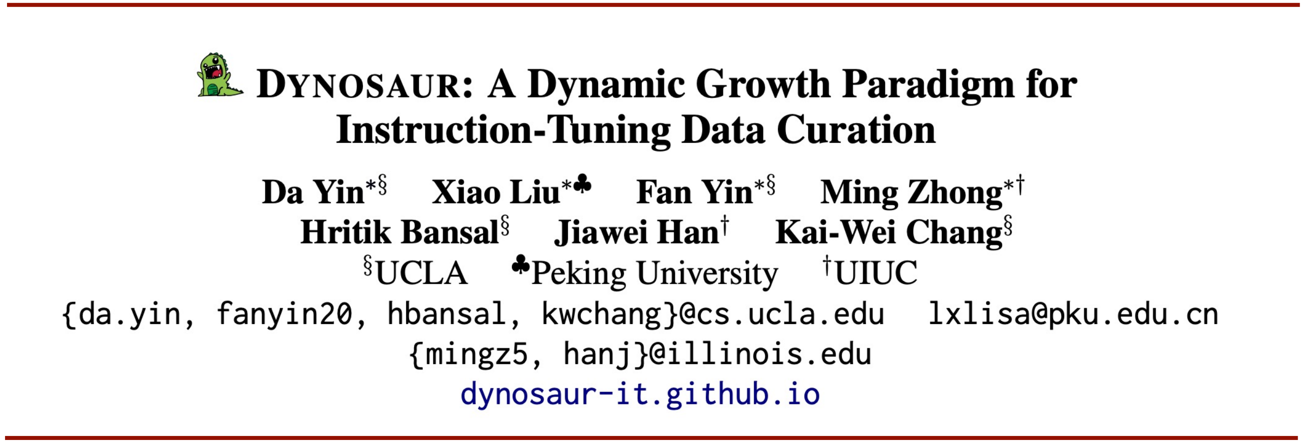
NLP&AI 겨울방학 세미나

Instruction 데이터 확장

2024. 02. 29.

발표자 장윤나



DYNOSAUR: A Dynamic Growth Paradigm for Instruction-Tuning Data Curation

Da Yin[*§] Xiao Liu[*♣] Fan Yin[*§] Ming Zhong[*†]
Hritik Bansal[§] Jiawei Han[†] Kai-Wei Chang[§]
[§]UCLA [♣]Peking University [†]UIUC
{da.yin, fanyin20, hbansal, kwchang}@cs.ucla.edu lxlisa@pku.edu.cn
{mingz5, hanj}@illinois.edu

dynosaur-it.github.io

EMNLP 2023

DYNOSAUR: A Dynamic Growth Paradigm for Instruction-Tuning Data Curation

1. Introduction

- Instruction-tuning dataset을 만들고자 하는 시도

  1. Manual annotation: 높은 품질, 그러나 **labor-intensive & costly**
  2. Distillate from a larger LLM: teacher LLM의 능력(factuality, problem solving skills)을 받기보단 모방

- Dataset repository의 annotation을 instruction tuning data로 만드는 DYNOSAUR 제안
- HuggingFace Dataset Platform🤗 의 메타데이터 활용
  - 데이터셋에 대한 description, name, data fields, annotation 등

Given a Gutenburg passage, generate its title

Predict the year when the book is published based on book title and authors

1. Introduction

Contributions

- Lower conversion cost
  - $11.5 for 800k instruction-tuning data ($500 for 52k instances – ALPACA, Instruction GPT-4)
- Effectiveness of instruction-tuning data
  - SUPER-NI에 대해 T5-3B, LLaMA-7B 모두 DYNOSAUR에 학습된 모델이 학습 데이터가 더 비싼 ALPACA, INSTRUCTION GPT-4, DOLLY보다 좋은 성능
- Supporting continuously improving models with new instruction data
  - 계속 업데이트 되는 많은 양의 데이터를 low cost (gpt-3.5-turbo) 덕분에 적극 활용이 가능
  - K개의 태스크에 학습된 모델과 새로운 L개의 태스크가 있을 때 generalization 능력과 forgetting을 줄이는 continual 학습 방법 제안

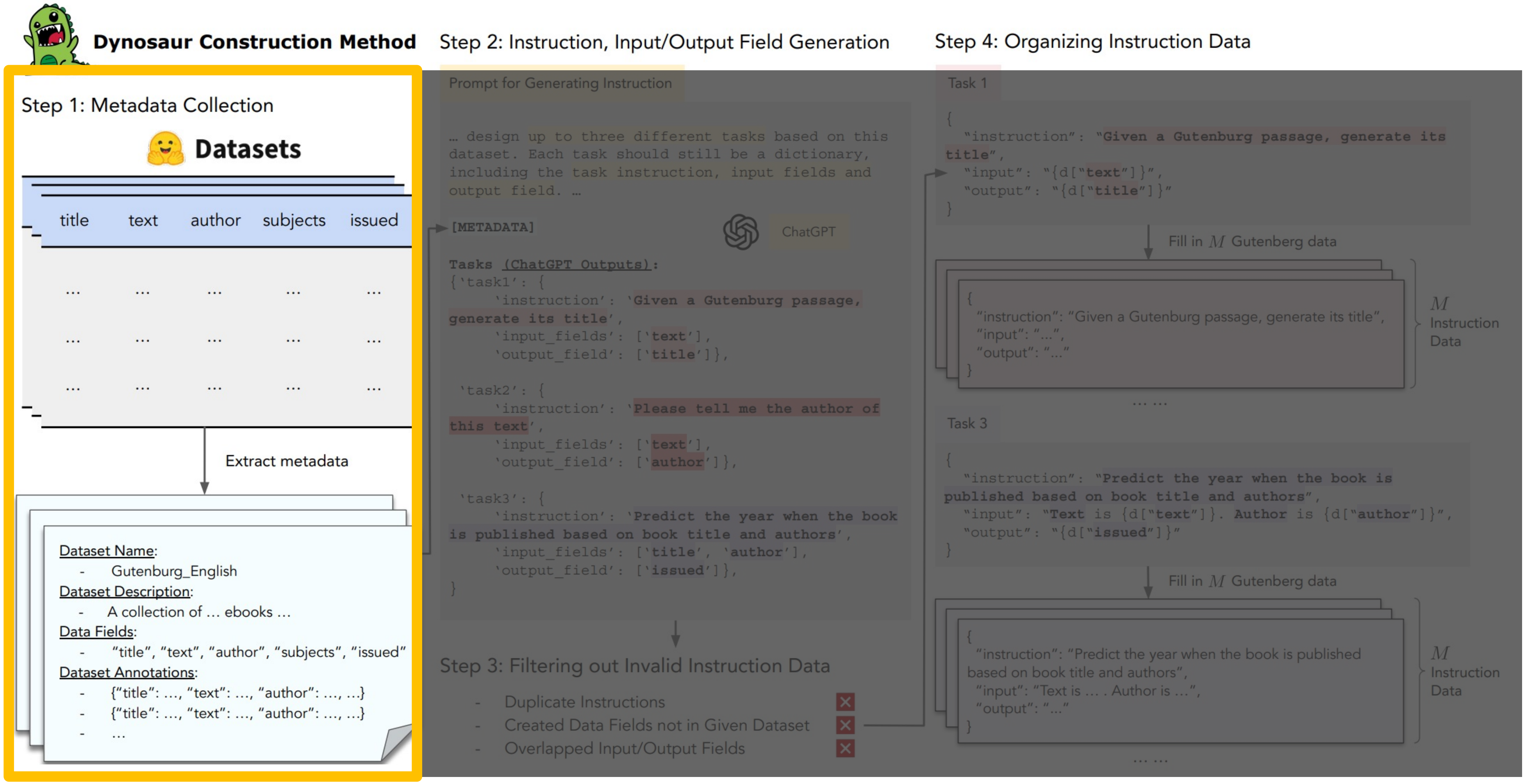
2. Collection of DYNOSAUR Data

Metadata Collection

- Dataset name: 이름에 도메인 혹은 태스크 정보가 들어가 있기도 함
- Dataset description: 자세한 정보, motivation, summary 등이 들어가 있기에 LLM이 instruction 만들 때 도움이 될 것으로 예상
- Data fields and annotations: data fields는 annotation의 key로 저장되어 있음. LLM은 어떤 field를 input/output으로 사용할지 결정해야 함

```
{
"title":…,
"text":…,
"author":…,
"subjects":…,
"issued":…
}
```

DYNOSAUR: A Dynamic Growth Paradigm for Instruction-Tuning Data Curation



Dynosaur Construction Method
Step 1: Metadata Collection
Datasets
Extract metadata
Step 2: Instruction, Input/Output Field Generation
Prompt for Generating Instruction
ChatGPT
Step 3: Filtering out Invalid Instruction Data
Step 4: Organizing Instruction Data
Task 1
Fill in M Gutenberg data
M Instruction Data
Task 3
Fill in M Gutenberg data
M Instruction Data
… …

title	text	author	subjects	issued
…	…	…	…	…
…	…	…	…	…
…	…	…	…	…

Dataset Name:
- Gutenburg_English

Dataset Description:
- A collection of … ebooks …

Data Fields:
- "title", "text", "author", "subjects", "issued"

Dataset Annotations:
- {"title": …, "text": …, "author": …, …}
- {"title": …, "text": …, "author": …, …}
- …

```
… design up to three different tasks based on this
dataset. Each task should still be a dictionary,
including the task instruction, input fields and
output field. …

[METADATA]

Tasks (ChatGPT Outputs):
{'task1': {
      'instruction': 'Given a Gutenburg passage,
generate its title',
      'input_fields': ['text'],
      'output_field': ['title']},

 'task2': {
      'instruction': 'Please tell me the author of
this text',
      'input_fields': ['text'],
      'output_field': ['author']},

 'task3': {
      'instruction': 'Predict the year when the book
is published based on book title and authors',
      'input_fields': ['title', 'author'],
      'output_field': ['issued']},
}
```

- Duplicate Instructions ☒
- Created Data Fields not in Given Dataset ☒
- Overlapped Input/Output Fields ☒

```
{
  "instruction": "Given a Gutenburg passage, generate its
title",
  "input": "{d["text"]}",
  "output": "{d["title"]}"
}
```

```
{
  "instruction": "Given a Gutenburg passage, generate its title",
  "input": "…",
  "output": "…"
}
```

```
{
  "instruction": "Predict the year when the book is
published based on book title and authors",
  "input": "Text is {d["text"]}. Author is {d["author"]}",
  "output": "{d["issued"]}"
}
```

```
{
  "instruction": "Predict the year when the book is published
based on book title and authors",
  "input": "Text is … . Author is …",
  "output": "…"
}
```

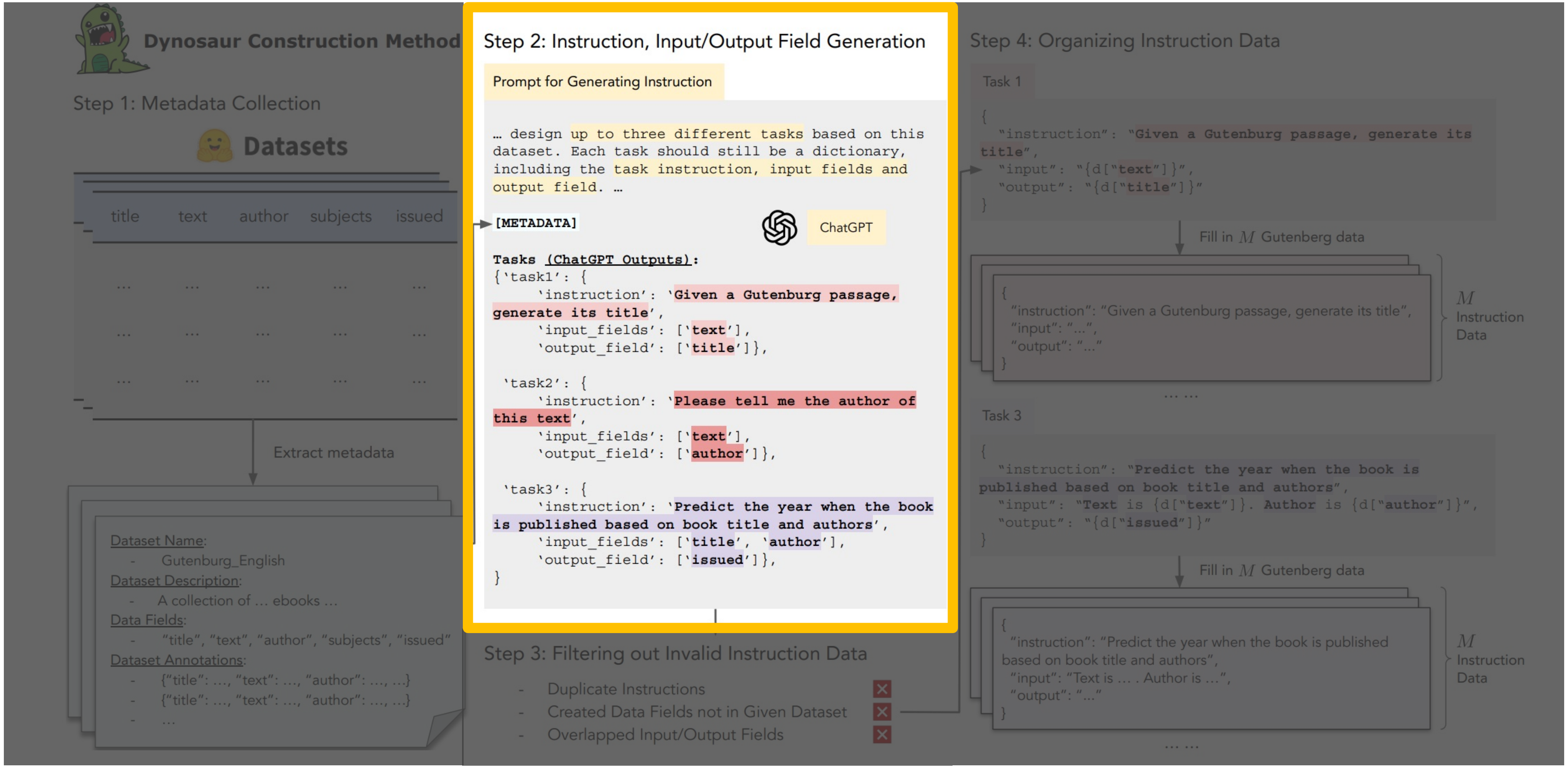
Figure 1: Overall pipeline of collecting DYNOSAUR data. "d" in Step 4 means each instance in Gutenberg dataset.

2. Collection of DYNOSAUR Data

Instruction and input/output field generation

- LLM은 **task instruction**과 **input, output field**를 생성해야 함
- In context learning이용:
  - 각 데이터셋을 dictionary 형태로 주고, 4개의 예시를 직접 만든 후 그 중 2개를 보여줌
- Description(데이터,태스크 제작 의도를 포함)을 줄지 말지에 따라 2개의 세팅
  - Description-aware generation: 데이터셋의 원래 의도에 맞춘 태스크를 생성할 수 있도록 하기 위함
  - Description-unaware generation: input/output field를 활용해 다양한 potential 태스크를 생성할 수 있도록 하기 위함

DYNOSAUR: A Dynamic Growth Paradigm for Instruction-Tuning Data Curation


Dynosaur Construction Method
Step 1: Metadata Collection
Datasets
title
text
author
subjects
issued
Extract metadata
Step 2: Instruction, Input/Output Field Generation
Prompt for Generating Instruction
ChatGPT
Step 4: Organizing Instruction Data
Task 1
Fill in M Gutenberg data
M Instruction Data
Task 3
Fill in M Gutenberg data
M Instruction Data

Dataset Name:
- Gutenberg_English

Dataset Description:
- A collection of … ebooks …

Data Fields:
- “title”, “text”, “author”, “subjects”, “issued”

Dataset Annotations:
- {“title”: …, “text”: …, “author”: …, …}
- {“title”: …, “text”: …, “author”: …, …}
- …

```
… design up to three different tasks based on this
dataset. Each task should still be a dictionary,
including the task instruction, input fields and
output field. …

[METADATA]

Tasks (ChatGPT Outputs):
{'task1': {
     'instruction': 'Given a Gutenberg passage,
generate its title',
     'input_fields': ['text'],
     'output_field': ['title']},

 'task2': {
     'instruction': 'Please tell me the author of
this text',
     'input_fields': ['text'],
     'output_field': ['author']},

 'task3': {
     'instruction': 'Predict the year when the book
is published based on book title and authors',
     'input_fields': ['title', 'author'],
     'output_field': ['issued']},
}
```

Step 3: Filtering out Invalid Instruction Data

- Duplicate Instructions ☒
- Created Data Fields not in Given Dataset ☒
- Overlapped Input/Output Fields ☒

```
{
  "instruction": "Given a Gutenberg passage, generate its
title",
  "input": "{d["text"]}",
  "output": "{d["title"]}"
}
```

```
{
  "instruction": "Given a Gutenberg passage, generate its title",
  "input": "...",
  "output": "..."
}
```

… …

```
{
  "instruction": "Predict the year when the book is
published based on book title and authors",
  "input": "Text is {d["text"]}. Author is {d["author"]}",
  "output": "{d["issued"]}"
}
```

```
{
  "instruction": "Predict the year when the book is published
based on book title and authors",
  "input": "Text is … . Author is …",
  "output": "..."
}
```

… …

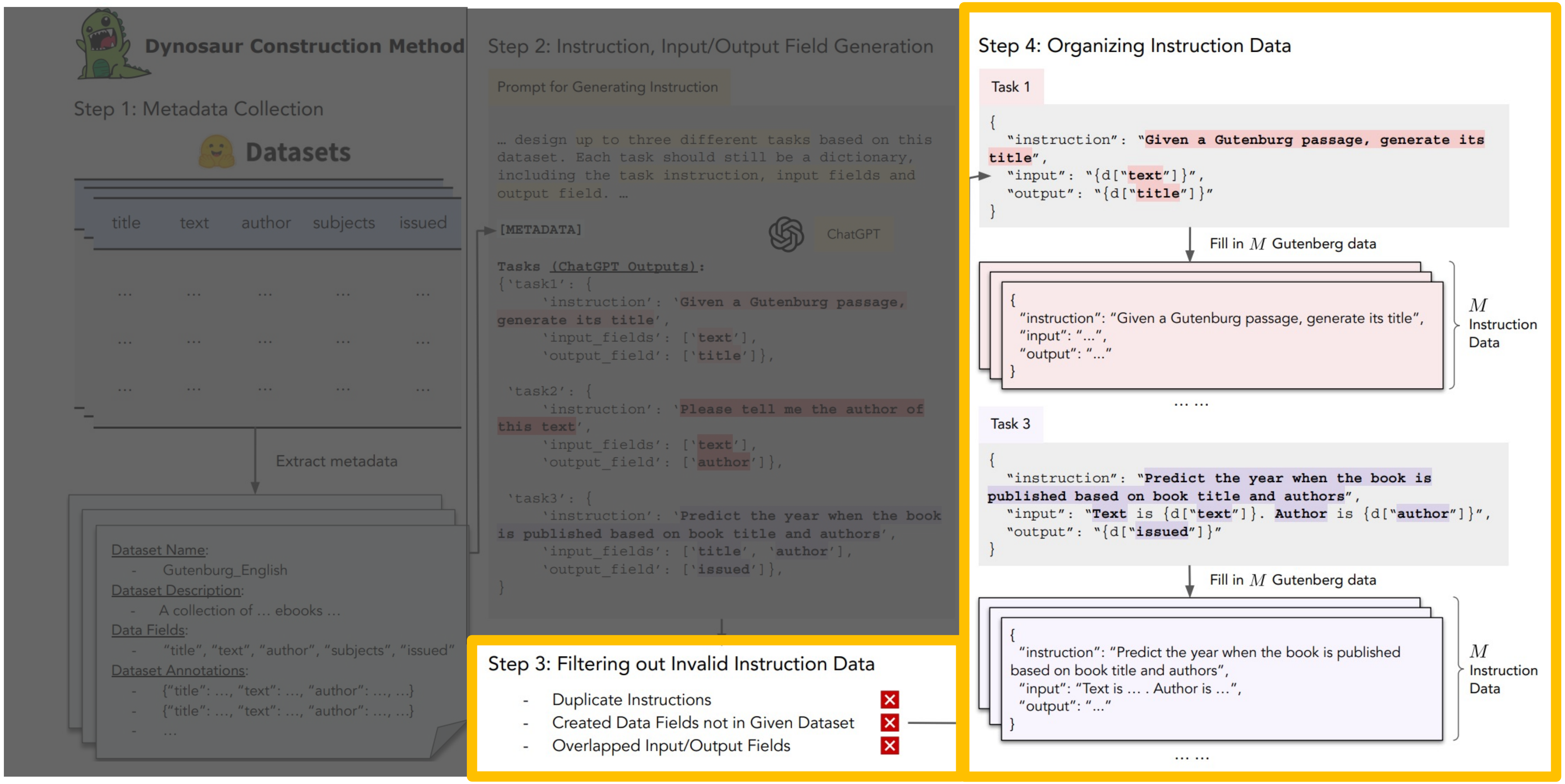
Figure 1: Overall pipeline of collecting DYNOSAUR data. “d” in Step 4 means each instance in Gutenberg dataset.

2. Collection of DYNOSAUR Data

Post-processing

- Filtering invalid tasks:
  - 1) 존재 X인 data field의 태스크인 경우, 2)1개 이상의 output field 가진 경우, 3) input/output 필드 겹치는 경우
  - Description-aware & -unware 에서 태스크 겹치는 경우 중복 생략
- Organizing instruction data
  - Input에 1개의 field만 있는 경우 그 value가 그대로 쓰이며 2개 이상인 경우 "`The [field name] is [value of the field].`"형태로 쓰임

DYNOSAUR: A Dynamic Growth Paradigm for Instruction-Tuning Data Curation



Dataset Name:
- Gutenburg_English

Dataset Description:
- A collection of … ebooks …

Data Fields:
- “title”, “text”, “author”, “subjects”, “issued”

Dataset Annotations:
- {“title”: …, “text”: …, “author”: …, …}
- {“title”: …, “text”: …, “author”: …, …}
- …

Step 2: Instruction, Input/Output Field Generation

Prompt for Generating Instruction

```
… design up to three different tasks based on this
dataset. Each task should still be a dictionary,
including the task instruction, input fields and
output field. …

[METADATA]
```

```
Tasks (ChatGPT Outputs):
{'task1': {
     'instruction': 'Given a Gutenburg passage,
generate its title',
     'input_fields': ['text'],
     'output_field': ['title']},

 'task2': {
     'instruction': 'Please tell me the author of
this text',
     'input_fields': ['text'],
     'output_field': ['author']},

 'task3': {
     'instruction': 'Predict the year when the book
is published based on book title and authors',
     'input_fields': ['title', 'author'],
     'output_field': ['issued']},
}
```

Step 3: Filtering out Invalid Instruction Data

- Duplicate Instructions ☒
- Created Data Fields not in Given Dataset ☒
- Overlapped Input/Output Fields ☒

Step 4: Organizing Instruction Data

Task 1

```
{
  "instruction": "Given a Gutenburg passage, generate its
title",
  "input": "{d["text"]}",
  "output": "{d["title"]}"
}
```

```
{
  “instruction”: “Given a Gutenburg passage, generate its title”,
  “input”: “…”,
  “output”: “…”
}
```

Task 3

```
{
  "instruction": "Predict the year when the book is
published based on book title and authors",
  "input": "Text is {d["text"]}. Author is {d["author"]}",
  "output": "{d["issued"]}"
}
```

```
{
  “instruction”: “Predict the year when the book is published
based on book title and authors”,
  “input”: “Text is … . Author is …”,
  “output”: “…”
}
```

Figure 1: Overall pipeline of collecting DYNOSAUR data. “d” in Step 4 means each instance in Gutenberg dataset.

DYNOSAUR: A Dynamic Growth Paradigm for Instruction-Tuning Data Curation

2. Collection of DYNOSAUR Data

Statistics and Cases

- 2,911 English datasets from HuggingFace (Feb 23, 2023)
- GPT-3.5-turbo에 넣고 13,610 개의 태스크를 생성하도록 함
- 각 태스크마다 200개의 sample을 뽑아 총 801,900 개의 샘플로 구성된 DYNOSAUR 데이터셋

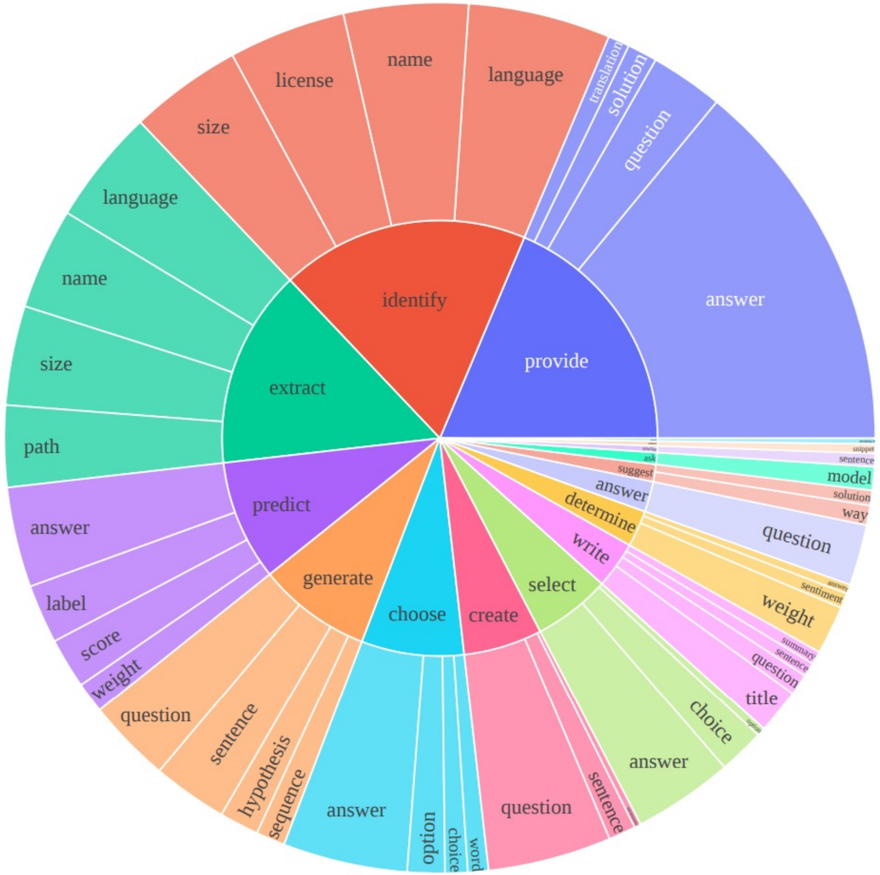
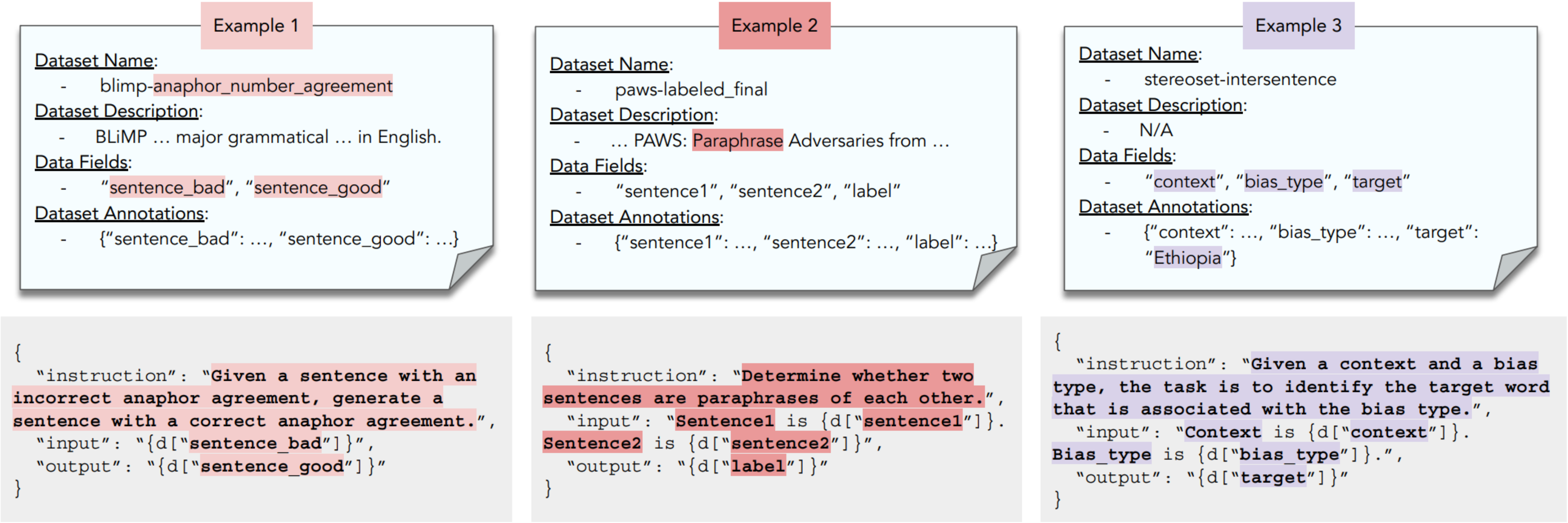


Figure 3: The top 20 most prevalent root verbs and their top 4 direct nouns in the instructions of DYNOSAUR.

2. Collection of DYNOSAUR Data



Example 1

Dataset Name:
- blimp-anaphor_number_agreement

Dataset Description:
- BLiMP … major grammatical … in English.

Data Fields:
- "sentence_bad", "sentence_good"

Dataset Annotations:
- {"sentence_bad": …, "sentence_good": …}

```
{
  "instruction": "Given a sentence with an
incorrect anaphor agreement, generate a
sentence with a correct anaphor agreement.",
  "input": "{d["sentence_bad"]}",
  "output": "{d["sentence_good"]}"
}
```

Example 2

Dataset Name:
- paws-labeled_final

Dataset Description:
- … PAWS: Paraphrase Adversaries from …

Data Fields:
- "sentence1", "sentence2", "label"

Dataset Annotations:
- {"sentence1": …, "sentence2": …, "label": …}

```
{
  "instruction": "Determine whether two
sentences are paraphrases of each other.",
  "input": "Sentence1 is {d["sentence1"]}.
Sentence2 is {d["sentence2"]}",
  "output": "{d["label"]}"
}
```

Example 3

Dataset Name:
- stereoset-intersentence

Dataset Description:
- N/A

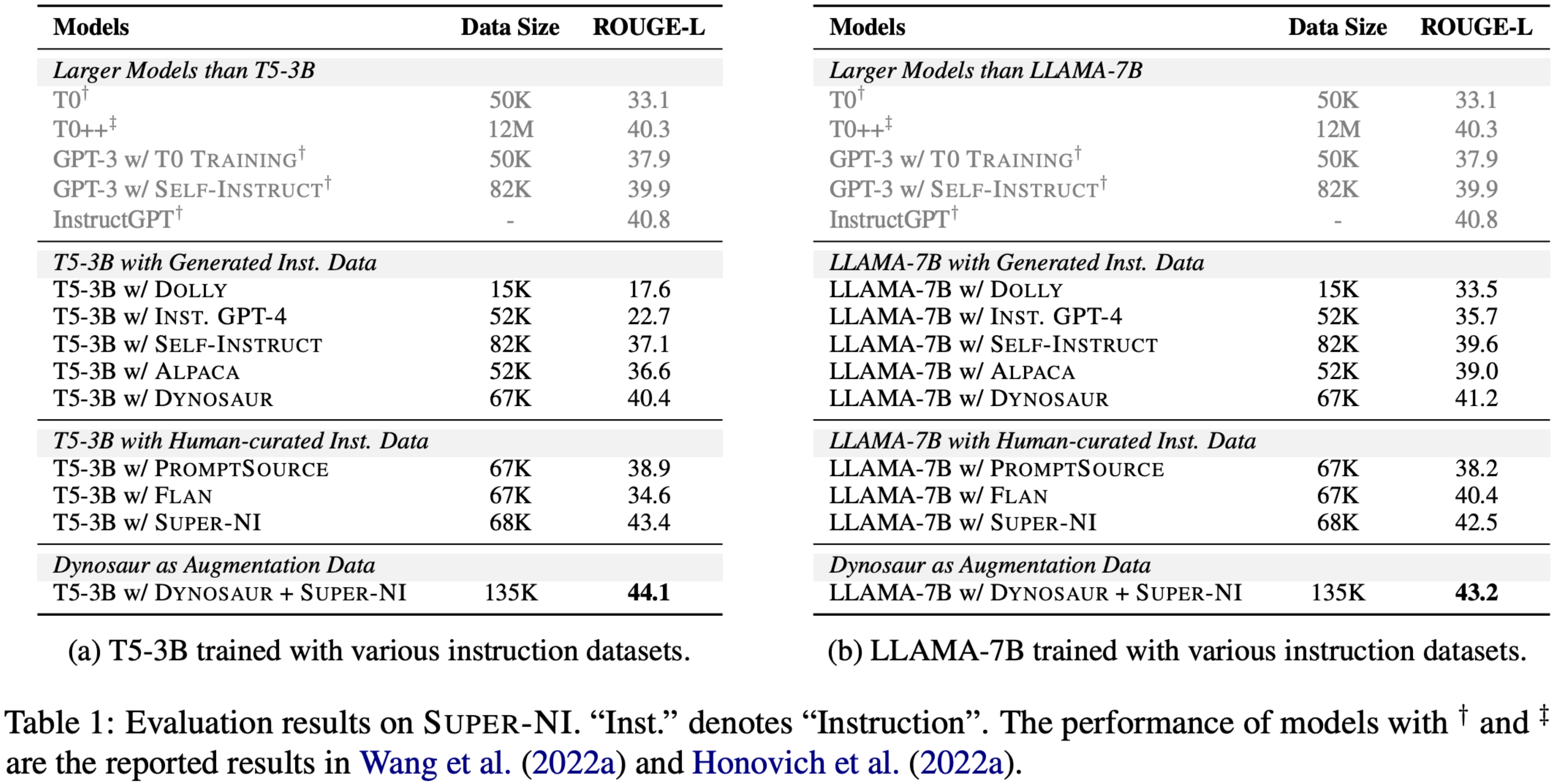
Data Fields:
- "context", "bias_type", "target"

Dataset Annotations:
- {"context": …, "bias_type": …, "target": "Ethiopia"}

```
{
  "instruction": "Given a context and a bias
type, the task is to identify the target word
that is associated with the bias type.",
  "input": "Context is {d["context"]}.
Bias_type is {d["bias_type"]}.",
  "output": "{d["target"]}"
}
```

Figure 2: Examples of the datasets and generated tasks. We only demonstrate one task based on each dataset for simplicity. We highlight the parts in metadata that benefit instruction generation.

3. Experiments



Models	Data Size	ROUGE-L
Larger Models than T5-3B		
T0[†]	50K	33.1
T0++[‡]	12M	40.3
GPT-3 w/ T0 Training[†]	50K	37.9
GPT-3 w/ Self-Instruct[†]	82K	39.9
InstructGPT[†]	-	40.8
T5-3B with Generated Inst. Data		
T5-3B w/ Dolly	15K	17.6
T5-3B w/ Inst. GPT-4	52K	22.7
T5-3B w/ Self-Instruct	82K	37.1
T5-3B w/ Alpaca	52K	36.6
T5-3B w/ Dynosaur	67K	40.4
T5-3B with Human-curated Inst. Data		
T5-3B w/ PromptSource	67K	38.9
T5-3B w/ Flan	67K	34.6
T5-3B w/ Super-NI	68K	43.4
Dynosaur as Augmentation Data		
T5-3B w/ Dynosaur + Super-NI	135K	**44.1**

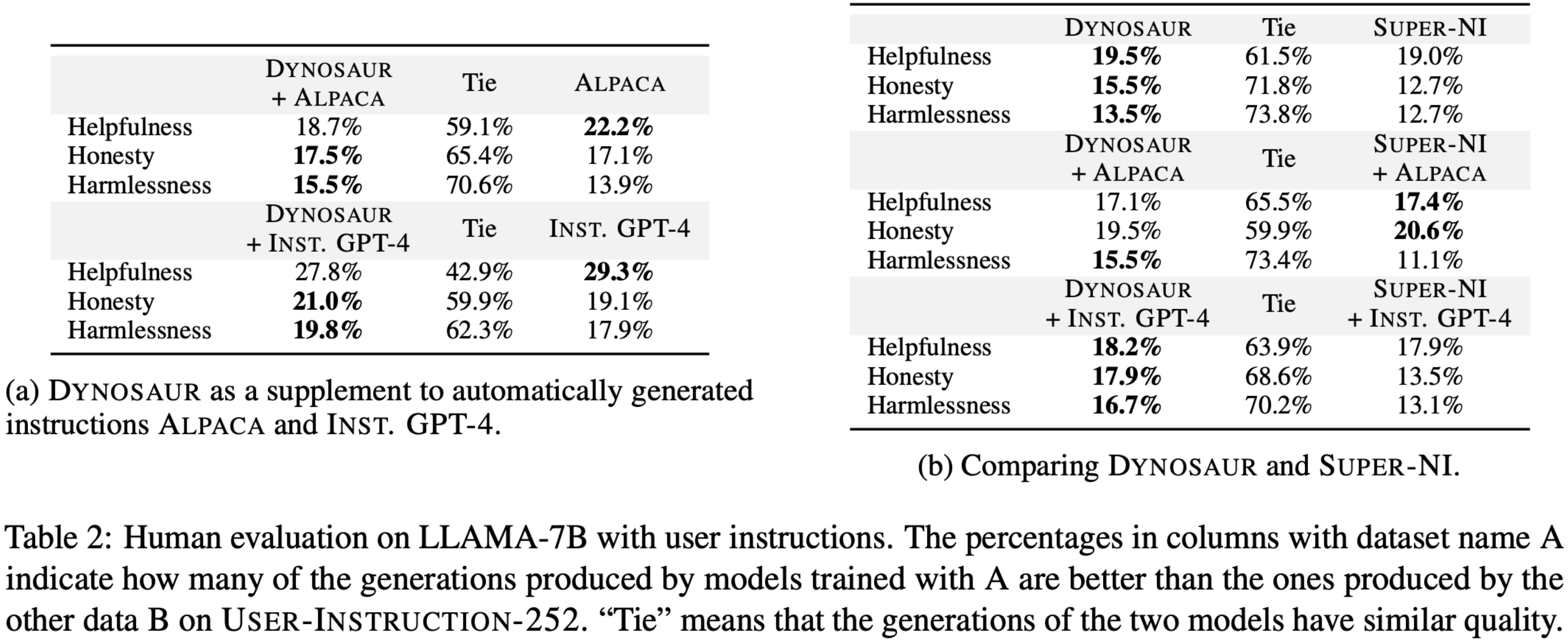
(a) T5-3B trained with various instruction datasets.

Models	Data Size	ROUGE-L
Larger Models than LLAMA-7B		
T0[†]	50K	33.1
T0++[‡]	12M	40.3
GPT-3 w/ T0 Training[†]	50K	37.9
GPT-3 w/ Self-Instruct[†]	82K	39.9
InstructGPT[†]	-	40.8
LLAMA-7B with Generated Inst. Data		
LLAMA-7B w/ Dolly	15K	33.5
LLAMA-7B w/ Inst. GPT-4	52K	35.7
LLAMA-7B w/ Self-Instruct	82K	39.6
LLAMA-7B w/ Alpaca	52K	39.0
LLAMA-7B w/ Dynosaur	67K	41.2
LLAMA-7B with Human-curated Inst. Data		
LLAMA-7B w/ PromptSource	67K	38.2
LLAMA-7B w/ Flan	67K	40.4
LLAMA-7B w/ Super-NI	68K	42.5
Dynosaur as Augmentation Data		
LLAMA-7B w/ Dynosaur + Super-NI	135K	**43.2**

(b) LLAMA-7B trained with various instruction datasets.

Table 1: Evaluation results on Super-NI. "Inst." denotes "Instruction". The performance of models with † and ‡ are the reported results in Wang et al. (2022a) and Honovich et al. (2022a).

3. Experiments



	DYNOSAUR + ALPACA	Tie	ALPACA
Helpfulness	18.7%	59.1%	**22.2%**
Honesty	**17.5%**	65.4%	17.1%
Harmlessness	**15.5%**	70.6%	13.9%
	DYNOSAUR + INST. GPT-4	Tie	INST. GPT-4
Helpfulness	27.8%	42.9%	**29.3%**
Honesty	**21.0%**	59.9%	19.1%
Harmlessness	**19.8%**	62.3%	17.9%

(a) DYNOSAUR as a supplement to automatically generated instructions ALPACA and INST. GPT-4.

	DYNOSAUR	Tie	SUPER-NI
Helpfulness	**19.5%**	61.5%	19.0%
Honesty	**15.5%**	71.8%	12.7%
Harmlessness	**13.5%**	73.8%	12.7%
	DYNOSAUR + ALPACA	Tie	SUPER-NI + ALPACA
Helpfulness	17.1%	65.5%	**17.4%**
Honesty	19.5%	59.9%	**20.6%**
Harmlessness	**15.5%**	73.4%	11.1%
	DYNOSAUR + INST. GPT-4	Tie	SUPER-NI + INST. GPT-4
Helpfulness	**18.2%**	63.9%	17.9%
Honesty	**17.9%**	68.6%	13.5%
Harmlessness	**16.7%**	70.2%	13.1%

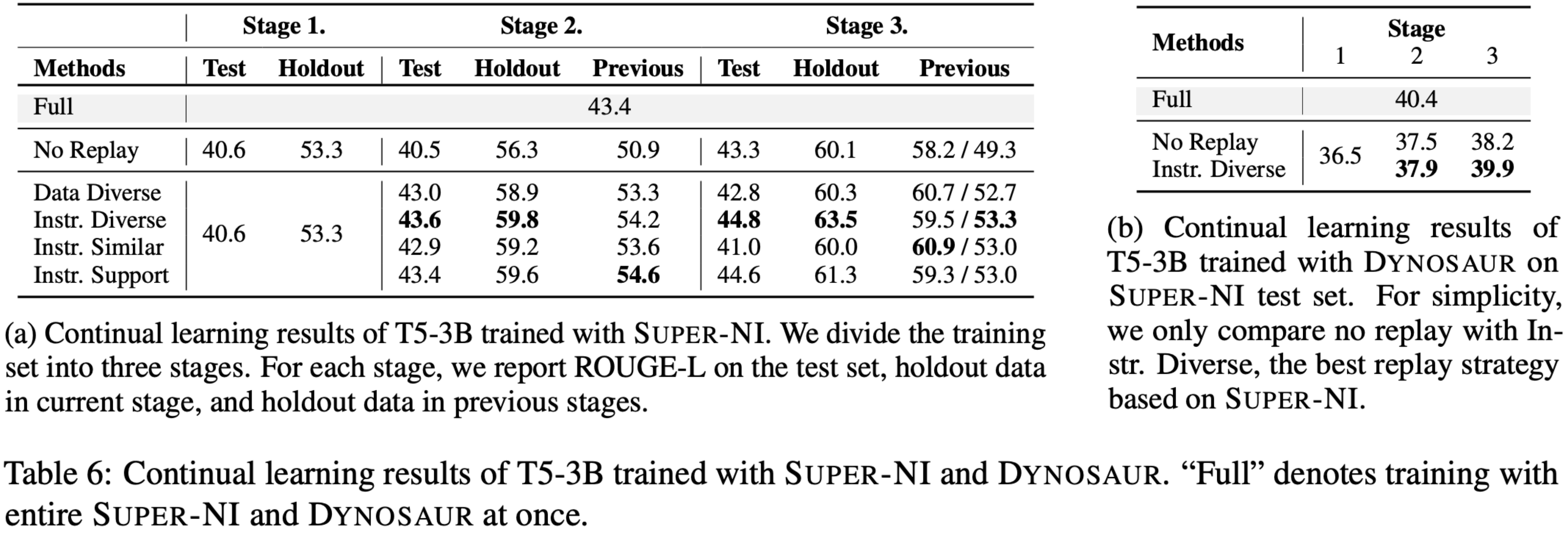
(b) Comparing DYNOSAUR and SUPER-NI.

Table 2: Human evaluation on LLAMA-7B with user instructions. The percentages in columns with dataset name A indicate how many of the generations produced by models trained with A are better than the ones produced by the other data B on USER-INSTRUCTION-252. “Tie” means that the generations of the two models have similar quality.

4. Continual Learning with Dynamically Growing Datasets

- Replay method (이전에 학습된 태스크 중 선택하여 이후에도 학습) 활용하여 continual learning
  - Do we need to replay history tasks?
  - Shall we replay tasks based on instructions or data?
  - Which tasks to replay?
- 실험 세팅
  1. No Replay: replay 태스크 없이 학습
  2. Instr. Diverse: current stage와 representation 기준으로 가장 다른 last stage의 태스크 replay
  3. Instr. Similar: current stage와 가장 비슷한 last stage의 태스크 replay
  4. Instr. Support: last stage의 가장 representative 태스크 replay
  5. Data Diverse: example data와 similarity 기준으로 다양한 task replay

4. Continual Learning with Dynamically Growing Datasets



	Stage 1.		Stage 2.			Stage 3.		
Methods	**Test**	**Holdout**	**Test**	**Holdout**	**Previous**	**Test**	**Holdout**	**Previous**
Full	43.4							
No Replay	40.6	53.3	40.5	56.3	50.9	43.3	60.1	58.2 / 49.3
Data Diverse	40.6	53.3	43.0	58.9	53.3	42.8	60.3	60.7 / 52.7
Instr. Diverse			**43.6**	**59.8**	54.2	**44.8**	**63.5**	59.5 / **53.3**
Instr. Similar			42.9	59.2	53.6	41.0	60.0	**60.9** / 53.0
Instr. Support			43.4	59.6	**54.6**	44.6	61.3	59.3 / 53.0

(a) Continual learning results of T5-3B trained with SUPER-NI. We divide the training set into three stages. For each stage, we report ROUGE-L on the test set, holdout data in current stage, and holdout data in previous stages.

Methods	**Stage** 1	 2	 3
Full	40.4		
No Replay	36.5	37.5	38.2
Instr. Diverse		**37.9**	**39.9**

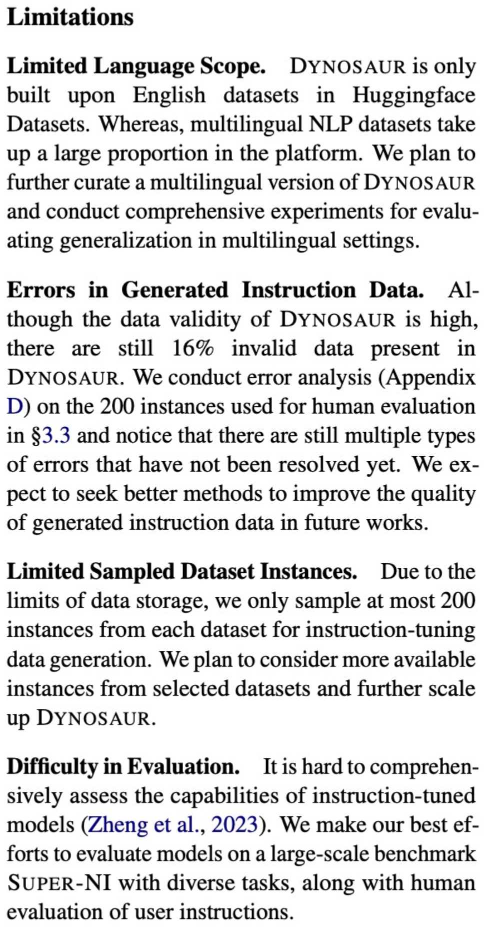
(b) Continual learning results of T5-3B trained with DYNOSAUR on SUPER-NI test set. For simplicity, we only compare no replay with Instr. Diverse, the best replay strategy based on SUPER-NI.

Table 6: Continual learning results of T5-3B trained with SUPER-NI and DYNOSAUR. “Full” denotes training with entire SUPER-NI and DYNOSAUR at once.

DYNOSAUR: A Dynamic Growth Paradigm for Instruction-Tuning Data Curation

5. Conclusion

- 기존 데이터 플랫폼을 활용하여 instruction 데이터 증강하는 방법론 제안
- Continual learning 방법론 제안한다 했으나 사실상 제안은 아니고 실험만
- Research scope와 limitation이 명확
- 데이터와 코드 공개하여 데이터 확장에 쓰기는 유용할 듯
  - https://github.com/WadeYin9712/Dynosaur/tree/main



Limitations

Limited Language Scope. DYNOSAUR is only built upon English datasets in Huggingface Datasets. Whereas, multilingual NLP datasets take up a large proportion in the platform. We plan to further curate a multilingual version of DYNOSAUR and conduct comprehensive experiments for evaluating generalization in multilingual settings.

Errors in Generated Instruction Data. Although the data validity of DYNOSAUR is high, there are still 16% invalid data present in DYNOSAUR. We conduct error analysis (Appendix D) on the 200 instances used for human evaluation in §3.3 and notice that there are still multiple types of errors that have not been resolved yet. We expect to seek better methods to improve the quality of generated instruction data in future works.

Limited Sampled Dataset Instances. Due to the limits of data storage, we only sample at most 200 instances from each dataset for instruction-tuning data generation. We plan to consider more available instances from selected datasets and further scale up DYNOSAUR.

Difficulty in Evaluation. It is hard to comprehensively assess the capabilities of instruction-tuned models (Zheng et al., 2023). We make our best efforts to evaluate models on a large-scale benchmark SUPER-NI with diverse tasks, along with human evaluation of user instructions.

EXPLORE-INSTRUCT: Enhancing Domain-Specific Instruction Coverage through Active Exploration

Fanqi Wan[1]*, Xinting Huang[2]†, Tao Yang[1], Xiaojun Quan[1]†, Wei Bi[2], Shuming Shi[2]
[1]School of Computer Science and Engineering, Sun Yat-sen University, China
[2]Tencent AI Lab
{wanfq, yangt225}@mail2.sysu.edu.cn, quanxj3@mail.sysu.edu.cn,
{timxthuang, victoriabi, shumingshi}@tencent.com

EMNLP 2023

Explore-Instruct: Enhancing Domain-Specific Instruction Coverage through Active Exploration

1. Introduction

- General domain에서 성공적인 LLM, specific domain에서의 성능 향상을 위한 최근 연구들
- Domain-specific instruction 생성의 두 가지 방향: 1) human-curated 2) data generated by LLMs
- 하지만 현재 방법론들은 그림과 같이 넓은 범위의 potential instruction을 cover하기 어려움 – human curation에 대한 과도한 의존?
- Explore-Instruct 제안을 통해 domain-specific instruction data의 coverage를 늘리고자 함

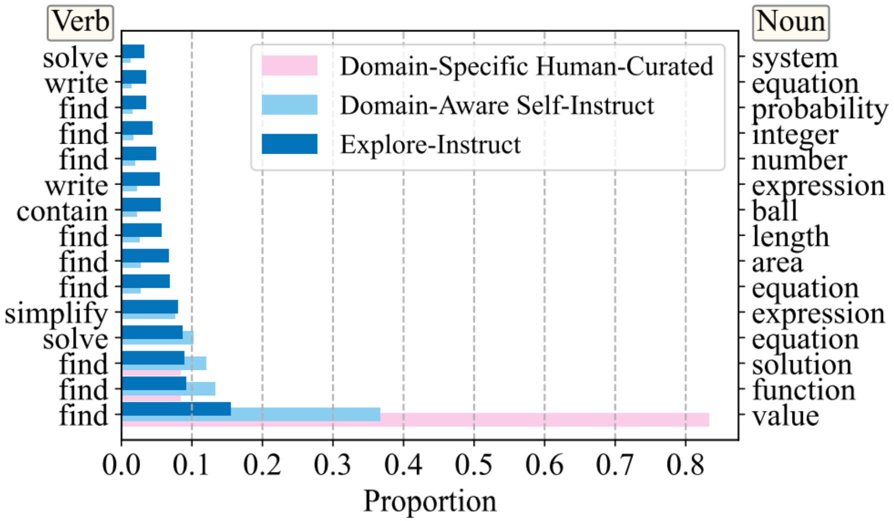


Figure 1: The top-15 most common verb-noun pairs in instructions of previous methods and EXPLORE-INSTRUCT for math problem-solving. It reveals an over-concentrated range of instructions in prior methods, while EXPLORE-INSTRUCT offers broader coverage.

Explore-Instruct: Enhancing Domain-Specific Instruction Coverage through Active Exploration

1. Introduction

- Domain space를 tree라고 간주하고 traverse하며 새로운 instruction 데이터를 생성
  - **Look ahead**: potential fine-grained sub-tasks를 탐구
  - **Backtracking exploration**: 검색 범위를 늘리기 위한 대체 branch를 찾아 domain 스펙트럼 넓힘
- Explore-Instruct의 효과를 rewriting, brainstorming, math 도메인에 실험

2. Methods – Domain Space Representation

- Domain-specific instruction의 coverage는 **breadth**, **depth**에 크게 영향을 받을 것이라 가정
  - Breath: 도메인 내 태스크 다양성
  - Depth: 도메인 내 전문성
- Domain space를 tree $\mathcal{T}$, node V는 task, edge E 태스크 간 hierarchical relationship (task-subtask)으로 모델링

2. Methods – Active Exploration Strategy

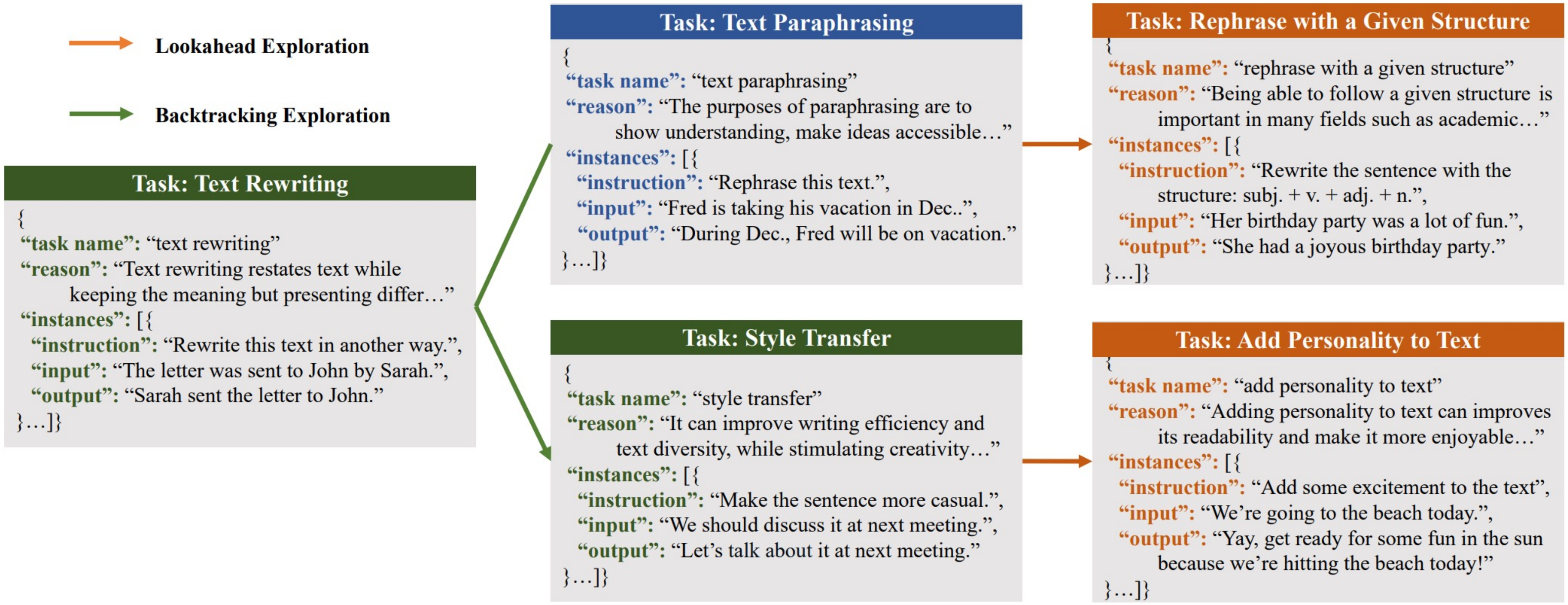
Lookahead Exploration

- Depth 방향으로 domain space를 탐색
- Task V_i가 있을 때 LLM은 이를 $\mathcal{T}$에 존재하는 M개의 sub-task로 나눔

Backtracking Exploration

- Search boundary를 넓히고 다양성을 높이기 위해 alternative branches를 찾음
- V_j가 있을 때 parent V_i를 찾고, LLM이 V_i의 breadth-wise M sub-task를 탐색하도록 함

2. Methods



Task: Text Rewriting

```
{
"task name": "text rewriting"
"reason": "Text rewriting restates text while
        keeping the meaning but presenting differ…"
"instances": [{
  "instruction": "Rewrite this text in another way.",
  "input": "The letter was sent to John by Sarah.",
  "output": "Sarah sent the letter to John."
}…]}
```

Task: Text Paraphrasing

```
{
"task name": "text paraphrasing"
"reason": "The purposes of paraphrasing are to
        show understanding, make ideas accessible…"
"instances": [{
  "instruction": "Rephrase this text.",
  "input": "Fred is taking his vacation in Dec..",
  "output": "During Dec., Fred will be on vacation."
}…]}
```

Task: Style Transfer

```
{
"task name": "style transfer"
"reason": "It can improve writing efficiency and
        text diversity, while stimulating creativity…"
"instances": [{
  "instruction": "Make the sentence more casual.",
  "input": "We should discuss it at next meeting.",
  "output": "Let's talk about it at next meeting."
}…]}
```

Task: Rephrase with a Given Structure

```
{
"task name": "rephrase with a given structure"
"reason": "Being able to follow a given structure is
        important in many fields such as academic…"
"instances": [{
  "instruction": "Rewrite the sentence with the
        structure: subj. + v. + adj. + n.",
  "input": "Her birthday party was a lot of fun.",
  "output": "She had a joyous birthday party."
}…]}
```

Task: Add Personality to Text

```
{
"task name": "add personality to text"
"reason": "Adding personality to text can improves
        its readability and make it more enjoyable…"
"instances": [{
  "instruction": "Add some excitement to the text",
  "input": "We're going to the beach today.",
  "output": "Yay, get ready for some fun in the sun
        because we're hitting the beach today!"
}…]}
```

Figure 2: The overview of our proposed EXPLORE-INSTRUCT. It involves two strategic operations: (1) lookahead exploration, and (2) backtracking exploration. The lookahead exploration delves into a multitude of potential fine-grained sub-tasks, whereas the backtracking exploration seeks alternative branches to widen the search boundary.

Explore-Instruct: Enhancing Domain-Specific Instruction Coverage through Active Exploration

3. Data-Centric Analysis

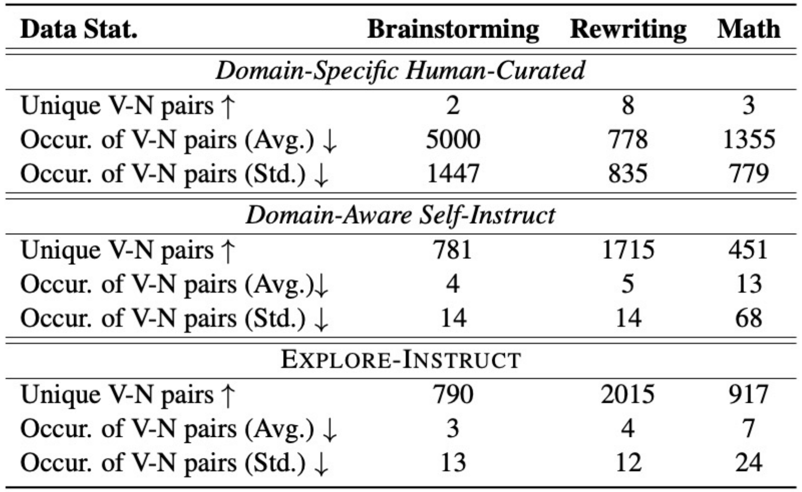
Baselines:

(1) **Domain-specific human-curated**: SuperNI, randomly selected 10,000 samples

(2) **Domain-aware self-instruct**: LLM이 만든 Explore-Instruct의 seed collection (depth K=0)

Statistics

- Unique V-N pairs의 수 기준으로 Explore-Instruct 데이터셋이 가장 다양함 – 특히 rewriting, math 도메인에서 더

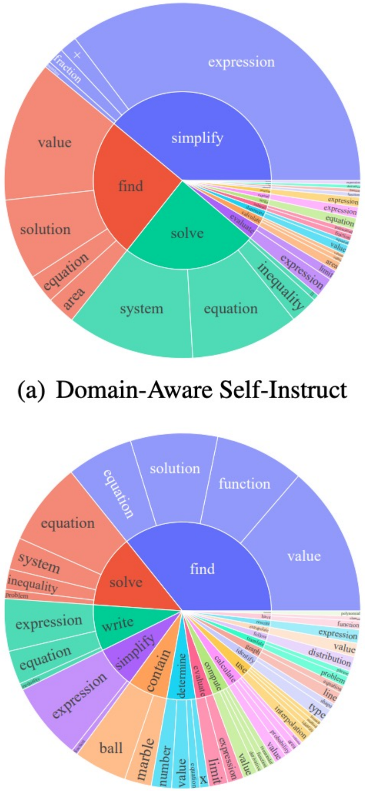


Data Stat.	**Brainstorming**	**Rewriting**	**Math**
Domain-Specific Human-Curated			
Unique V-N pairs ↑	2	8	3
Occur. of V-N pairs (Avg.) ↓	5000	778	1355
Occur. of V-N pairs (Std.) ↓	1447	835	779
Domain-Aware Self-Instruct			
Unique V-N pairs ↑	781	1715	451
Occur. of V-N pairs (Avg.)↓	4	5	13
Occur. of V-N pairs (Std.) ↓	14	14	68
EXPLORE-INSTRUCT			
Unique V-N pairs ↑	790	2015	917
Occur. of V-N pairs (Avg.) ↓	3	4	7
Occur. of V-N pairs (Std.) ↓	13	12	24

Table 1: Statistics of verb-noun (V-N) pairs in the instructions obtained from Domain-Specific Human-Curated, Domain-Aware Self-Instruct, and EXPLORE-INSTRUCT in different domains.

3. Data-Centric Analysis

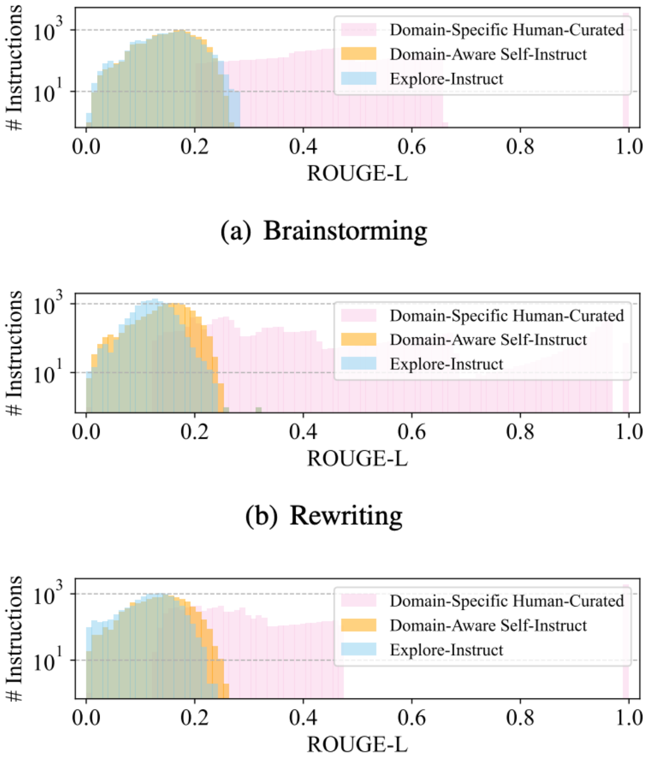
다양한 V-N pair



(a) Domain-Aware Self-Instruct

(b) EXPLORE-INSTRUCT

Figure 3: The root verb-noun pairs in instructions of (a) Domain-Aware Self-Instruct and (b) EXPLORE-INSTRUCT in the math domain, where the inner circle of the plot represents the root verb of the generated instructions, and the outer circle represents the direct nouns.



(a) Brainstorming

(b) Rewriting

(c) Math

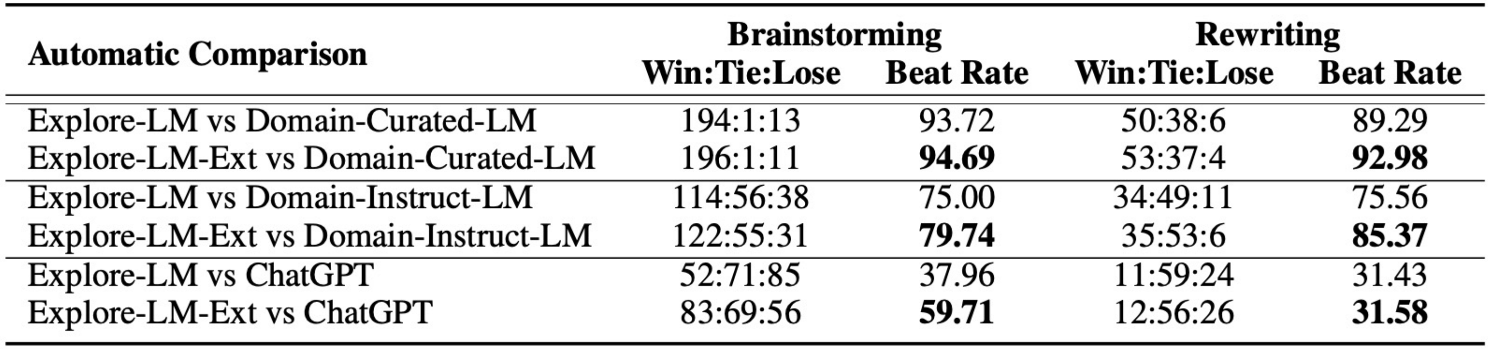
Figure 4: The distribution of average ROUGE-L overlap between generated and existing instructions of Domain-Specific Human-Curated, Domain-Aware Self-Instruct, and EXPLORE-INSTRUCT in different domains.

기존 instruction과의
Overlapping R-L dist.
제안하는 방법이
가장 적은 중복 보임

4. Experimental Settings

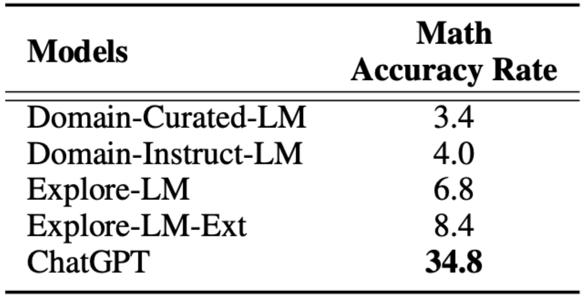
- Benchmarks:
  - Rewriting: BELLE (translated)
  - Brainstorming: BELLE (translated)
  - Math: 500 questions from MATH
- Baselines (LLaMA 7B):
  - Explore-LM: Explore-Instruct로 학습된 모델
  - DomainCurated-LM: Human-Curated 데이터로 학습된 모델
  - Domain-Instruct-LM: Domain-Aware Self-Instruct 데이터로 학습된 모델
- 학습 Data: 10,000 samples from each subtasks

5. Results and Analysis



Automatic Comparison	Brainstorming		Rewriting	
	Win:Tie:Lose	Beat Rate	Win:Tie:Lose	Beat Rate
Explore-LM vs Domain-Curated-LM	194:1:13	93.72	50:38:6	89.29
Explore-LM-Ext vs Domain-Curated-LM	196:1:11	**94.69**	53:37:4	**92.98**
Explore-LM vs Domain-Instruct-LM	114:56:38	75.00	34:49:11	75.56
Explore-LM-Ext vs Domain-Instruct-LM	122:55:31	**79.74**	35:53:6	**85.37**
Explore-LM vs ChatGPT	52:71:85	37.96	11:59:24	31.43
Explore-LM-Ext vs ChatGPT	83:69:56	**59.71**	12:56:26	**31.58**

Table 2: Automatic evaluation results in the brainstorming and rewriting domains. It demonstrates that Explore-LM outperforms multiple baselines with a large Beat Rate and nearly matches the performance of ChatGPT. Additionally, with a substantial increase in training instances, the performance of Explore-LM-Ext can be further enhanced.



Models	Math Accuracy Rate
Domain-Curated-LM	3.4
Domain-Instruct-LM	4.0
Explore-LM	6.8
Explore-LM-Ext	8.4
ChatGPT	**34.8**

Table 3: Automatic evaluation results in the math domain. The results illustrate that Explore-LM achieves significant improvements on baseline models.

Explore-Instruct: Enhancing Domain-Specific Instruction Coverage through Active Exploration

5. Results and Analysis – Human evaluatioin

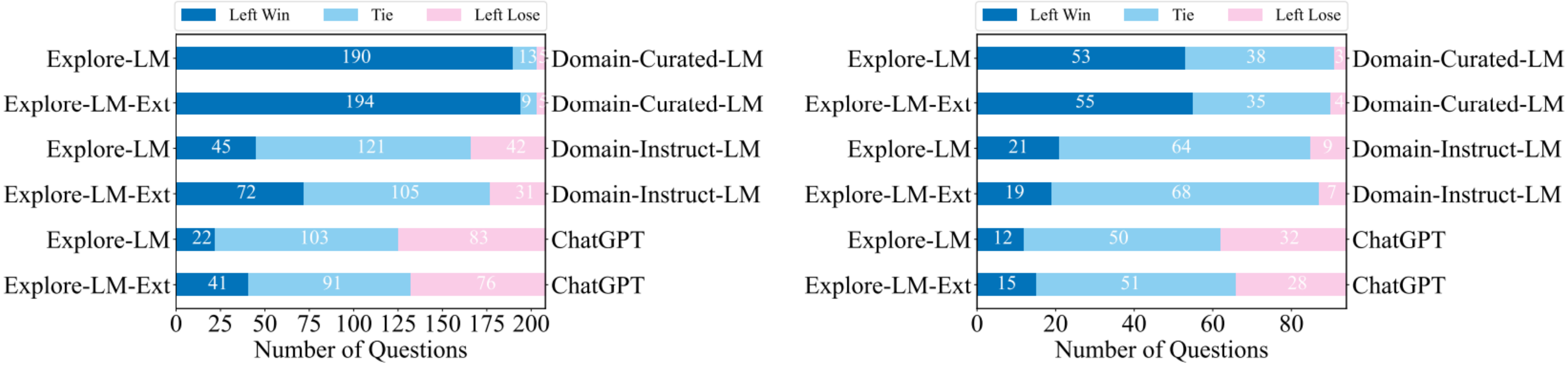


Figure 5: Human evaluation results in the brainstorming (Left) and rewriting (Right) domains.

5. Results and Analysis

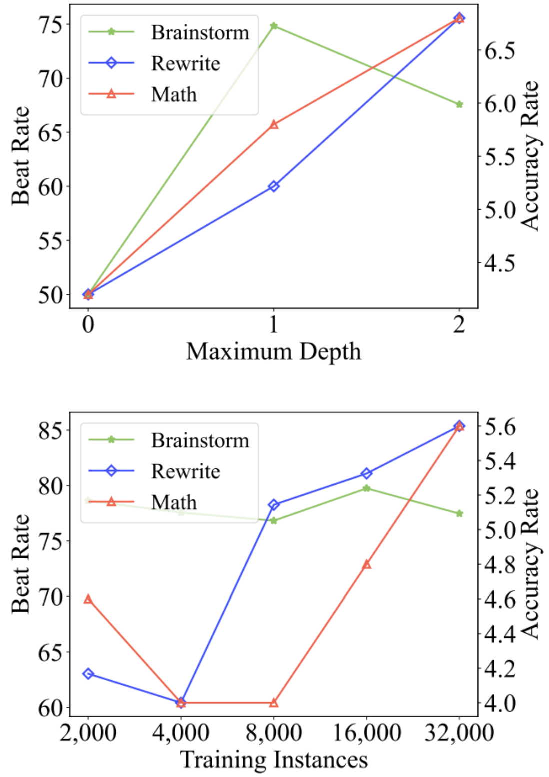


Figure 6: Performance of Explore-LM with different maximum exploration depths (Upper) and the number of training instances (Down).

6. Conclusion

- 도메인 내에서 다양한, 세부적인 태스크를 요구하는 경우 instruction augmentation용으로 사용 가능할 듯
- Rewriting, brainstorming, math만 뽑은 이유가 불명확

Q&A

감사합니다.