media and network lab

KDD2024 BARCELONA, SPAIN

清華大學 Tsinghua University

# LLM4DyG: Can Large Language Models Solve Spatial-Temporal Problems on Dynamic Graphs

Zeyang Zhang[1], Xin Wang[1,2], Ziwei Zhang[1], Haoyang Li[1], Yijian Qin[1], Wenwu Zhu[1,2]

Department of Computer Science and Technology, Tsinghua University[1]

Beijing National Research Center for Information Science and Technology[2]

# Outline

- Background
- Related Works
- LLM4DyG Framework
- Experimental Results
- Conclusion

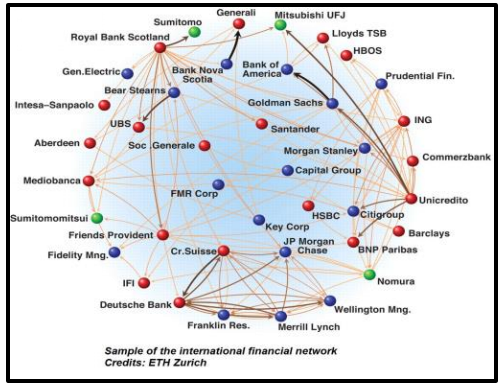
# Background

- Many types of data are dynamic graphs
- Dynamic graphs contain spatial-temporal relations

**Social Networks**

**Device Networks**


**Finance Networks**




**Traffic Networks**

# Background

① In the era of LLMs, we handle data with natural language (NL)

NL Query

When ?
What ?
Whether ?
How ?
Why ?

LLMs





NL Answer

At time ...
It is ...
Yes ...
By ...
Because ...

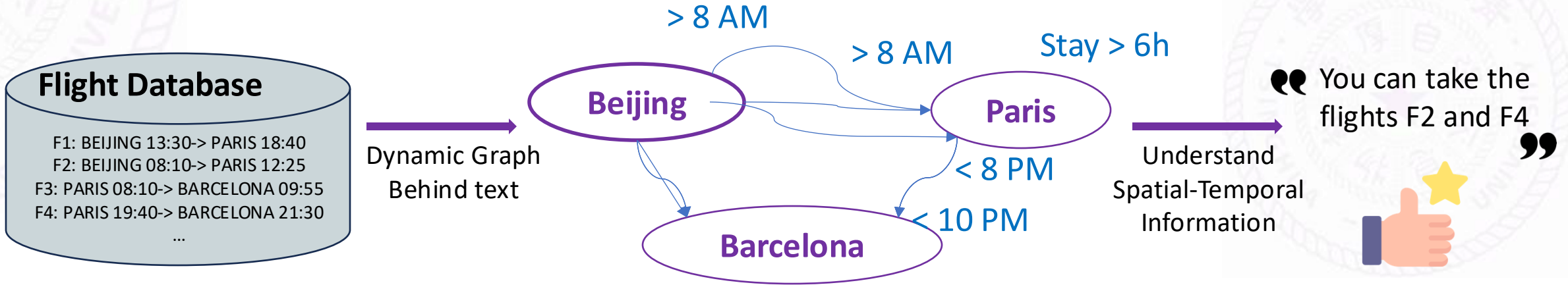
② NL Queries Can Include Implicit Dynamic Graphs
(Spatial-Temporal Relations)

# Background

For example, …

I'd like to fly from Beijing to Barcelona, with a departure time no earlier than 8:00 AM and an arrival time no later than 10:00 PM. Additionally, I'd like to have a half-day layover in Paris during the daytime to visit friends. Could you please suggest suitable flights?

**③ LLMs have to understand spatial-temporal information to give right answers**

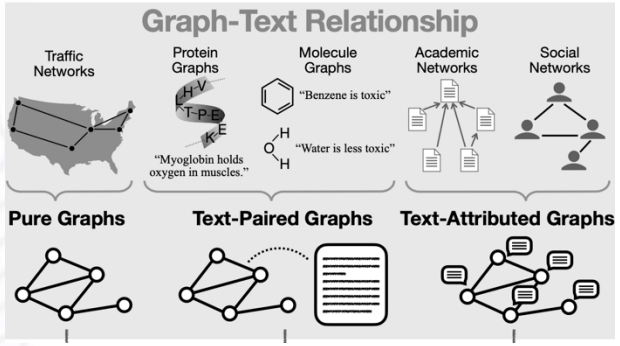


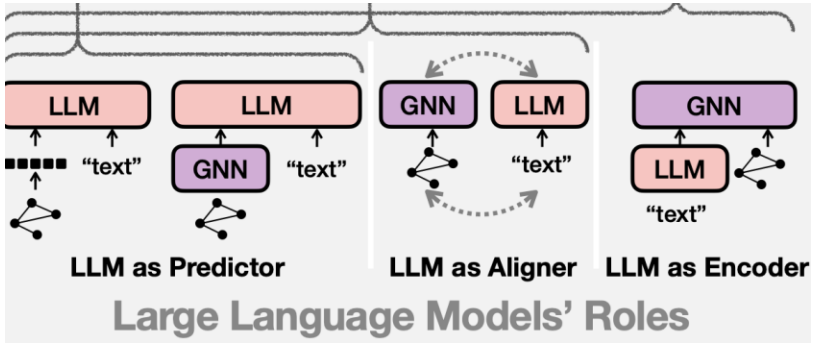

**But … Can LLMs understand the spatial-temporal information on dynamic graphs in natural language?**

**Remained Unexplored !**

# Related Works

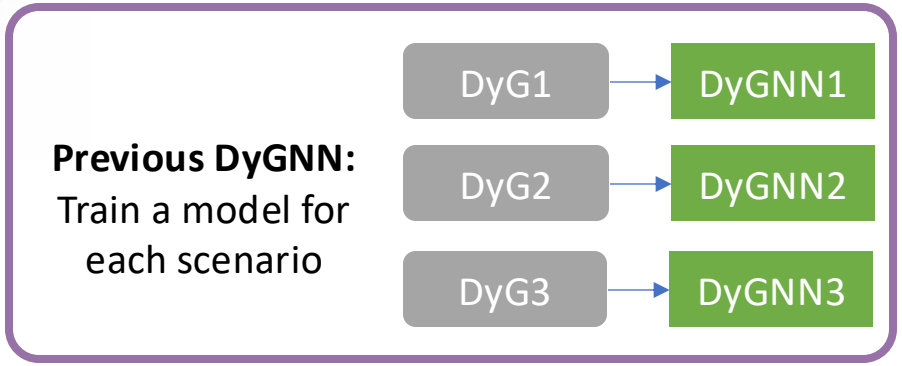
- Existing works of LLMs for static graphs
  - ignore the rich temporal information in ubiquitous dynamic graphs

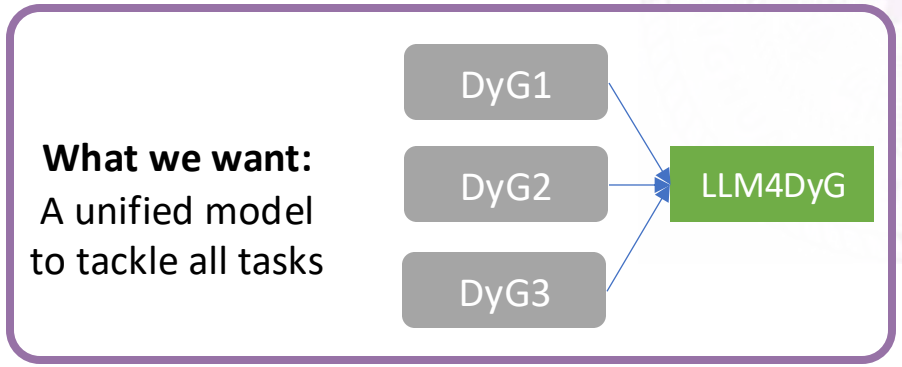



Picture Credit: Large Language Models on Graphs: A Comprehensive Survey

- Existing works for dynamic GNNs
  - unable to tackle spatial-temporal tasks with in-context learning

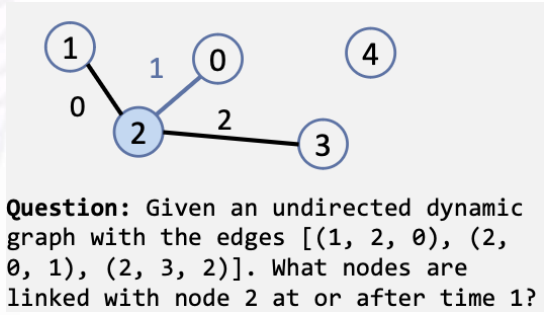


# Challenges

- The first study to evaluate LLMs' spatial-temporal understanding abilities on dynamic graphs, with following challenges:

NL Query → LLMs → NL Answer




**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?



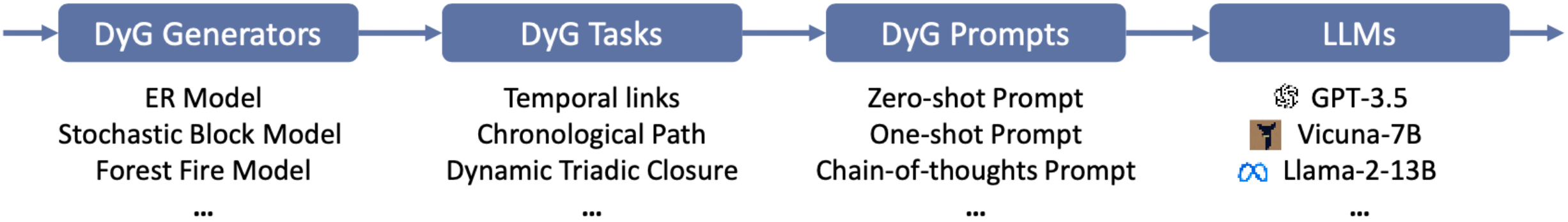


**Answer: [0, 3]**

1. How to assess temporal and structural information separately and simultaneously?
2. How to design prompts to consider spatial-temporal information in natural language?

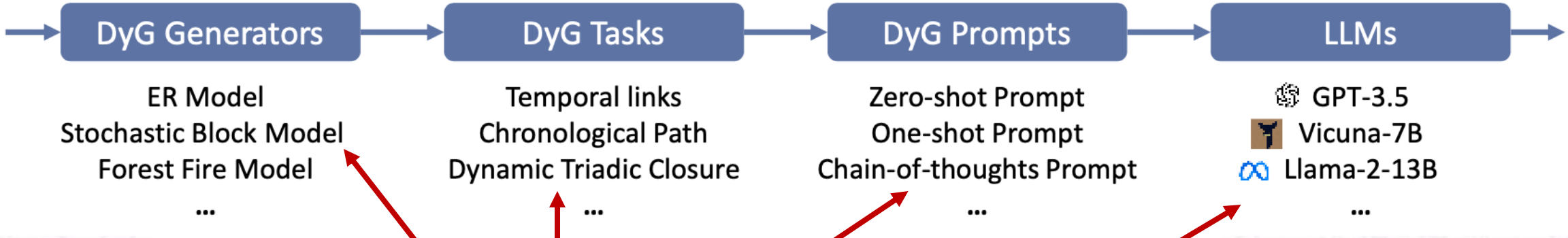
# LLM4DyG Framework

## ① Pipeline & Library with controlled environments and data



# LLM4DyG Framework

## ① Pipeline & Library with controlled environments and data




```python
import llm4dyg
# initialize modules
dygen = llm4dyg.DyGraphGenERCon()
obj_task = llm4dyg.load_task(task, args)
dygprompt = llm4dyg.DyGraphPrompt(obj_task, args = args)

# generate data
info = dygen.sample_dynamic_graph(T = T, N = N , p = p, seed = seed)
qa = obj_task.generate_qa(info)
prompt_qa = dygprompt.generate_prompt_qa(**qa)

# generate response
answer = send_prompt(model, prompt_qa['prompt'])
metric = obj_task.evaluate(qa, answer["content"])
```

Codes are highly modularized & extendable & open-sourced

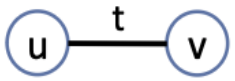
https://github.com/wondergo2017/LLM4DyG

Advantages:

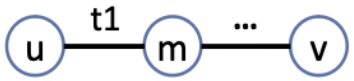
- Controllable environments to test various factors
- Pure evaluation of spatial-temporal understanding

# LLM4DyG Framework

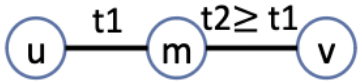
## ② Nine Designed Tasks: *When*, *What*, *Whether* spatial-temporal patterns take place



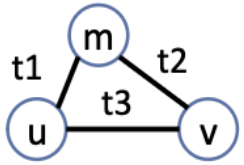
Temporal Link



Path to Connect Nodes



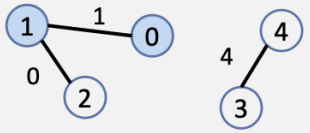
Temporal Path (u→v)



Temporal Triadic Closure

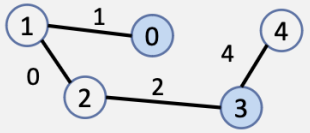
### Temporal

**When link**



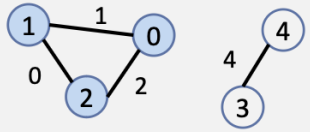
**Question:** Given an undirected dynamic graph with the edges [(1, 2, 0), (0, 1, 1), (3, 4, 4)]. When are node 0 and node 1 linked?
**Answer: 1**

**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**

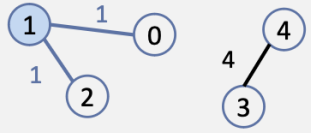
**When triadic closure**



**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**

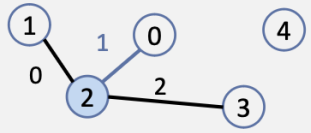
### Spatial

**What neighbors at time**



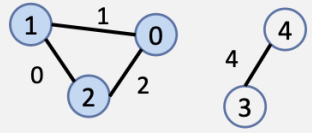
**Question:** Given an undirected dynamic graph with the edges [(1, 2, 1), (0, 1, 1), (3, 4, 4)]. What nodes are linked with node 1 at time 1?
**Answer: [0, 2]**

**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]**

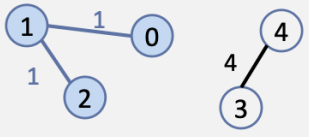
**Check triadic closure**



**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**

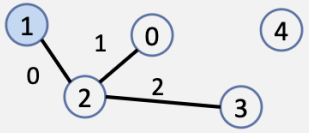
### Spatial-Temporal

**Check temporal path**



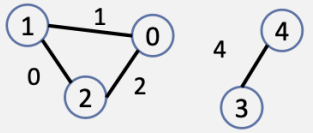
**Question:** Given an undirected dynamic graph with the edges [(1, 2, 1), (0, 1, 1), (3, 4, 4)]. Did nodes 0, 1, 2 form a chronological path?
**Answer: Yes**

**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]**

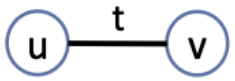
**Sort edge by time**



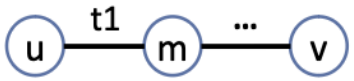
**Question:** Given an undirected dynamic graph with the edges [(2, 0, 2), (3, 4, 4), (1, 2, 0), (0, 1, 1)]. Sort the edges by time from earliest to latest.
**Answer: [(1, 2, 0), (0, 1, 1), (2, 0, 2), (3, 4, 4)].**

# LLM4DyG Framework

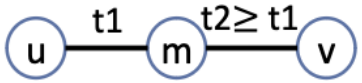
## ② Nine Designed Tasks: *When*, *What*, *Whether* spatial-temporal patterns take place



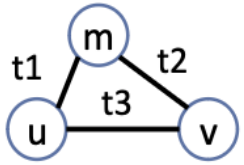
Temporal Link



Path to Connect Nodes



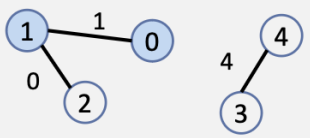
Temporal Path (u→v)



Temporal Triadic Closure

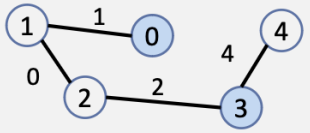
### Temporal

**When link**



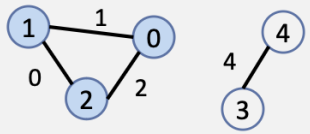
**Question:** Given an undirected dynamic graph with the edges [(1, 2, 0), (0, 1, 1), (3, 4, 4)]. When are node 0 and node 1 linked?
**Answer: 1**

**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**

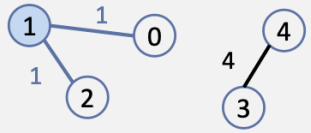
**When triadic closure**



**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**

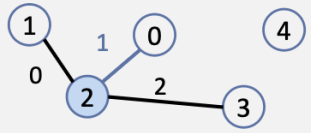
### Spatial

**What neighbors at time**



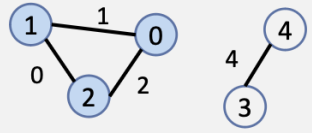
**Question:** Given an undirected dynamic graph with the edges [(1, 2, 1), (0, 1, 1), (3, 4, 4)]. What nodes are linked with node 1 at time 1?
**Answer: [0, 2]**

**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]**

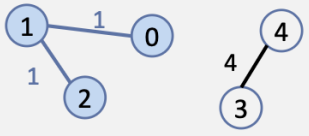
**Check triadic closure**



**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**

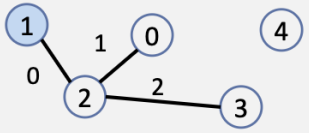
### Spatial-Temporal

**Check temporal path**



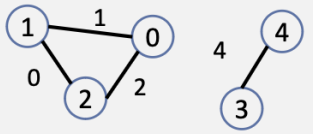
**Question:** Given an undirected dynamic graph with the edges [(1, 2, 1), (0, 1, 1), (3, 4, 4)]. Did nodes 0, 1, 2 form a chronological path?
**Answer: Yes**

**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]**

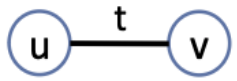
**Sort edge by time**



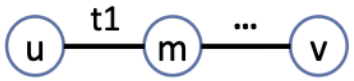
**Question:** Given an undirected dynamic graph with the edges [(2, 0, 2), (3, 4, 4), (1, 2, 0), (0, 1, 1)]. Sort the edges by time from earliest to latest.
**Answer: [(1, 2, 0), (0, 1, 1), (2, 0, 2), (3, 4, 4)].**

# LLM4DyG Framework

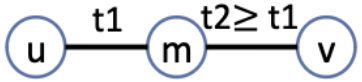
## ② Nine Designed Tasks: *When*, *What*, *Whether* spatial-temporal patterns take place



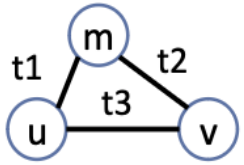
Temporal Link



Path to Connect Nodes



Temporal Path (u→v)



Temporal Triadic Closure

From temporal to spatial-temporal

From local to global

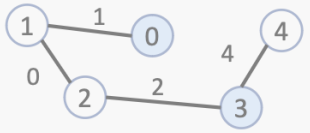
From simple to complex

**Temporal**

When link

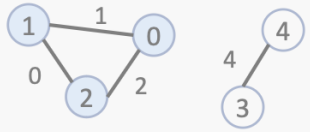
**Question:** Given an undirected dynamic graph with the edges [(1, 2, 0), (0, 1, 1), (3, 4, 4)]. When are node 0 and node 1 linked?
**Answer:** 1

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

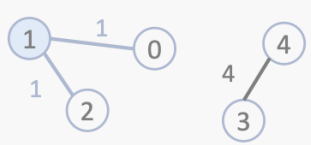
When triadic closure



**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

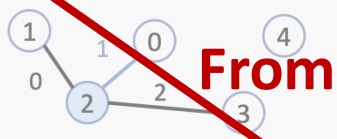
**Spatial**

What neighbors at time



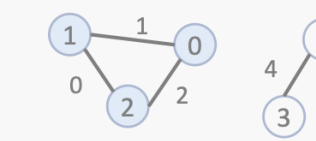
**Question:** Given an undirected dynamic graph with the edges [(1, 2, 1), (0, 1, 1), (3, 4, 4)]. What nodes are linked with node 1 at time 1?
**Answer:** [0, 2]

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]

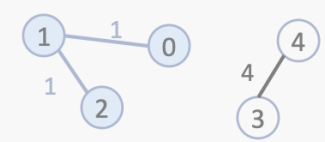
Check triadic closure



**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

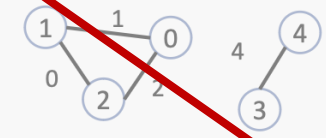


**Question:** Given an undirected dynamic graph with the edges [(1, 2, 1), (0, 1, 1), (3, 4, 4)]. Did nodes 0, 1, 2 form a chronological path?
**Answer:** Yes

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)].

# Experimental Results - Main

➢Obs.1. LLMs show preliminary spatial-temporal abilities

| Task | | Temporal | | | Spatial | | | Spatial-Temporal | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Data | model | when link | when connect | when tclosure | neighbor at time | neighbor in periods | check tclosure | check tpath | find tpath | sort edge |
| N = 5 | GPT-3.5 | 68.0±2.8 | 97.7±0.9 | 52.7±2.4 | 86.0±2.2 | 42.3±1.7 | 69.0±2.2 | 58.7±2.1 | 79.0±4.1 | 78.0±1.4 |
| | Random | 3.2 | 20.0 | 20.0 | 3.2 | 3.2 | 50.0 | 50.0 | 9.3 | 13.1 |
| | Δ | +64.8 | +77.7 | +32.7 | +82.8 | +39.1 | +19.0 | +8.7 | +69.7 | +64.9 |
| N = 10 | GPT-3.5 | 33.7±2.1 | 77.0±2.9 | 73.0±1.6 | 34.0±1.4 | 15.7±4.2 | 66.7±4.5 | 63.7±2.6 | 78.3±6.0 | 29.3±4.0 |
| | Random | 3.2 | 20.0 | 20.0 | 0.1 | 0.1 | 50.0 | 50.0 | 6.7 | 0.0 |
| | Δ | +30.4 | +57.0 | +53.0 | +33.9 | +15.6 | +16.7 | +13.7 | +71.6 | +29.3 |
| N = 20 | GPT-3.5 | 40.3±1.7 | 17.7±4.2 | 63.3±0.9 | 17.7±1.7 | 2.0±0.8 | 64.3±7.3 | 57.0±2.2 | 85.0±0.8 | 0.0±0.0 |
| | Random | 3.2 | 20.0 | 20.0 | 0.0 | 0.0 | 50.0 | 50.0 | 7.3 | 0.0 |
| | Δ | +37.1 | -2.3 | +43.3 | +17.7 | +2.0 | +14.3 | +7.0 | +77.7 | 0.0 |
| Avg. | GPT-3.5 | 47.3±1.2 | 64.1±0.3 | 63.0±1.0 | 45.9±3.1 | 20.0±0.8 | 66.7±2.9 | 59.8±0.8 | 80.8±0.3 | 35.8±2.0 |
| | Random | 3.2 | 20.0 | 20.0 | 1.1 | 1.1 | 50.0 | 50.0 | 7.8 | 4.4 |
| | Δ | +44.1 | +44.1 | +43.0 | +44.8 | +18.9 | +16.7 | +9.8 | +73.0 | +31.4 |

Understand data and reason instead of random guess

# Experimental Results - Data

- Obs.2. Most tasks have increasing difficulty for LLMs as the graph size grows

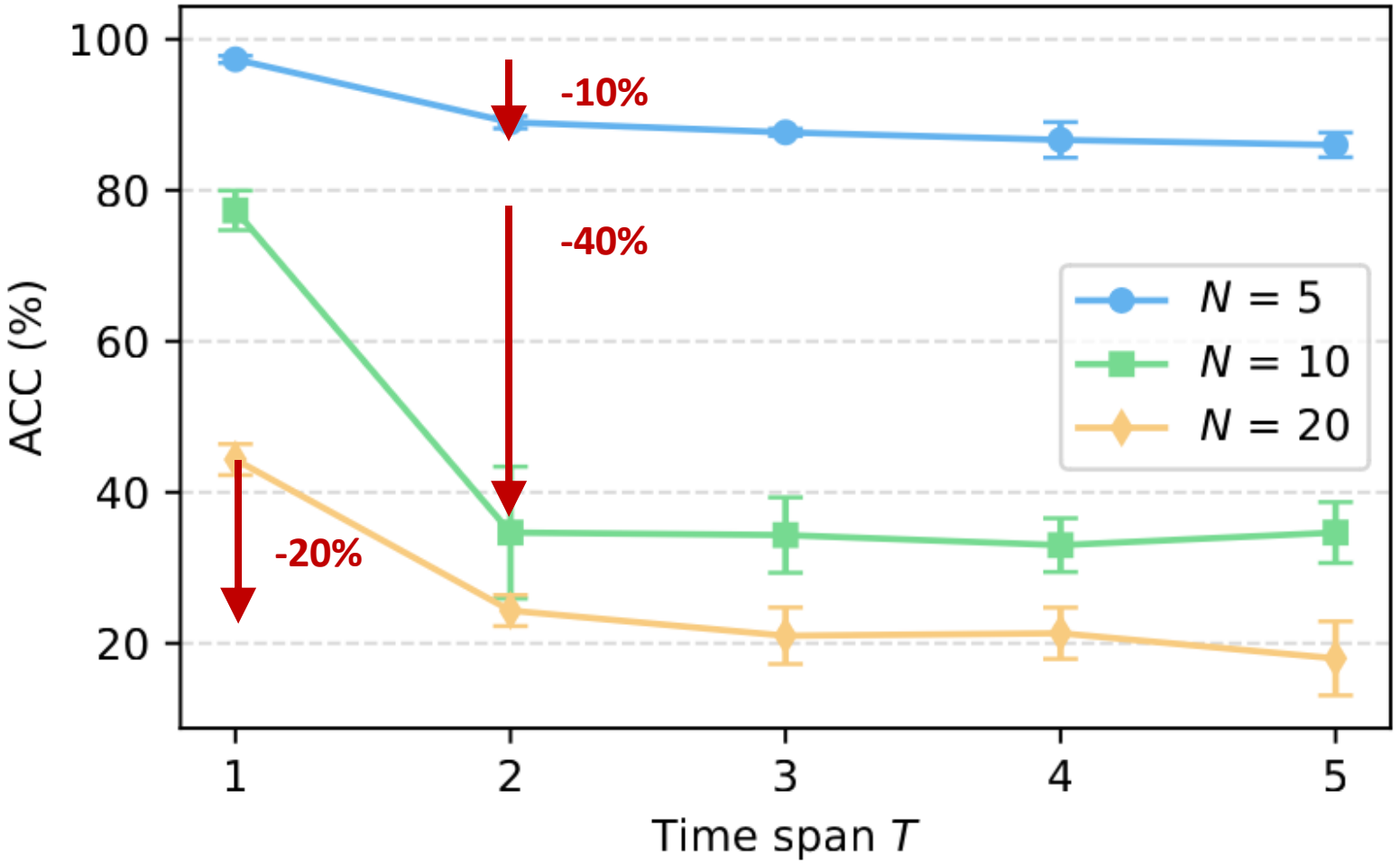
| Task | | Temporal | | | Spatial | | | Spatial-Temporal | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Data | model | when link | when connect | when tclosure | neighbor at time | neighbor in periods | check tclosure | check tpath | find tpath | sort edge |
| N = 5 | GPT-3.5 | 68.0±2.8 | 97.7±0.9 | 52.7±2.4 | 86.0±2.2 | 42.3±1.7 | 69.0±2.2 | 58.7±2.1 | 79.0±4.1 | 78.0±1.4 |
| | Random | 3.2 | 20.0 | 20.0 | 3.2 | 3.2 | 50.0 | 50.0 | 9.3 | 13.1 |
| | Δ | +64.8 | +77.7 | +32.7 | +82.8 | +39.1 | +19.0 | +8.7 | +69.7 | +64.9 |
| N = 10 | GPT-3.5 | 33.7±2.1 | 77.0±2.9 | 73.0±1.6 | 34.0±1.4 | 15.7±4.2 | 66.7±4.5 | 63.7±2.6 | 78.3±6.0 | 29.3±4.0 |
| | Random | 3.2 | 20.0 | 20.0 | 0.1 | 0.1 | 50.0 | 50.0 | 6.7 | 0.0 |
| | Δ | +30.4 | +57.0 | +53.0 | +33.9 | +15.6 | +16.7 | +13.7 | +71.6 | +29.3 |
| N = 20 | GPT-3.5 | 40.3±1.7 | 17.7±4.2 | 63.3±0.9 | 17.7±1.7 | 2.0±0.8 | 64.3±7.3 | 57.0±2.2 | 85.0±0.8 | 0.0±0.0 |
| | Random | 3.2 | 20.0 | 20.0 | 0.0 | 0.0 | 50.0 | 50.0 | 7.3 | 0.0 |
| | Δ | +37.1 | -2.3 | +43.3 | +17.7 | +2.0 | +14.3 | +7.0 | +77.7 | 0.0 |
| Avg. | GPT-3.5 | 47.3±1.2 | 64.1±0.3 | 63.0±1.0 | 45.9±3.1 | 20.0±0.8 | 66.7±2.9 | 59.8±0.8 | 80.8±0.3 | 35.8±2.0 |
| | Random | 3.2 | 20.0 | 20.0 | 1.1 | 1.1 | 50.0 | 50.0 | 7.8 | 4.4 |
| | Δ | +44.1 | +44.1 | +43.0 | +44.8 | +18.9 | +16.7 | +9.8 | +73.0 | +31.4 |

-20% -52% -27% -3% -49%

-60% -17% -13% -2% -29%

# Experimental Results - Data

- Obs.4. Temporal information adds additional difficulties compared to static graphs.



# Experimental Results - Prompt

➢Obs.5. General prompting techniques do not consistently help

| Task | Temporal | | | Spatial | | | Spatial-Temporal | | |
|---|---|---|---|---|---|---|---|---|---|
| Prompt Method | when link | when connect | when tclosure | neighbor at time | neighbor in periods | check tclosure | check tpath | find tpath | sort edge |
| zero-shot | 2.3±0.5 | 73.3±2.1 | 68.0±0.8 | **36.0±4.3** | 4.3±2.1 | **70.7±1.7** | **66.0±5.4** | 56.3±9.0 | 33.7±7.4 |
| one-shot | **33.7±2.1** | **77.0±2.9** | 73.0±1.6 | 34.0±1.4 | **15.7±4.2** | 66.7±4.5 | 63.7±2.6 | 78.3±6.0 | 29.3±4.0 |
| zero-shot COT | 1.0±0.8 | 58.3±1.2 | 70.0±1.6 | 32.0±0.8 | 4.3±2.6 | 55.0±1.4 | 62.3±2.9 | 58.0±9.1 | **44.7±0.5** |
| one-shot COT | 10.3±0.5 | 76.0±2.4 | **80.0±1.6** | 27.7±1.9 | 13.0±3.6 | 57.7±2.1 | 57.7±3.4 | **81.3±2.6** | 24.7±2.4 |

**Mixed Results for COT**

# Experimental Results - Prompt

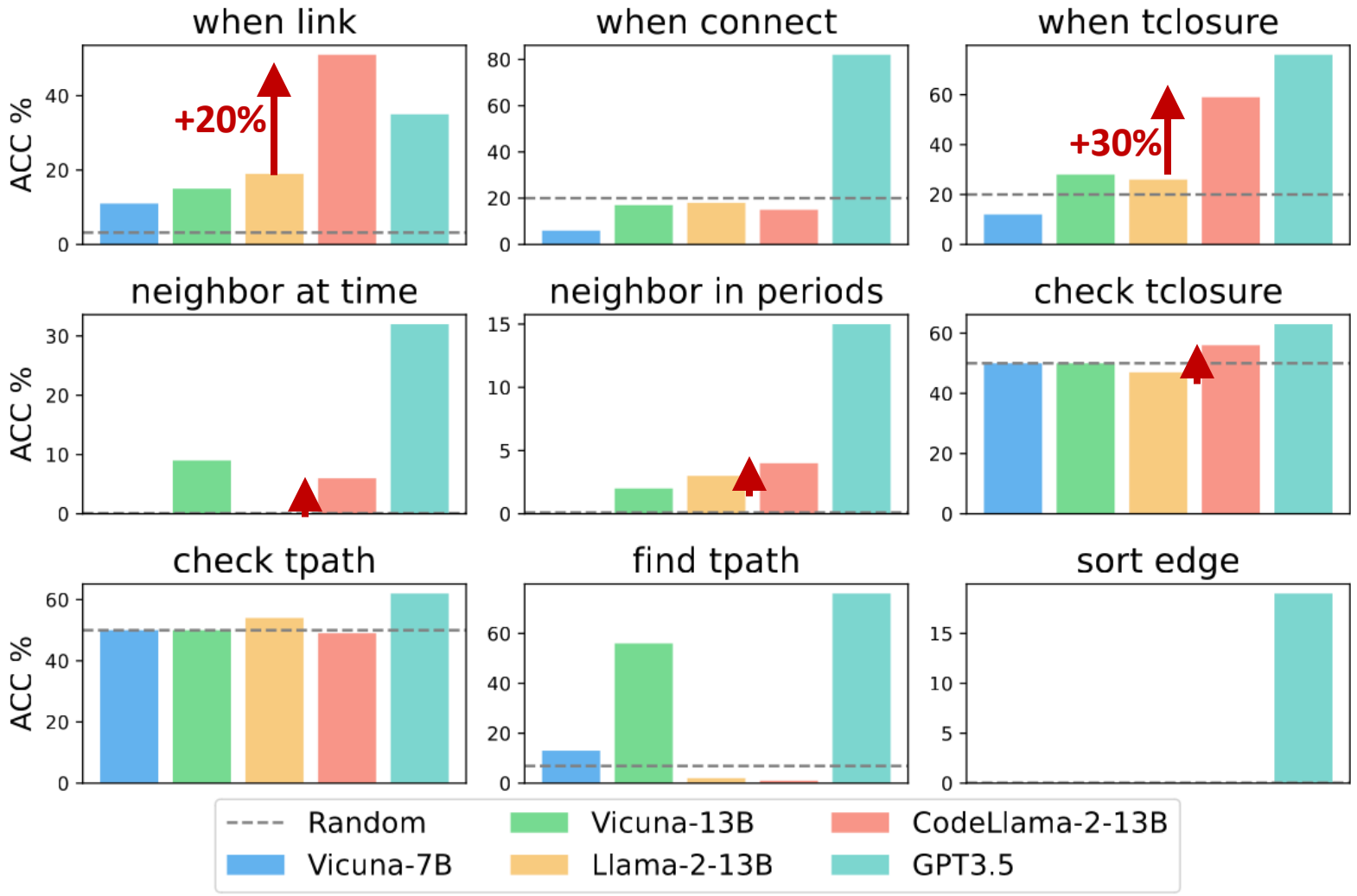
- Obs.6. We explore Disentangled Spatial-Temporal Thoughts: instructing LLMs to separately tackle spatial and temporal information significantly improves the performance

| Task | Temporal | | | Spatial | | | Spatial-Temporal | | |
|---|---|---|---|---|---|---|---|---|---|
| Prompting methods | when link | when connect | when tclosure | neighbor at time | neighbor in periods | check tclosure | check tpath | find tpath | sort edge |
| one-shot prompt | 33.7±2.1 | 77.0±2.9 | 73.0±1.6 | 34.0±1.4 | 15.7±4.2 | 66.7±4.5 | **63.7**±2.6 | 78.3±6.0 | 29.3±4.0 |
| v1: Think (about) nodes and then time | 40.0±1.6 | 77.0±4.1 | **74.0**±1.4 | 34.0±0.8 | 15.0±4.2 | **69.3**±1.7 | 61.0±3.3 | **79.0**±7.5 | 30.0±3.6 |
| v2: Think (about) time and then nodes | 37.3±2.6 | 76.7±3.4 | 73.3±0.5 | 31.7±1.9 | 15.7±3.4 | 67.0±2.9 | 61.3±1.9 | **79.0**±7.5 | **30.7**±3.9 |
| v3: Pick nodes and then time | 59.3±2.1 | **77.0**±2.4 | 68.0±0.8 | 35.0±2.9 | **16.7**±4.7 | 65.0±3.7 | 62.3±2.9 | 78.0±5.4 | 30.0±2.9 |
| v4: Pick time and then nodes | **76.7**±1.7 | 76.3±3.9 | 68.7±0.9 | **35.7**±2.5 | 15.3±3.3 | 65.3±2.9 | 63.3±2.6 | 78.3±5.8 | 29.3±2.9 |

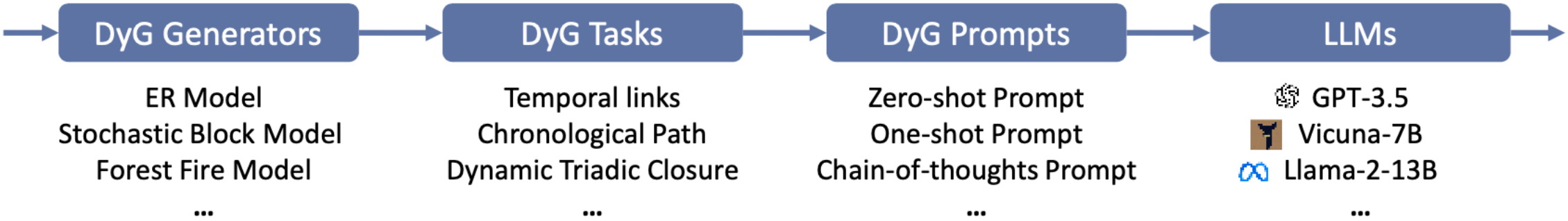
+40%

# Experimental Results - Model

- Obs.7. Training on codes usually helps
- Obs.8 Model scale matters



# Conclusion




- The first study to evaluate LLMs' spatial-temporal understanding abilities on dynamic graphs
- We propose LLM4DyG: nine specially designed tasks for LLMs with controllable environments and data generation.
- We analyze the impacts of various factors including different data generators, data statistics, prompting techniques, and LLMs on the model performance.
- We propose Disentangled Spatial-Temporal Thoughts (DST2) prompt method to enhance LLMs' spatial-temporal understanding abilities.

# Thanks!

For more details, please check out our paper !

[KDD'24] LLM4DyG: Can Large Language Models Solve Spatial-Temporal Problems on Dynamic Graphs


Contact:

Zeyang Zhang

Final-year Ph.D. in Tsinghua University

zy-zhang20@mails.tsinghua.edu.cn

https://zzythu.com


Code:

https://github.com/wondergo2017/LLM4DyG