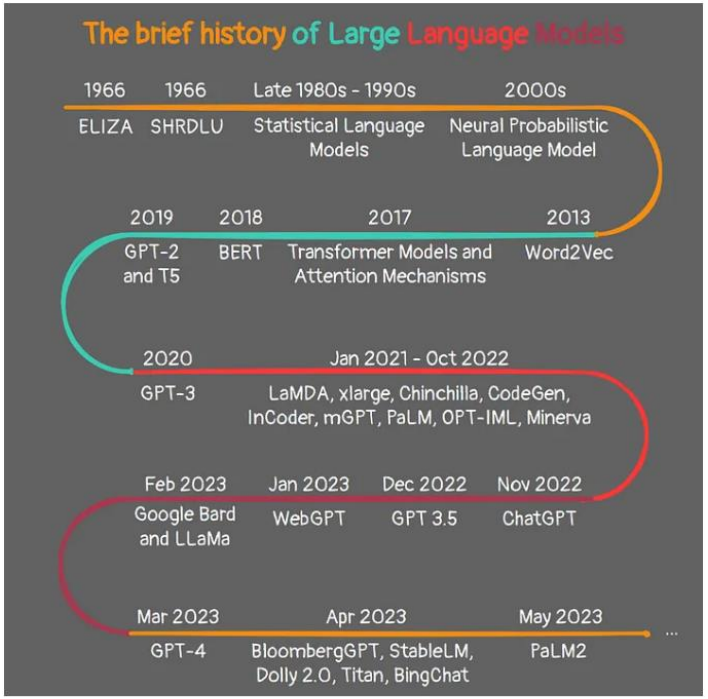


Figure 2

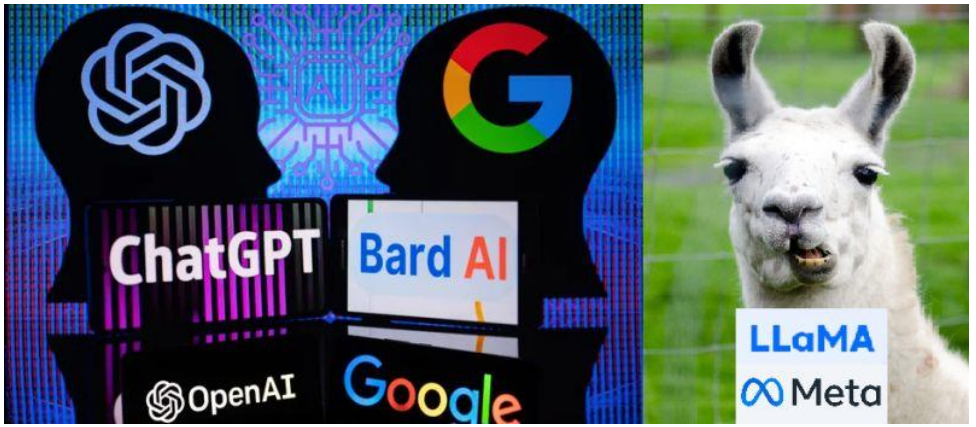


Figure 3

Figure 1 source : https://www.thedatahunt.com/en-insight/what-is-llm

Figure 2 source: https://levelup.gitconnected.com/the-brief-history-of-large-language-models-a-journey-from-eliza-to-gpt-4-and-google-bard-167c614af5af

Figure 3 source: https://bigdataanalyticsnews.com/open-source-alternatives-chatgpt-bard/

1. Recap

❖ Lack of Korean R&D domain dataset

인공지능 학습용 데이터 구축 · 활용 고도화 방안(요약)

Ⅰ 인공지능 학습용 데이터 구축 추진배경

□ 정부는 '**인공지능 국가전략**'('19.12), '**디지털 뉴딜**'('20.7)을 통한 디지털 전환 가속에 대비하며 대표 프로젝트로 '**데이터 댐**' **구축** 본격화('20.9~)

○ 전 산업·사회적인 **인공지능 도입·확산**의 **핵심자원**이 될 **인공지능 학습용 데이터 구축·개방**에 **'25년**까지 **2.5조원**의 **대규모 투자** 계획

※ '17~'19년 21종 → '25년까지 총 1,300여종 구축·개방 추진

< (참고) 인공지능 학습용 데이터 구축의 필요성 >

◇ **인공지능 모델**의 **데이터 학습량**은 **성능과 비례**하며, 각 분야로 인공지능 기술이 확산·발전되기 위해서는 **분야별 고품질·대규모 인공지능 학습용 데이터 확보**가 필수

◇ 단, **데이터 수집·가공**에 **시간·비용 소요**'가 커서 중소·스타트업, 대학 등의 **인공지능 도입·확산**에 **장벽**이 되므로, **국내 실정에 맞는 데이터**의 **양적·질적 확충** 요구 증대

* 국내 AI·데이터 기업은 AI 개발 시간의 80%, 비용의 75%가 데이터 확보에 소요된다고 응답('20, NIA)

➡ 지속 제기되는 **데이터 부족 문제 해소**와 **디지털 전환 가속화**를 위한 **체계적인 데이터 자원 확보·활용**을 뒷받침할 **전략방안 마련** 필요

Figure1

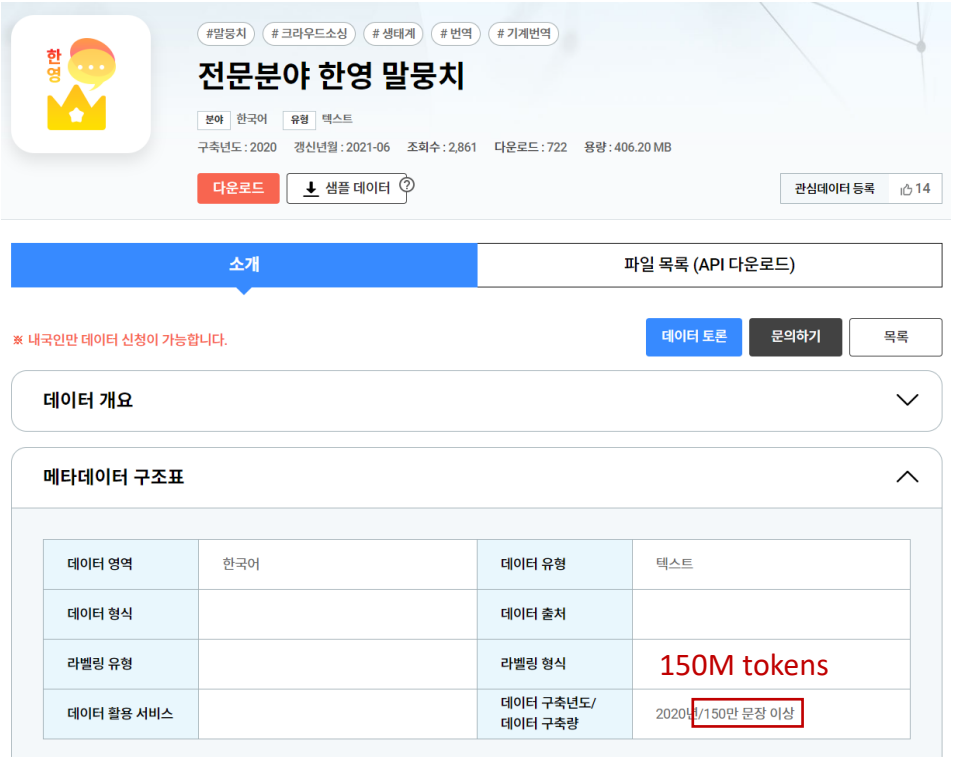


Figure2

Figure 1 source : 디지털 뉴딜의 성공을 위한 대표 프로젝트,인공지능(AI) 학습용 데이터 구축·활용 고도화 방안 (2022.01), https://snurnd.snu.ac.kr/?q=board/bd006/view/8157/download/11976

Figure 2 source : https://aihub.or.kr/aihubdata/data/view.do?currMenu=115&topMenu=100&aihubDataSe=realm&dataSetSn=111

1. Recap

❖ There are many Korean papers and reports as PDF in R&D domain

대상 데이터

구분	데이터 건수	범위
저자	467,574	-
논문	925,212	1948 - 2015
보고서	129,808	1900 - 2015
특허	370,024	2005 - 2014
저널	108,607	1541 - 2015
프로시딩	258,206	1541 - 2011
생물정보	2,231	자생종자식물, 고등균류
계	2,261,662	-

Figure1

Figure2

Figure 1 source : http://lod.ndsl.kr/home/intro/dataset.jsp

Figure 2 source : KISTI ScienceON, https://scienceon.kisti.re.kr/

1. Recap

❖ Nougat

Nougat: Neural Optical Understanding for Academic Documents

Lukas Blecher* **Guillem Cucurull** **Thomas Scialom** **Robert Stojnic**

Meta AI

Abstract

Scientific knowledge is predominantly stored in books and scientific journals, often in the form of PDFs. However, the PDF format leads to a loss of semantic information, particularly for mathematical expressions. We propose Nougat (**N**eural **O**ptical **U**nderstanding for **A**cademic Documen**t**s), a Visual Transformer model that performs an *Optical Character Recognition* (OCR) task for processing scientific documents into a markup language, and demonstrate the effectiveness of our model on a new dataset of scientific documents. The proposed approach offers a promising solution to enhance the accessibility of scientific knowledge in the digital age, by bridging the gap between human-readable documents and machine-readable text. We release the models and code to accelerate future work on scientific text recognition.

Figure10

❖ End – end architecture

❖ This method for academic documents.

Figure 10 source : Nougat: Neural Optical Understanding for Academic Documents (2023), https://doi.org/10.48550/arXiv.2308.13418

❖ End-end architecture

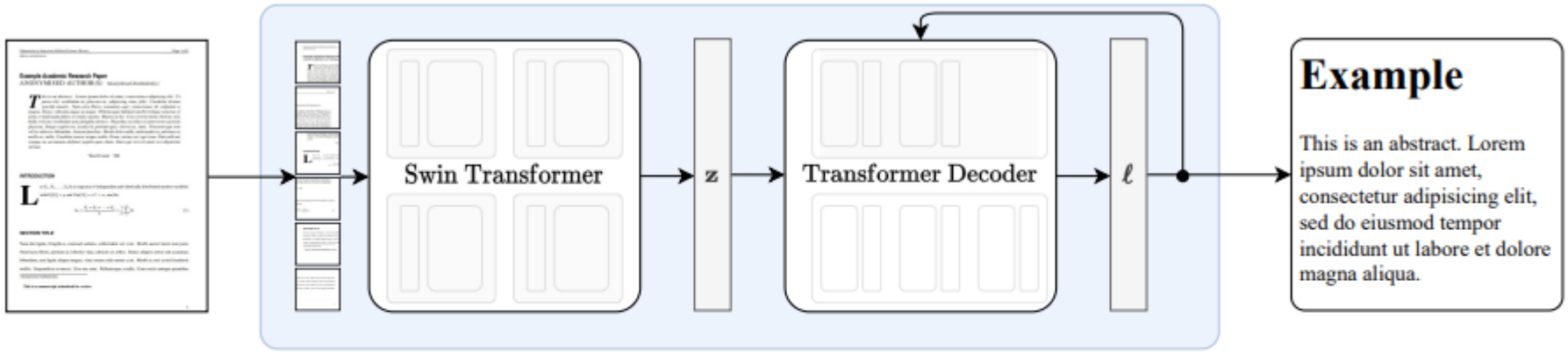


Figure 1: Our simple end-to-end architecture followin Donut [28]. The Swin Transformer encoder takes a document image and converts it into latent embeddings, which are subsequently converted to a sequence of tokens in a auto-regressive manner

Figure11

Figure 11 source : Nougat: Neural Optical Understanding for Academic Documents (2023), https://doi.org/10.48550/arXiv.2308.13418

Donut[28] in Figure 11 source : OCR-free Document Understanding Transformer (2021), https://doi.org/10.48550/arXiv.2111.15664

1. Recap

output values. These are concatenated and once again projected, resulting in the final values, as depicted in Figure 2.

Multi-head attention allows the model to jointly attend to information from different representation subspaces at different positions. With a single attention head, averaging inhibits this.

$$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, ..., \text{head}_\text{h})W^O$$
$$\text{where head}_\text{i} = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V)$$

Where the projections are parameter matrices $W_i^Q \in \mathbb{R}^{d_{\text{model}} \times d_k}$, $W_i^K \in \mathbb{R}^{d_{\text{model}} \times d_k}$, $W_i^V \in \mathbb{R}^{d_{\text{model}} \times d_v}$ and $W^O \in \mathbb{R}^{hd_v \times d_{\text{model}}}$.

In this work we employ $h = 8$ parallel attention layers, or heads. For each of these we use $d_k = d_v = d_{\text{model}}/h = 64$. Due to the reduced dimension of each head, the total computational cost is similar to that of single-head attention with full dimensionality.

3.2.3 Applications of Attention in our Model

The Transformer uses multi-head attention in three different ways:

- In "encoder-decoder attention" layers, the queries come from the previous decoder layer, and the memory keys and values come from the output of the encoder. This allows every position in the decoder to attend over all positions in the input sequence. This mimics the typical encoder-decoder attention mechanisms in sequence-to-sequence models such as [38, 2, 9].
- The encoder contains self-attention layers. In a self-attention layer all of the keys, values and queries come from the same place, in this case, the output of the previous layer in the encoder. Each position in the encoder can attend to all positions in the previous layer of the encoder.
- Similarly, self-attention layers in the decoder allow each position in the decoder to attend to all positions in the decoder up to and including that position. We need to prevent leftward information flow in the decoder to preserve the auto-regressive property. We implement this inside of scaled dot-product attention by masking out (setting to $-\infty$) all values in the input of the softmax which correspond to illegal connections. See Figure 2.

3.3 Position-wise Feed-Forward Networks

In addition to attention sub-layers, each of the layers in our encoder and decoder contains a fully connected feed-forward network, which is applied to each position separately and identically. This consists of two linear transformations with a ReLU activation in between.

$$\text{FFN}(x) = \max(0, xW_1 + b_1)W_2 + b_2 \quad (2)$$

While the linear transformations are the same across different positions, they use different parameters from layer to layer. Another way of describing this is as two convolutions with kernel size 1. The dimensionality of input and output is $d_{\text{model}} = 512$, and the inner-layer has dimensionality $d_{ff} = 2048$.

3.4 Embeddings and Softmax

Similarly to other sequence transduction models, we use learned embeddings to convert the input tokens and output tokens to vectors of dimension d_{model}. We also use the usual learned linear transformation and softmax function to convert the decoder output to predicted next-token probabilities. In our model, we share the same weight matrix between the two embedding layers and the pre-softmax linear transformation, similar to [30]. In the embedding layers, we multiply those weights by $\sqrt{d_{\text{model}}}$.

5

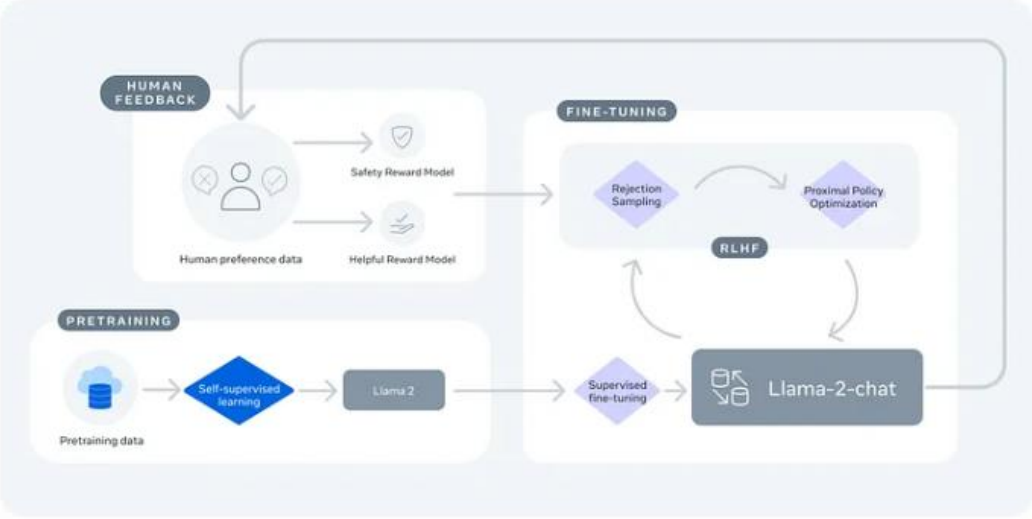

```
output values. These are concatenated and once again projected, resulting in the final values, as depicted in Figure 2.

Multi-head attention allows the model to jointly attend to information from different representation subspaces at different p

\[\mathrm{MultiHead}(Q,K,V) =\mathrm{Concat}(\mathrm{head}_{1},...,\mathrm{head}_{\mathrm{h}})W^ {O}\] \[\text{where }\mathrm

Where the projections are parameter matrices \(W_{i}^{Q}\in\mathbb{R}^{d_{\text{model}}\times d_{k}}\), \(W_{i}^{K}\in\mathbb

In this work we employ \(h=8\) parallel attention layers, or heads. For each of these we use \(d_{k}=d_{v}=d_{\text{model}}/h

#### 3.2.3 Applications of Attention in our Model

The Transformer uses multi-head attention in three different ways:

* In "encoder-decoder attention" layers, the queries come from the previous decoder layer, and the memory keys and values com
* The encoder contains self-attention layers. In a self-attention layer all of the keys, values and queries come from the sam
* Similarly, self-attention layers in the decoder allow each position in the decoder to attend to all positions in the decode

### Position-wise Feed-Forward Networks

In addition to attention sub-layers, each of the layers in our encoder and decoder contains a fully connected feed-forward ne

\[\mathrm{FFN}(x)=\max(0,xW_{1}+b_{1})W_{2}+b_{2} \tag{2}\]

While the linear transformations are the same across different positions, they use different parameters from layer to layer.

### Embeddings and Softmax

Similarly to other sequence transduction models, we use learned embeddings to convert the input tokens and output tokens to v
```

1. Recap

- Since **a lot of resources (GPU)** are required for training, we have limited time to use them.
- We need a dataset to train LLM right away.
- I tried to build a dataset through **Nougat**, but the first LLM model was decided to learn with the dataset we had.
- I refined and constructed the **Pre-train dataset**, and **instruction tuning dataset** that allows LLM to speak adding scientific knowledge.
  - The institution has a data set that has been **verified** in the domain of **science**.
  - Some of the institution's datasets can have a good effect on training. The dataset has also been added.

2. Technical Methods

❖ LLama2



Benchmarks

Llama 2 outperforms other open source language models on many external benchmarks, including reasoning, coding, proficiency, and knowledge tests.

Benchmark (Higher is better)	MPT (7B)	Falcon (7B)	Llama-2 (7B)	Llama-2 (13B)	MPT (30B)	Falcon (40B)	Llama-1 (65B)	Llama-2 (70B)
MMLU	26.8	26.2	45.3	54.8	46.9	55.4	63.4	68.9
TriviaQA	59.6	56.8	68.9	77.2	71.3	78.6	84.5	85.0
Natural Questions	17.8	18.1	22.7	28.0	23.0	29.5	31.0	33.0
GSM8K	6.8	6.8	14.6	28.7	15.2	19.6	50.9	56.8
HumanEval	18.3	N/A	12.8	18.3	25.0	N/A	23.7	29.9
AGIEval (English tasks only)	23.5	21.2	29.3	39.1	33.8	37.0	47.6	54.2
BoolQ	75.0	67.5	77.4	81.7	79.0	83.1	85.3	85.0
HellaSwag	76.4	74.1	77.2	80.7	79.9	83.6	84.2	85.3
OpenBookQA	51.4	51.6	58.6	57.0	52.0	56.6	60.2	60.2
QuAC	37.7	18.8	39.7	44.8	41.1	43.3	39.8	49.3
Winogrande	68.3	66.3	69.2	72.8	71.0	76.9	77.0	80.2

Figure source : https://ai.meta.com/llama/

2. Technical Methods

- Open and efficient foundation language models
- LLaMA is an **open-source**, **decoder-only LLMs** built upon the transformer architecture.
  - LLaMA also **incorporates** improvements utilized in different models, such as **pre-normalization**, **SwiGLU activation**, and **rotary embeddings**.
  - LLaMA is available in four different model sizes: **7B**, **13B**, **33B**, and **65B**.
  - LLaMA has been **pre-trained** with a **standard language modeling task** using **publicly available sources**, such as **crawled texts**, **books**, **Wikipedia**, and **preprint papers**.
  - LLaMA's **training set** consists of **roughly 1.4T tokens**, with the **majority in English** and a small fraction in **other European languages**.
  - Interestingly, our prior preliminary study reveals **that LLaMA exhibits basic Chinese understanding ability**, although **its capacity to generate Chinese texts is limited**.

Figure source : https://ai.meta.com/llama/

Chinese-LLaMA-Alpaca-2 v3.0 released long context LLMs (16K)

cn中文 | English | 文档/Docs | 提问/Issues | 讨论/Discussions | 竞技场/Arena

中文LLaMA & Alpaca大模型

Chinese LLaMA & Alpaca Large Language Models

To promote open research of large models in the Chinese NLP community, this project has open-sourced the **Chinese LLaMA model and the Alpaca large model with instruction fine-tuning**. These models expand the Chinese vocabulary based on the original LLaMA and use Chinese data for secondary pre-training, further enhancing Chinese basic semantic understanding. Additionally, the project uses Chinese instruction data for fine-tuning on the basis of the Chinese LLaMA, significantly improving the model's understanding and execution of instructions.

Technical Report (V2) : [Cui, Yang, and Yao, 2023] Efficient and Effective Text Encoding for Chinese LLaMA and Alpaca

Main contents of this project:

- Extended Chinese vocabulary on top of original LLaMA with significant encode/decode efficiency
- Open-sourced the Chinese LLaMA (general purpose) and Alpaca (instruction-tuned)
- Open-sourced the pre-training and instruction finetuning (SFT) scripts for further tuning on user's data
- Quickly deploy and experience the quantized version of the large model on CPU/GPU of your laptop (personal PC)
- Support transformers, llama.cpp, text-generation-webui, LlamaChat, LangChain, , privateGPT, etc.
- Released versions: 7B (basic, **Plus**, **Pro**), 13B (basic, **Plus**, **Pro**), 33B (basic, **Plus**, **Pro**)

News

[Aug 14, 2023] Chinese-LLaMA-Alpaca-2 v2.0 released. We open-source Chinese-LLaMA-2-13B and Chinese-Alpaca-2-13B. See https://github.com/ymcui/Chinese-LLaMA-Alpaca-2

[July 19, 2023] Release v5.0: Release Alpaca-Pro models, significantly improve generation quality. Along with Plus-33B models.

[July 19, 2023] We are launching Chinese-LLaMA-Alpaca-2 project.

[July 10, 2023] Beta channel preview, know coming updates in advance. See Discussion

[July 7, 2023] The Chinese-LLaMA-Alpaca family welcomes a new member: Visual Chinese-LLaMA-Alpaca model for visual question answering and chat. The 7B test version is available.

[June 30, 2023] 8K context support with llama.cpp. See Discussion. For 4K+ context support with transformers, see PR#705.

[June 16, 2023] Release v4.1: New technical report, add C-Eval inference script, add low-resource model merging script, etc.

[June 8, 2023] Release v4.0: LLaMA/Alpaca 33B versions are available. We also add privateGPT demo, C-Eval results, etc.

https://github.com/ymcui/Chinese-LLaMA-Alpaca-2

2. Technical Methods

- **Natural language processing** (NLP) field has **witnessed a substantial paradigm shift** with the **advent of Large Language Models** (LLMs).
  - Notably, the **GPT family** have garnered **significant attention**.
  - **ChatGPT**, evolved from **Instruct GPT**, serves as an advanced conversational AI model capable of conducting **context-aware**, **human-like interactions**.

- However, **as impactful as LLMs** have been, their implementation comes with **inherent limitations** that **hamper transparent** and **open research**.
  - A major concern is their proprietary nature, which **restricts access to the models**.
  - Furthermore, the **vast computational resources** necessary for training and deploying these models present a challenge for researchers with limited resources.

- To tackle these, **various open-source models** have been released.
  - **LLaMA** and **Alpaca** serve as notable examples of such initiatives.
  - **These open-source LLMs** are intended to facilitate academic research and accelerate progress within the NLP field.

https://github.com/ymcui/Chinese-LLaMA-Alpaca-2

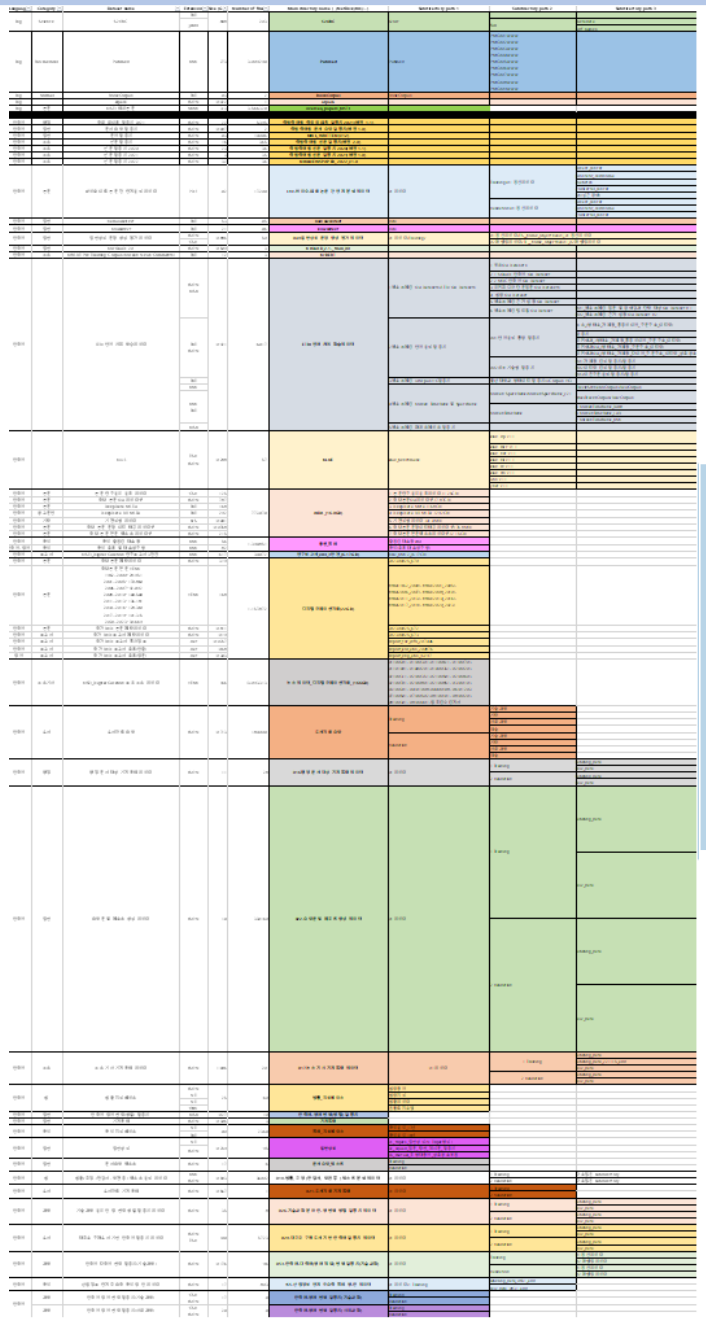
2. Technical Methods

- Despite the considerable strides made by **LLaMA** and **Alpaca** in NLP, they exhibit **inherent limitations** concerning native **support for Chinese language tasks**.
  - **Their vocabularies** contain **only a few hundred Chinese tokens**, substantially hindering their efficiency in encoding and decoding Chinese text.

- **Contributions** & **Solutions**:
  - We **enhance LLaMA's Chinese understanding ability** by **extending** the original **vocabulary** with an **additional 20,000 Chinese tokens**.
  - We employ the **Low-Rank Adaptation** (LoRA) approach to facilitate **efficient training and deployment** of the Chinese LLaMA and Alpaca models.
  - We **evaluate** the performance of the proposed LLaMA and Alpaca models in **instruction-following tasks** and **natural language understanding tasks.**
  - We make the resources and findings of **our study publicly available**, fostering further research and collaboration in the NLP community and **encouraging the adaptation** of LLaMA and Alpaca models **to other languages**.

https://github.com/ymcui/Chinese-LLaMA-Alpaca-2

3. Experiment



❖ Size : 1,995.01050 (GB)

❖ Num of Files : 23,619,937 (EA)

Category

- Bio-medical
- Normal
- Science
- 과학 (KR)
- 기타 (KR)
- 논문 (KR)
- 뉴스 (KR)
- 도서 (KR)
- 법 (KR)
- 보고서 (KR)
- 참고문헌 (KR)
- 특허 (KR)
- 행정 (KR)
- 일반 (KR)

3. Experiment

❖ English dataset


```
{
  "title": "Analyzing Relationships of Necessity Not Just  But Also",
  "abstract": "Analyzing relationships of necessity is important for both scholarly and applied research questions in the social sciences. An often-used technique for identifying such relationships– (fsQCA)–has limited ability to make the most out of the data used. The set-theoretical technique fsQCA makes statements  (e.g., “a condition or configuration is necessary or not for an outcome”), thereby ignoring the variation . We propose to apply a recently developed technique for identifying relationships of necessity that can make both statements  and , thus making full use of variation in the data:  (NCA). With its ability to also make statements  (“a specific level of a condition is necessary or not for a specific level of the outcome”), NCA can complement the  analysis of necessity with fsQCA.",
  "context": [
    {
      "title": "Introduction",
      "context": [
        "Identifying relationships of necessity is of key interest in the social sciences and beyond. Examples of work focusing on relationships of necessity include studies in policy science () and organizational sciences (). But “for any research area one can find important necessary condition hypotheses,” as the 150 examples from a large variety of areas in :65–66) testify. A condition, or variable, is necessary when the outcome does not exist without it (i.e., if  = 1 then  = 1) and the condition does not automatically produce the outcome (i.e., when  = 1, can be either 1 or 0). A necessary condition is a bottleneck, a constraint, for the outcome to exist. Identifying such conditions is thereby useful for both applied and fundamental research questions, because many questions concern the prerequisites for a particular outcome of interest (e.g., democracy, peace, economic growth, successful business performance, and sales performance). Necessary conditions that are relevant provide actionable knowledge that can “have very powerful policy implications” (:203).The technique that is nowadays often used for the analysis of necessary conditions that are beyond dichotomous, that is, conditions that can have other levels than just 0 (absent) or 1 (present), is  (fsQCA; , ; ). FsQCA is a set-theoretical technique based on fuzzy set theory and formal logic that can identify conditions or combinations of conditions (configurations) that are minimally sufficient and/or necessary for an outcome. The analysis of necessity in fsQCA focuses on identifying what we label necessary conditions . These are statements of the form: “ (either the presence or absence of a condition or configuration) is necessary for the presence or absence of .” Even though a fuzzy set itself has more levels, this results in “qualitative” statements about necessity.In this article, we argue that great strides can be made by not only analyzing such necessary relationships in kind, but also what we label necessary relationships . The latter make full use of the existing variation in fuzzy-set membership scores, allowing researchers to identify  [] is needed for  []. The latter results in quantitative statements about necessity. Such statements enable researchers to answer both applied research questions (such as what level of intelligence is required for a specific level of job performance?) and more fundamental ones (such as the research question of  seminal study: which level of economic development is necessary for a high level of democracy?). Necessary Condition Analysis [NCA] (), a recently developed technique for analyzing relationships of necessity that is also applicable to set-theoretical thinking, can answer precisely these kinds of questions. Complementing fsQCA with NCA yields results that are more precise or complete and can thereby contribute to theory development, theory testing, and/or offer policy advice or actionable knowledge. Complementing fsQCA with NCA is especially useful when the researcher uses fuzzy sets (as opposed to crisp sets) because the variation that is relevant goes beyond the distinction between “in” and “out” of a set. In those instances, largely ignoring the existing variation in degree is a missed opportunity. NCA takes this opportunity.Especially in fields in which fsQCA has a longer history (such as political science and sociology), many fsQCA applications follow so-called good practice by performing an analysis of necessity prior to a sufficiency analysis (:405). Still, the results of the sufficiency analysis typically form the study’s core. Also the methodological discussion on (fs)QCA concentrates on the sufficiency analysis, as for instance the Spring 2014 symposium on set-theoretical comparative methods (especially [fs]QCA) in the American Political Science Association Qualitative and Multimethod Research newsletter testifies, or the 2014 symposium on QCA in  on .There are a few exceptions to this focus on sufficiency in the QCA literature.  pays explicit attention to necessary conditions in his analysis of 24 journal publications using fsQCA in the fields of comparative politics, international relations, and sociology published between 2010 and 2013. Mello finds that 14 of the 18 studies testing for necessary conditions identified one or several of such conditions. Moreover,  pleaded for making the best of QCA possibilities when it comes to analyzing relationship of necessity. To this end, they introduced a new operation called systematic necessity assessment that enables the identification of what we label necessary OR-configurations (see below). In such configurations, either \n or \n is necessary for the outcome. For example, green apple (e.g., Granny Smith) OR red apple (e.g., Pink Lady) is necessary for making an apple pie. Bol and Luppi’s approach is a welcome contribution to the literature because of its multivariate nature (i.e., focusing on configurations of conditions). A final exception is  work on identifying so-called typical, deviant, and irrelevant cases after performing a QCA analysis of necessity. They propose to use the variation in set membership across cases to differentiate between these types of cases (e.g., :222). Our article adds to this body of research, taking up the call of :18) to place “(…) more emphasis [on] necessary conditions,” focusing especially on how to identify and evaluate them.This article is structured as follows. First, we briefly introduce how fsQCA analyzes relationships of necessity, whereby we also present a necessary condition typology (the second section). We then discuss how NCA analyzes relationships of necessity (the third section). Next, we present a reanalysis of data from a published article that theorizes–among other things–relationships of necessity (), using both fsQCA and NCA (the fourth section). Subsequently, we compare the findings of the two analyses (the fifth section). The final section draws conclusions (the sixth section)."
      ]
    },
    {
      "title": "Analyzing Relationships of Necessity with fsQCA",
      "context": [
        "Before we proceed, we first want to stress that we employ fsQCA as a data analysis  that identifies empirical patterns in the data. In addition to being a technique, QCA–in all its variants–is also a research  (). QCA as an approach includes an iterative process of data collection, from ideas to evidence and back; model specification; a holistic view of cases; case selection, and so on (see :378). Employing fsQCA as a technique means that we focus only on the so-called analytical moment () when cases have been selected and all conditions and the outcome have been calibrated (:379). We do so because NCA is mainly a data analysis technique, that is, focuses on this analytical moment, and presumes that meaningful data are available after proper case selection and measurement, and perhaps data transformation. Of course, this does not preclude scholars from employing (fs)QCA as an approach in their study as a whole, while using NCA for the analytical moment.Let us first address some issues regarding the type of and variation in the data that are used in both techniques. For both NCA and fsQCA, the data need to be reliable, valid, and–especially, but not exclusively, in the case of fsQCA–calibrated. Calibration (as used in QCA) is the transformation of what is typically called raw data (we prefer original data) into crisp sets or fuzzy sets. A fuzzy set is a “(…) a fine-grained, [pseudo] continuous measure that has been carefully calibrated using substantive and theoretical knowledge relevant to set membership” (:7). In fsQCA, three qualitative thresholds (fully in the set, fully out of the set, and neither in nor out of a set) are defined and quantified as 1, 0, and 0.5, respectively. Using these qualitative anchors, each case is scored quantitatively according to the degree of set membership (e.g., one case has 0.2 membership, another case has 0.4 membership). Hence, fsQCA captures variation across cases in degree. It also captures variation in kind by considering a case out of the set when scoring <0.5 and in the set when scoring >0.5. Fuzzy sets’ ability to also capture variation in set membership in degree is considered an important advantage over crisp set QCA, which can capture only variation in kind (fully out of the set [0] or fully in the set [1]; ).We propose to make full use of the variation in the degree of fuzzy-set membership scores in the analysis of necessity. NCA can do precisely this. Using fuzzy-set data in NCA allows for a more precise, or complete, interpretation of the necessary condition(s). For example, when using the verbal labels that can be attached to a fuzzy set membership score (almost fully in the set, more out than in of the set, etc.), the results of an NCA analysis can be interpreted for example as “being almost fully in the set (membership score of condition =0.8) is necessary for the outcome (membership score of the outcome ≥0.5),” or “being more out than in the set (membership score of condition =0.4) is necessary for an outcome that is almost fully in the set (membership score of the outcome =0.8).” This is particularly useful for research questions about what level of a condition is necessary for what level of the outcome (such as in the Lipset example above).Both techniques assume–in line with most data analysis techniques, including regression–that the condition or configuration () potentially causes the outcome (), in that  precedes  and could be related to  and that the scores of  and  are reliable, valid (and calibrated–see above). Note that while fsQCA can only be conducted on fuzzy-set data, NCA can be conducted on any type of data (including fuzzy sets) as long as the data are meaningful. To reveal better the differences and similarities between the analysis of necessity with fsQCA and NCA, we propose the following typology of relationships of necessity. The analysis of type 1 is a bivariate analysis (one condition and one outcome); the analysis of types 2 and 3 is a multivariate analysis (more than one condition and one outcome). In type 1, a condition is  for the outcome. Type 1 has two subtypes. Type 1A is the , that is, the condition is either absent (0) or present (1). Type 1B is the beyond-dichotomous necessary condition (, i.e., with a finite number of levels, or , i.e., an infinite number of levels). Type 2 is what we label the . In this configuration, \n and \n are each necessary, as is their AND-combination. For instance, apple AND flour each are individually necessary for an apple pie, making apple and flour a necessary AND-configuration (as are the other necessary ingredients for making an apple pie). Finally, type 3 is the , in which either \n or \n (green apple or red apple) is necessary for the outcome (apple pie). In the existing set-theoretical literature, the conditions constituting necessary OR-configurations are often labeled SUIN conditions: Sufficient but Unnecessary part of an Insufficient but Necessary configuration (:126). Here, we use the more general term necessary OR-configuration instead. There are three categories of necessary OR-configurations: (3A) configurations including redundant conditions (since every minimally necessary condition or configuration to which a condition is added will still be necessary), (3B) configurations of which the different components are direct logical equivalents (red or green apple), that is, can be captured by a higher-order construct (apple; ), and (3C) configurations of which the components are indirect logical equivalents (e.g., green apple or pentyl pentanoate–a chemical with apple scent–both of which may give a sensation of apple in an apple pie). A necessary OR-configuration that includes redundant conditions (3A) is not minimal and should therefore be avoided (see note 4). Necessary OR-configurations (3B) and (3C) could be useful for theory development or testing and for policy advice and are therefore worth identifying. Still, we consider single necessary conditions (type 1) and their AND-configurations (type 2) as more meaningful theoretically than the necessary-OR configurations (type 3), because types 1 and 2 refer to single ingredients that are true prerequisites–and thus truly necessary–for a particular outcome. Because NCA is geared toward exactly this–the truly single necessary condition(s)–currently NCA does not focus on necessary OR-configurations. FsQCA can identify all three types of necessary relationships to make in kind statements. An individual condition is either necessary for the presence or absence an outcome or not (type 1B); a necessary AND-configuration is necessary for an outcome or not (type 2); and a necessary OR-configuration is necessary for an outcome or not (type 3). Set-theoretically, a relationship of necessity exists if the fuzzy-set membership scores of  are at least as large as the membership scores of . The following relationship should thus hold:  ≤ . In an  plane or plot, which  displays, this is the case if all cases are on or below the diagonal.If researchers follow best practice, (fs)QCA starts with the bivariate analysis to identify any individually necessary conditions (type 1). When there are such conditions, it continues to identify if there are also necessary AND-configurations (type 2). When there are no individually necessary conditions (and thus also no necessary AND-configurations since these consist of individually necessary conditions), it identifies necessary OR-configurations (type 3).In . the subset relationship of necessity is
```


By Syed Hasnain Raza Shah

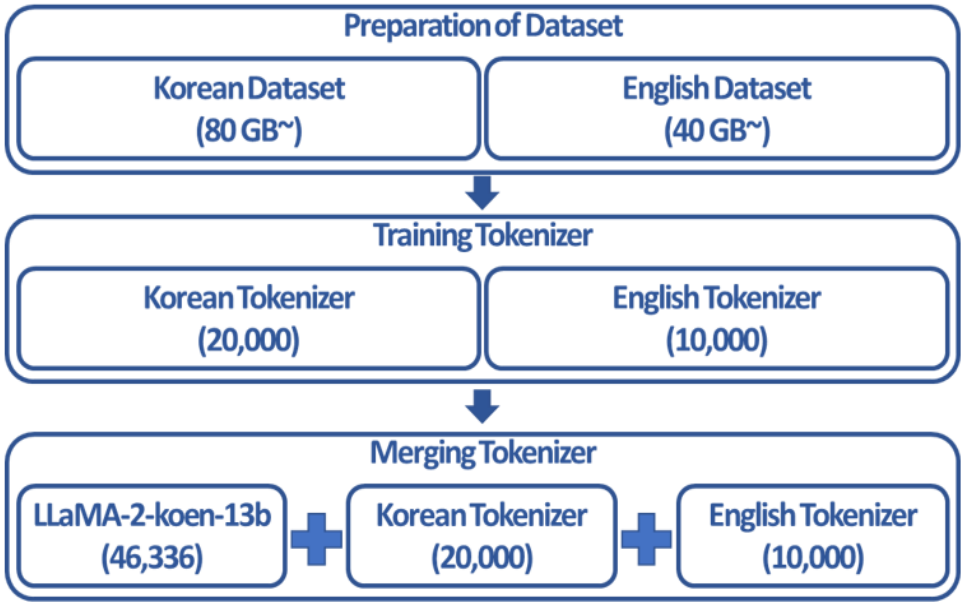
3. Experiment

❖ Pre-training

- ❖ Training Machine: 14 A100
- ❖ Base Model: llama-2-koen-13b
- ❖ Tokenizer: LLaMA-2-koen-SCI-13b
- ❖ Dataset:
  - Korean (89GB) + English (45GB)

- ❖ Merging Tokenizer
  - LLaMA-2-koen-13b Tokenizer
    - vocab_size=46,336
  - New_Tokenizer
    - Korean vocab_size=20,000
    - English vocab_size=10,000
  - LLaMA-2-koen-SCI-13b Tokenizer
    - vocab_size=56,252 (New_Tokens=9,916)

*Vocab_size of LLaMA2-13b: 32,000



3. Experiment

❖ Instruction tuning

Becaus of time limitation

❖ Instruction Tuning Dataset (4.1M)

- Summary 1.6M
- Research report QA 1M
- KISTI Paper QA (0.8M)
- QA (0.7M)
  - Code, Mathetics, Commonsence, etc.

❖ Instruction Tuning Dataset (2.5M)

- Research report QA 1M
- KISTI Paper QA (0.8M)
- QA (0.7M)
  - Code, Mathetics, Commonsence, etc.

Thank you