August 3–7, 2025
KDD2025

SoAy: A Solution-based LLM API-using Methodology for Academic Information Seeking

Yuanchun Wang, Jifan Yu, Zijun Yao, Jing Zhang, Yuyang Xie, Shangqing Tu, Yiyang Fu, Youhe Feng, Jinkai Zhang, Jingyao Zhang, Bowen Huang, Yuanyao Li, Huihui Yuan, Lei Hou, Juanzi Li, Jie Tang

Seeking Academic Metadata

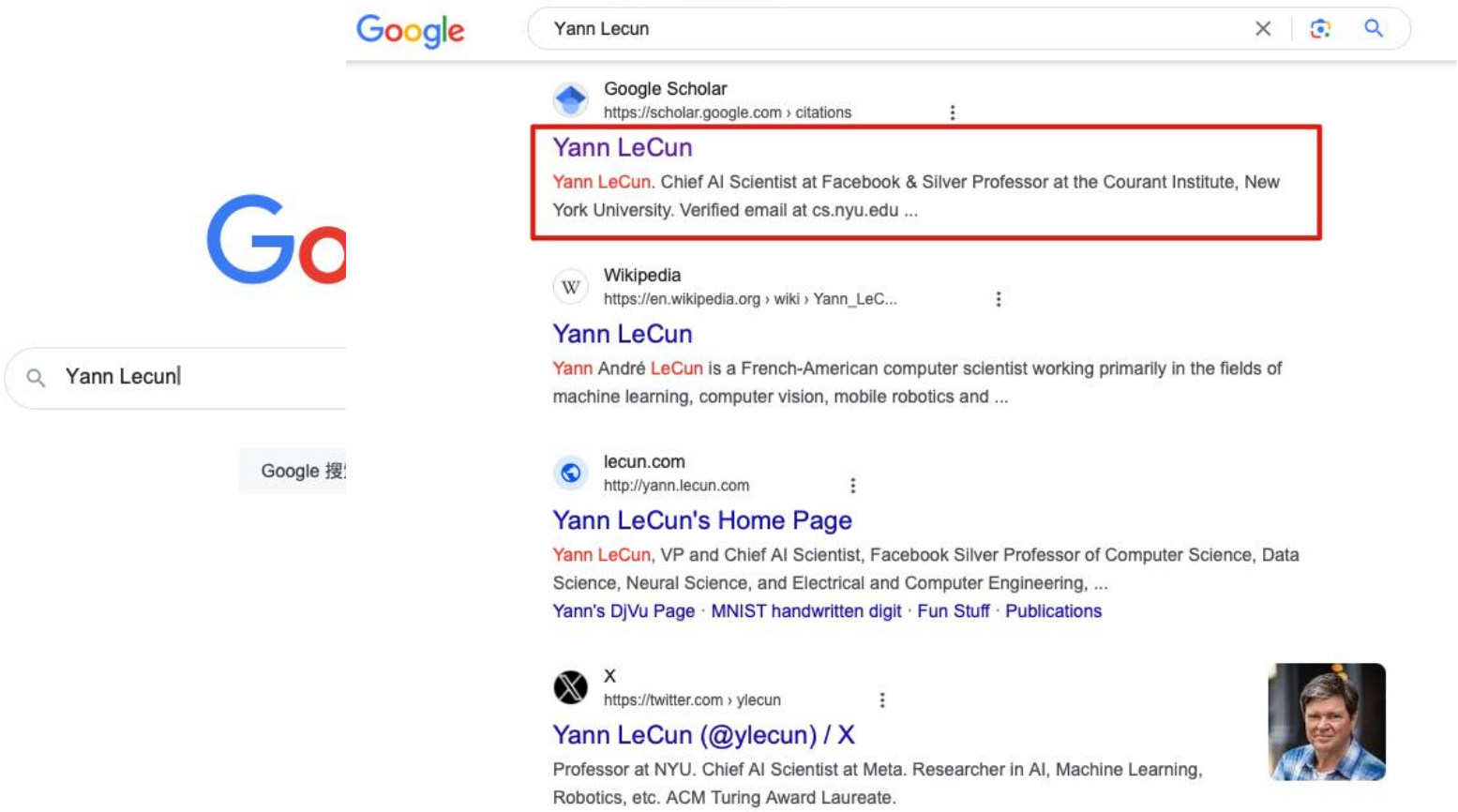
Query: How many times has New York University's Yann LeCun's most cited publication been cited?

Step 1: Typing Keywords in the searching box

Seeking Academic Metadata

Query: How many times has New York University's Yann LeCun's most cited publication been cited?

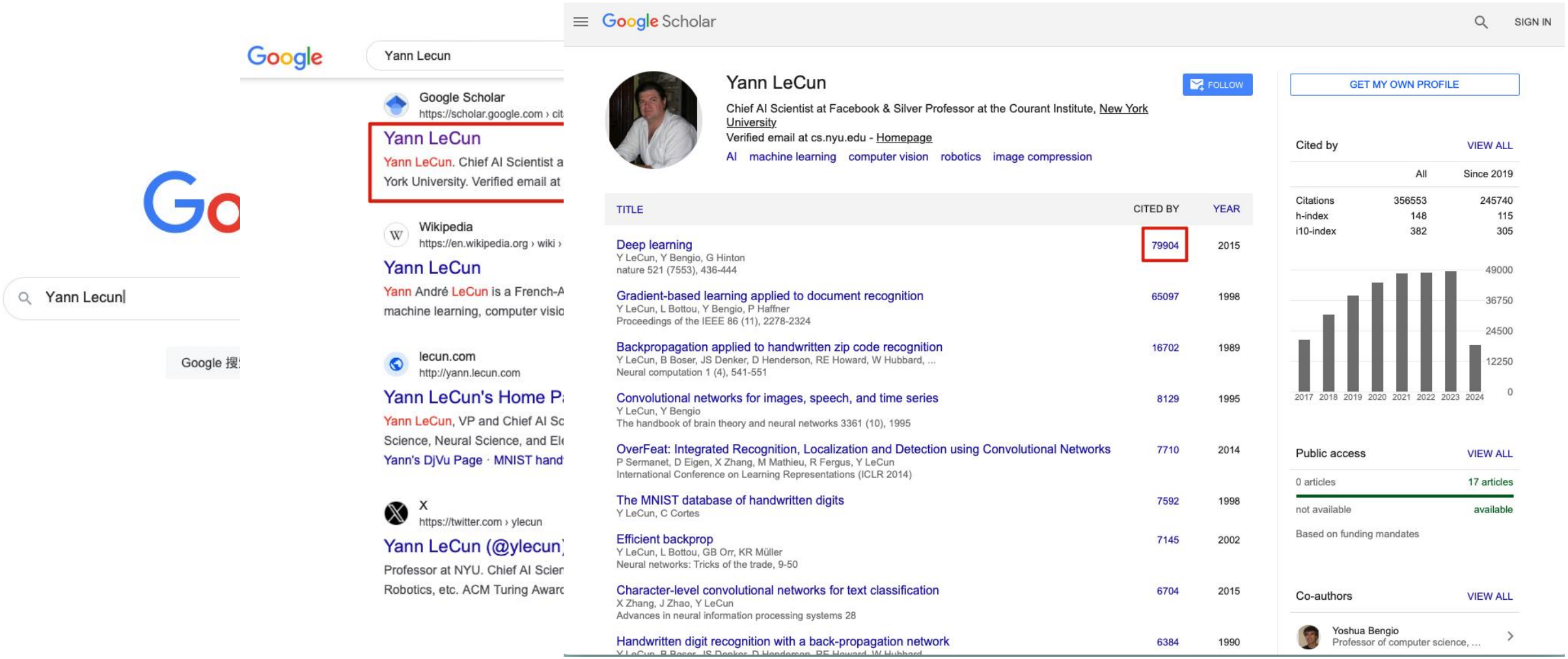
Step 2: Find the probable item in the results list.



Seeking Academic Metadata

Query: How many times has New York University's Yann LeCun's most cited publication been cited?

Step 3: Seek the target information on the page



Academic Information Systems API Calling

Query: How many times has New York University's Yann LeCun's most cited publication been cited?

ID	API name	Type	Parameter(s)	Return
1	searchPerson	fuzzy	name, organization, interest	[person_id, name, num_citation, interest, num_pubs, organization]
2	searchPublication	fuzzy	publication_info	[pub_id, title, year]
3	getCoauthors	exact	person_id	[id, name, relation]
4	getPersonInterest	exact	person_id	list of interests
5	getPublication	exact	pub_id	abstract, author_list, num_citation
6	getPersonBasicInfo	exact	pub_id	person_id, name, gender, organization, position, bio, education_experience, email
7	getPersonPubs	exact	person_id	[authors_name_list, pub_id, title, num_citation, year]

Yann LeCun, NYU ➤ searchPerson ➤ [{ person_id: ec0f***jsk, person_name: Yann LeCun, ... }]

ec0f***jsk ➤ getPersonPubs ➤ [{ pub_id : al4k***8fa, ...}, { pub_id : 79pa***rjk, ...}, { pub_id : q2f4***n3c, ...}...]

al4k***8fa
79pa***rjk
q2f4***n3c
➤ getPublication ➤ [{ title: Efficient Backprop, citation: 7145}, { title: Deeplearning, citation: 79904}, { title: The Minist Database, citation: 7592}...]

Academic Information Systems API Calling

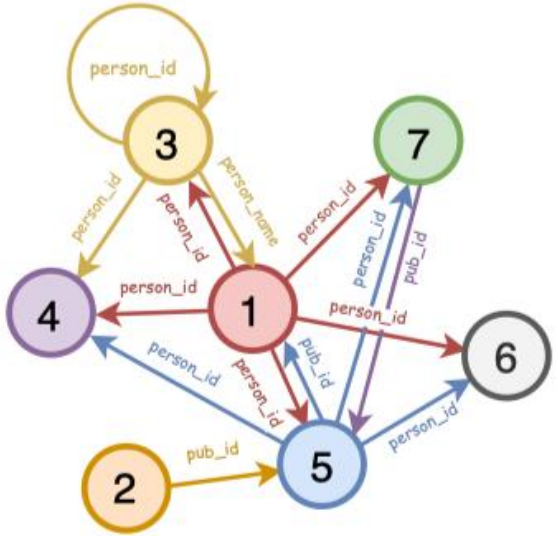
Query: How many times has New York University's Yann LeCun's most cited publication been cited?

Academic API Features:

Coupling: Specific API calling order
Efficiency: High efficiency requirement

Yann LeCun, NYU ➤ searchPerson ➤ getPersonPubs ➤ getPublication ➤ 79904

ID	API name	Type	Parameter(s)	Return
1	searchPerson	fuzzy	name, organization, interest	[person_id, name, num_citation, interest, num_pubs, organization]
2	searchPublication	fuzzy	publication_info	[pub_id, title, year]
3	getCoauthors	exact	person_id	[id, name, relation]
4	getPersonInterest	exact	person_id	list of interests
5	getPublication	exact	pub_id	abstract, author_list, num_citation
6	getPersonBasicInfo	exact	pub_id	person_id, name, gender, organization, position, bio, education_experience, email
7	getPersonPubs	exact	person_id	[authors_name_list, pub_id, title, num_citation, year]

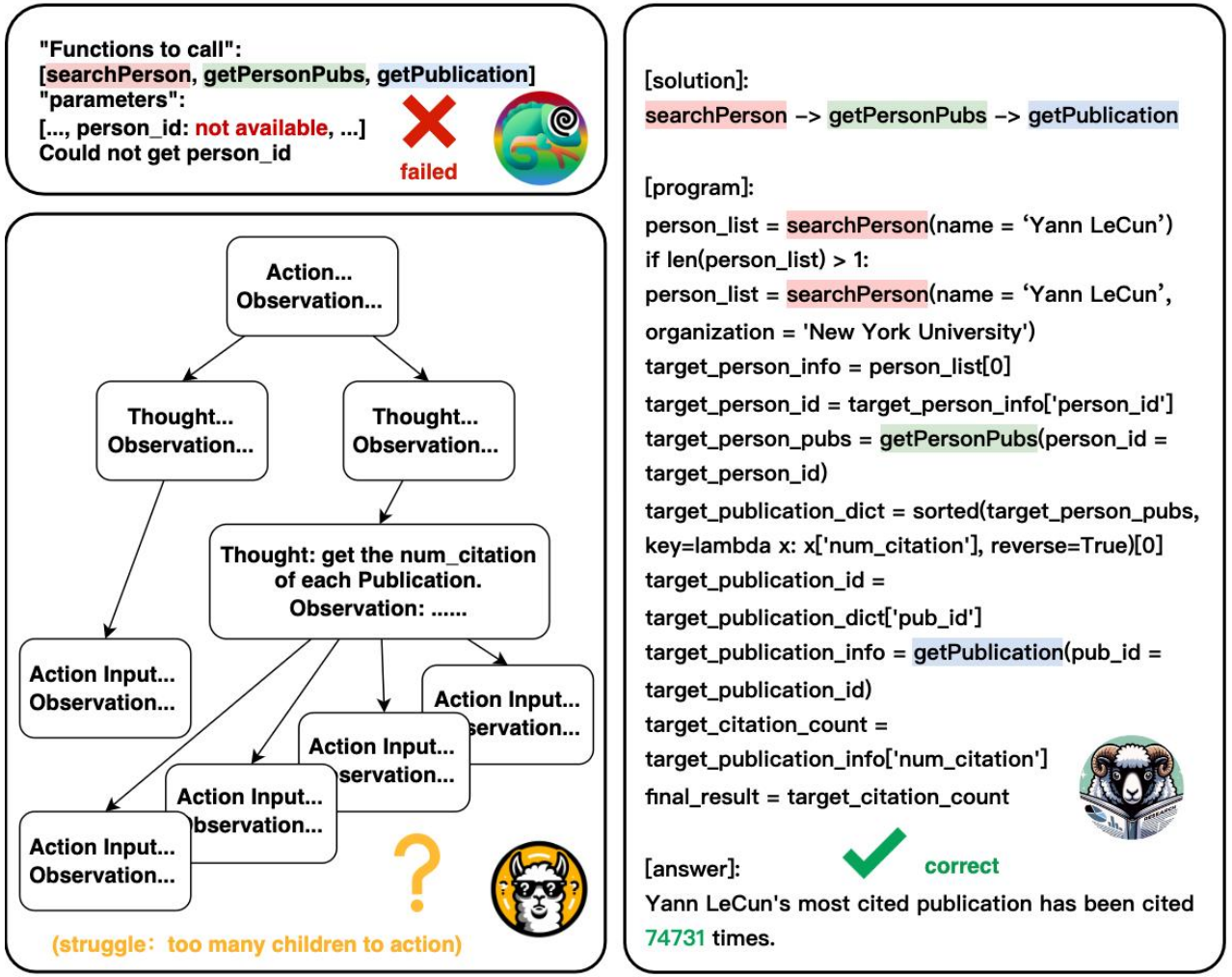


LLM API-Using

Query: How many times has New York University's Yann LeCun's most cited publication been cited?

Retrieval & Execution:
Failed to handle
API Coupling

DFSDT Reasoning:
Could not meet
the Efficiency needs



SoAy:

Pre-defined Solution
&
Solution-based Program
Generation

Fig.1 Different API-using sturctures facing the same academic question.

SoAy: SoAPIs Applying Framework

Query: How many times has New York University's Yann LeCun's most cited publication been cited?

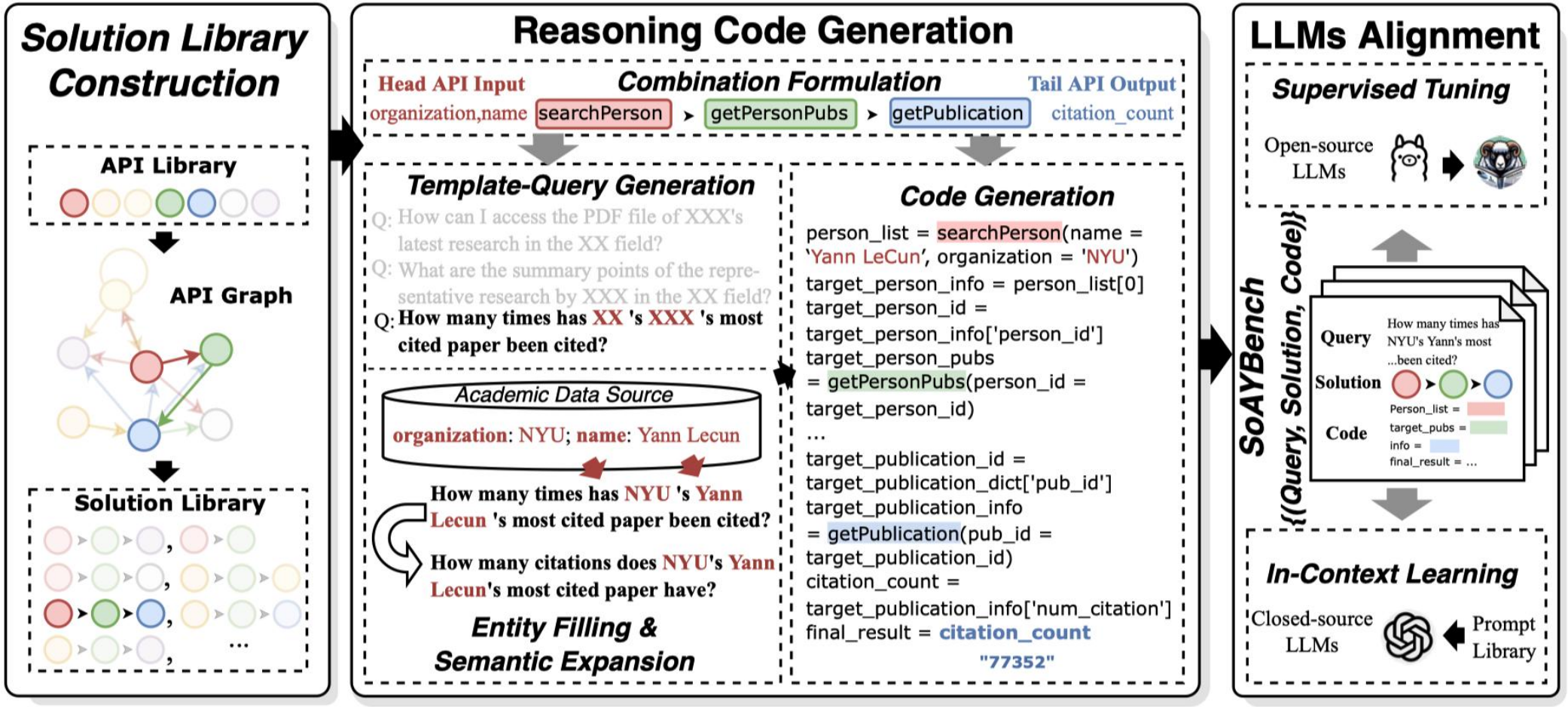


Fig.2 SoAy Framework.

SoAyBench

To assess API utilization capabilities, it is essential to publish the foundational APIs of AMiner for LLMs to invoke and provide a test set composed of academic {Query, Code, Answer} triplets for evaluation.

However, given the dynamic nature of academic data, with scholar and publication information rapidly changing, maintaining a test set with static answers proves challenging.

To address this challenge, **we clone AMiner's SoAPIs at a specific point in time** to create a static service, from which we generate a corresponding static test set.

SoAyBench now are open-sourced at : 🤗 Hugging Face:
https://huggingface.co/datasets/frederickwang99/SoAyBench

Question statistics in SoAyBench.

Question Type	One-hop	Two-hop	Three-hop	**Total**
Scholar	540	1,800	540	2,980
Publication	180	180	720	1,080
Total	720	1,980	1,230	**3,960**

SoAyEval

We outline five types of evaluation metrics.

* EM: Both the retrieved solution and answer Exactly Match the ground truth.
* DS: The answer is correct, but a Different Solution is retrieved compared to the ground truth.
* WS: The answer is wrong due to a Wrong Solution.
* WP: The solution is correct but the answer is wrong, due to a Wrong Program generated for the solution, which can be executed but yields the wrong answer.
* EE: Execution Error, which may by caused by the generation of a nonexecutable program or network errors during the APIs request.

$$\mathrm{ACC} = \mathrm{EM} + \mathrm{DS}$$

$$\mathrm{Score} = \frac{w_1 \cdot \mathrm{ACC}_1 + w_2 * \cdot \mathrm{ACC}_2 + w_3 \cdot \mathrm{ACC}_3}{w_1 + w_2 + w_3}$$

Results on SoAyBench - Part I

Results on SoAyBench. DS, WS, WP and EE are differenct types of error, ACC denotes a accurate answer, EM means exact match, not only the answer but also the solution. Score is a weighted sum of the ACC score on different question types.

Method	Version	Question Type	Error Rate↓				EM(%)	ACC(%)	Score
			DS(%)	WS(%)	WP(%)	EE(%)			
ToolLLaMA	7B	one-hop	12.50±8.00	24.31±13.26	1.39±0.00	54.17±16.01	7.64±5.20	20.14	
		two-hop	10.10±4.10	47.22±12.28	0.76±2.27	38.13±9.62	3.79±2.92	13.89	16.72
		three-hop	11.51±6.53	38.10±14.27	1.19±3.57	43.25±13.07	5.95±4.59	17.46	
GPT-DFSDT	3.5	one-hop	55.56±21.06	15.28±7.80	4.86±0.00	21.53±10.67	2.78±0.00	58.33	
		two-hop	29.55±11.47	34.34±9.23	4.29±3.64	25.76±8.65	6.06±4.11	35.61	43.22
		three-hop	38.10±15.09	28.57±11.35	3.17±2.50	25.00±8.87	5.16±6.19	43.25	
	3.5-16k	one-hop	25.69±10.91	9.72±5.00	2.78±0.00	22.92±9.47	38.89±15.60	64.58	
		two-hop	16.92±7.76	15.91±6.05	3.28±1.31	46.97±7.13	16.92±4.99	33.84	43.67
		three-hop	18.65±7.37	15.48±5.63	2.78±0.00	38.49±10.43	24.60±8.53	43.25	
	4	one-hop	27.78±9.60	2.08±0.00	4.17±5.00	28.47±6.82	37.50±10.91	65.28	
		two-hop	26.26±8.89	9.60±4.88	17.93±5.40	15.15±5.39	31.06±9.12	57.32	58.16
		three-hop	22.22±8.65	7.54±4.46	17.06±6.96	19.05±6.45	34.13±9.87	56.35	
GPT-SoAY	3.5	one-hop	27.78±8.70	15.97±7.73	3.47±0.00	13.19±7.80	39.58±9.12	67.36	
		two-hop	33.84±4.94	9.60±4.75	6.06±2.81	13.13±7.12	37.37±5.06	71.21	67.30
		three-hop	22.22±6.43	12.70±5.91	9.52±4.42	13.10±6.72	42.46±6.00	64.68	
	3.5-16k	one-hop	28.47±11.67	15.28±6.12	1.39±0.00	17.36±7.78	37.50±9.07	65.97	
		two-hop	35.86±6.01	7.32±3.41	5.30±2.18	15.91±7.16	35.61±4.65	71.46	66.76
		three-hop	23.02±7.16	10.32±4.99	8.33±3.42	17.46±7.37	40.87±6.26	63.89	
	4	one-hop	0.00±0.00	0.00±0.00	1.39±0.00	2.78±0.00	95.83±5.70	95.83	
		two-hop	15.91±4.71	1.26±0.00	9.34±1.07	2.02±1.69	71.46±3.74	87.37	86.57
		three-hop	6.75±0.00	0.40±0.00	14.68±1.68	1.98±0.00	76.19±3.25	82.94	

Results on SoAyBench - Part II

Results on SoAyBench. DS, WS, WP and EE are differenct types of error, ACC denotes a accurate answer, EM means exact match, not only the answer but also the solution. Score is a weighted sum of the ACC score on different question types.

GPT-SoAY	3.5	one-hop	27.78±8.70	15.97±7.73	3.47±0.00	13.19±7.80	39.58±9.12	67.36	
		two-hop	33.84±4.94	9.60±4.75	6.06±2.81	13.13±7.12	37.37±5.06	71.21	67.30
		three-hop	22.22±6.43	12.70±5.91	9.52±4.42	13.10±6.72	42.46±6.00	64.68	
	3.5-16k	one-hop	28.47±11.67	15.28±6.12	1.39±0.00	17.36±7.78	37.50±9.07	65.97	
		two-hop	35.86±6.01	7.32±3.41	5.30±2.18	15.91±7.16	35.61±4.65	71.46	66.76
		three-hop	23.02±7.16	10.32±4.99	8.33±3.42	17.46±7.37	40.87±6.26	63.89	
	4	one-hop	0.00±0.00	0.00±0.00	1.39±0.00	2.78±0.00	95.83±5.70	95.83	
		two-hop	15.91±4.71	1.26±0.00	9.34±1.07	2.02±1.69	71.46±3.74	87.37	86.57
		three-hop	6.75±0.00	0.40±0.00	14.68±1.68	1.98±0.00	76.19±3.25	82.94	
SoAYLLaMA	Chat-7B	one-hop	0.00±0.00	0.00±0.00	0.00±0.00	0.69±0.00	99.31±2.94	99.31	
		two-hop	0.00±0.00	0.00±0.00	20.20±3.84	2.53±1.97	77.27±2.70	77.27	85.76
		three-hop	0.00±0.00	0.00±0.00	9.92±3.56	3.17±2.50	86.90±2.72	86.90	
	Code-7B	one-hop	0.69±0.00	0.00±0.00	0.69±0.00	5.56±4.37	93.06±7.50	93.75	
		two-hop	0.25±0.00	3.28±0.00	7.07±2.75	4.80±3.69	84.60±5.18	84.85	88.95
		three-hop	0.40±0.00	0.00±0.00	4.76±2.14	5.16±4.57	89.68±6.54	90.08	
	Code-13B	one-hop	0.00±0.00	0.00±0.00	1.39±0.00	0.00±0.00	98.61±4.03	98.61	
		two-hop	0.00±0.00	2.27±0.00	14.14±2.14	0.51±0.00	83.08±3.32	83.08	**92.74**
		three-hop	0.00±0.00	0.00±0.00	2.38±2.86	0.40±0.00	97.22±4.28	97.22	

Efficiency & Online Evaluation

To evaluate how efficient are SoAy, we compare the average response time of different methods (second).

Method	7B	13B	3.5	3.5-16k	4
ToolLLaMA	45.10	/	/	/	/
GPT-DFSDT	/	/	39.12	53.73	70.92
SoAyGPT	/	/	6.15	6.40	26.05
SoAyLLaMA-Code	1.12	1.35	/	/	/

To test weather SoAy could meet the need of real-world user requirement, we implement SoAy as an online application, gather 56 real user demands from the logs, and invite 10 annotators to conduct human evaluation.

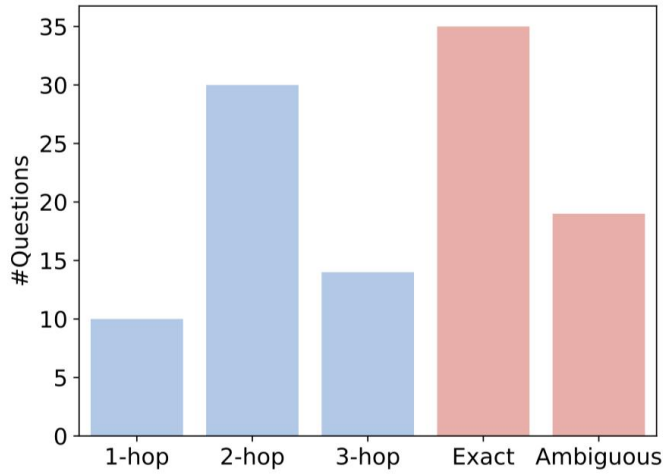


Fig.3 Online Gathered Question statistics.

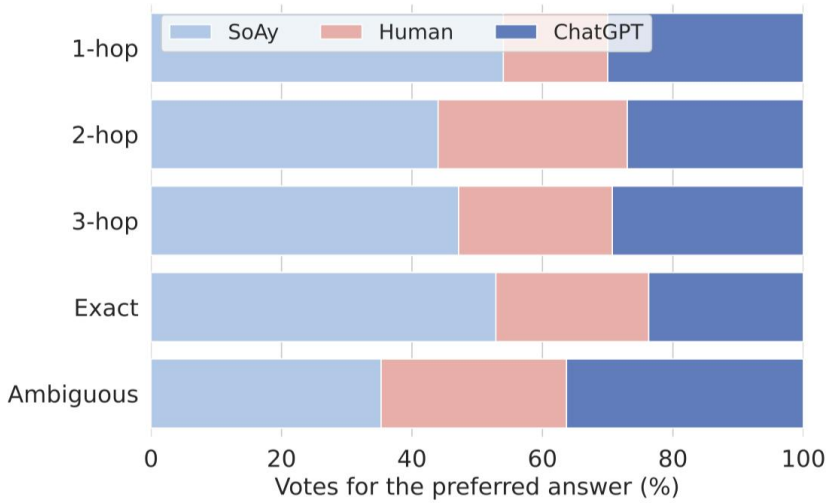


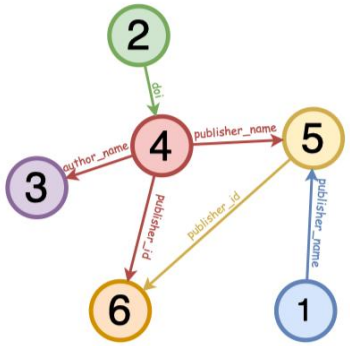
Fig.4 Results of Online Human Evaluation

Deployment on other Academic Platforms

AMiner APIs are NOT the only that face the coupling challenges.
We also deployment SoAy on two other open-sourced scenarios: OpenLibrary and CrossRef

ID	API name	Type	Parameter(s)	Return
1	searchPublisherBySubject	fuzzy	subject	[publisher_name, doi_count]
2	searchWorksByTitle	fuzzy	work_title	[type, author, doi, publisher]
3	searchWorksByAuthor	fuzzy	author_name	[works_title, works_doi]
4	getWorksByDoi	exact	doi	[author_name, work_title, publisher_name, type, reference_count]
5	getPublisherBasicInfo	exact	publisher_name	[publisher_id, current_dois, backfile_dois, total_dois, doi_prefix]
6	getPublisherWorks	exact	publisher_id	[works_title, doi, works_author]

(a) CrossrefAPI Library



(b) CrossrefAPI Graph

Solution	Parameter(s)	Return	Question Template
searchPublisherBySubject	subject	publisher_name	Please list some publishers in the XXX field.
searchPublisherBySubject → getPublisherBasicInfo	subject	publisher_id	Please give me some publishers' id of crossref about the field of XXX.
seachPublisherBySubject → getPublisherBasicInfo → getPublisherWorks	subject	doi	Can you list some articles' DOI numbers in the field of XXX?
searchWorksByTitle	work_title	type	I want to know the type of work XXX.

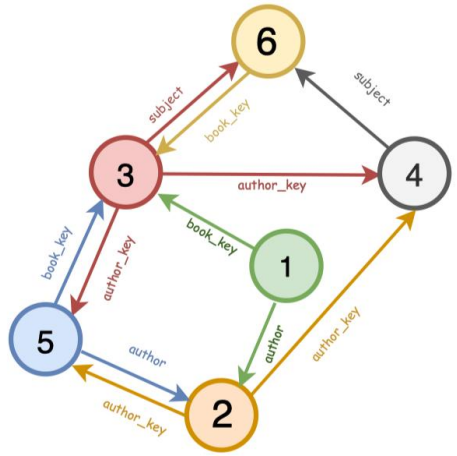
(c) Solution Library (partly shown)

ID	API name	Type	Parameter(s)	Return
1	searchBook	fuzzy	book_info	[book_key, title, author_name, year]
2	searchAuthor	fuzzy	author_info	[author_key, name, list of alternate_names]
3	getBook	exact	book_key	description, list of author, title, first_publish, list of subjects
4	getAuthorBasicInfo	exact	author_key	name, list of alternate_names, birth_date, work_count, top_work, top_subjects
5	getAuthorWorks	exact	author_key, amount	[book_key, title, subjects]
6	searchSubject	fuzzy	subject	[book_key, title]

(a) SoAPI Library

Solution	Parameter(s)	Return	Question Template
searchSubject	subject	list of books	Please list some books on XXX topic.
searchAuthor→getAuthorWorks	author_info	list of books	Which works were written by XXX?
searchBook→getBook	book_info	book_description	Introduce some information about XXX.
searchBook→getBook→getAuthorWorks	book_info	list of books	What other books has the author of XXX written?

(c) Solution Library (partly shown)



(b) SoAPI Graph

Conclusion

Impact and Beneficial Groups

In the context of **LLM tool-learning**, our contributions are as follows:

- We propose **a method** of enabling large-scale models to understand SoAPIs with **complex interrelationships** by utilizing pre-defined solutions.
- By employing code generation techniques, we enable LLM not only to execute multiple APIs but also to support the **execution of APIs** using sequential, branching, and looping structures.
- We release a **benchmark** for evaluating the ability to use SoAPIs with a cloned environment from AMiner.
- We propose a **automation** tool-aware data constructing method and release a **model** SoAyLLaMA, which is trained on these data.

For developers working on **Specific Domain applications**, our contributions are:

- We provide a framework that allows for the **rapid construction** of applications, facilitating natural language interactions between users and service-specific data, using the existing data system APIs (SoAPIs).
- We have applied this framework **in practice** on the AMiner academic information system and validated its feasibility through real user requirements.

Thank you!

Yuanchun's WeChat

Yuanchun's HomePage

Yuanchun's X Page