SCITAB: A Challenging Benchmark for Compositional Reasoning and Claim Verification on Scientific Tables

Xinyuan Lu*[1,2], Liangming Pan*[3], Qian Liu[4], Preslav Nakov[5], Min-Yen Kan[2]

[1] ISEP Program, NUS Graduate School [2] National University of Singapore

[3] University of California, Santa Barbara [4] Sea AI Lab [5] MBZUAI

EMNLP 2023 (Long Paper)

Presenter: Xinyuan Lu

Motivation

- Scientific claim verification aims to help system users assess the veracity of a scientific claim relative to a corpus of research literature.
- Scientific claims are inherently linked to the experimental data they stem from, which are often represented in tables and figures.

- By automatically verifying whether scientific claims are grounded to the tables and figures from which they are derived, we can effectively safeguard the scientific process by promoting reliability of research findings, and therefore reinforce public trust in research outcomes.

Motivation

- Existing benchmarks on scientific claim verification:
  - SciFact *(Wadden et al., 2020)*
  - COVID-Fact *(Saakyan et al., 2021)*
  - Sci-Fact Open *(Wadden et al., 2022)*
- Limitations:
  - First, the claims are crowd-sourced rather than collected from real scientific papers. This leads to problems such as bias in human annotation, a lack of diversity, and shallow claims that do not reflect the complexity of scientific reasoning.
  - Second, the claims in the existing benchmarks are solely validated against text-based evidence, primarily paper abstracts.

Fact or Fiction: Verifying Scientific Claims
COVID-Fact: Fact Extraction and Verification of Real-World Claims on COVID-19 Pandemic
SciFact-Open: Towards open-domain scientific claim verification

Our SCITAB Dataset: Complex and Realistic

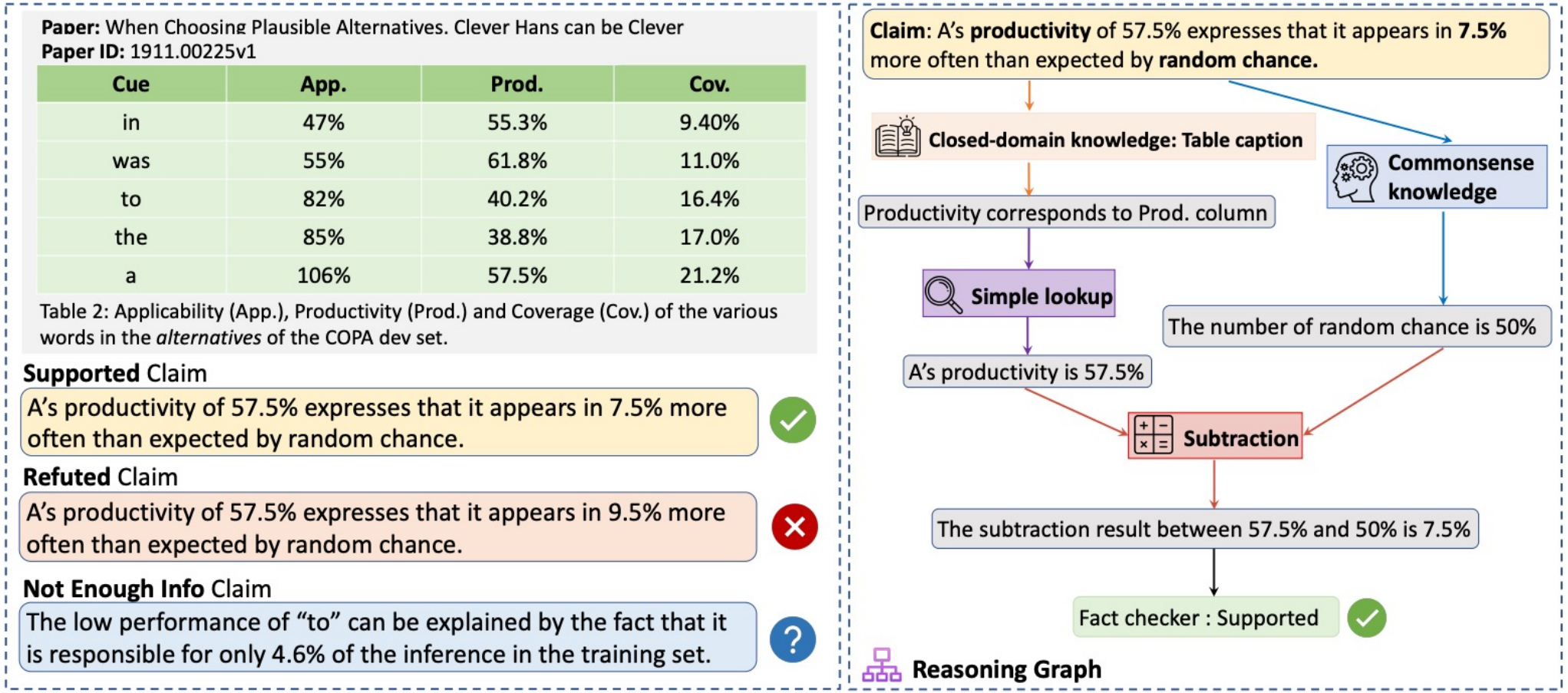


Figure 1: An example of our SCITAB dataset (left) and its corresponding reasoning graph (right). Each data entry contains *paper name*, *paper id*, *table*, one *claim*, and its corresponding *label* (Supported, Refuted, Not Enough Info).

Data Source

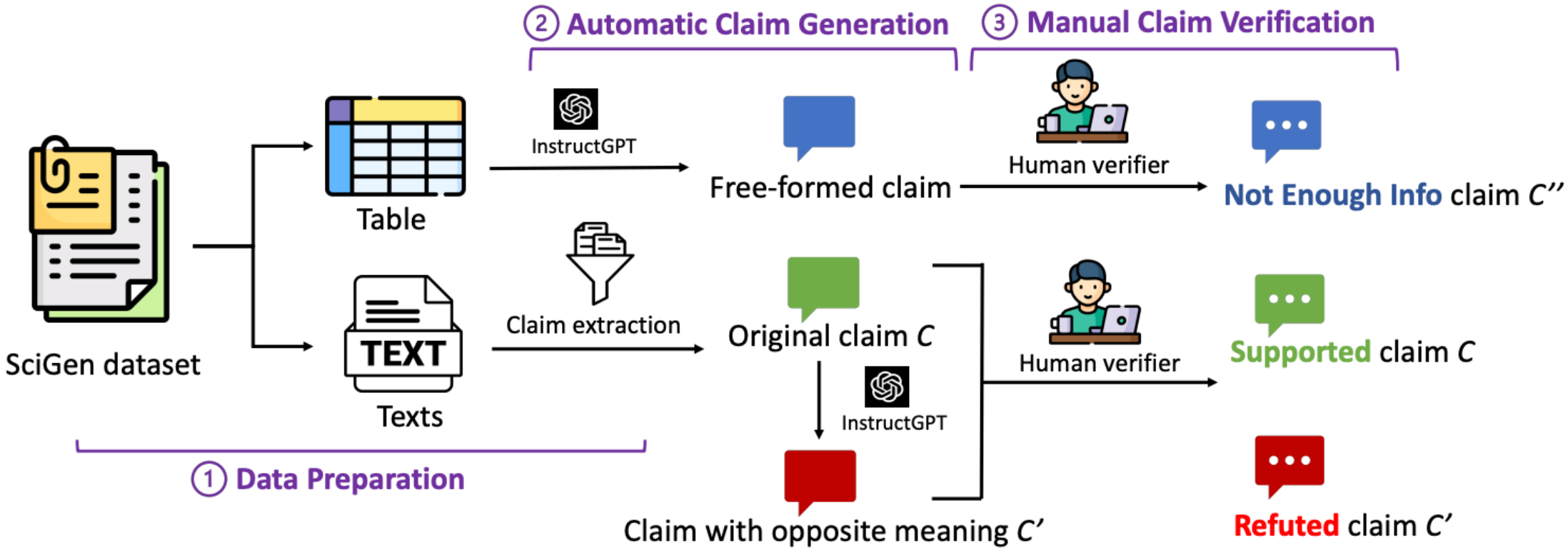
SciGen: a Dataset for Reasoning-Aware Text Generation from Scientific Tables

Nafise Sadat Moosavi[1], **Andreas Rücklé**[1]*, **Dan Roth**[2], **Iryna Gurevych**[1]

[1]Ubiquitous Knowledge Processing Lab (UKP Lab)
Department of Computer Science, Technical University of Darmstadt
https://www.ukp.tu-darmstadt.de
[2]Department of Computer and Information Science, UPenn
https://www.seas.upenn.edu/~danroth

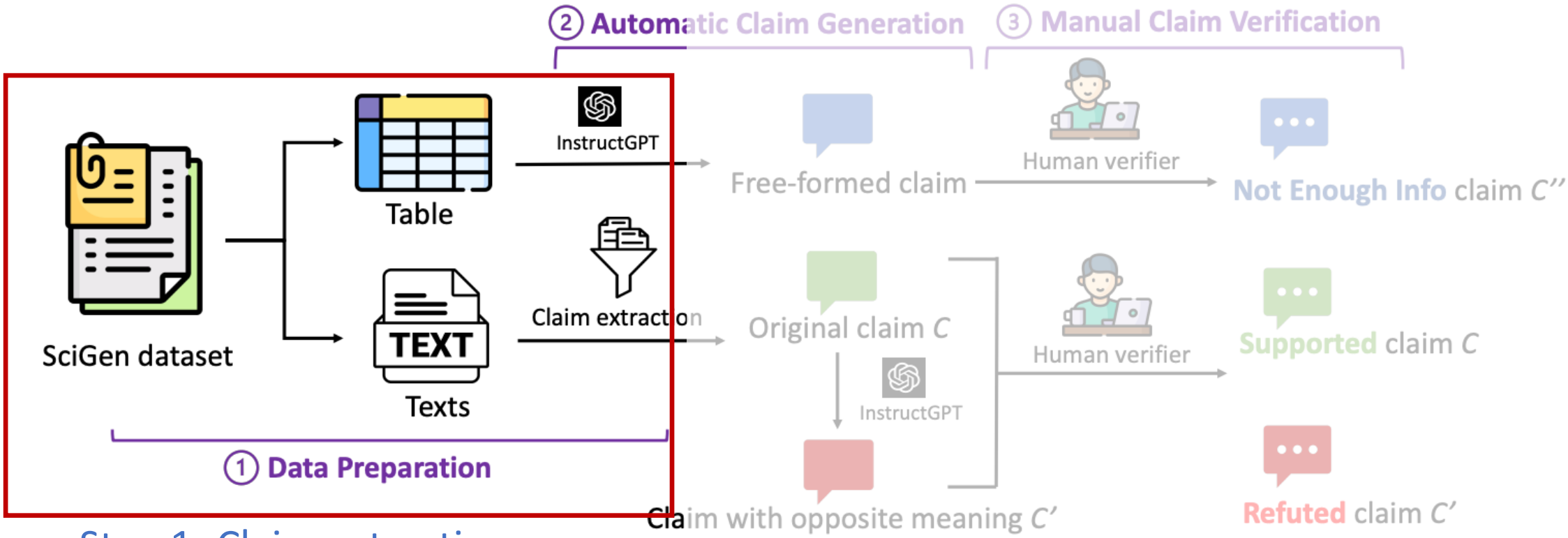
- SciGen: a dataset for arithmetic reasoning in computer science tables.
- They collect real claims from the papers rather than crowd-sourcing them.
- The original task: data-to-text generation for scientific tables.
- Focus on arithmetic reasoning (e.g., argMax, argMin, comparison, subtraction).
- We use SciGen as our data source.

Dataset Construction



- We construct the dataset in the modality of **human--model collaboration**. (Human-in-the-loop)

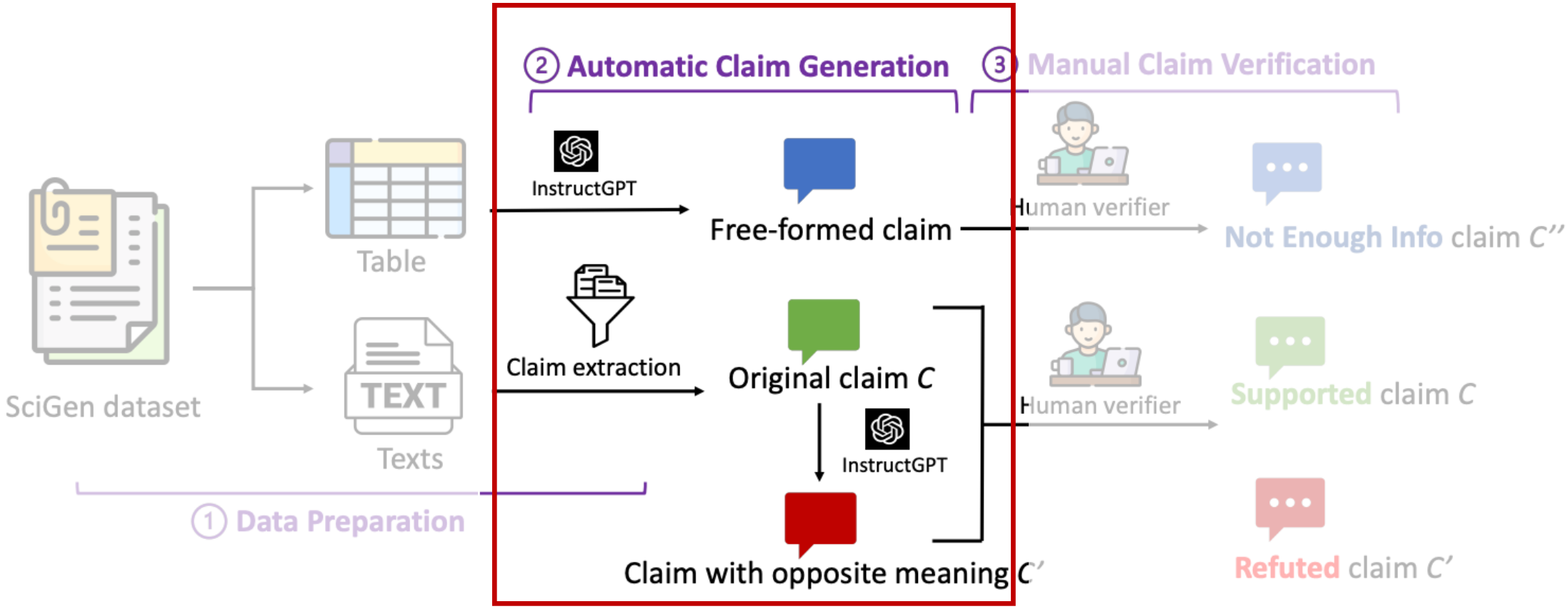
Dataset Construction – STEP 1



Step 1- Claim extraction

- Some sentences in SciGen are describing tables or providing background that are not scientific claims. We only keep claims that reflect scientific findings. Those are **Supported Claims**.

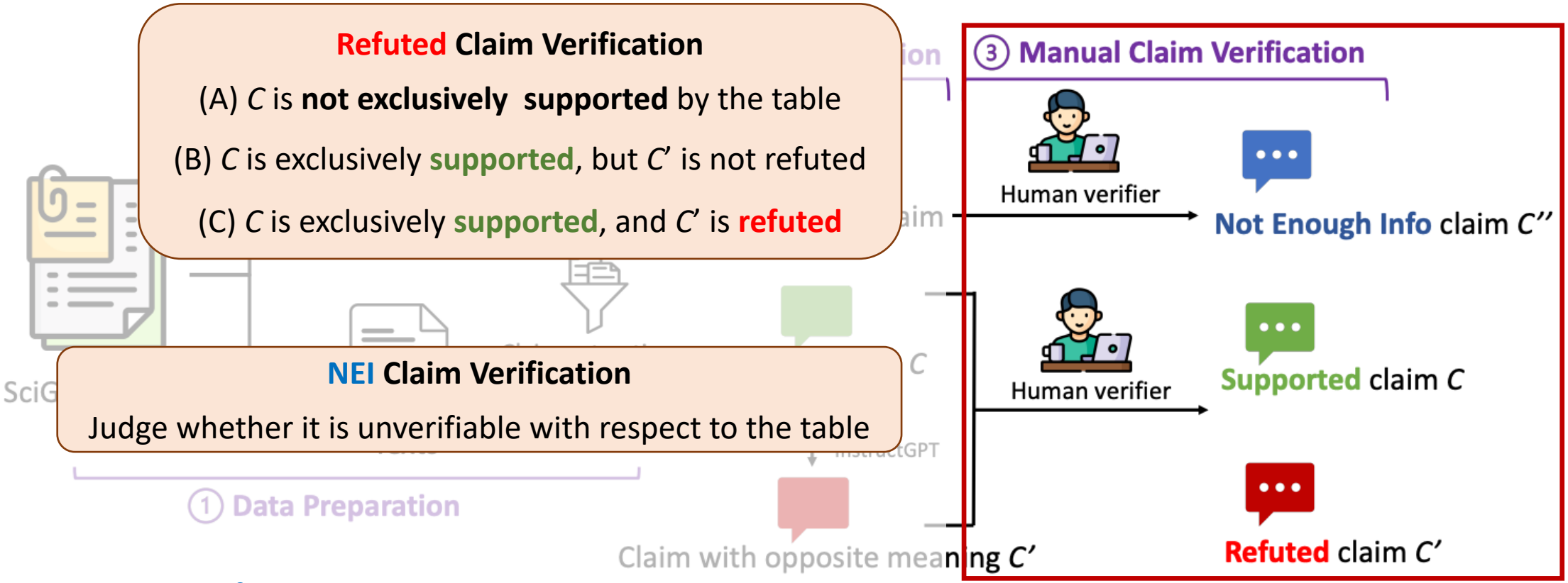
Dataset Construction – STEP 2



Step 2: Generating Refuted and NEI claims automatically

- Refuted claim generation: Reverse the meaning of each supported claim.
- Generate free-formed claims based on in-context examples.

Dataset Construction – STEP 3



Step 3: Human verification

- Refuted claims verification: Humans to filter the generated claims, keeping the actual **Refuted Claims**.
- NEI claims verification: We find that most generated claims are not grounded to the table. Humans to filter those as **NEI Claims**.

Dataset Annotation Process

- Recruiting Annotators
  - 12 university students majoring in computer science
  - Registration forms →Training session → Data annotation → Quality check → Post-annotation survey
  - Each sample is annotated by two annotators.
- Time Duration
  - ~ 4 months.
- Inter-annotator Agreement
  - Refuted claim verification: 0.630 for a total of 872 samples.
  - NEI claim verification: 0.719 for a total of 900 samples.
  - This indicate substantial agreement between annotators.
- Statistics: Support 457; Refuted 412; NEI 357; Total 1,225 claims

Dataset Analysis – Compared to other datasets

Key Differences:

1. Annotator: Experts
2. Maximum Reasoning Hops
3. Balanced 3-class Distributions

Statistics		TabFact	FEVEROUS	SEM-TAB-FACTS	SCITAB
Domain		Wiki Tables	Wiki Tables	Scientific Articles	Scientific Articles
Annotator		AMT	AMT	AMT	Experts
Max. Reasoning Hops		7	2	1	11
Veracity	Supported	54%	56%	58%	37%
	Refuted	46%	39%	38%	34%
	NEI	—	5%	4%	29%
Total # of Claims		117,854	87,026	5,715	1,225
Avg. claims per table		7.11	0.07	5.27	6.16

Table 1: Comparison of SCITAB to three recent table fact verification datasets: TabFact (Chen et al., 2020), FEVEROUS (Aly et al., 2021), and SEM-TAB-FACTS (Wang et al., 2021). The table presents statistics related to the domain, annotator (AMT represents Amazon Mechanical Turk), maximum reasoning hops, veracity labels percentage of each dataset, the total number of claims, and average claims per table.

Dataset Analysis – Reasoning Structure

1. **Reasoning Types**
   - We adopt 14 reasoning types proposed by INFOTABS (Gupta et al., 2020).
   - Define two unique Types: Closed-domain and Open-domain Knowledge
   - SCITAB contains **diverse** reasoning types.

Function Names	Descriptions	Prop. (%)
Simple lookup	Retrieve the value for a specific cell.	20.6
Comparison	Compare two numbers.	19.5
Closed-domain knowledge	Extract information from context sentences in the table caption or article.	12.1
Open-domain knowledge	Extract additional information required by domain experts.	5.3
Commonsense knowledge	Extract commonsense knowledge necessary for claim verification.	5.3
Subtract	Perform subtraction of two numbers.	5.3
Divide	Perform division of two numbers.	5.3
Rank	Determine the rank of a set of numbers.	5.3
Different / Same	Determine if two numbers are different or the same.	5.3
Add	Calculate the sum of two numbers.	4.0
Max / Min	Retrieve the maximum or minimum number from a set of numbers.	3.1
Col / Rowname	Retrieve the column or row name from the table.	3.1
Trend same/different	Determine the trend for two columns or rows, whether they are the same or different.	2.9
Set check	Verify if a value belongs to a set of numbers.	2.9

Table 2: The function names, descriptions, and their proportions in our SCITAB dataset.

INFOTABS: Inference on Tables as Semi-structured Data

Dataset Analysis – Reasoning Structure

2. Reasoning Depths

- The reasoning depths are measured by the number of required reasonings in each claim.
- **Shallow** claims (1-2 reasoning steps) vs **Deep** claims (3+ reasoning steps).
- Average: 4.76; Max Depth: 11
- 86% are deep claims.

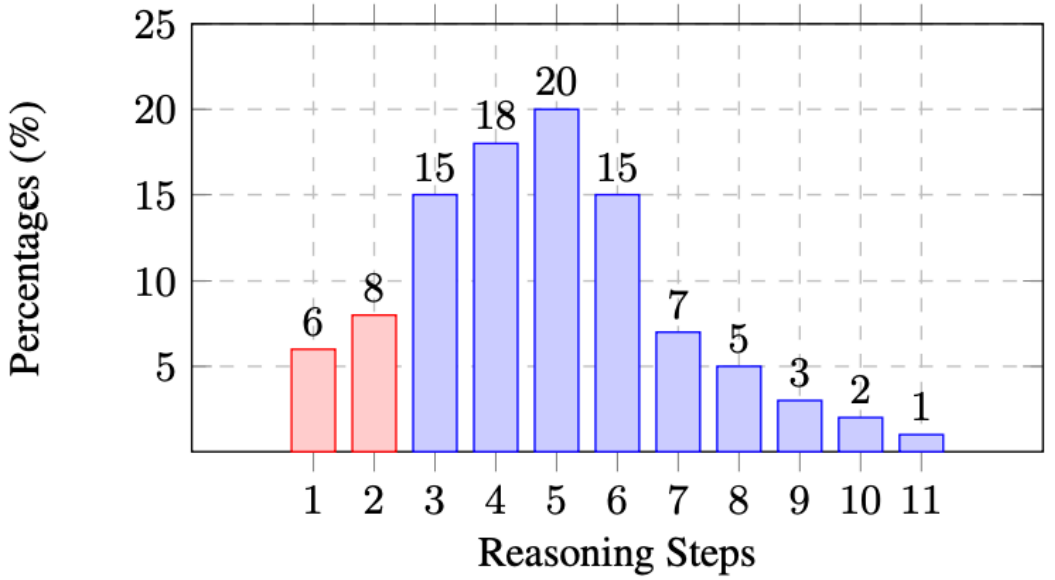


Figure 3: The distribution histogram of reasoning steps in our SCITAB dataset. The x-axis is the reasoning steps in each claim, and the y-axis is the frequency for each reasoning step. The shallow claims (with 1–2 reasoning steps) are highlighted in red, while the deep claims (with 3+ reasoning steps) are highlighted in blue.

Dataset Analysis – Reasoning Structure

3. Reasoning Graphs

The reasoning depths and reasoning graphs show the **complexity** of SCITAB.

Paper: When Choosing Plausible Alternatives. Clever Hans can be Clever
Paper ID: 1911.00225v1

Cue	App.	Prod.	Cov.
in	47%	55.3%	9.40%
was	55%	61.8%	11.0%
to	82%	40.2%	16.4%
the	85%	38.8%	17.0%
a	106%	57.5%	21.2%

Table 2: Applicability (App.), Productivity (Prod.) and Coverage (Cov.) of the various words in the *alternatives* of the COPA dev set.

Supported Claim

A's productivity of 57.5% expresses that it appears in 7.5% more often than expected by random chance.

Refuted Claim

A's productivity of 57.5% expresses that it appears in 9.5% more often than expected by random chance.

Not Enough Info Claim

The low performance of "to" can be explained by the fact that it is responsible for only 4.6% of the inference in the training set.

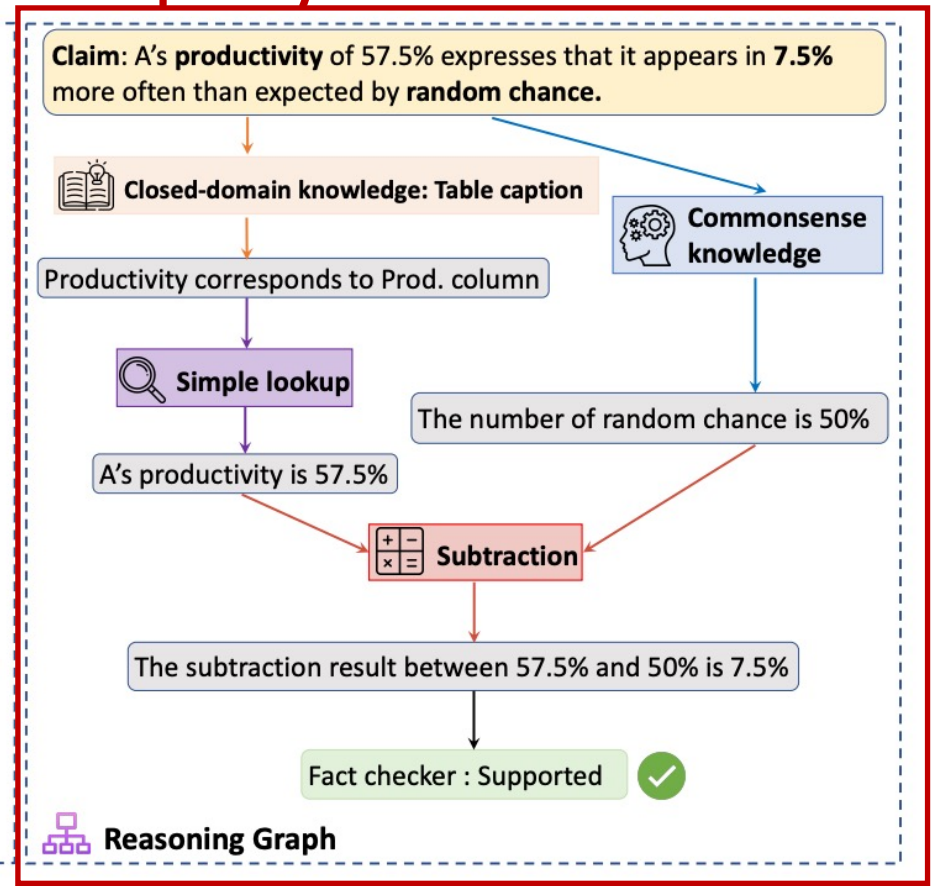


Figure 1: An example of our SCITAB dataset (left) and its corresponding reasoning graph (right). Each data entry contains *paper name*, *paper id*, *table*, one *claim*, and its corresponding *label* (Supported, Refuted, Not Enough Info).

Dataset Analysis – Refuted and NEI claims

- Unlike SciFact, SCITAB exhibits a greater diversity in refuted claims.
- Unique types of errors that are more reflective of the complexities in real-world scientific claims:
  - incorrect approximation words
  - the claim is partially right.

- The NEI claims also exhibit diverse reasoning patterns.
- The two most common features:
  - insufficient evidence in the table
  - the lack of background knowledge.

- These distinct refuted and NEI reasoning types highlight the unique features of SCITAB, making it a more comprehensive and realistic representation of the challenges faced in real-world scientific fact-checking.

Refuted Reasons	**Prop. (%)**
The calculation result is wrong.	41.7
The approximation word is wrong.	33.3
The claim is partially right.	10.0
The values in the claim do not match.	8.3
The operation type is wrong.	6.7
NEI Reasons	**Prop. (%)**
The claim does not have enough matching evidence.	33.3
The claim lacks open-domain knowledge.	25.0
The claim lacks closed-domain knowledge.	15.0
The claim refers to another table.	11.7
The claim contains vague pronouns.	8.3
The claim omits specific information.	6.7

Table 3: Refuted and NEI reasons and their estimated proportions (Prop.) in SCITAB.

Benchmark Evaluation

- **Task Definition**

  Given a claim *C* and (table + caption) *T* , a table fact-checking model *F* predicts a label *Y* to verify whether *C* is **supported, refuted, or does not have enough information (NEI) to be verified** by the information in *T* .

- **Models**
  - Table-based LLMs
  - Encoder–Decoder LLMs, e.g., Flan-T5
  - Open-sourced LLMs
  - Close-sourced LLMs (black-box, not replicable)
  - Human performance (upper bound)

Benchmark Evaluation Results

	Models	# of Para.	Zero-shot		In-Context	
			2-class	3-class	2-class	3-class
I. Table-based LLMs	TAPAS-large (Tabfact) (Herzig et al., 2020)	340M	50.30	—	—	—
	TAPEX-large (Tabfact) (Liu et al., 2022b)	400M	56.06	—	—	—
	TAPEX-Zero-large (Liu et al., 2023b)	780M	48.28	29.72	42.44	23.47
	TAPEX-Zero-XL (Liu et al., 2023b)	3B	49.77	34.30	42.12	25.62
II. Encoder–Decoder LLMs	Flan-T5-base (Chung et al., 2022)	250M	47.38	26.56	44.82	24.09
	Flan-T5-large (Chung et al., 2022)	780M	51.58	32.55	49.62	27.30
	FLan-T5-XL (Chung et al., 2022)	3B	52.41	**38.05**	48.05	29.21
	Flan-T5-XXL (Chung et al., 2022)	11B	59.60	34.91	**60.48**	34.04
III. Open source LLMs	Alpaca-7B (Taori et al., 2023)	7B	37.22	27.59	40.46	28.95
	Vicuna-7B (Chiang et al., 2023)	7B	**63.62**	32.47	50.35	34.26
	Vicuna-13B (Chiang et al., 2023)	13B	41.82	29.63	55.11	**35.16**
	LLaMA-7B (Touvron et al., 2023)	7B	49.05	32.26	45.24	27.17
	LLaMA-13B (Touvron et al., 2023)	13B	53.97	37.18	44.39	32.66
IV. Close source LLMs	InstructGPT (Ouyang et al., 2022)	175B	68.44	41.41	68.10	41.58
	InstructGPT+CoT (Ouyang et al., 2022)	175B	—	—	68.46	42.60
	PoT (Chen et al., 2022)	175B	—	—	63.79	—
	GPT-4 (OpenAI, 2023)	—	<u>78.22</u>	<u>64.80</u>	<u>77.98</u>	<u>63.21</u>
	GPT-4+CoT (OpenAI, 2023)	—	—	—	76.85	62.77
	Human	—	—	—	92.40	84.73

Underlined text: best performance from I to IV; Bold text: best performance from I to III

Benchmark Evaluation Results

	Models	# of Para.	Zero-shot		In-Context	
			2-class	3-class	2-class	3-class
I. Table-based LLMs	TAPAS-large (Tabfact) (Herzig et al., 2020)	340M	50.30	—	—	—
	TAPEX-large (Tabfact) (Liu et al., 2022b)	400M	56.06	—	—	—
	TAPEX-Zero-large (Liu et al., 2023b)	780M	48.28	29.72	42.44	23.47
	TAPEX-Zero-XL (Liu et al., 2023b)	3B	49.77	34.30	42.12	25.62
II. Encoder–Decoder LLMs	Flan-T5-base (Chung et al., 2022)	250M	47.38	26.56	44.82	24.09
	Flan-T5-large (Chung et al., 2022)	780M	51.58	32.55	49.62	27.30
	FLan-T5-XL (Chung et al., 2022)	3B	52.41	**38.05**	48.05	29.21
	Flan-T5-XXL (Chung et al., 2022)	11B	59.60	34.91	**60.48**	34.04
III. Open source LLMs	Alpaca-7B (Taori et al., 2023)	7B	37.22	27.59	40.46	28.95
	Vicuna-7B (Chiang et al., 2023)	7B	**63.62**	32.47	50.35	34.26
	Vicuna-13B (Chiang et al., 2023)	13B	41.82	29.63	55.11	**35.16**
	LLaMA-7B (Touvron et al., 2023)	7B	49.05	32.26	45.24	27.17
	LLaMA-13B (Touvron et al., 2023)	13B	53.97	37.18	44.39	32.66
IV. Closed source LLMs	GPT-4+CoT (OpenAI, 2023)	—	—	—	76.85	62.77
	Human	—	—	—	92.40	84.73

1. Generally, all open source LLMs, including encoder–decoder models, do not achieve very promising results on SCITAB and they still have a large gap from human performance.

Underlined text: best performance from I to IV; Bold text: best performance from I to III

Benchmark Evaluation Results

	Models	# of Para.	Zero-shot		In-Context	
			2-class	3-class	2-class	3-class
I. Table-based LLMs	TAPAS-large (Tabfact) (Herzig et al., 2020)	340M	50.30	—	—	—
	TAPEX-large (Tabfact) (Liu et al., 2022b)	400M	56.06	—	—	—
	TAPEX-Zero-large (Liu et al., 2023b)	780M	48.28	29.72	42.44	23.47
	TAPEX-Zero-XL (Liu et al., 2023b)	3B	49.77	34.30	42.12	25.62
II. Encoder–Decoder LLMs	Flan-T5-base (Chung et al., 2022)	250M	47.38	26.56	44.82	24.09
	Flan-T5-large (Chung et al., 2022)	780M	51.58	32.55	49.62	27.30
	FLan-T5-XL (Chung et al., 2022)	3B	52.41	**38.05**	48.05	29.21
	Flan-T5-XXL (Chung et al., 2022)	11B	59.60	34.91	**60.48**	34.04
	Alpaca-7B (Taori et al., 2023)	7B	37.22	27.59	40.46	28.95
IV. Close source LLMs	InstructGPT (Ouyang et al., 2022)	175B	68.44	41.41	68.10	41.58
	InstructGPT+CoT (Ouyang et al., 2022)	175B	—	—	68.46	42.60
	PoT (Chen et al., 2022)	175B	—	—	63.79	—
	GPT-4 (OpenAI, 2023)	—	78.22	64.80	77.98	63.21
	GPT-4+CoT (OpenAI, 2023)	—	—	—	76.85	62.77
	Human	—	—	—	92.40	84.73

2. Counter-intuitively, table-based LLMs do not outperform models pre-trained on pure texts, e.g., Flan-T5.

Underlined text: best performance from I to IV; Bold text: best performance from I to III

Benchmark Evaluation Results

	Models	# of Para.	Zero-shot		In-Context	
			2-class	3-class	2-class	3-class
I. Table-based LLMs	TAPAS-large (Tabfact) (Herzig et al., 2020)	340M	50.30	—	—	—
	TAPEX-large (Tabfact) (Liu et al., 2022b)	400M	56.06	—	—	—
	TAPEX-Zero-large (Liu et al., 2023b)	780M	48.28	29.72	42.44	23.47
	TAPEX-Zero-XL (Liu et al., 2023b)	3B	49.77	34.30	42.12	25.62
II. Encoder–Decoder LLMs	Flan-T5-base (Chung et al., 2022)	250M	47.38	26.56	44.82	24.09
	Flan-T5-large (Chung et al., 2022)	780M	51.58	32.55	49.62	27.30
	FLan-T5-XL (Chung et al., 2022)	3B	52.41	**38.05**	48.05	29.21
	Flan-T5-XXL (Chung et al., 2022)	11B	59.60	34.91	**60.48**	34.04
III. Decoder only LLMs				.59	40.46	28.95
				.47	50.35	34.26
				.63	55.11	**35.16**
				.26	45.24	27.17
	LLaMA-13B (Touvron et al., 2023)	13B	53.97	37.18	44.39	32.66
IV. Close source LLMs	InstructGPT (Ouyang et al., 2022)	175B	68.44	41.41	68.10	41.58
	InstructGPT+CoT (Ouyang et al., 2022)	175B	—	—	68.46	42.60
	PoT (Chen et al., 2022)	175B	—	—	63.79	—
	GPT-4 (OpenAI, 2023)	—	78.22	64.80	77.98	63.21
	GPT-4+CoT (OpenAI, 2023)	—	—	—	76.85	62.77
	Human	—	—	—	92.40	84.73

3. The results in the 3-class setting are notably poorer than those in the 2-class setting.

Underlined text: best performance from I to IV; Bold text: best performance from I to III

Benchmark Evaluation Results

	Models	# of Para.	Zero-shot		In-Context	
			2-class	3-class	2-class	3-class
I. Table-based LLMs	TAPAS-large (Tabfact) (Herzig et al., 2020)	340M	50.30	—	—	—
	TAPEX-large (Tabfact) (Liu et al., 2022b)	400M	56.06	—	—	—
	TAPEX-Zero-large (Liu et al., 2023b)	780M	48.28	29.72	42.44	23.47
	TAPEX-Zero-XL (Liu et al., 2023b)	3B	49.77	34.30	42.12	25.62
II. Encoder–Decoder LLMs	Flan-T5-base (Chung et al., 2022)	250M	47.38	26.56	44.82	24.09
	Flan-T5-large (Chung et al., 2022)	780M	51.58	32.55	49.62	27.30
	FLan-T5-XL (Chung et al., 2022)	3B	52.41	**38.05**	48.05	29.21
	Flan-T5-XXL (Chung et al., 2022)	11B	59.60	34.91	**60.48**	34.04
III. Decoder-only LLMs			37.22	27.59	40.46	28.95
			63.62	32.47	50.35	34.26
			41.82	29.63	55.11	**35.16**
			49.05	32.26	45.24	27.17
			53.97	37.18	44.39	32.66
IV. Close source LLMs	InstructGPT (Ouyang et al., 2022)	175B	68.44	41.41	68.10	41.58
	InstructGPT+CoT (Ouyang et al., 2022)	175B	—	—	68.46	42.60
	PoT (Chen et al., 2022)	175B	—	—	63.79	—
	GPT-4 (OpenAI, 2023)	—	<u>78.22</u>	<u>64.80</u>	<u>77.98</u>	<u>63.21</u>
	GPT-4+CoT (OpenAI, 2023)	—	—	—	76.85	62.77
	Human	—	—	—	92.40	84.73

4. Interestingly, the provision of in-context examples does not result in improved performance for the majority of models.

Underlined text: best performance from I to IV; Bold text: best performance from I to III

Benchmark Evaluation Results

	Models	# of Para.	Zero-shot		In-Context	
			2-class	3-class	2-class	3-class
I. Table-based LLMs	TAPAS-large (Tabfact) (Herzig et al., 2020)	340M	50.30	—	—	—
	TAPEX-large (Tabfact) (Liu et al., 2022b)	400M	56.06	—	—	—
	TAPEX-Zero-large (Liu et al., 2023b)	780M	48.28	29.72	42.44	23.47
	TAPEX-Zero-XL (Liu et al., 2023b)	3B	49.77	34.30	42.12	25.62
	Flan-T5-base (Chung et al., 2022)	250M	47.38	26.56	44.82	24.09
	Flan-T5-XXL (Chung et al., 2022)	11B	59.60	34.91	**60.48**	34.04
III. Open source LLMs	Alpaca-7B (Taori et al., 2023)	7B	37.22	27.59	40.46	28.95
	Vicuna-7B (Chiang et al., 2023)	7B	**63.62**	32.47	50.35	34.26
	Vicuna-13B (Chiang et al., 2023)	13B	41.82	29.63	55.11	**35.16**
	LLaMA-7B (Touvron et al., 2023)	7B	49.05	32.26	45.24	27.17
	LLaMA-13B (Touvron et al., 2023)	13B	53.97	37.18	44.39	32.66
IV. Close source LLMs	InstructGPT (Ouyang et al., 2022)	175B	68.44	41.41	68.10	41.58
	InstructGPT+CoT (Ouyang et al., 2022)	175B	—	—	68.46	42.60
	PoT (Chen et al., 2022)	175B	—	—	63.79	—
	GPT-4 (OpenAI, 2023)	—	<u>78.22</u>	<u>64.80</u>	<u>77.98</u>	<u>63.21</u>
	GPT-4+CoT (OpenAI, 2023)	—	—	—	76.85	62.77
	Human	—	—	—	92.40	84.73

5. Closed source LLMs perform better than open source LLMs.

Underlined text: best performance from I to IV; Bold text: best performance from I to III

Error Analysis: InstructGPT and GPT-4

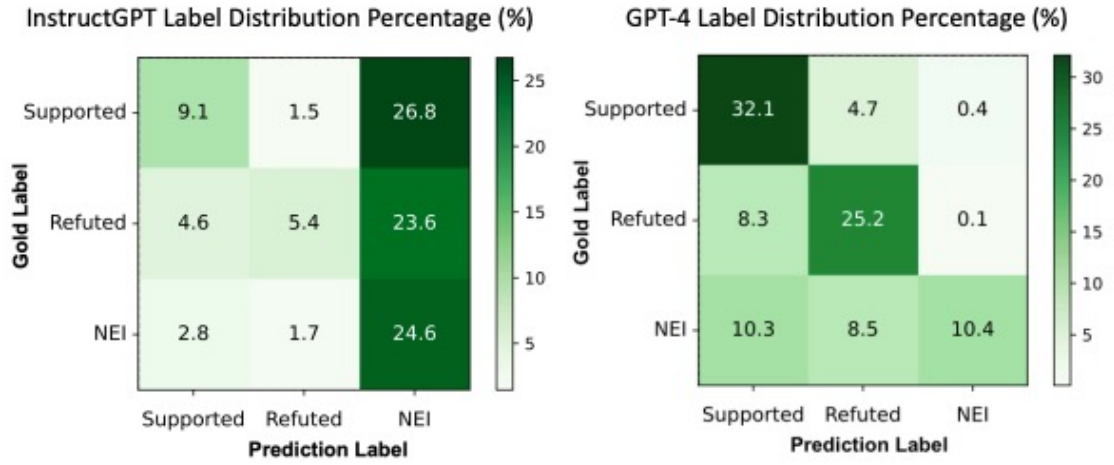


Figure 4: Confusion matrix for InstructGPT (left) and GPT-4 (right) on zero-shot 3-class classification.

- We find that both models displayed difficulty in accurately predicting the NEI class.
- InstructGPT displays a pattern of “less confident”, frequently classifying supported and refuted claims as ‘NEI’.
- In contrast, GPT-4 exhibits a pattern of “overconfidence”, incorrectly categorizing NEI claims as either supported or refuted.

Error Analysis: Program-of-Thoughts

Error Type	Estimated Proportion (%)
I. Grounding errors	50
II. Ambiguity errors	22
III. Calculation errors	20
IV. Program errors	8

Table 5: The error types and their estimated proportions for incorrectly-predicted samples in PoT.

(i) Grounding errors, where the program incorrectly associates data with the respective cells in the table;

(ii) Ambiguity errors, where the claim contains ambiguous expressions that the program fails to represent;

(iii) Calculation errors, where incorrect float digits in the Python code lead to inaccurate calculation results.

(iv) Program errors, which encompass mistakes such as incorrect or missing arguments/- variables, and erroneous operations;

Summary

- We construct a diagnostic dataset SCITAB, which contains complex scientific claims that requires multiple types of reasoning to verify.
- Unlike the other datasets, we use claims from original scientific papers as supported claims to ensure **realistic** and **complex** claims that require in-depth reasoning.
- We conduct an in-depth analysis to show the **complexity** and **diversity** of the claims.
- We evaluate the SOTA models on our dataset, find it presents a challenging task for current models.

Paper

Thanks!

Any questions?

Xinyuan Lu

Email: luxinyuan@u.nus.edu

Twitter
@ShellyLu00

Personal
Homepage

Data