Can Large Language Models Analyze Graphs like Professionals ? A Benchmark, Datasets and Models

NeurIPS 2024 Datasets and Benchmarks Track

Ran Li

Tsinghua University

Outline

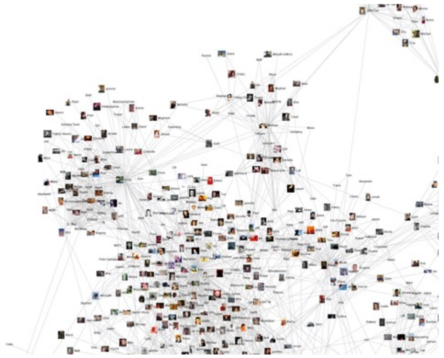
- Background
- Benchmark
- Datasets and Models
- Future Directions

Outline

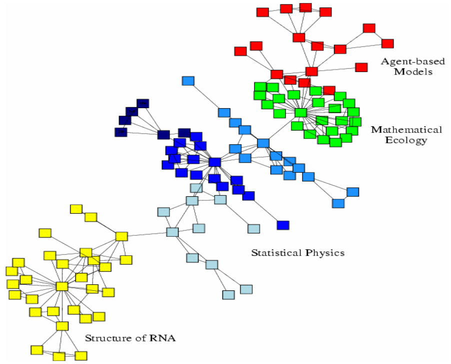
- Background
- Benchmark
- Datasets and Models
- Future Directions

Graph Data

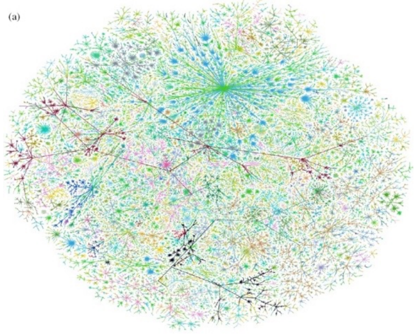
Graph (network) is a common language for describing relational data.



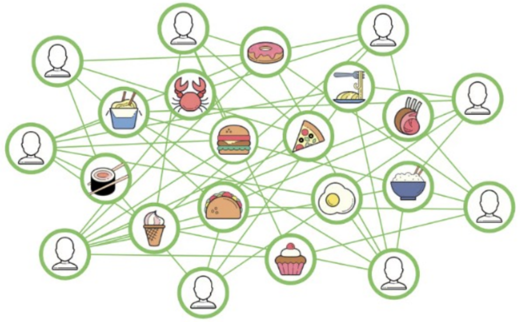
Social Network



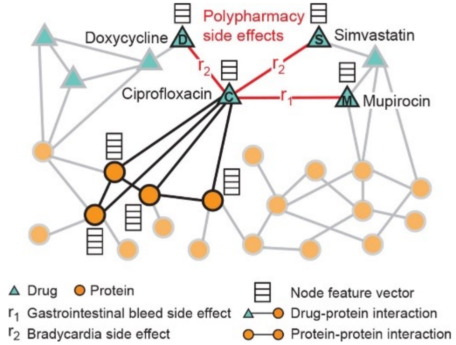
Citation Network



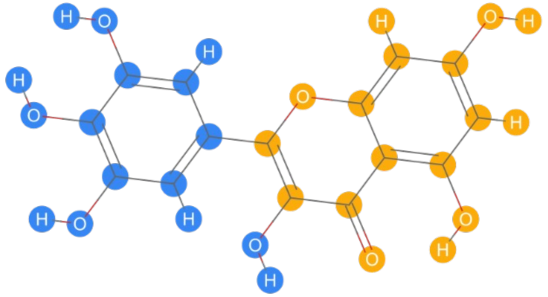
Internet



User-item Graph


Polypharmacy side effects
Doxycycline
D
Simvastatin
S
r2
r2
Ciprofloxacin
C
r1
Mupirocin
M
Drug
Protein
Node feature vector
r1 Gastrointestinal bleed side effect
r2 Bradycardia side effect
Drug-protein interaction
Protein-protein interaction

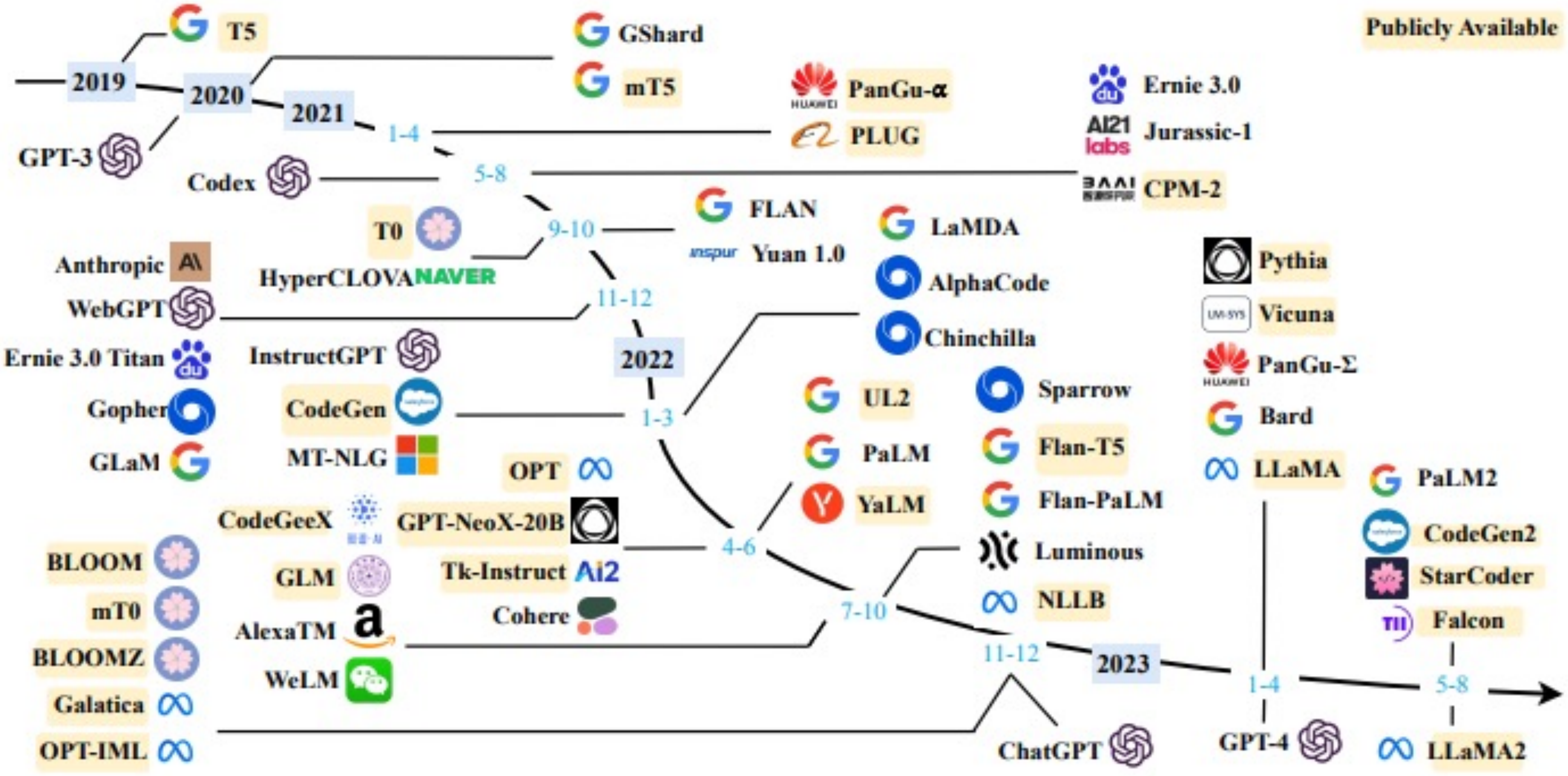
Drug Interaction Graph



Molecule Graph

Large Language Models (LLMs)

With billions of parameters, LLMs have shown abilities towards artificial general intelligence (AGI), e.g., understanding, reasoning, planning, etc.

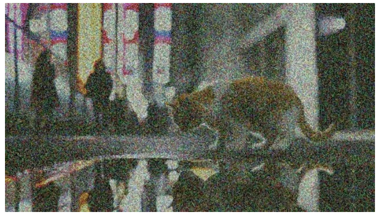


[2] Zhao W X, Zhou K, Li J, et al. A survey of large language models[J]. arXiv preprint arXiv:2303.18223, 2023.

The Need for Graph Reasoning

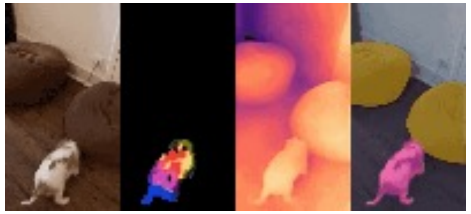
A large model is any model that is trained on broad data and can be adapted to a wide range of downstream tasks.

Language



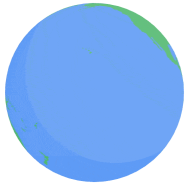
OpenAI × GPT4

Vision



Meta × DINOv2

Speech



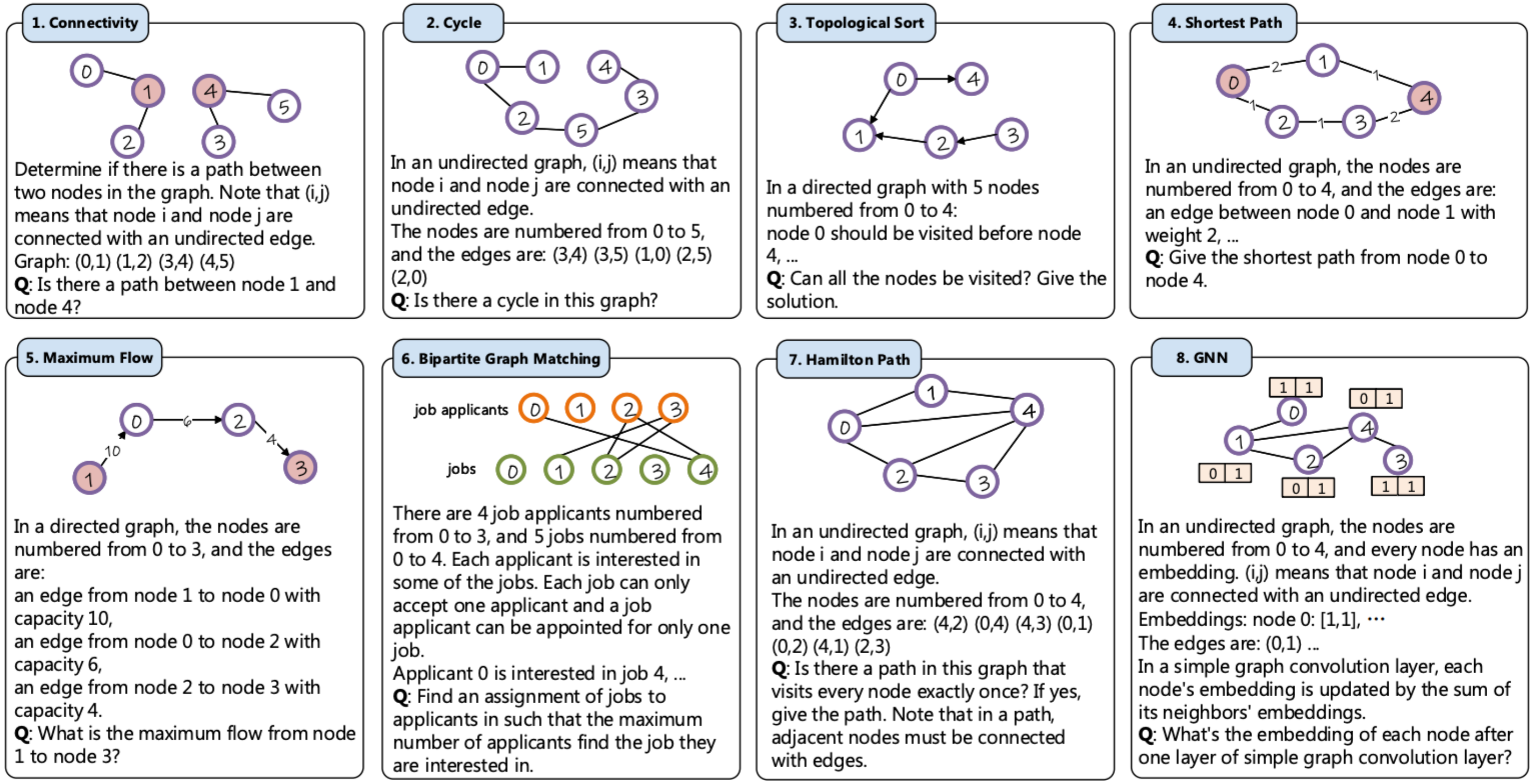
Google × USM

Large models have become a reality in language, vision, and speech, but not good at graph reasoning.

[1] R. Bommasani, D. A. Hudson, E. Adeli, R. Altman, S. Arora, S. von Arx, M. S. Bernstein, J. Bohg, A. Bosselut, E. Brun-skill, et al., "On the opportunities and risks of foundation models," arXiv preprint arXiv:2108.07258, 2021

Previous Benchmarks for Graph Reasoning

NLGraph is a benchmark to explore the capability of large language models in analyzing graph-related problems.



1. Connectivity

Determine if there is a path between two nodes in the graph. Note that (i,j) means that node i and node j are connected with an undirected edge.
Graph: (0,1) (1,2) (3,4) (4,5)
Q: Is there a path between node 1 and node 4?

2. Cycle

In an undirected graph, (i,j) means that node i and node j are connected with an undirected edge.
The nodes are numbered from 0 to 5, and the edges are: (3,4) (3,5) (1,0) (2,5) (2,0)
Q: Is there a cycle in this graph?

3. Topological Sort

In a directed graph with 5 nodes numbered from 0 to 4:
node 0 should be visited before node 4, ...
Q: Can all the nodes be visited? Give the solution.

4. Shortest Path

In an undirected graph, the nodes are numbered from 0 to 4, and the edges are:
an edge between node 0 and node 1 with weight 2, ...
Q: Give the shortest path from node 0 to node 4.

5. Maximum Flow

In a directed graph, the nodes are numbered from 0 to 3, and the edges are:
an edge from node 1 to node 0 with capacity 10,
an edge from node 0 to node 2 with capacity 6,
an edge from node 2 to node 3 with capacity 4.
Q: What is the maximum flow from node 1 to node 3?

6. Bipartite Graph Matching

There are 4 job applicants numbered from 0 to 3, and 5 jobs numbered from 0 to 4. Each applicant is interested in some of the jobs. Each job can only accept one applicant and a job applicant can be appointed for only one job.
Applicant 0 is interested in job 4, ...
Q: Find an assignment of jobs to applicants in such that the maximum number of applicants find the job they are interested in.

7. Hamilton Path

In an undirected graph, (i,j) means that node i and node j are connected with an undirected edge.
The nodes are numbered from 0 to 4, and the edges are: (4,2) (0,4) (4,3) (0,1) (0,2) (4,1) (2,3)
Q: Is there a path in this graph that visits every node exactly once? If yes, give the path. Note that in a path, adjacent nodes must be connected with edges.

8. GNN

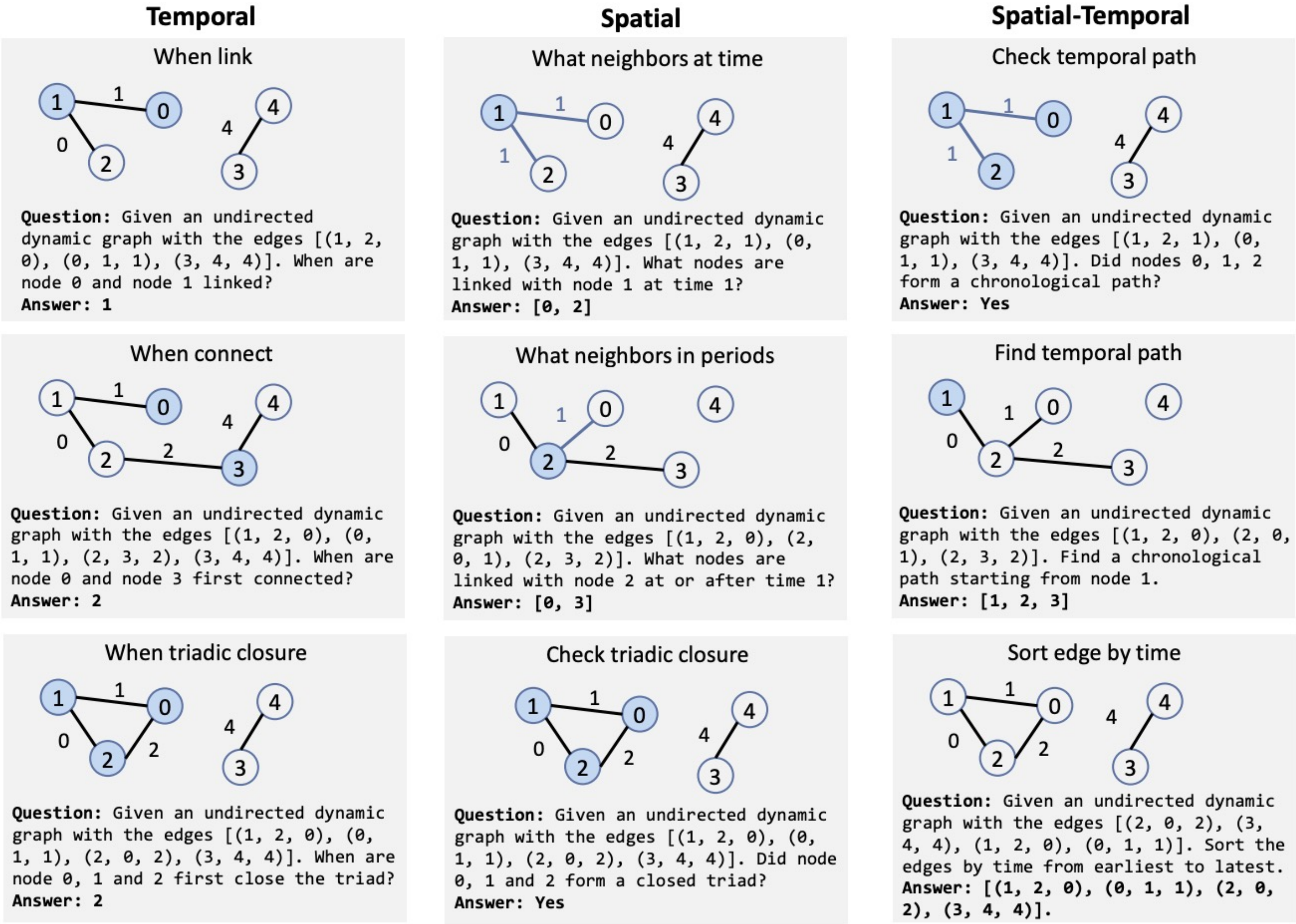
In an undirected graph, the nodes are numbered from 0 to 4, and every node has an embedding. (i,j) means that node i and node j are connected with an undirected edge.
Embeddings: node 0: [1,1], ···
The edges are: (0,1) ...
In a simple graph convolution layer, each node's embedding is updated by the sum of its neighbors' embeddings.
Q: What's the embedding of each node after one layer of simple graph convolution layer?

It has 8 types of problems, including basic graph theory and GNN.

[3] Heng Wang et al. Can language models solve graph problems in natural language? Advances in Neural Information Processing Systems, volume 36, pages 30840–30861. Curran Associates, Inc., 2023.

Previous Benchmarks for Graph Reasoning

LLM4DyG: A benchmark for dynamic graphs with spatial and temporal problems.



Temporal

When link

Question: Given an undirected dynamic graph with the edges [(1, 2, 0), (0, 1, 1), (3, 4, 4)]. When are node 0 and node 1 linked?
Answer: 1

When connect

Question: Given an undirected dynamic graph with the edges [(1, 2, 0), (0, 1, 1), (2, 3, 2), (3, 4, 4)]. When are node 0 and node 3 first connected?
Answer: 2

When triadic closure

Question: Given an undirected dynamic graph with the edges [(1, 2, 0), (0, 1, 1), (2, 0, 2), (3, 4, 4)]. When are node 0, 1 and 2 first close the triad?
Answer: 2

Spatial

What neighbors at time

Question: Given an undirected dynamic graph with the edges [(1, 2, 1), (0, 1, 1), (3, 4, 4)]. What nodes are linked with node 1 at time 1?
Answer: [0, 2]

What neighbors in periods

Question: Given an undirected dynamic graph with the edges [(1, 2, 0), (2, 0, 1), (2, 3, 2)]. What nodes are linked with node 2 at or after time 1?
Answer: [0, 3]

Check triadic closure

Question: Given an undirected dynamic graph with the edges [(1, 2, 0), (0, 1, 1), (2, 0, 2), (3, 4, 4)]. Did node 0, 1 and 2 form a closed triad?
Answer: Yes

Spatial-Temporal

Check temporal path

Question: Given an undirected dynamic graph with the edges [(1, 2, 1), (0, 1, 1), (3, 4, 4)]. Did nodes 0, 1, 2 form a chronological path?
Answer: Yes

Find temporal path

Question: Given an undirected dynamic graph with the edges [(1, 2, 0), (2, 0, 1), (2, 3, 2)]. Find a chronological path starting from node 1.
Answer: [1, 2, 3]

Sort edge by time

Question: Given an undirected dynamic graph with the edges [(2, 0, 2), (3, 4, 4), (1, 2, 0), (0, 1, 1)]. Sort the edges by time from earliest to latest.
Answer: [(1, 2, 0), (0, 1, 1), (2, 0, 2), (3, 4, 4)].

[4] Zeyang Zhang, Xin Wang, Ziwei Zhang, Haoyang Li, Yijian Qin, and Wenwu Zhu. Llm4dyg: Can large language models solve spatial-temporal problems on dynamic graphs? https://arxiv.org/pdf/2310.17110, 2024.

Outline

- Background
- Benchmark
- Datasets and Models
- Future Directions

Motivation of ProGraph Benchmark

Motivation

- Previous LLM benchmarks for graph analysis have several drawbacks:
  - Graphs have to be inputted via prompts, and thus the graph nodes size are quite small.
    - Typically a few dozens of nodes.
  - The benchmarks require step-by-step reasoning ability of LLMs.
    - The reasoning depths of current LLMs are still shadow.
  - The questions are abstract and monotonous in form.
    - Lacking context from real-world application scenarios.

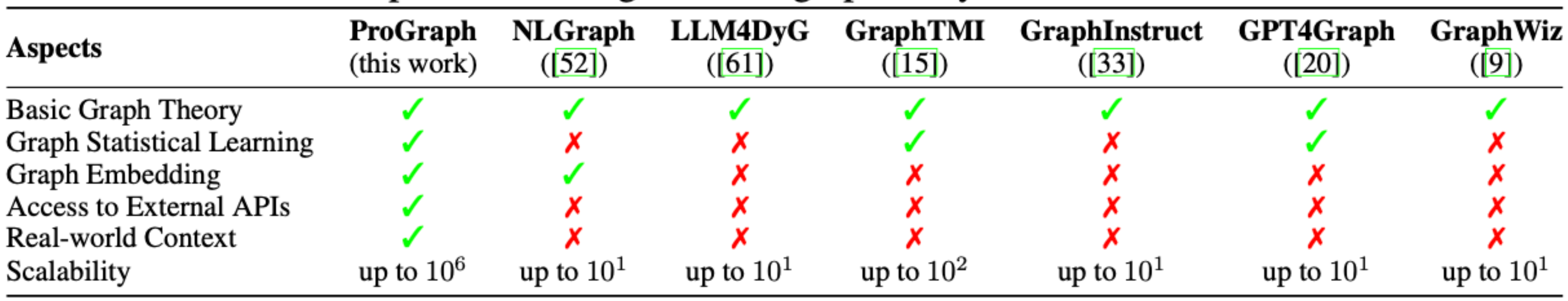
Motivation of ProGraph Benchmark

Motivation

- Consider the scenario that
  - A human expert is asked to find the shortest path in a million-scale graph…
  - She will probably write a few lines of Python codes based on NetworkX, instead of directly reasoning over the raw inputs.

Can large language models analyze graphs like professionals?

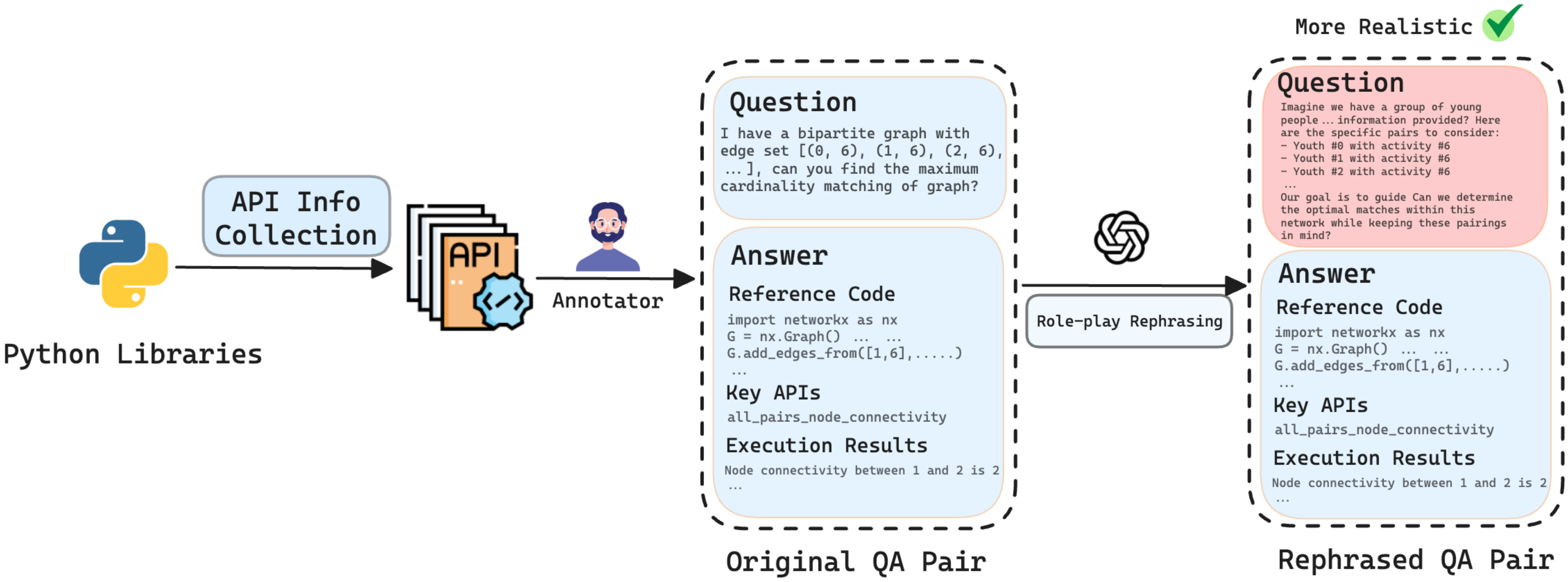
Table 1: Comparisons among different graph analysis benchmarks for LLMs.



Aspects	**ProGraph** (this work)	**NLGraph** ([52])	**LLM4DyG** ([61])	**GraphTMI** ([15])	**GraphInstruct** ([33])	**GPT4Graph** ([20])	**GraphWiz** ([9])
Basic Graph Theory	✓	✓	✓	✓	✓	✓	✓
Graph Statistical Learning	✓	✗	✗	✓	✗	✓	✗
Graph Embedding	✓	✓	✗	✗	✗	✗	✗
Access to External APIs	✓	✗	✗	✗	✗	✗	✗
Real-world Context	✓	✗	✗	✗	✗	✗	✗
Scalability	up to 10^6	up to 10^1	up to 10^1	up to 10^2	up to 10^1	up to 10^1	up to 10^1

ProGraph Benchmark

We propose GraphPro benchmark to evaluate the capability of LLMs in leveraging external APIs for graph analysis.



Original QA Pair

Question

I have a bipartite graph with edge set [(0, 6), (1, 6), (2, 6), …], can you find the maximum cardinality matching of graph?

Answer

Reference Code

```
import networkx as nx
G = nx.Graph()  …   …
G.add_edges_from([1,6],.....)
…
```

Key APIs

all_pairs_node_connectivity

Execution Results

```
Node connectivity between 1 and 2 is 2
…
```

Rephrased QA Pair

Question

Imagine we have a group of young people … information provided? Here are the specific pairs to consider:
- Youth #0 with activity #6
- Youth #1 with activity #6
- Youth #2 with activity #6

…

Our goal is to guide Can we determine the optimal matches within this network while keeping these pairings in mind?

Answer

Reference Code

```
import networkx as nx
G = nx.Graph()  …   …
G.add_edges_from([1,6],.....)
…
```

Key APIs

all_pairs_node_connectivity

Execution Results

```
Node connectivity between 1 and 2 is 2
…
```

ProGraph Benchmark

Statistics of ProGraph and a task example.

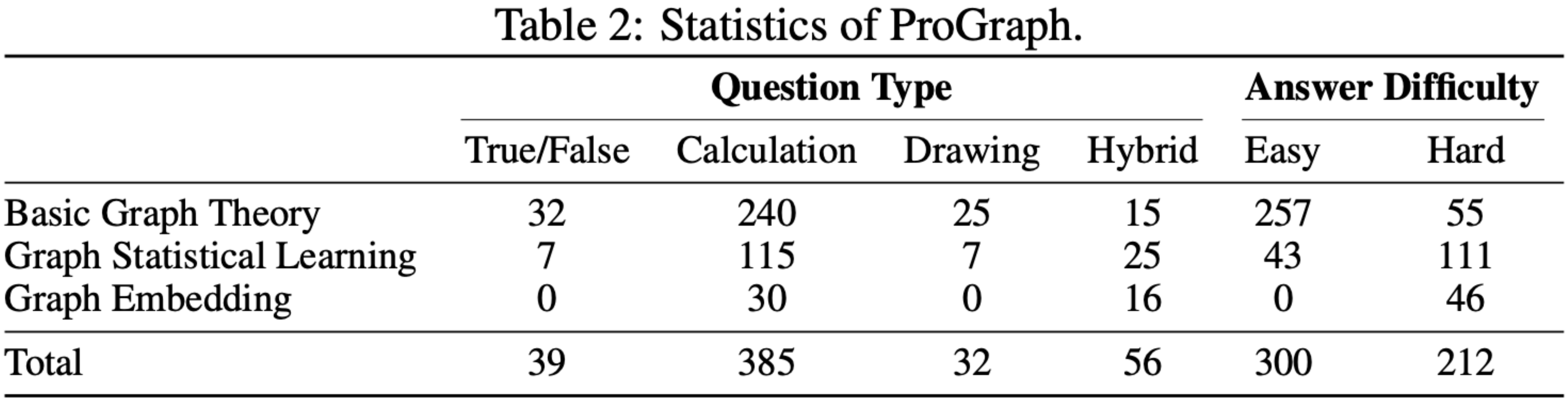
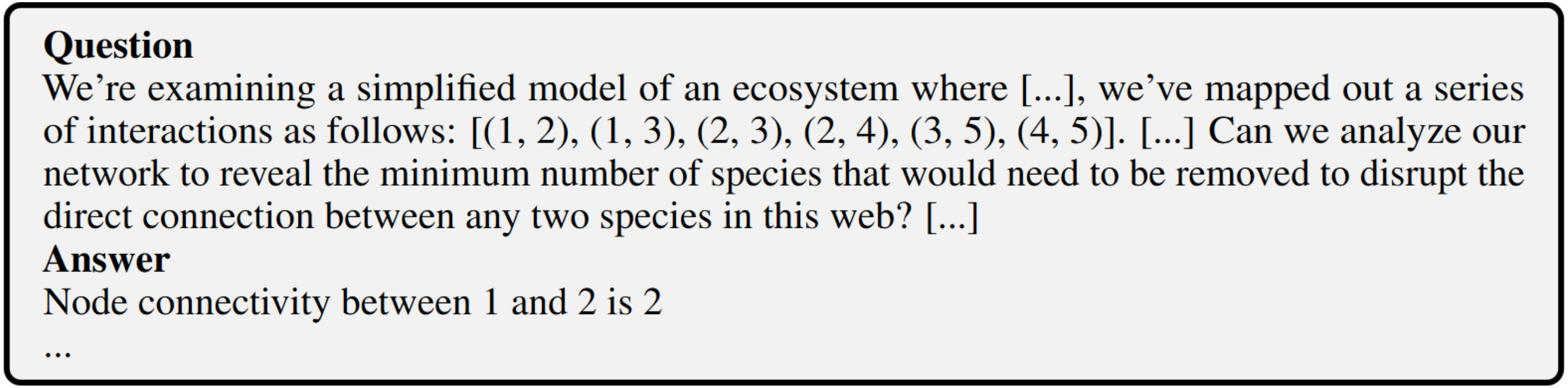


Table 2: Statistics of ProGraph.

	Question Type				Answer Difficulty	
	True/False	Calculation	Drawing	Hybrid	Easy	Hard
Basic Graph Theory	32	240	25	15	257	55
Graph Statistical Learning	7	115	7	25	43	111
Graph Embedding	0	30	0	16	0	46
Total	39	385	32	56	300	212



Question

We're examining a simplified model of an ecosystem where [...], we've mapped out a series of interactions as follows: [(1, 2), (1, 3), (2, 3), (2, 4), (3, 5), (4, 5)]. [...] Can we analyze our network to reveal the minimum number of species that would need to be removed to disrupt the direct connection between any two species in this web? [...]

Answer

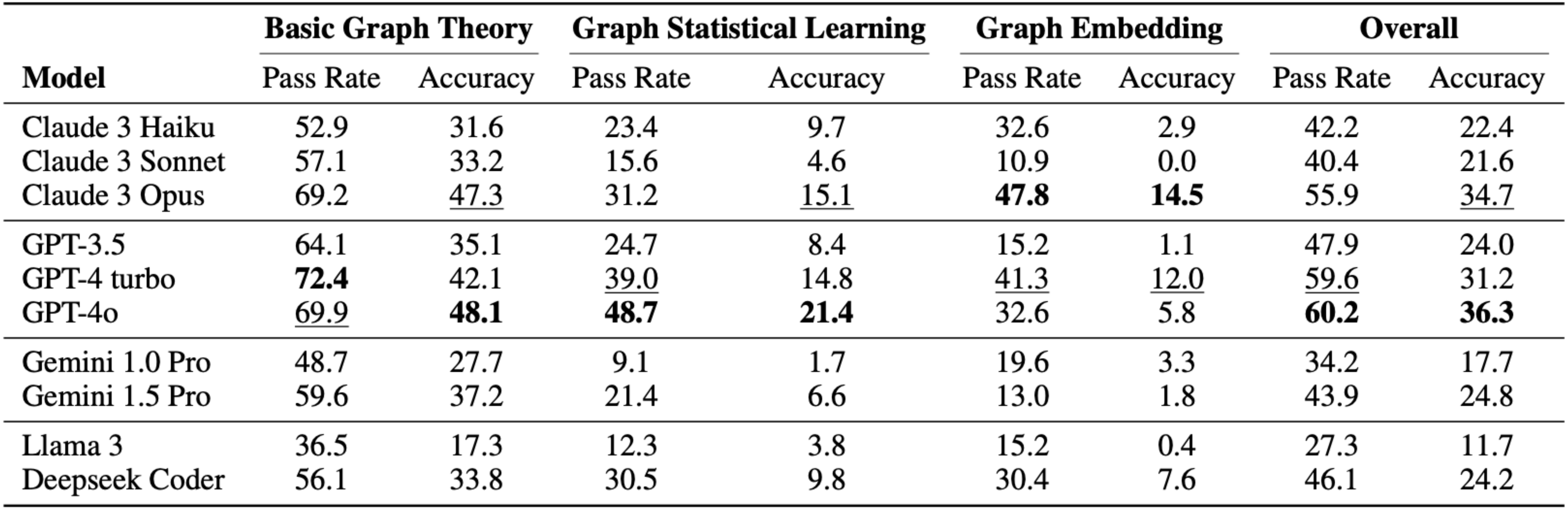
Node connectivity between 1 and 2 is 2

...

Experiments

We conducted experiments on the GraphPro benchmark and evaluated the capability of mainstream LLMs for graph analysis.

Table 6: Performance (%) of different models on ProGraph.



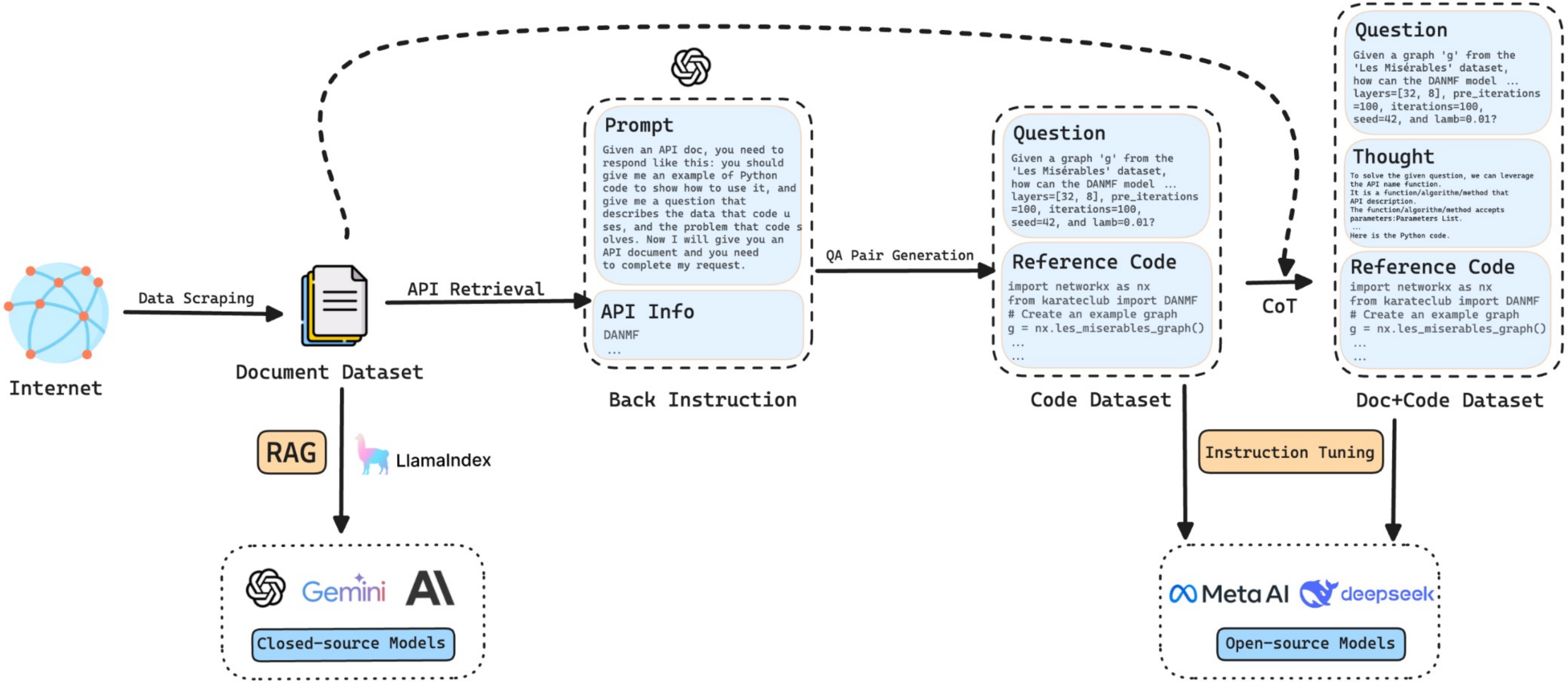
	Basic Graph Theory		**Graph Statistical Learning**		**Graph Embedding**		**Overall**	
Model	Pass Rate	Accuracy	Pass Rate	Accuracy	Pass Rate	Accuracy	Pass Rate	Accuracy
Claude 3 Haiku	52.9	31.6	23.4	9.7	32.6	2.9	42.2	22.4
Claude 3 Sonnet	57.1	33.2	15.6	4.6	10.9	0.0	40.4	21.6
Claude 3 Opus	69.2	47.3	31.2	15.1	**47.8**	**14.5**	55.9	34.7
GPT-3.5	64.1	35.1	24.7	8.4	15.2	1.1	47.9	24.0
GPT-4 turbo	**72.4**	42.1	39.0	14.8	41.3	12.0	59.6	31.2
GPT-4o	69.9	**48.1**	**48.7**	**21.4**	32.6	5.8	**60.2**	**36.3**
Gemini 1.0 Pro	48.7	27.7	9.1	1.7	19.6	3.3	34.2	17.7
Gemini 1.5 Pro	59.6	37.2	21.4	6.6	13.0	1.8	43.9	24.8
Llama 3	36.5	17.3	12.3	3.8	15.2	0.4	27.3	11.7
Deepseek Coder	56.1	33.8	30.5	9.8	30.4	7.6	46.1	24.2

Outline

- Background
- Benchmark
- Datasets and Models
- Future Directions

LLM4Graph Datasets

- We propose LLM4Graph datasets to enhance the capability of LLMs for graph analysis.
  - API documents of six Python libraries:
    - can be used to improve closed-sourced LLMs via RAG（top-k：3，5，7）
  - Auto-generated QA pairs via back-instructing GPT-4:
    - can be used to improve open-sourced LLMs via instruction tuning



Back Instruction

Prompt

Given an API doc, you need to respond like this: you should give me an example of Python code to show how to use it, and give me a question that describes the data that code uses, and the problem that code solves. Now I will give you an API document and you need to complete my request.

API Info

DANMF
...

Code Dataset

Question

Given a graph 'g' from the 'Les Misérables' dataset, how can the DANMF model ... layers=[32, 8], pre_iterations =100, iterations=100, seed=42, and lamb=0.01?

Reference Code

```
import networkx as nx
from karateclub import DANMF
# Create an example graph
g = nx.les_miserables_graph()
...
...
```

Doc+Code Dataset

Question

Given a graph 'g' from the 'Les Misérables' dataset, how can the DANMF model ... layers=[32, 8], pre_iterations =100, iterations=100, seed=42, and lamb=0.01?

Thought

To solve the given question, we can leverage the API name function.
It is a function/algorithm/method that API description.
The function/algorithm/method accepts parameters:Parameters List.
...
Here is the Python code.

Reference Code

```
import networkx as nx
from karateclub import DANMF
# Create an example graph
g = nx.les_miserables_graph()
...
...
```

Closed-source Models

Open-source Models

LLM4Graph Datasets

- We propose LLM4Graph datasets to enhance the capability of LLMs for graph analysis.
    - API documents of six Python libraries:
        - can be used to improve closed-sourced LLMs via RAG（top-k：3，5，7）
    - Auto-generated QA pairs via back-instructing GPT-4:
        - can be used to improve open-sourced LLMs via instruction tuning

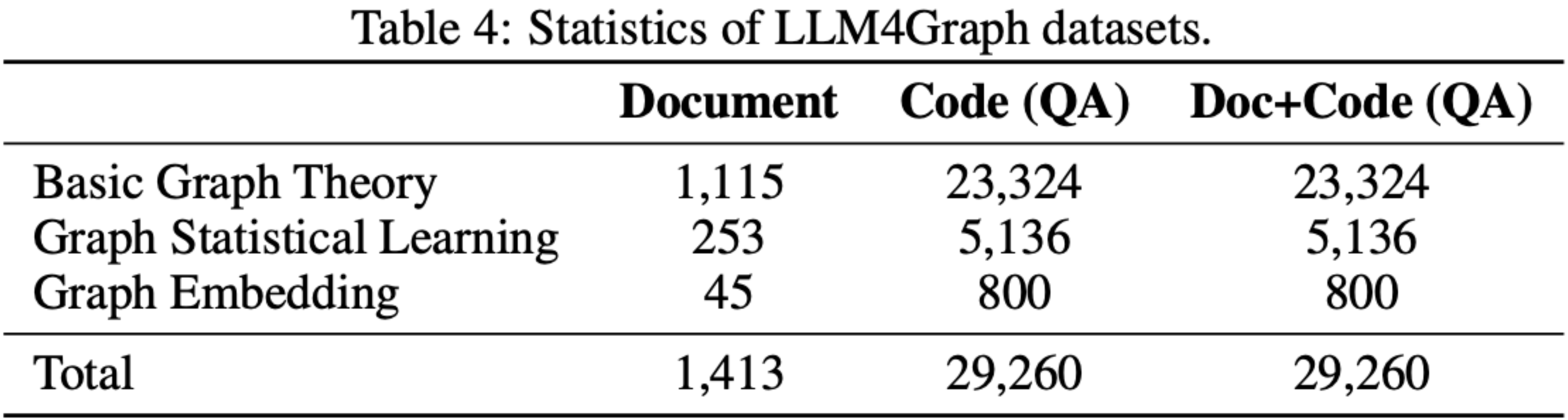


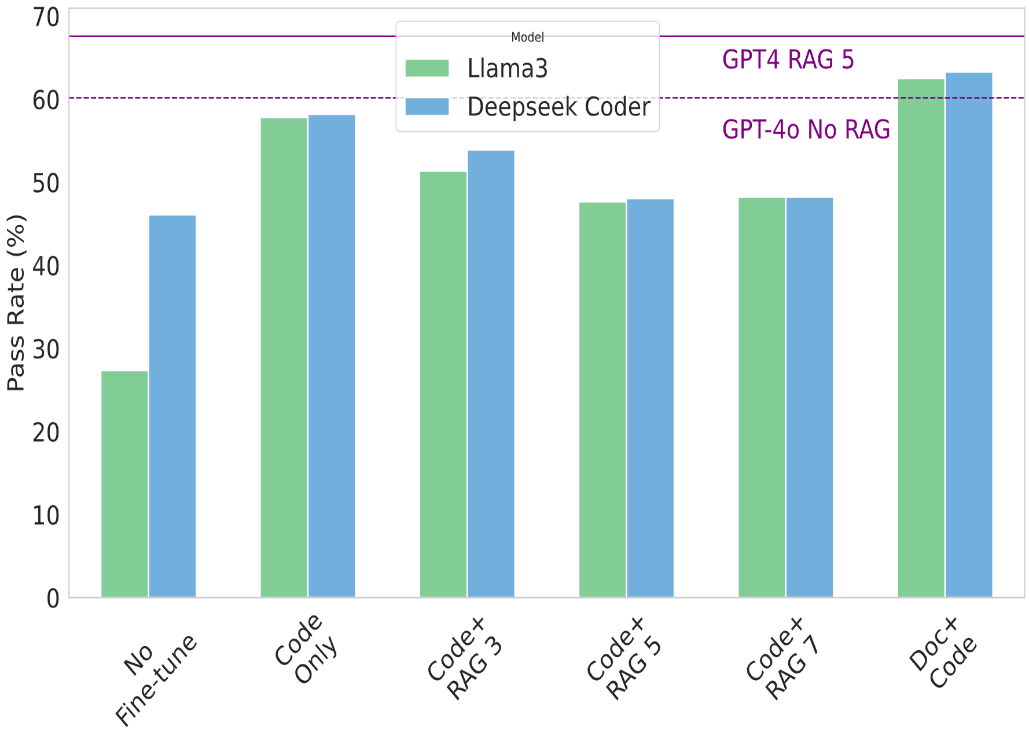
Table 4: Statistics of LLM4Graph datasets.

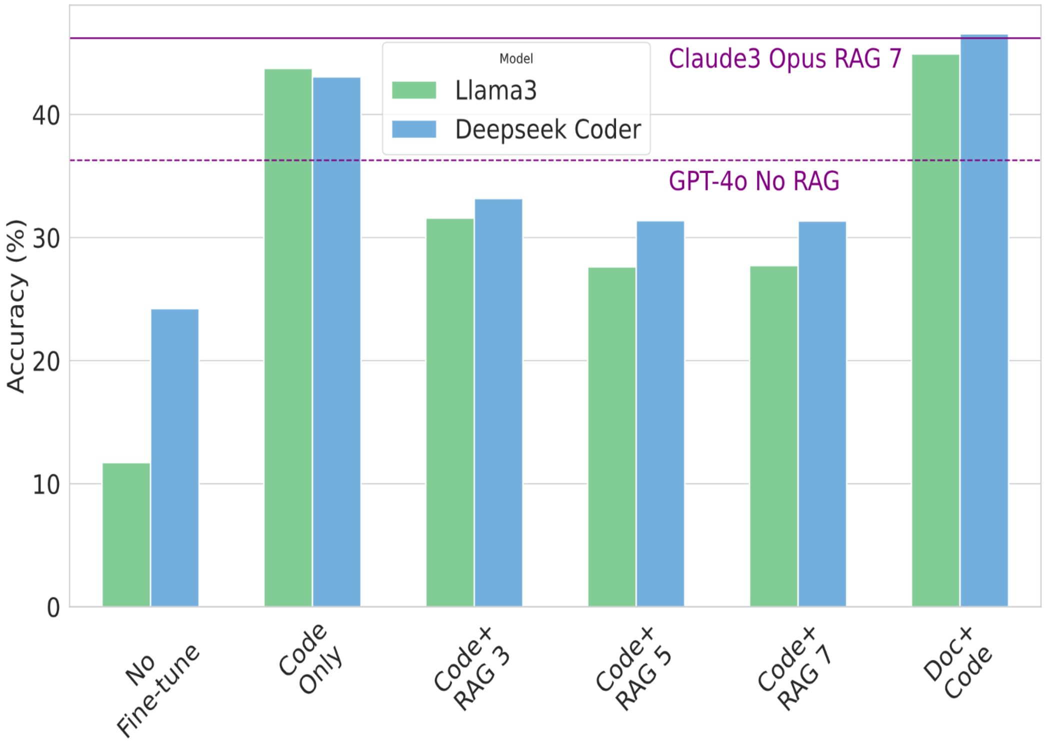
	Document	**Code (QA)**	**Doc+Code (QA)**
Basic Graph Theory	1,115	23,324	23,324
Graph Statistical Learning	253	5,136	5,136
Graph Embedding	45	800	800
Total	1,413	29,260	29,260

Experiments

- The accuracies of closed-source models (Claude, GPT and Gemini) on ProGraph are 25-36%, and can be improved to 37-46% with RAG using LLM4Graph as the retrieval pool.

- The accuracies of open-source models (Llama3 and Deepseek Coder) are only 12-24%, but can be improved to 45-47% through instruction-tuning on LLM4Graph.



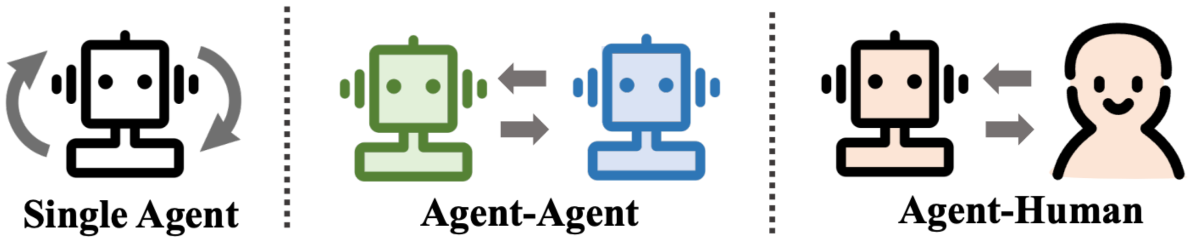


Outline

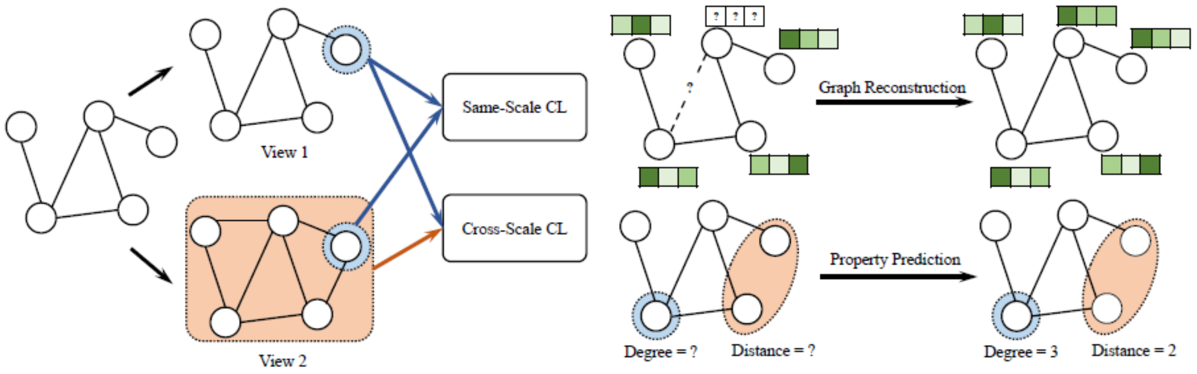
- Background
- Benchmark
- Datasets and Models
- Future Directions

Future Directions

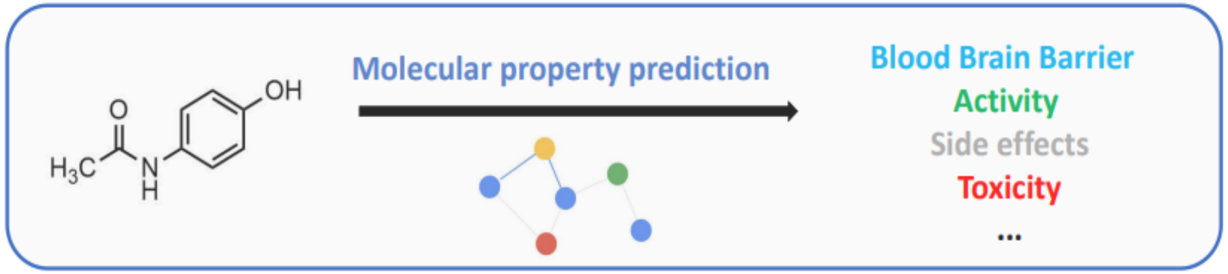
1. LLM-based Agent / Multi-agent System
   - Reasoning/Tool-using/Decision-making
   - Collaboration/Debate/Competition



2. Graph Foundation Model
   - Transformer/Mamba
   - Other general models/methods.



3. Application/Evaluation
   - Drug discovery, Urban Computing…
   - Human/AI Feedback
   - Safety/Privacy Issues



Thanks