AAAR-1.0: Assessing AI's Potential to Assist Research

[1]Renze Lou, [2]Hanzi Xu, [3]Sijia Wang, [4] Jiangshu Du, [1]Ryo Kamoi, [1]Xiaoxin Lu, [5]Jian Xie, [6]Yuxuan Sun, [1]Yusen Zhang, [1]Janice Ahn, [1]Hongchao Fang, [1]Zhuoyang Zou, [1]Wenchao Ma, [7]Xi Li, [8]Kai Zhang, [9]Congying Xia, [3]Lifu Huang, [1]Wenpeng Yin

[1] Pennsylvania State University; [2] Netflix; [3] University of California, Davis; [4] University of Illinois Chicago; [5] Fudan University; [6] Zhejiang University; [7] University of Alabama at Birmingham; [8] Ohio State University; [9] Salesforce Research

{renze.lou, wenpeng}@psu.edu

Renze Lou

Speaker

Prof. Wenpeng Yin

AI for Science:

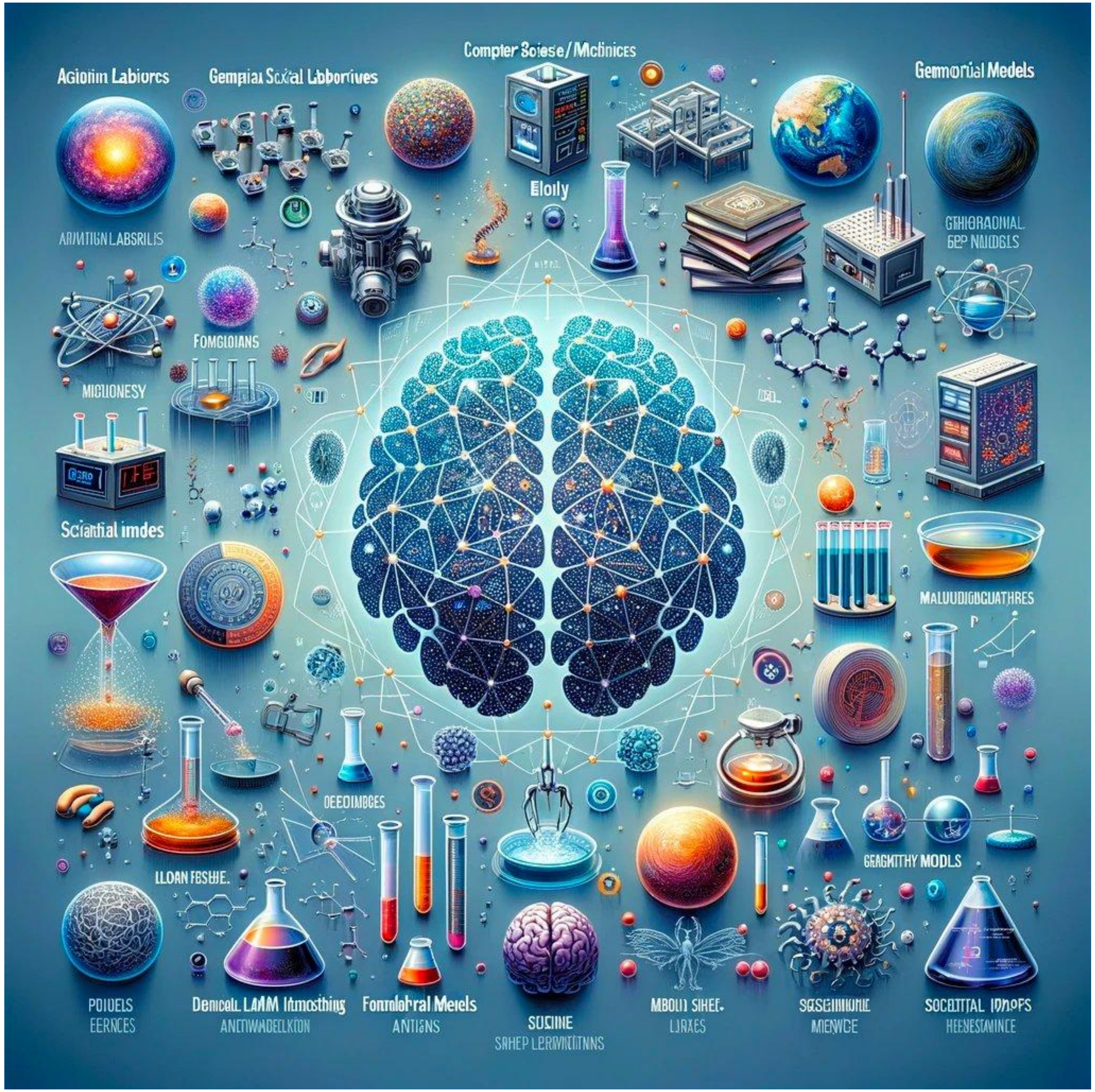


image comes from https://ai4sciencecommunity.github.io/

We are leveraging AI to advance various scientific domains:

- *Social Science, Finance, Medicine, and GeoScience*.

Our specific focus:

- **AI for AI science**
- Harnessing AI to drive advancements in AI research and development

AI for AI Science:

The categories of AI's "science":

Engineering/Data Science:

- **low-level** tasks, e.g., coding, data analyses, data discovery [1][2][3].
- Expectation of **objective** and concrete results.
- **Easy** to assess and quantify.

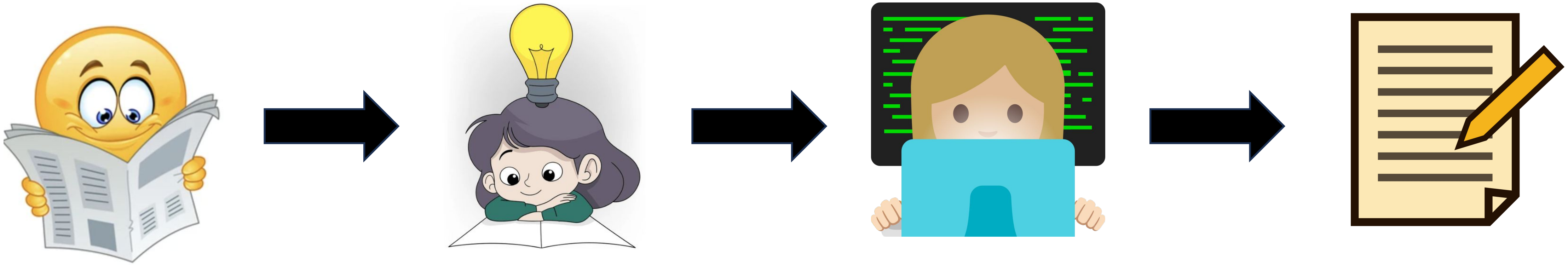
Research science:

- **high-level** research activities, e.g., generating research ideas, writing and reviewing papers [4][5].
- Results are **subjective** and largely influenced by personal taste.
- **Challenging** to assess.

We lack a **benchmark** to transparently evaluate LLMs' capabilities in assisting **AI research science.** 🤔

AI research daily:

We are suffering from the laborious research activities! 😫



Reading/Reviewing Paper **Brainstorming Ideas** **Designing Experiment/Coding** **Writing Paper**

Can we leverage AI to alleviate the labor-intensive aspects of conducting AI research? 🤔

Existing Works/Tools:

AI for AI research science:

- AI-scientist
- MLR-Copilot

We are missing a benchmark to transparently assess the AI’s power in doing AI research

Our Benchmark:

AAAR: **A**ssessing **A**I's Potential to **A**ssist **R**esearch

Consist of 4 presentative tasks extracted from our research daily:

(i) ***Equation Inference*** 🌟

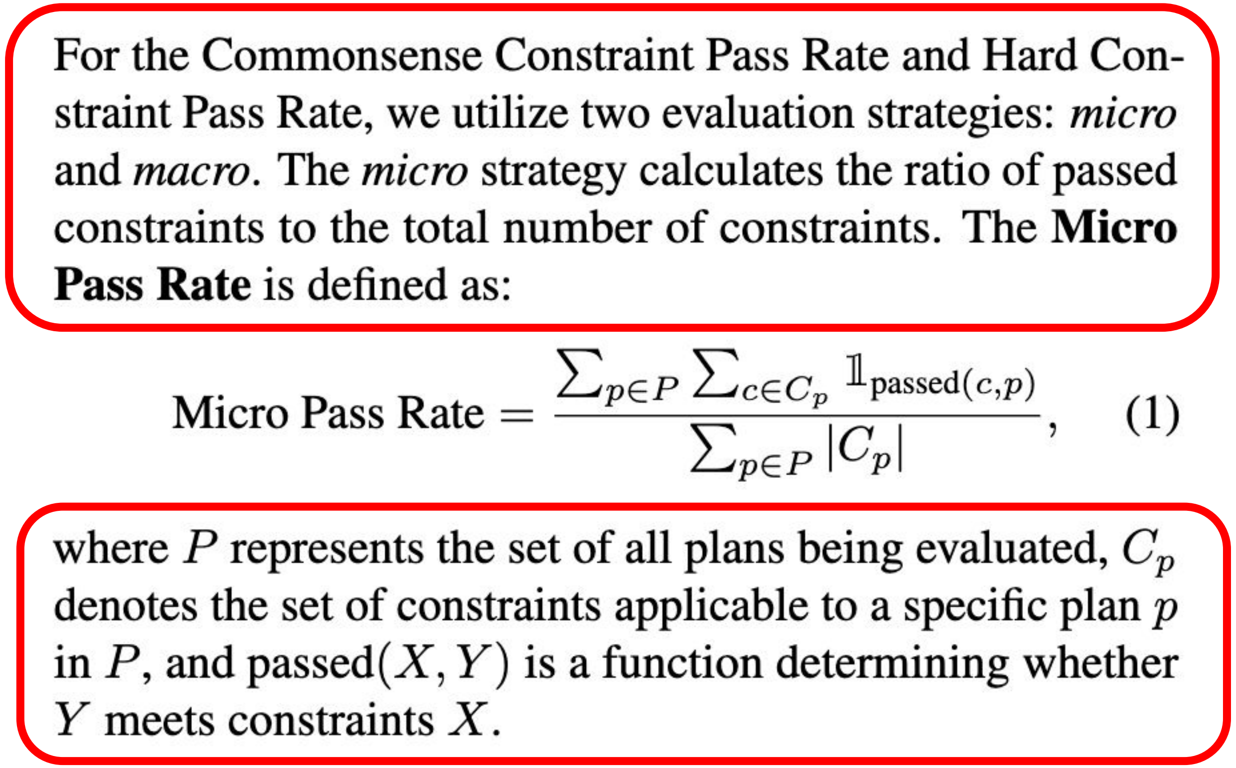
(ii) ***Experiment Design*** 🧪

(iii) ***Paper Weakness*** 🔍

(iv) ***Review Critique*** ✍️

Our Benchmark:

(i) *Equation Inference* 🌟



For the Commonsense Constraint Pass Rate and Hard Constraint Pass Rate, we utilize two evaluation strategies: *micro* and *macro*. The *micro* strategy calculates the ratio of passed constraints to the total number of constraints. The **Micro Pass Rate** is defined as:

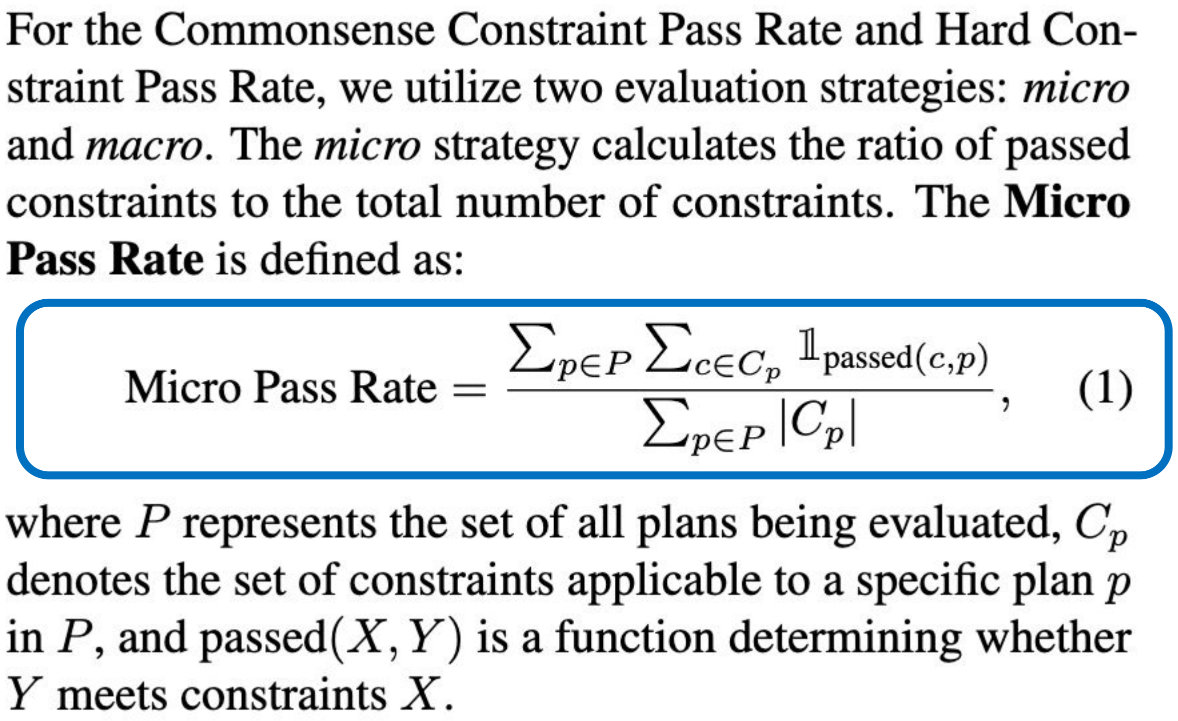
$$\text{Micro Pass Rate} = \frac{\sum_{p \in P} \sum_{c \in C_p} \mathbb{1}_{\text{passed}(c,p)}}{\sum_{p \in P} |C_p|}, \qquad (1)$$

where P represents the set of all plans being evaluated, C_p denotes the set of constraints applicable to a specific plan p in P, and $\text{passed}(X, Y)$ is a function determining whether Y meets constraints X.

surrounding context

Our Benchmark:

(i) *Equation Inference* 🌟



For the Commonsense Constraint Pass Rate and Hard Constraint Pass Rate, we utilize two evaluation strategies: *micro* and *macro*. The *micro* strategy calculates the ratio of passed constraints to the total number of constraints. The **Micro Pass Rate** is defined as:

$$\text{Micro Pass Rate} = \frac{\sum_{p \in P} \sum_{c \in C_p} \mathbb{1}_{\text{passed}(c,p)}}{\sum_{p \in P} |C_p|}, \quad (1)$$

where P represents the set of all plans being evaluated, C_p denotes the set of constraints applicable to a specific plan p in P, and $\text{passed}(X, Y)$ is a function determining whether Y meets constraints X.

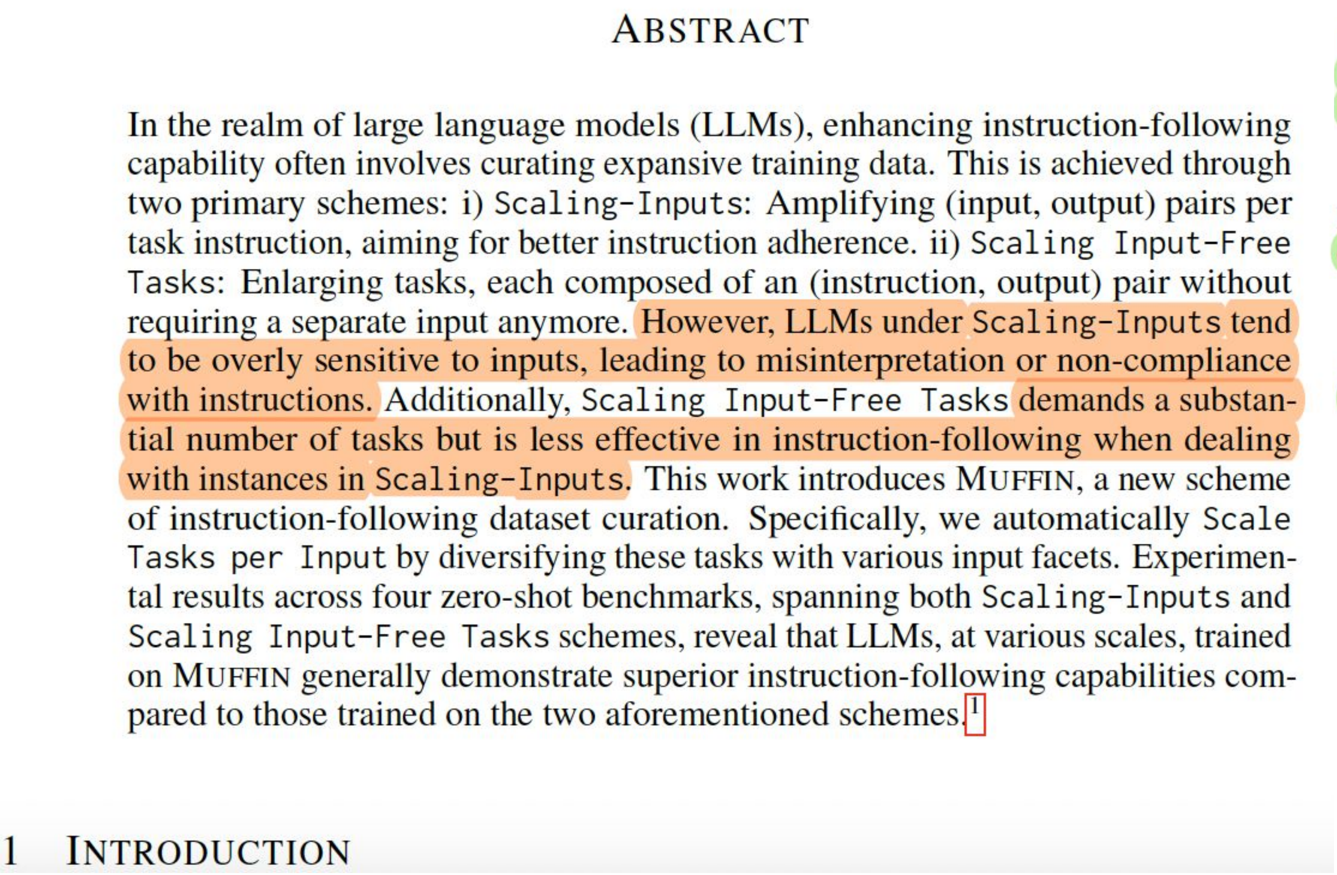
equation (mathematical definition)

Our Benchmark:

(ii) ***Experiment Design*** 🧪

research gap / motivation

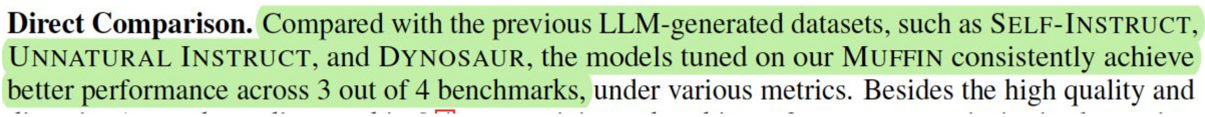
designed experiments/settings



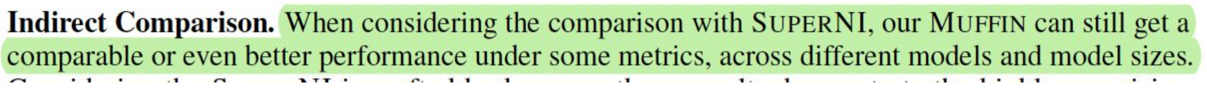
ABSTRACT

In the realm of large language models (LLMs), enhancing instruction-following capability often involves curating expansive training data. This is achieved through two primary schemes: i) Scaling-Inputs: Amplifying (input, output) pairs per task instruction, aiming for better instruction adherence. ii) Scaling Input-Free Tasks: Enlarging tasks, each composed of an (instruction, output) pair without requiring a separate input anymore. However, LLMs under Scaling-Inputs tend to be overly sensitive to inputs, leading to misinterpretation or non-compliance with instructions. Additionally, Scaling Input-Free Tasks demands a substantial number of tasks but is less effective in instruction-following when dealing with instances in Scaling-Inputs. This work introduces MUFFIN, a new scheme of instruction-following dataset curation. Specifically, we automatically Scale Tasks per Input by diversifying these tasks with various input facets. Experimental results across four zero-shot benchmarks, spanning both Scaling-Inputs and Scaling Input-Free Tasks schemes, reveal that LLMs, at various scales, trained on MUFFIN generally demonstrate superior instruction-following capabilities compared to those trained on the two aforementioned schemes.[1]

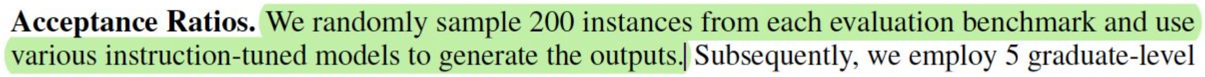
1 INTRODUCTION



Direct Comparison. Compared with the previous LLM-generated datasets, such as SELF-INSTRUCT, UNNATURAL INSTRUCT, and DYNOSAUR, the models tuned on our MUFFIN consistently achieve better performance across 3 out of 4 benchmarks, under various metrics. Besides the high quality and



Indirect Comparison. When considering the comparison with SUPERNI, our MUFFIN can still get a comparable or even better performance under some metrics, across different models and model sizes.

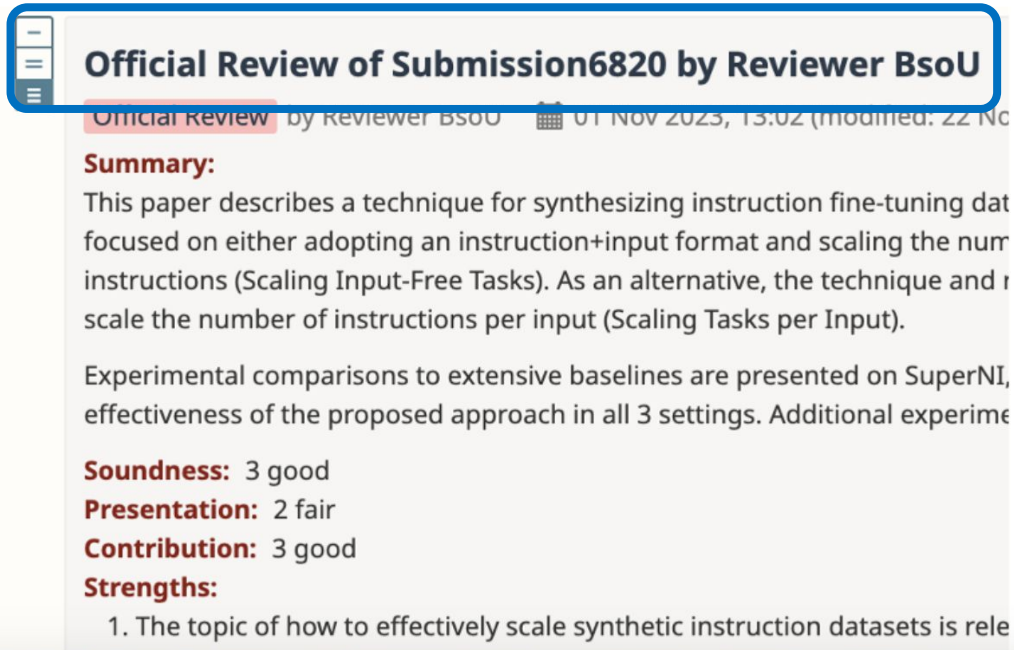


Acceptance Ratios. We randomly sample 200 instances from each evaluation benchmark and use various instruction-tuned models to generate the outputs. Subsequently, we employ 5 graduate-level

...

Our Benchmark:

(iii) ***Paper Weakness*** 🔍 review



Official Review of Submission6820 by Reviewer BsoU

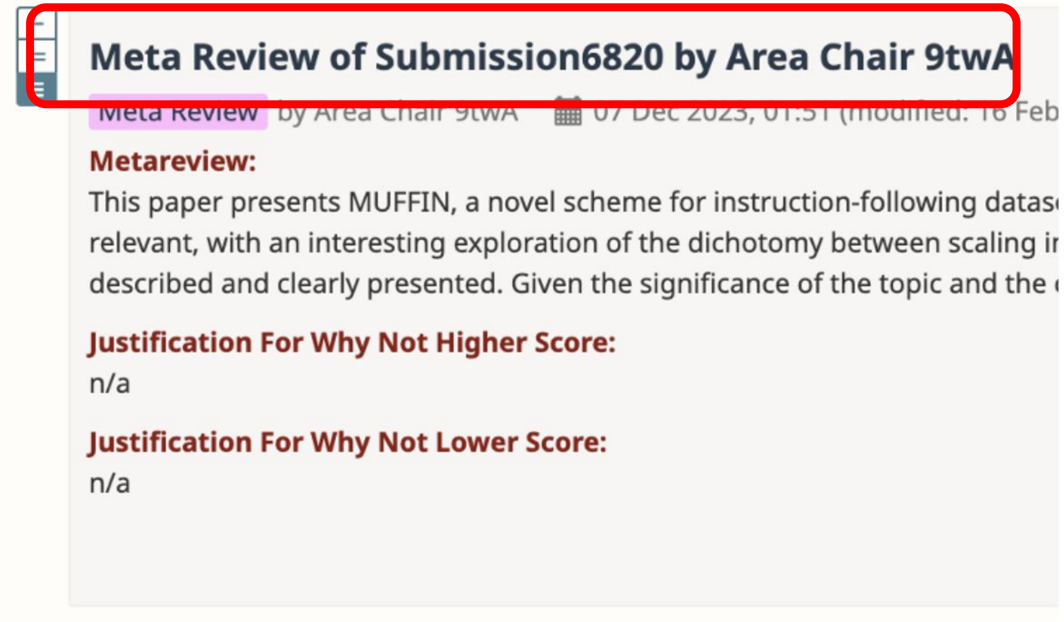
Official Review by Reviewer BsoU 01 Nov 2023, 13:02 (modified: 22 No

Summary:
This paper describes a technique for synthesizing instruction fine-tuning dat
focused on either adopting an instruction+input format and scaling the num
instructions (Scaling Input-Free Tasks). As an alternative, the technique and r
scale the number of instructions per input (Scaling Tasks per Input).

Experimental comparisons to extensive baselines are presented on SuperNI,
effectiveness of the proposed approach in all 3 settings. Additional experime

Soundness: 3 good
Presentation: 2 fair
Contribution: 3 good
Strengths:
1. The topic of how to effectively scale synthetic instruction datasets is rele

(iv) ***Review Critique*** ✍️ meta-review



Meta Review of Submission6820 by Area Chair 9twA

Meta Review by Area Chair 9twA 07 Dec 2023, 01:51 (modified: 16 Feb

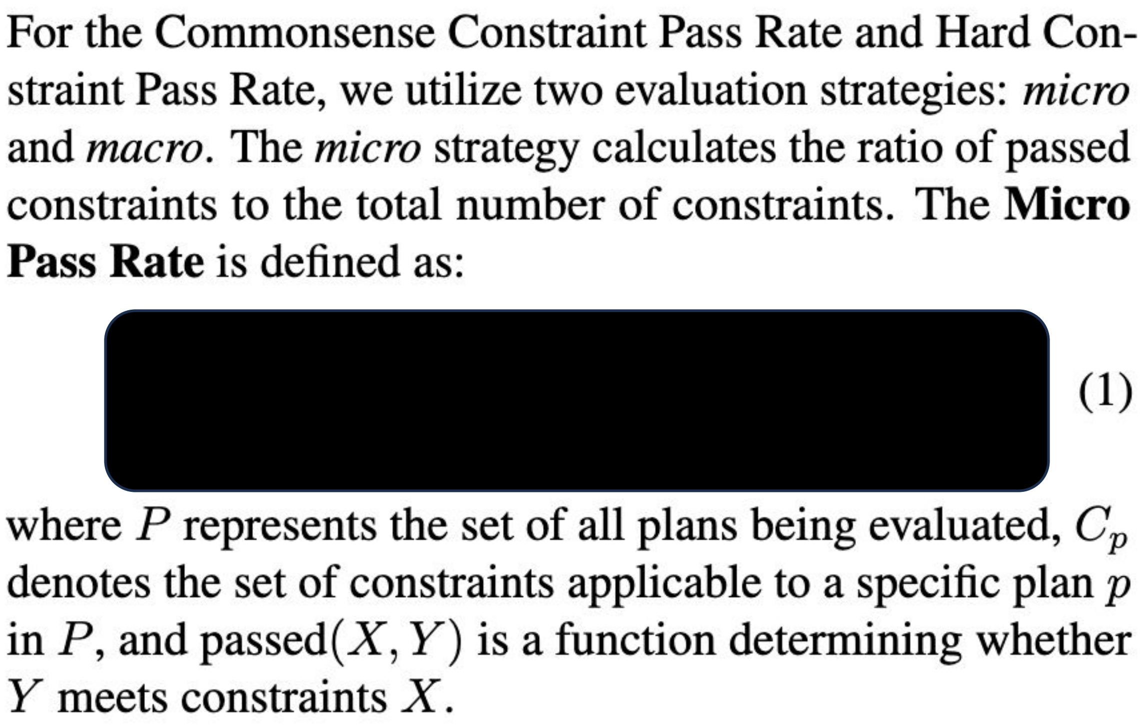
Metareview:
This paper presents MUFFIN, a novel scheme for instruction-following datas
relevant, with an interesting exploration of the dichotomy between scaling in
described and clearly presented. Given the significance of the topic and the

Justification For Why Not Higher Score:
n/a

Justification For Why Not Lower Score:
n/a

🤔How to make a testbed for all the tasks?

Task#1: Equation Inference



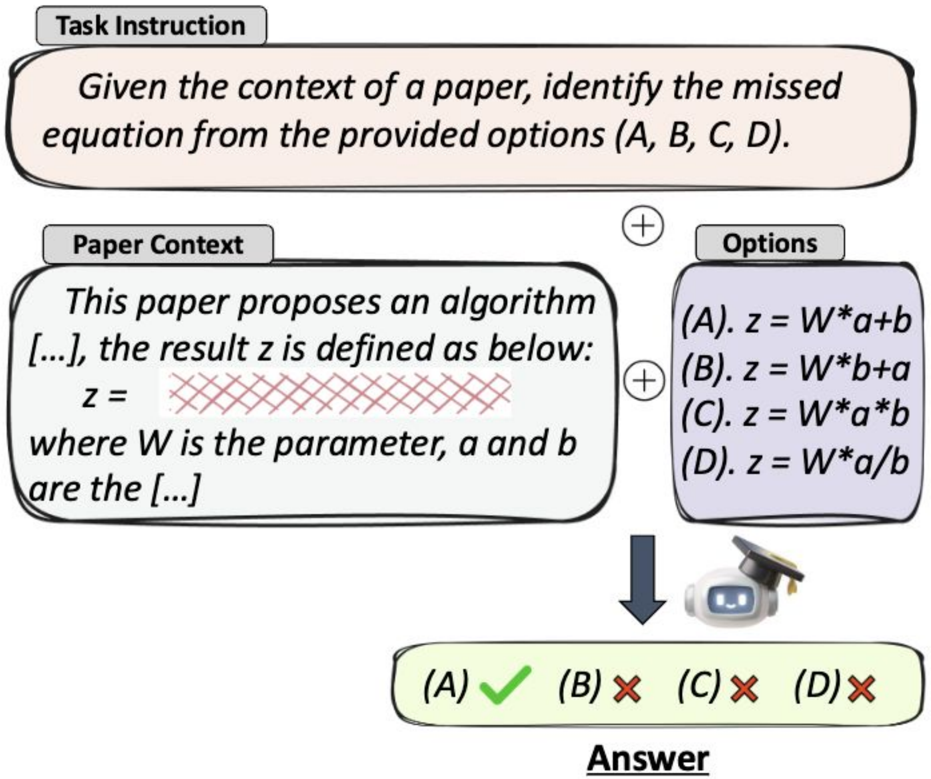
For the Commonsense Constraint Pass Rate and Hard Constraint Pass Rate, we utilize two evaluation strategies: *micro* and *macro*. The *micro* strategy calculates the ratio of passed constraints to the total number of constraints. The **Micro Pass Rate** is defined as:

[redacted] (1)

where P represents the set of all plans being evaluated, C_p denotes the set of constraints applicable to a specific plan p in P, and $\text{passed}(X, Y)$ is a function determining whether Y meets constraints X.

Based on the surrounding context, infer the correct mathematical equation for the algorithm.

Task#1: Equation Inference



Task Instruction

Given the context of a paper, identify the missed equation from the provided options (A, B, C, D).

Paper Context

This paper proposes an algorithm […], the result z is defined as below:

$z =$

where W is the parameter, a and b are the […]

Options

(A). $z = W*a+b$

(B). $z = W*b+a$

(C). $z = W*a*b$

(D). $z = W*a/b$

(A) ✓ (B) ✗ (C) ✗ (D) ✗

Answer

Task #1: Equation Inference

Binary classification setting: infer the correctness of an equation based on the surrounding paper context

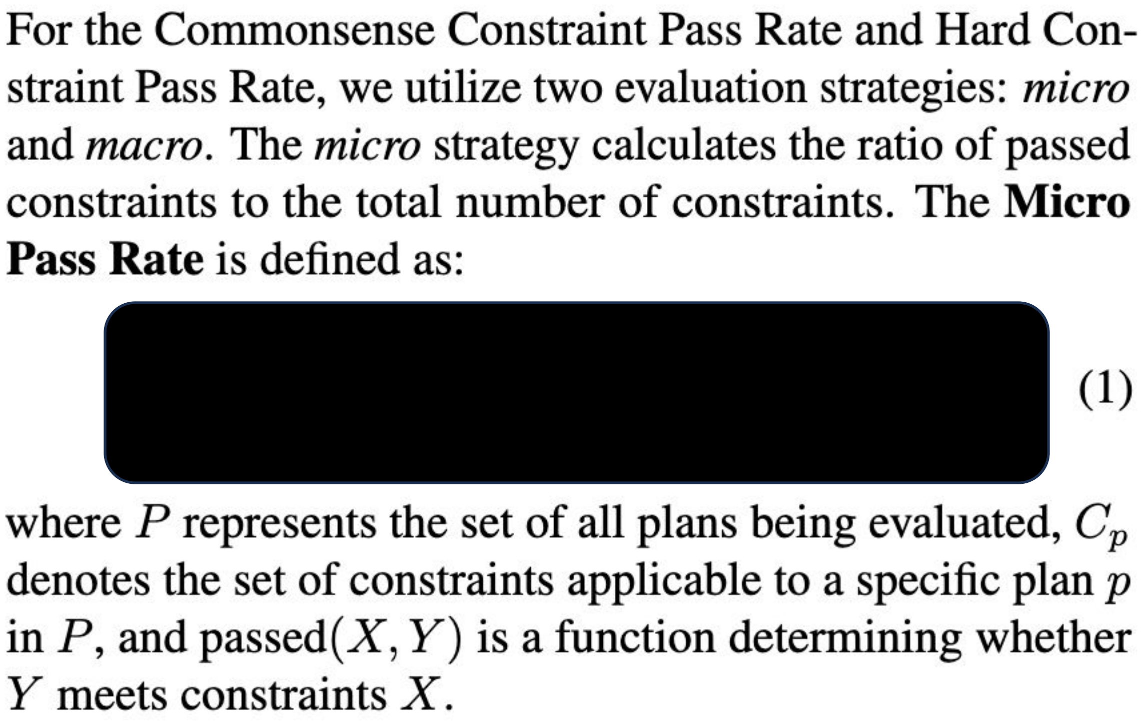
Input:

- Surrounding paper context
- A candidate equation

Output:

- Correctness of the equation (0 or 1).

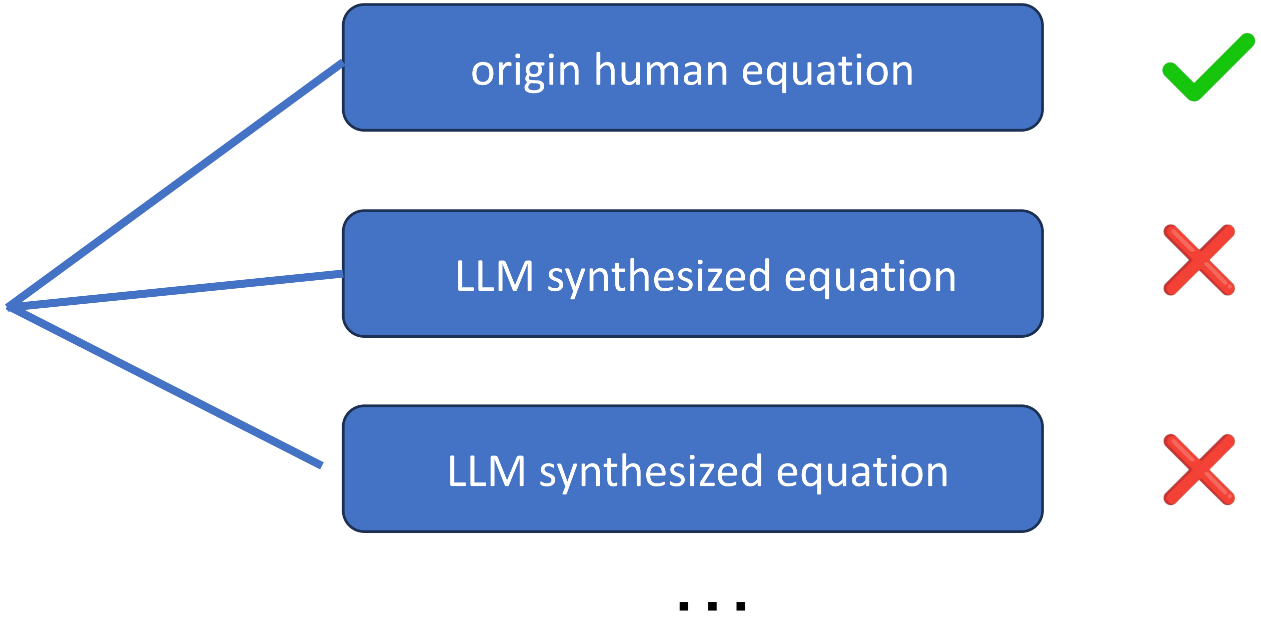
Task#1: Equation Inference



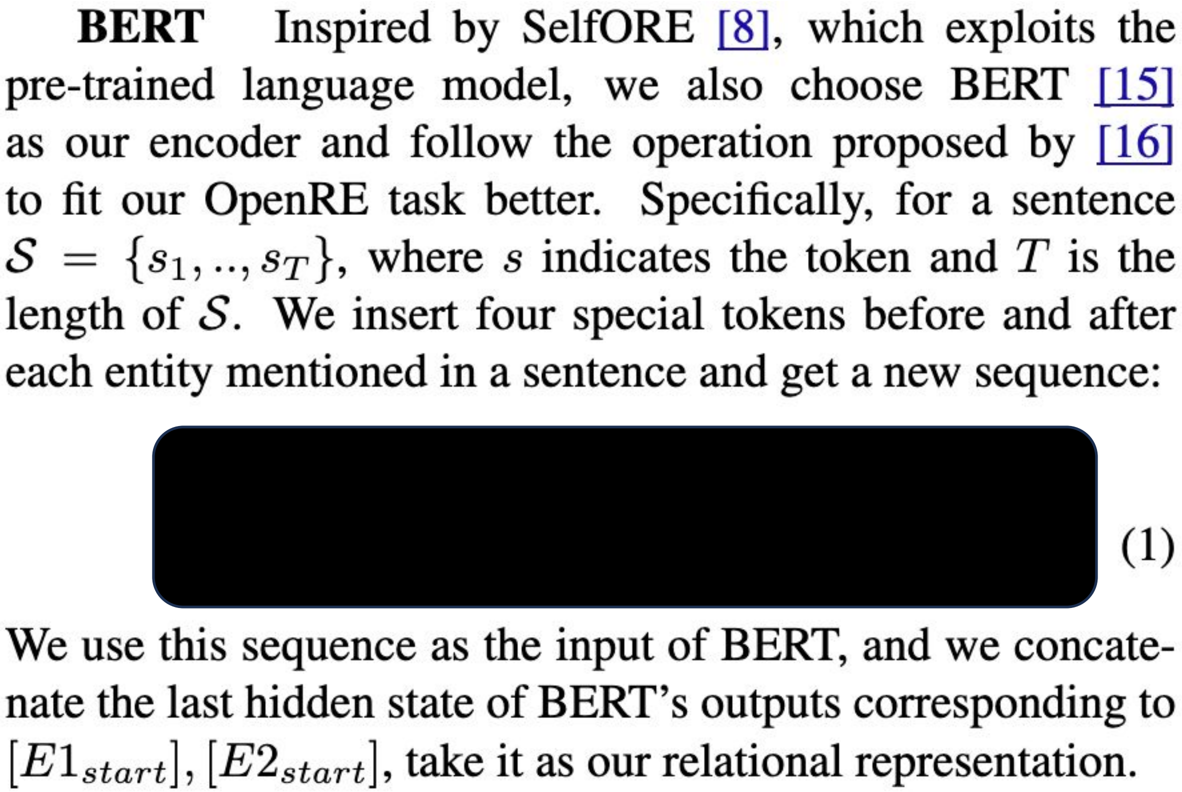
For the Commonsense Constraint Pass Rate and Hard Constraint Pass Rate, we utilize two evaluation strategies: *micro* and *macro*. The *micro* strategy calculates the ratio of passed constraints to the total number of constraints. The **Micro Pass Rate** is defined as:

[redacted] (1)

where P represents the set of all plans being evaluated, C_p denotes the set of constraints applicable to a specific plan p in P, and passed(X, Y) is a function determining whether Y meets constraints X.



Task#1: Equation Inference



BERT Inspired by SelfORE [8], which exploits the pre-trained language model, we also choose BERT [15] as our encoder and follow the operation proposed by [16] to fit our OpenRE task better. Specifically, for a sentence $\mathcal{S} = \{s_1, .., s_T\}$, where s indicates the token and T is the length of $\mathcal{S}$. We insert four special tokens before and after each entity mentioned in a sentence and get a new sequence:

[redacted] (1)

We use this sequence as the input of BERT, and we concatenate the last hidden state of BERT's outputs corresponding to $[E1_{start}], [E2_{start}]$, take it as our relational representation.

Binary classification setting: infer the correctness of an equation based on the surrounding paper context

Input:

- Surrounding paper context
- A candidate equation

Output:

- Correctness of the equation (0 or 1).

Task#1: Equation Inference

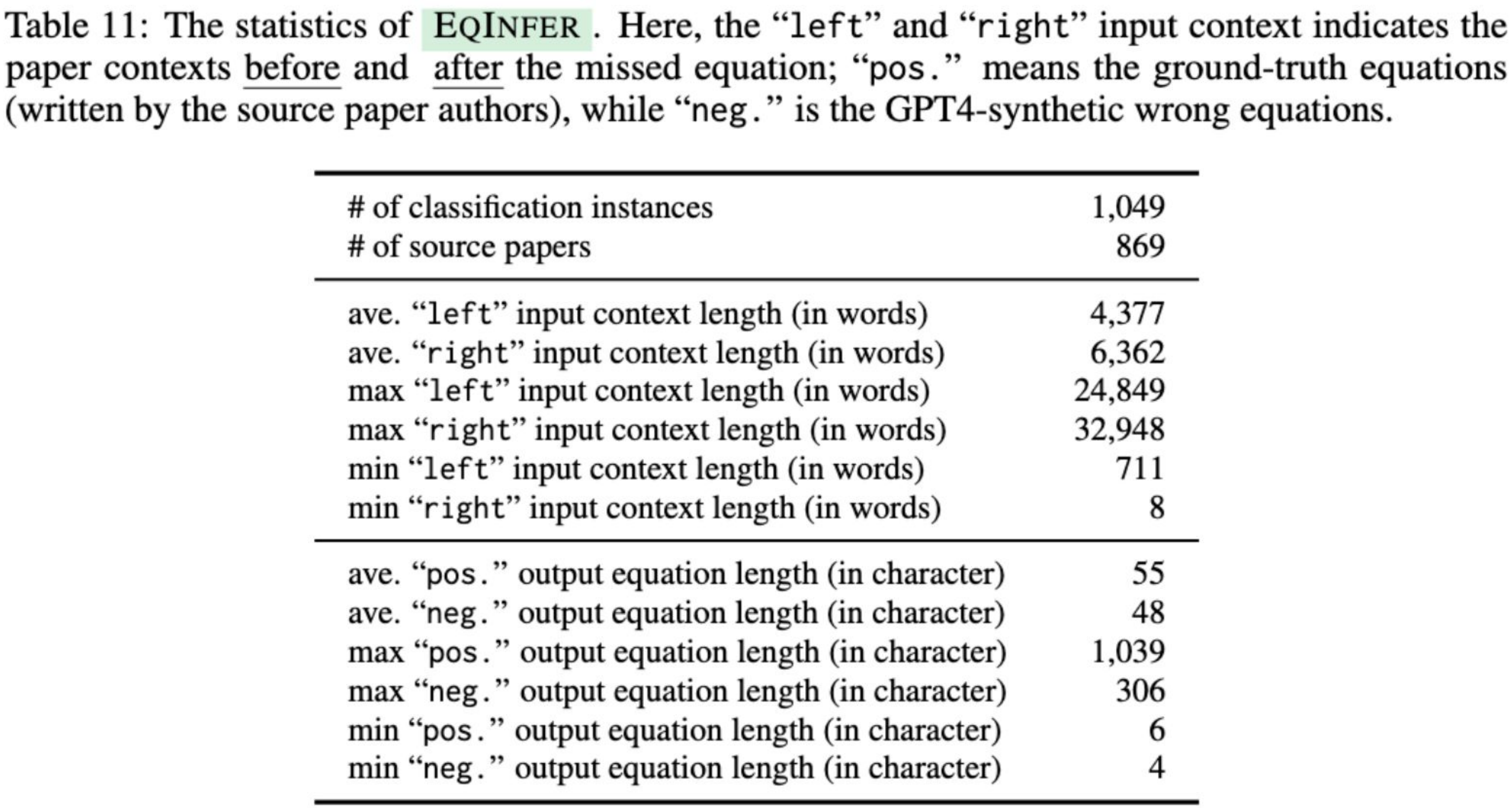
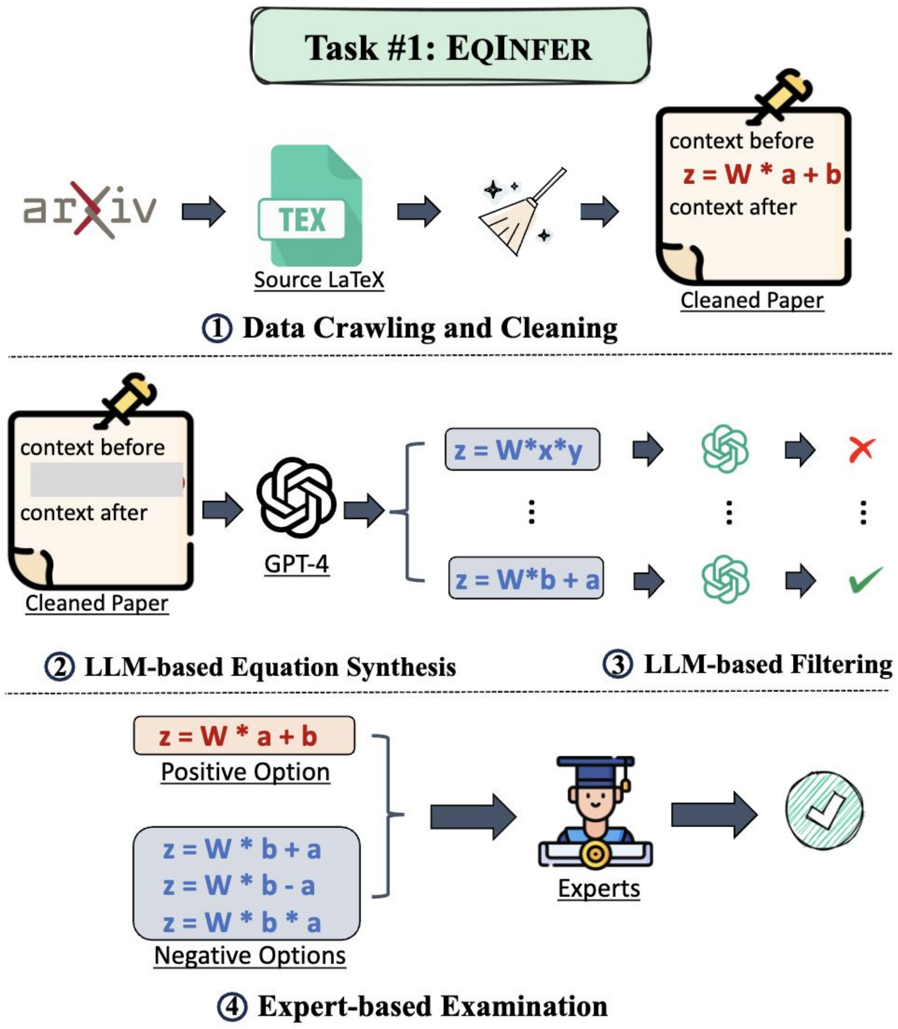


Table 11: The statistics of EQINFER . Here, the "left" and "right" input context indicates the paper contexts before and after the missed equation; "pos." means the ground-truth equations (written by the source paper authors), while "neg." is the GPT4-synthetic wrong equations.

# of classification instances	1,049
# of source papers	869
ave. "left" input context length (in words)	4,377
ave. "right" input context length (in words)	6,362
max "left" input context length (in words)	24,849
max "right" input context length (in words)	32,948
min "left" input context length (in words)	711
min "right" input context length (in words)	8
ave. "pos." output equation length (in character)	55
ave. "neg." output equation length (in character)	48
max "pos." output equation length (in character)	1,039
max "neg." output equation length (in character)	306
min "pos." output equation length (in character)	6
min "neg." output equation length (in character)	4

Equation Inference: data collection



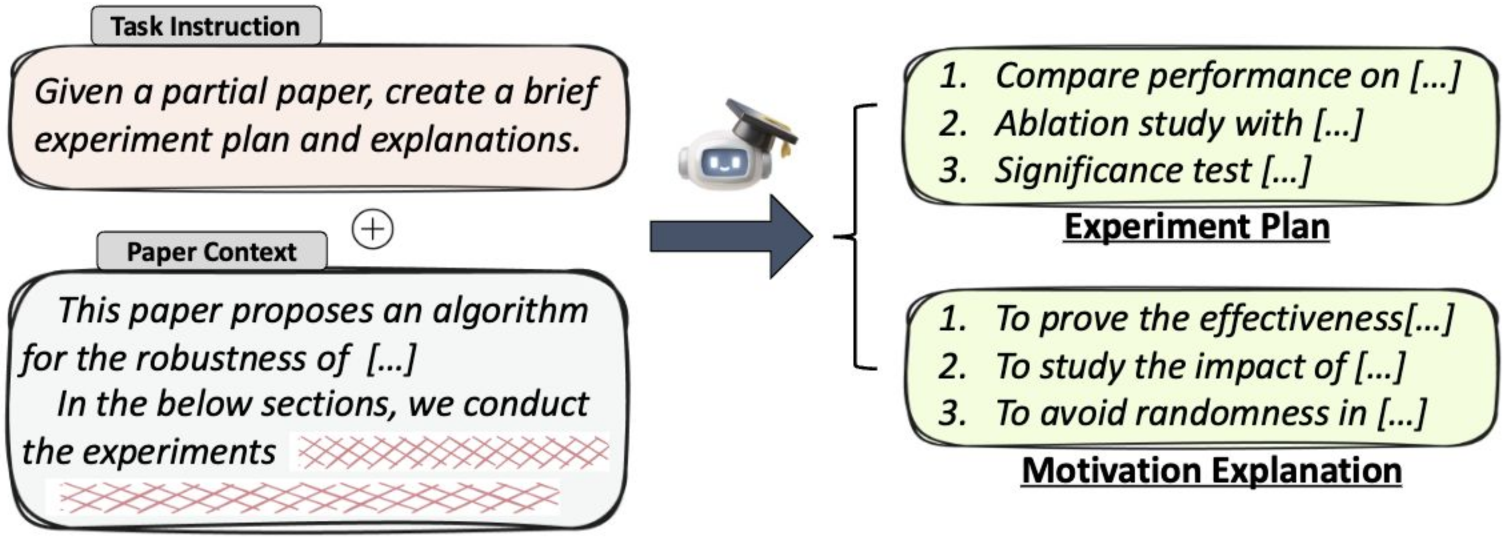
1. <u>Crawling source papers from arXiv:</u>
   - Use source LaTeX is more precise than extracting the text from PDF.
   - Make sure all papers are peer-reviewed, i.e., accepted by well-known conferences.

2. <u>Synthesizing negative equations:</u>
   - Prompt LLM to craft equations based on the surrounding context, i.e., negative options

3. <u>Filtering low-quality negative equations:</u>
   - Only keep the negative equations aligned with the context to avoid any shortcuts.

4. <u>Human examination:</u>
   - After compilation, all negative equations should be different from the positive counterparts, i.e., truly negative.

Task#2: Experiment Design


Task Instruction
Given a partial paper, create a brief experiment plan and explanations.
⊕
Paper Context
This paper proposes an algorithm for the robustness of [...]
In the below sections, we conduct the experiments
1. Compare performance on [...]
2. Ablation study with [...]
3. Significance test [...]
Experiment Plan
1. To prove the effectiveness[...]
2. To study the impact of [...]
3. To avoid randomness in [...]
Motivation Explanation

Task #2: Experiment Design

List generation: suggest a list of experiments based on the research background, along with the explanation.

<u>Input:</u>

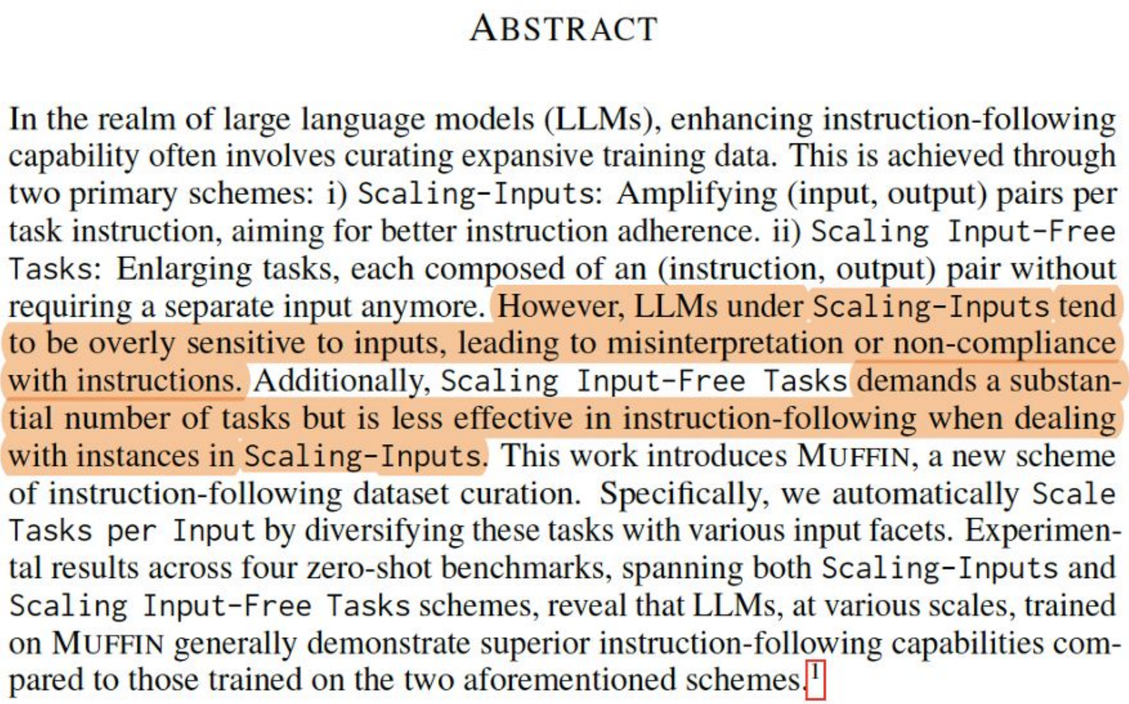
- Research background & motivation (usually the paper “abstract” and “introduction”).

<u>Output:</u>

- Experiment plan: a list of suggested experiments
- Motivation Explanation: a list of explanations (one-one corresponding to the experiments).

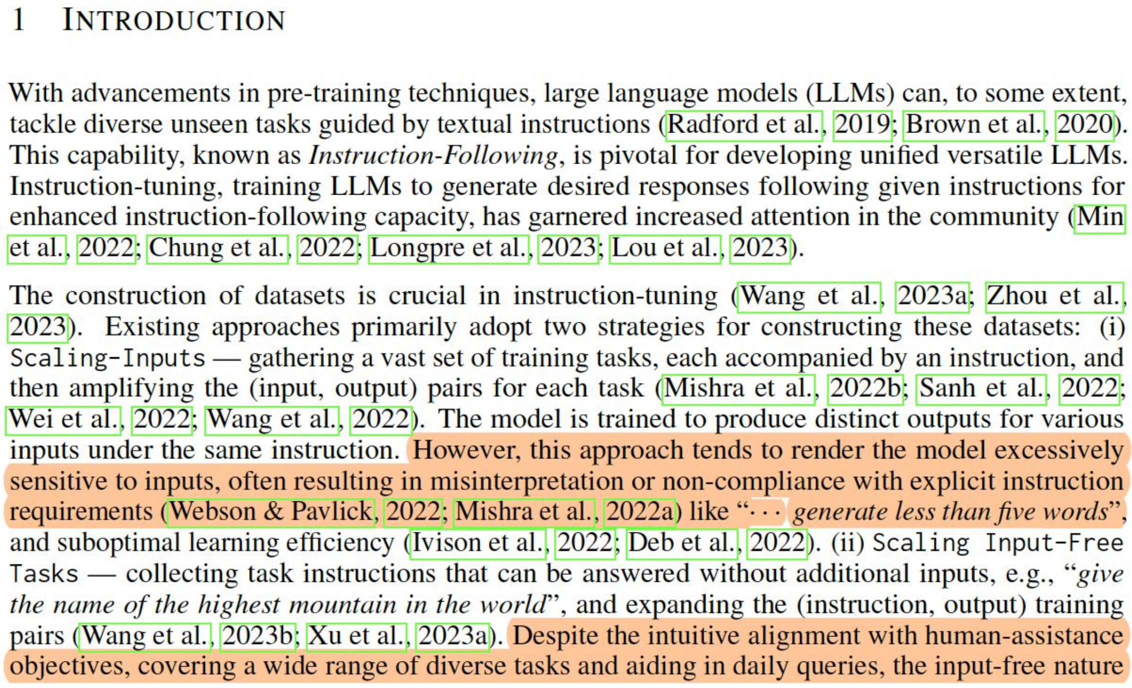
Task#2: Experiment Design

input



ABSTRACT

In the realm of large language models (LLMs), enhancing instruction-following capability often involves curating expansive training data. This is achieved through two primary schemes: i) `Scaling-Inputs`: Amplifying (input, output) pairs per task instruction, aiming for better instruction adherence. ii) `Scaling Input-Free Tasks`: Enlarging tasks, each composed of an (instruction, output) pair without requiring a separate input anymore. However, LLMs under `Scaling-Inputs` tend to be overly sensitive to inputs, leading to misinterpretation or non-compliance with instructions. Additionally, `Scaling Input-Free Tasks` demands a substantial number of tasks but is less effective in instruction-following when dealing with instances in `Scaling-Inputs`. This work introduces MUFFIN, a new scheme of instruction-following dataset curation. Specifically, we automatically `Scale Tasks per Input` by diversifying these tasks with various input facets. Experimental results across four zero-shot benchmarks, spanning both `Scaling-Inputs` and `Scaling Input-Free Tasks` schemes, reveal that LLMs, at various scales, trained on MUFFIN generally demonstrate superior instruction-following capabilities compared to those trained on the two aforementioned schemes.[1]

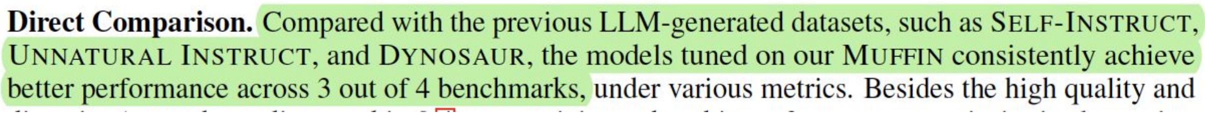


1 INTRODUCTION

With advancements in pre-training techniques, large language models (LLMs) can, to some extent, tackle diverse unseen tasks guided by textual instructions (Radford et al., 2019; Brown et al., 2020). This capability, known as *Instruction-Following*, is pivotal for developing unified versatile LLMs. Instruction-tuning, training LLMs to generate desired responses following given instructions for enhanced instruction-following capacity, has garnered increased attention in the community (Min et al., 2022; Chung et al., 2022; Longpre et al., 2023; Lou et al., 2023).

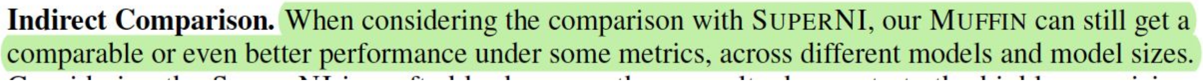
The construction of datasets is crucial in instruction-tuning (Wang et al., 2023a; Zhou et al., 2023). Existing approaches primarily adopt two strategies for constructing these datasets: (i) `Scaling-Inputs` — gathering a vast set of training tasks, each accompanied by an instruction, and then amplifying the (input, output) pairs for each task (Mishra et al., 2022b; Sanh et al., 2022; Wei et al., 2022; Wang et al., 2022). The model is trained to produce distinct outputs for various inputs under the same instruction. However, this approach tends to render the model excessively sensitive to inputs, often resulting in misinterpretation or non-compliance with explicit instruction requirements (Webson & Pavlick, 2022; Mishra et al., 2022a) like "··· *generate less than five words*", and suboptimal learning efficiency (Ivison et al., 2022; Deb et al., 2022). (ii) `Scaling Input-Free Tasks` — collecting task instructions that can be answered without additional inputs, e.g., "*give the name of the highest mountain in the world*", and expanding the (instruction, output) training pairs (Wang et al., 2023b; Xu et al., 2023a). Despite the intuitive alignment with human-assistance objectives, covering a wide range of diverse tasks and aiding in daily queries, the input-free nature

output



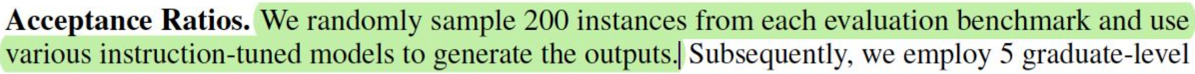
Direct Comparison. Compared with the previous LLM-generated datasets, such as SELF-INSTRUCT, UNNATURAL INSTRUCT, and DYNOSAUR, the models tuned on our MUFFIN consistently achieve better performance across 3 out of 4 benchmarks, under various metrics. Besides the high quality and

benchmark comparison: […]



Indirect Comparison. When considering the comparison with SUPERNI, our MUFFIN can still get a comparable or even better performance under some metrics, across different models and model sizes.

human evaluation: […]

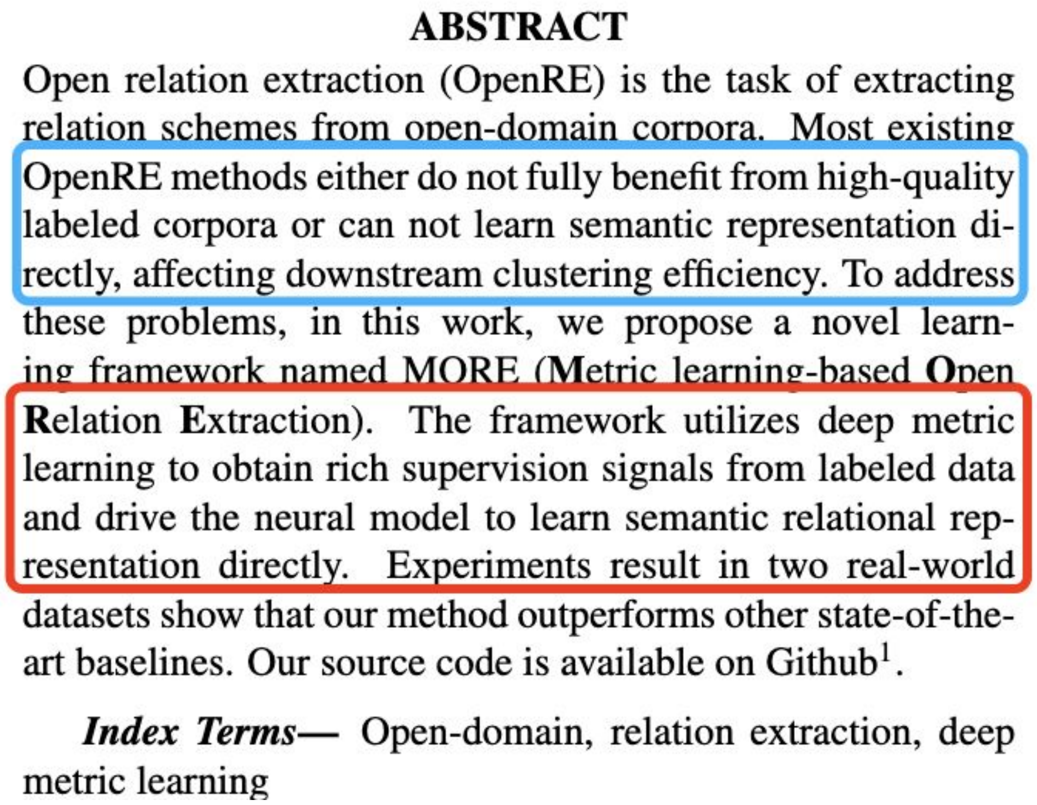


Acceptance Ratios. We randomly sample 200 instances from each evaluation benchmark and use various instruction-tuned models to generate the outputs. Subsequently, we employ 5 graduate-level

ablation study: […]

…

Task#2: Experiment Design



ABSTRACT

Open relation extraction (OpenRE) is the task of extracting relation schemes from open-domain corpora. Most existing OpenRE methods either do not fully benefit from high-quality labeled corpora or can not learn semantic representation directly, affecting downstream clustering efficiency. To address these problems, in this work, we propose a novel learning framework named MORE (**M**etric learning-based **O**pen **R**elation **E**xtraction). The framework utilizes deep metric learning to obtain rich supervision signals from labeled data and drive the neural model to learn semantic relational representation directly. Experiments result in two real-world datasets show that our method outperforms other state-of-the-art baselines. Our source code is available on Github[1].

Index Terms— Open-domain, relation extraction, deep metric learning

List generation: suggest a list of experiments based on the research background, along with the explanation.

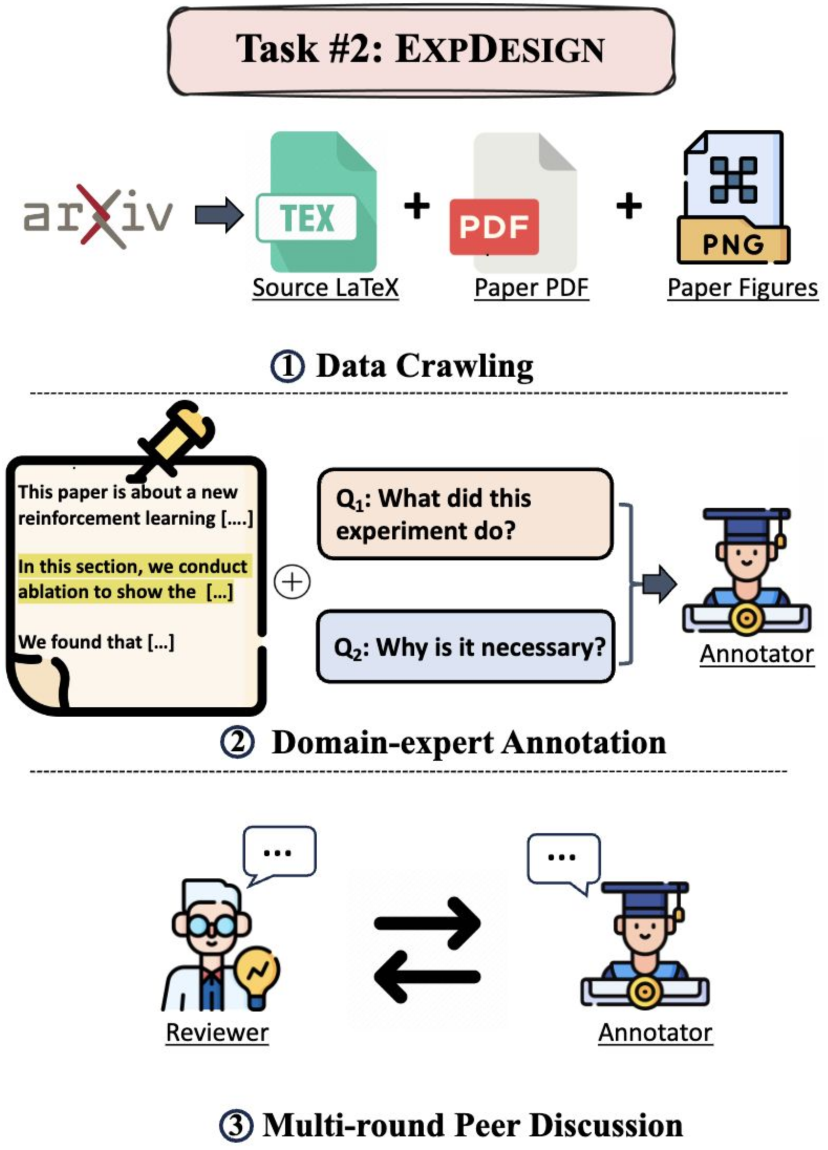
Input:

- Research background & motivation (usually the paper "abstract" and "introduction").

Output:

- **Experiment plan**: a list of suggested experiments
- **Motivation Explanation**: a list of explanations (one-one corresponding to the experiments).

Experiment Design: data collection



Task #2: EXPDESIGN
arXiv
TEX
Source LaTeX
+
PDF
Paper PDF
+
PNG
Paper Figures
① Data Crawling
This paper is about a new reinforcement learning [....]
In this section, we conduct ablation to show the [...]
We found that [...]
Q₁: What did this experiment do?
Q₂: Why is it necessary?
Annotator
② Domain-expert Annotation
...
...
Reviewer
Annotator
③ Multi-round Peer Discussion

1. Crawling source papers from arXiv:
   - Use source LaTeX is more precise than extracting the text from PDF.
   - Make sure all papers are peer-reviewed, i.e., accepted by well-known conferences.

2. Expert annotation:
   - Identify all the **necessary** experiments in the original paper
   - Summarize **what** and **Why.**

3. Peer discussion:
   - Another expert raises ambiguities (if any), e.g., necessity of experiments, and fallacy in the annotation.

Task#2: Experiment Design

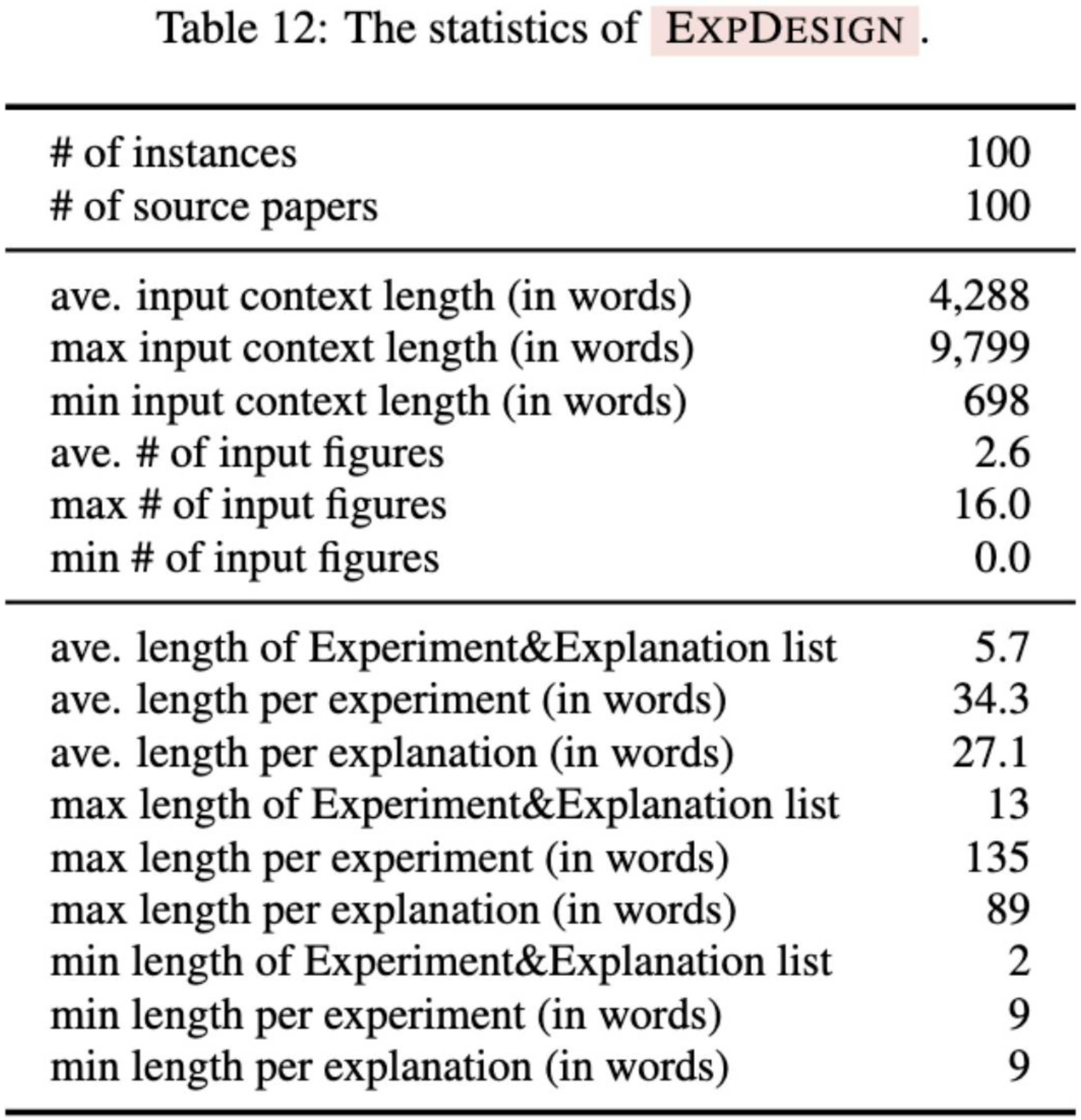
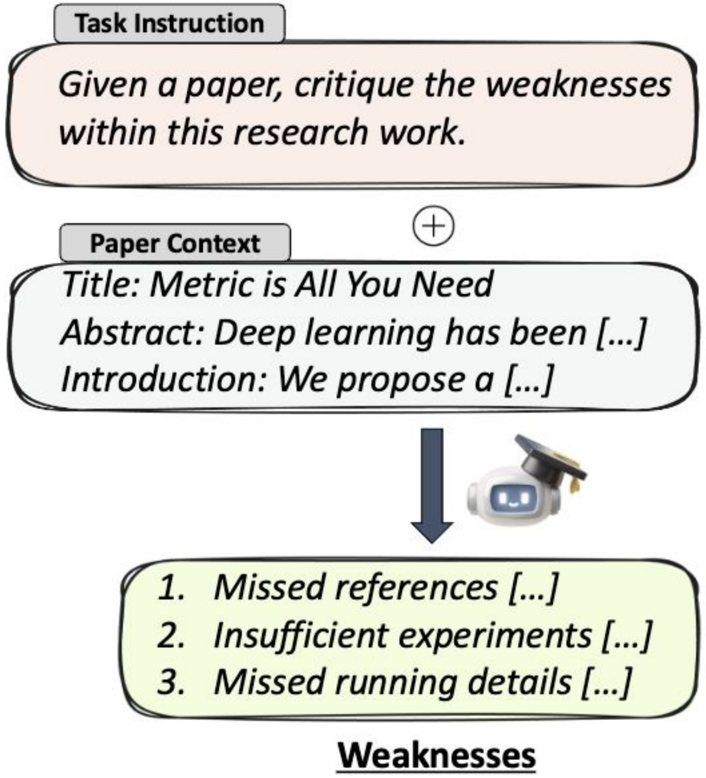


Table 12: The statistics of EXPDESIGN .

# of instances	100
# of source papers	100
ave. input context length (in words)	4,288
max input context length (in words)	9,799
min input context length (in words)	698
ave. # of input figures	2.6
max # of input figures	16.0
min # of input figures	0.0
ave. length of Experiment&Explanation list	5.7
ave. length per experiment (in words)	34.3
ave. length per explanation (in words)	27.1
max length of Experiment&Explanation list	13
max length per experiment (in words)	135
max length per explanation (in words)	89
min length of Experiment&Explanation list	2
min length per experiment (in words)	9
min length per explanation (in words)	9

Task#3: Paper Weakness



Task Instruction

Given a paper, critique the weaknesses within this research work.

Paper Context

Title: Metric is All You Need

Abstract: Deep learning has been […]

Introduction: We propose a […]

1. *Missed references […]*
2. *Insufficient experiments […]*
3. *Missed running details […]*

Weaknesses

Task #3: Paper Weakness

List generation: review the paper draft, and generate a list of weaknesses

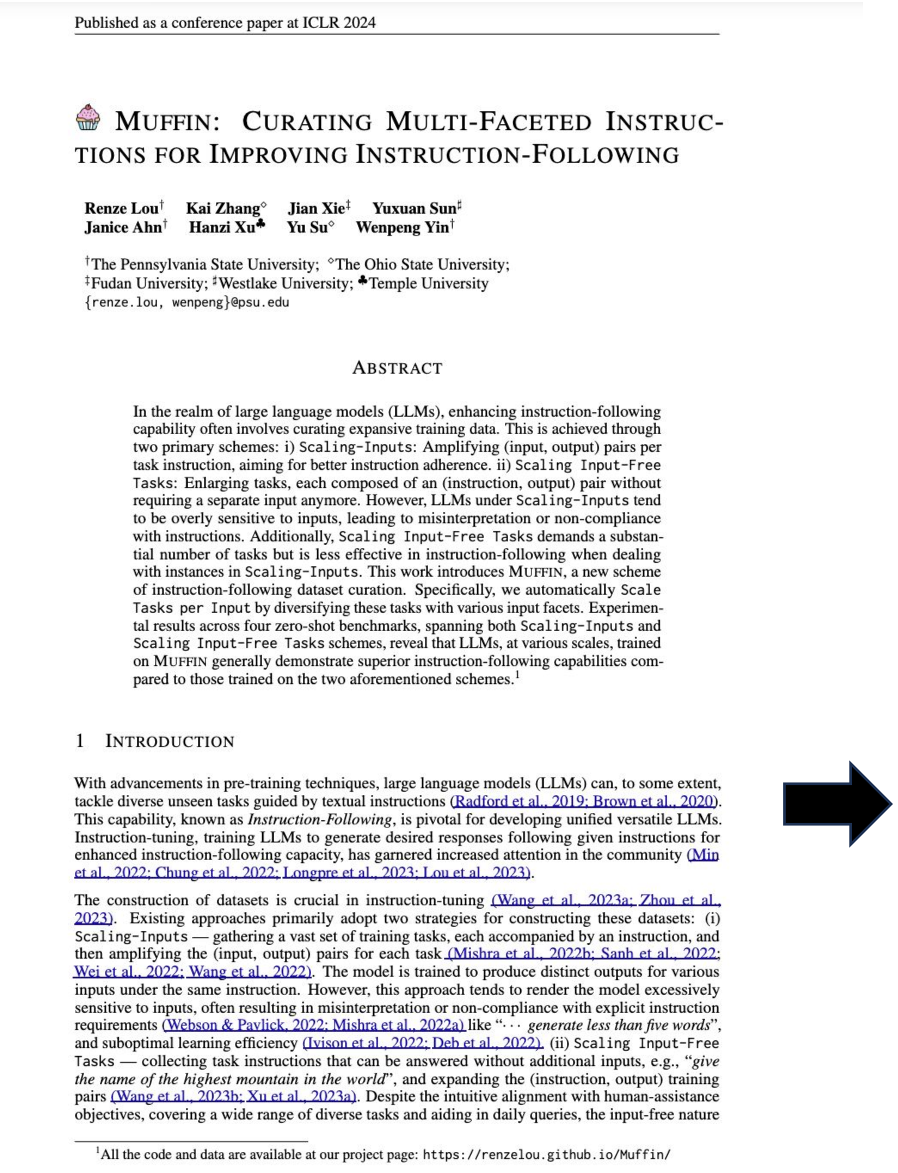
Input:

- Paper draft: the whole content of a paper (under review).

Output:

- Paper weaknesses: **multiple lists** of weaknesses.

Task#3: Paper Weakness



Published as a conference paper at ICLR 2024

MUFFIN: CURATING MULTI-FACETED INSTRUCTIONS FOR IMPROVING INSTRUCTION-FOLLOWING

Renze Lou[†] Kai Zhang[◇] Jian Xie[‡] Yuxuan Sun[♯]
Janice Ahn[†] Hanzi Xu[♣] Yu Su[◇] Wenpeng Yin[†]

[†]The Pennsylvania State University; [◇]The Ohio State University;
[‡]Fudan University; [♯]Westlake University; [♣]Temple University
{renze.lou, wenpeng}@psu.edu

ABSTRACT

In the realm of large language models (LLMs), enhancing instruction-following capability often involves curating expansive training data. This is achieved through two primary schemes: i) `Scaling-Inputs`: Amplifying (input, output) pairs per task instruction, aiming for better instruction adherence. ii) `Scaling Input-Free Tasks`: Enlarging tasks, each composed of an (instruction, output) pair without requiring a separate input anymore. However, LLMs under `Scaling-Inputs` tend to be overly sensitive to inputs, leading to misinterpretation or non-compliance with instructions. Additionally, `Scaling Input-Free Tasks` demands a substantial number of tasks but is less effective in instruction-following when dealing with instances in `Scaling-Inputs`. This work introduces MUFFIN, a new scheme of instruction-following dataset curation. Specifically, we automatically `Scale Tasks per Input` by diversifying these tasks with various input facets. Experimental results across four zero-shot benchmarks, spanning both `Scaling-Inputs` and `Scaling Input-Free Tasks` schemes, reveal that LLMs, at various scales, trained on MUFFIN generally demonstrate superior instruction-following capabilities compared to those trained on the two aforementioned schemes.[1]

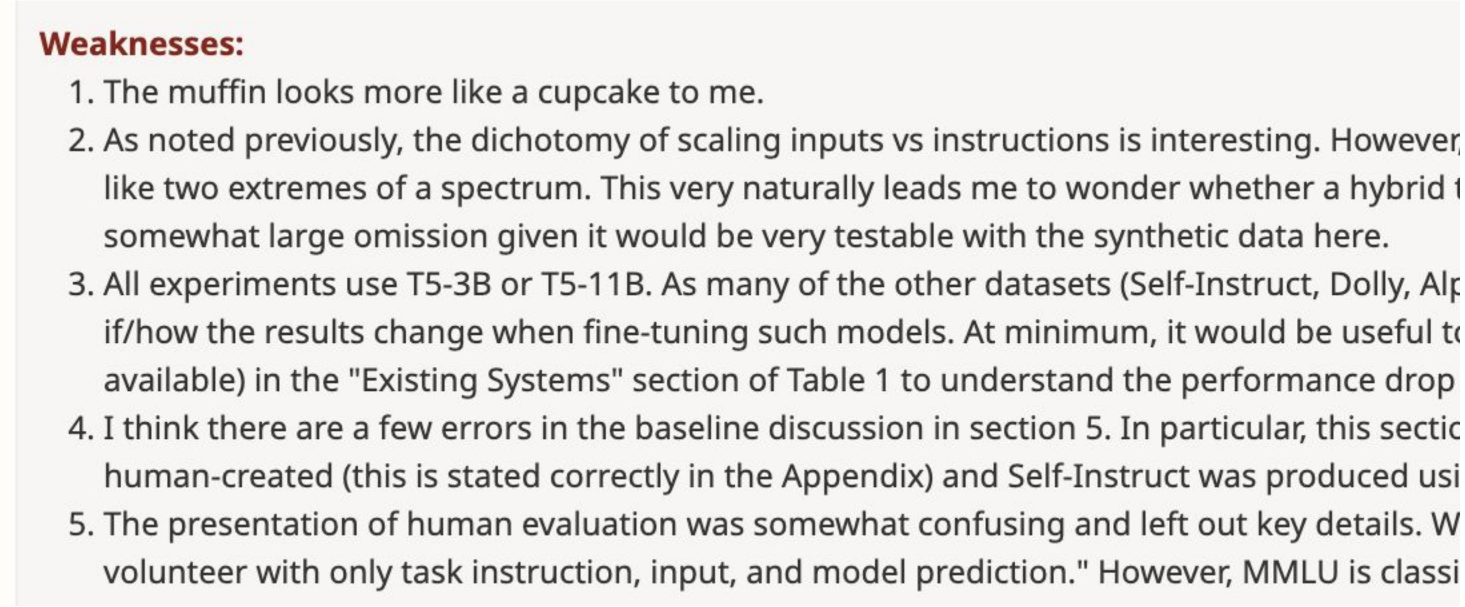
1 INTRODUCTION

With advancements in pre-training techniques, large language models (LLMs) can, to some extent, tackle diverse unseen tasks guided by textual instructions (Radford et al., 2019; Brown et al., 2020). This capability, known as *Instruction-Following*, is pivotal for developing unified versatile LLMs. Instruction-tuning, training LLMs to generate desired responses following given instructions for enhanced instruction-following capacity, has garnered increased attention in the community (Min et al., 2022; Chung et al., 2022; Longpre et al., 2023; Lou et al., 2023).

The construction of datasets is crucial in instruction-tuning (Wang et al., 2023a; Zhou et al., 2023). Existing approaches primarily adopt two strategies for constructing these datasets: (i) `Scaling-Inputs` — gathering a vast set of training tasks, each accompanied by an instruction, and then amplifying the (input, output) pairs for each task (Mishra et al., 2022b; Sanh et al., 2022; Wei et al., 2022; Wang et al., 2022). The model is trained to produce distinct outputs for various inputs under the same instruction. However, this approach tends to render the model excessively sensitive to inputs, often resulting in misinterpretation or non-compliance with explicit instruction requirements (Webson & Pavlick, 2022; Mishra et al., 2022a) like "··· *generate less than five words*", and suboptimal learning efficiency (Ivison et al., 2022; Deb et al., 2022). (ii) `Scaling Input-Free Tasks` — collecting task instructions that can be answered without additional inputs, e.g., "*give the name of the highest mountain in the world*", and expanding the (instruction, output) training pairs (Wang et al., 2023b; Xu et al., 2023a). Despite the intuitive alignment with human-assistance objectives, covering a wide range of diverse tasks and aiding in daily queries, the input-free nature

[1] All the code and data are available at our project page: https://renzelou.github.io/Muffin/

input

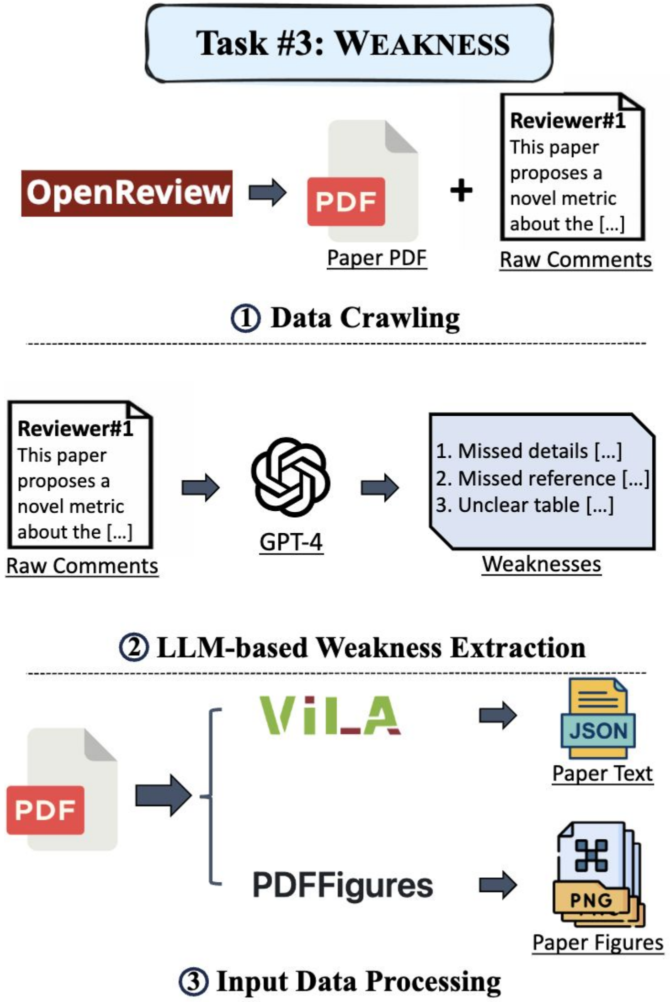


Weaknesses:

1. The muffin looks more like a cupcake to me.
2. As noted previously, the dichotomy of scaling inputs vs instructions is interesting. However, like two extremes of a spectrum. This very naturally leads me to wonder whether a hybrid t somewhat large omission given it would be very testable with the synthetic data here.
3. All experiments use T5-3B or T5-11B. As many of the other datasets (Self-Instruct, Dolly, Alp if/how the results change when fine-tuning such models. At minimum, it would be useful to available) in the "Existing Systems" section of Table 1 to understand the performance drop
4. I think there are a few errors in the baseline discussion in section 5. In particular, this sectio human-created (this is stated correctly in the Appendix) and Self-Instruct was produced usi
5. The presentation of human evaluation was somewhat confusing and left out key details. W volunteer with only task instruction, input, and model prediction." However, MMLU is classi

output

Paper Weakness: data collection

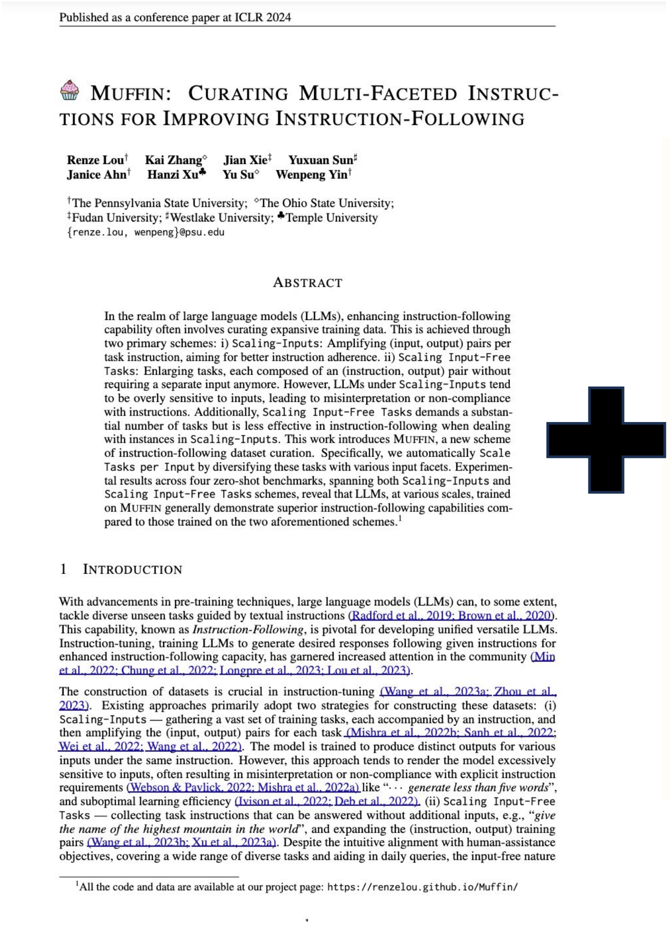


1. Crawling source papers from OpenReview:
   - We need under-review paper drafts (in PDF), along with the reviewer’s comments.

2. Weakness extraction:
   - Utilize LLMs to extract weaknesses from the raw comments (keep the human text) and formulate all the points as a list.

3. Paper PDF processing:
   - Process source paper PDF into text/figures.

Task#4: Review Critique



Published as a conference paper at ICLR 2024

MUFFIN: CURATING MULTI-FACETED INSTRUCTIONS FOR IMPROVING INSTRUCTION-FOLLOWING

Renze Lou[†] Kai Zhang[◇] Jian Xie[‡] Yuxuan Sun[♯]
Janice Ahn[†] Hanzi Xu[♣] Yu Su[◇] Wenpeng Yin[†]

[†]The Pennsylvania State University; [◇]The Ohio State University;
[‡]Fudan University; [♯]Westlake University; [♣]Temple University
{renze.lou, wenpeng}@psu.edu

ABSTRACT

In the realm of large language models (LLMs), enhancing instruction-following capability often involves curating expansive training data. This is achieved through two primary schemes: i) `Scaling-Inputs`: Amplifying (input, output) pairs per task instruction, aiming for better instruction adherence. ii) `Scaling Input-Free Tasks`: Enlarging tasks, each composed of an (instruction, output) pair without requiring a separate input anymore. However, LLMs under `Scaling-Inputs` tend to be overly sensitive to inputs, leading to misinterpretation or non-compliance with instructions. Additionally, `Scaling Input-Free Tasks` demands a substantial number of tasks but is less effective in instruction-following when dealing with instances in `Scaling-Inputs`. This work introduces MUFFIN, a new scheme of instruction-following dataset curation. Specifically, we automatically `Scale Tasks per Input` by diversifying these tasks with various input facets. Experimental results across four zero-shot benchmarks, spanning both `Scaling-Inputs` and `Scaling Input-Free Tasks` schemes, reveal that LLMs, at various scales, trained on MUFFIN generally demonstrate superior instruction-following capabilities compared to those trained on the two aforementioned schemes.[1]

1 INTRODUCTION

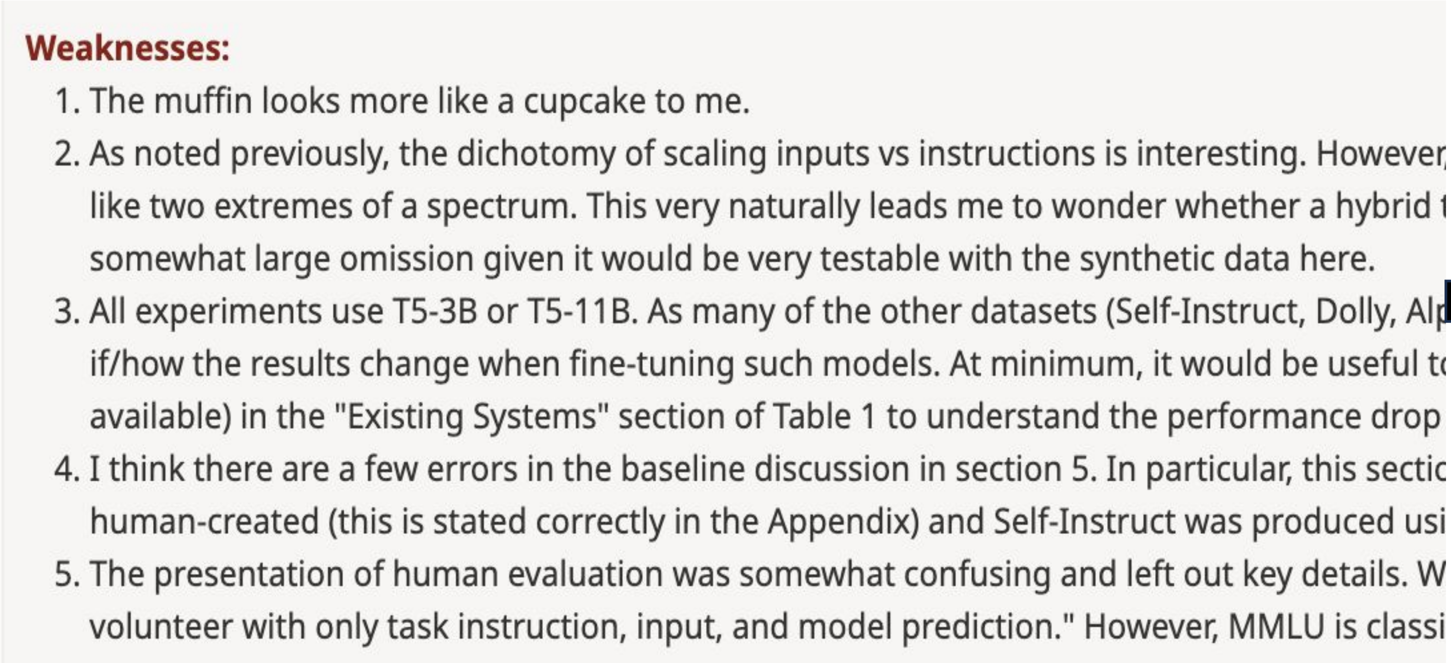
With advancements in pre-training techniques, large language models (LLMs) can, to some extent, tackle diverse unseen tasks guided by textual instructions (Radford et al., 2019; Brown et al., 2020). This capability, known as *Instruction-Following*, is pivotal for developing unified versatile LLMs. Instruction-tuning, training LLMs to generate desired responses following given instructions for enhanced instruction-following capacity, has garnered increased attention in the community (Min et al., 2022; Chung et al., 2022; Longpre et al., 2023; Lou et al., 2023).

The construction of datasets is crucial in instruction-tuning (Wang et al., 2023a; Zhou et al., 2023). Existing approaches primarily adopt two strategies for constructing these datasets: (i) `Scaling-Inputs` — gathering a vast set of training tasks, each accompanied by an instruction, and then amplifying the (input, output) pairs for each task (Mishra et al., 2022b; Sanh et al., 2022; Wei et al., 2022; Wang et al., 2022). The model is trained to produce distinct outputs for various inputs under the same instruction. However, this approach tends to render the model excessively sensitive to inputs, often resulting in misinterpretation or non-compliance with explicit instruction requirements (Webson & Pavlick, 2022; Mishra et al., 2022a) like "··· *generate less than five words*", and suboptimal learning efficiency (Ivison et al., 2022; Deb et al., 2022). (ii) `Scaling Input-Free Tasks` — collecting task instructions that can be answered without additional inputs, e.g., "*give the name of the highest mountain in the world*", and expanding the (instruction, output) training pairs (Wang et al., 2023b; Xu et al., 2023a). Despite the intuitive alignment with human-assistance objectives, covering a wide range of diverse tasks and aiding in daily queries, the input-free nature

[1] All the code and data are available at our project page: https://renzelou.github.io/Muffin/

input

+



Weaknesses:

1. The muffin looks more like a cupcake to me.
2. As noted previously, the dichotomy of scaling inputs vs instructions is interesting. However, like two extremes of a spectrum. This very naturally leads me to wonder whether a hybrid t somewhat large omission given it would be very testable with the synthetic data here.
3. All experiments use T5-3B or T5-11B. As many of the other datasets (Self-Instruct, Dolly, Alp if/how the results change when fine-tuning such models. At minimum, it would be useful to available) in the "Existing Systems" section of Table 1 to understand the performance drop
4. I think there are a few errors in the baseline discussion in section 5. In particular, this sectic human-created (this is stated correctly in the Appendix) and Self-Instruct was produced usi
5. The presentation of human evaluation was somewhat confusing and left out key details. W volunteer with only task instruction, input, and model prediction." However, MMLU is classi

✗
✓
✗
✓
✓

output

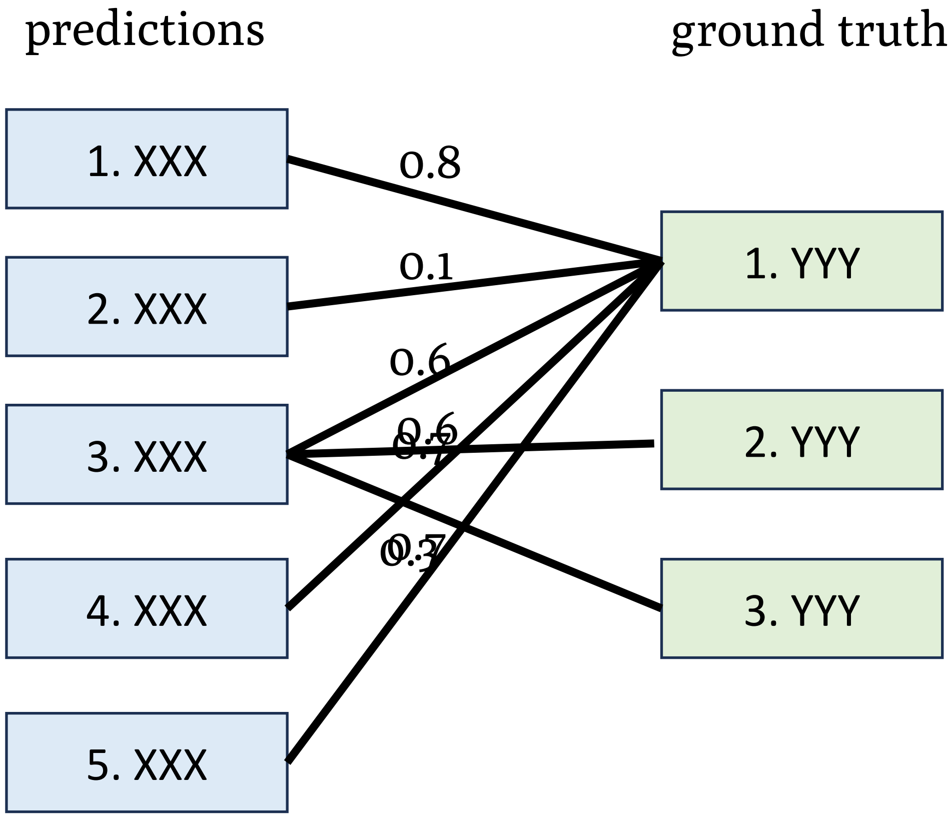
Evaluation Metrics

🤔Key challenge: how to measure the overlap between two lists?

- Semantic-based score
- Information entailment score (LLM-as-judge)

Semantic-based score

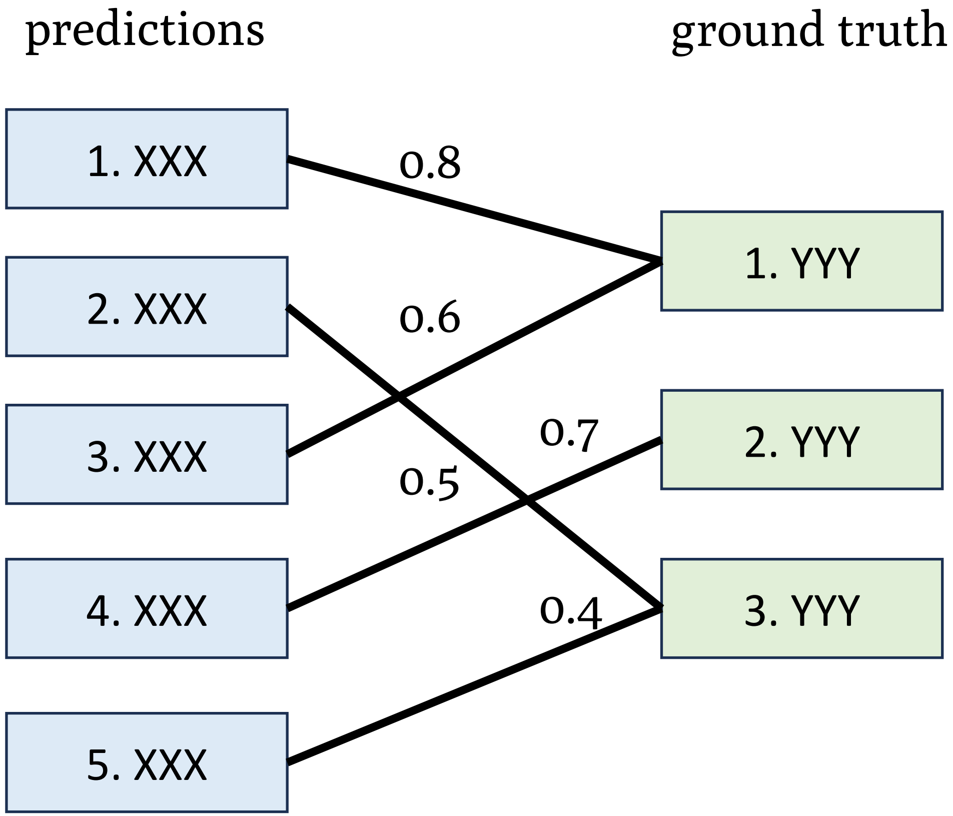
“Soft” version of F1 score: calculate the semantic similarity among the items from both lists



S-Recall: (0.8+0.6+0.7) / 3 = 0.7

Semantic-based score

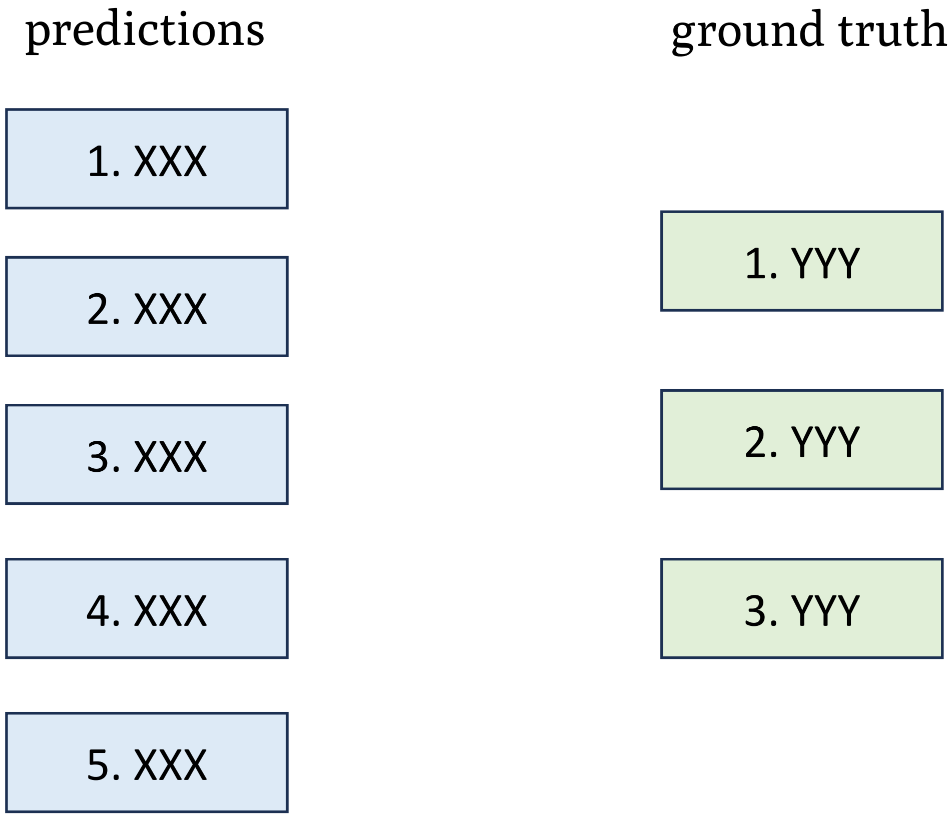
“Soft” version of F1 score: calculate the semantic similarity among the items from both lists



S-Precision: (0.8+0.6+0.7+0.5+0.4) / 5 = 0.6

Semantic-based score

“Soft” version of F1 score: calculate the semantic similarity among the items from both lists



predictions

1. XXX
2. XXX
3. XXX
4. XXX
5. XXX

ground truth

1. YYY
2. YYY
3. YYY

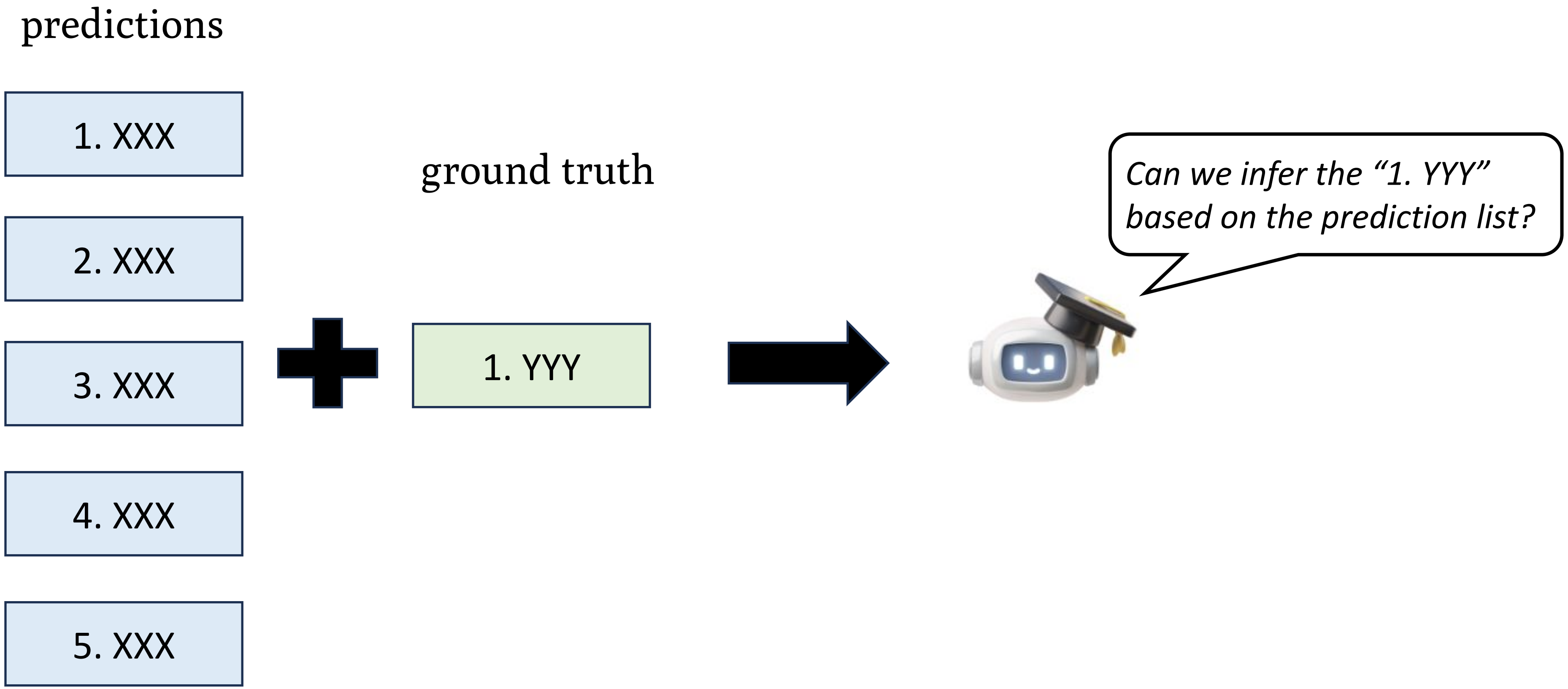
S-Recall: (0.8+0.6+0.7) / 3 = 0.7

S-Precision: (0.8+0.6+0.7+0.5+0.4) / 5 = 0.6

S-F1: 2*R*P / (R+P) = 0.65

Information entailment score (LLM-as-judge)

F1 score via LLM-as-judge: use LLM as the evaluator instead of using semantic similarity.



predictions

1. XXX

2. XXX

3. XXX

4. XXX

5. XXX

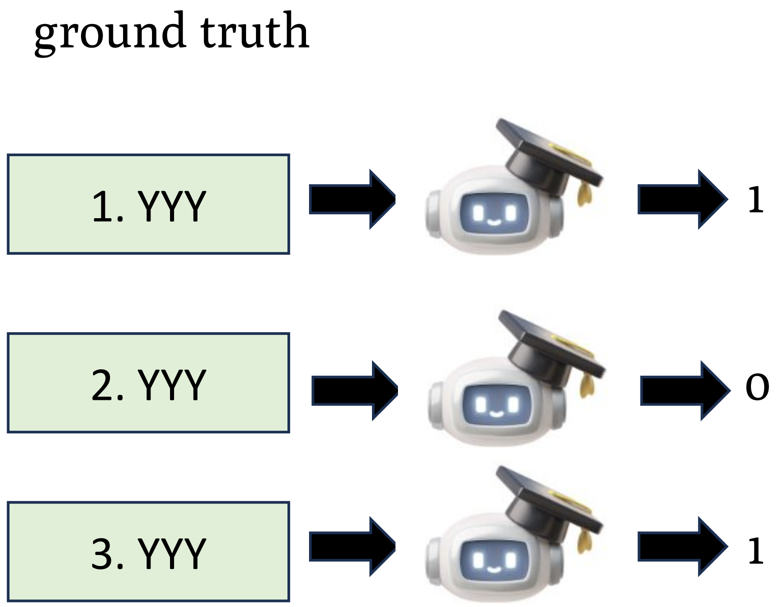
ground truth

1. YYY

Can we infer the "1. YYY" based on the prediction list?

Information entailment score (LLM-as-judge)

F1 score via LLM-as-judge: use LLM as the evaluator instead of using semantic similarity.



ground truth

1. YYY → 1

2. YYY → 0

3. YYY → 1

En-Recall: (1+0+1) / 3 = 0.67

En-Precision = ...

En-F1 = ...

inter-/intra list diversity

TODO: currently skip it, cuz the ITF-IDF is not from this project. also don't have time to introduce all the details (15 mins talk).

Performances & Results:

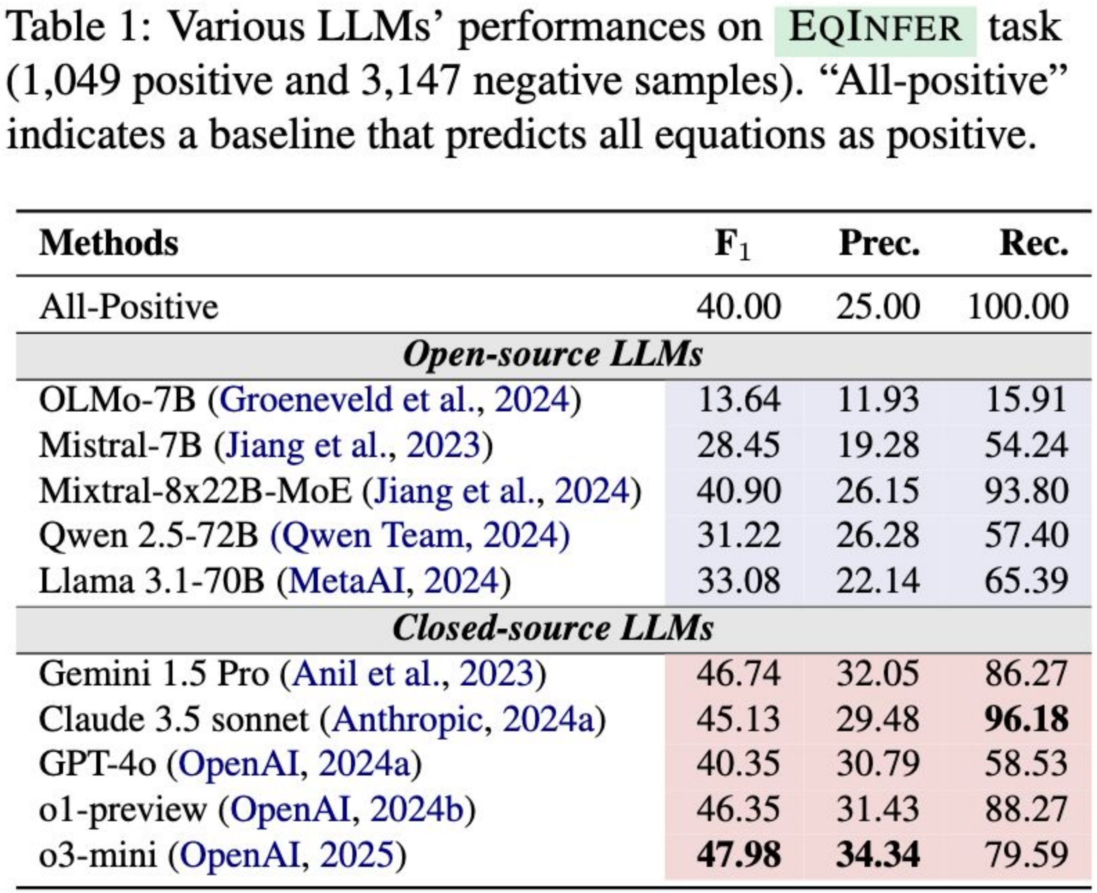


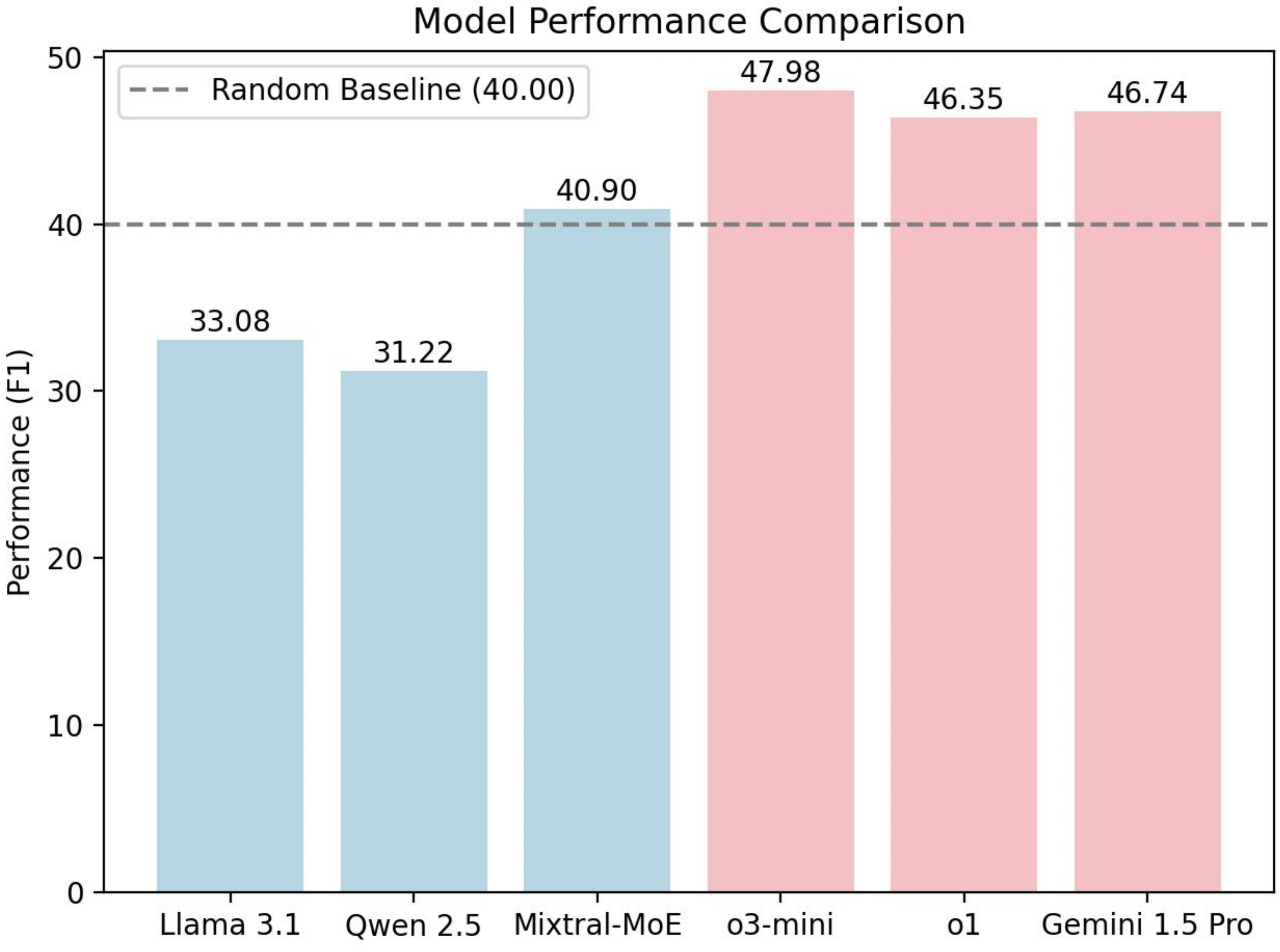
Table 1: Various LLMs' performances on EQINFER task (1,049 positive and 3,147 negative samples). "All-positive" indicates a baseline that predicts all equations as positive.

Methods	F_1	Prec.	Rec.
All-Positive	40.00	25.00	100.00
Open-source LLMs			
OLMo-7B (Groeneveld et al., 2024)	13.64	11.93	15.91
Mistral-7B (Jiang et al., 2023)	28.45	19.28	54.24
Mixtral-8x22B-MoE (Jiang et al., 2024)	40.90	26.15	93.80
Qwen 2.5-72B (Qwen Team, 2024)	31.22	26.28	57.40
Llama 3.1-70B (MetaAI, 2024)	33.08	22.14	65.39
Closed-source LLMs			
Gemini 1.5 Pro (Anil et al., 2023)	46.74	32.05	86.27
Claude 3.5 sonnet (Anthropic, 2024a)	45.13	29.48	**96.18**
GPT-4o (OpenAI, 2024a)	40.35	30.79	58.53
o1-preview (OpenAI, 2024b)	46.35	31.43	88.27
o3-mini (OpenAI, 2025)	**47.98**	**34.34**	79.59

- A simple baseline that predicts all equations as positive: achieves 40% F1 (due to the data distribution)

- Compared to the All-Positive baseline, the performance superiority of the strong close-source LLMs is not significant; the best LLM on this task only obtains 47.98%.

Performances & Results:



Equation Inference

- The strongest close-source LLMs on this task only obtained 47.98%.
- Some models are actually random guessing.

Observations:

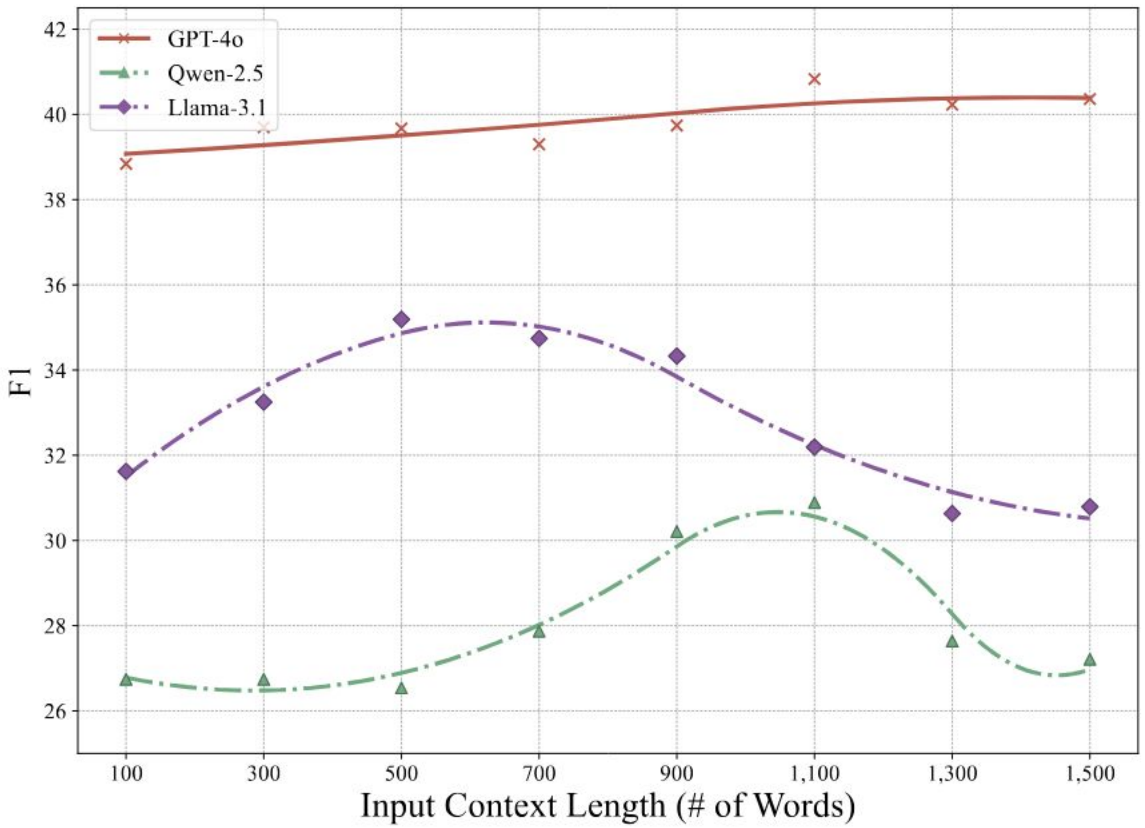


Figure 4: The input context length scaling trend on the EQINFER task.

- For some open-source LLMs: an appropriate context length can boost the performance;

- While not for those strong close-source LLMs.

Performances & Results:

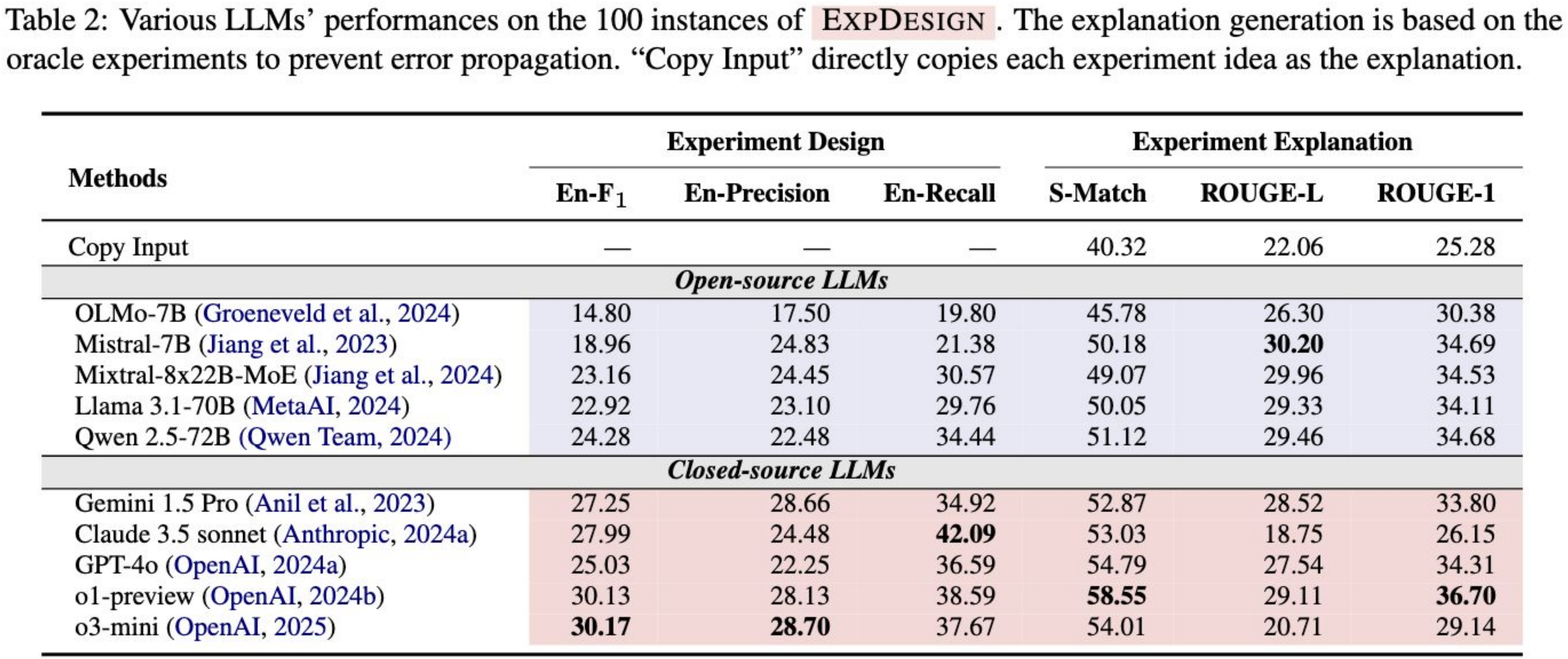
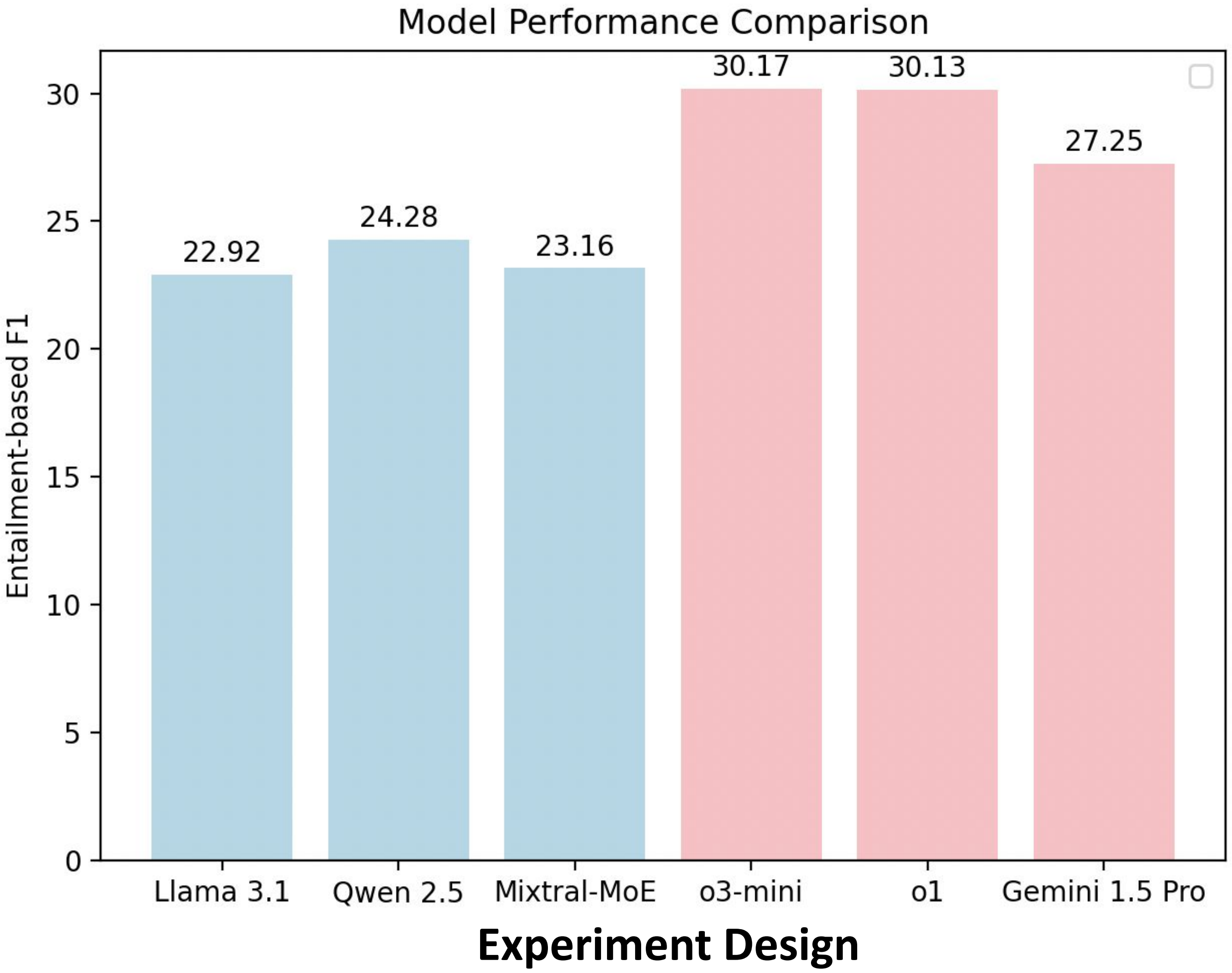


Table 2: Various LLMs' performances on the 100 instances of EXPDESIGN . The explanation generation is based on the oracle experiments to prevent error propagation. "Copy Input" directly copies each experiment idea as the explanation.

Methods	Experiment Design			Experiment Explanation		
	En-F_1	En-Precision	En-Recall	S-Match	ROUGE-L	ROUGE-1
Copy Input	—	—	—	40.32	22.06	25.28
Open-source LLMs						
OLMo-7B (Groeneveld et al., 2024)	14.80	17.50	19.80	45.78	26.30	30.38
Mistral-7B (Jiang et al., 2023)	18.96	24.83	21.38	50.18	**30.20**	34.69
Mixtral-8x22B-MoE (Jiang et al., 2024)	23.16	24.45	30.57	49.07	29.96	34.53
Llama 3.1-70B (MetaAI, 2024)	22.92	23.10	29.76	50.05	29.33	34.11
Qwen 2.5-72B (Qwen Team, 2024)	24.28	22.48	34.44	51.12	29.46	34.68
Closed-source LLMs						
Gemini 1.5 Pro (Anil et al., 2023)	27.25	28.66	34.92	52.87	28.52	33.80
Claude 3.5 sonnet (Anthropic, 2024a)	27.99	24.48	**42.09**	53.03	18.75	26.15
GPT-4o (OpenAI, 2024a)	25.03	22.25	36.59	54.79	27.54	34.31
o1-preview (OpenAI, 2024b)	30.13	28.13	38.59	**58.55**	29.11	**36.70**
o3-mini (OpenAI, 2025)	**30.17**	**28.70**	37.67	54.01	20.71	29.14

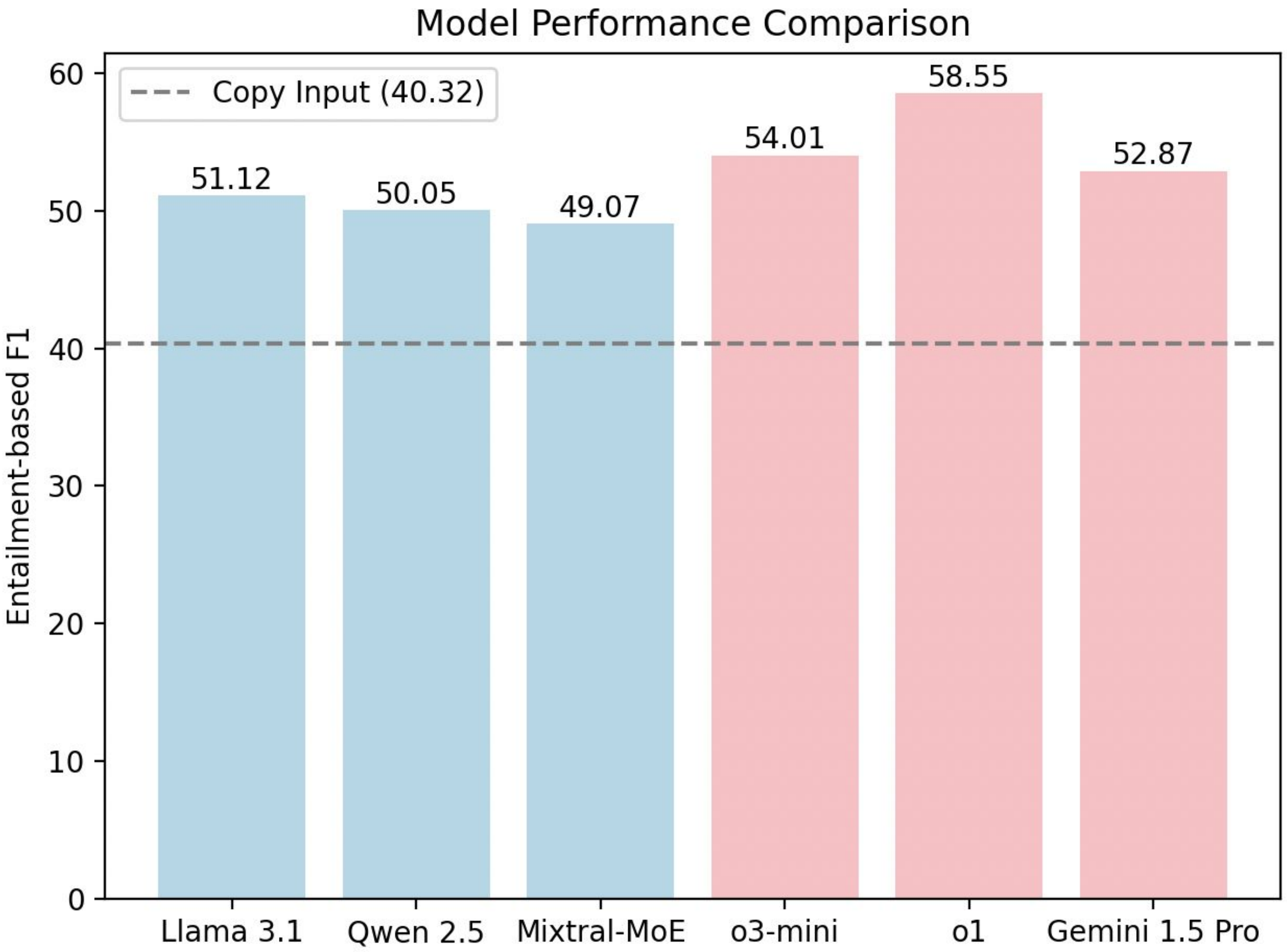
- Experiment Design: LLMs consistently miss ground-truth experiments from the origin paper (**low recall**), and they tend to generate more novel experiments that didn't show in the origin paper (**low precision**).
- Experiment Explanation: negative correlation between S-Match and ROUGE score. Simply copying input can get a relatively high ROUGE --- the importance of adapting the proposed semantic-based metrics.

Performances & Results:



- LLMs consistently miss ground-truth experiments from the origin paper (**low recall**);
- LLMs tend to generate more novel experiments that didn’t show in the origin paper (**low precision**); some of them are useful though.

Performances & Results:



Explanation Generation

- **Oracle setting**: ground truth experiments are given.
- The explanation itself is easier, but we have to consider the error propagation.

Observations:

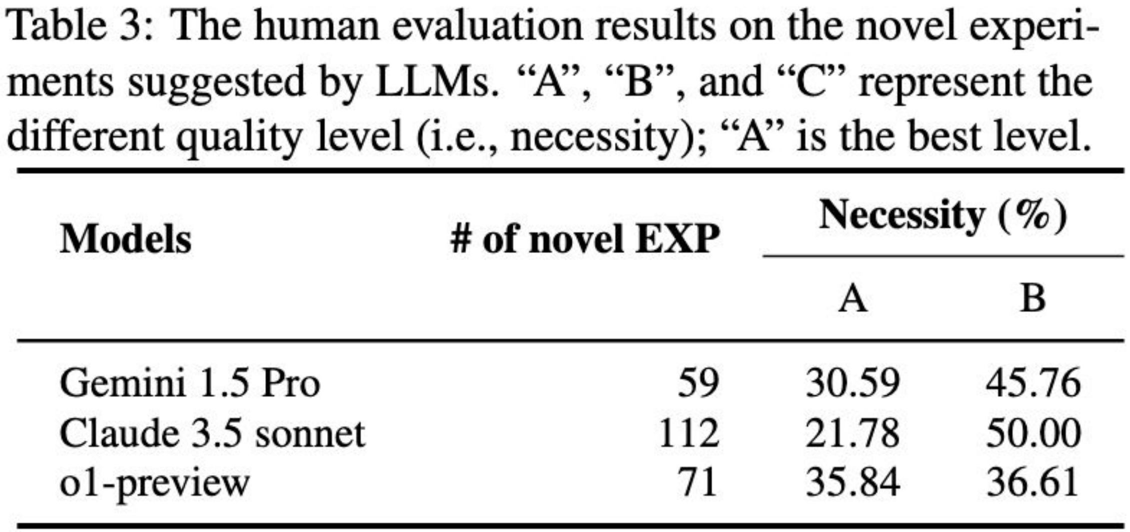


Table 3: The human evaluation results on the novel experiments suggested by LLMs. “A”, “B”, and “C” represent the different quality level (i.e., necessity); “A” is the best level.

Models	**# of novel EXP**	**Necessity (%)**	
		A	B
Gemini 1.5 Pro	59	30.59	45.76
Claude 3.5 sonnet	112	21.78	50.00
o1-preview	71	35.84	36.61

Positive evidence of using LLMs in assisting experiment design:

- LLMs can generate a considerable amount of **novel experiments**.
- Some of them are found to be really helpful and can be treated **as complementary experiments** with human experiments.

Observations:

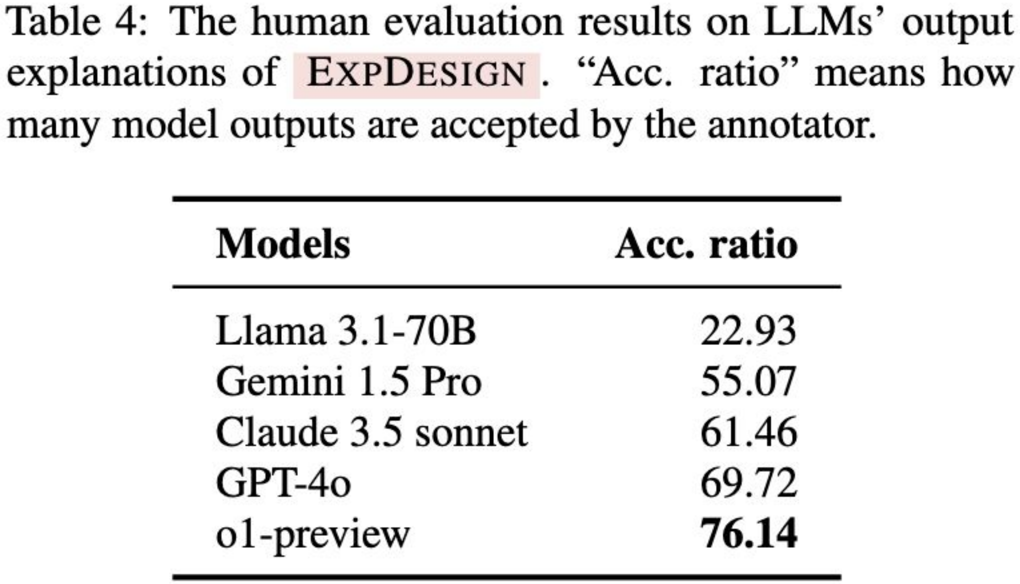


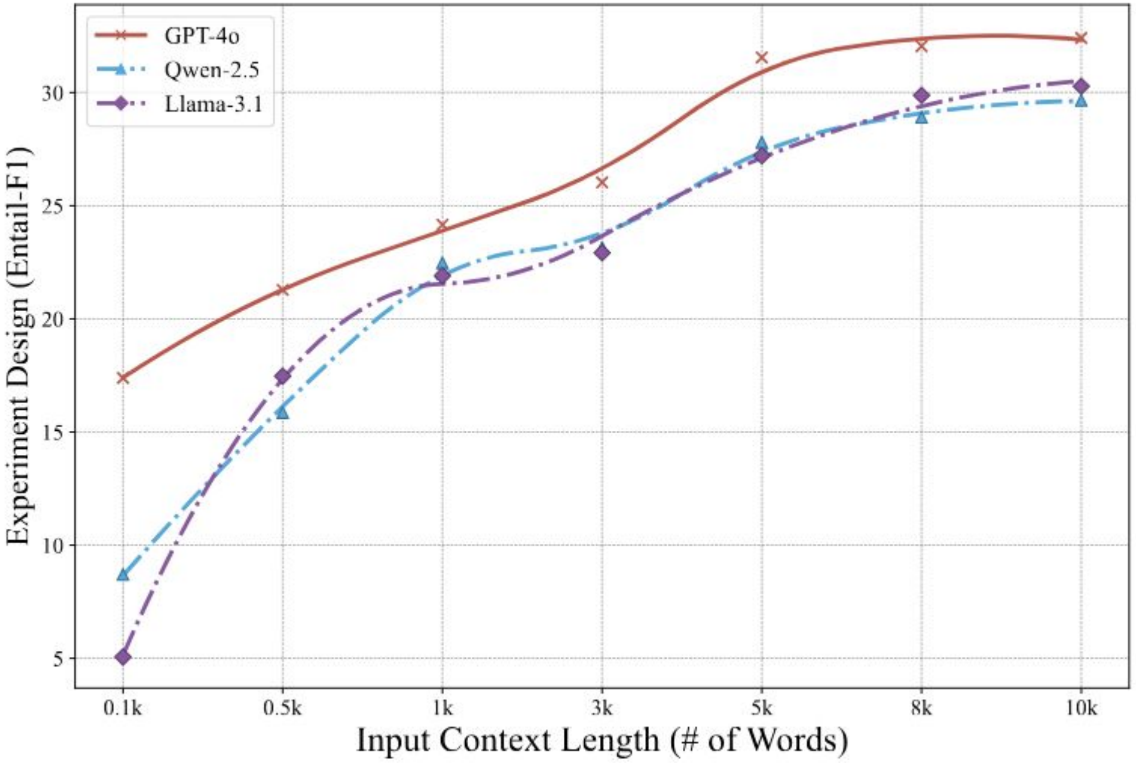
Table 4: The human evaluation results on LLMs' output explanations of EXPDESIGN . "Acc. ratio" means how many model outputs are accepted by the annotator.

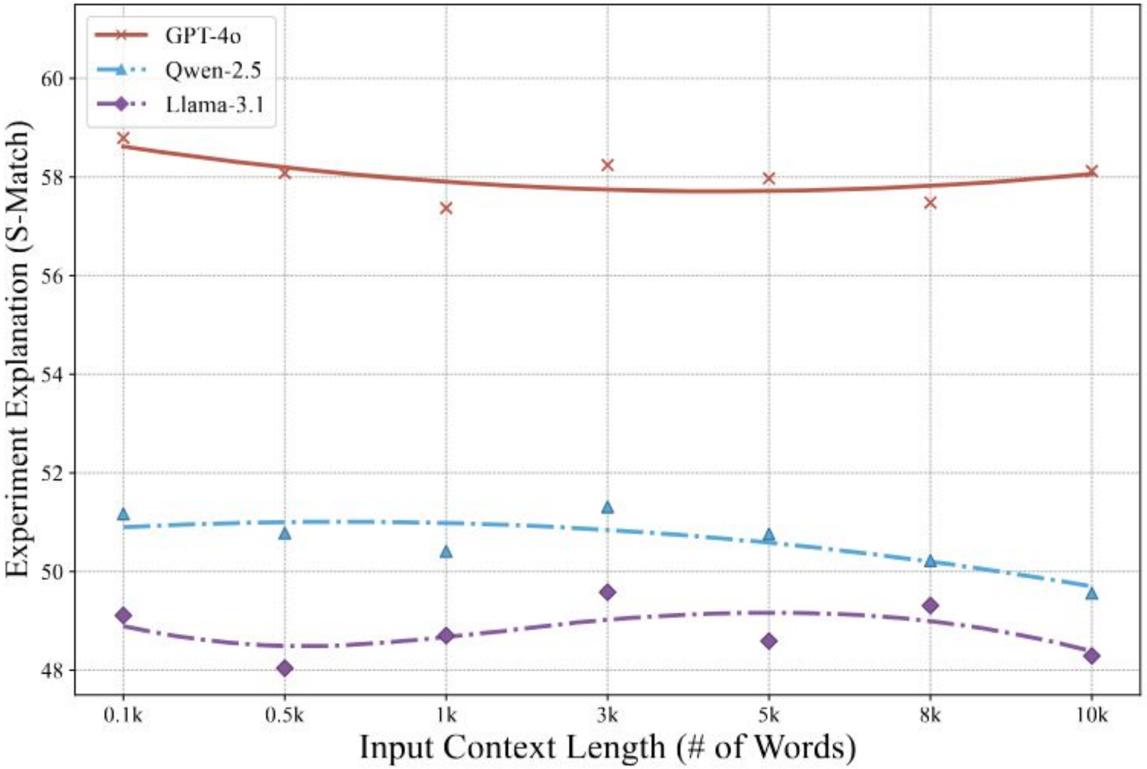
Models	Acc. ratio
Llama 3.1-70B	22.93
Gemini 1.5 Pro	55.07
Claude 3.5 sonnet	61.46
GPT-4o	69.72
o1-preview	**76.14**

The usefulness of the proposed S-Match metric:

- **Perfect correlation** between the S-Match scores and the human evaluation score.

Observations:





The input context (background information) can help experiment design (to some degree), while not for the explanation generation.

Observations:

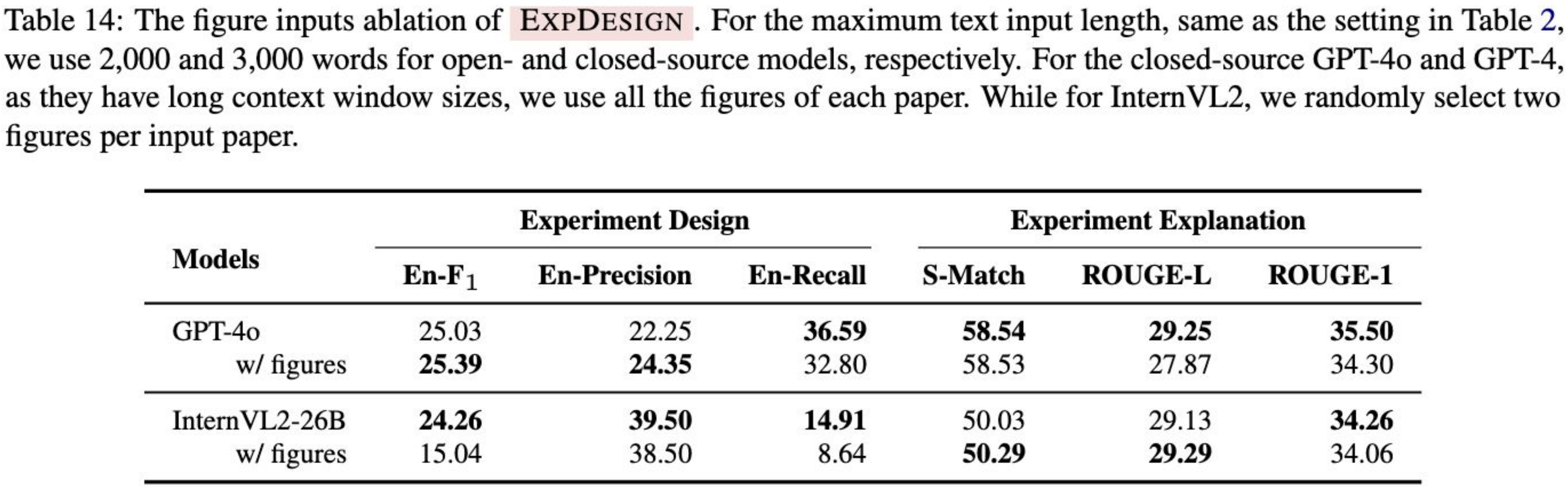


Table 14: The figure inputs ablation of ExpDesign . For the maximum text input length, same as the setting in Table 2, we use 2,000 and 3,000 words for open- and closed-source models, respectively. For the closed-source GPT-4o and GPT-4, as they have long context window sizes, we use all the figures of each paper. While for InternVL2, we randomly select two figures per input paper.

Models	**Experiment Design**			**Experiment Explanation**		
	En-F$_1$	**En-Precision**	**En-Recall**	**S-Match**	**ROUGE-L**	**ROUGE-1**
GPT-4o	25.03	22.25	**36.59**	**58.54**	**29.25**	**35.50**
w/ figures	**25.39**	**24.35**	32.80	58.53	27.87	34.30
InternVL2-26B	**24.26**	**39.50**	**14.91**	50.03	29.13	**34.26**
w/ figures	15.04	38.50	8.64	**50.29**	**29.29**	34.06

- Paper figures don't contribute much to the performance.
- It's hard for the current LMMs to leverage the rich multi-modal information from a scientific paper.

Performances & Results:

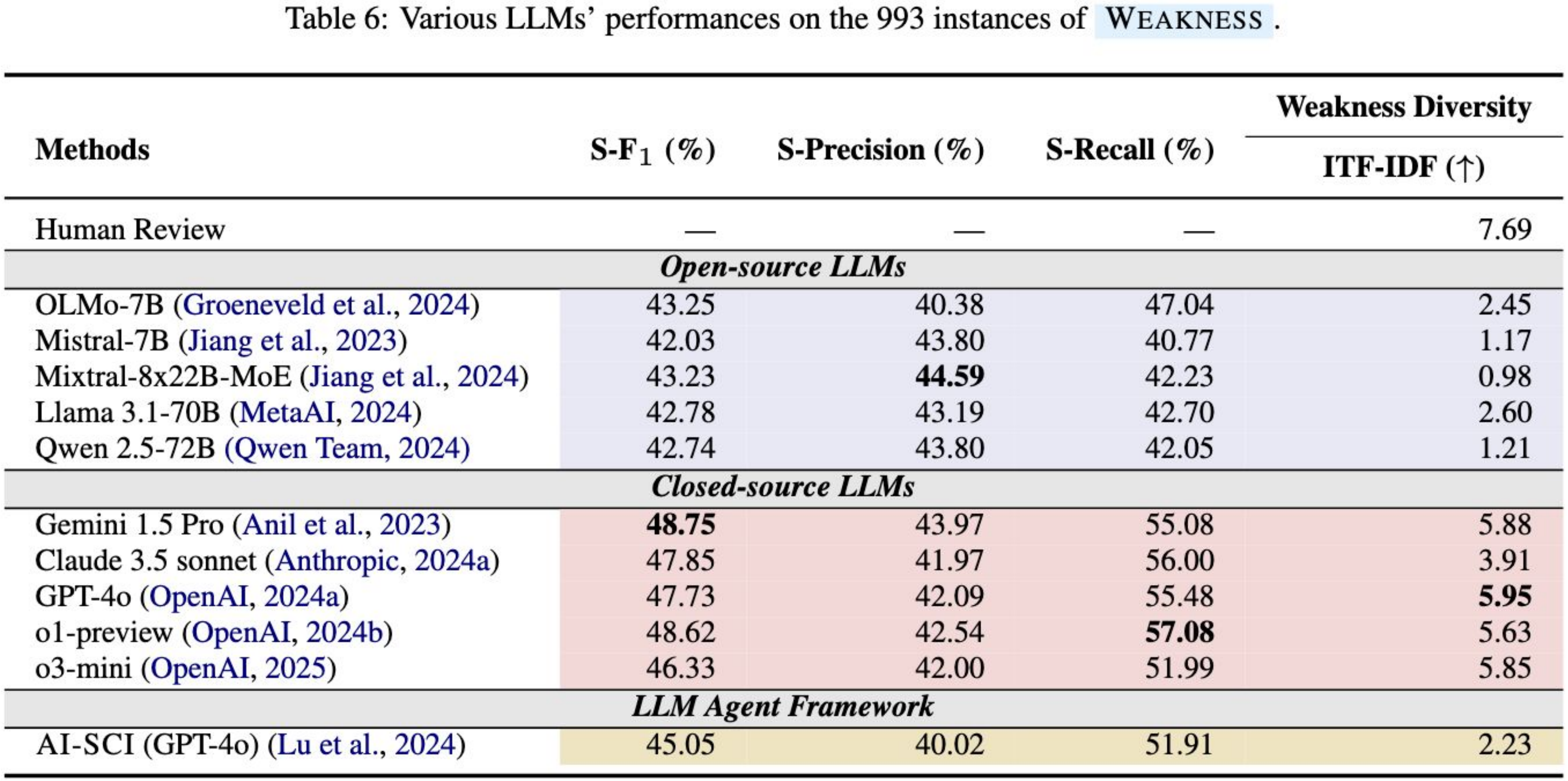
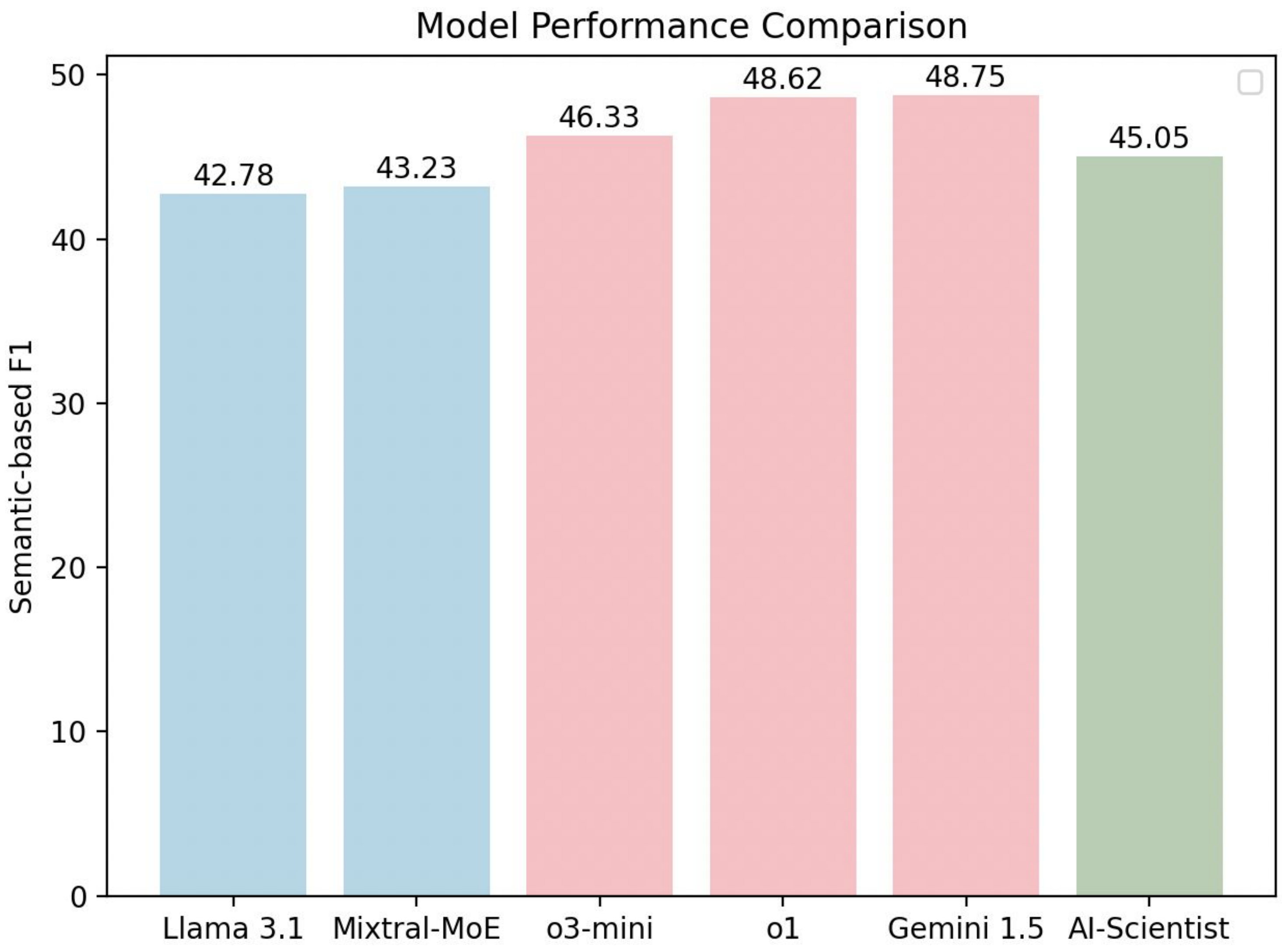


Table 6: Various LLMs' performances on the 993 instances of WEAKNESS .

Methods	$S\text{-}F_1$ (%)	S-Precision (%)	S-Recall (%)	Weakness Diversity
				ITF-IDF (↑)
Human Review	—	—	—	7.69
Open-source LLMs				
OLMo-7B (Groeneveld et al., 2024)	43.25	40.38	47.04	2.45
Mistral-7B (Jiang et al., 2023)	42.03	43.80	40.77	1.17
Mixtral-8x22B-MoE (Jiang et al., 2024)	43.23	**44.59**	42.23	0.98
Llama 3.1-70B (MetaAI, 2024)	42.78	43.19	42.70	2.60
Qwen 2.5-72B (Qwen Team, 2024)	42.74	43.80	42.05	1.21
Closed-source LLMs				
Gemini 1.5 Pro (Anil et al., 2023)	**48.75**	43.97	55.08	5.88
Claude 3.5 sonnet (Anthropic, 2024a)	47.85	41.97	56.00	3.91
GPT-4o (OpenAI, 2024a)	47.73	42.09	55.48	**5.95**
o1-preview (OpenAI, 2024b)	48.62	42.54	**57.08**	5.63
o3-mini (OpenAI, 2025)	46.33	42.00	51.99	5.85
LLM Agent Framework				
AI-SCI (GPT-4o) (Lu et al., 2024)	45.05	40.02	51.91	2.23

- Compared with human review, most LLM-generated weaknesses are vague and lack the necessary knowledge about some frontier research works.

Performances & Results:



Paper Weaknesses

- Compared with human review, most LLM-generated weaknesses are vague and lack the necessary knowledge about some frontier research works.

Observations:

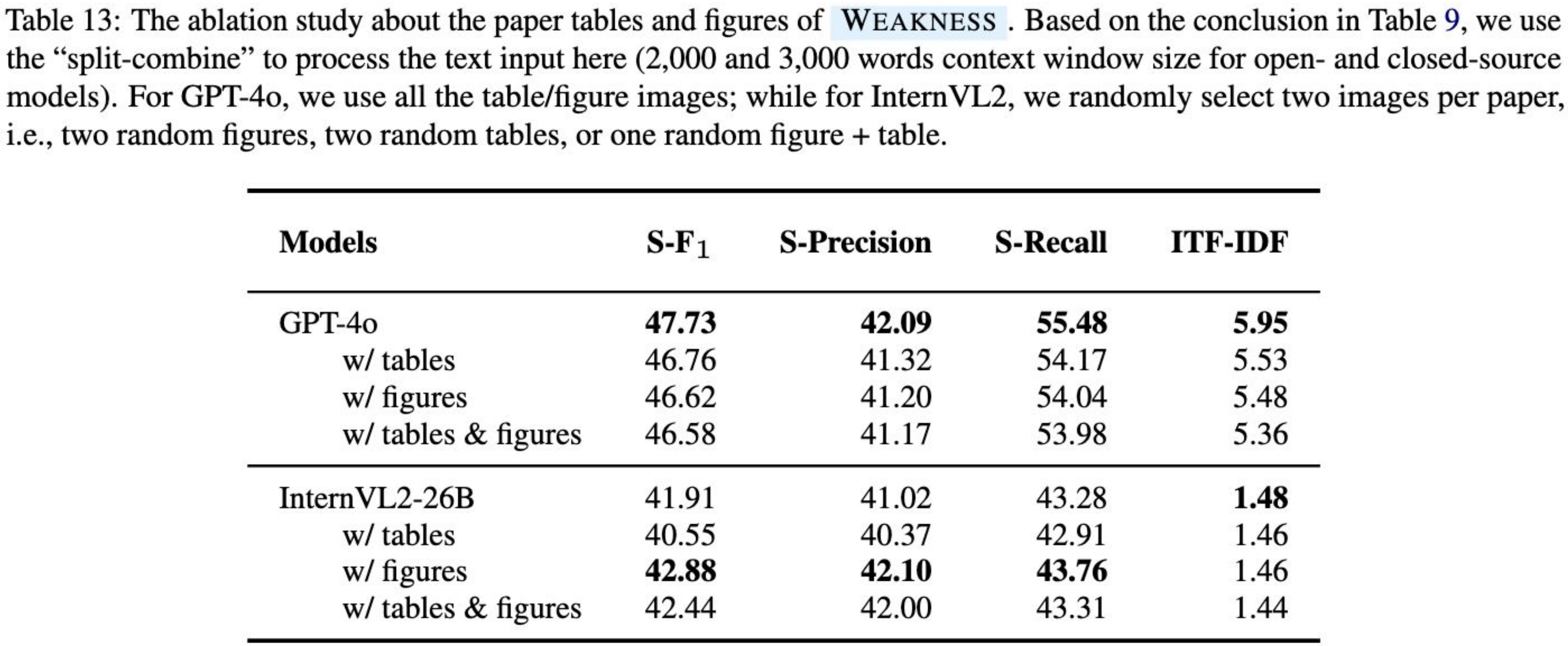
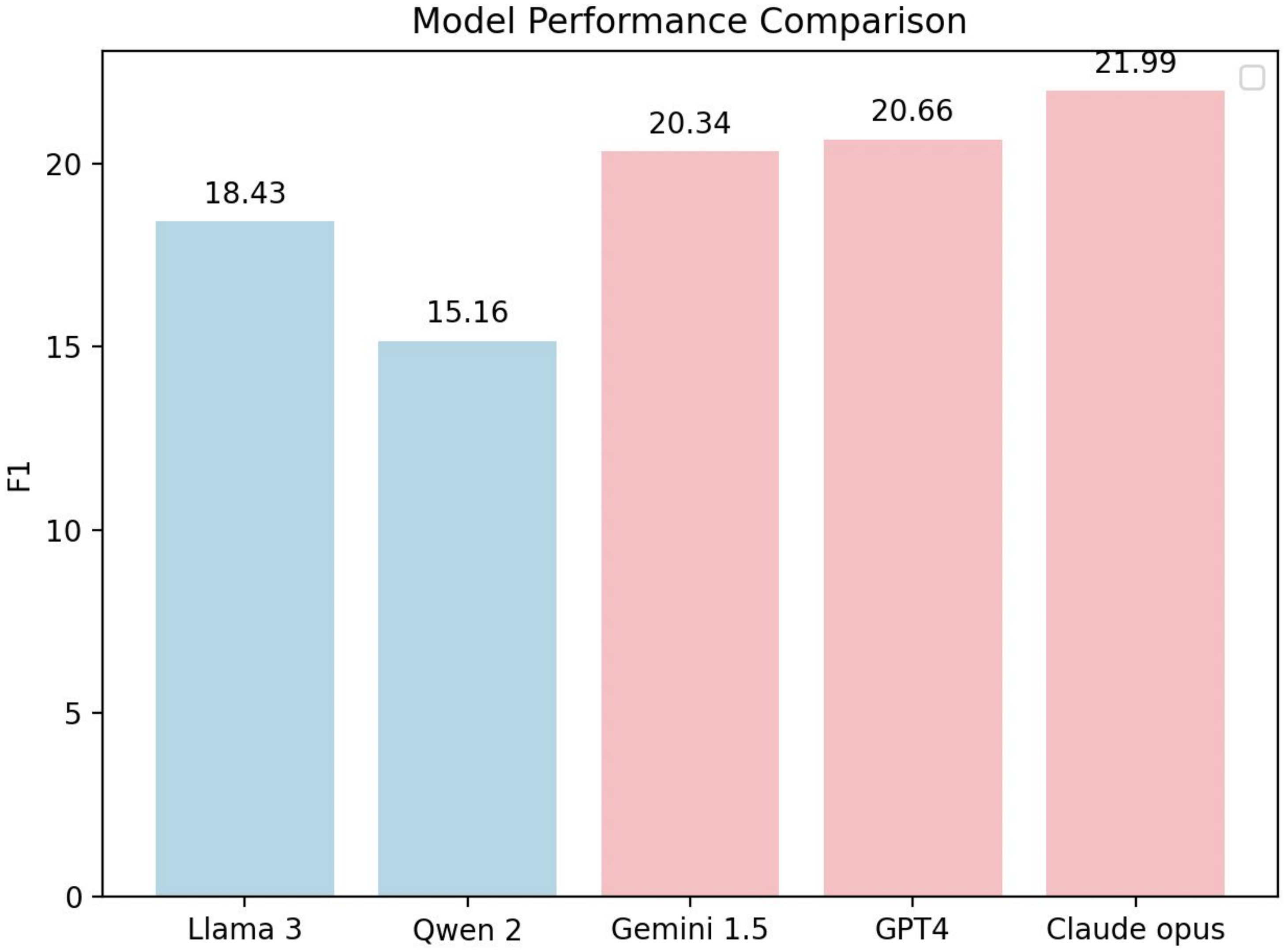


Table 13: The ablation study about the paper tables and figures of WEAKNESS . Based on the conclusion in Table 9, we use the "split-combine" to process the text input here (2,000 and 3,000 words context window size for open- and closed-source models). For GPT-4o, we use all the table/figure images; while for InternVL2, we randomly select two images per paper, i.e., two random figures, two random tables, or one random figure + table.

Models	S-F_1	S-Precision	S-Recall	ITF-IDF
GPT-4o	**47.73**	**42.09**	**55.48**	**5.95**
w/ tables	46.76	41.32	54.17	5.53
w/ figures	46.62	41.20	54.04	5.48
w/ tables & figures	46.58	41.17	53.98	5.36
InternVL2-26B	41.91	41.02	43.28	**1.48**
w/ tables	40.55	40.37	42.91	1.46
w/ figures	**42.88**	**42.10**	**43.76**	1.46
w/ tables & figures	42.44	42.00	43.31	1.44

LMMs struggles with reasoning over information-intensive images, especially table images.

Performances & Results:



Review Critique

- Compared with human ACs, the LLMs often struggle with identifying the correct reliability of the review viewpoint.

References:

[1]. Chan, Jun Shern, et al. "Mle-bench: Evaluating machine learning agents on machine learning engineering." arXiv preprint arXiv:2410.07095 (2024).

[2]. Chen, Ziru, et al. "Scienceagentbench: Toward rigorous assessment of language agents for data-driven scientific discovery." arXiv preprint arXiv:2410.05080 (2024).

[3]. *Huang, Qian, et al. "Mlagentbench: Evaluating language agents on machine learning experimentation." arXiv preprint arXiv:2310.03302 (2023).*

[4]. *Si, Chenglei, Diyi Yang, and Tatsunori Hashimoto. "Can llms generate novel research ideas? a large-scale human study with 100+ nlp researchers." arXiv preprint arXiv:2409.04109 (2024).*

[5]. *Liang, Weixin, et al. "Can large language models provide useful feedback on research papers? A large-scale empirical analysis." NEJM AI 1.8 (2024): AIoa2400196.*

Thanks.

Q&A

Paper

Website