MACIEJ BESTA

WITH TORSTEN HOEFLER & MANY MEMBERS OF SPCL

Graphs in the AI Era: Towards Next-Generation Graph Databases

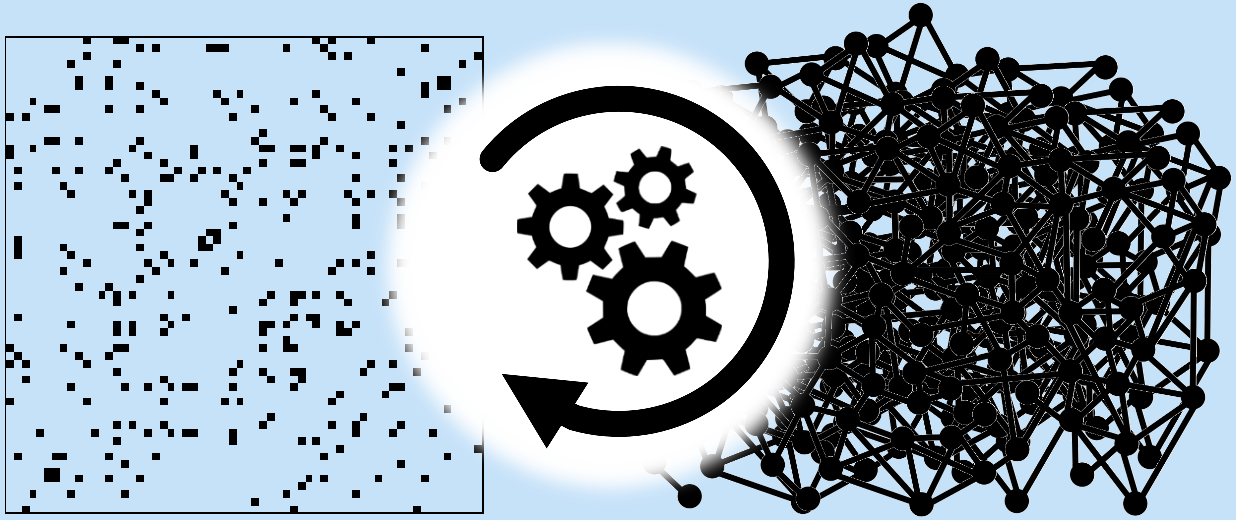
My Research Overview

My Research Overview

Sparse & irregular computing

My Research Overview

Sparse & irregular computing

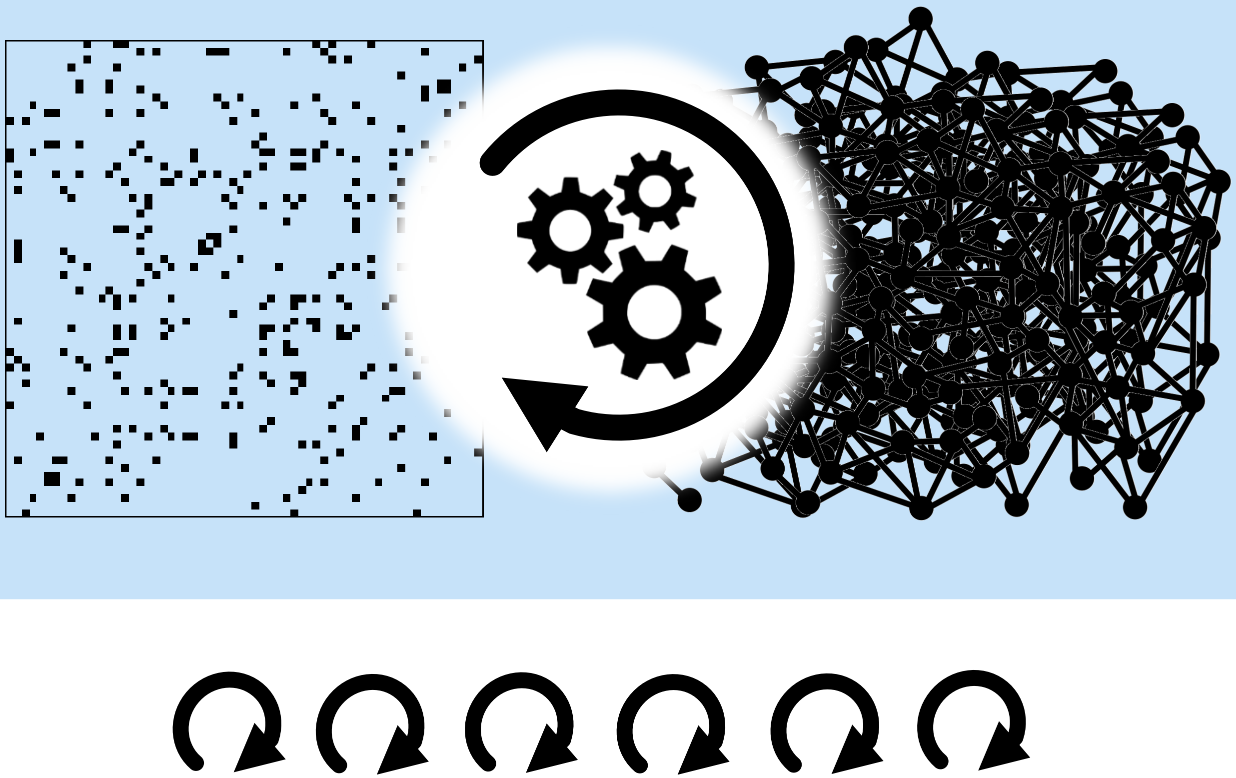


My Research Overview

Sparse & irregular computing

...

- LLM ecosystems & agent infrastructures
- Generative AI training & inference
- Representation Learning & Graph Neural Networks
- Graph databases



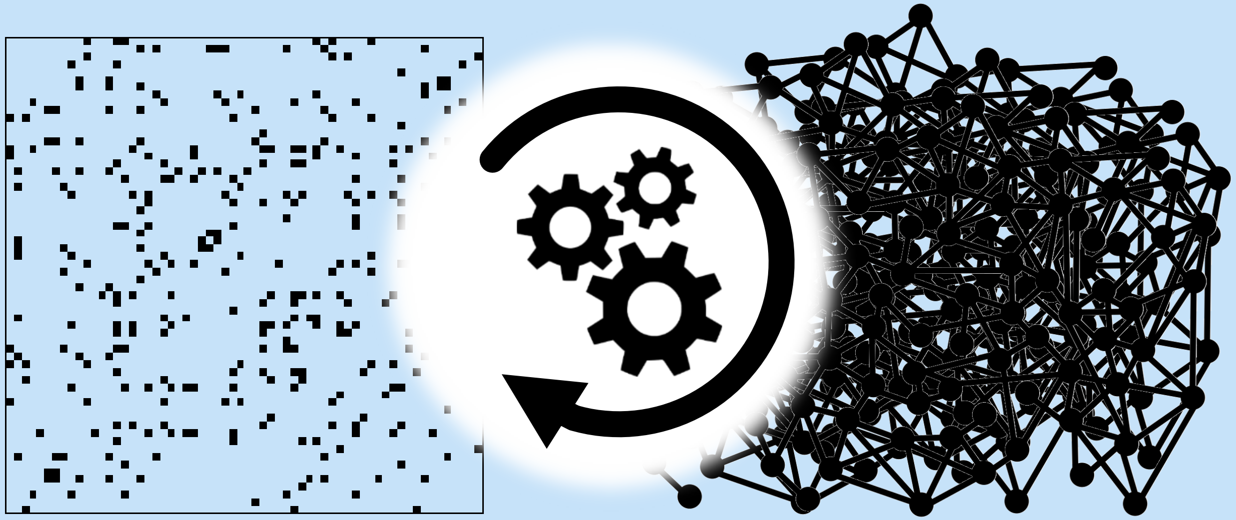
- Graph compression & summarization
- Graph pattern matching
- [Parallel] graph algorithms
- Streaming, dynamic, temporal graphs

My Research Overview

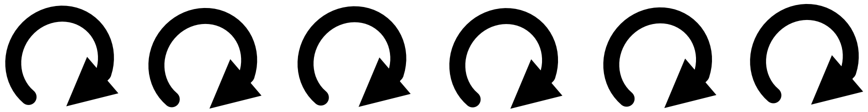
Sparse & irregular computing

...

- LLM ecosystems & agent infrastructures
- Generative AI training & inference
- Representation Learning & Graph Neural Networks
- Graph databases



- Graph compression & summarization
- Graph pattern matching
- [Parallel] graph algorithms
- Streaming, dynamic, temporal graphs



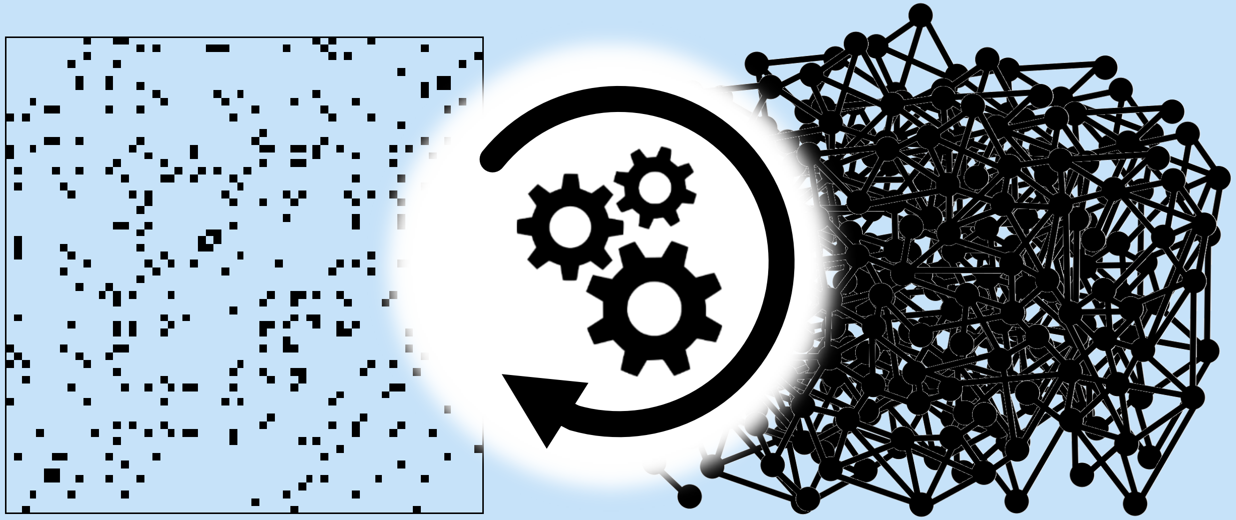
Massively parallel settings

My Research Overview

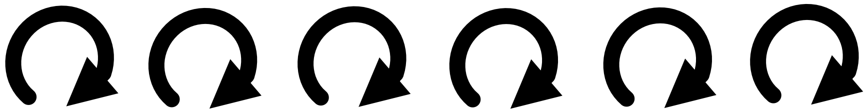
Sparse & irregular computing

…

- LLM ecosystems & agent infrastructures
- Generative AI training & inference
- Representation Learning & Graph Neural Networks
- Graph databases



- Graph compression & summarization
- Graph pattern matching
- [Parallel] graph algorithms
- Streaming, dynamic, temporal graphs



Fast

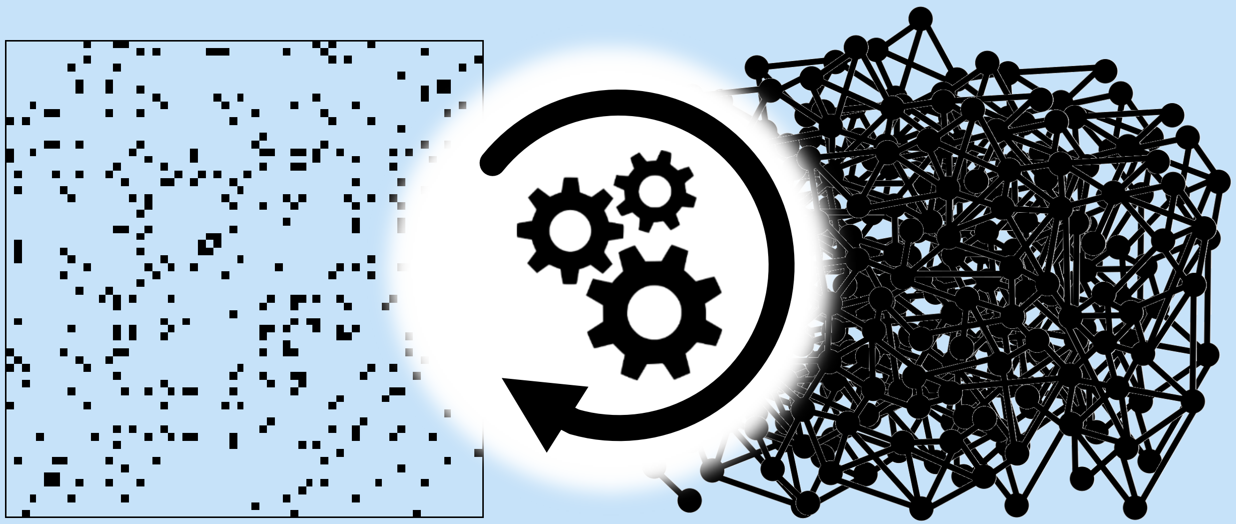
Massively parallel settings

My Research Overview

Sparse & irregular computing

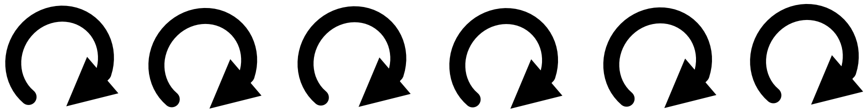
...

- LLM ecosystems & agent infrastructures
- Generative AI training & inference
- Representation Learning & Graph Neural Networks
- Graph databases



- Graph compression & summarization
- Graph pattern matching
- [Parallel] graph algorithms
- Streaming, dynamic, temporal graphs

Effective



Fast

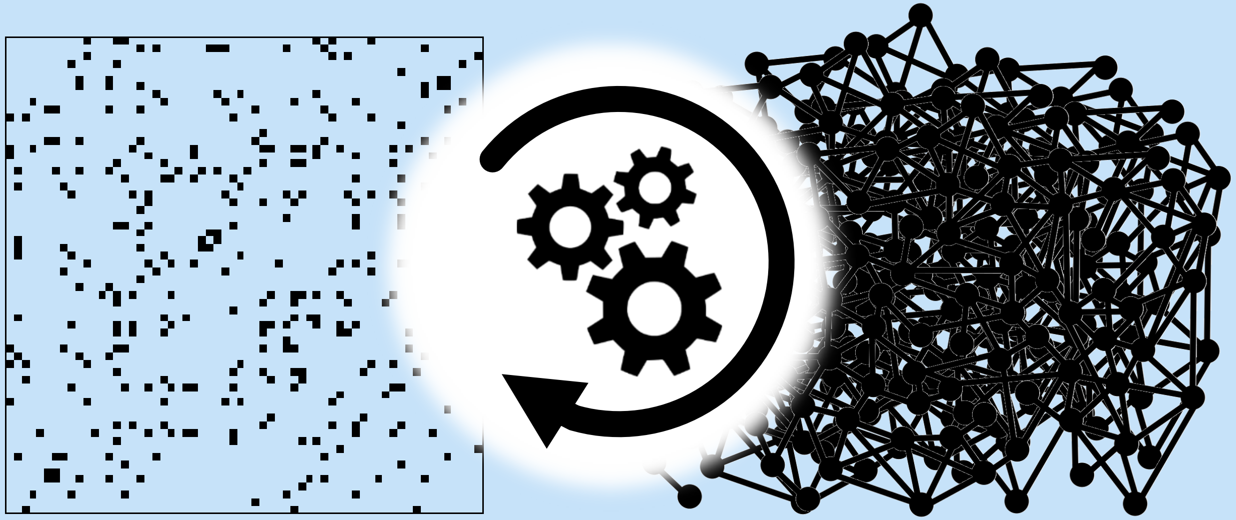
Massively parallel settings

My Research Overview

Sparse & irregular computing

...

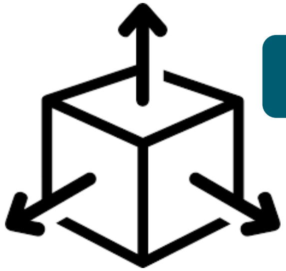
- LLM ecosystems & agent infrastructures
- Generative AI training & inference
- Representation Learning & Graph Neural Networks
- Graph databases



- Graph compression & summarization
- Graph pattern matching
- [Parallel] graph algorithms
- Streaming, dynamic, temporal graphs

Effective

Fast



Scalable

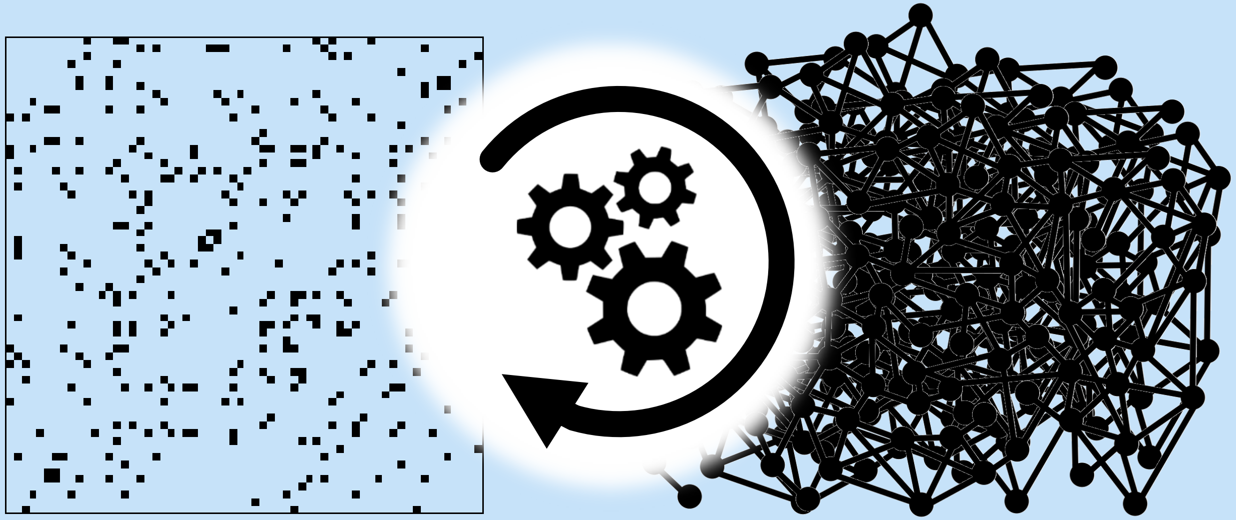
Massively parallel settings

My Research Overview

Sparse & irregular computing

...

- LLM ecosystems & agent infrastructures
- Generative AI training & inference
- Representation Learning & Graph Neural Networks
- Graph databases



- Graph compression & summarization
- Graph pattern matching
- [Parallel] graph algorithms
- Streaming, dynamic, temporal graphs

Effective

Fast

Scalable

Simple

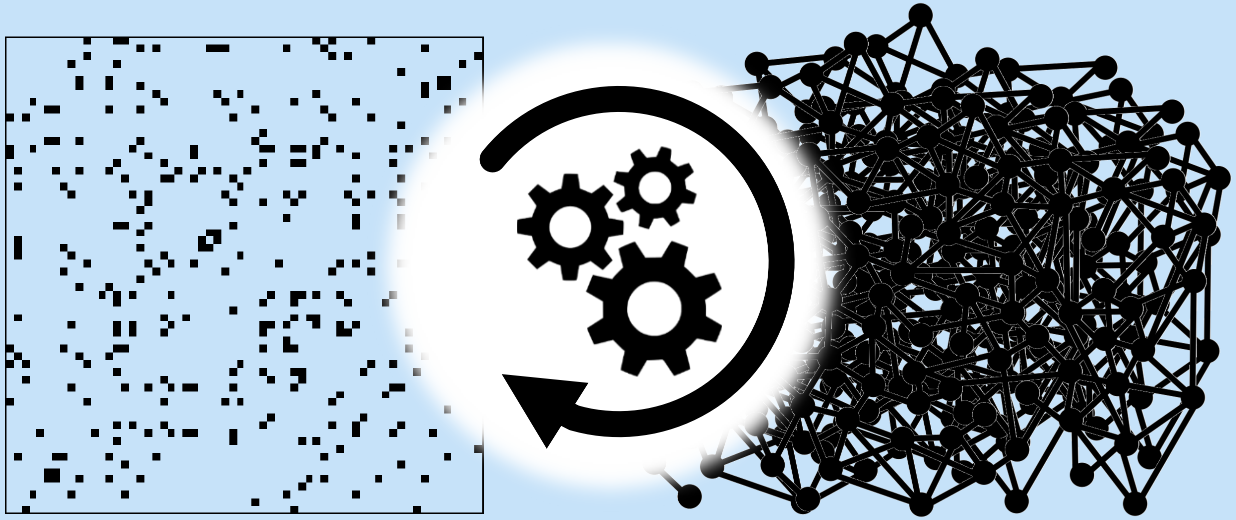
Massively parallel settings

My Research Overview

Sparse & irregular computing

...

- LLM ecosystems & agent infrastructures
- Generative AI training & inference
- Representation Learning & Graph Neural Networks
- Graph databases



- Graph compression & summarization
- Graph pattern matching
- [Parallel] graph algorithms
- Streaming, dynamic, temporal graphs

Effective

Getting to foundations

Massively parallel settings

Fast

Scalable

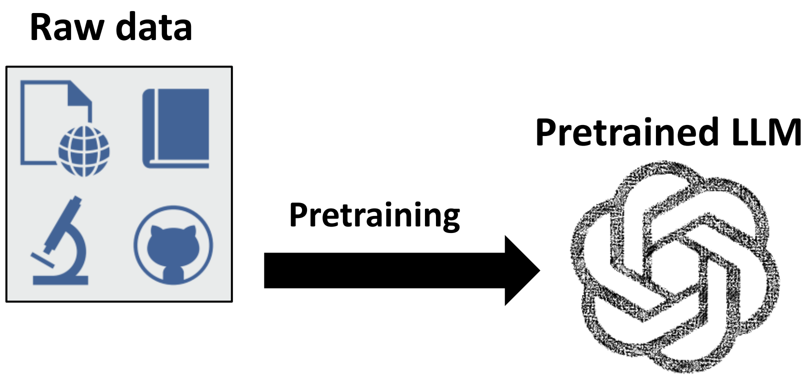
Simple

Overview of the LLM Processing Pipeline

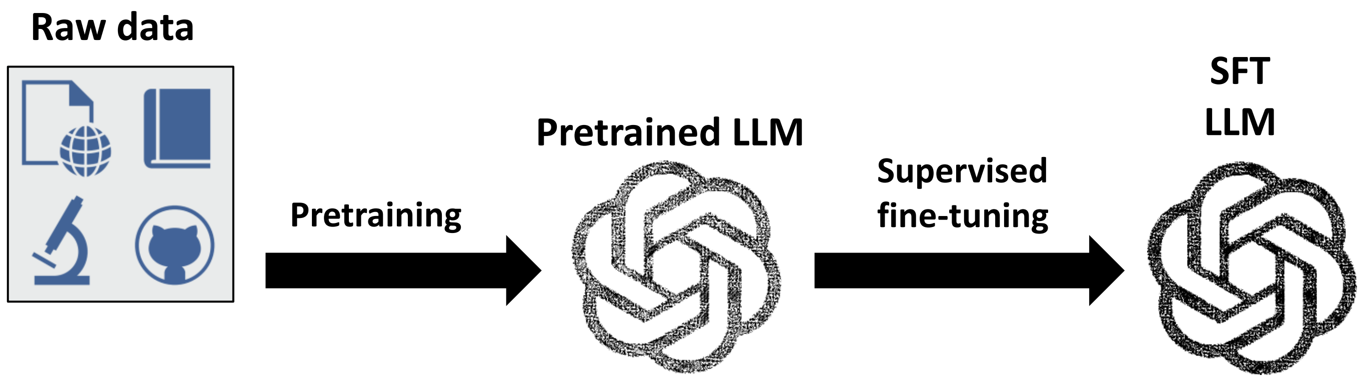
Overview of the LLM Processing Pipeline

Raw data

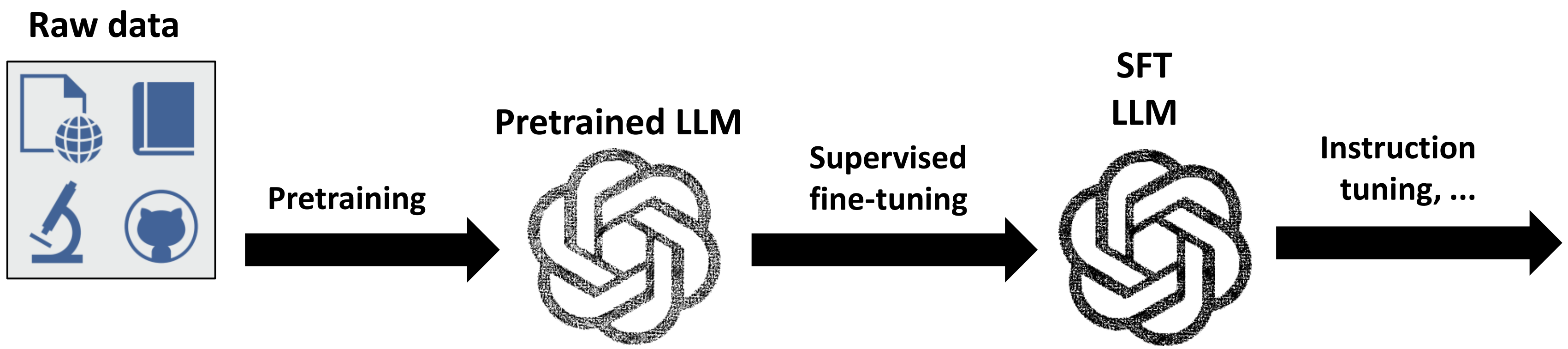
Overview of the LLM Processing Pipeline



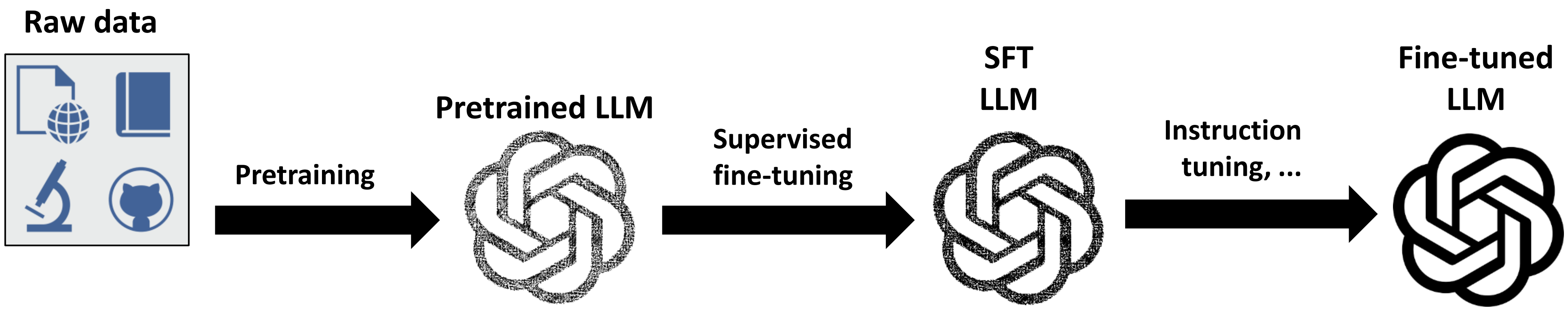
Overview of the LLM Processing Pipeline



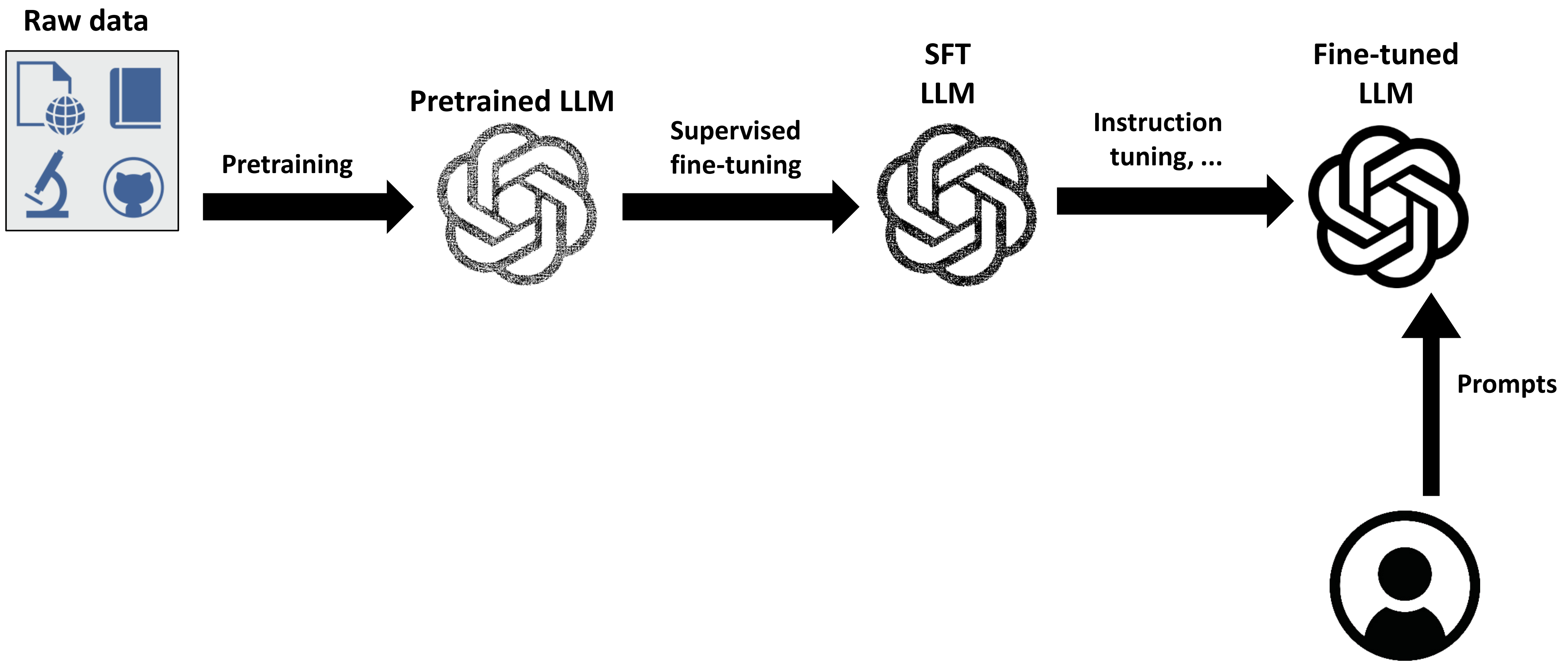
Overview of the LLM Processing Pipeline



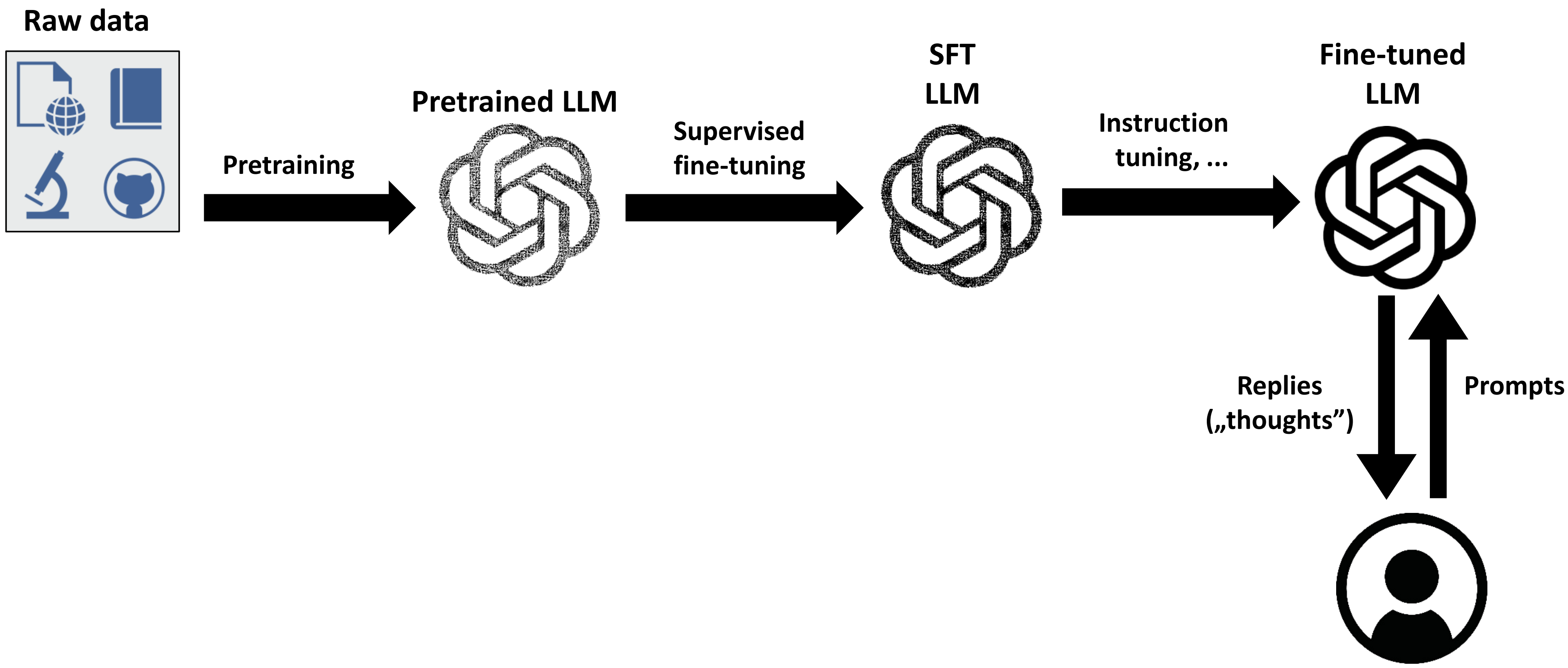
Overview of the LLM Processing Pipeline



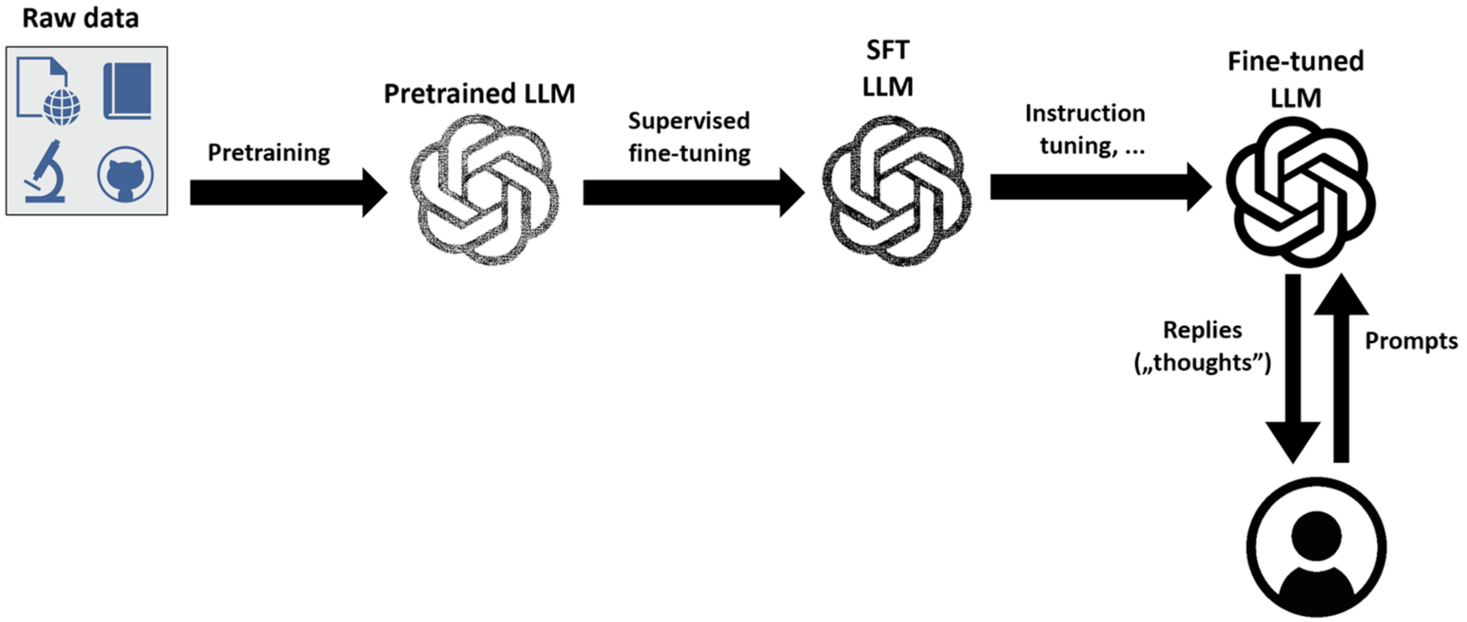
Overview of the LLM Processing Pipeline



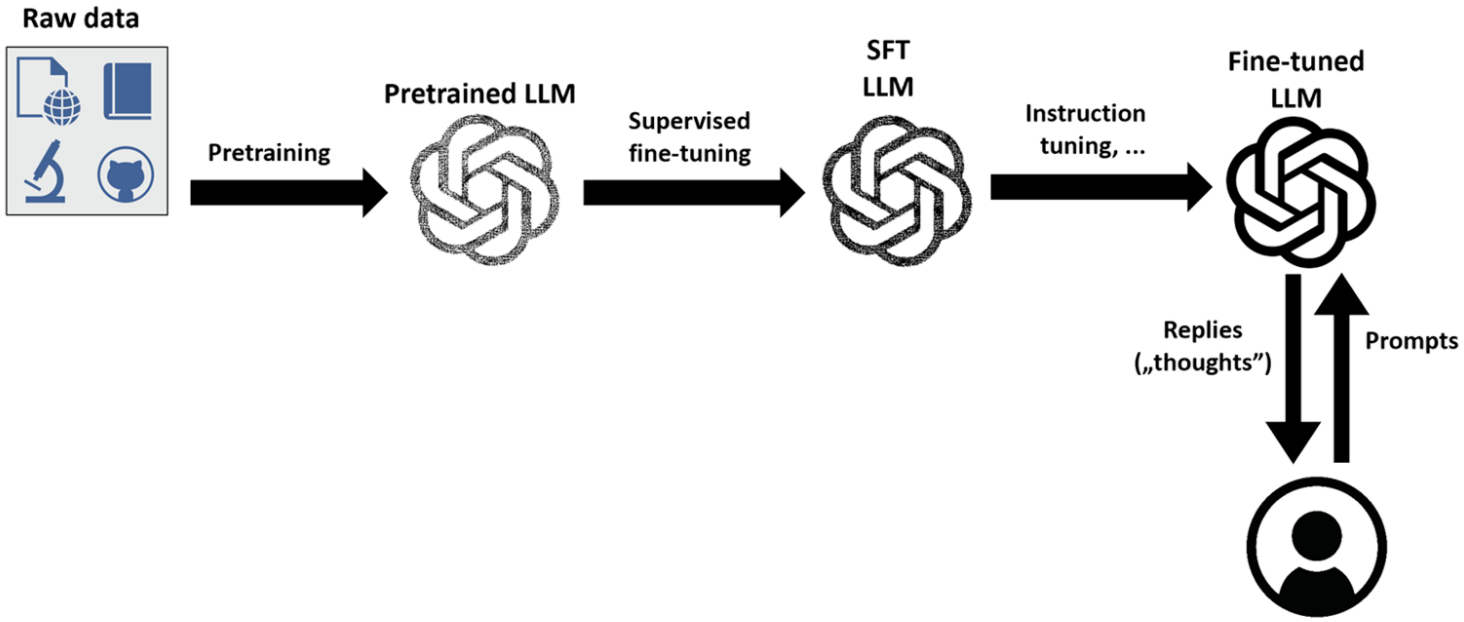
Overview of the LLM Processing Pipeline



The Emergence of the „Generative AI Ecosystem"



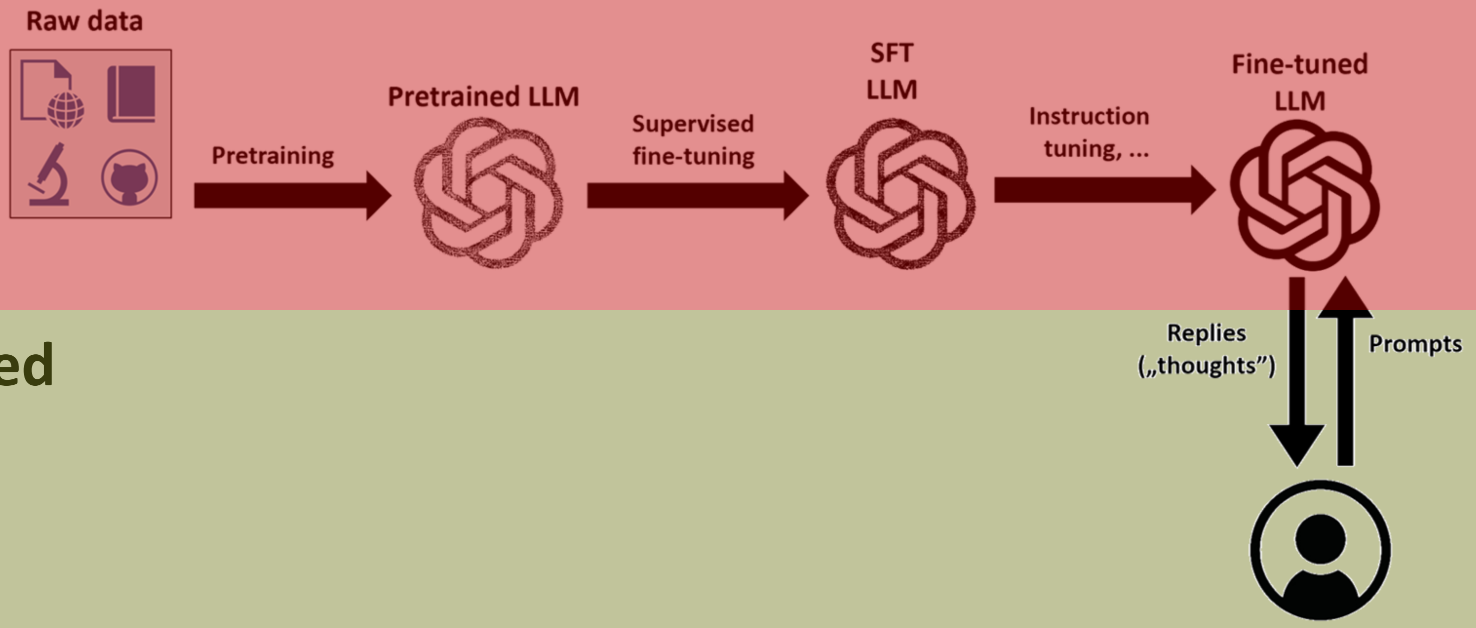
The „Generative AI Ecosystem”



The „Generative AI Ecosystem"

Training related

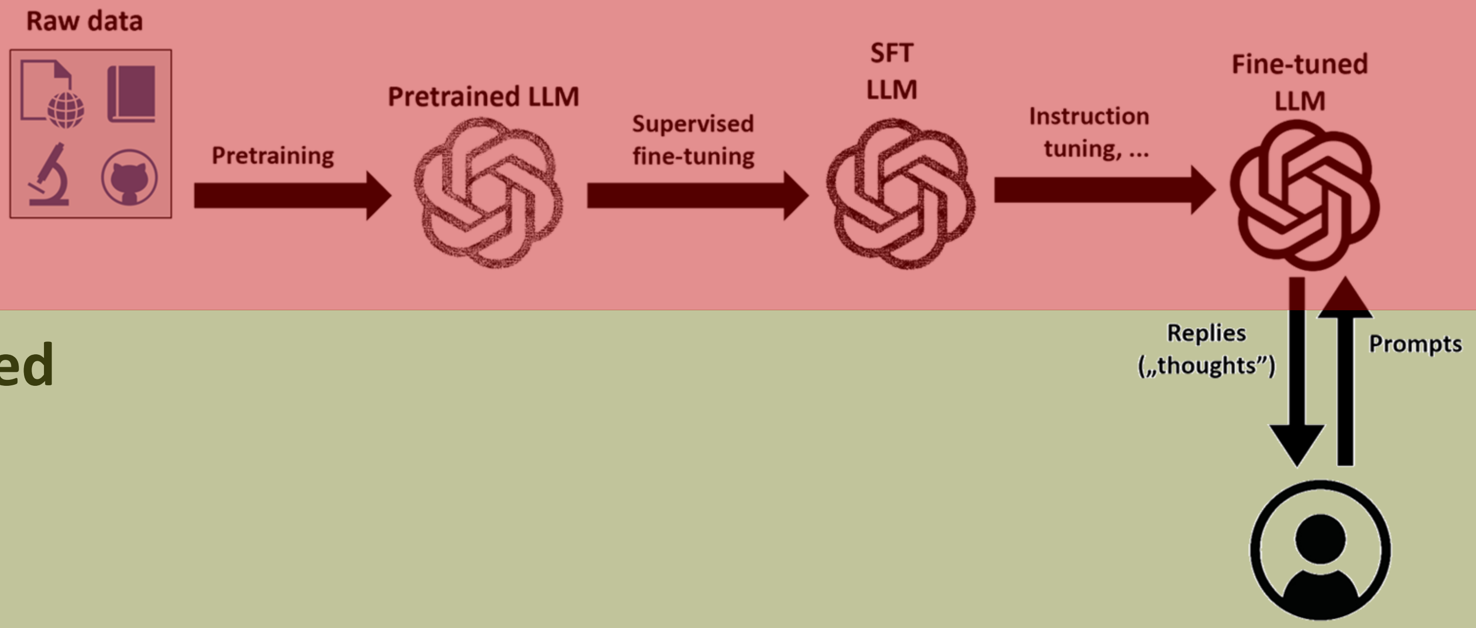
Inference related



The „Generative AI Ecosystem"

Training related

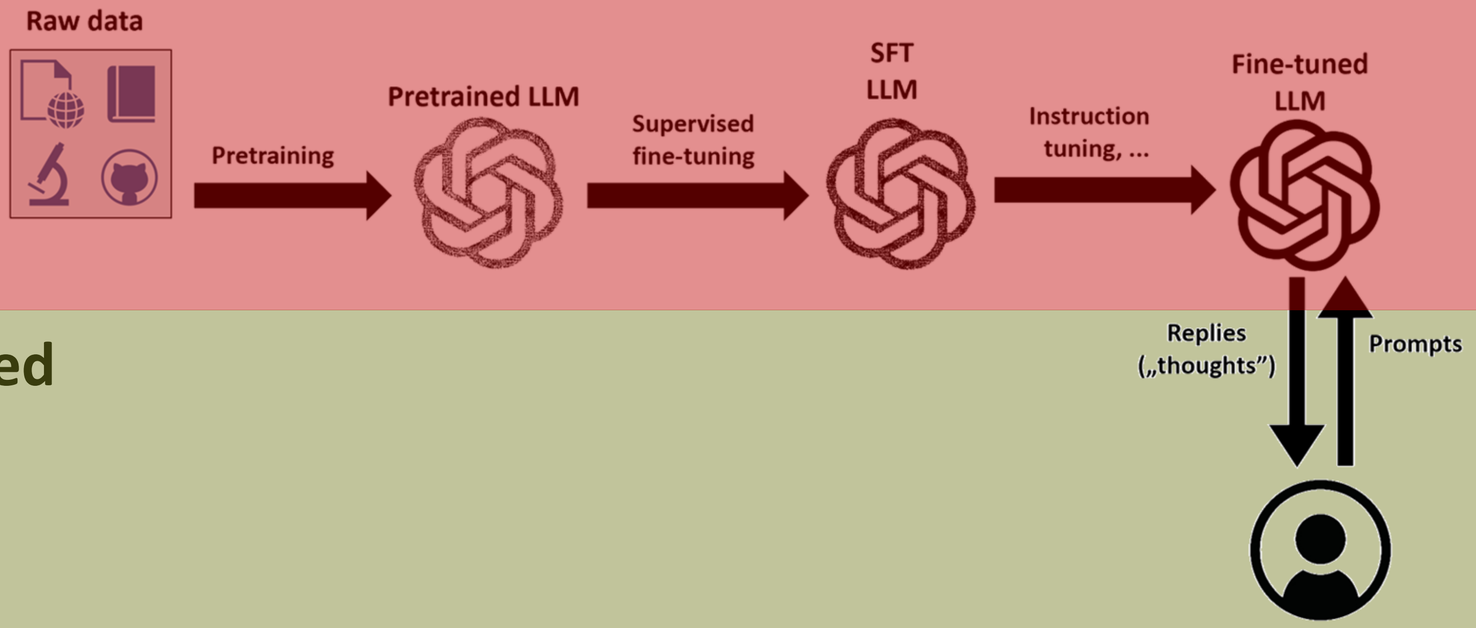
Inference related



Prompt engineering

The „Generative AI Ecosystem"

Training related



Inference related

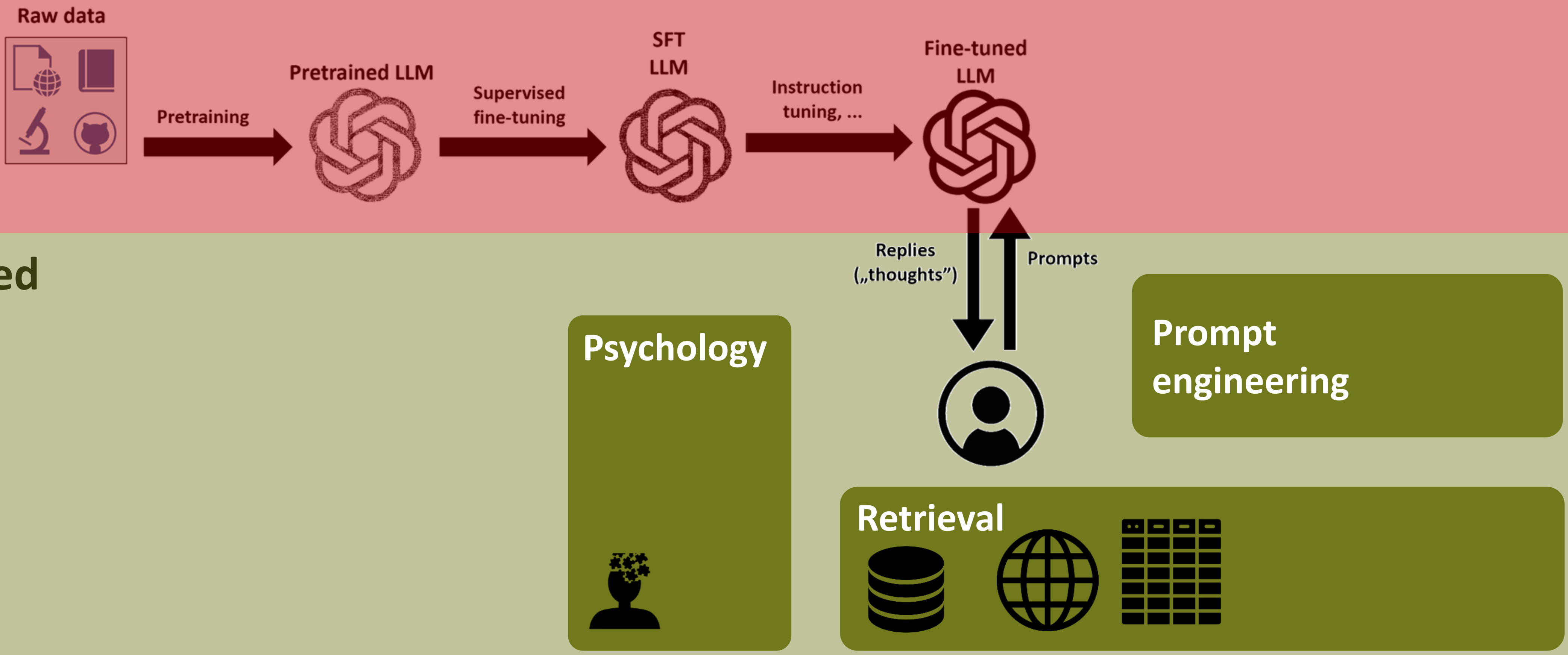
Prompt engineering

Retrieval

The „Generative AI Ecosystem"

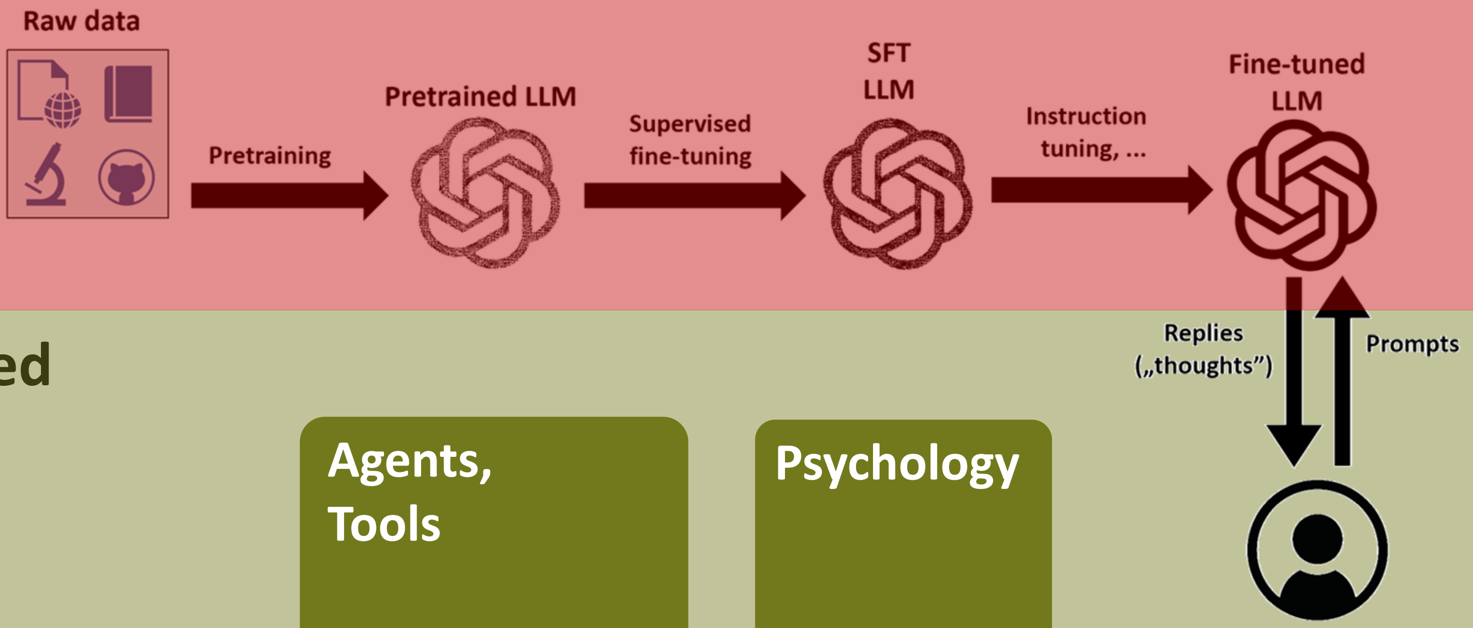
Training related

Inference related



The „Generative AI Ecosystem"

Training related



Inference related

Agents, Tools

Psychology

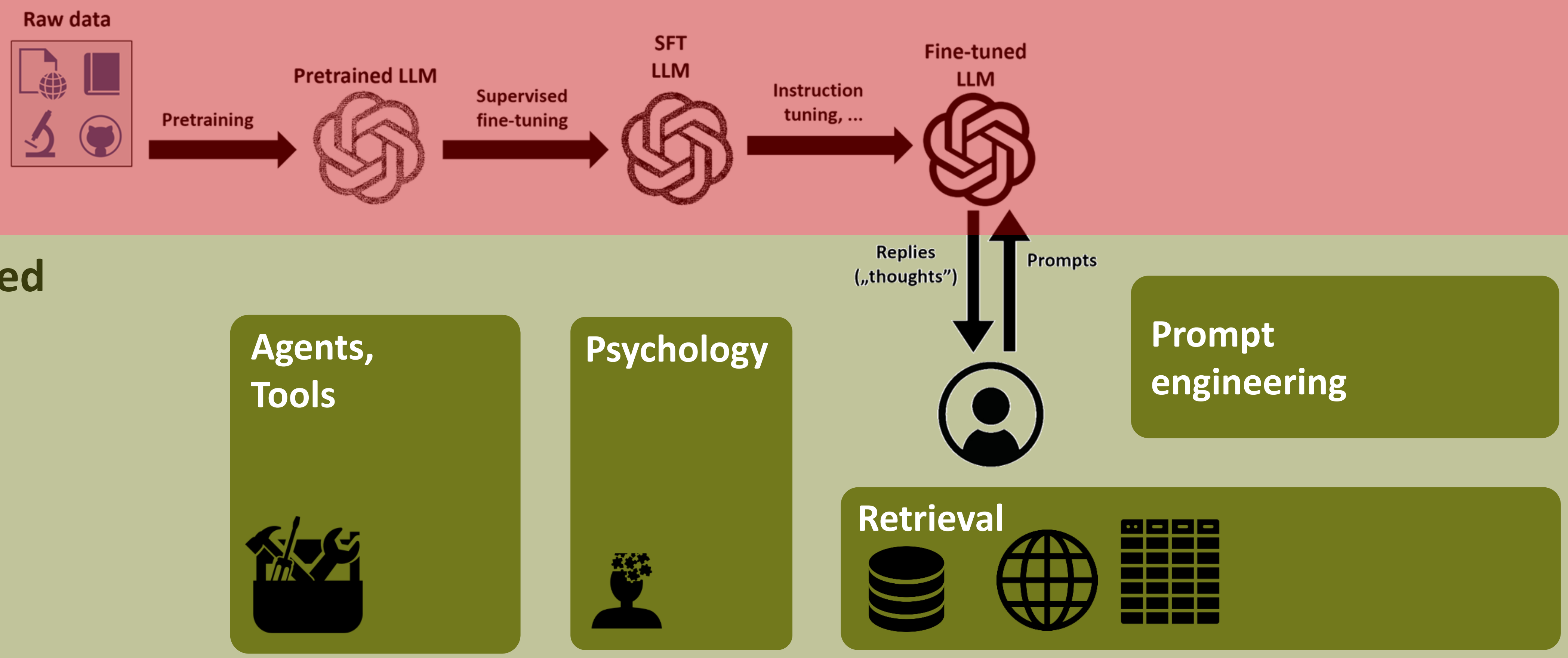
Prompt engineering

Retrieval

Graphs in the LLM Pipeline: Overview

Training related

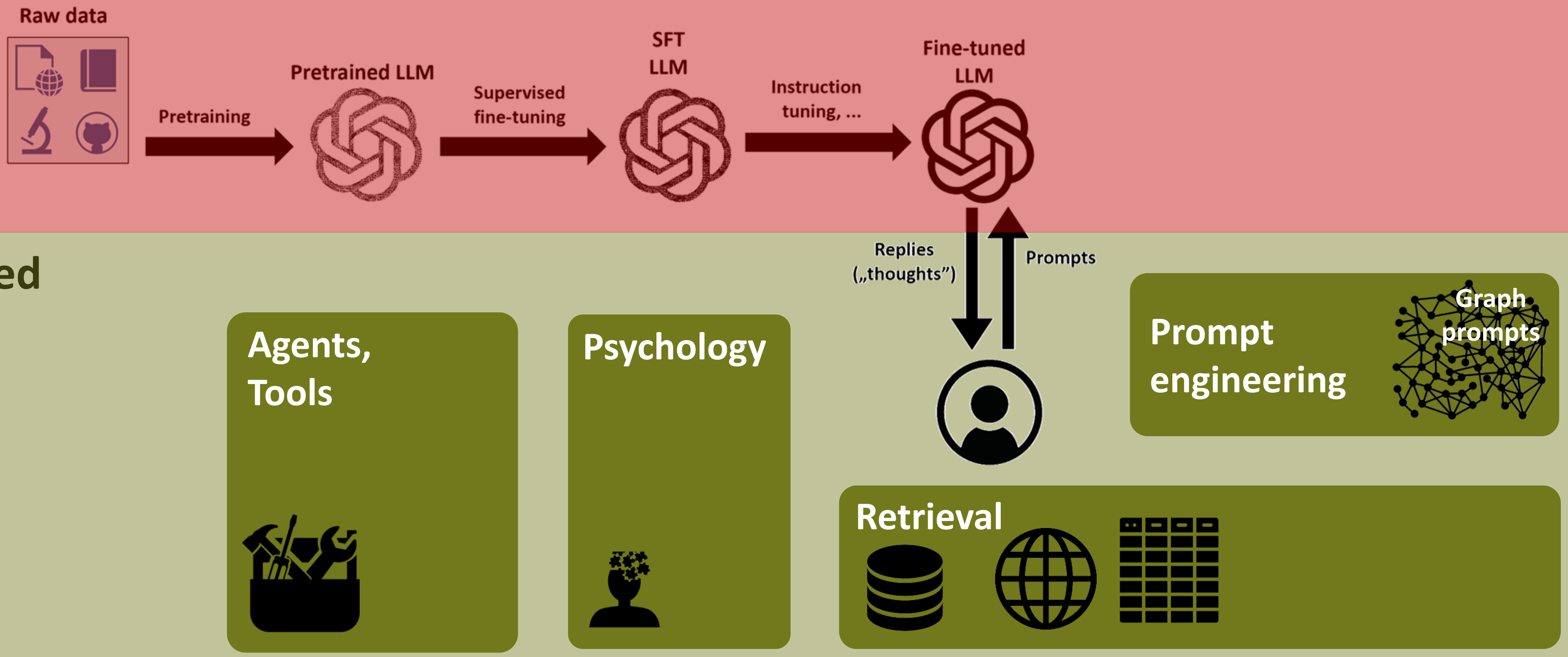
Inference related



Graphs in the LLM Pipeline: Overview

Training related

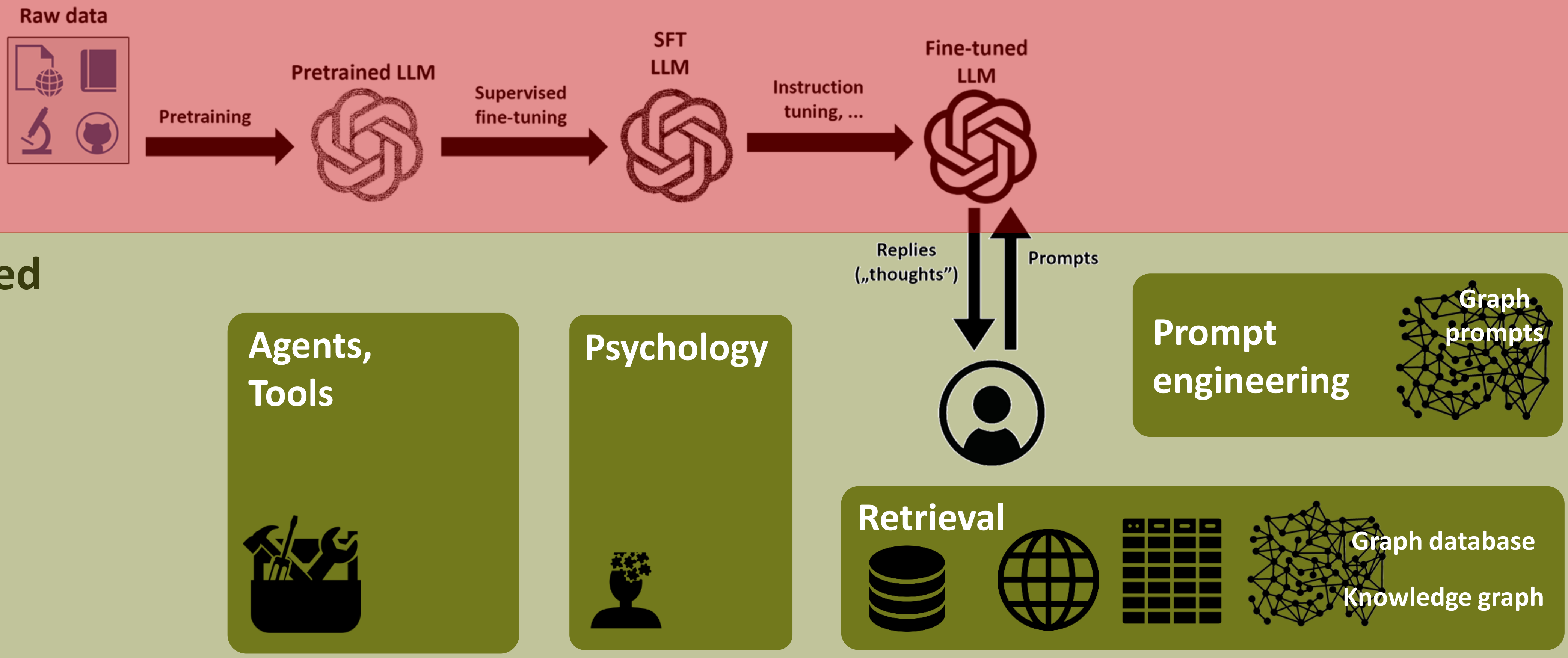
Inference related



Graphs in the LLM Pipeline: Overview

Training related

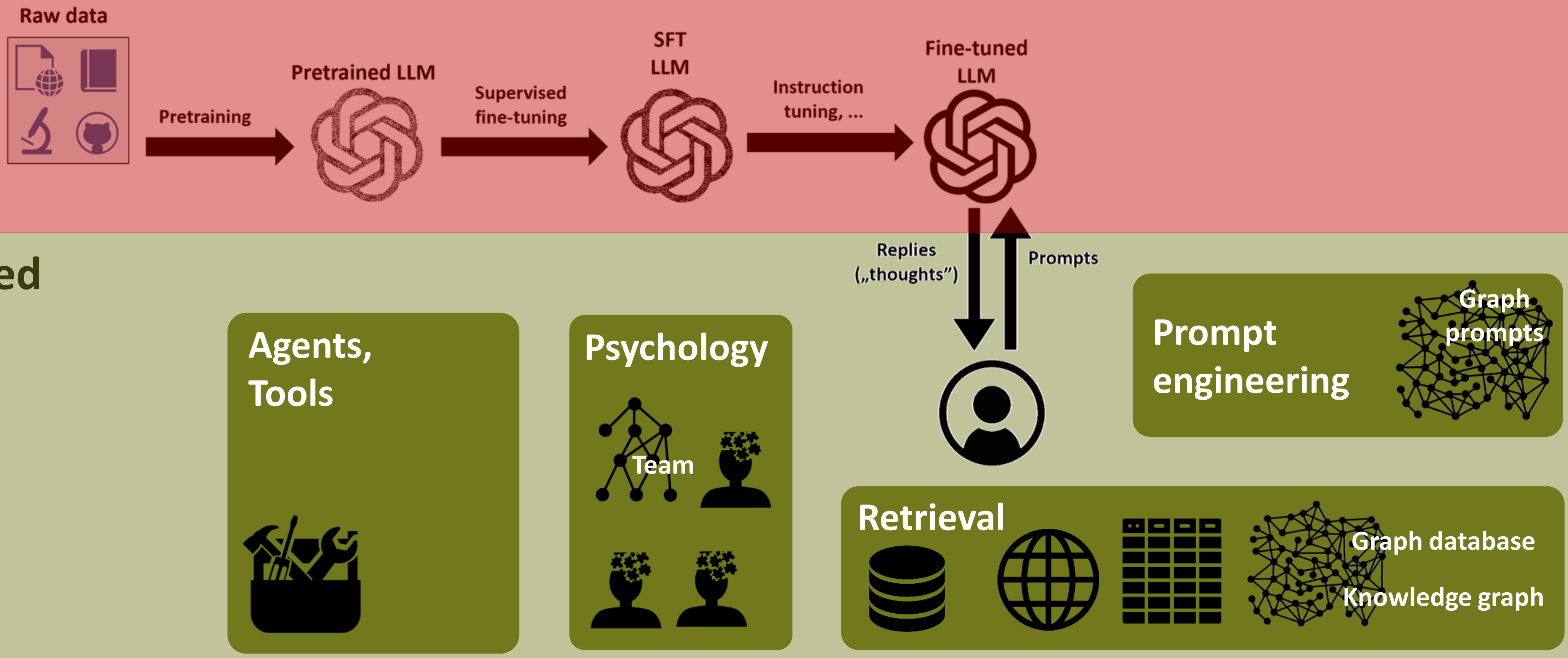
Inference related



Graphs in the LLM Pipeline: Overview

Training related

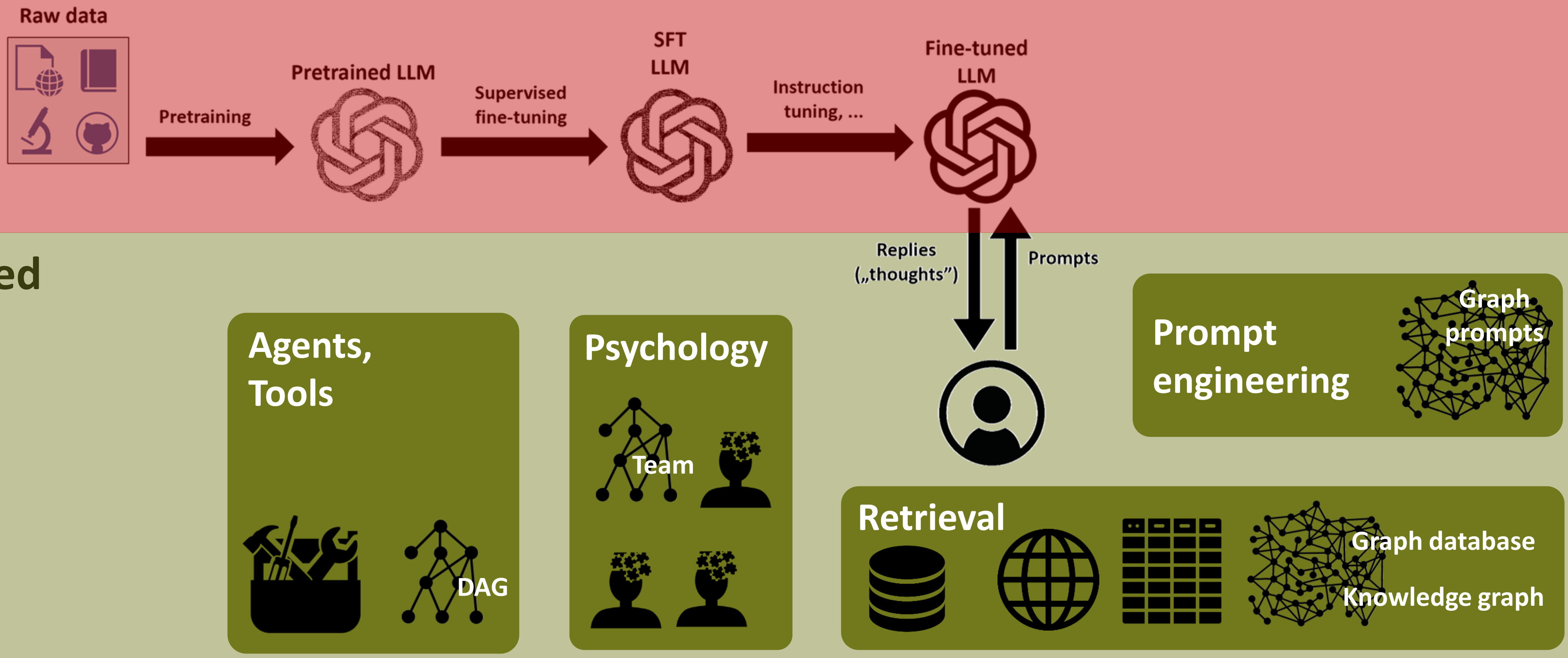
Inference related



Graphs in the LLM Pipeline: Overview

Training related

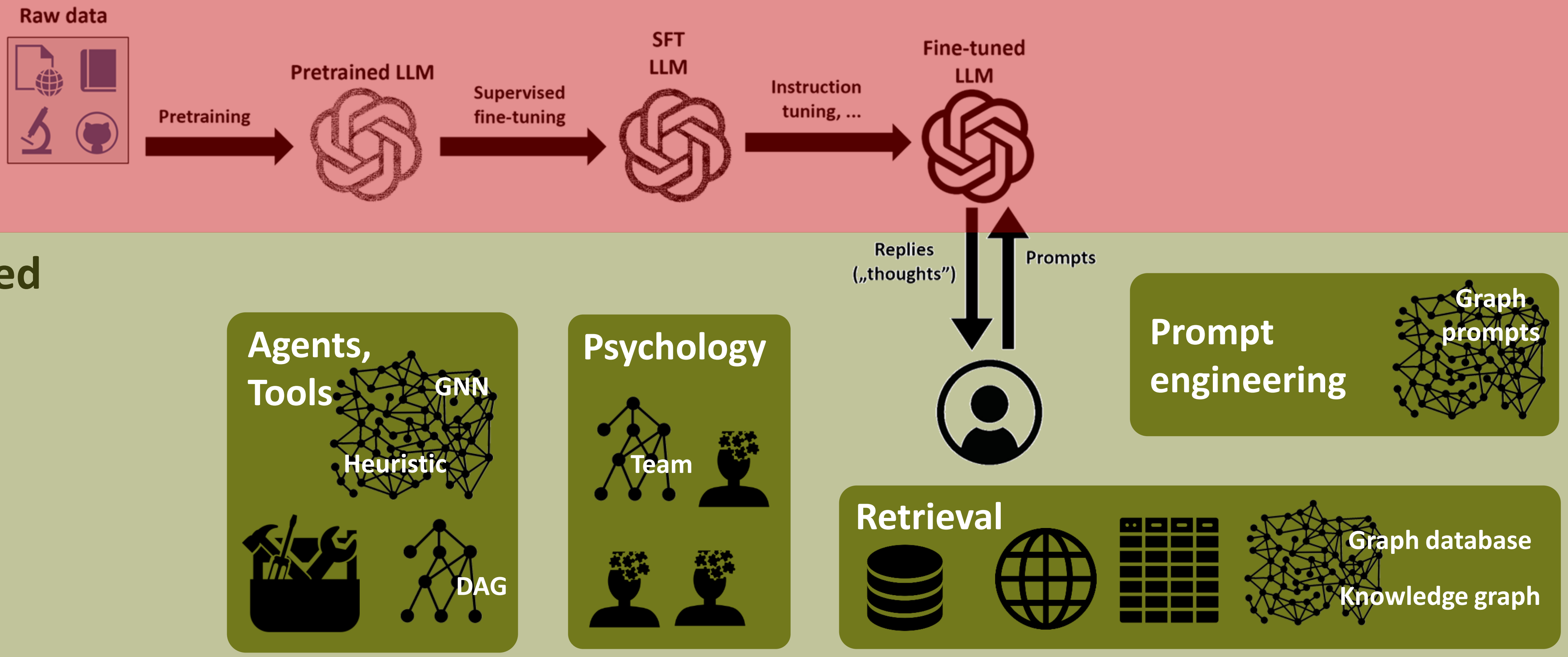
Inference related



Graphs in the LLM Pipeline: Overview

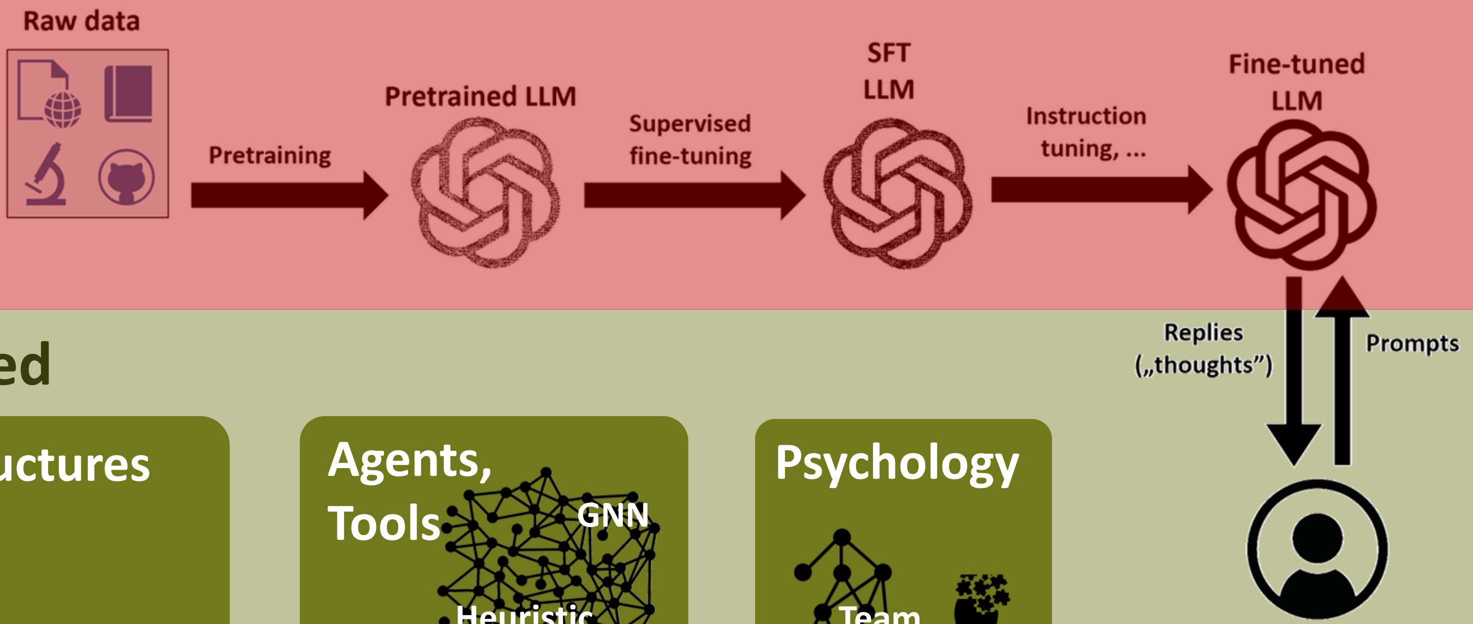
Training related

Inference related



Graphs in the LLM Pipeline: Overview

Training related



Inference related

Reasoning Structures

Agents, Tools
GNN
Heuristic
DAG

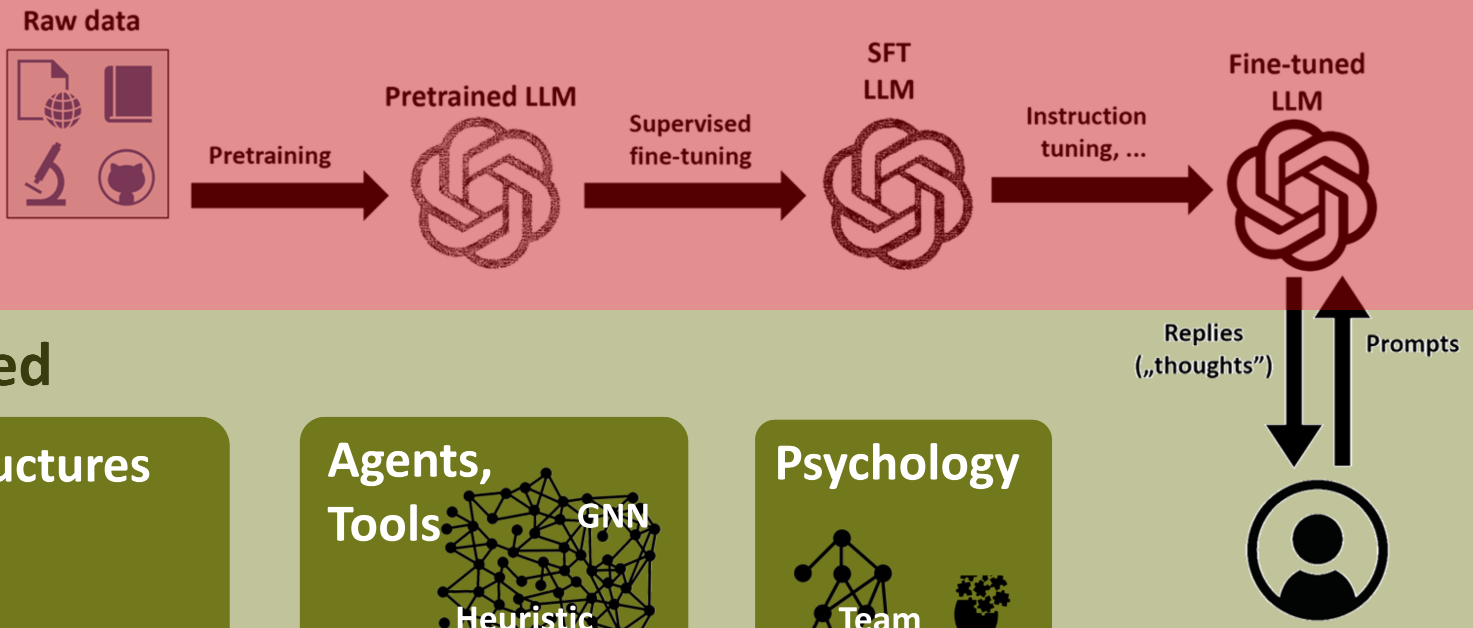
Psychology
Team

Prompt engineering
Graph prompts

Retrieval
Graph database
Knowledge graph

Graphs in the LLM Pipeline: Overview

Training related

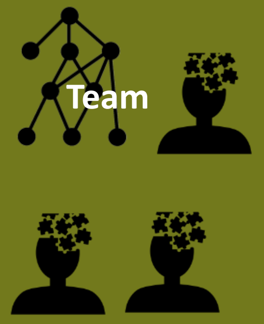


Inference related

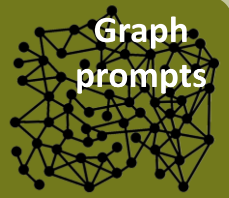
Reasoning Structures

Agents, Tools

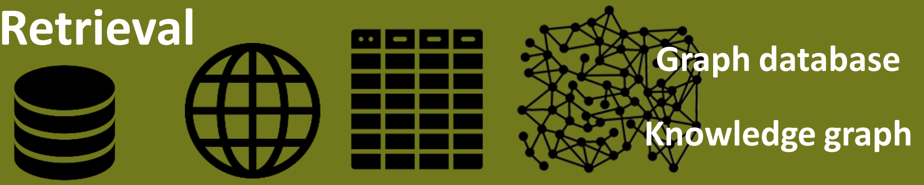
Psychology



Prompt engineering

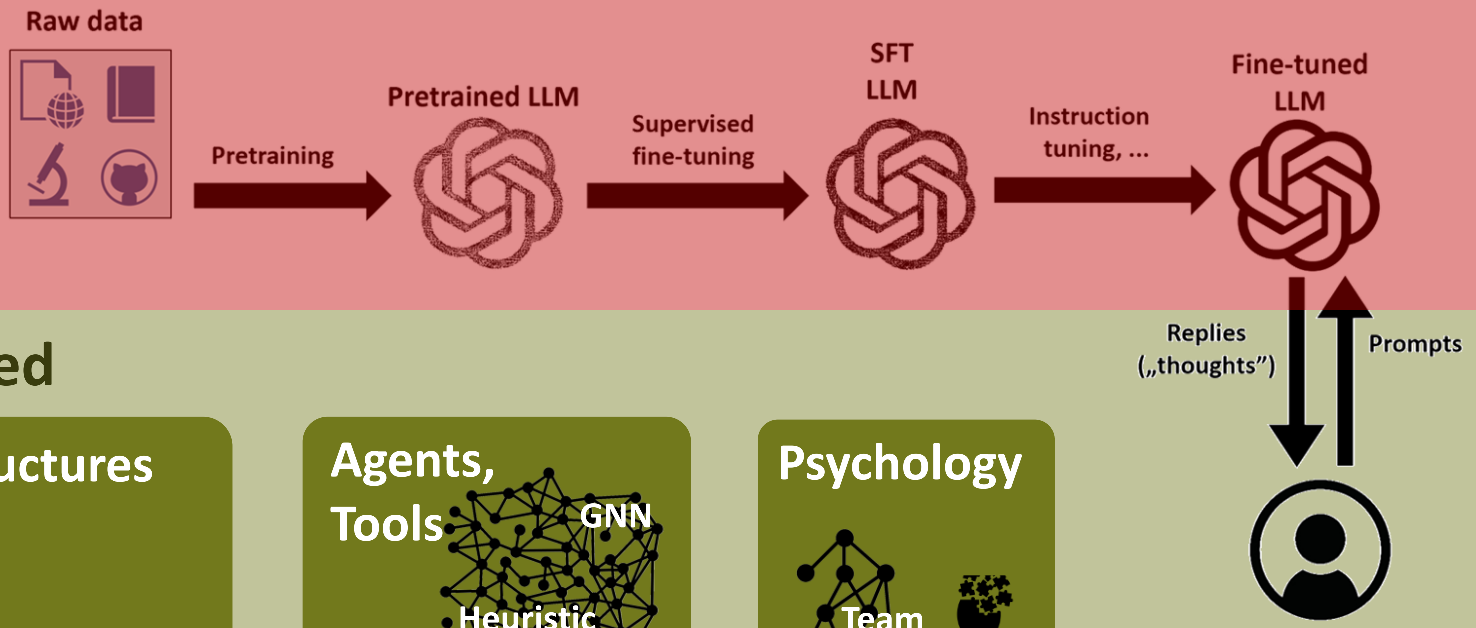


Retrieval



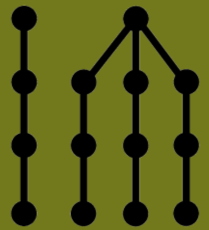
Graphs in the LLM Pipeline: Overview

Training related



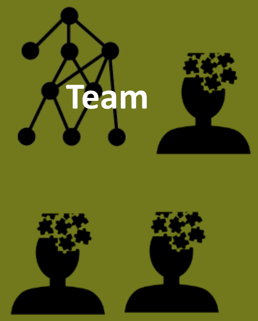
Inference related

Reasoning Structures

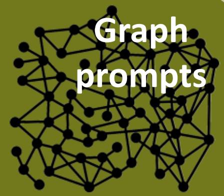


Agents, Tools

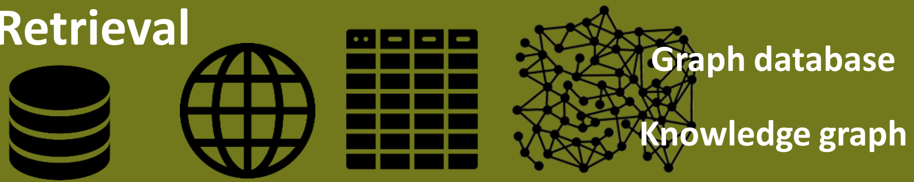
Psychology



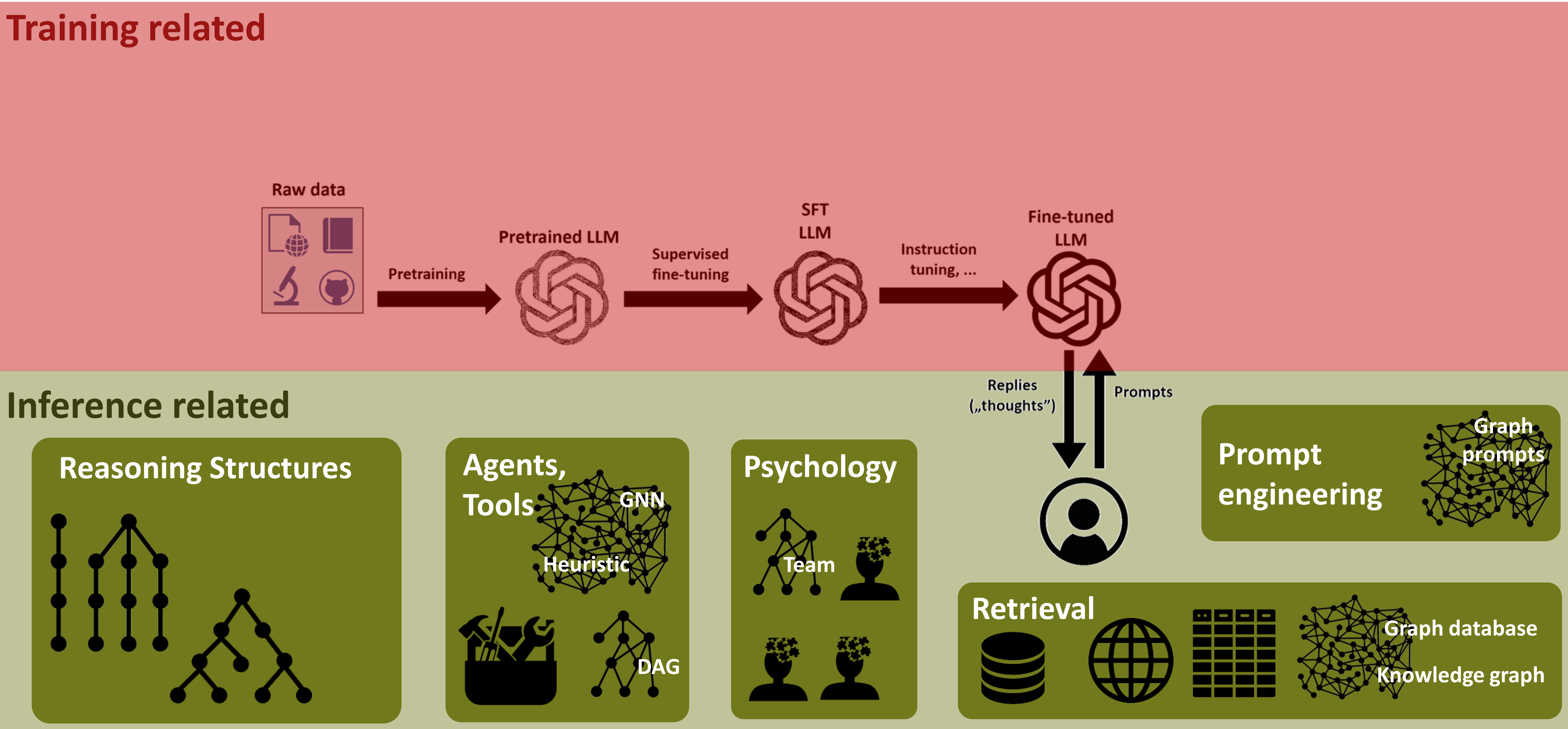
Prompt engineering



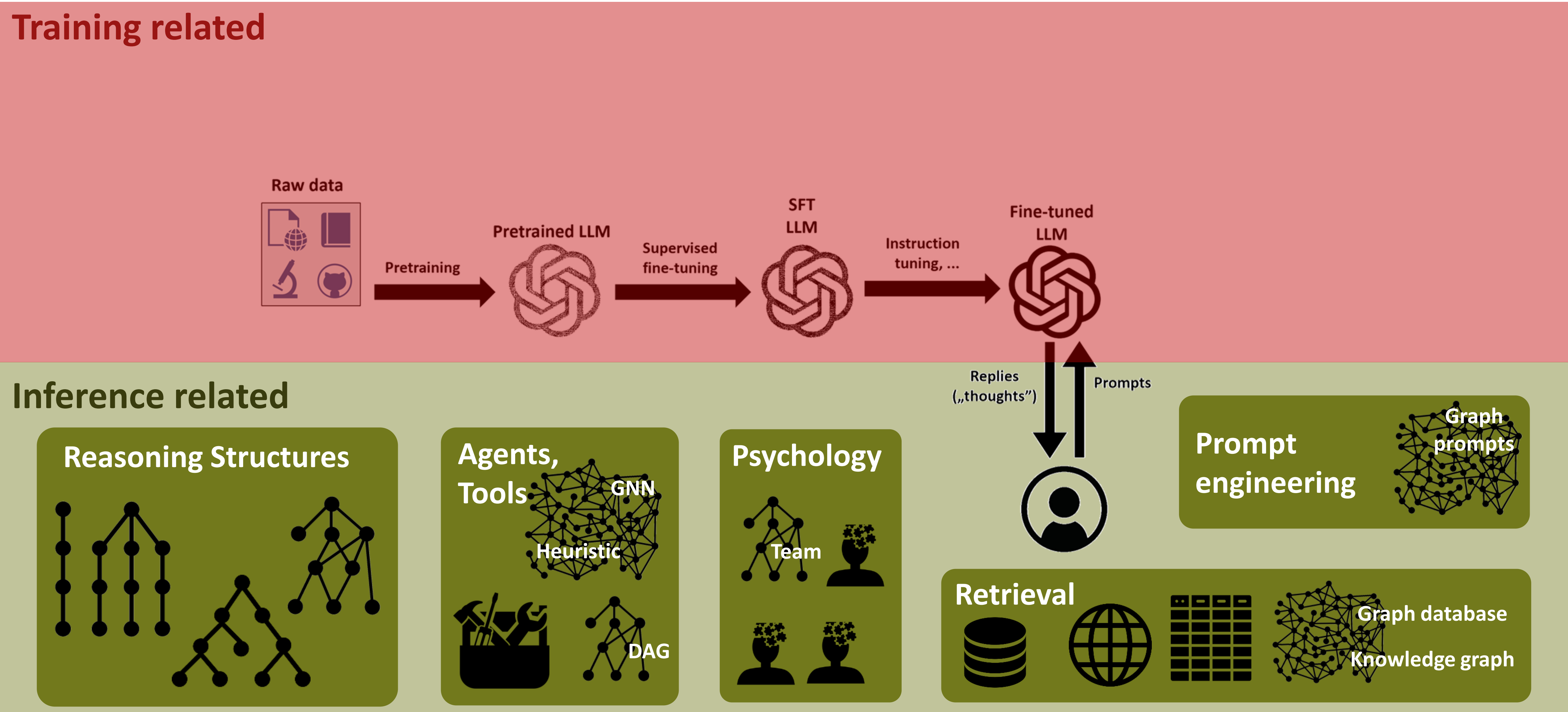
Retrieval



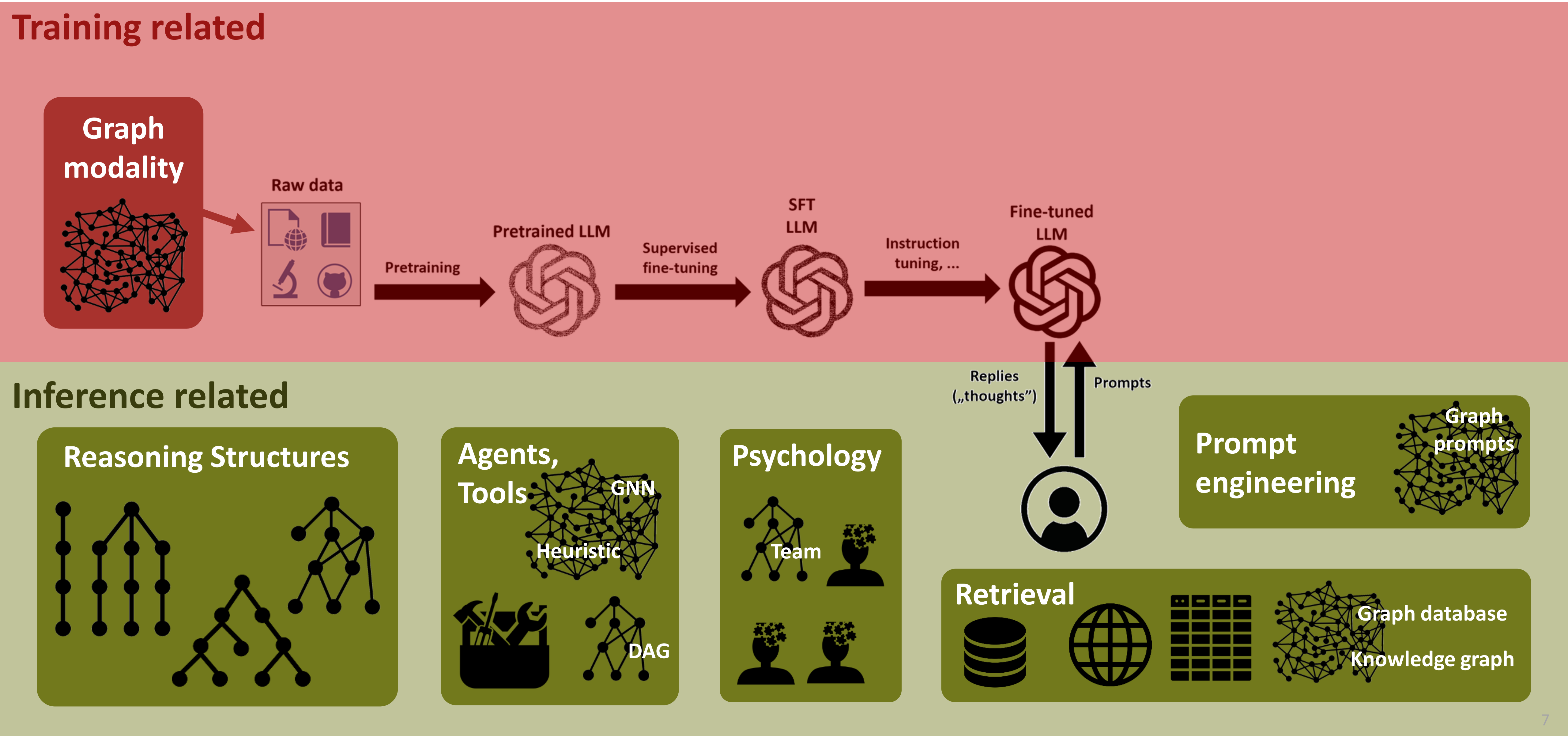
Graphs in the LLM Pipeline: Overview



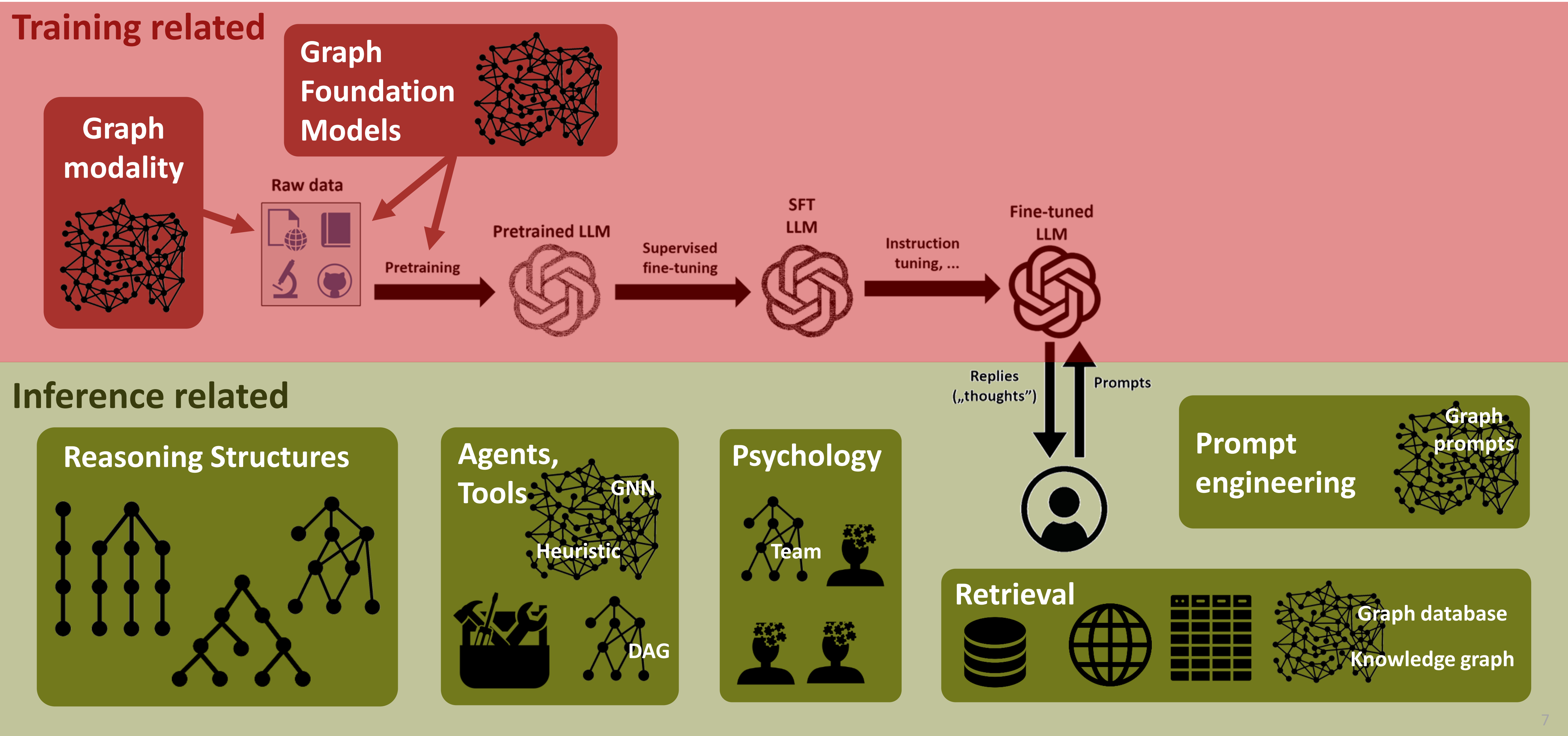
Graphs in the LLM Pipeline: Overview



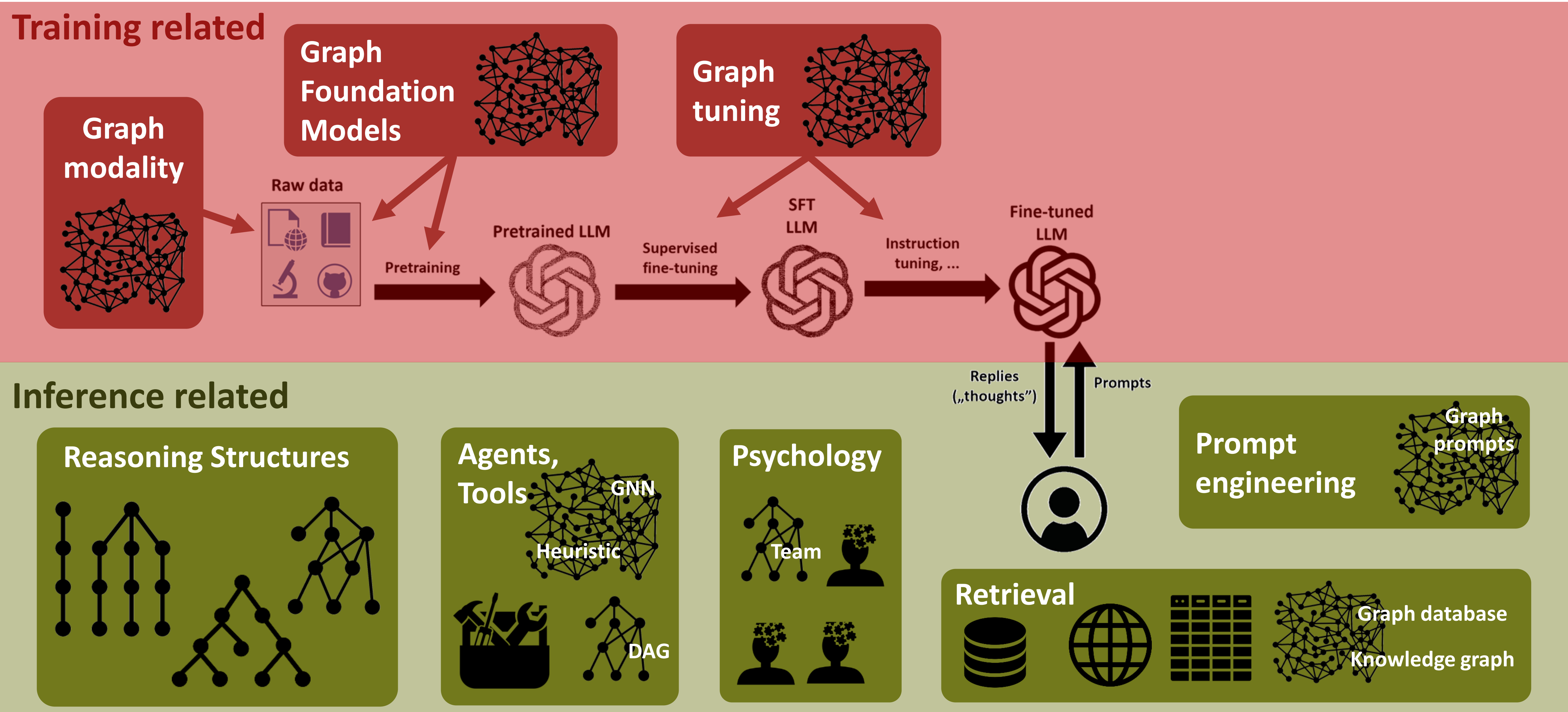
Graphs in the LLM Pipeline: Overview



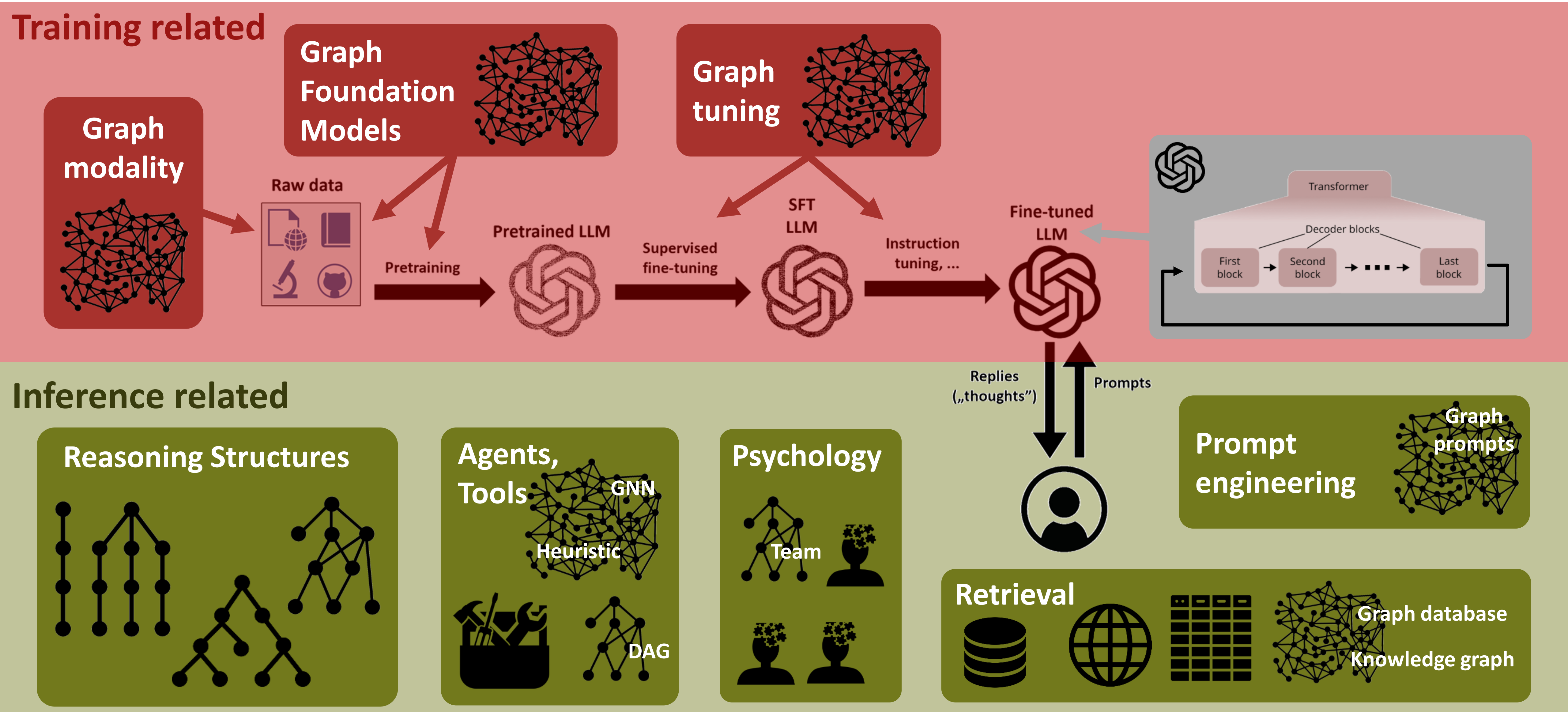
Graphs in the LLM Pipeline: Overview



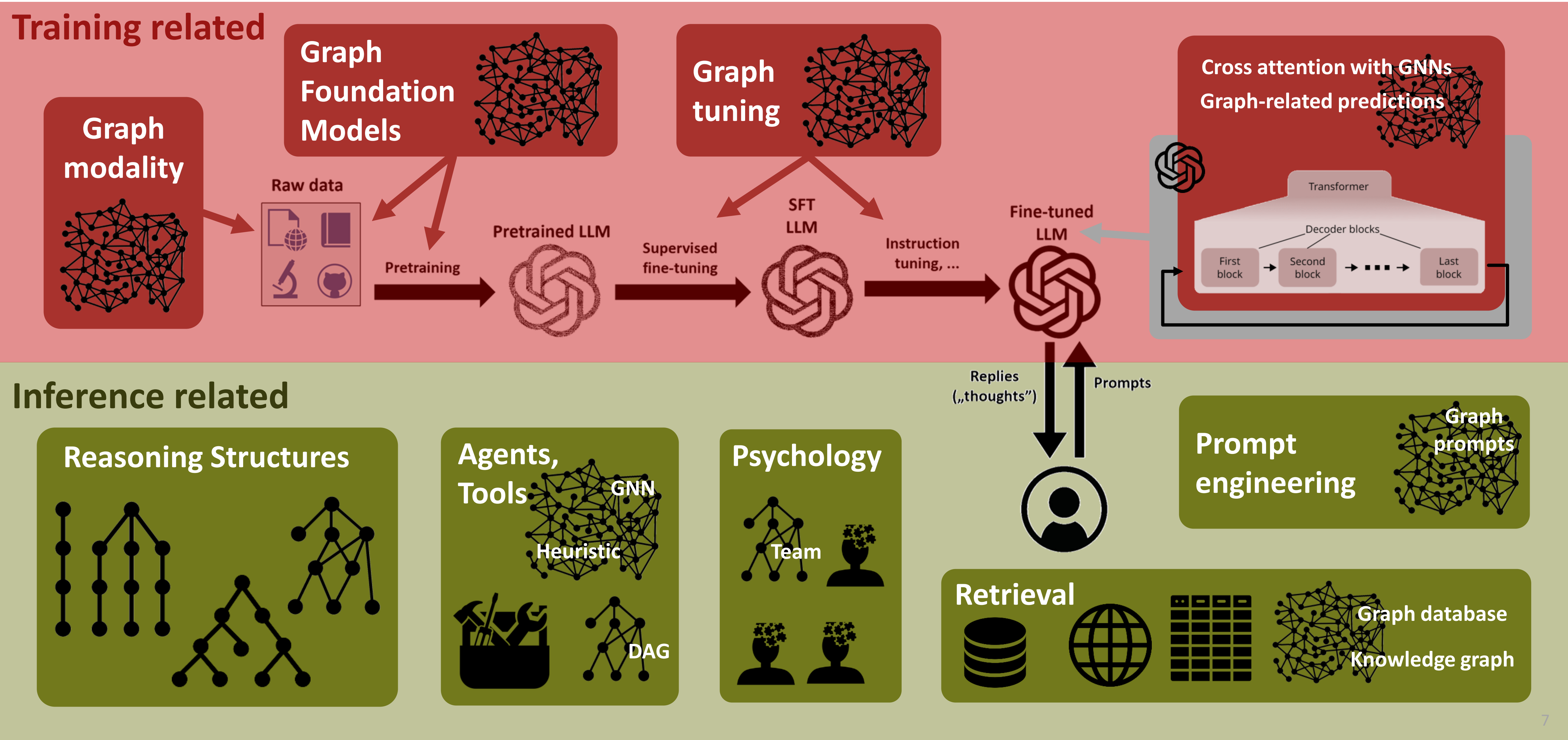
Graphs in the LLM Pipeline: Overview



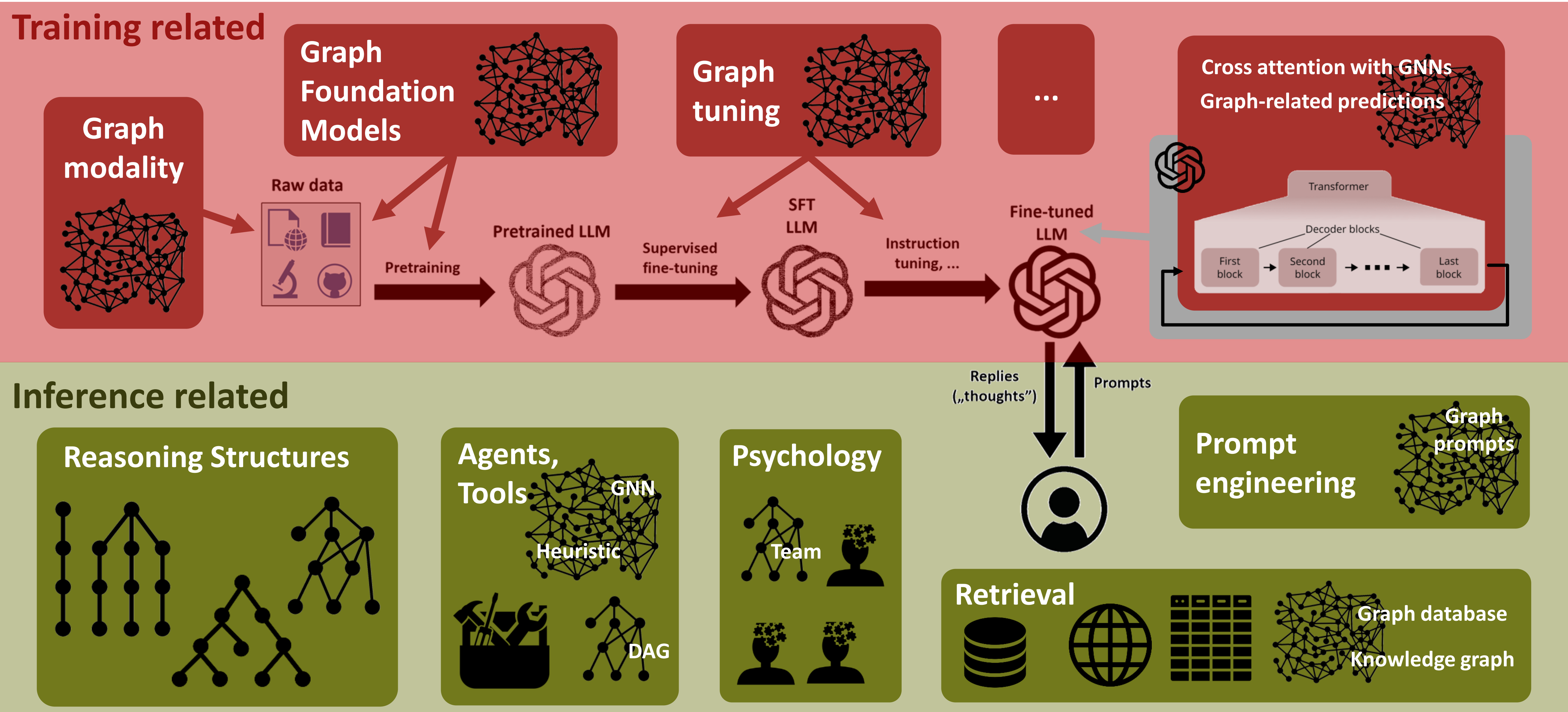
Graphs in the LLM Pipeline: Overview



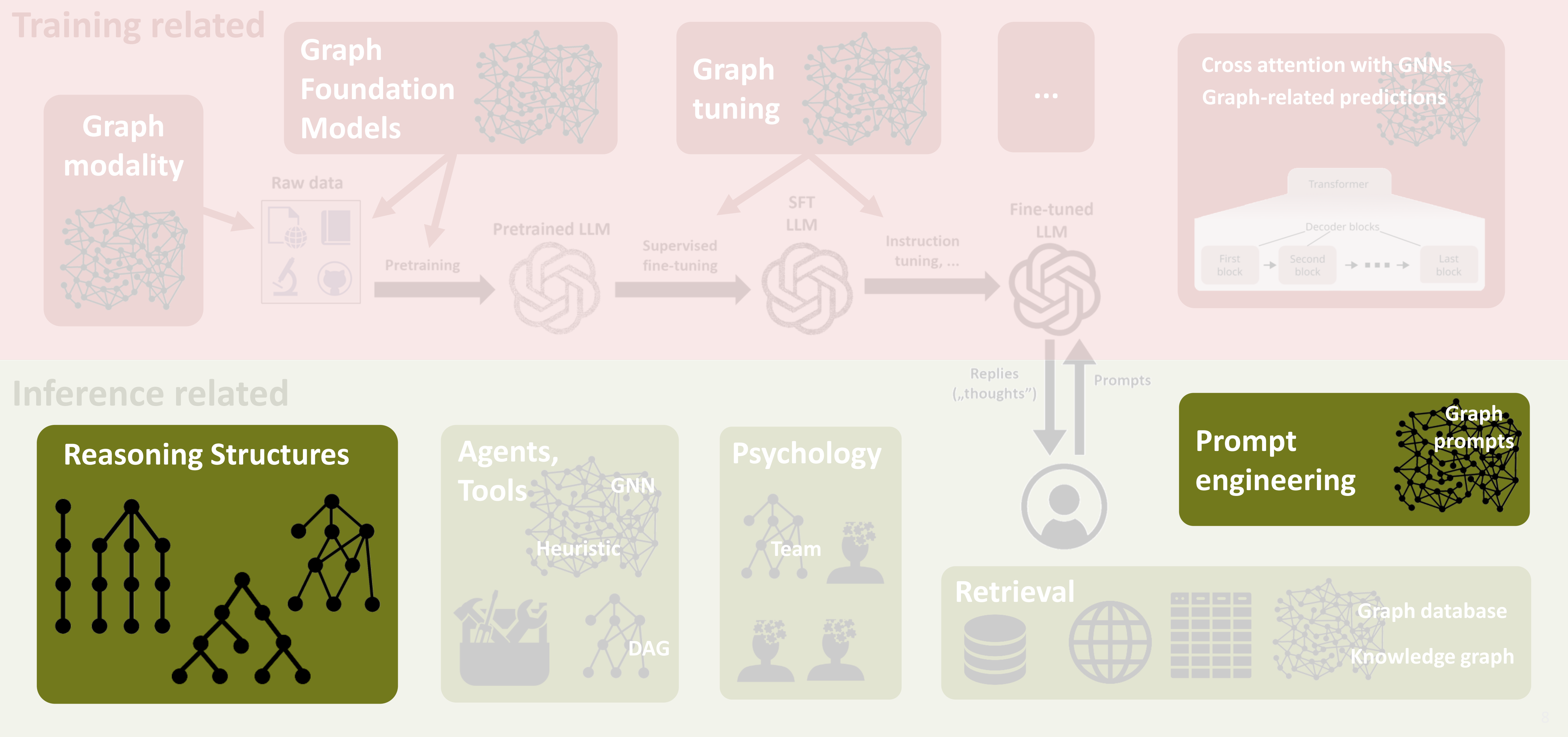
Graphs in the LLM Pipeline: Overview



Graphs in the LLM Pipeline: Overview



Graphs in the LLM Pipeline: Overview

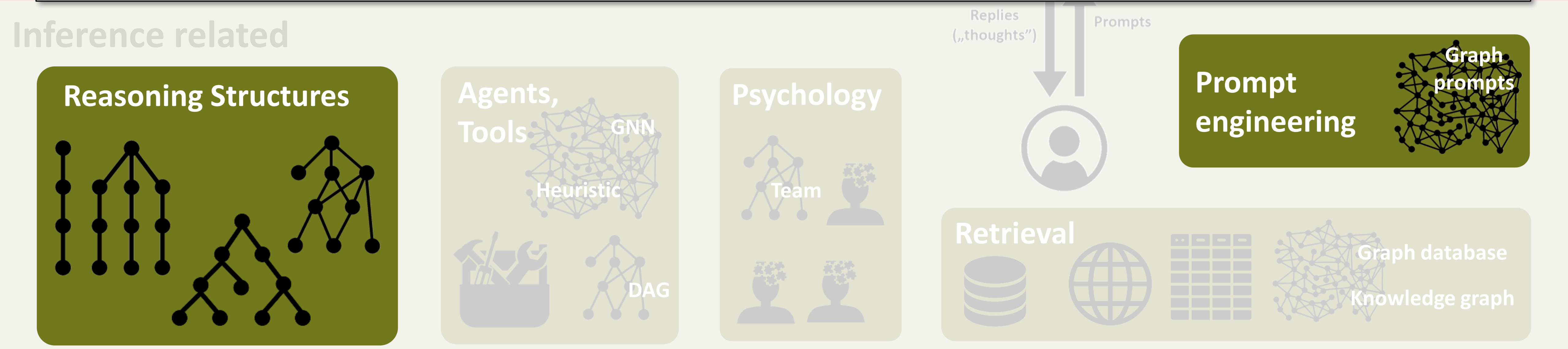


Graphs in the LLM Pipeline: Overview

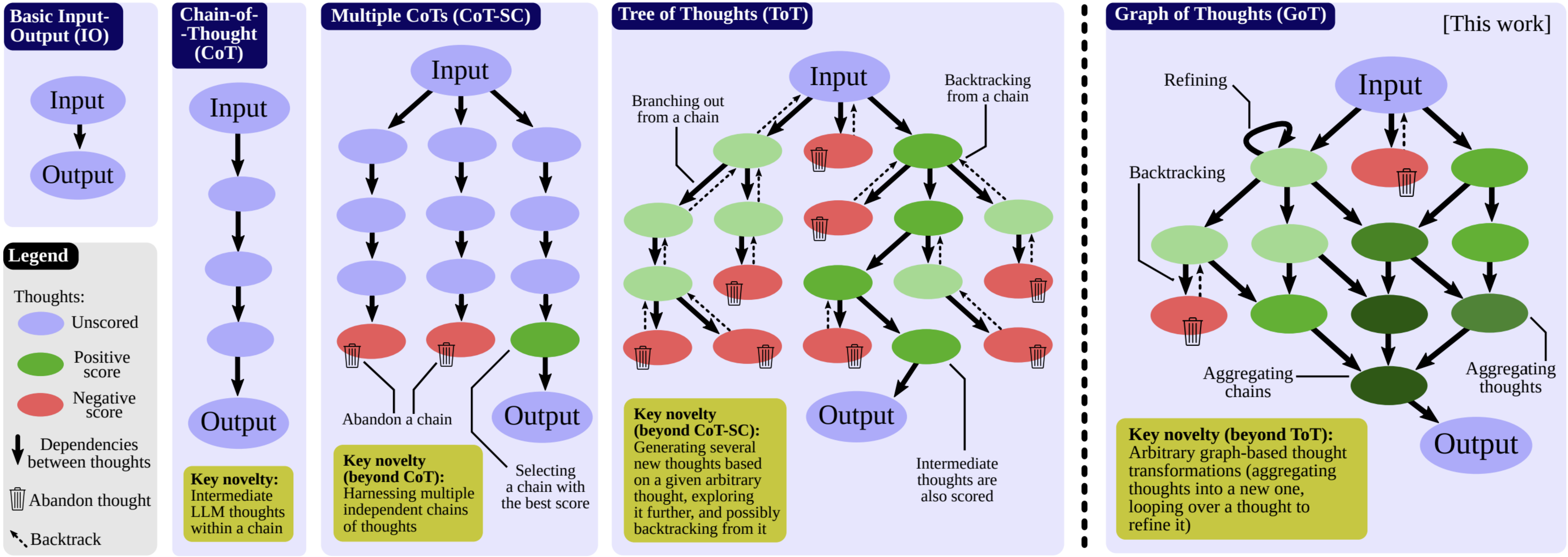
https://github.com/spcl/graph-of-thoughts, @AAAI'24

Graph of Thoughts: Solving Elaborate Problems with Large Language Models

Maciej Besta[1*], Nils Blach[1*], Ales Kubicek[1], Robert Gerstenberger[1], Lukas Gianinazzi[1], Joanna Gajda[2], Tomasz Lehmann[2], Michał Podstawski[3], Hubert Niewiadomski[2], Piotr Nyczyk[2], Torsten Hoefler[1]
[1]ETH Zurich, [2]Cledar, [3]Warsaw University of Technology
bestam@inf.ethz.ch, nils.blach@inf.ethz.ch, htor@inf.ethz.ch

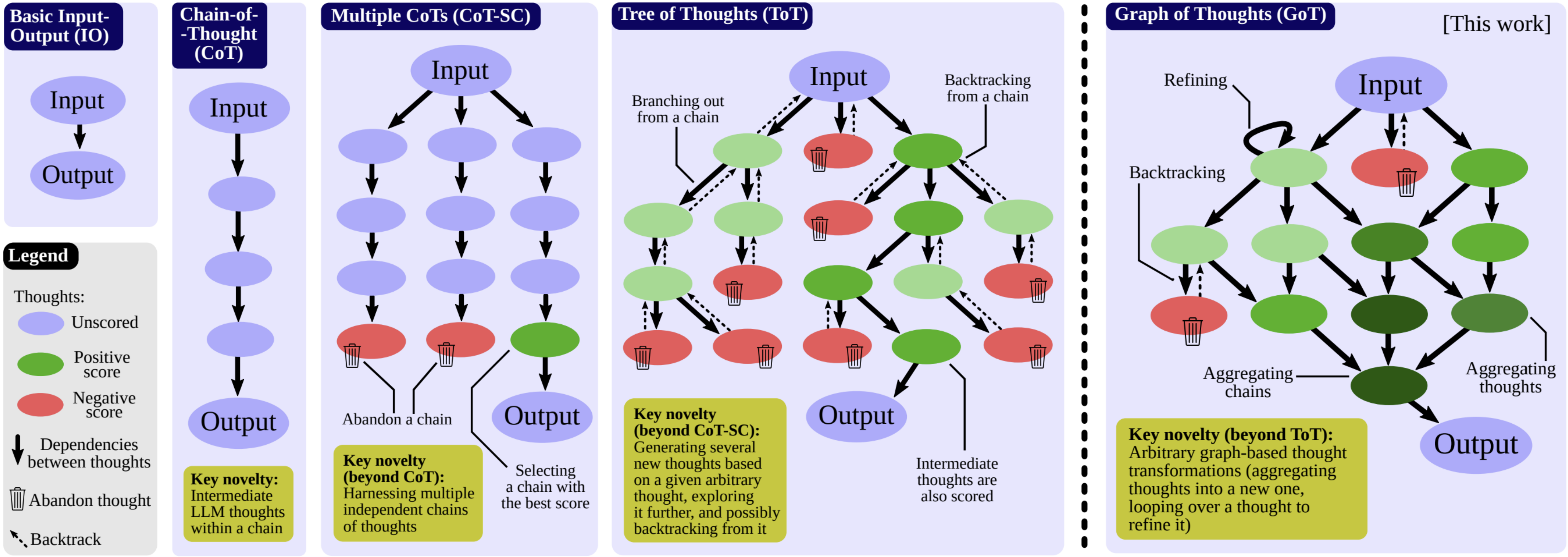


Reasoning Paradigms



Reasoning Paradigms

Graph as a model of the reasoning process



Graphs in the LLM Pipeline: Overview

Training related

https://github.com/spcl/graph-of-thoughts, @AAAI'24

Graph of Thoughts: Solving Elaborate Problems with Large Language Models

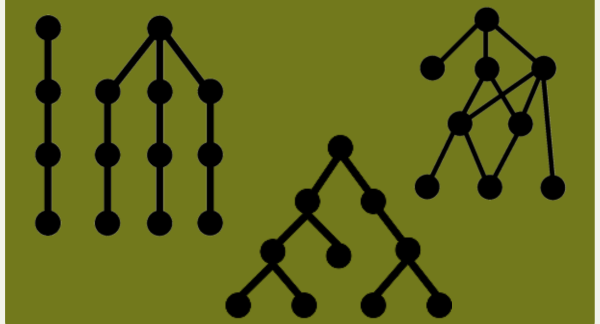
Maciej Besta[1*], Nils Blach[1*], Ales Kubicek[1], Robert Gerstenberger[1],
Lukas Gianinazzi[1], Joanna Gajda[2], Tomasz Lehmann[2], Michał Podstawski[3],
Hubert Niewiadomski[2], Piotr Nyczyk[2], Torsten Hoefler[1]
[1]ETH Zurich, [2]Cledar, [3]Warsaw University of Technology
bestam@inf.ethz.ch, nils.blach@inf.ethz.ch, htor@inf.ethz.ch

Inference related

Replies („thoughts”)

Prompts

Reasoning Structures



Agents, Tools

GNN

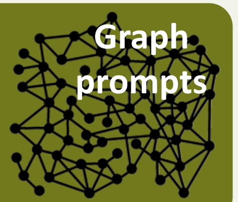
Heuristic

DAG

Psychology

Team

Prompt engineering



Retrieval

Graph database

Knowledge graph

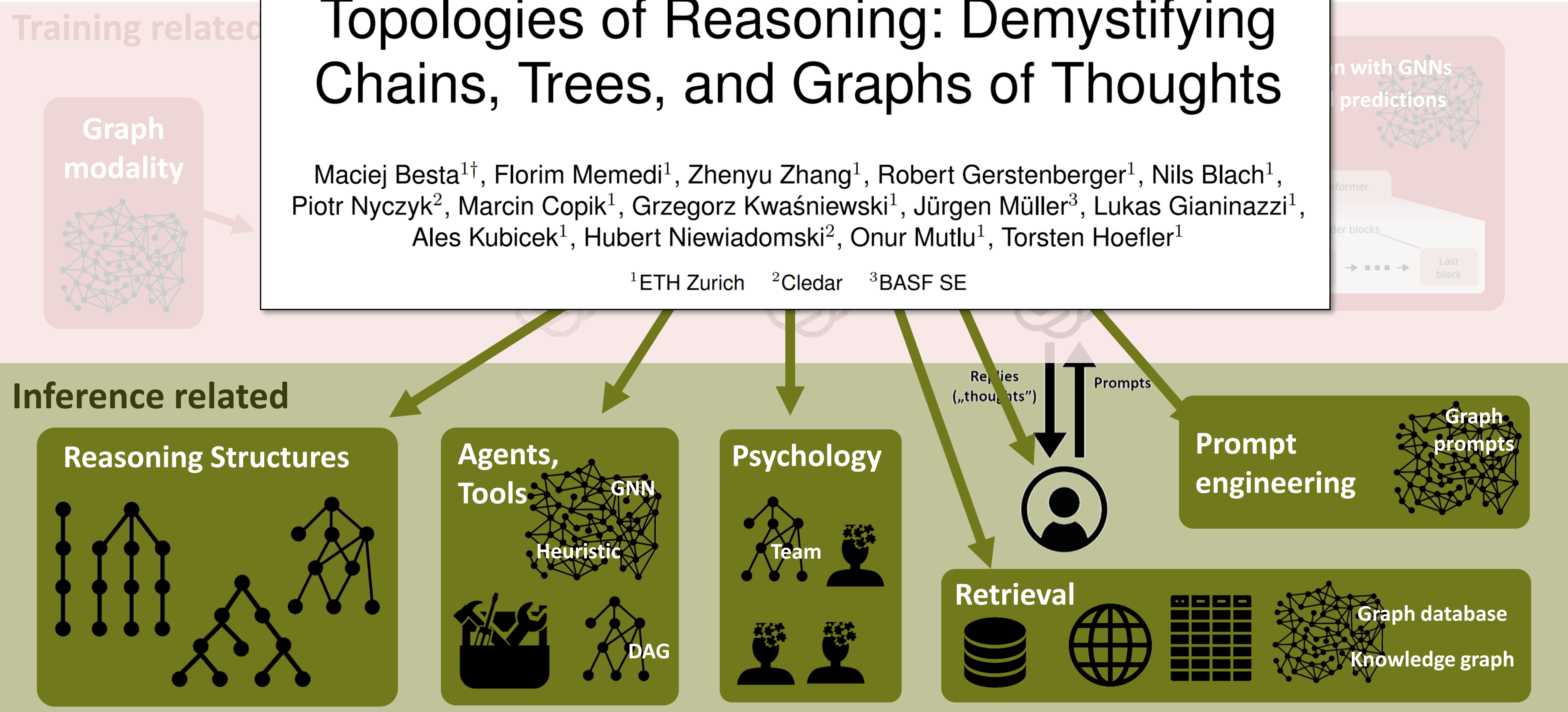
Graphs in the

Training related

Topologies of Reasoning: Demystifying Chains, Trees, and Graphs of Thoughts

Maciej Besta[1†], Florim Memedi[1], Zhenyu Zhang[1], Robert Gerstenberger[1], Nils Blach[1], Piotr Nyczyk[2], Marcin Copik[1], Grzegorz Kwaśniewski[1], Jürgen Müller[3], Lukas Gianinazzi[1], Ales Kubicek[1], Hubert Niewiadomski[2], Onur Mutlu[1], Torsten Hoefler[1]

[1]ETH Zurich [2]Cledar [3]BASF SE



Graphs in the LLM Pipeline: Overview

Trai[...]

Affordable AI Assistants with Knowledge Graph of Thoughts

Maciej Besta† [1] Lorenzo Paleari [1] Jia Hao Andrea Jiang [1] Robert Gerstenberger [1] You Wu [1] Patrick Iff [1]
Ales Kubicek [1] Piotr Nyczyk [2] Diana Khimey [1] Jón Gunnar Hannesson [1] Grzegorz Kwaśniewski [1]
Marcin Copik [1] Hubert Niewiadomski [2] Torsten Hoefler [1]

Inference related

Agents, Tools

Agents, Tools

Graph as a model of the agent's task and its state

Agents, Tools

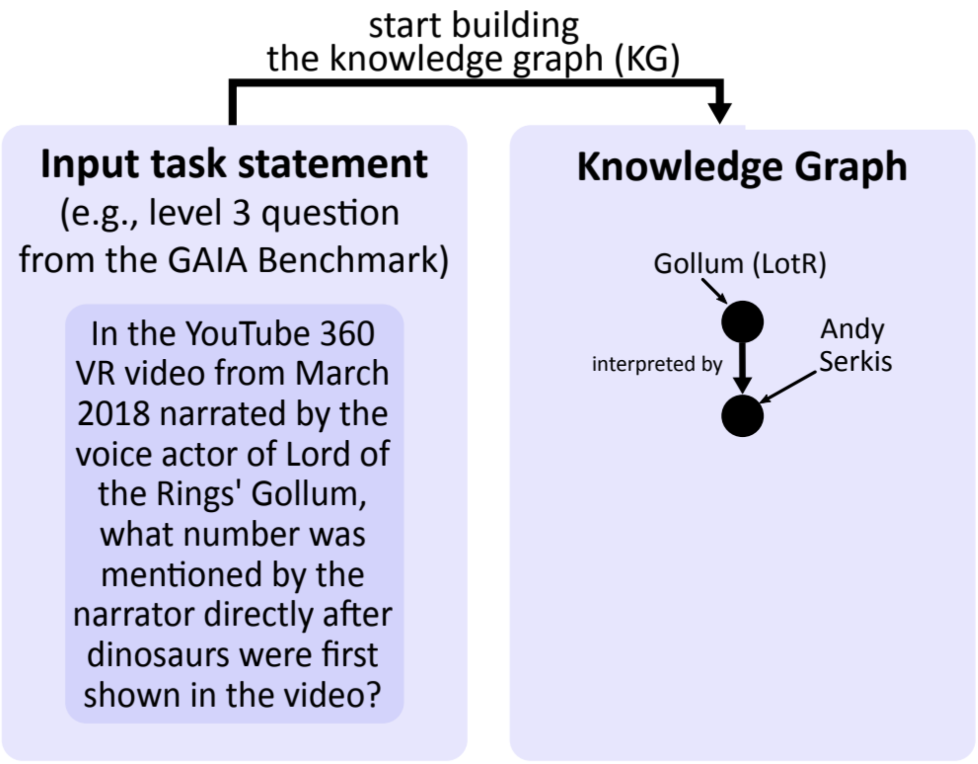
Graph as a model of the agent’s task and its state

Input task statement
(e.g., level 3 question from the GAIA Benchmark)

In the YouTube 360 VR video from March 2018 narrated by the voice actor of Lord of the Rings' Gollum, what number was mentioned by the narrator directly after dinosaurs were first shown in the video?

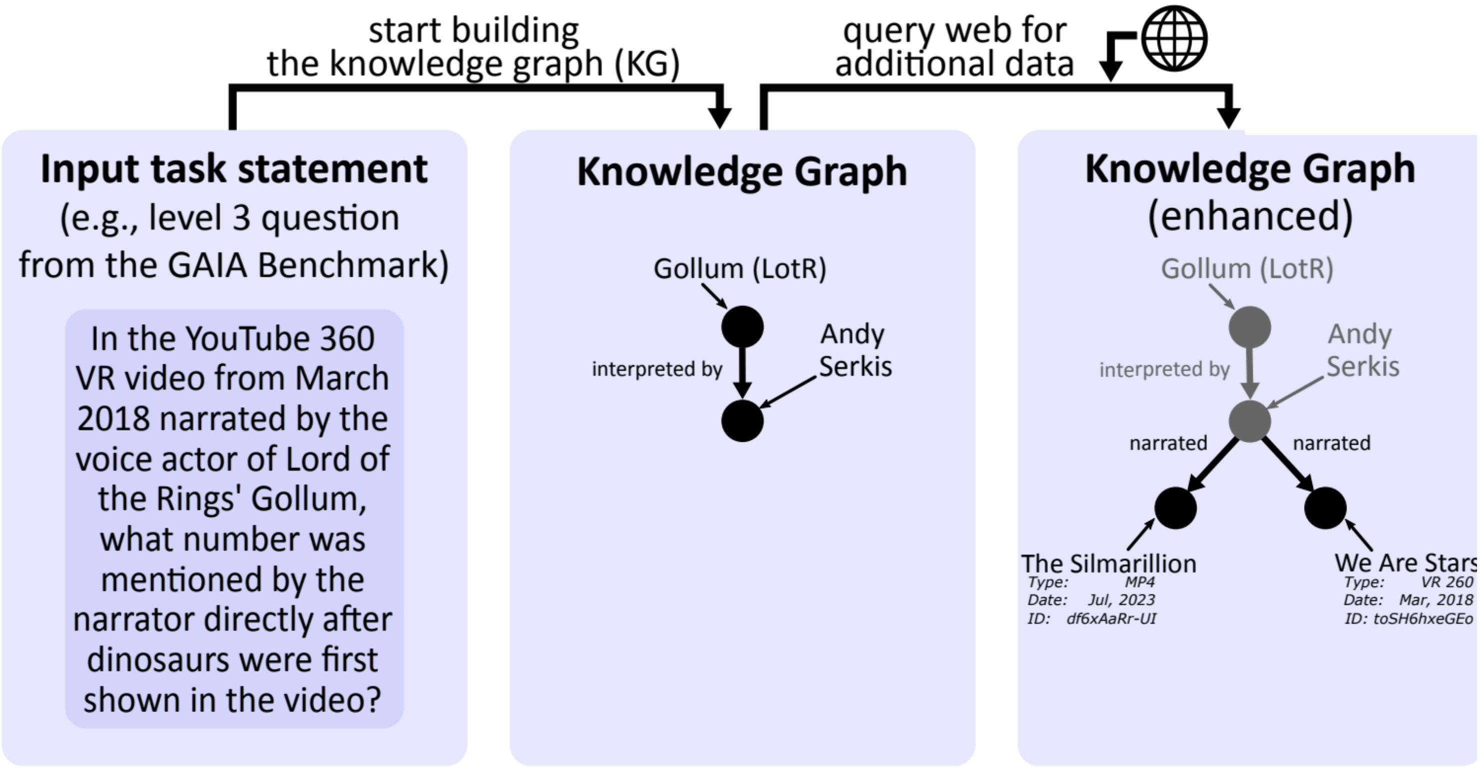
Agents, Tools

Graph as a model of the agent’s task and its state



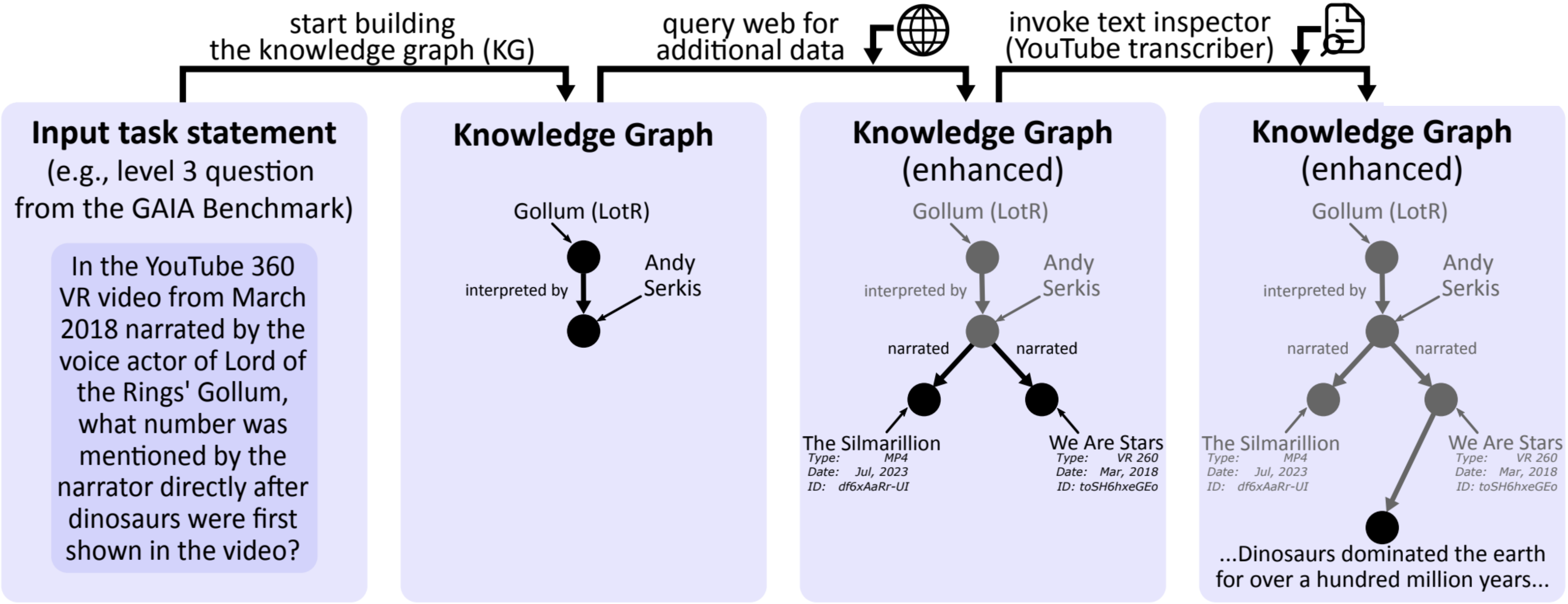
Agents, Tools

Graph as a model of the agent's task and its state



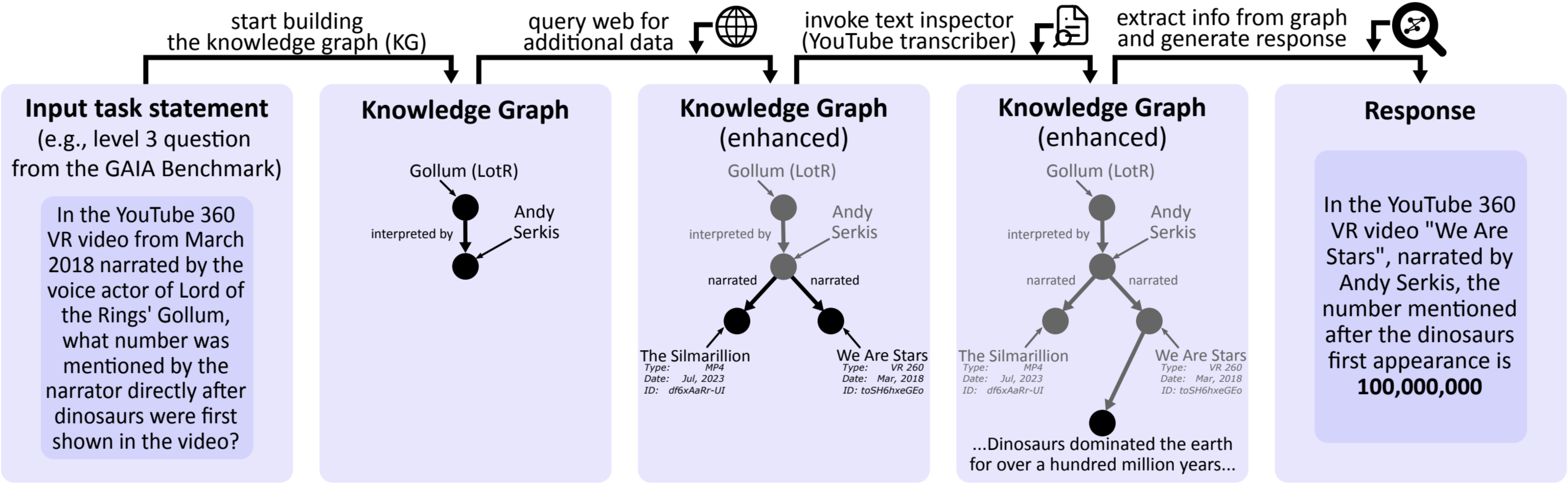
Agents, Tools

Graph as a model of the agent's task and its state



Agents, Tools

Graph as a model of the agent's task and its state



Graphs in the LLM Pipeline: Overview

Affordable AI Assistants with Knowledge Graph of Thoughts

Maciej Besta† [1] Lorenzo Paleari [1] Jia Hao Andrea Jiang [1] Robert Gerstenberger [1] You Wu [1] Patrick Iff [1]
Ales Kubicek [1] Piotr Nyczyk [2] Diana Khimey [1] Jón Gunnar Hannesson [1] Grzegorz Kwaśniewski [1]
Marcin Copik [1] Hubert Niewiadomski [2] Torsten Hoefler [1]

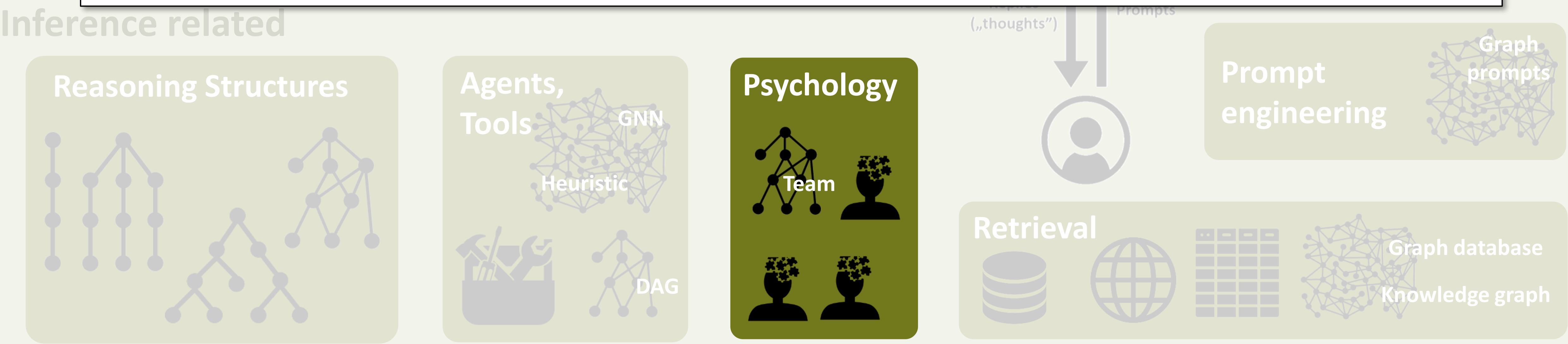
Graphs in the LLM Pipeline: Overview

Psychologically Enhanced AI Agents

Maciej Besta[1†], Shriram Chandran[1], Robert Gerstenberger[1], Mathis Lindner[1], Marcin Chrapek[1], Sebastian Hermann Martschat[2], Taraneh Ghandi[2*], Patrick Iff[1], Hubert Niewiadomski[3,4], Piotr Nyczyk[3,4], Jürgen Müller[2], Torsten Hoefler[1]

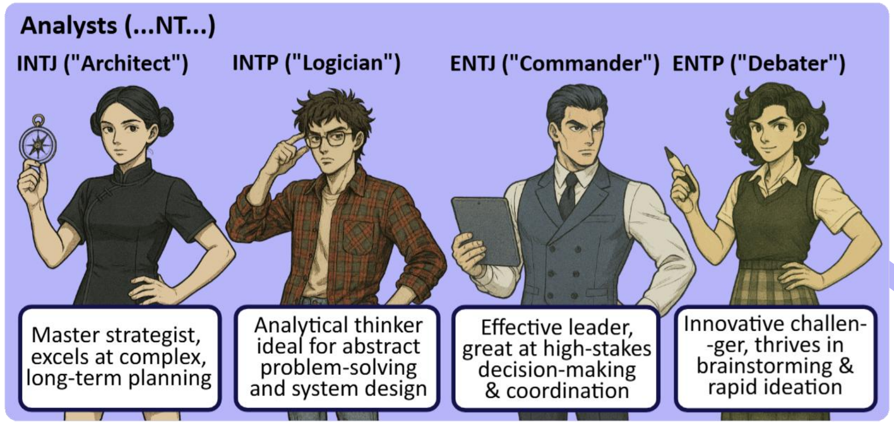
[1]ETH Zurich [2]BASF SE [3]Cledar [4]IDEAS Research Institute

†*Corresponding author*

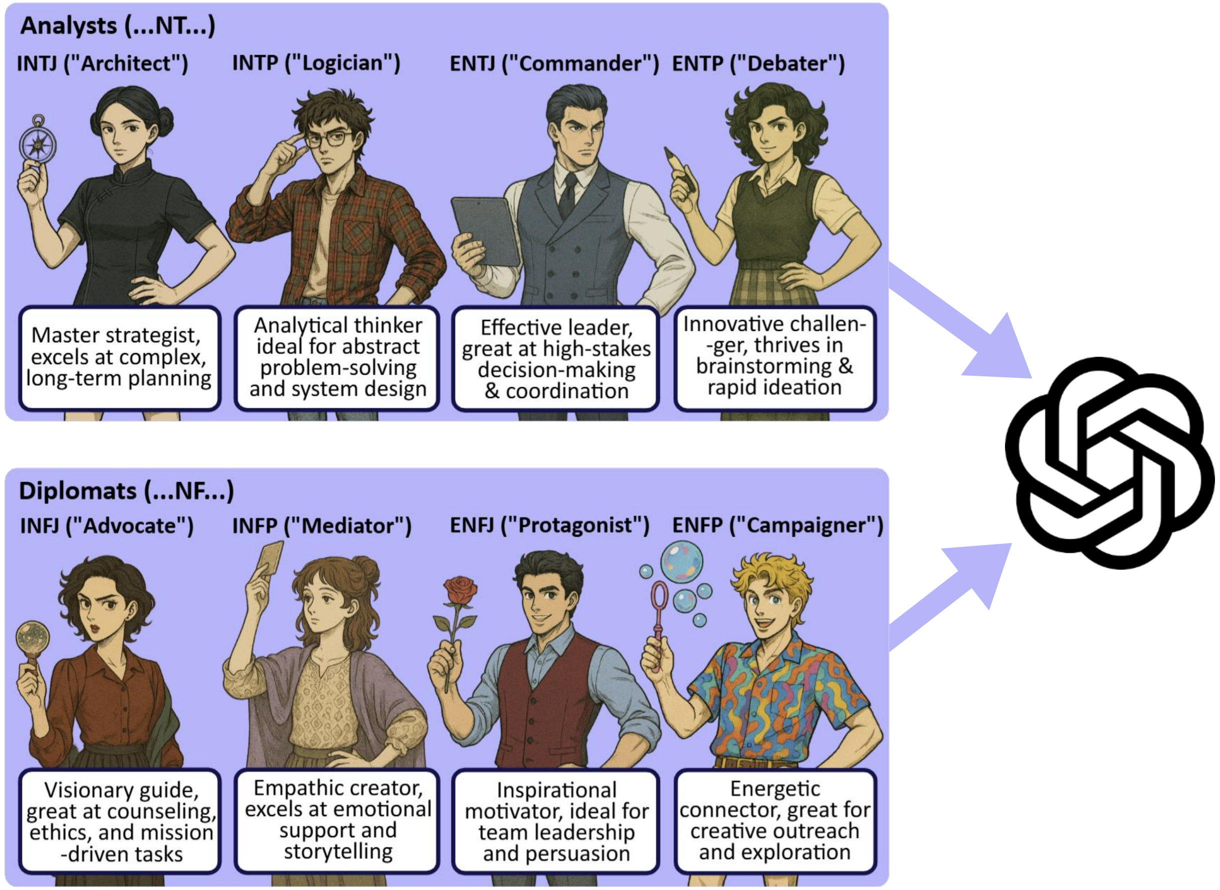


Psychology of Agents

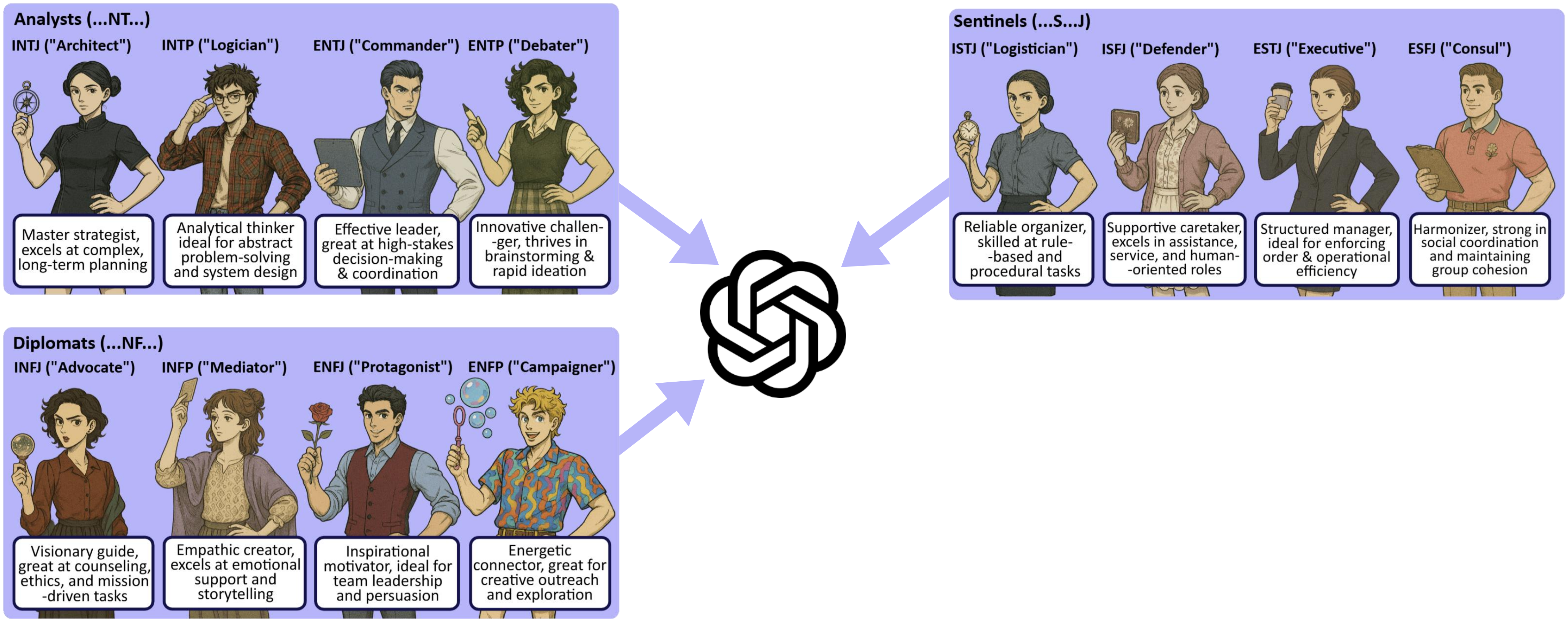
Psychology of Agents



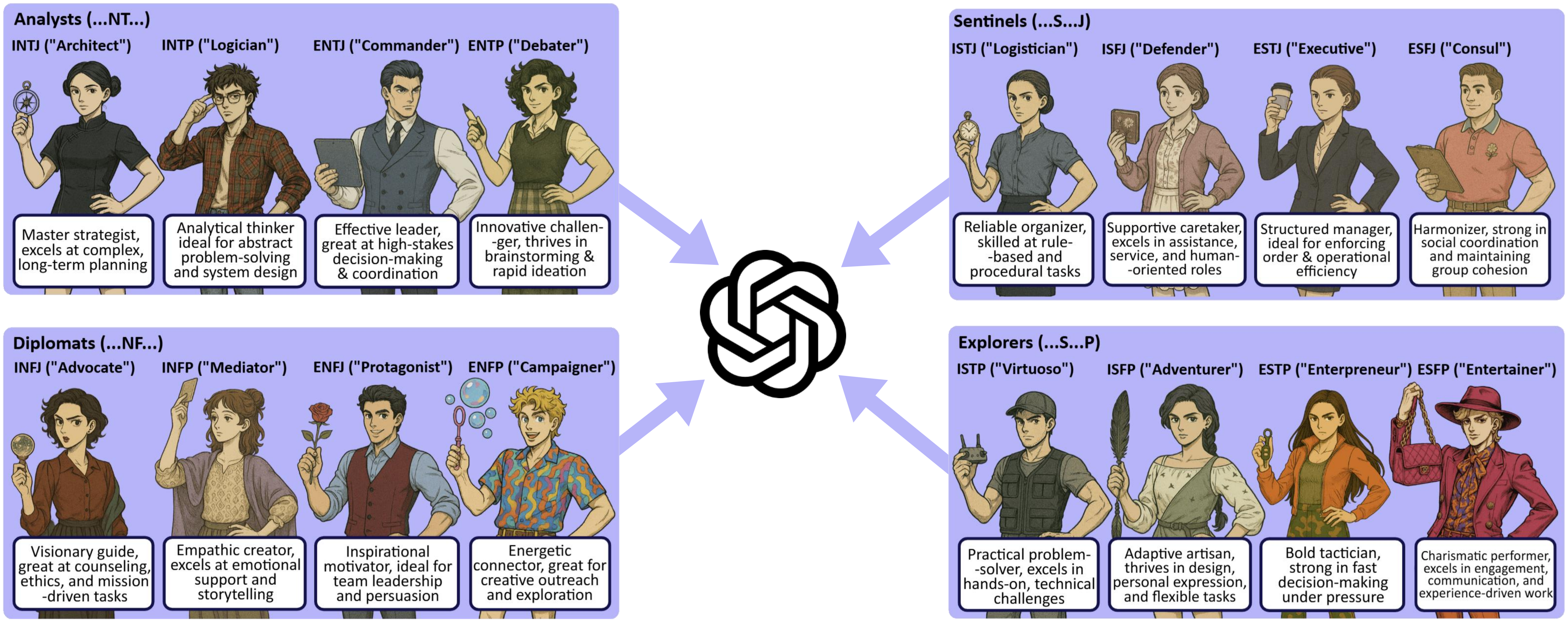
Psychology of Agents



Psychology of Agents

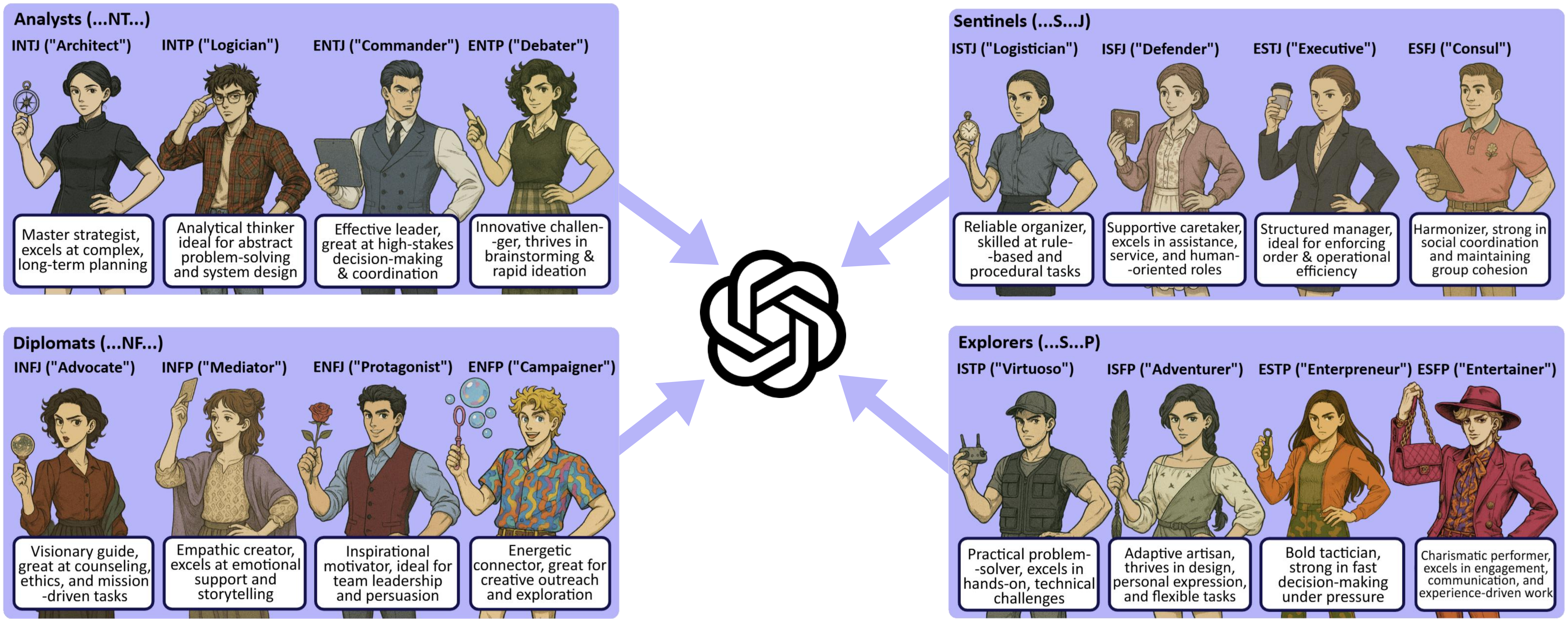


Psychology of Agents



Psychology of Agents

Graph as a model of the inter-agent communication pattern

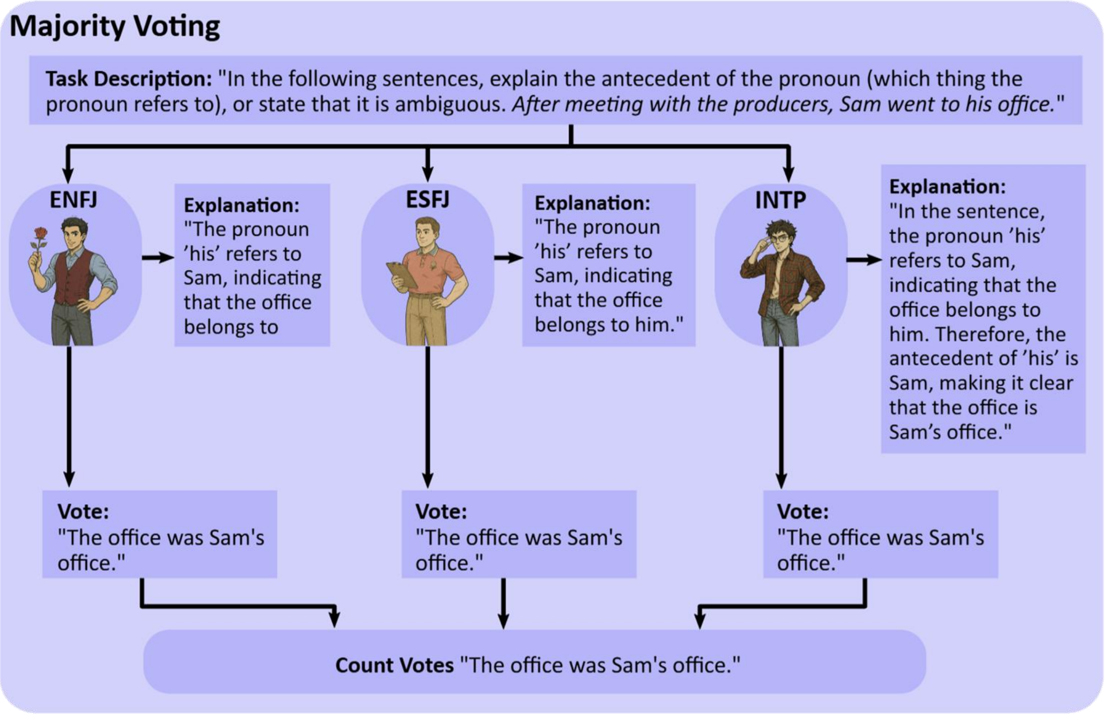


Psychology of Agents

Graph as a model of the inter-agent communication pattern

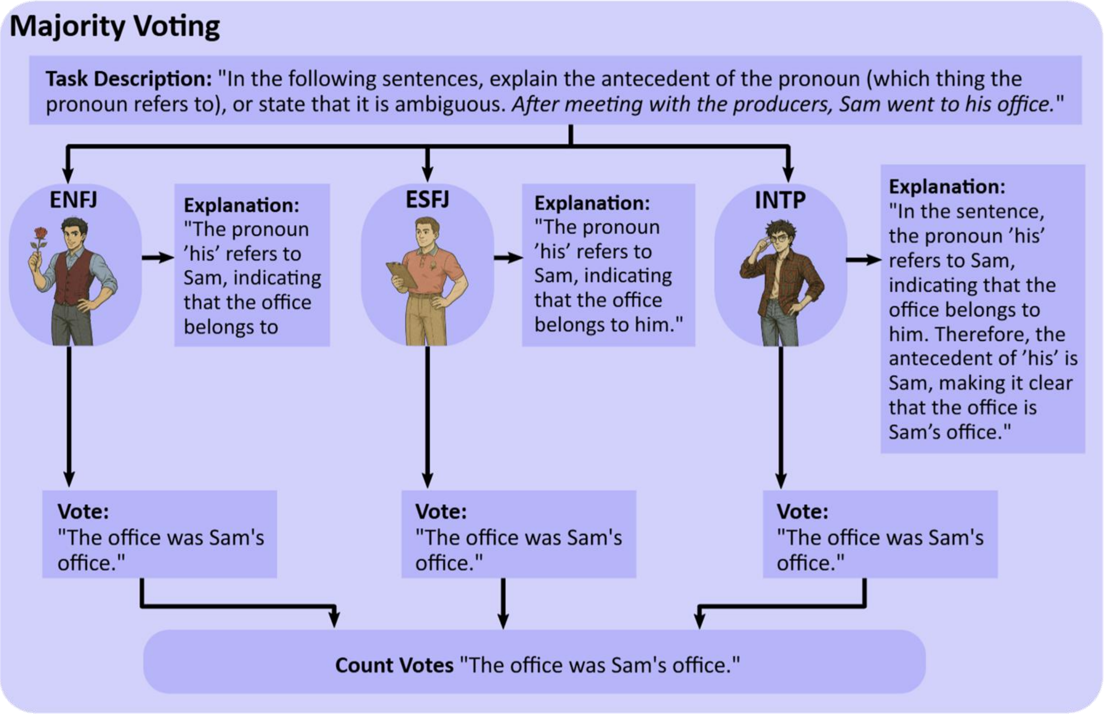
Psychology of Agents

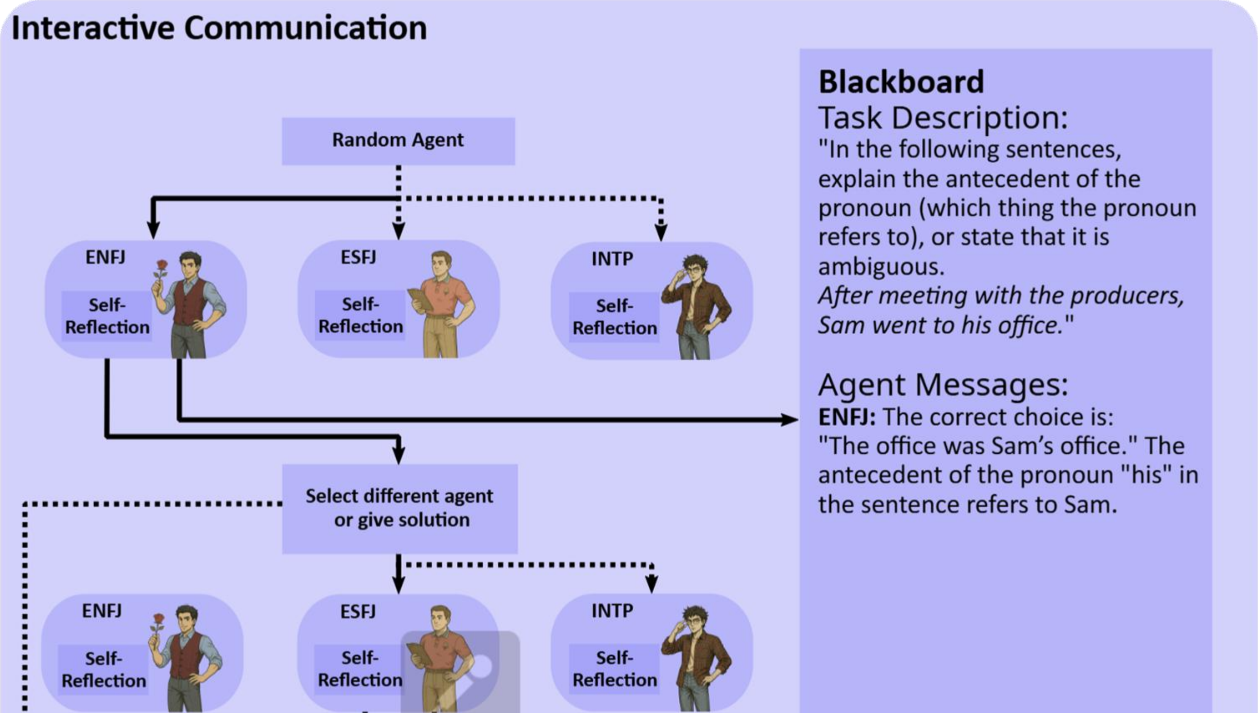
Graph as a model of the inter-agent communication pattern



Psychology of Agents

Graph as a model of the inter-agent communication pattern





Graphs in the LLM Pipeline: Overview

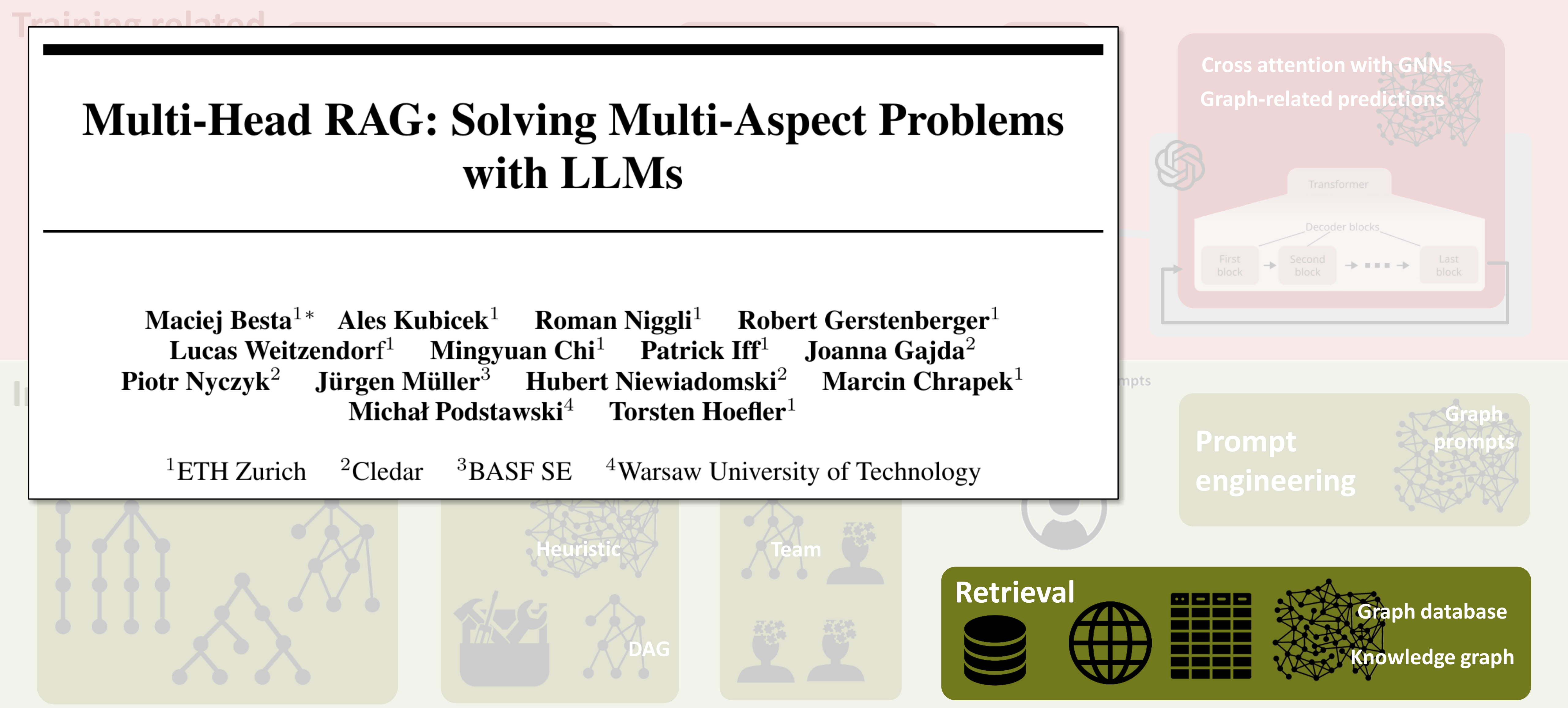
Psychologically Enhanced AI Agents

Maciej Besta[1†], Shriram Chandran[1], Robert Gerstenberger[1], Mathis Lindner[1], Marcin Chrapek[1], Sebastian Hermann Martschat[2], Taraneh Ghandi[2*], Patrick Iff[1], Hubert Niewiadomski[3,4], Piotr Nyczyk[3,4], Jürgen Müller[2], Torsten Hoefler[1]

[1]ETH Zurich [2]BASF SE [3]Cledar [4]IDEAS Research Institute

†Corresponding author

Graphs in the LLM Pipeline: Overview



Multi-Head RAG: Solving Multi-Aspect Problems with LLMs

Maciej Besta[1*] Ales Kubicek[1] Roman Niggli[1] Robert Gerstenberger[1]
Lucas Weitzendorf[1] Mingyuan Chi[1] Patrick Iff[1] Joanna Gajda[2]
Piotr Nyczyk[2] Jürgen Müller[3] Hubert Niewiadomski[2] Marcin Chrapek[1]
Michał Podstawski[4] Torsten Hoefler[1]

[1]ETH Zurich [2]Cledar [3]BASF SE [4]Warsaw University of Technology

Graphs in the LLM

Training related

Graph modality

Grap... Four... Mod...

Raw data

...ss attention with GNNs
...ph-related predictions

Transformer
Decoder blocks
Second block
Last block

CheckEmbed: Effective Verification of LLM Solutions to Open-Ended Tasks

Maciej Besta* ETH Zurich
Lorenzo Paleari ETH Zurich
Marcin Copik ETH Zurich
Robert Gerstenberger ETH Zurich
Ales Kubicek ETH Zurich
Piotr Nyczyk Cledar
Patrick Iff ETH Zurich
Eric Schreiber ETH Zurich
Tanja Srindran ETH Zurich
Tomasz Lehmann Cledar, Warsaw University of Technology
Hubert Niewiadomski Cledar, IDEAS Research Institute
Torsten Hoefler ETH Zurich

Inference related

Reasoning Structures

Agents, Tools
GNN
Heuristic
DAG

Psychology
Team

Replies („thoughts")
Prompts

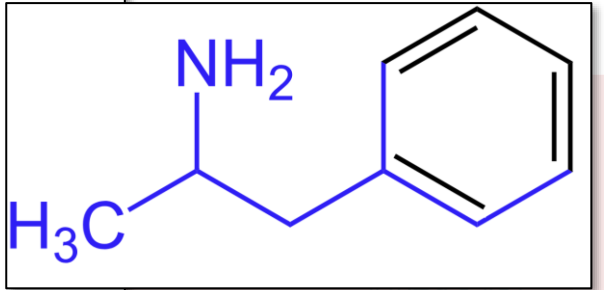
Prompt engineering
Graph prompts

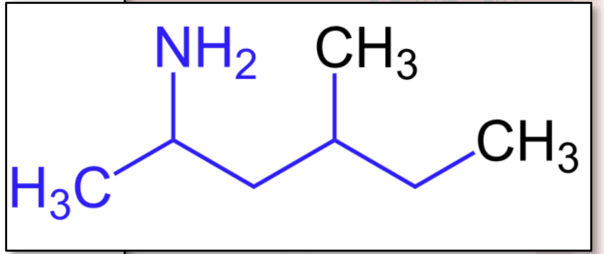
Retrieval
Graph database
Knowledge graph

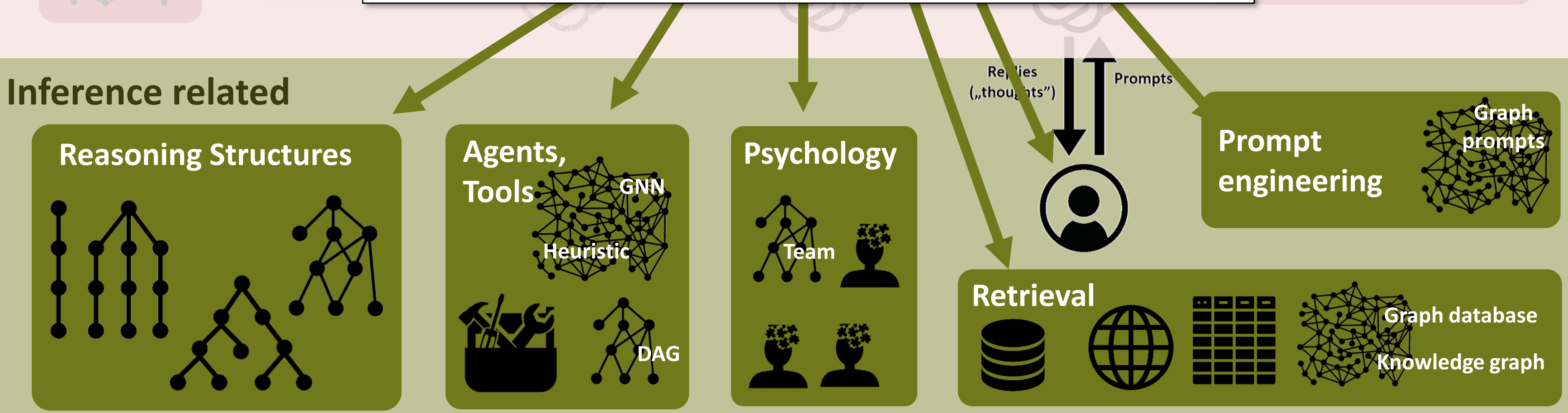
Graphs in the LLM

CHECKEMBED: Effective Verification of LLM Solutions to Open-Ended Tasks

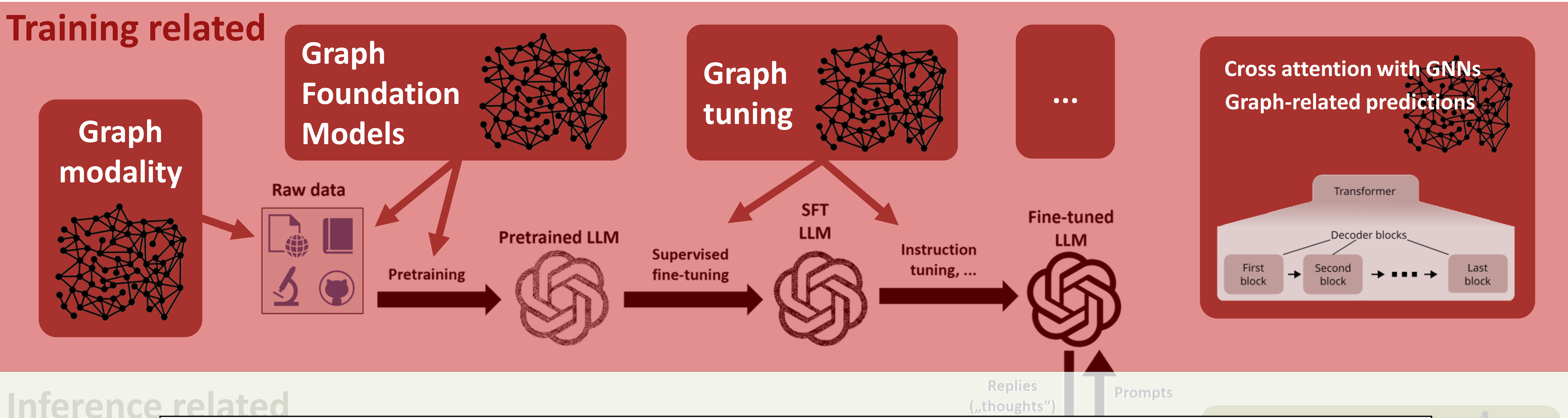
Maciej Besta* ETH Zurich
Lorenzo Paleari ETH Zurich
Marcin Copik ETH Zurich
Robert Gerstenberger ETH Zurich
Ales Kubicek ETH Zurich
Piotr Nyczyk Cledar
Patrick Iff ETH Zurich
Eric Schreiber ETH Zurich
Tanja Srindran ETH Zurich
Tomasz Lehmann Cledar, Warsaw University of Technology
Hubert Niewiadomski Cledar, IDEAS Research Institute
Torsten Hoefler ETH Zurich







Graphs in the LLM Pipeline: Overview



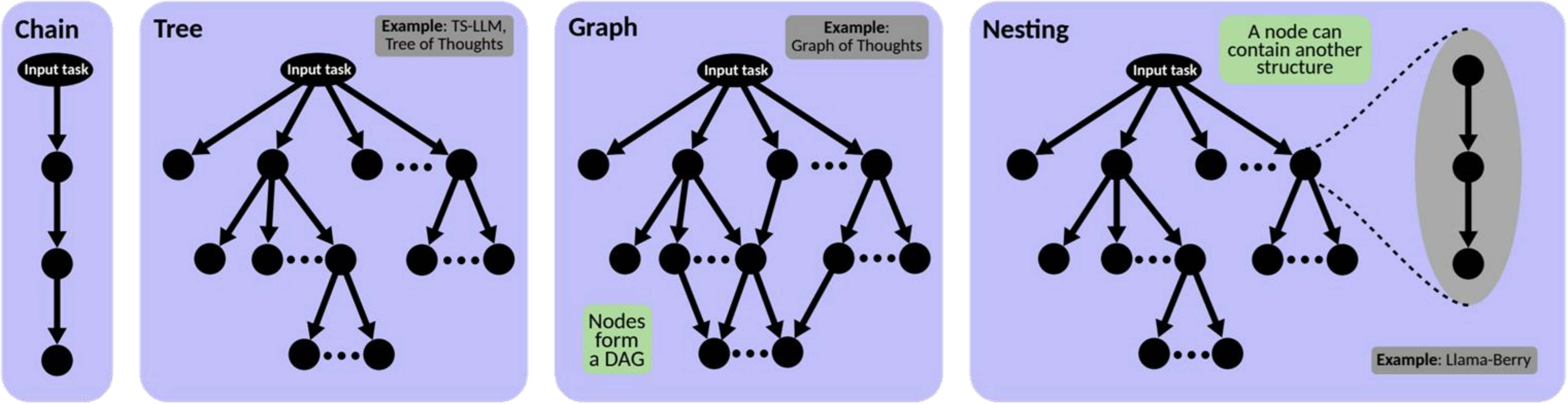
Reasoning Language Models: A Blueprint

Maciej Besta[1†], Julia Barth[1], Eric Schreiber[1], Ales Kubicek[1], Afonso Catarino[1], Robert Gerstenberger[1], Piotr Nyczyk[2], Patrick Iff[1], Yueling Li[3], Sam Houliston[1], Tomasz Sternal[1], Marcin Copik[1], Grzegorz Kwaśniewski[1], Jürgen Müller[3], Łukasz Flis[4], Hannes Eberhard[1], Hubert Niewiadomski[2], Torsten Hoefler[1]

[†] *Corresponding author* [1]ETH Zurich [2]Cledar [3]BASF SE [4]Cyfronet AGH

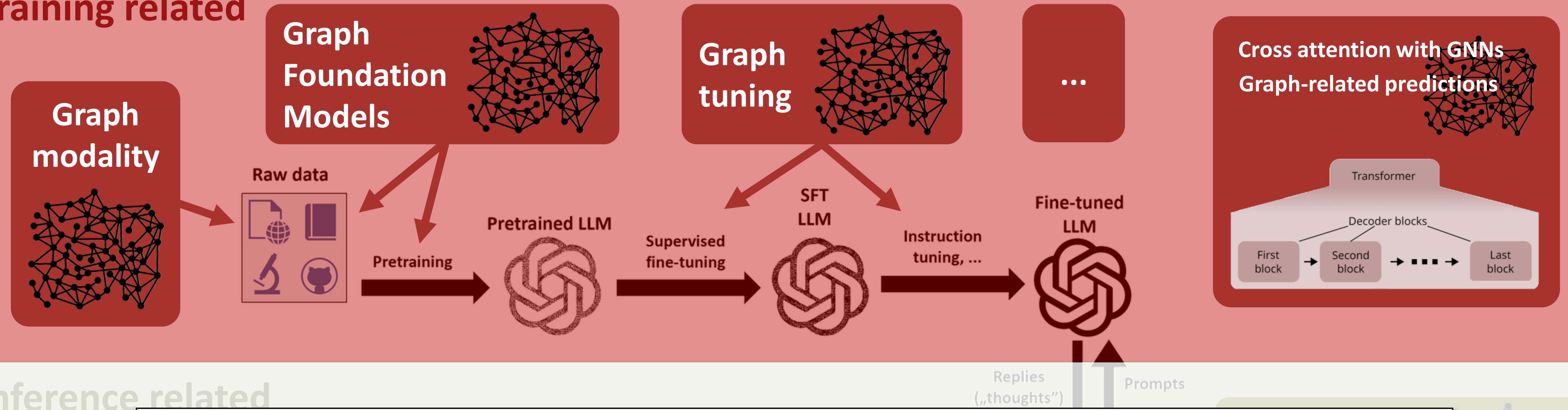
Reasoning Paradigms

Graph as a model of the reasoning process (training)



Graphs in the LLM Pipeline: Overview

Training related



Inference related

Reason

Graph prompts

database

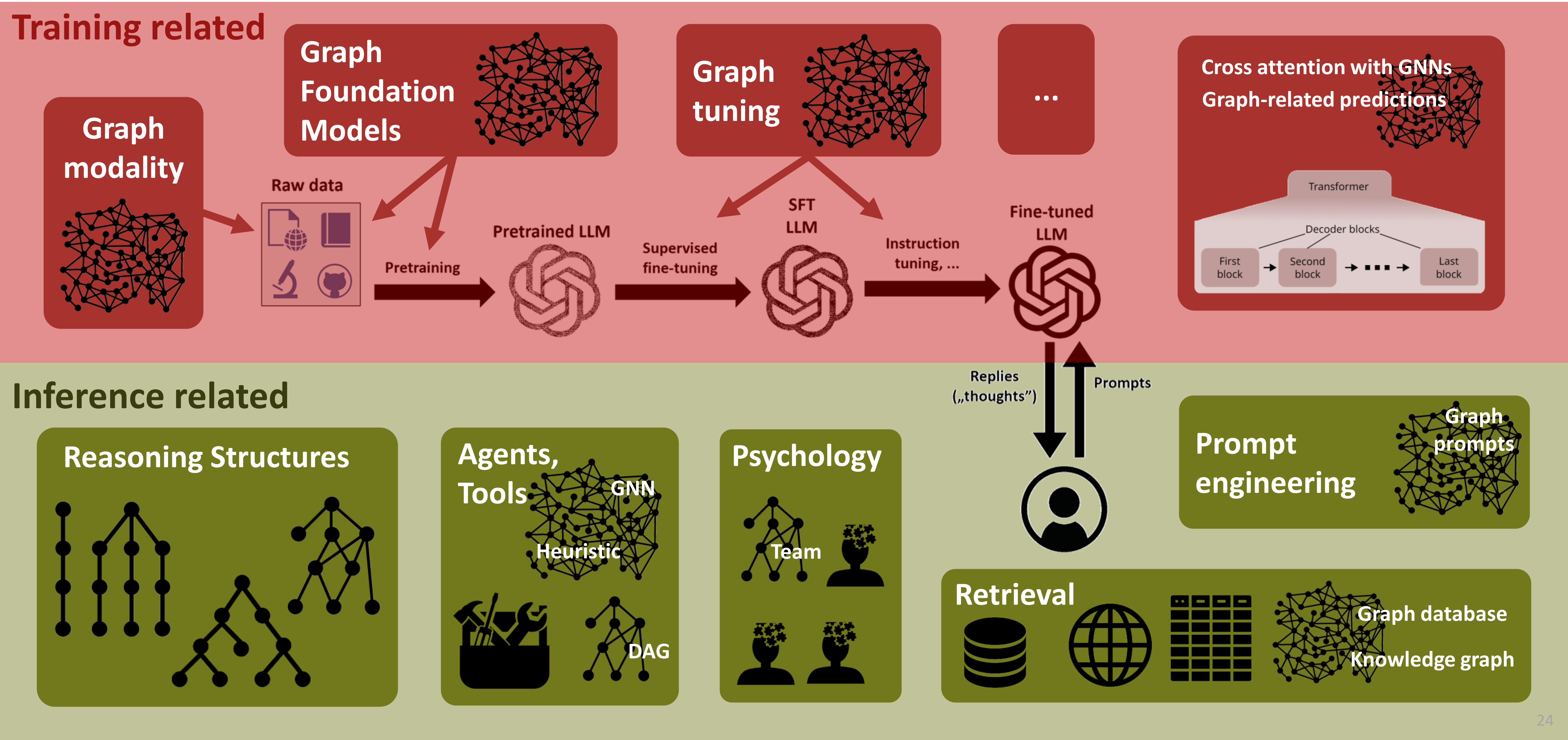
edge graph

Reasoning Language Models: A Blueprint

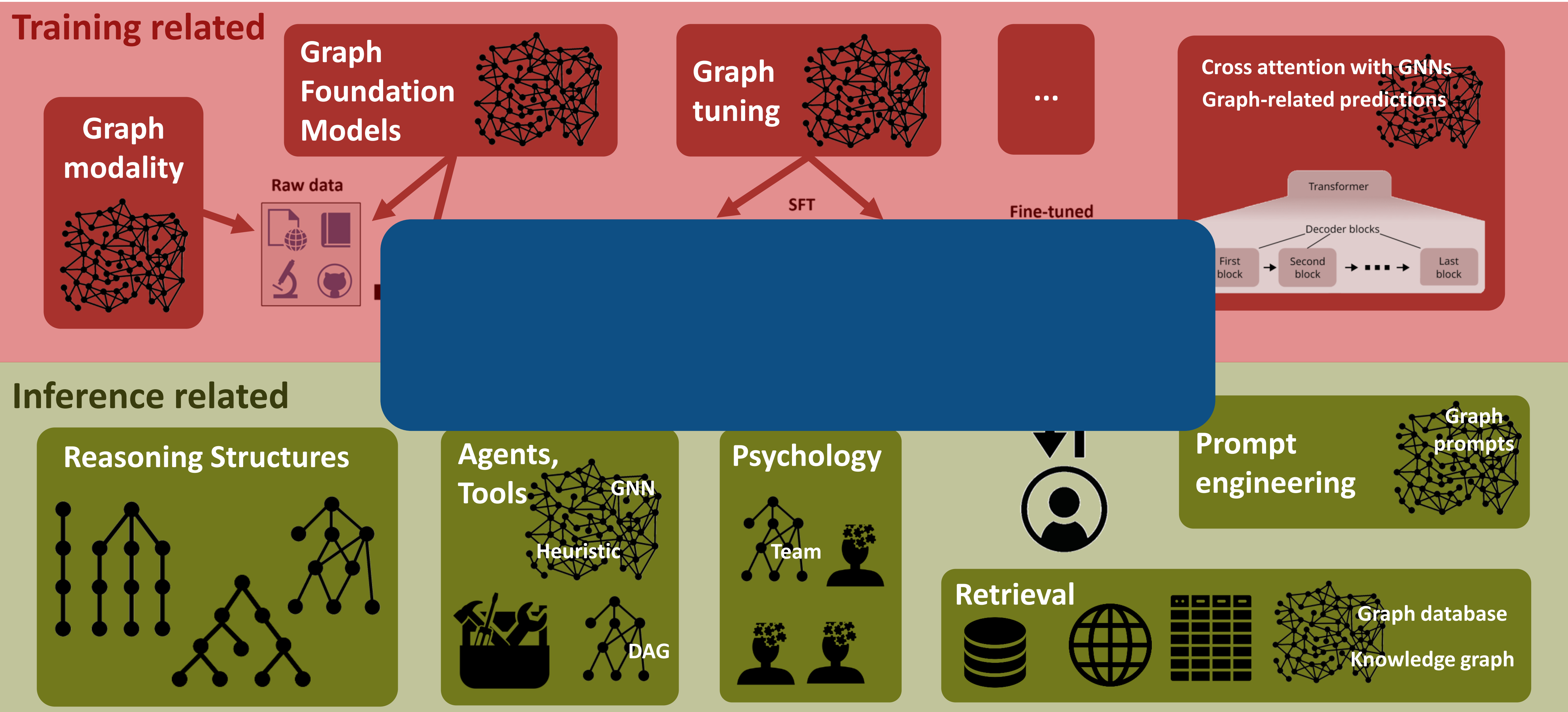
Maciej Besta[1†], Julia Barth[1], Eric Schreiber[1], Ales Kubicek[1], Afonso Catarino[1], Robert Gerstenberger[1], Piotr Nyczyk[2], Patrick Iff[1], Yueling Li[3], Sam Houliston[1], Tomasz Sternal[1], Marcin Copik[1], Grzegorz Kwaśniewski[1], Jürgen Müller[3], Łukasz Flis[4], Hannes Eberhard[1], Hubert Niewiadomski[2], Torsten Hoefler[1]

[†] *Corresponding author* [1] ETH Zurich [2] Cledar [3] BASF SE [4] Cyfronet AGH

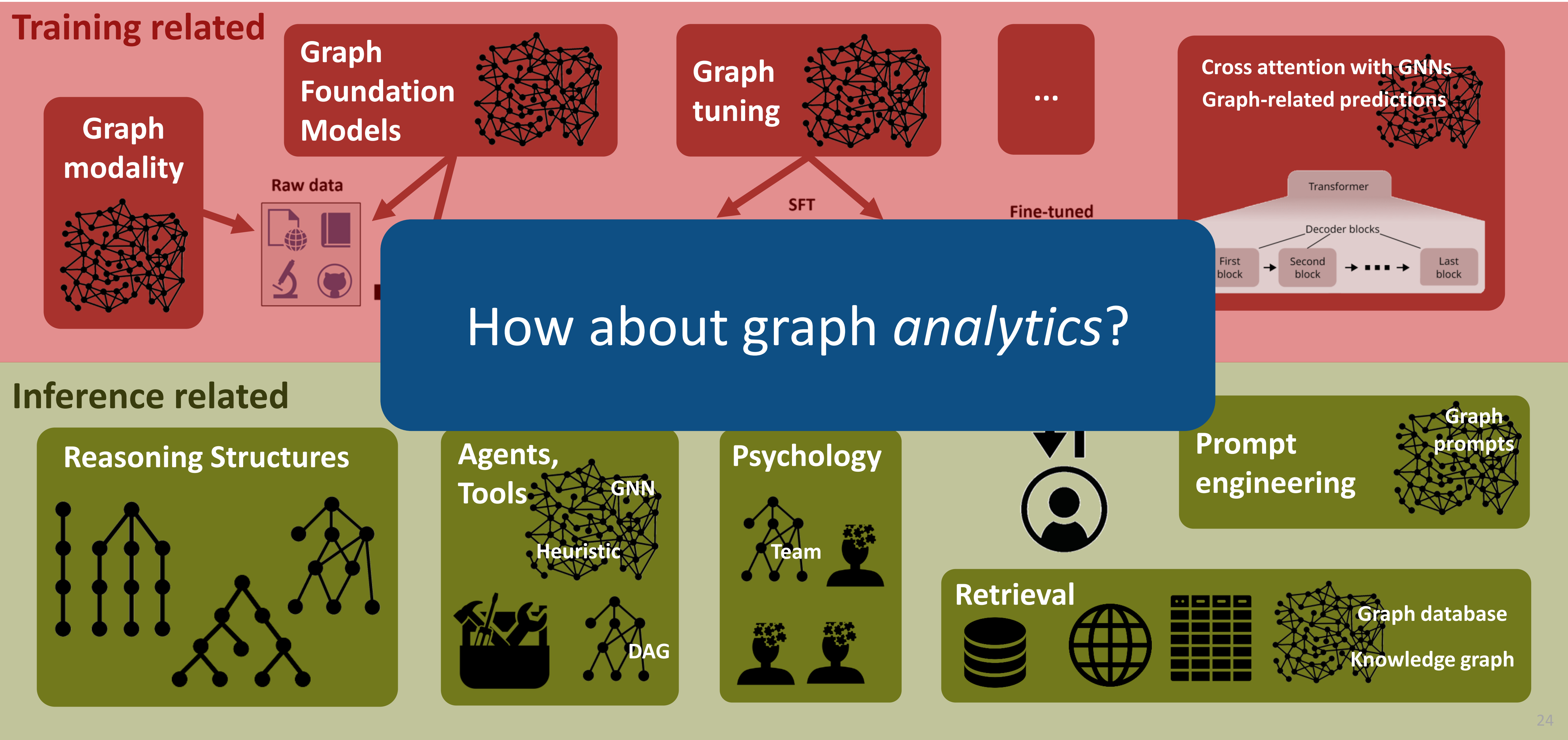
Graphs in the LLM Pipeline: Overview



Graphs in the LLM Pipeline: Overview



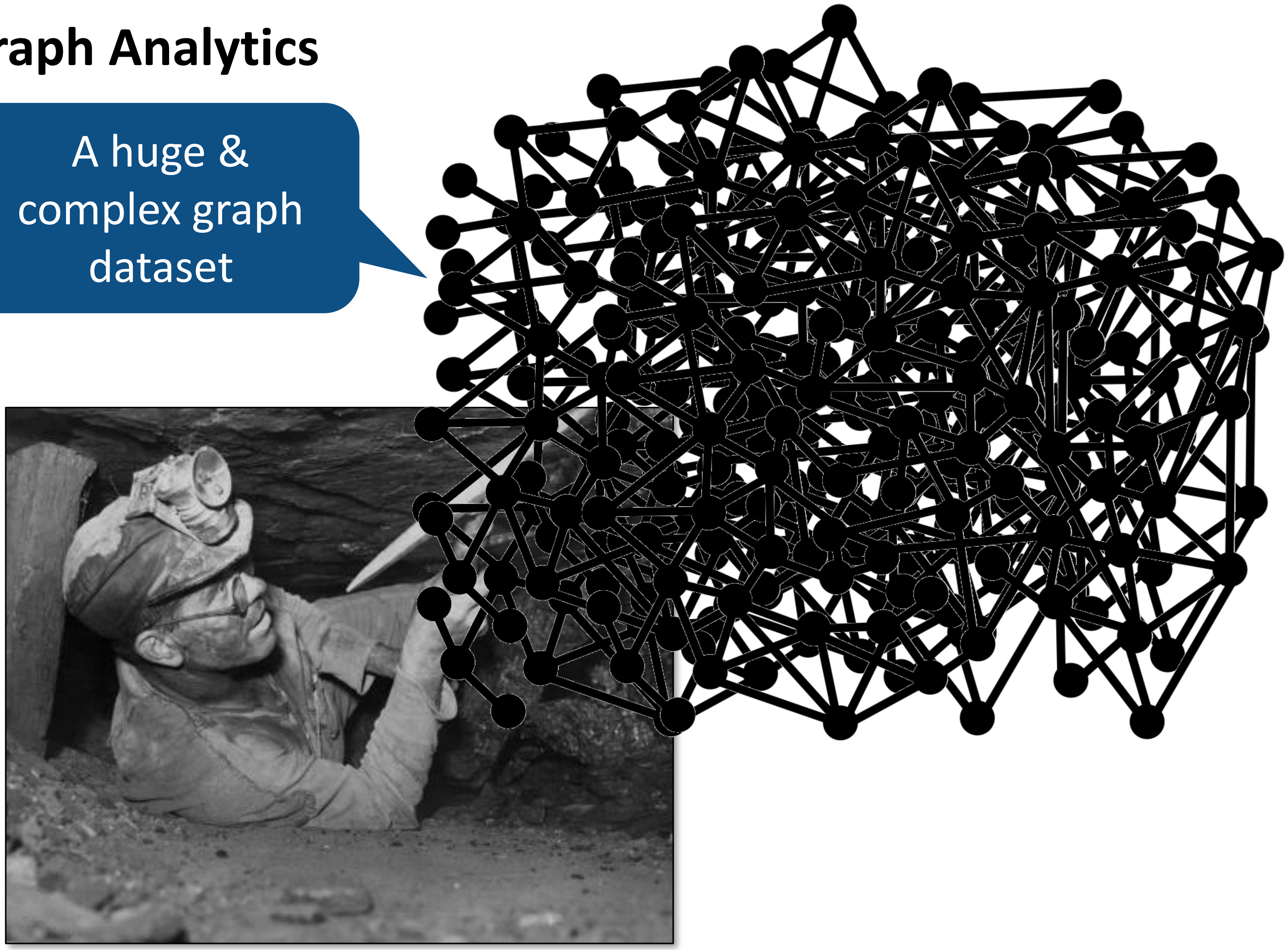
Graphs in the LLM Pipeline: Overview



Graph Analytics

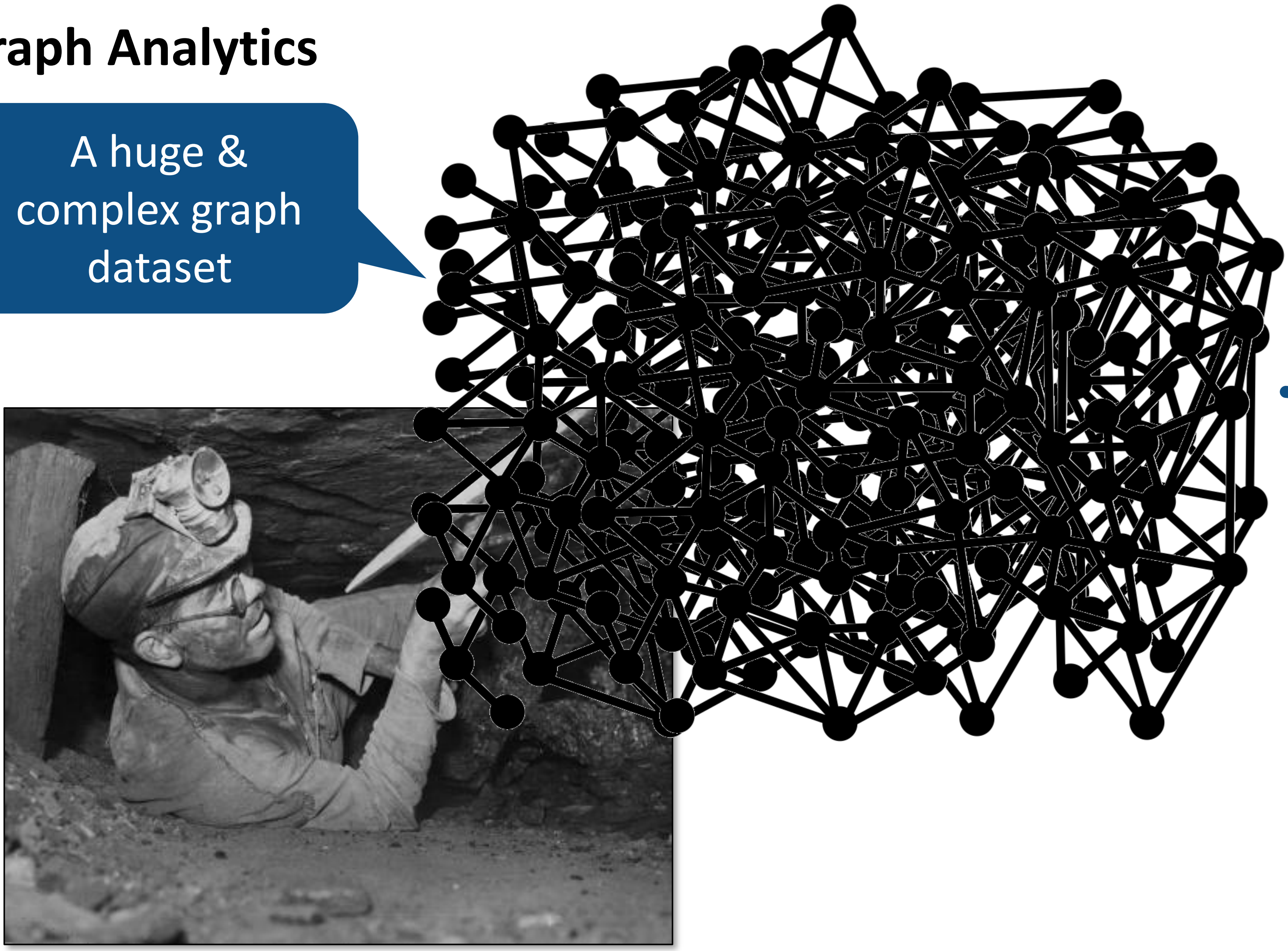
Graph Analytics

A huge & complex graph dataset

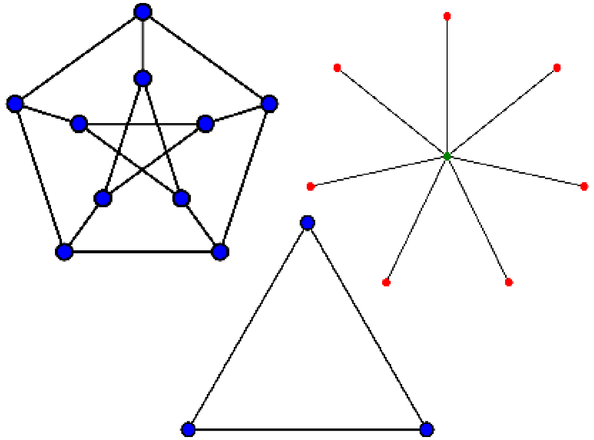


Graph Analytics

A huge & complex graph dataset

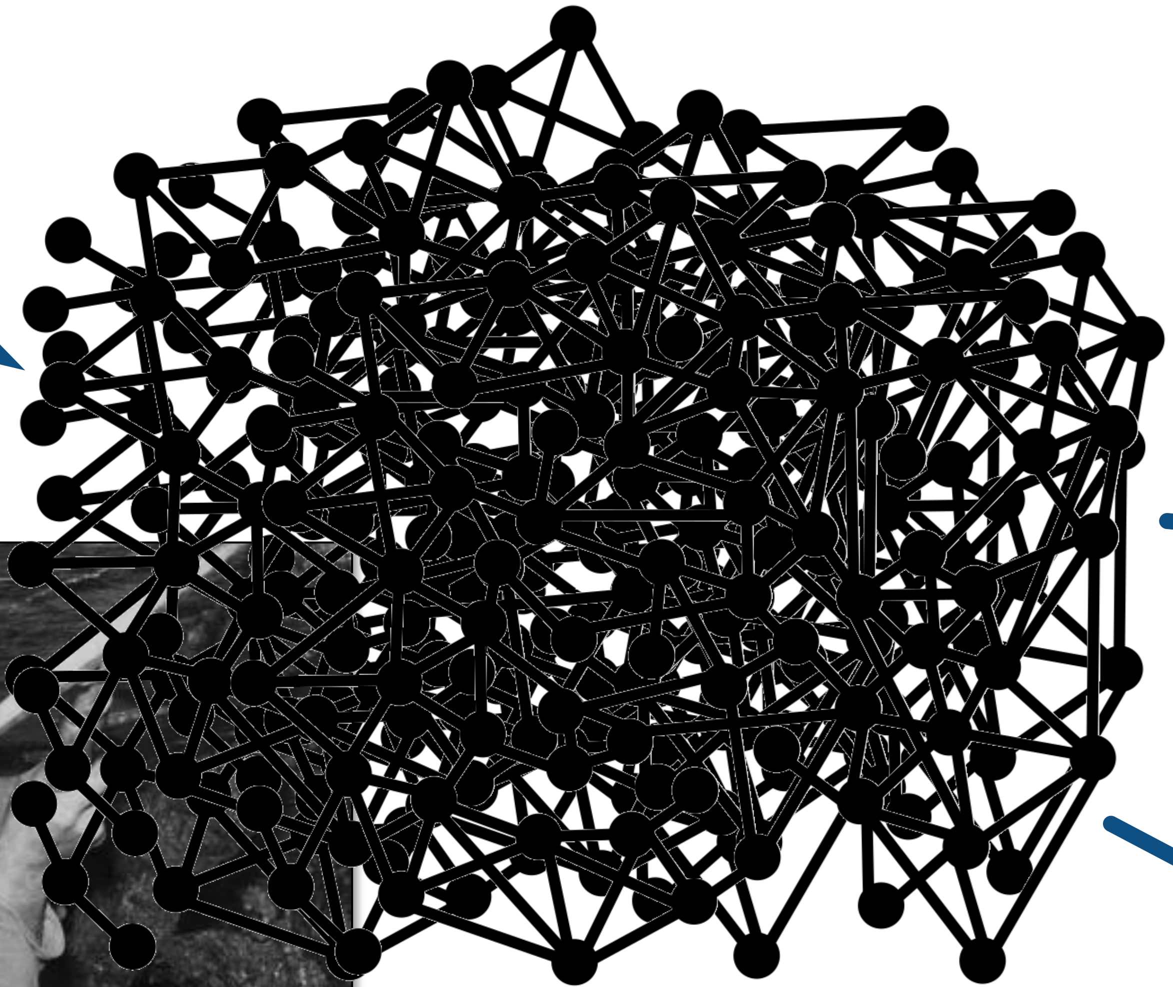


Pattern counting (triangles, higher-order cliques, dense subgraphs, ...)

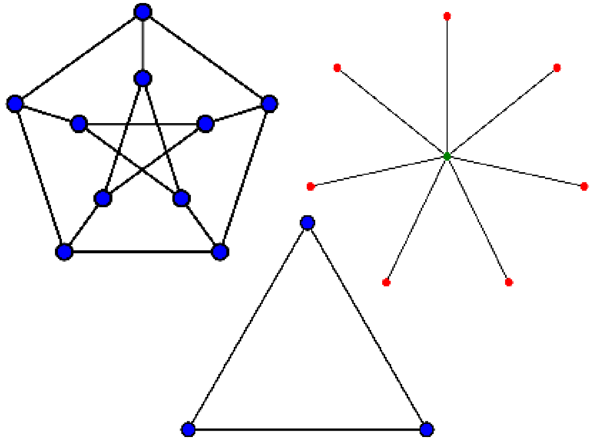


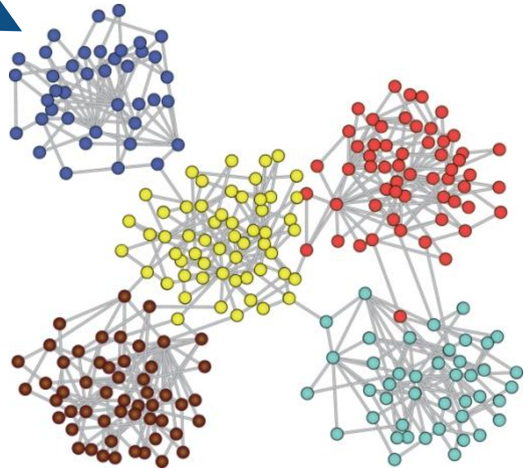
Graph Analytics

A huge & complex graph dataset



Pattern counting (triangles, higher-order cliques, dense subgraphs, ...)

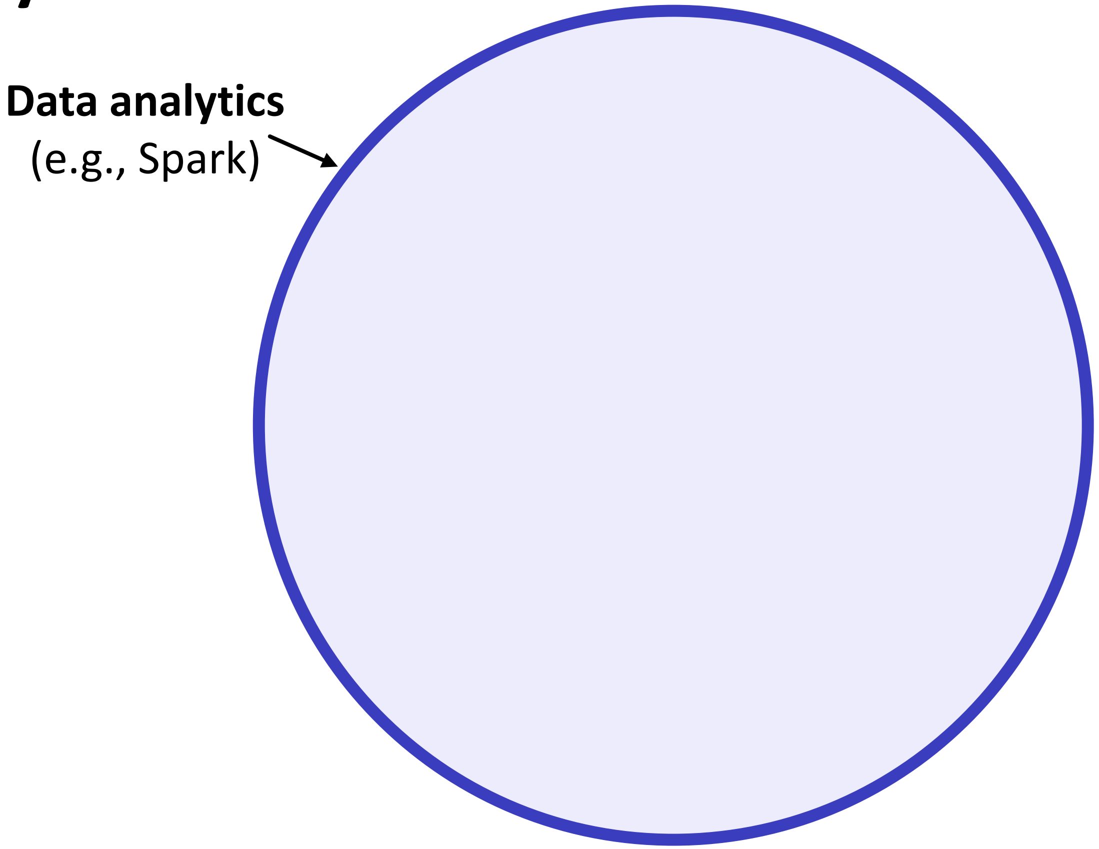




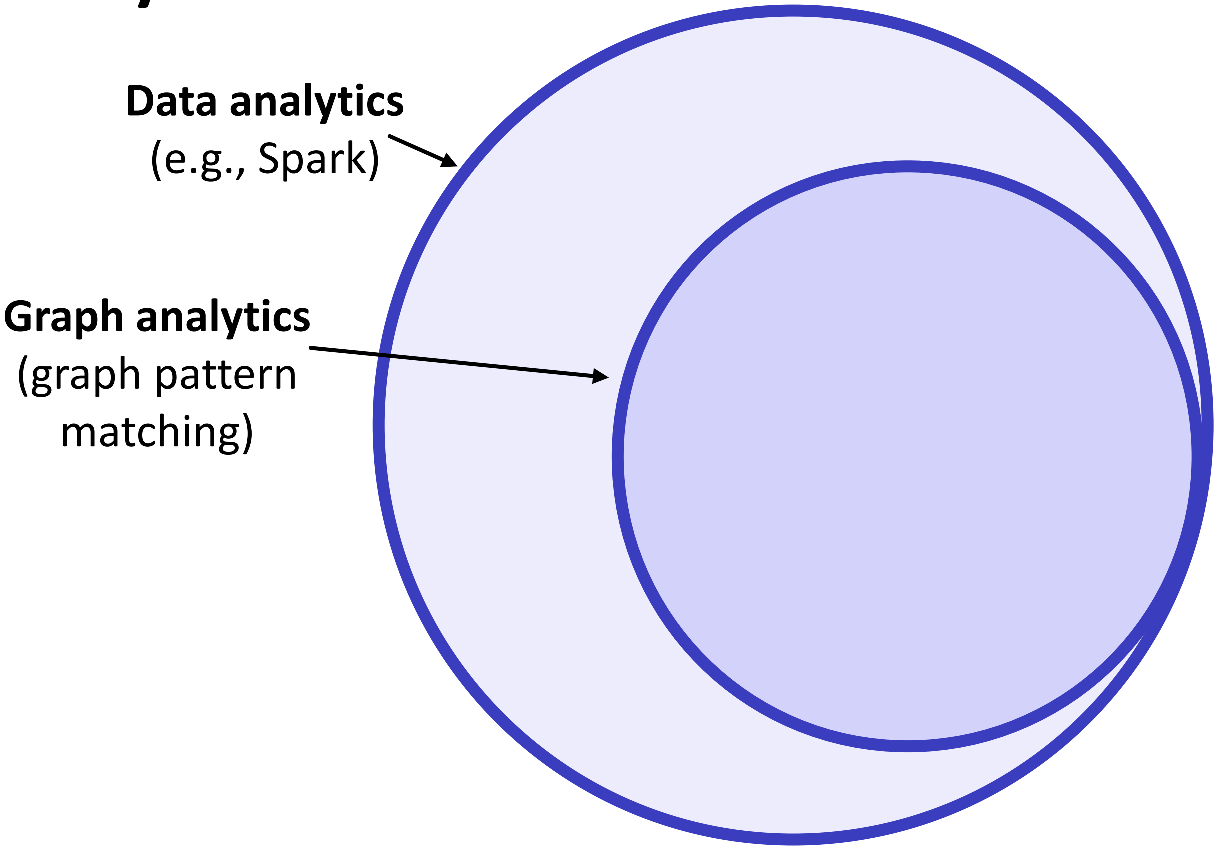
Clustering, Link Prediction, Vertex Similarity, ...

Graph Analytics + LLMs

Graph Analytics + LLMs



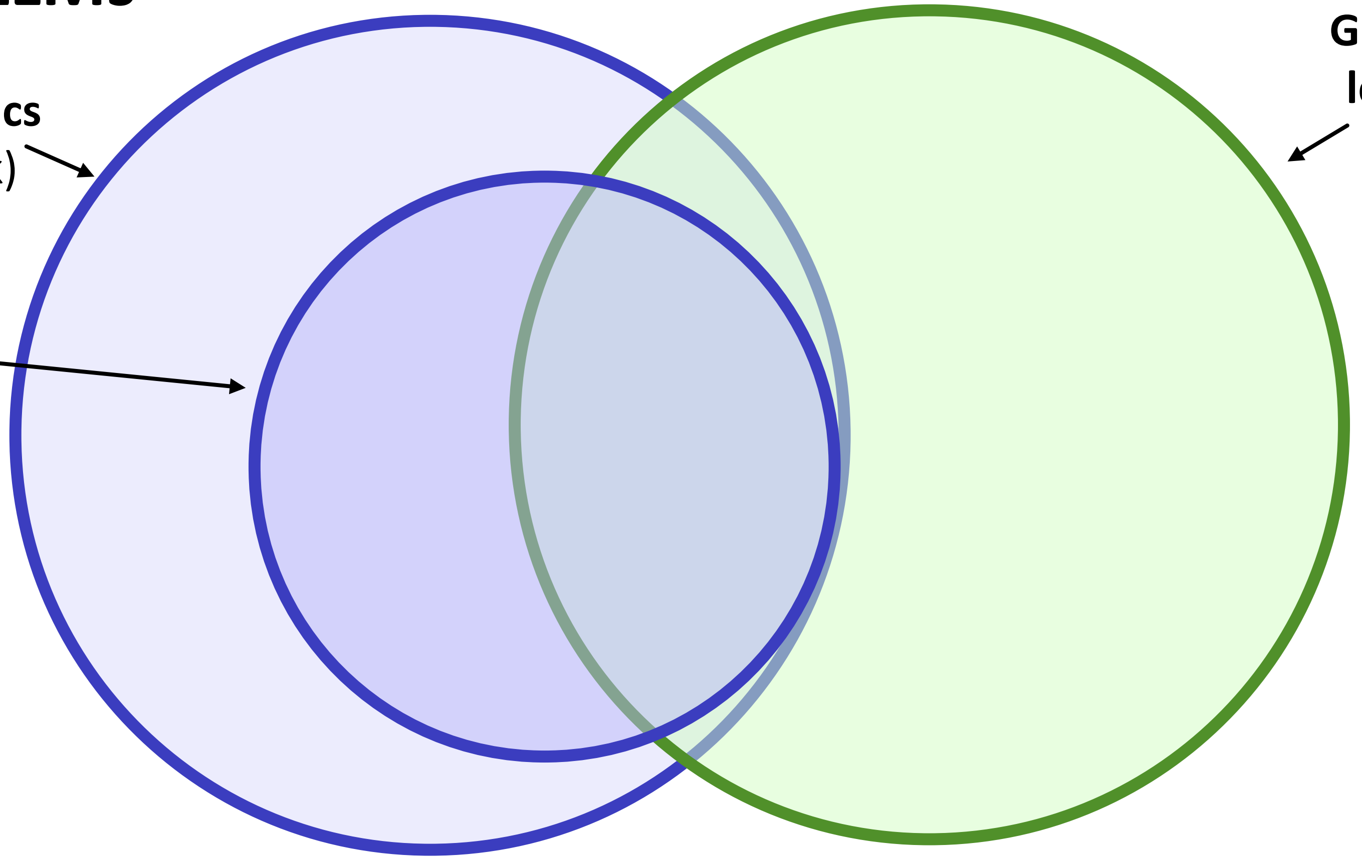
Graph Analytics + LLMs



Graph Analytics + LLMs

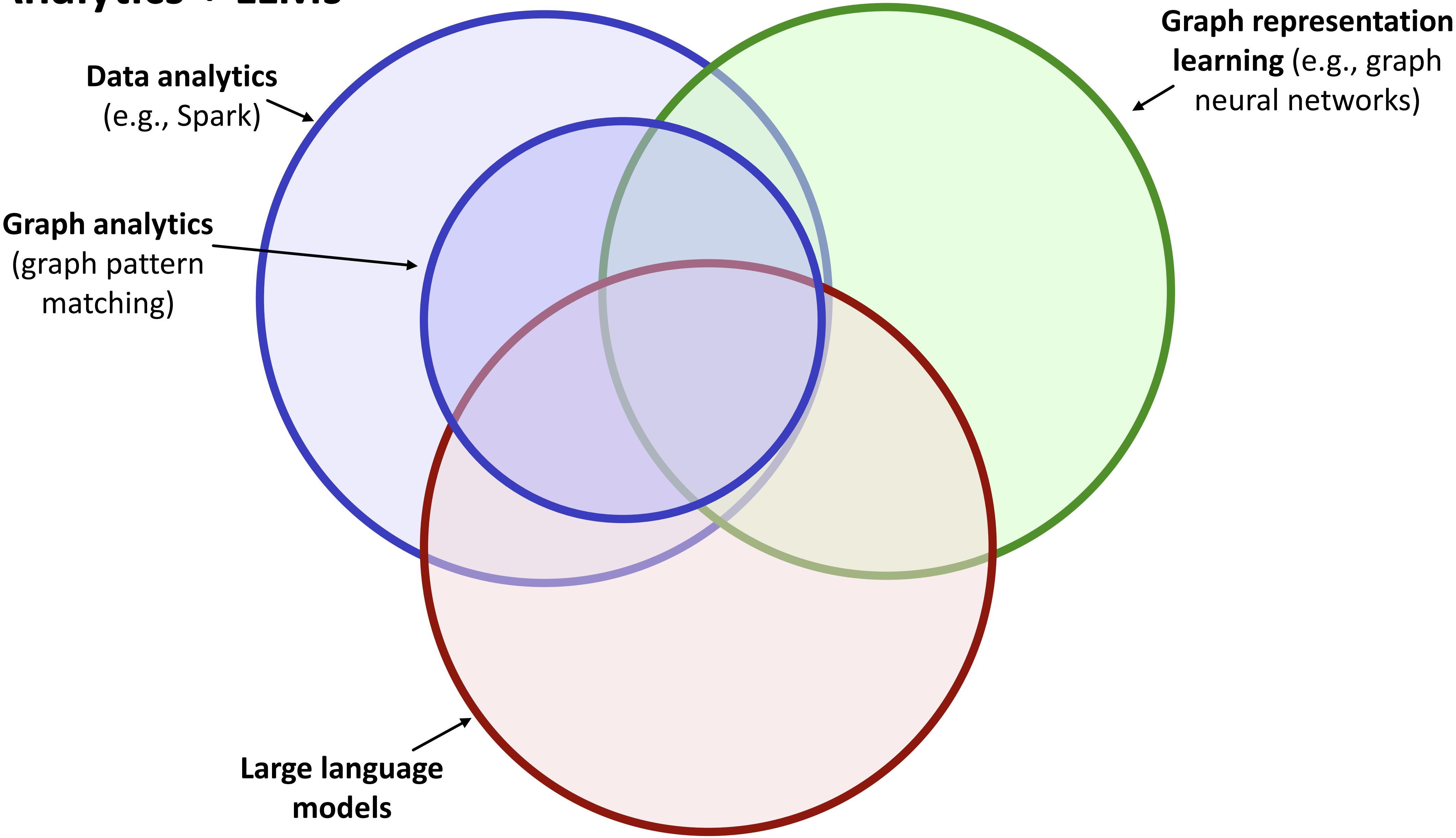
Data analytics
(e.g., Spark)

Graph analytics
(graph pattern matching)

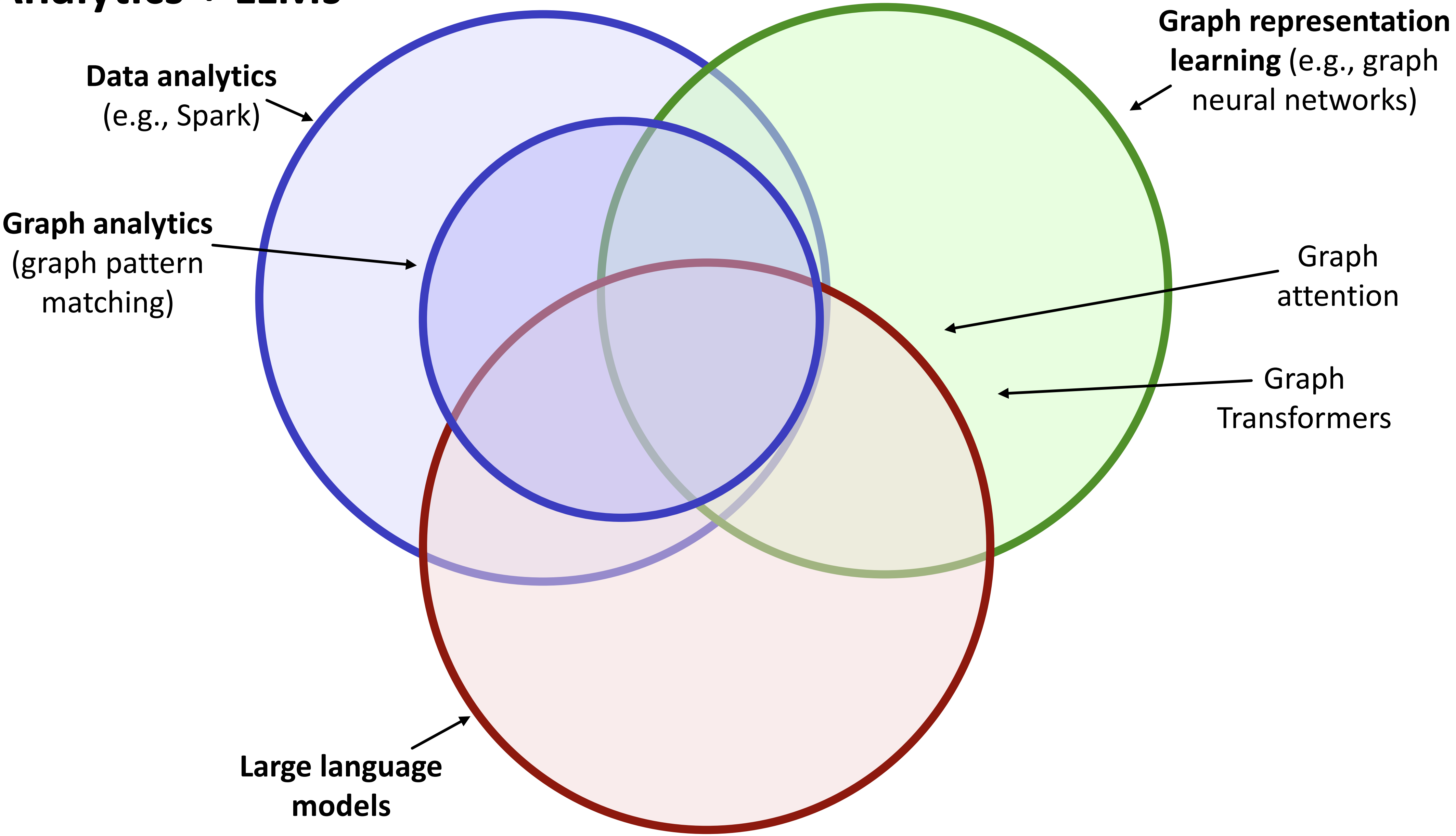


Graph representation learning (e.g., graph neural networks)

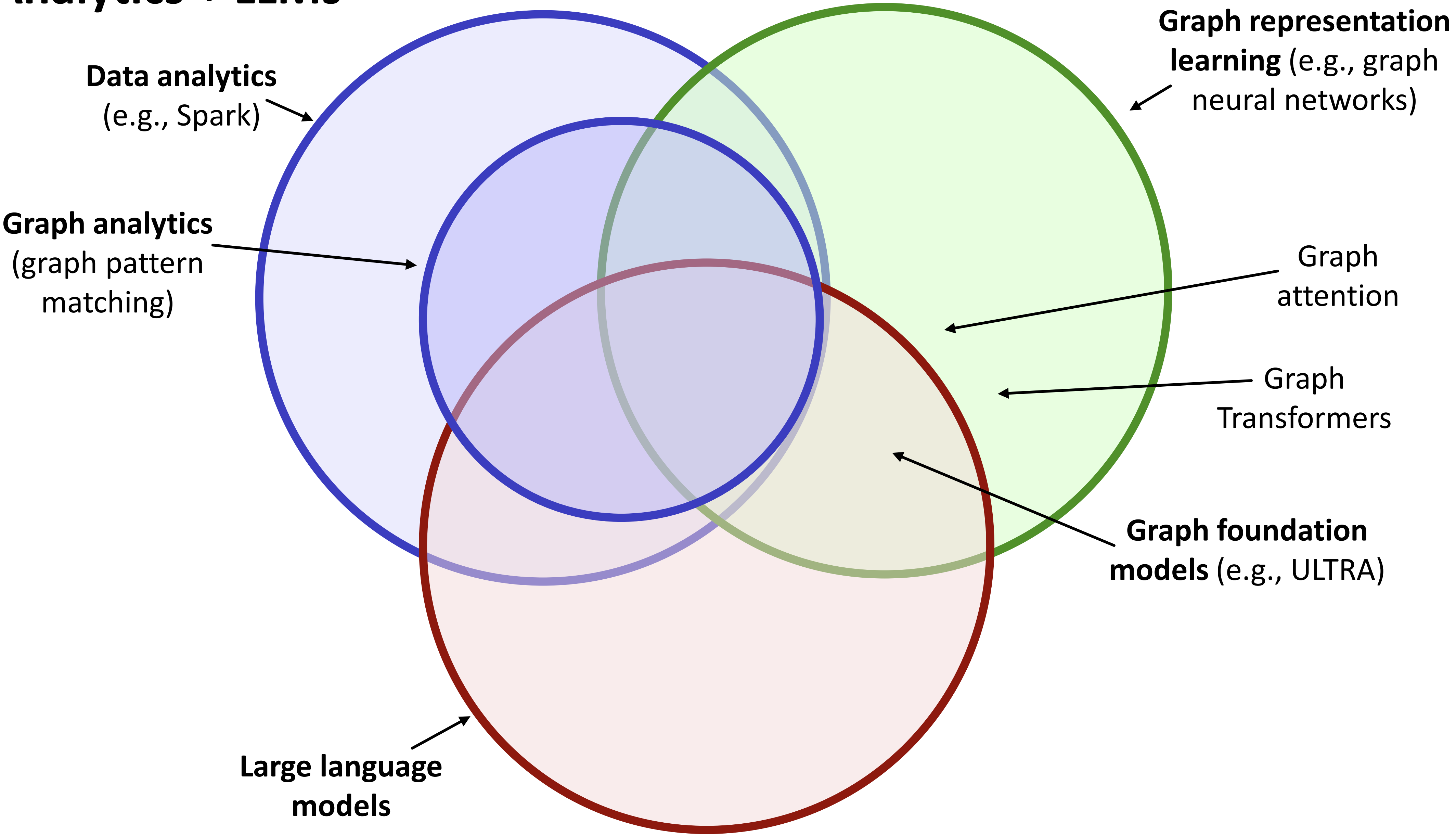
Graph Analytics + LLMs



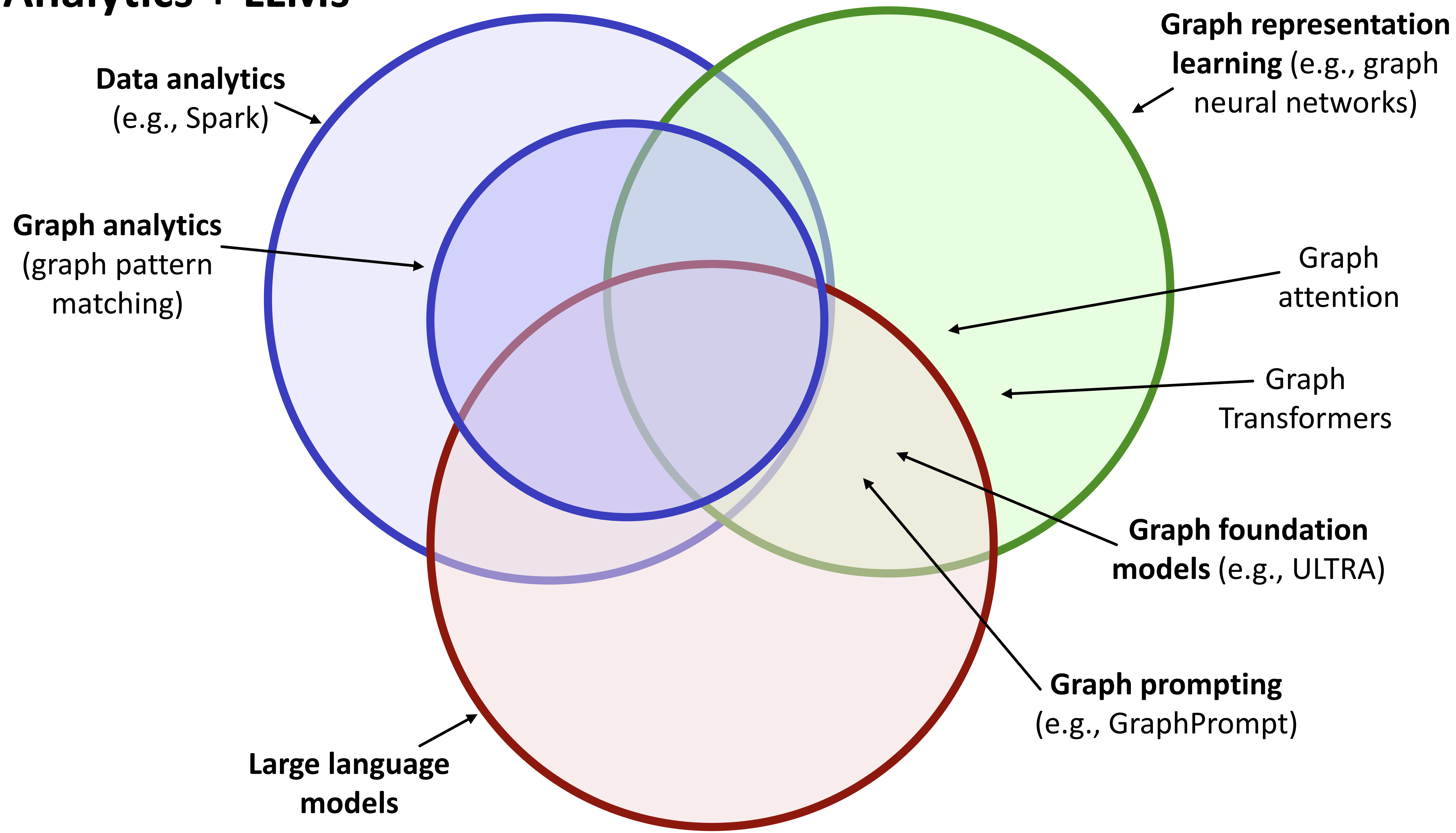
Graph Analytics + LLMs



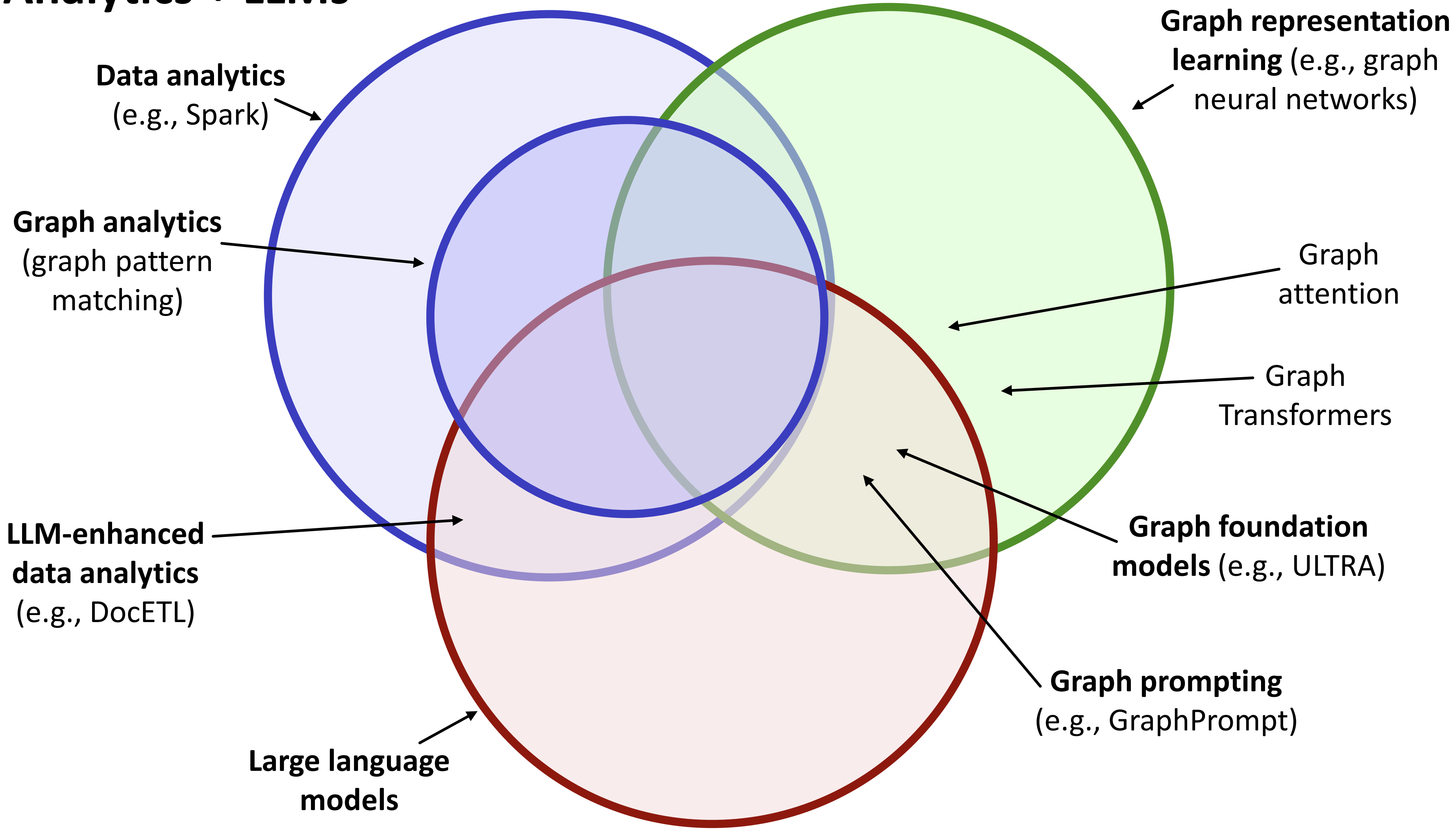
Graph Analytics + LLMs



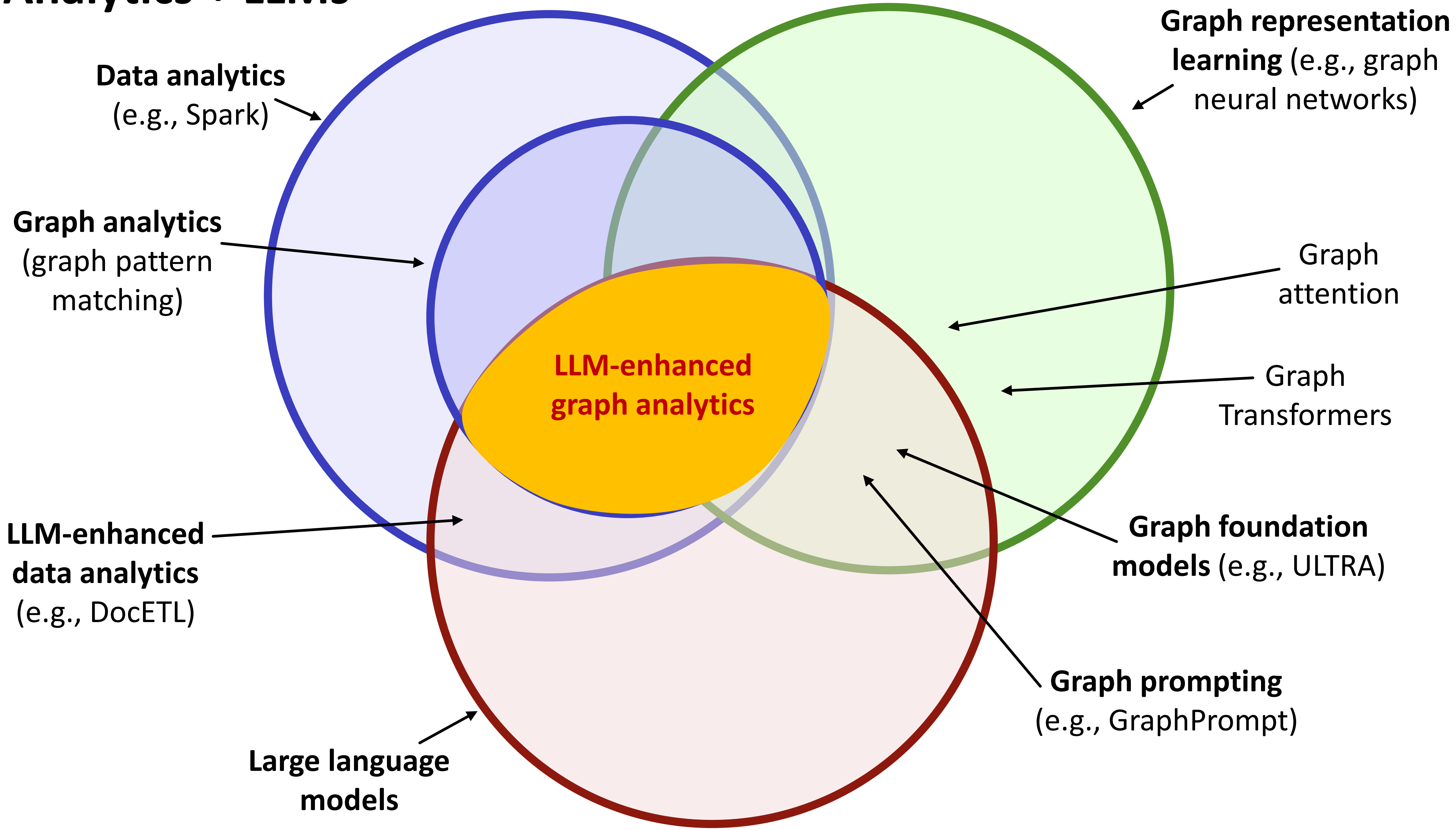
Graph Analytics + LLMs



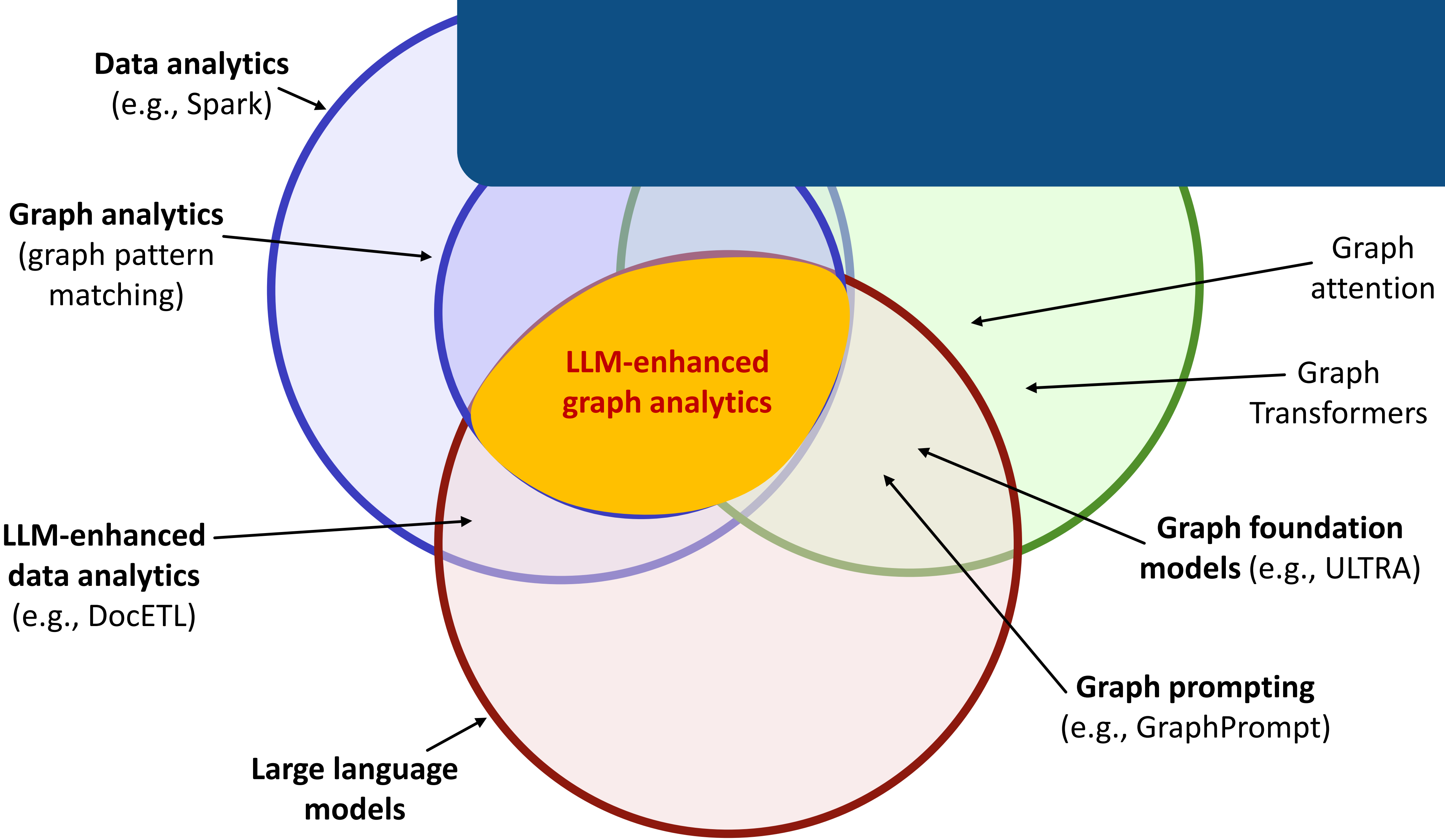
Graph Analytics + LLMs



Graph Analytics + LLMs

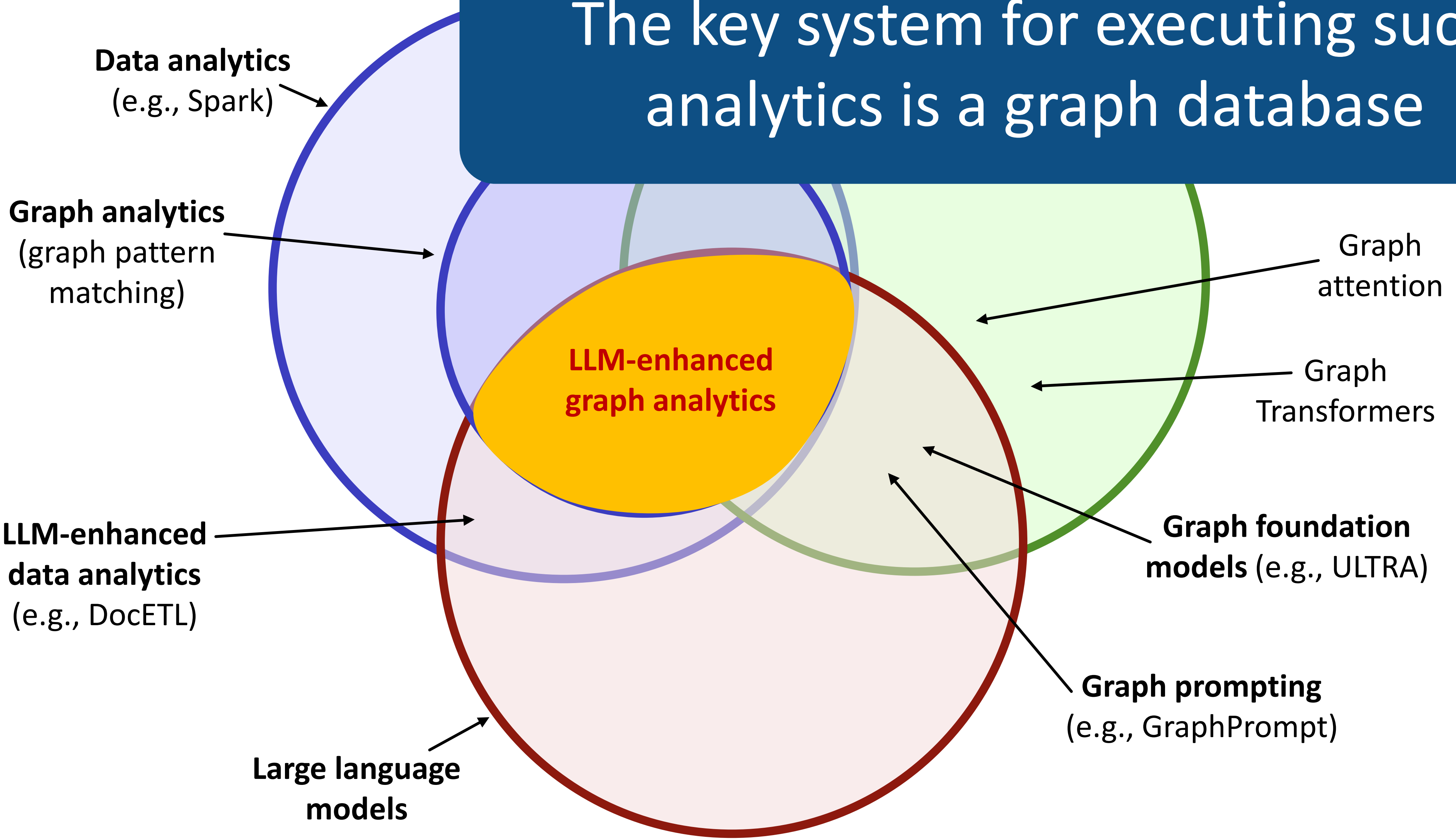


Graph Analytics + LLMs

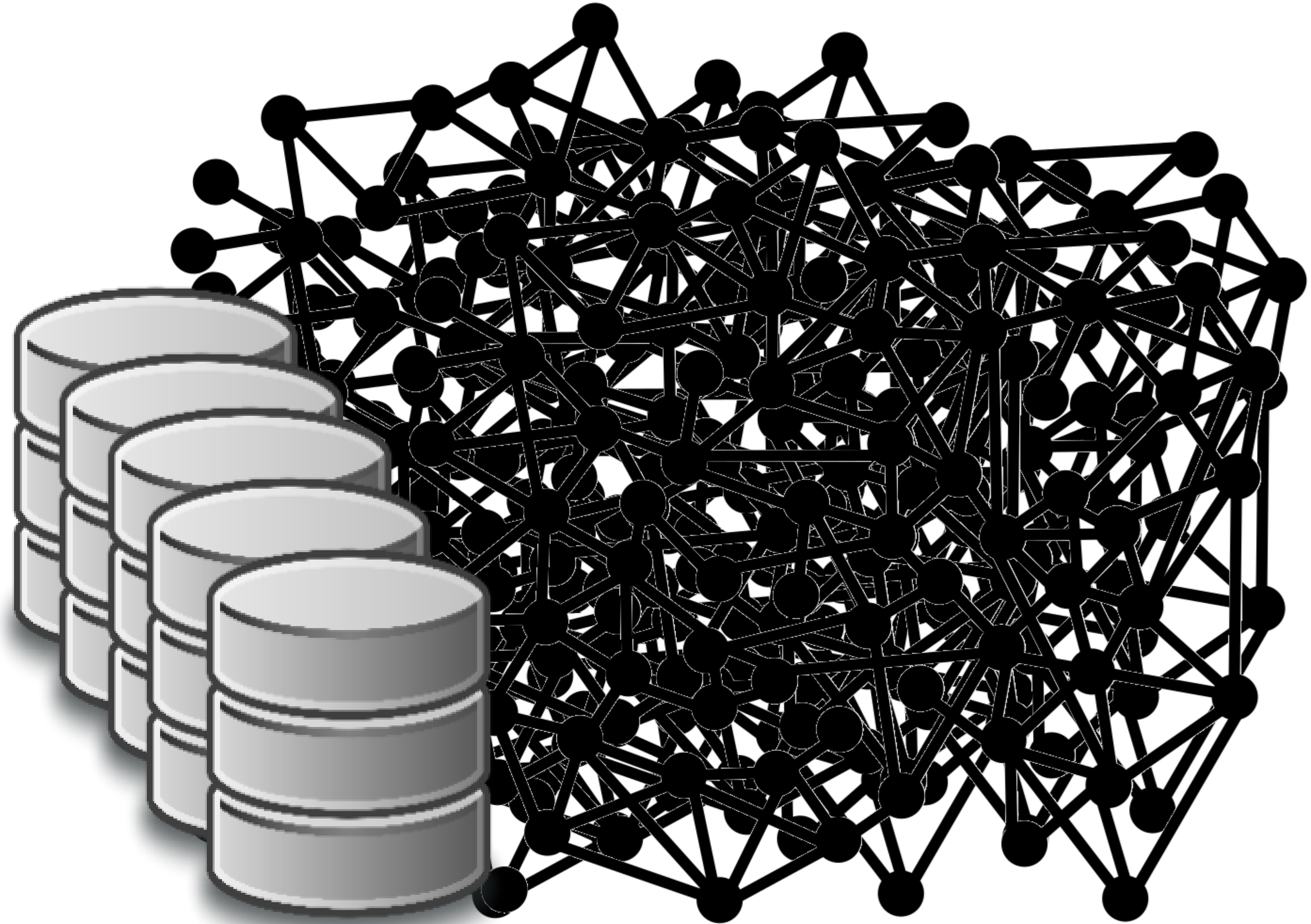


Graph Analytics + LLMs

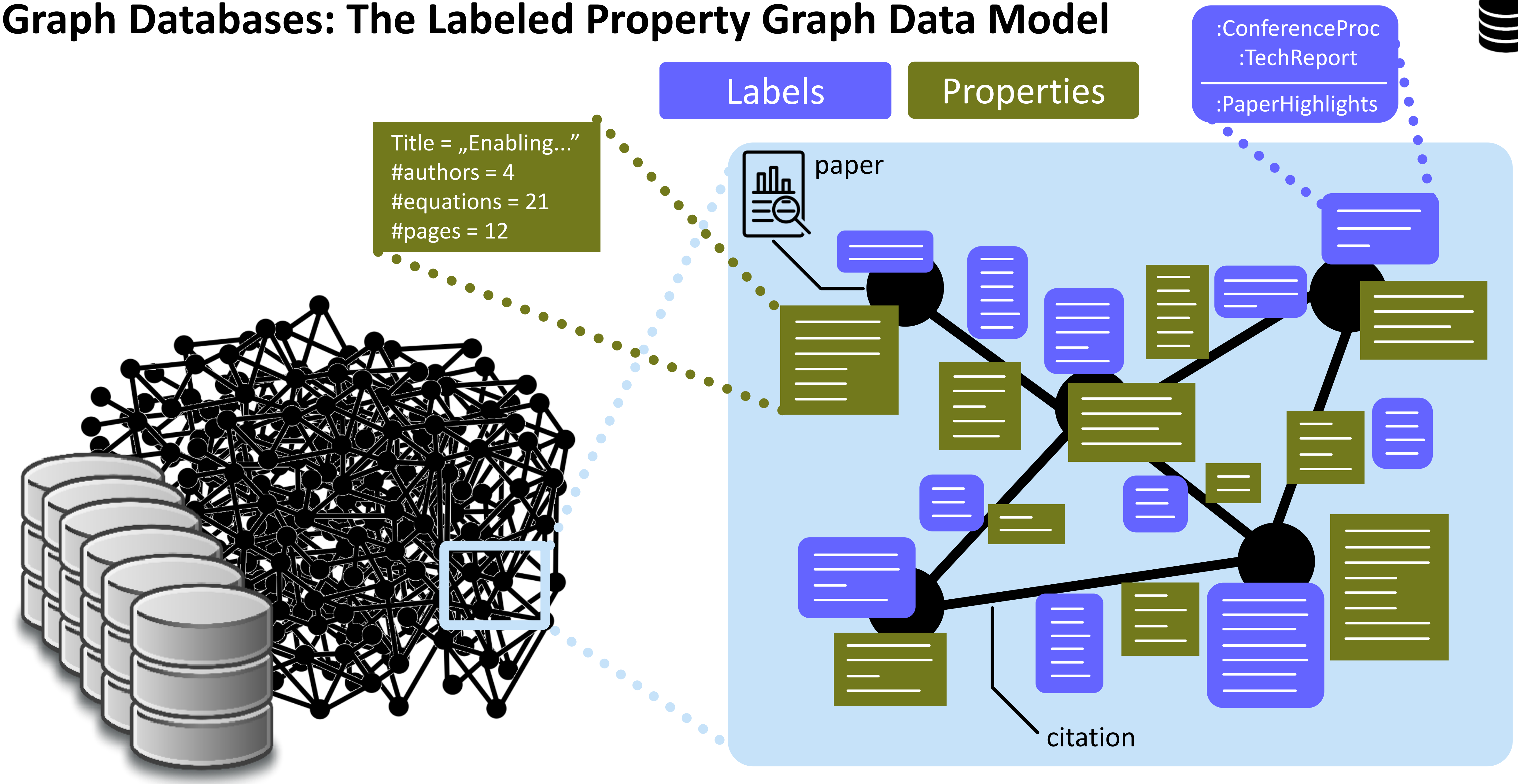
The key system for executing such analytics is a graph database



Graph Databases (GDBs): A Very Brief Introduction



Graph Databases: The Labeled Property Graph Data Model

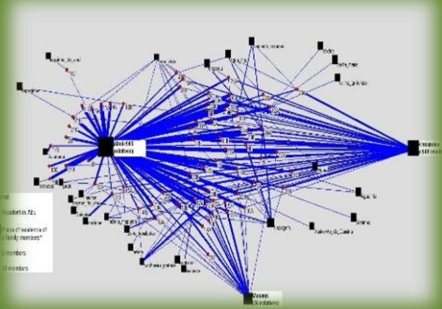


Graph Databases: Where Do We Use Them?

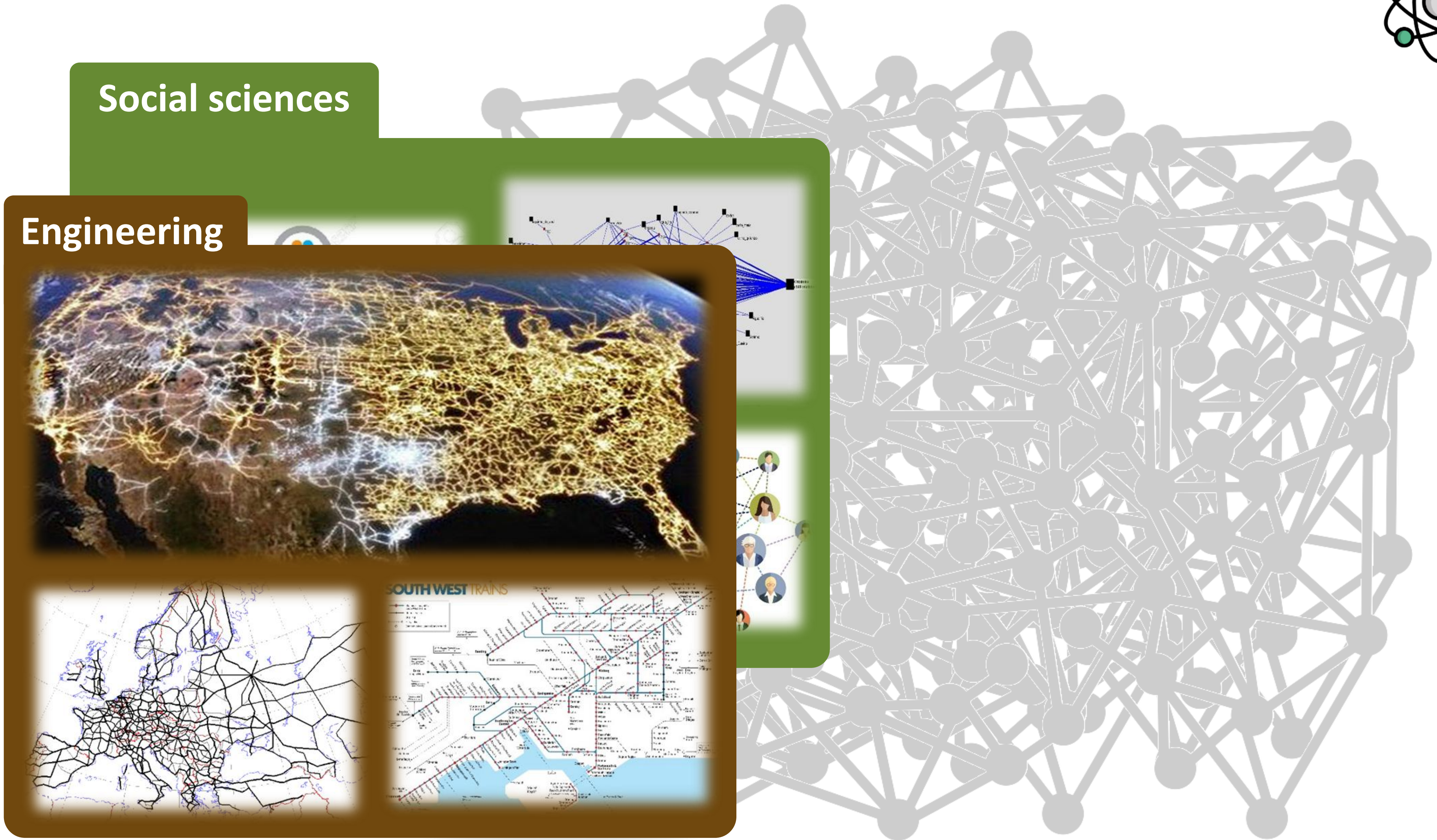
Graph Databases: Where Do We Use Them?

Graph Databases: Where Do We Use Them?

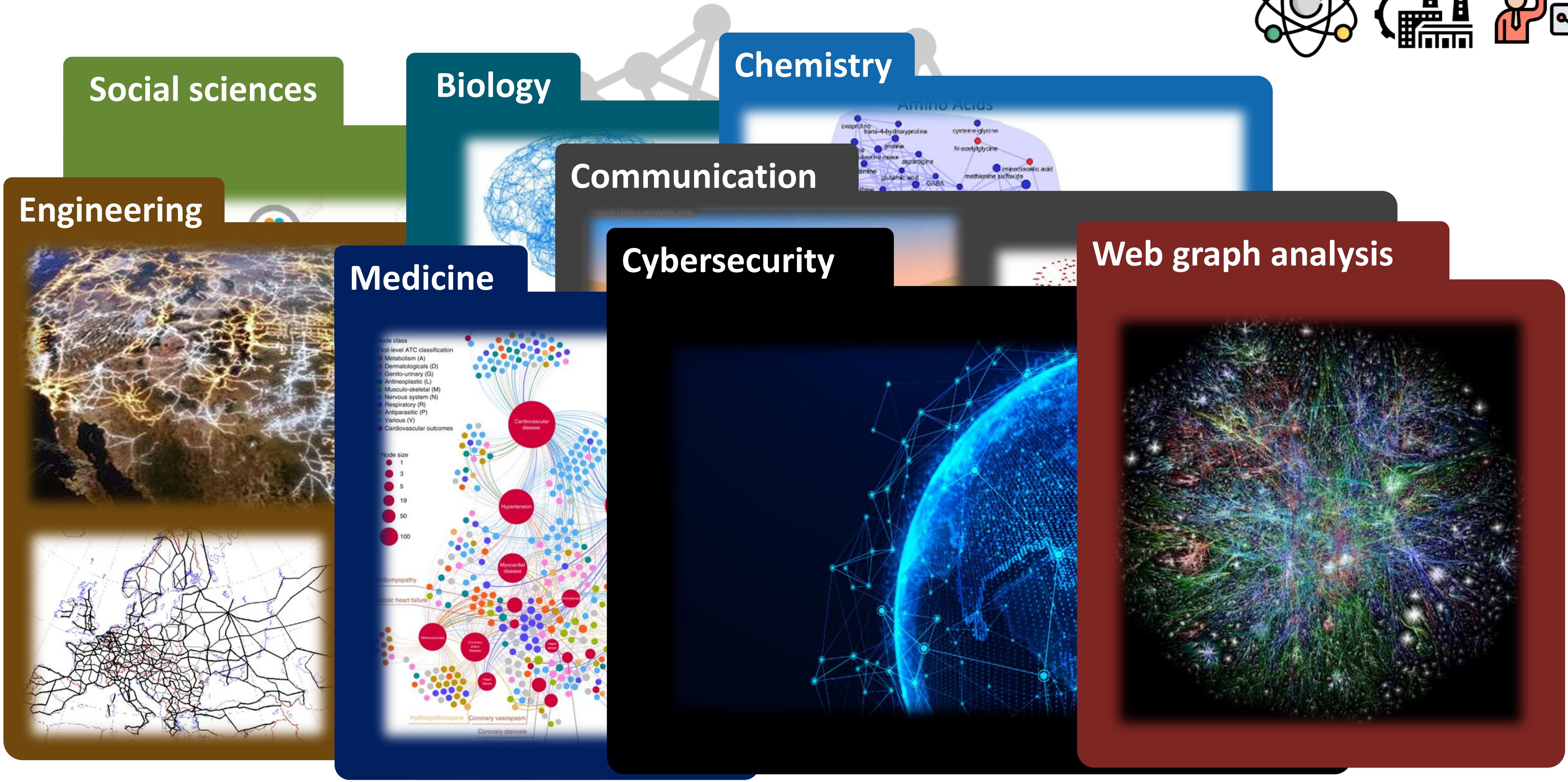
Social sciences



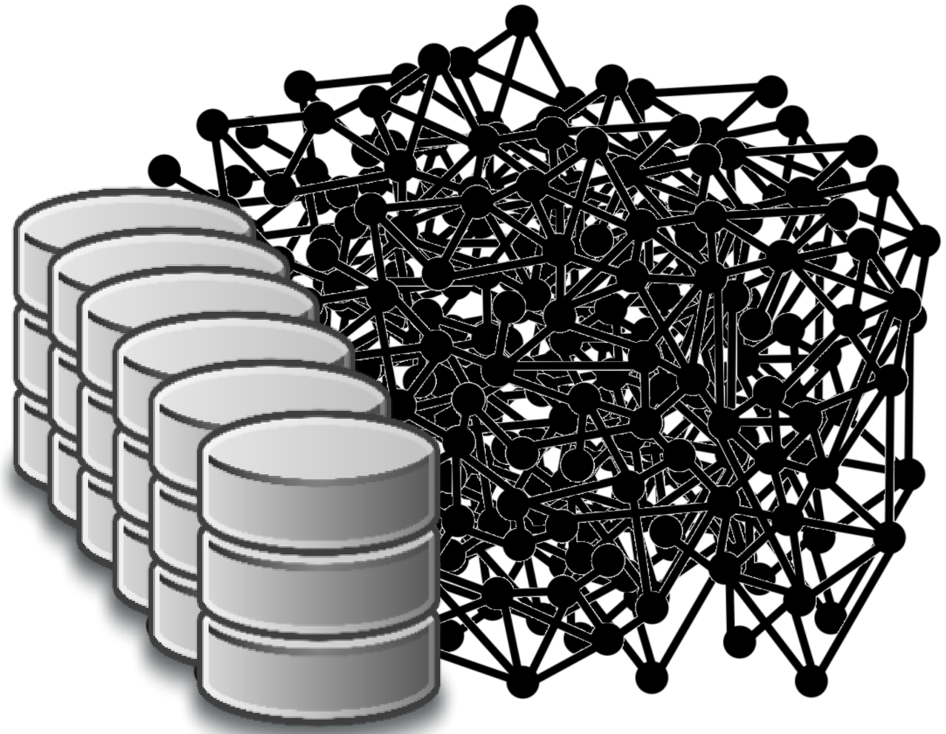
Graph Databases: Where Do We Use Them?



Graph Databases: Where Do We Use Them?



That's why there are lots of them!



That's why there are lots of them!

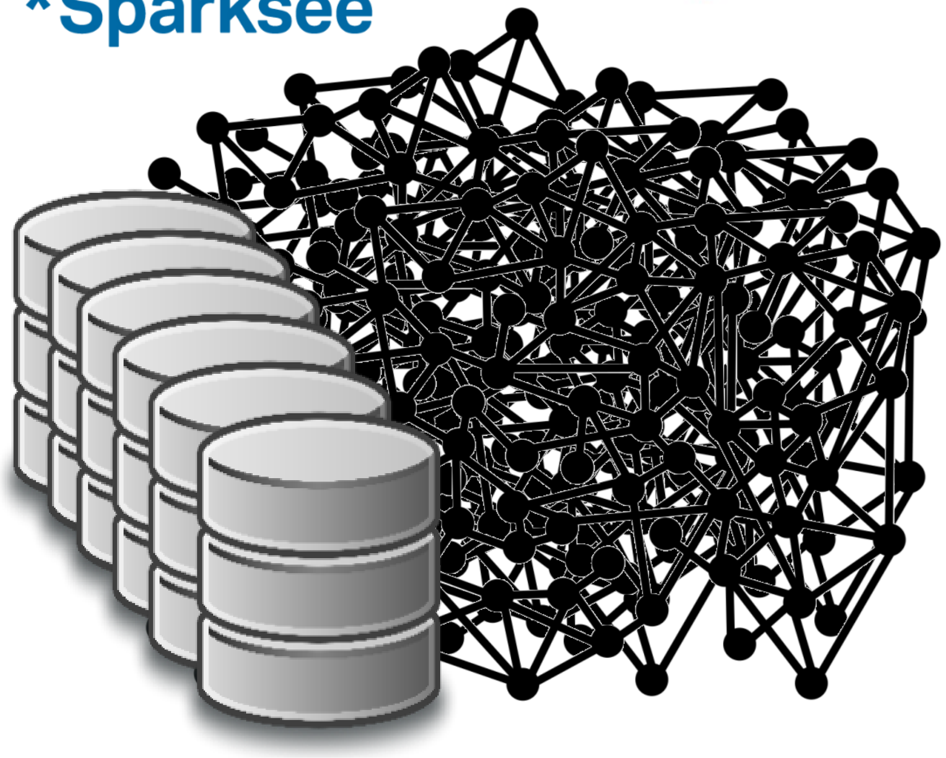
Dgraph STARDOG NebulaGraph ArangoDB

VelocityDB InfiniteGraph

GALAXYBASE AGENS Graph Database HugeGraph ANZO Amazon Neptune fauna

graphbase.ai neo4j BangDB ULTIPA Vaticle TypeDB blazegraph™ GraphDB

*Sparksee ArcadeDB



OrientDB® TITAN

KATANA GRAPH

MEM GRAPH

FlockDB Franz Inc. AllegroGraph TigerGraph

JanusGraph

TerminusDB KÙZU

Setting: From Natural Language to Graph Analytics

Setting: From Natural Language to Graph Analytics

Goal: let a user ask a question in **natural language (NL)** and obtain an answer by querying a **graph database**.

Setting: From Natural Language to Graph Analytics

Goal: let a user ask a question in **natural language (NL)** and obtain an answer by querying a **graph database**.

In the graph setting, this is often called **Text2Cypher**: generating a Cypher query from an NL request, given the graph schema.

Setting: From Natural Language to Graph Analytics

Goal: let a user ask a question in **natural language (NL)** and obtain an answer by querying a **graph database**.

In the graph setting, this is often called **Text2Cypher**: generating a Cypher query from an NL request, given the graph schema.

Setting: From Natural Language to Graph Analytics

Goal: let a user ask a question in **natural language (NL)** and obtain an answer by querying a **graph database**.

In the graph setting, this is often called **Text2Cypher**: generating a Cypher query from an NL request, given the graph schema.

Setting: From Natural Language to Graph Analytics

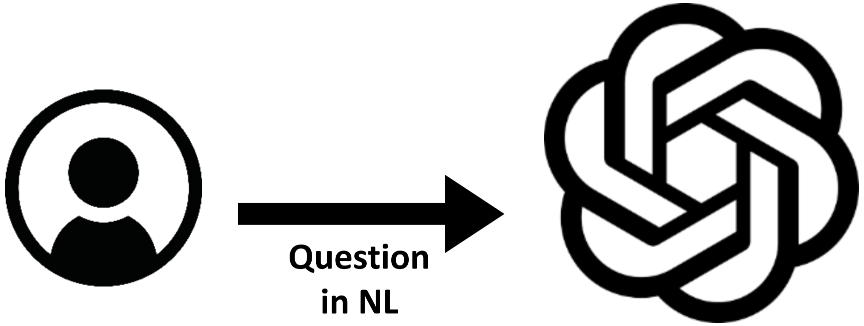
Goal: let a user ask a question in **natural language (NL)** and obtain an answer by querying a **graph database**.

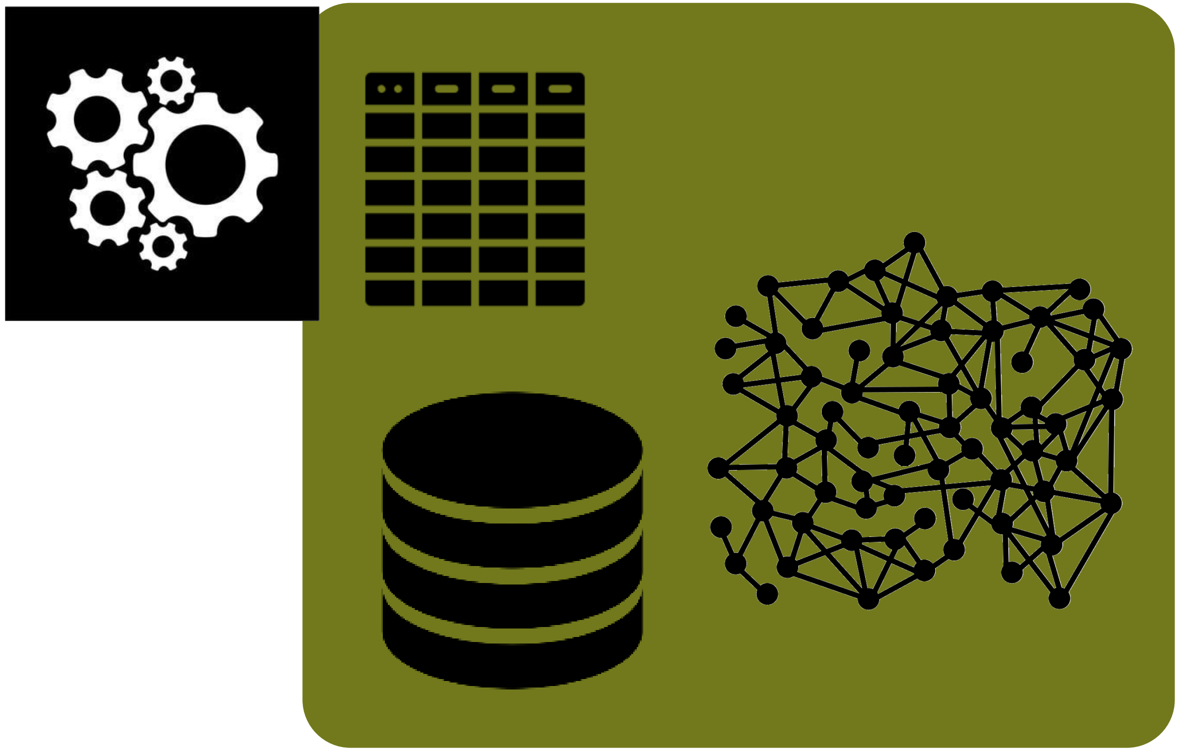
In the graph setting, this is often called **Text2Cypher**: generating a Cypher query from an NL request, given the graph schema.

Setting: From Natural Language to Graph Analytics

Goal: let a user ask a question in **natural language (NL)** and obtain an answer by querying a **graph database**.

In the graph setting, this is often called **Text2Cypher**: generating a Cypher query from an NL request, given the graph schema.

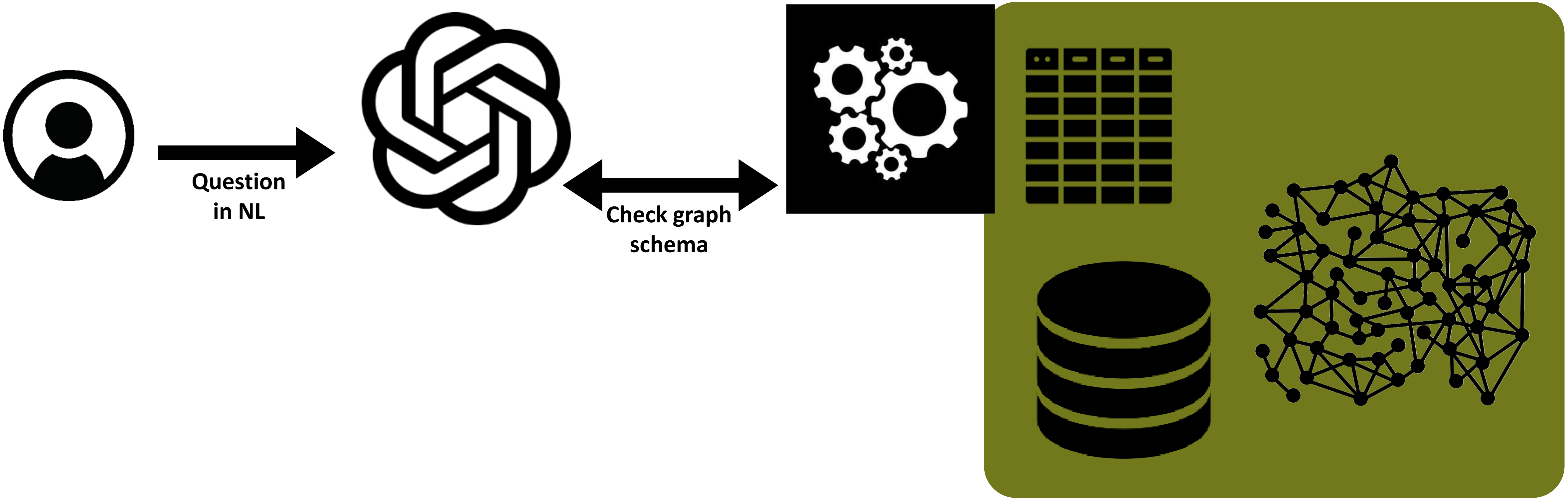




Setting: From Natural Language to Graph Analytics

Goal: let a user ask a question in **natural language (NL)** and obtain an answer by querying a **graph database**.

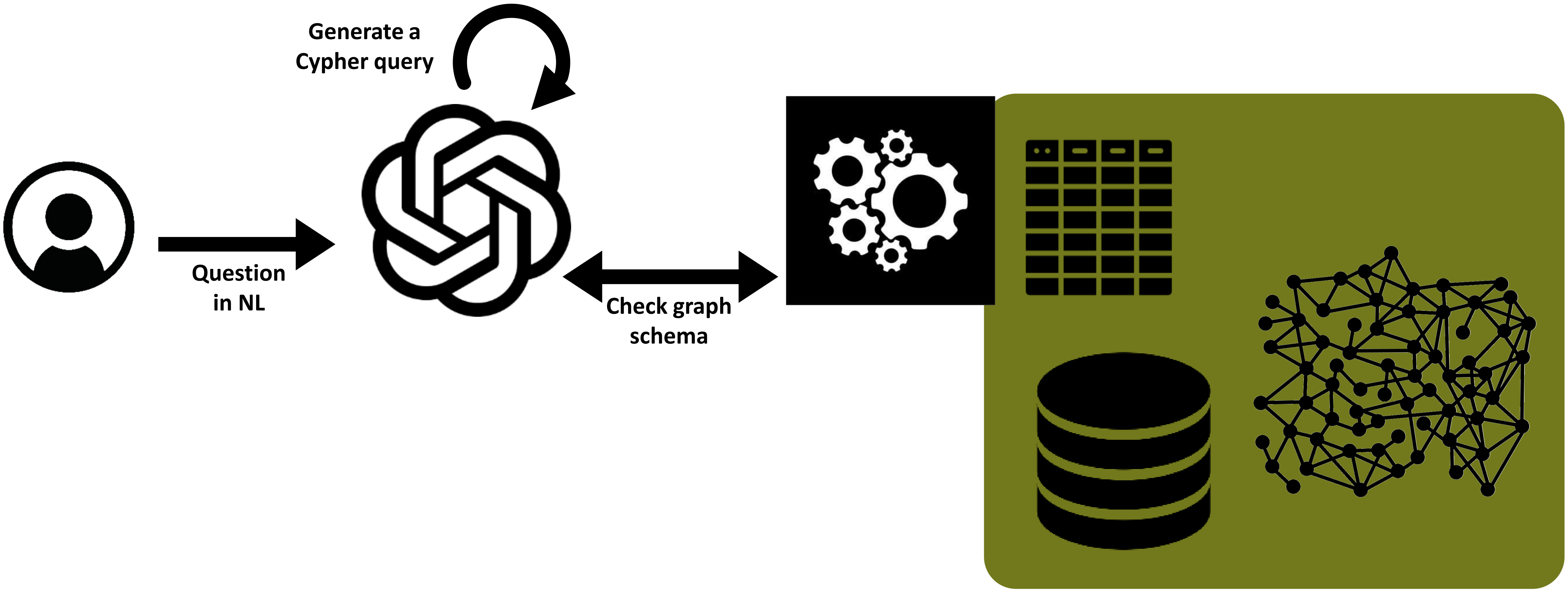
In the graph setting, this is often called **Text2Cypher**: generating a Cypher query from an NL request, given the graph schema.



Setting: From Natural Language to Graph Analytics

Goal: let a user ask a question in **natural language (NL)** and obtain an answer by querying a **graph database**.

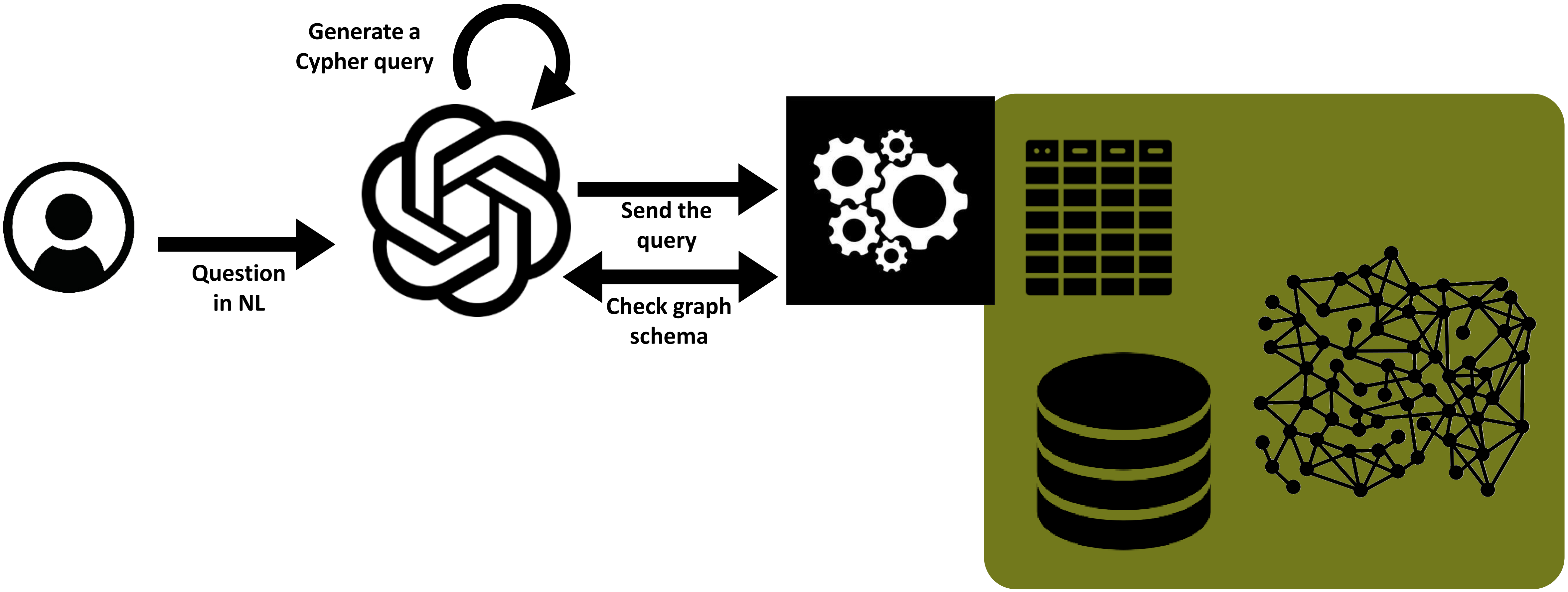
In the graph setting, this is often called **Text2Cypher**: generating a Cypher query from an NL request, given the graph schema.



Setting: From Natural Language to Graph Analytics

Goal: let a user ask a question in **natural language (NL)** and obtain an answer by querying a **graph database**.

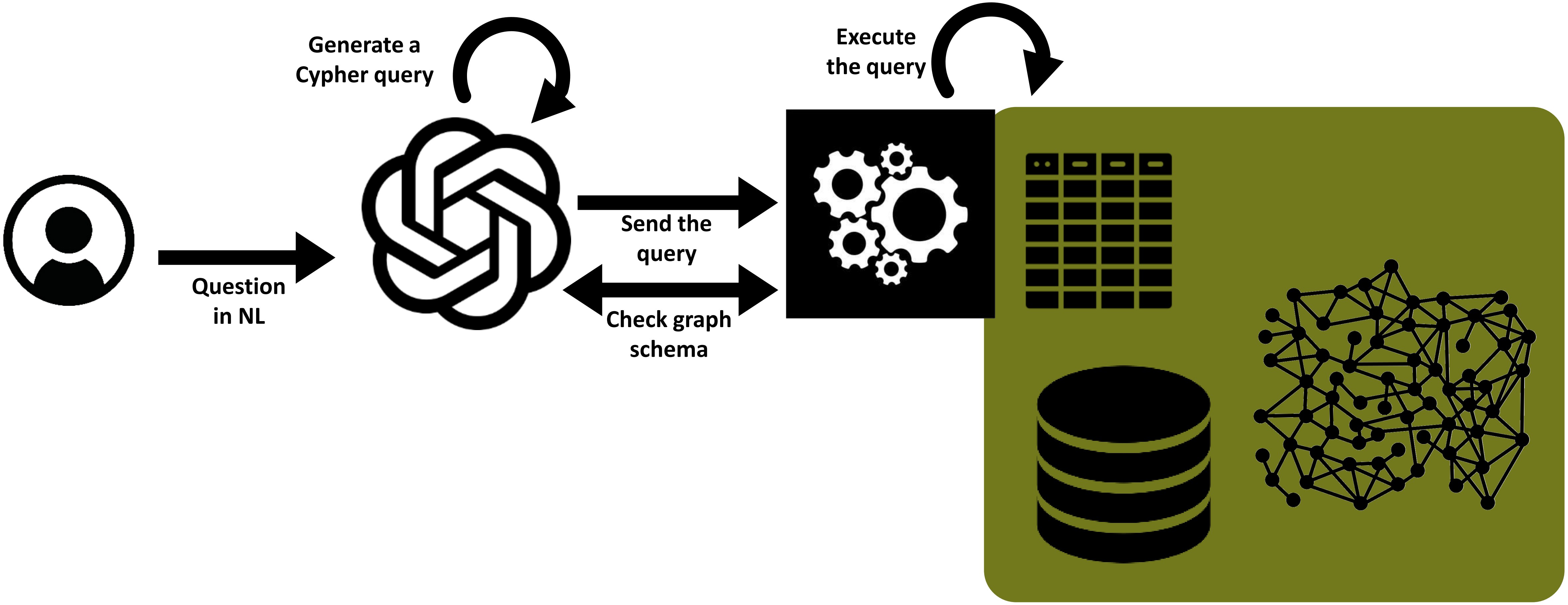
In the graph setting, this is often called **Text2Cypher**: generating a Cypher query from an NL request, given the graph schema.



Setting: From Natural Language to Graph Analytics

Goal: let a user ask a question in **natural language (NL)** and obtain an answer by querying a **graph database**.

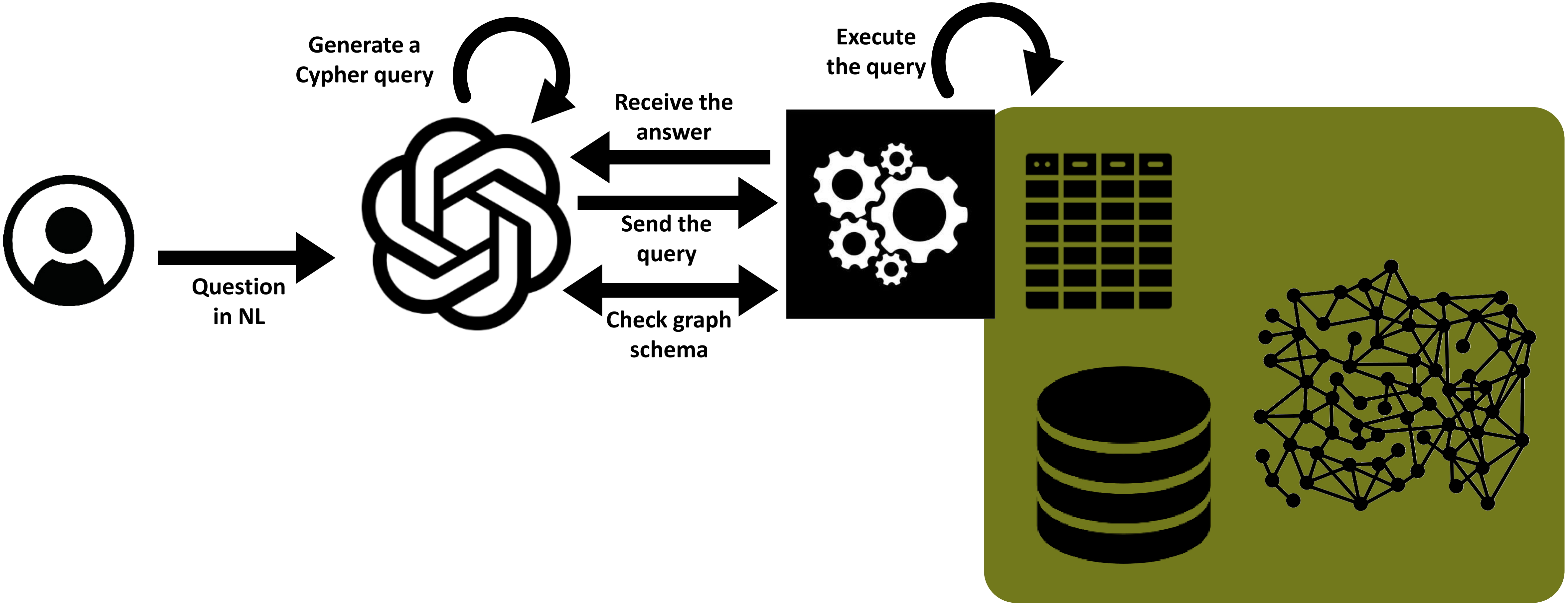
In the graph setting, this is often called **Text2Cypher**: generating a Cypher query from an NL request, given the graph schema.



Setting: From Natural Language to Graph Analytics

Goal: let a user ask a question in **natural language (NL)** and obtain an answer by querying a **graph database**.

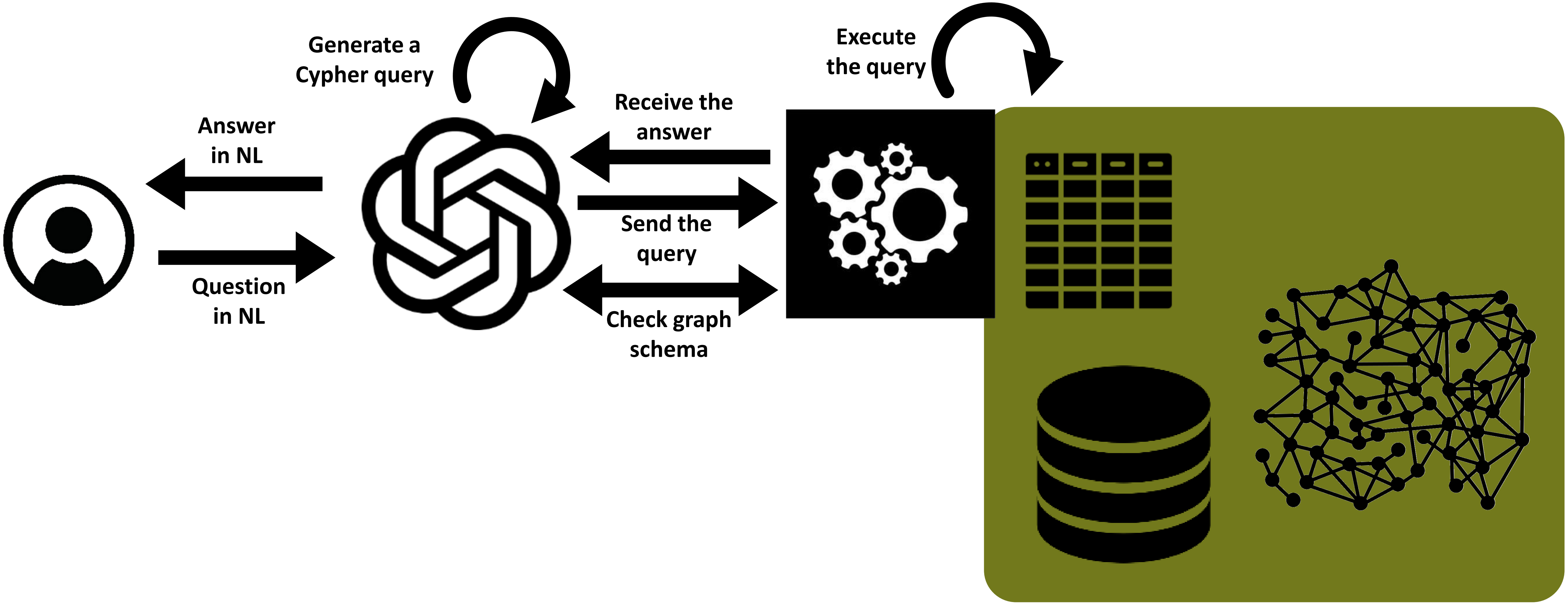
In the graph setting, this is often called **Text2Cypher**: generating a Cypher query from an NL request, given the graph schema.



Setting: From Natural Language to Graph Analytics

Goal: let a user ask a question in **natural language (NL)** and obtain an answer by querying a **graph database**.

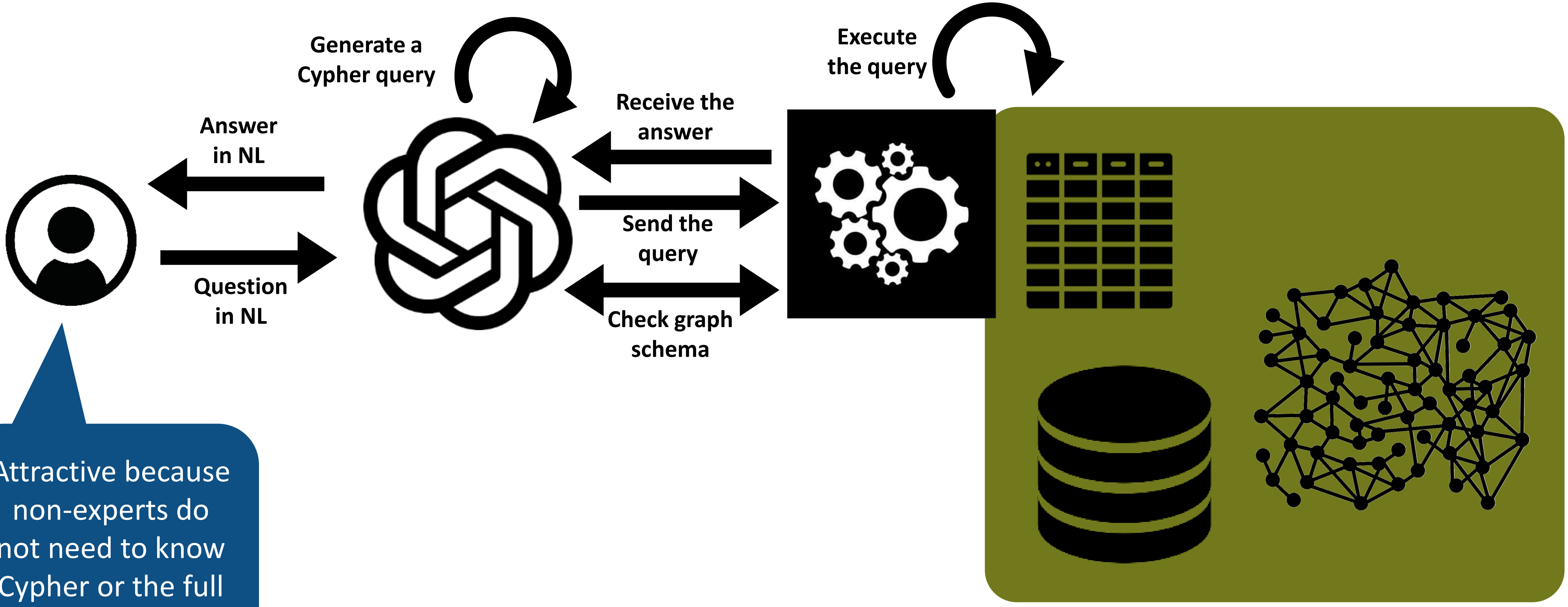
In the graph setting, this is often called **Text2Cypher**: generating a Cypher query from an NL request, given the graph schema.



Setting: From Natural Language to Graph Analytics

Goal: let a user ask a question in **natural language (NL)** and obtain an answer by querying a **graph database**.

In the graph setting, this is often called **Text2Cypher**: generating a Cypher query from an NL request, given the graph schema.

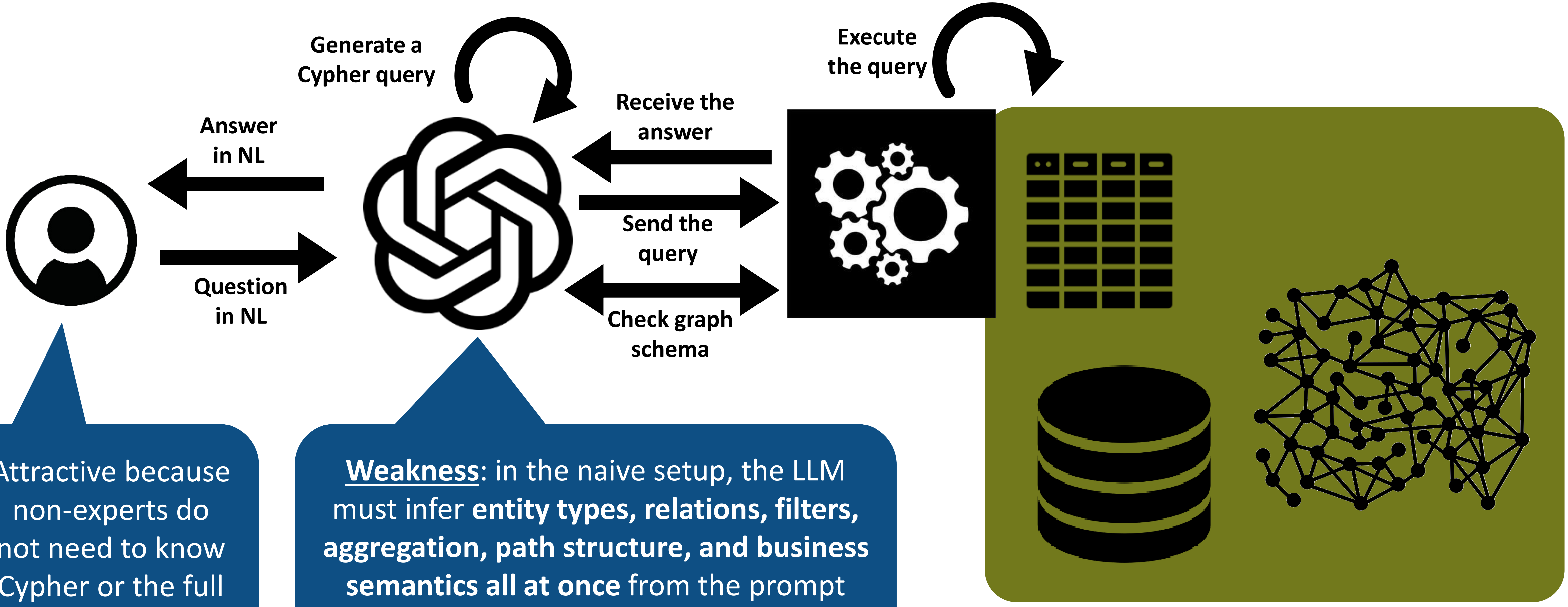


Attractive because non-experts do not need to know Cypher or the full graph schema

Setting: From Natural Language to Graph Analytics

Goal: let a user ask a question in **natural language (NL)** and obtain an answer by querying a **graph database**.

In the graph setting, this is often called **Text2Cypher**: generating a Cypher query from an NL request, given the graph schema.



Attractive because non-experts do not need to know Cypher or the full graph schema

Weakness: in the naive setup, the LLM must infer **entity types, relations, filters, aggregation, path structure, and business semantics all at once** from the prompt and schema alone

Setting: From Natural Language to Graph Analytics

Goal: let a user ask a question in **natural language (NL)** and obtain an answer by querying a **graph database**.

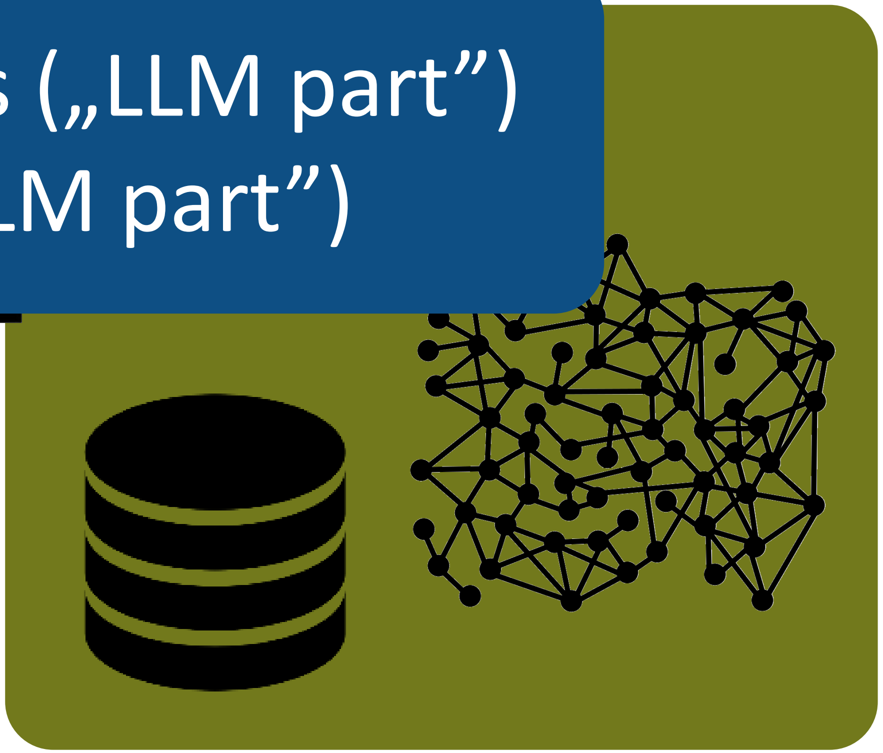
In the graph setting, this is often called **Text2Cypher**: generating a Cypher query from an NL request, given the graph schema.

Generate a Cypher query

Execute the query

in NL

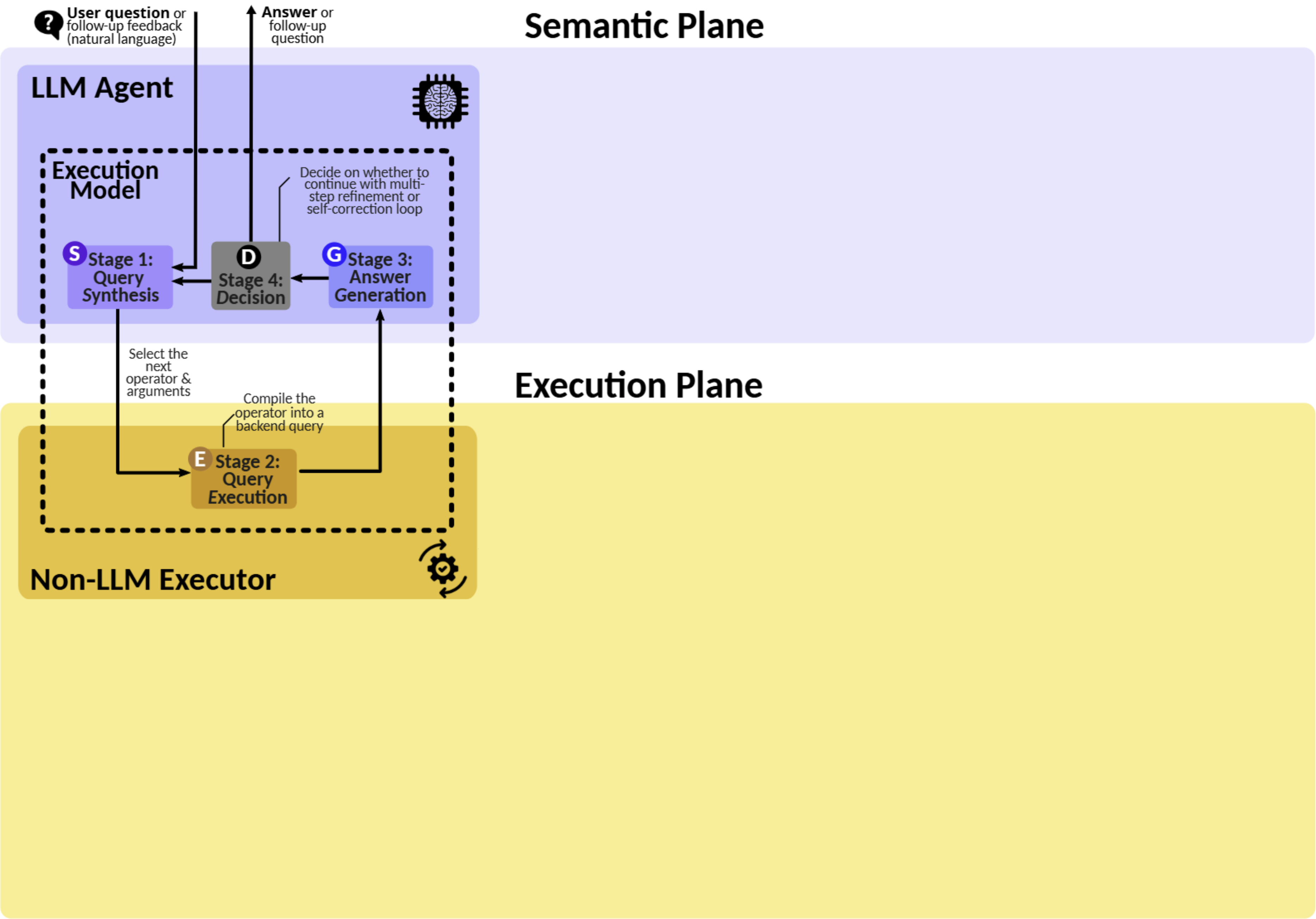
Check graph schema

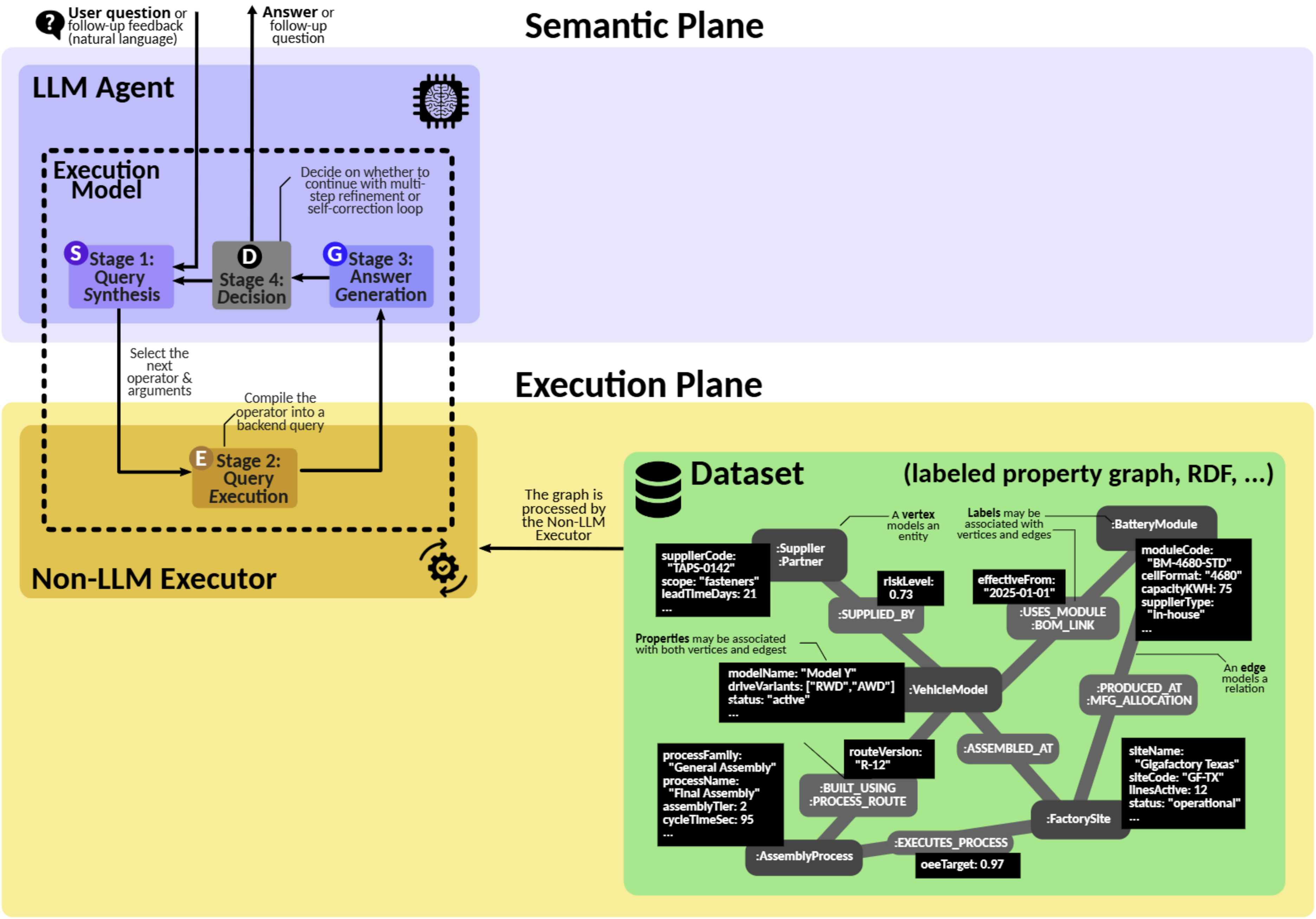


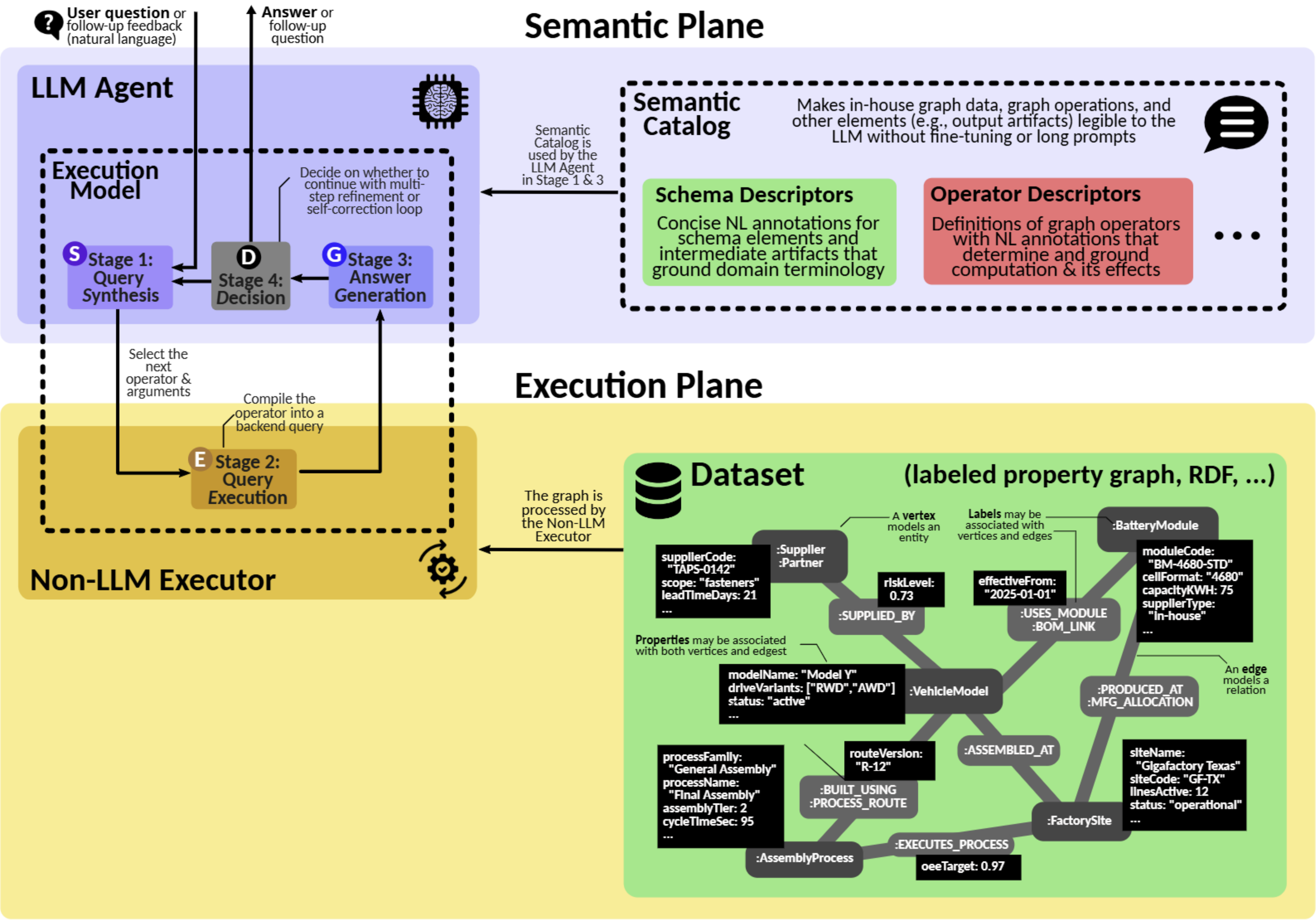
Solution: separate semantics („LLM part") from execution („Non-LLM part")

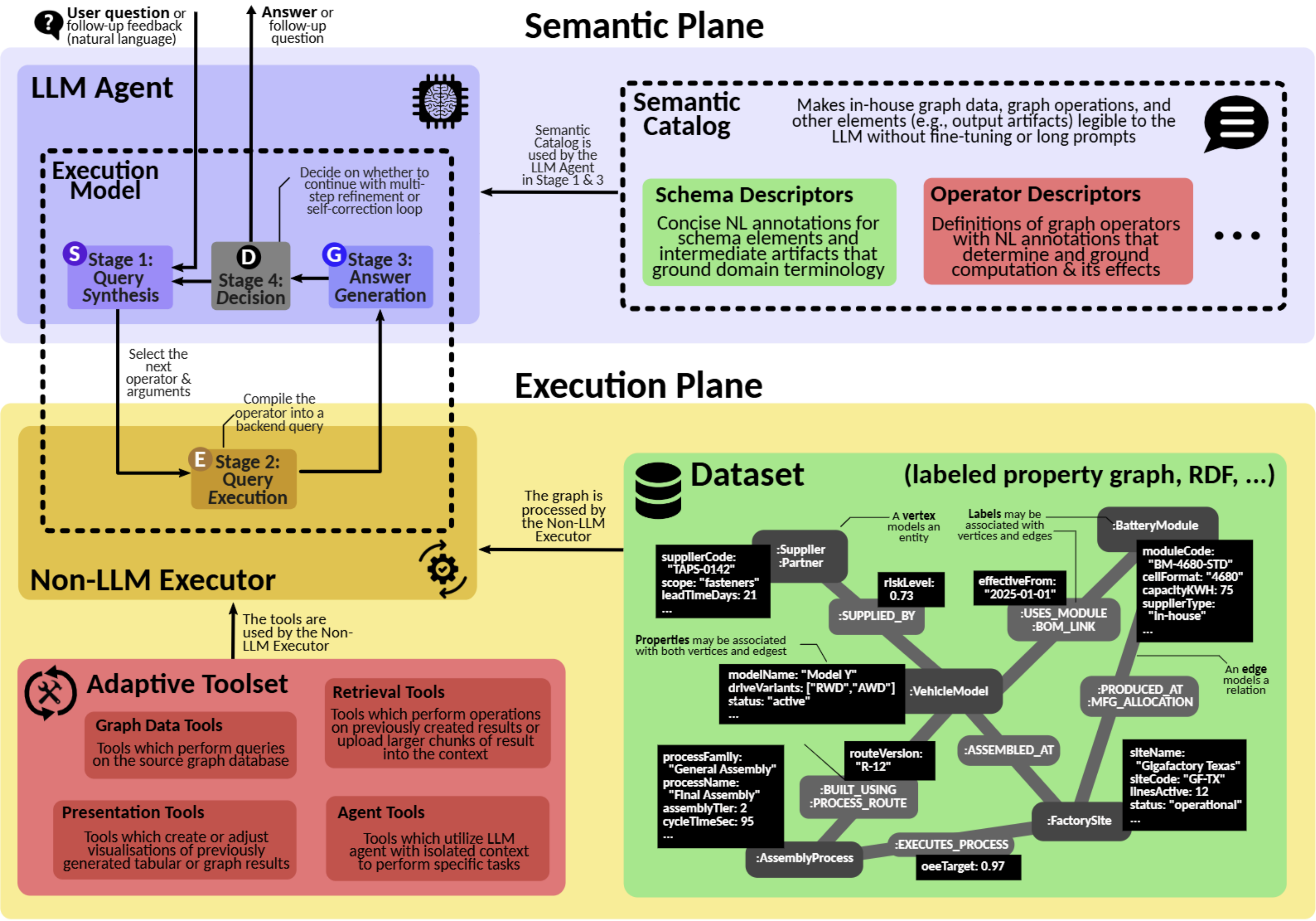
Attractive because non-experts do not need to know Cypher or the full graph schema

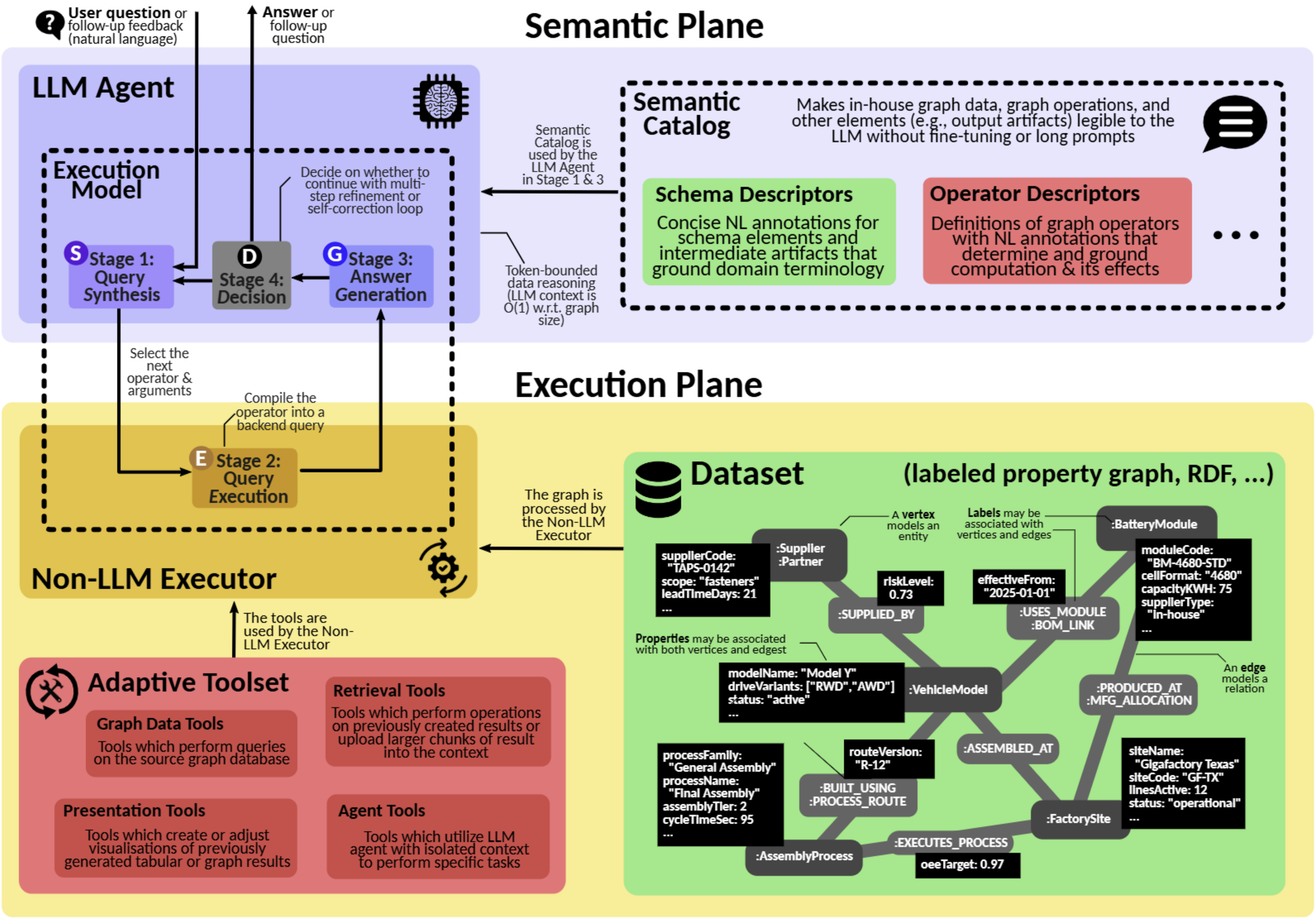
Weakness: in the naive setup, the LLM must infer **entity types, relations, filters, aggregation, path structure, and business semantics all at once** from the prompt and schema alone


User question or follow-up feedback (natural language)
Answer or follow-up question
Semantic Plane
LLM Agent
Execution Model
Decide on whether to continue with multi-step refinement or self-correction loop
S Stage 1: Query Synthesis
D Stage 4: Decision
G Stage 3: Answer Generation
Select the next operator & arguments
Compile the operator into a backend query
Execution Plane
E Stage 2: Query Execution
Non-LLM Executor

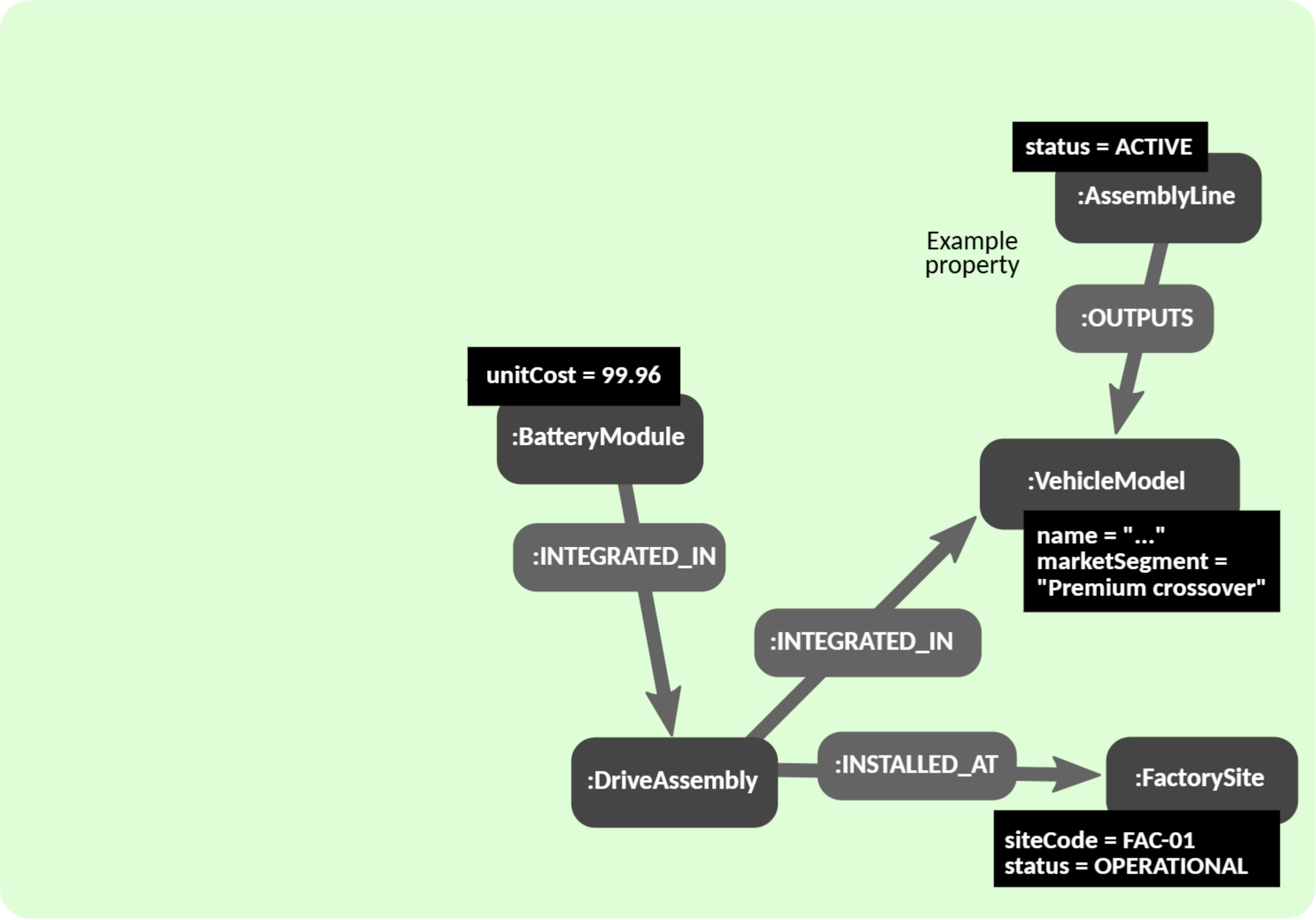

User question or follow-up feedback (natural language)
Answer or follow-up question
Semantic Plane
LLM Agent
Execution Model
Decide on whether to continue with multi-step refinement or self-correction loop
S Stage 1: Query Synthesis
D Stage 4: Decision
G Stage 3: Answer Generation
Select the next operator & arguments
Compile the operator into a backend query
Execution Plane
E Stage 2: Query Execution
Non-LLM Executor
The graph is processed by the Non-LLM Executor
Dataset
(labeled property graph, RDF, ...)
A vertex models an entity
Labels may be associated with vertices and edges
:BatteryModule
moduleCode: "BM-4680-STD" cellFormat: "4680" capacityKWH: 75 supplierType: "in-house" ...
:Supplier :Partner
supplierCode: "TAPS-0142" scope: "fasteners" leadTimeDays: 21 ...
riskLevel: 0.73
:SUPPLIED_BY
effectiveFrom: "2025-01-01"
:USES_MODULE :BOM_LINK
Properties may be associated with both vertices and edgest
modelName: "Model Y" driveVariants: ["RWD","AWD"] status: "active" ...
:VehicleModel
An edge models a relation
:PRODUCED_AT :MFG_ALLOCATION
routeVersion: "R-12"
:ASSEMBLED_AT
processFamily: "General Assembly" processName: "Final Assembly" assemblyTier: 2 cycleTimeSec: 95 ...
:BUILT_USING :PROCESS_ROUTE
siteName: "Gigafactory Texas" siteCode: "GF-TX" linesActive: 12 status: "operational" ...
:FactorySite
:EXECUTES_PROCESS
oeeTarget: 0.97
:AssemblyProcess


Semantic Plane
User question or follow-up feedback (natural language)
Answer or follow-up question
LLM Agent
Execution Model
Decide on whether to continue with multi-step refinement or self-correction loop
S Stage 1: Query Synthesis
D Stage 4: Decision
G Stage 3: Answer Generation
Semantic Catalog is used by the LLM Agent in Stage 1 & 3
Semantic Catalog
Makes in-house graph data, graph operations, and other elements (e.g., output artifacts) legible to the LLM without fine-tuning or long prompts
Schema Descriptors
Concise NL annotations for schema elements and intermediate artifacts that ground domain terminology
Operator Descriptors
Definitions of graph operators with NL annotations that determine and ground computation & its effects
...
Select the next operator & arguments
Compile the operator into a backend query
Execution Plane
E Stage 2: Query Execution
Non-LLM Executor
The graph is processed by the Non-LLM Executor
Dataset
(labeled property graph, RDF, ...)
A vertex models an entity
Labels may be associated with vertices and edges
:BatteryModule
moduleCode: "BM-4680-STD" cellFormat: "4680" capacityKWH: 75 supplierType: "in-house" ...
:Supplier :Partner
supplierCode: "TAPS-0142" scope: "fasteners" leadTimeDays: 21 ...
riskLevel: 0.73
:SUPPLIED_BY
effectiveFrom: "2025-01-01"
:USES_MODULE :BOM_LINK
Properties may be associated with both vertices and edgest
modelName: "Model Y" driveVariants: ["RWD","AWD"] status: "active" ...
:VehicleModel
An edge models a relation
:PRODUCED_AT :MFG_ALLOCATION
routeVersion: "R-12"
:ASSEMBLED_AT
processFamily: "General Assembly" processName: "Final Assembly" assemblyTier: 2 cycleTimeSec: 95 ...
:BUILT_USING :PROCESS_ROUTE
siteName: "Gigafactory Texas" siteCode: "GF-TX" linesActive: 12 status: "operational" ...
:FactorySite
:EXECUTES_PROCESS
:AssemblyProcess
oeeTarget: 0.97

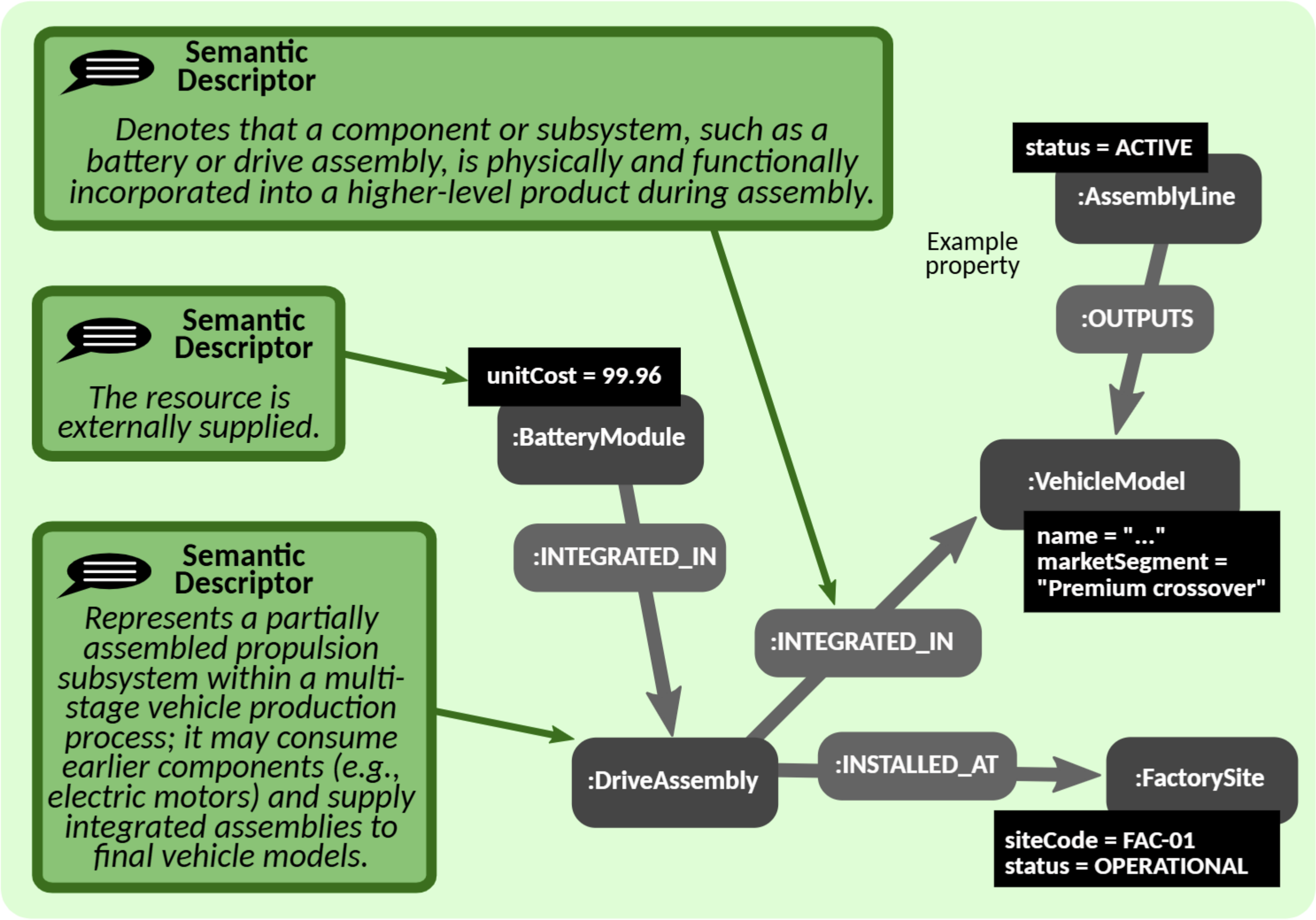

User question or follow-up feedback (natural language)
Answer or follow-up question
Semantic Plane
LLM Agent
Execution Model
Decide on whether to continue with multi-step refinement or self-correction loop
S Stage 1: Query Synthesis
D Stage 4: Decision
G Stage 3: Answer Generation
Semantic Catalog is used by the LLM Agent in Stage 1 & 3
Semantic Catalog
Makes in-house graph data, graph operations, and other elements (e.g., output artifacts) legible to the LLM without fine-tuning or long prompts
Schema Descriptors
Concise NL annotations for schema elements and intermediate artifacts that ground domain terminology
Operator Descriptors
Definitions of graph operators with NL annotations that determine and ground computation & its effects
...
Select the next operator & arguments
Compile the operator into a backend query
Execution Plane
E Stage 2: Query Execution
Non-LLM Executor
The graph is processed by the Non-LLM Executor
The tools are used by the Non-LLM Executor
Adaptive Toolset
Graph Data Tools
Tools which perform queries on the source graph database
Retrieval Tools
Tools which perform operations on previously created results or upload larger chunks of result into the context
Presentation Tools
Tools which create or adjust visualisations of previously generated tabular or graph results
Agent Tools
Tools which utilize LLM agent with isolated context to perform specific tasks
Dataset
(labeled property graph, RDF, ...)
A vertex models an entity
Labels may be associated with vertices and edges
:BatteryModule
:Supplier :Partner
supplierCode: "TAPS-0142" scope: "fasteners" leadTimeDays: 21 ...
riskLevel: 0.73
effectiveFrom: "2025-01-01"
moduleCode: "BM-4680-STD" cellFormat: "4680" capacityKWH: 75 supplierType: "in-house" ...
:SUPPLIED_BY
:USES_MODULE :BOM_LINK
Properties may be associated with both vertices and edgest
modelName: "Model Y" driveVariants: ["RWD","AWD"] status: "active" ...
:VehicleModel
:PRODUCED_AT :MFG_ALLOCATION
An edge models a relation
processFamily: "General Assembly" processName: "Final Assembly" assemblyTier: 2 cycleTimeSec: 95 ...
routeVersion: "R-12"
:ASSEMBLED_AT
:BUILT_USING :PROCESS_ROUTE
siteName: "Gigafactory Texas" siteCode: "GF-TX" linesActive: 12 status: "operational" ...
:FactorySite
:EXECUTES_PROCESS
:AssemblyProcess
oeeTarget: 0.97


User question or follow-up feedback (natural language)
Answer or follow-up question
Semantic Plane
LLM Agent
Execution Model
Decide on whether to continue with multi-step refinement or self-correction loop
S Stage 1: Query Synthesis
D Stage 4: Decision
G Stage 3: Answer Generation
Semantic Catalog is used by the LLM Agent in Stage 1 & 3
Token-bounded data reasoning (LLM context is O(1) w.r.t. graph size)
Semantic Catalog
Makes in-house graph data, graph operations, and other elements (e.g., output artifacts) legible to the LLM without fine-tuning or long prompts
Schema Descriptors
Concise NL annotations for schema elements and intermediate artifacts that ground domain terminology
Operator Descriptors
Definitions of graph operators with NL annotations that determine and ground computation & its effects
• • •
Select the next operator & arguments
Compile the operator into a backend query
Execution Plane
E Stage 2: Query Execution
Non-LLM Executor
The graph is processed by the Non-LLM Executor
The tools are used by the Non-LLM Executor
Adaptive Toolset
Graph Data Tools
Tools which perform queries on the source graph database
Retrieval Tools
Tools which perform operations on previously created results or upload larger chunks of result into the context
Presentation Tools
Tools which create or adjust visualisations of previously generated tabular or graph results
Agent Tools
Tools which utilize LLM agent with isolated context to perform specific tasks
Dataset
(labeled property graph, RDF, ...)
A vertex models an entity
Labels may be associated with vertices and edges
:BatteryModule
:Supplier :Partner
supplierCode: "TAPS-0142" scope: "fasteners" leadTimeDays: 21 ...
riskLevel: 0.73
:SUPPLIED_BY
effectiveFrom: "2025-01-01"
:USES_MODULE :BOM_LINK
moduleCode: "BM-4680-STD" cellFormat: "4680" capacityKWH: 75 supplierType: "in-house" ...
Properties may be associated with both vertices and edgest
modelName: "Model Y" driveVariants: ["RWD","AWD"] status: "active" ...
:VehicleModel
:PRODUCED_AT :MFG_ALLOCATION
An edge models a relation
processFamily: "General Assembly" processName: "Final Assembly" assemblyTier: 2 cycleTimeSec: 95 ...
routeVersion: "R-12"
:BUILT_USING :PROCESS_ROUTE
:ASSEMBLED_AT
siteName: "Gigafactory Texas" siteCode: "GF-TX" linesActive: 12 status: "operational" ...
:FactorySite
:EXECUTES_PROCESS
:AssemblyProcess
oeeTarget: 0.97

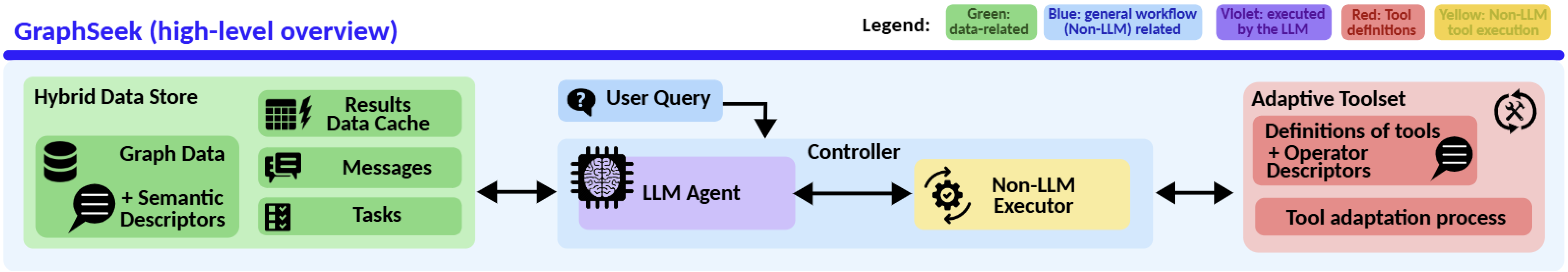
Example Semantic Descriptors



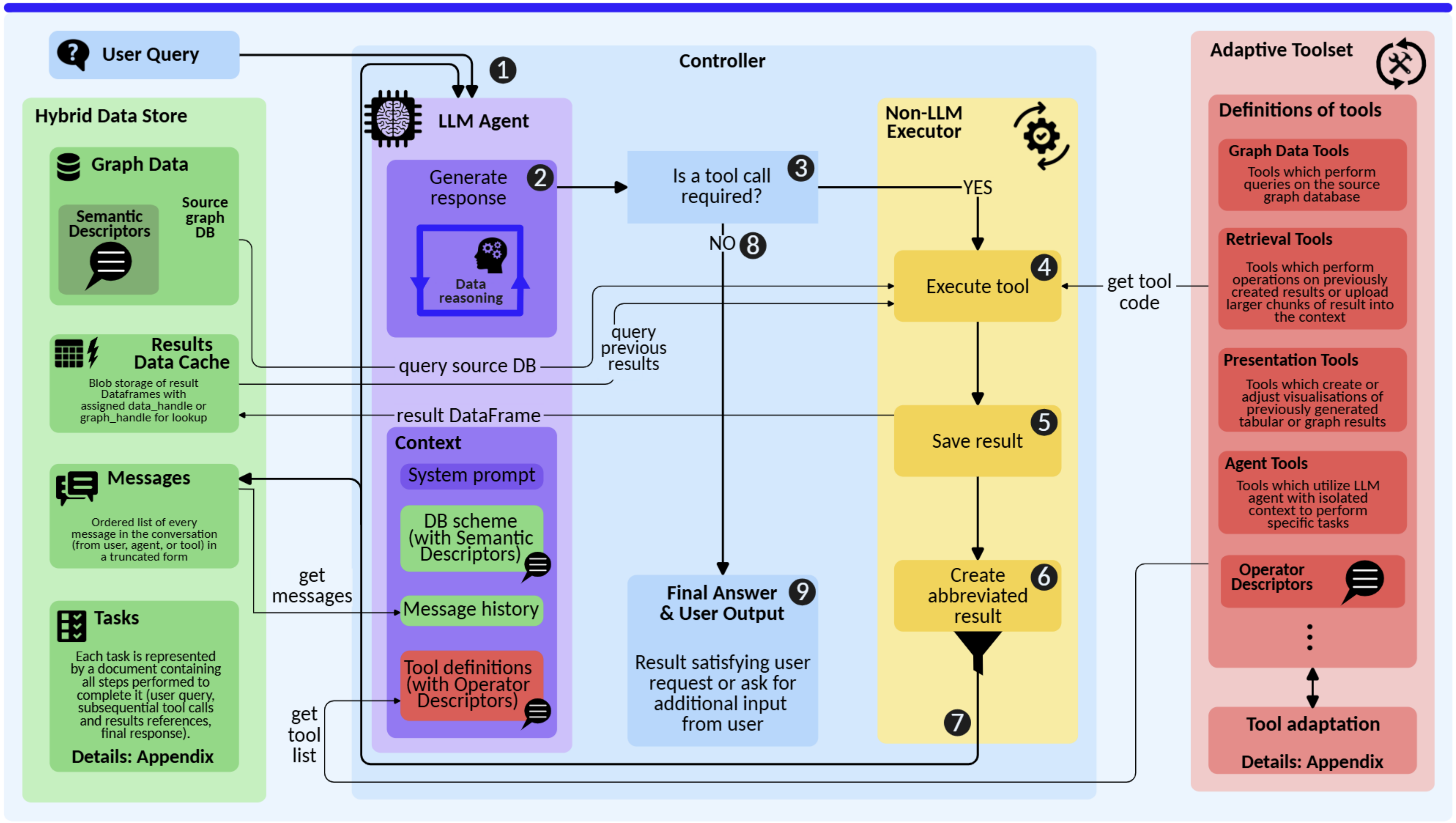
Example Semantic Descriptors



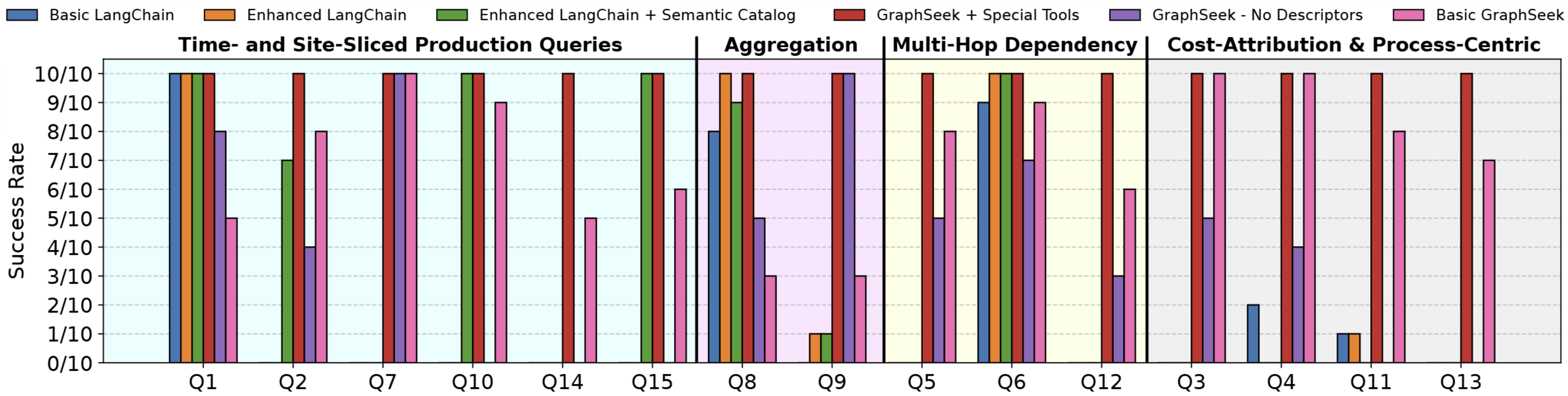
GraphSeek Design & Implementation



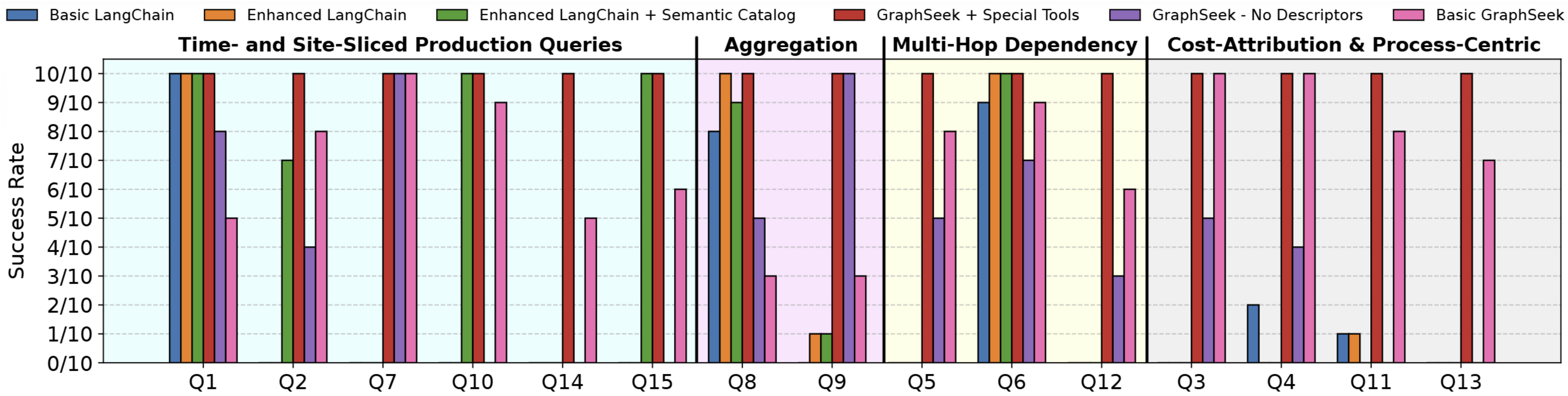
GraphSeek Design & Implementation



Evaluation

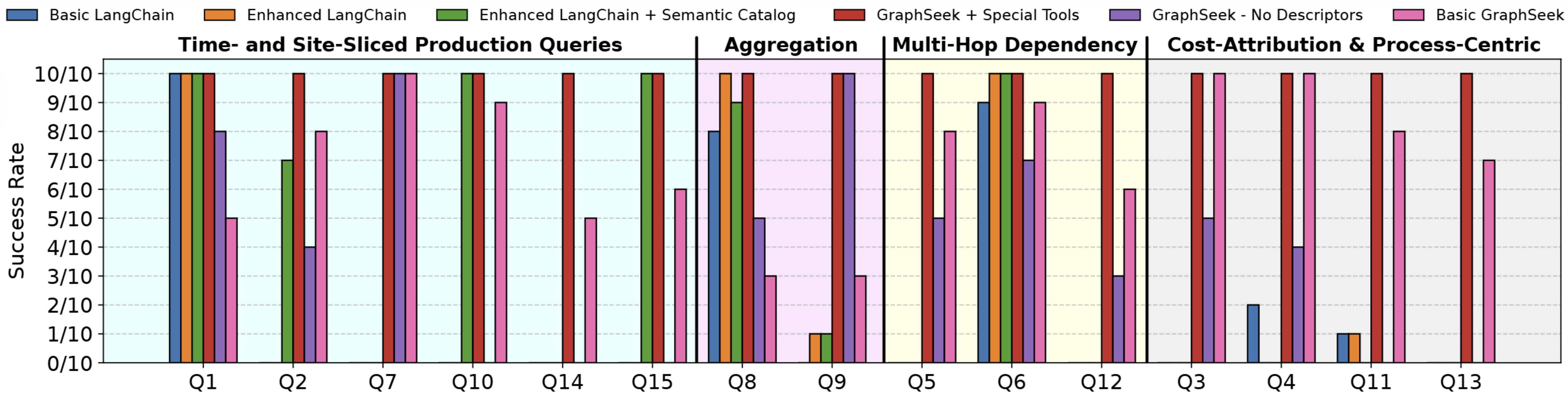

Basic LangChain
Enhanced LangChain
Enhanced LangChain + Semantic Catalog
GraphSeek + Special Tools
GraphSeek - No Descriptors
Basic GraphSeek
Time- and Site-Sliced Production Queries
Aggregation
Multi-Hop Dependency
Cost-Attribution & Process-Centric
Success Rate
10/10
9/10
8/10
7/10
6/10
5/10
4/10
3/10
2/10
1/10
0/10
Q1
Q2
Q7
Q10
Q14
Q15
Q8
Q9
Q5
Q6
Q12
Q3
Q4
Q11
Q13

Evaluation



How many base-tier drive assemblies do we produce at each factory site?

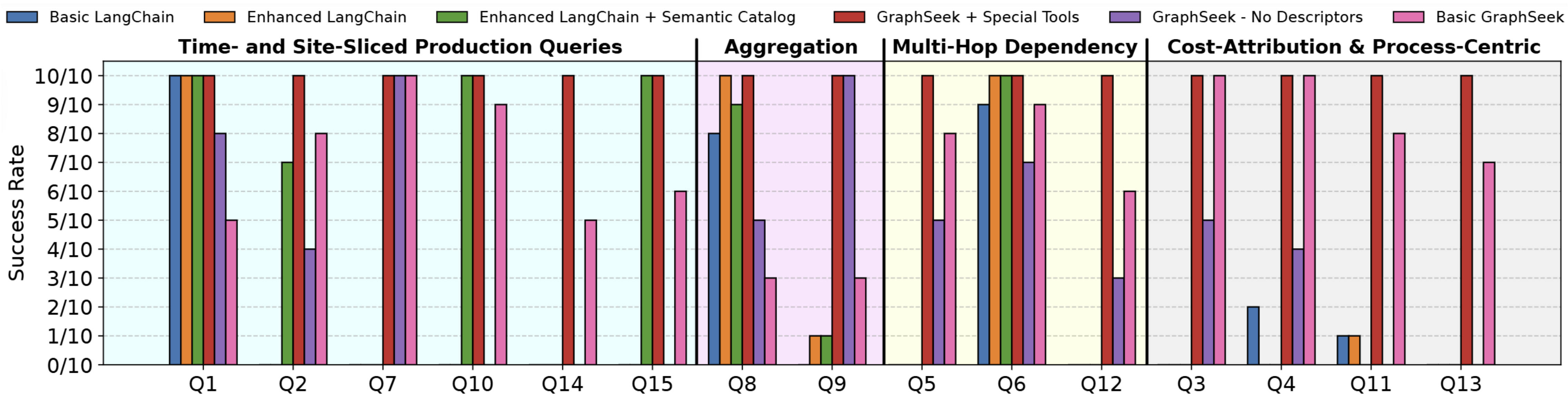
Evaluation



How many base-tier drive assemblies do we produce at each factory site?

Show me the unique manufacturing blueprint for vehicle model EV-X7

Evaluation

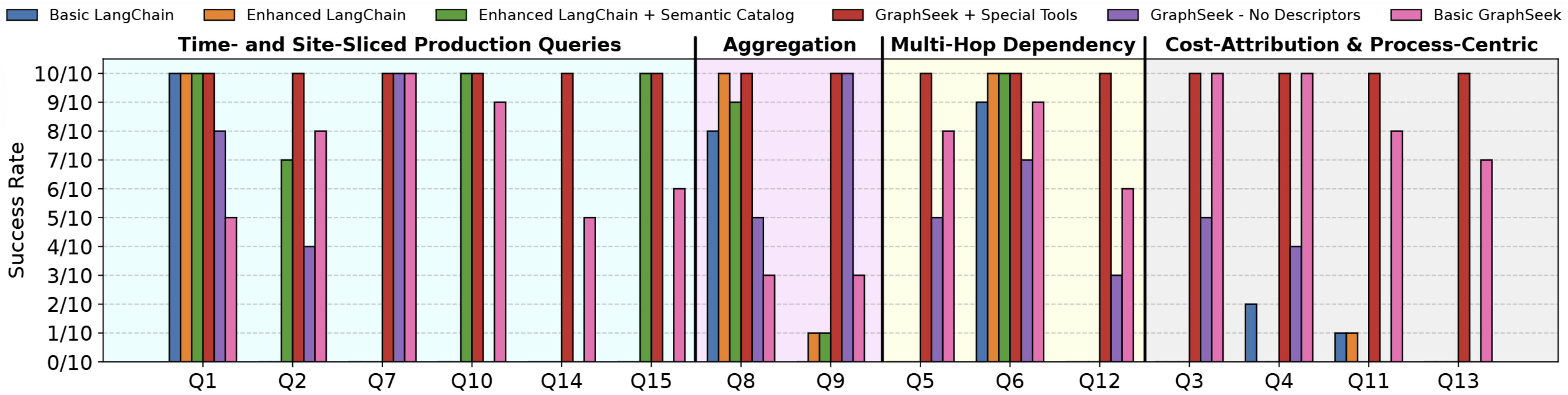


How many base-tier drive assemblies do we produce at each factory site?

Show me the unique manufacturing blueprint for vehicle model EV-X7

At which factory site is the component with the highest market price manufactured?

Evaluation



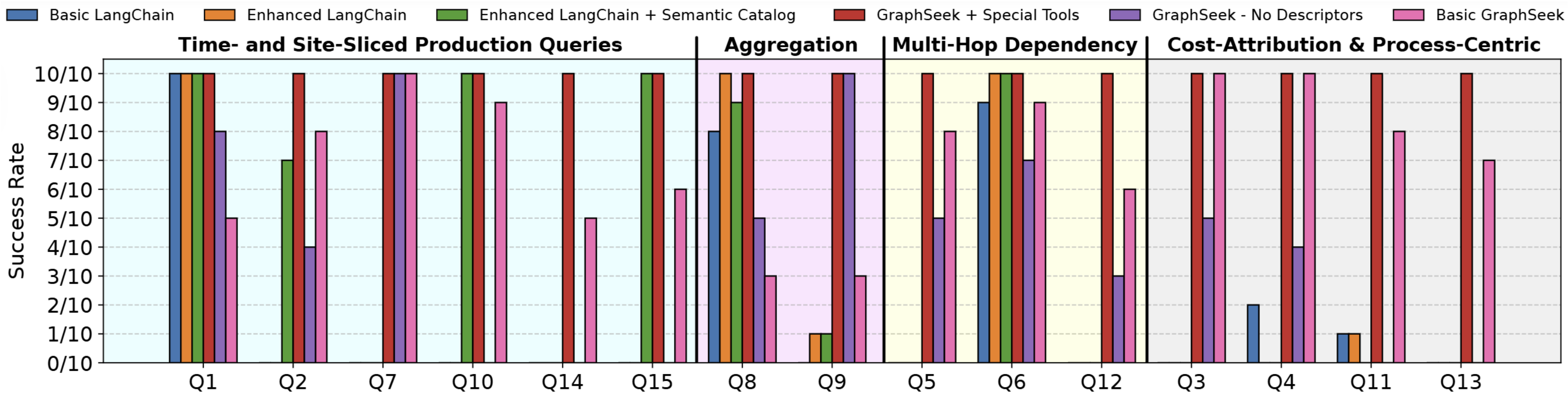
How many base-tier drive assemblies do we produce at each factory site?

Show me the unique manufacturing blueprint for vehicle model EV-X7

At which factory site is the component with the highest market price manufactured?

Check unique production plans for three vehicle models whose names start with ’B’, and tell me which one uses the largest number of modules.

Evaluation



How many base-tier drive assemblies do we produce at each factory site?

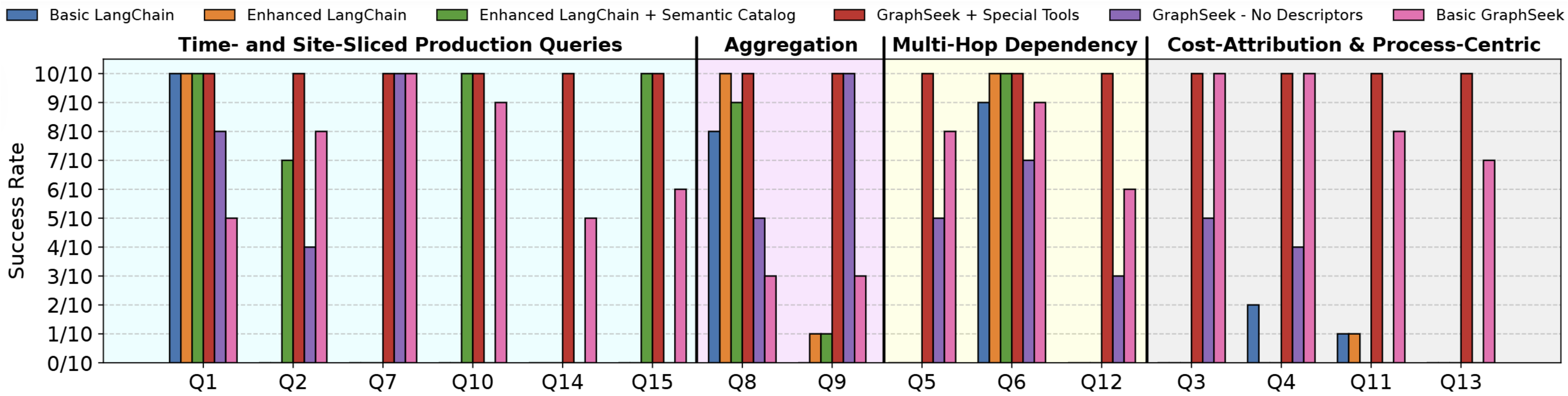
Show me the unique manufacturing blueprint for vehicle model EV-X7

At which factory site is the component with the highest market price manufactured?

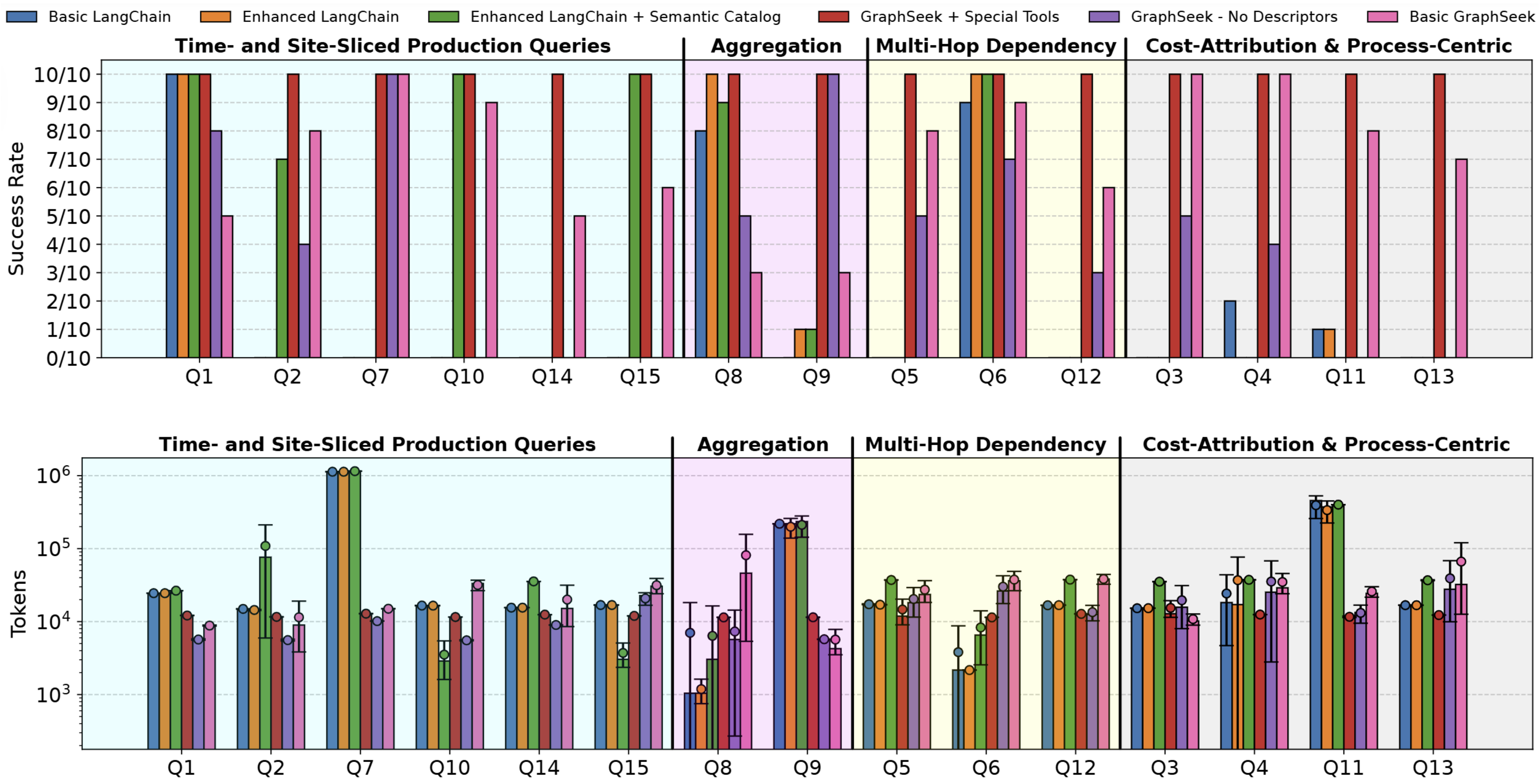
Check unique production plans for three vehicle models whose names start with ’B’, and tell me which one uses the largest number of modules.

What is the highest module cost?

Evaluation

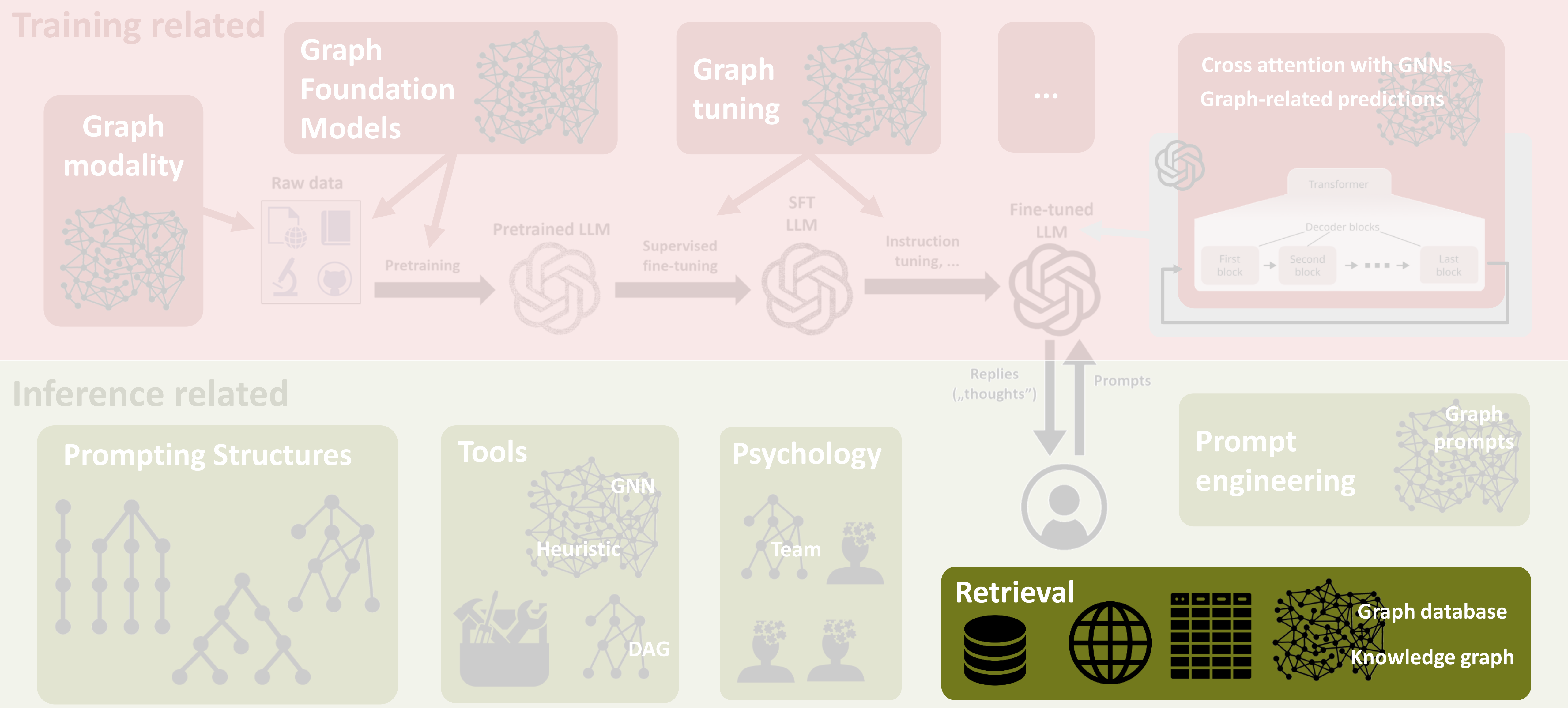

Basic LangChain
Enhanced LangChain
Enhanced LangChain + Semantic Catalog
GraphSeek + Special Tools
GraphSeek - No Descriptors
Basic GraphSeek
Time- and Site-Sliced Production Queries
Aggregation
Multi-Hop Dependency
Cost-Attribution & Process-Centric
Success Rate
10/10
9/10
8/10
7/10
6/10
5/10
4/10
3/10
2/10
1/10
0/10
Q1
Q2
Q7
Q10
Q14
Q15
Q8
Q9
Q5
Q6
Q12
Q3
Q4
Q11
Q13

Evaluation



Basic LangChain
Enhanced LangChain
Enhanced LangChain + Semantic Catalog
GraphSeek + Special Tools
GraphSeek - No Descriptors
Basic GraphSeek
Time- and Site-Sliced Production Queries
Aggregation
Multi-Hop Dependency
Cost-Attribution & Process-Centric
Success Rate
10/10
9/10
8/10
7/10
6/10
5/10
4/10
3/10
2/10
1/10
0/10
Q1
Q2
Q7
Q10
Q14
Q15
Q8
Q9
Q5
Q6
Q12
Q3
Q4
Q11
Q13
Tokens
10^6
10^5
10^4
10^3

Graphs in the LLM Pipeline: Overview



Graphs in the LLM Pipeline: Overview

It's all about LLMs these days...
But AI is not just LLMs!

Graphs + Deep Learning = Graphs Neural Networks (GNNs)

Graphs + Deep Learning = Graphs Neural Networks (GNNs)

Article

A graph placement methodology for fast chip design

https://doi.org/10.1038/s41586-021-03544-w
Received: 3 November 2020
Accepted: 13 April 2021
Published online: 9 June 2021
Check for updates

Azalia Mirhoseini[1,4] ✉, Anna Goldie[1,3,4] ✉, Mustafa Yazgan[2], Joe Wenjie Jiang[1], Ebrahim Songhori[1], Shen Wang[1], Young-Joon Lee[2], Eric Johnson[1], Omkar Pathak[2], Azade Nazi[1], Jiwoo Pak[2], Andy Tong[2], Kavya Srinivasa[2], William Hang[3], Emre Tuncer[2], Quoc V. Le[1], James Laudon[1], Richard Ho[2], Roger Carpenter[2] & Jeff Dean[1]

Chip floorplanning is the engineering task of designing the physical layout of a computer chip. Despite five decades of research[1], chip floorplanning has defied

Graphs + Deep Learning = Graphs Neural Networks (GNNs)

Article

A graph placement methodology for fast chip design

https://doi.org/10.1038/s41586-021-03
Received: 3 November 2020
Accepted: 13 April 2021
Published online: 9 June 2021
Check for updates

Article

Advancing mathematics by guiding human intuition with AI

https://doi.org/10.1038/s41586-021-04086-x
Received: 10 July 2021
Accepted: 30 September 2021
Published online: 1 December 2021
Open access

Alex Davies[1✉], Petar Veličković[1], Lars Buesing[1], Sam Blackwell[1], Daniel Zheng[1], Nenad Tomašev[1], Richard Tanburn[1], Peter Battaglia[1], Charles Blundell[1], András Juhász[2], Marc Lackenby[2], Geordie Williamson[3], Demis Hassabis[1] & Pushmeet Kohli[1✉]

The practice of mathematics involves discovering patterns and using these to formulate and prove conjectures, resulting in theorems. Since the 1960s,

Graphs + Deep Learning = Graphs Neural Networks (GNNs)

Article

A graph placement methodology for fast chip design

https://doi.org/10.1038/s41586-021-03

Received: 3 November 2020

Accepted: 13 April 2021

Published online: 9 June 2021

Check for updates

Article

Advancing mathematics by guiding human intuition with AI

https://doi.org/10.1038/s41586-021-04086-x

Received: 10 July 2021

Accepted: 30 September 2021

Published online: 1 December 2021

Open access

Alex Da
Nenad
Marc La

The pra
formul

Article

Highly accurate protein structure prediction with AlphaFold

https://doi.org/10.1038/s41586-021-03819-2

Received: 11 May 2021

Accepted: 12 July 2021

Published online: 15 July 2021

Open access

Check for updates

John Jumper[1,4]✉, Richard Evans[1,4], Alexander Pritzel[1,4], Tim Green[1,4], Michael Figurnov[1,4], Olaf Ronneberger[1,4], Kathryn Tunyasuvunakool[1,4], Russ Bates[1,4], Augustin Žídek[1,4], Anna Potapenko[1,4], Alex Bridgland[1,4], Clemens Meyer[1,4], Simon A. A. Kohl[1,4], Andrew J. Ballard[1,4], Andrew Cowie[1,4], Bernardino Romera-Paredes[1,4], Stanislav Nikolov[1,4], Rishub Jain[1,4], Jonas Adler[1], Trevor Back[1], Stig Petersen[1], David Reiman[1], Ellen Clancy[1], Michal Zielinski[1], Martin Steinegger[2,3], Michalina Pacholska[1], Tamas Berghammer[1], Sebastian Bodenstein[1], David Silver[1], Oriol Vinyals[1], Andrew W. Senior[1], Koray Kavukcuoglu[1], Pushmeet Kohli[1] & Demis Hassabis[1,4]✉

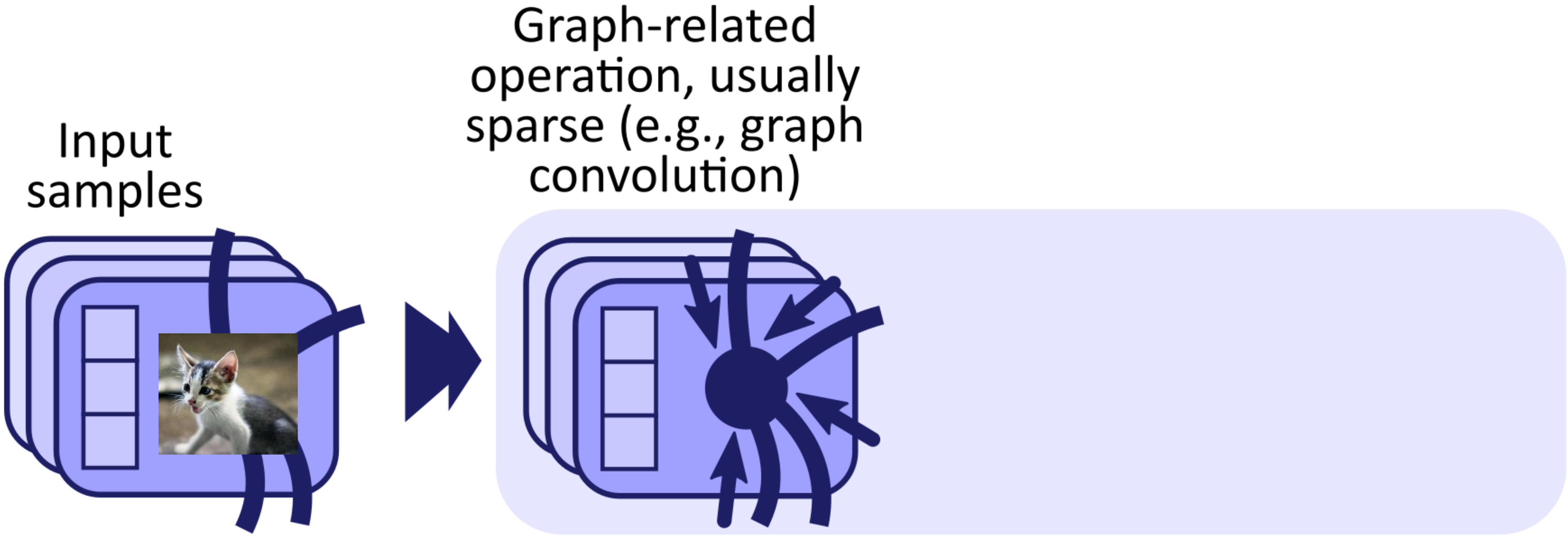
Proteins are essential to life, and understanding their structure can facilitate a

A Single GNN Layer

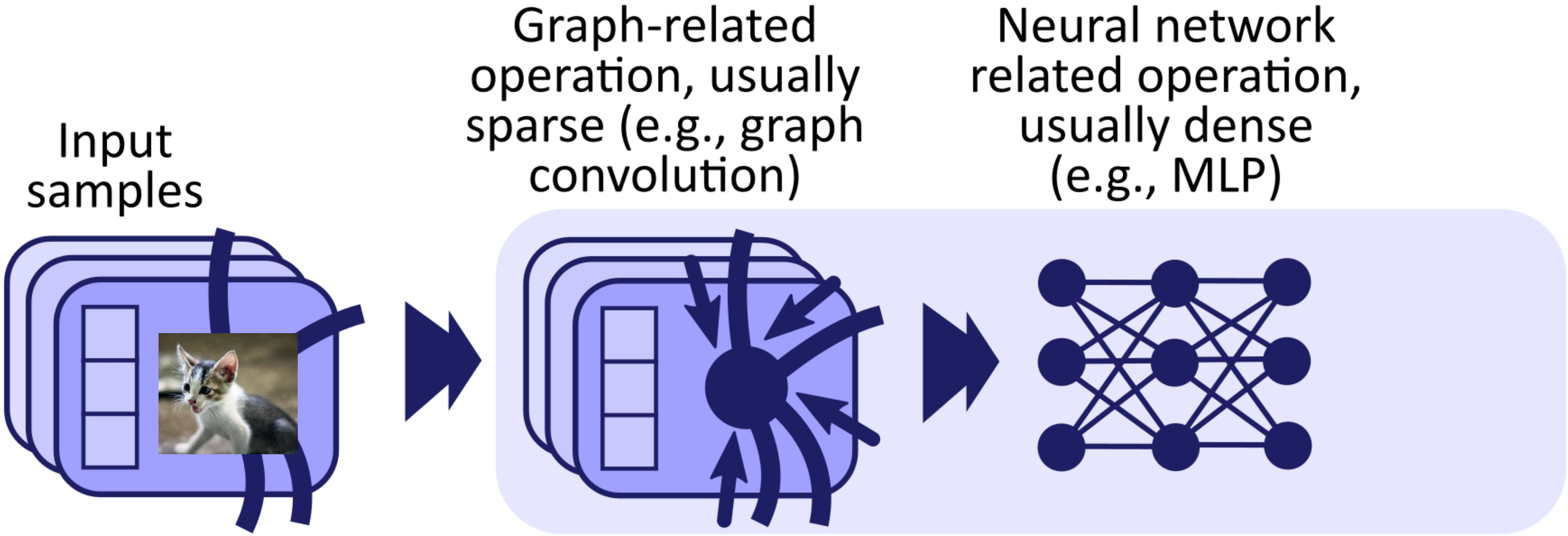
A Single GNN Layer

Input samples

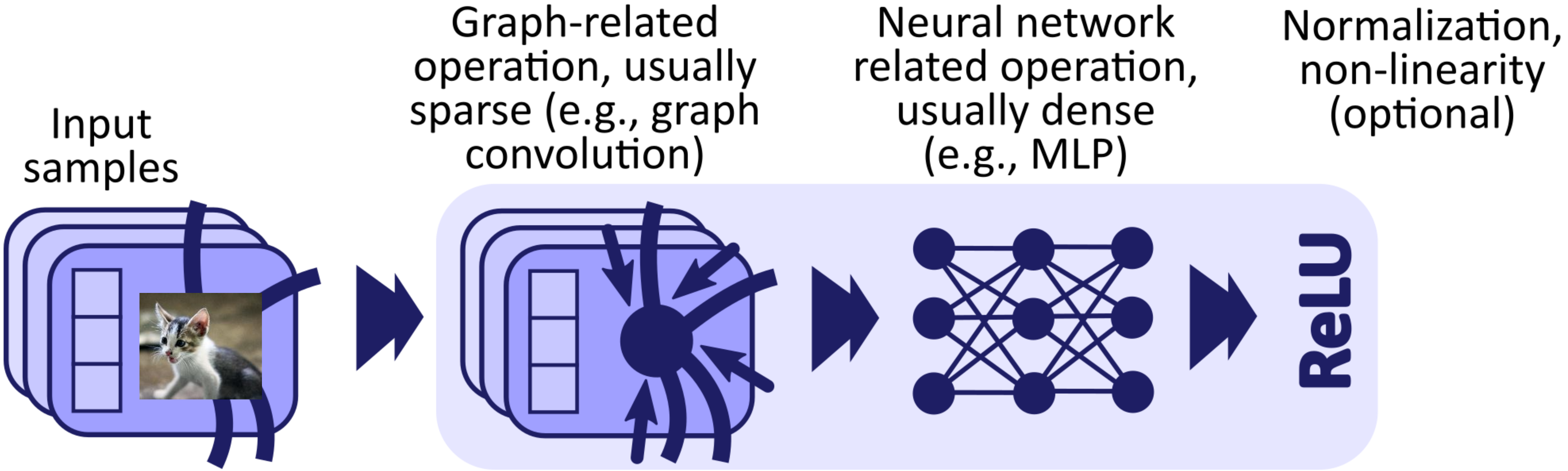
A Single GNN Layer



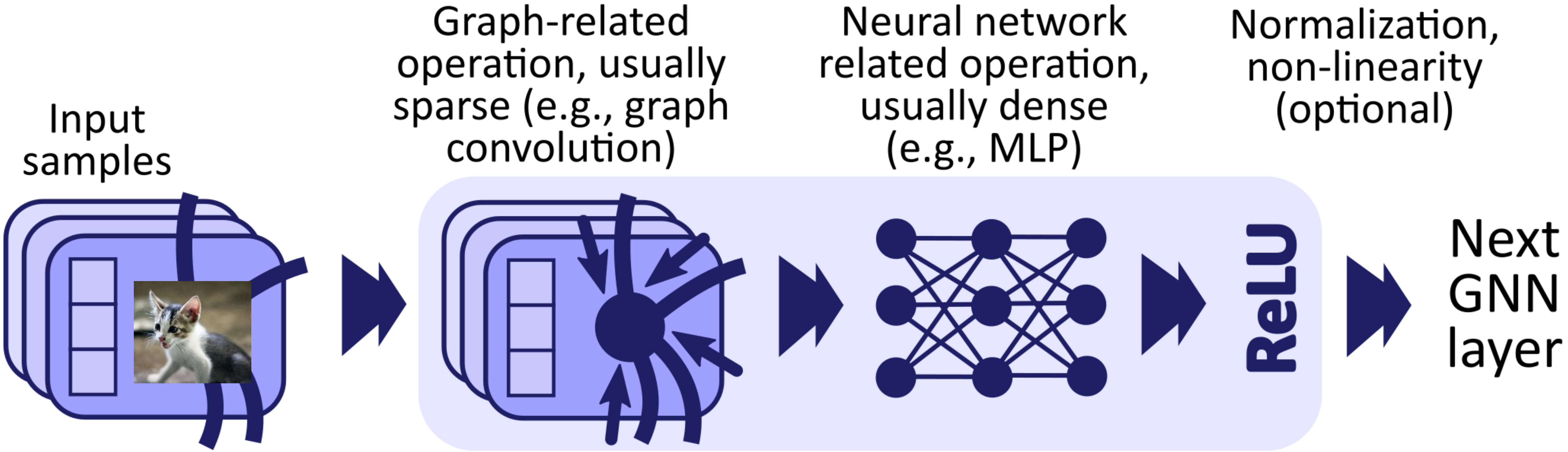
A Single GNN Layer



A Single GNN Layer

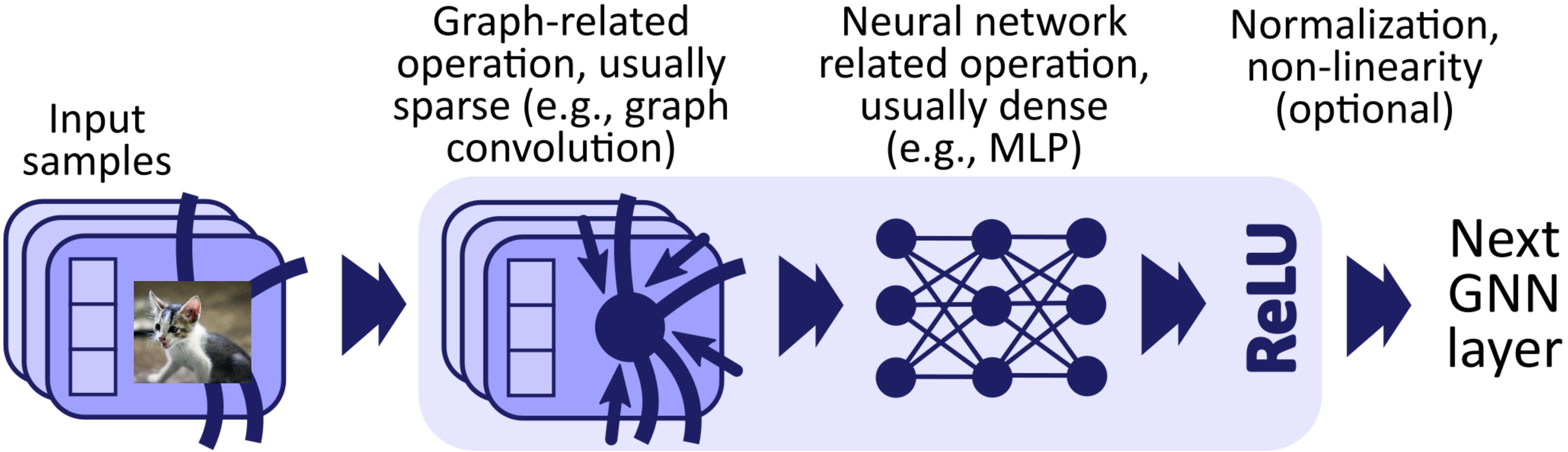


A Single GNN Layer



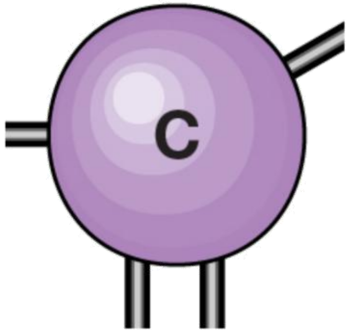
A Single GNN Layer

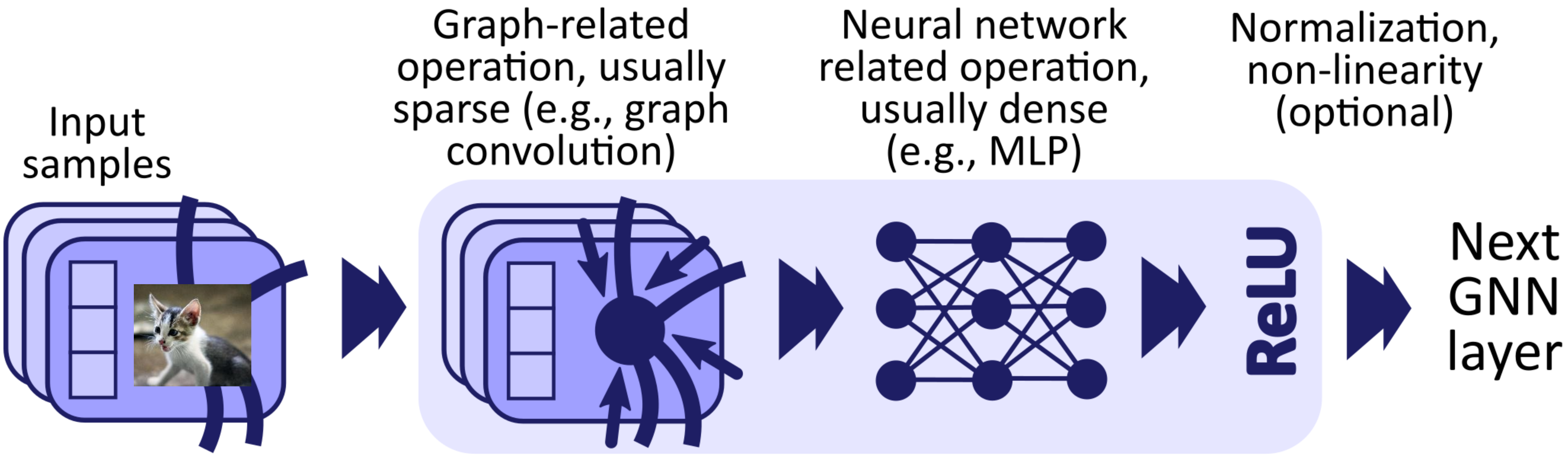
Classification or regression of nodes



A Single GNN Layer

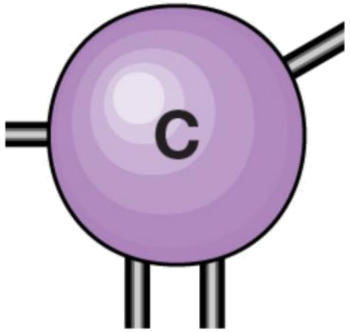
Classification or regression of nodes





A Single GNN Layer

Classification or regression of nodes



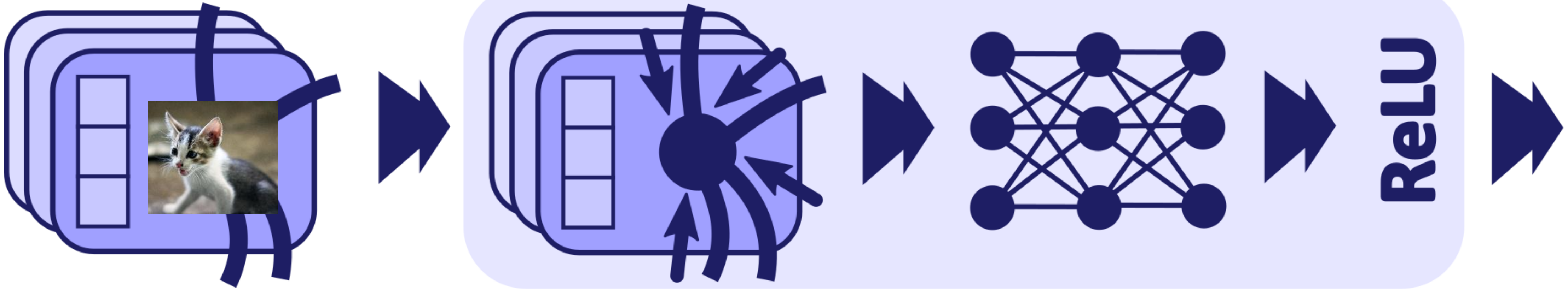
Classification or regression of edges

Input samples

Graph-related operation, usually sparse (e.g., graph convolution)

Neural network related operation, usually dense (e.g., MLP)

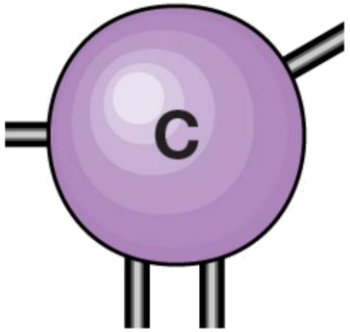
Normalization, non-linearity (optional)



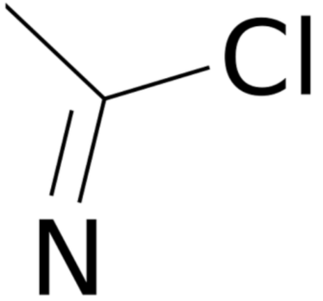
Next GNN layer

A Single GNN Layer

Classification or regression of nodes



Classification or regression of edges

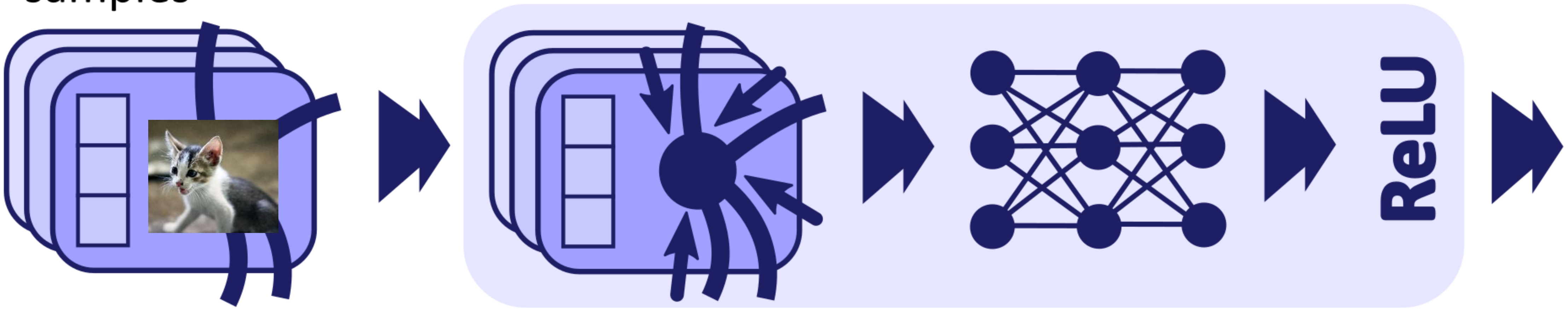


Input samples

Graph-related operation, usually sparse (e.g., graph convolution)

Neural network related operation, usually dense (e.g., MLP)

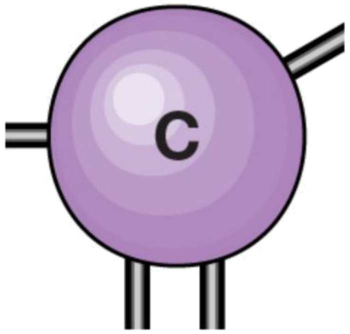
Normalization, non-linearity (optional)



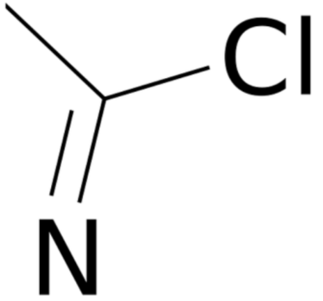
Next GNN layer

A Single GNN Layer

Classification or regression of nodes



Classification or regression of edges



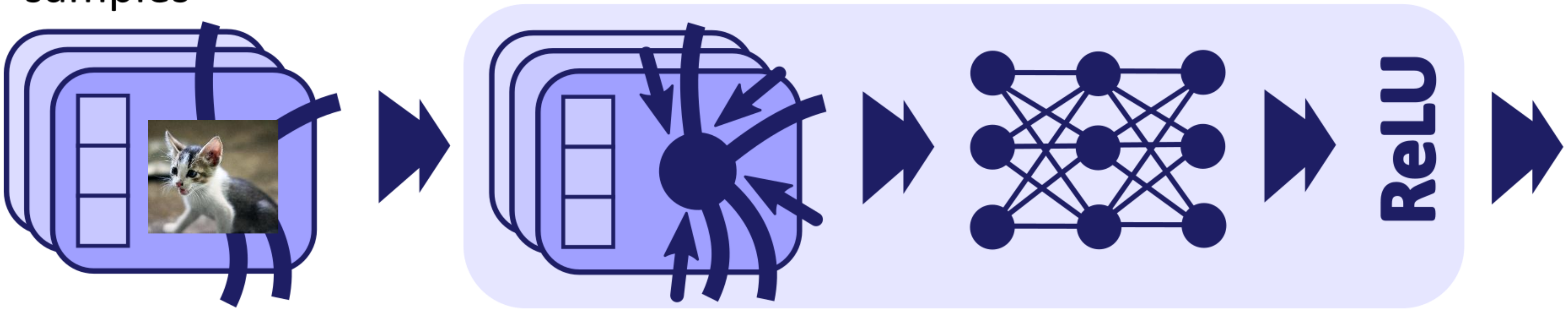
Classification or regression of graphs

Input samples

Graph-related operation, usually sparse (e.g., graph convolution)

Neural network related operation, usually dense (e.g., MLP)

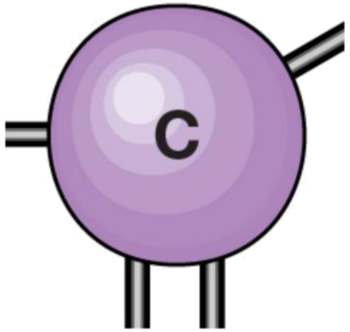
Normalization, non-linearity (optional)



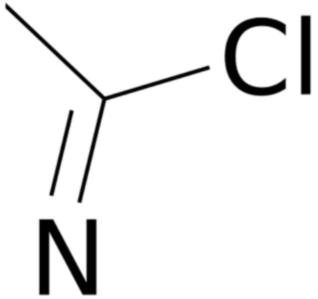
Next GNN layer

A Single GNN Layer

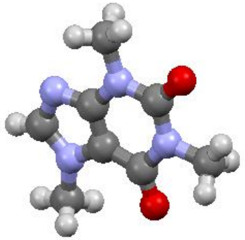
Classification or regression of nodes



Classification or regression of edges



Classification or regression of graphs

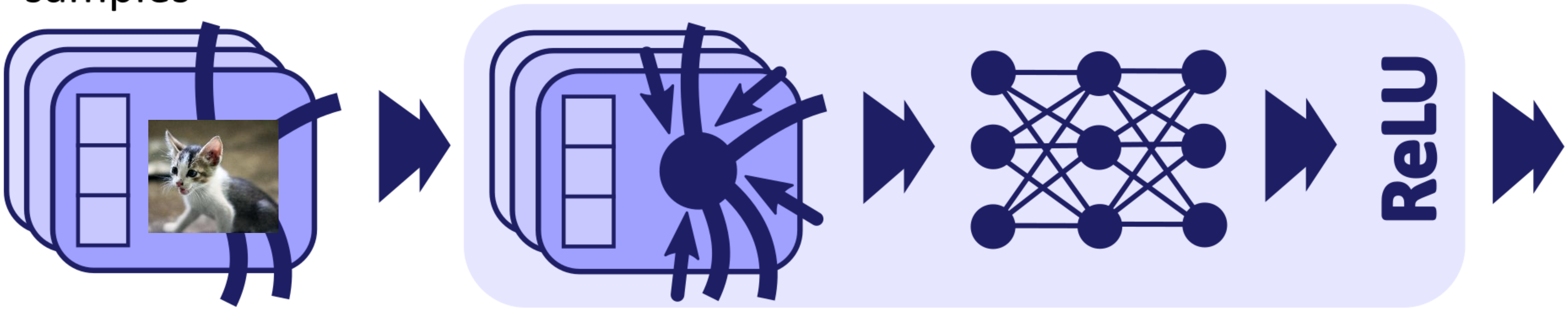


Input samples

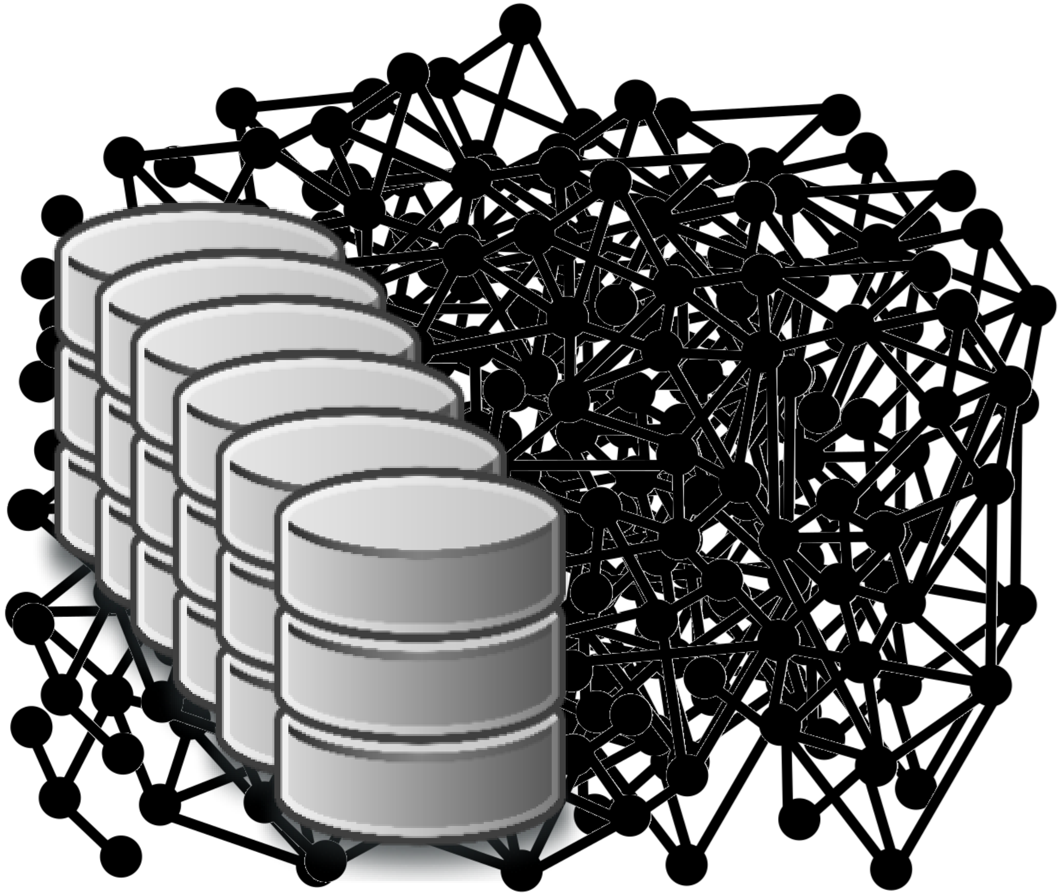
Graph-related operation, usually sparse (e.g., graph convolution)

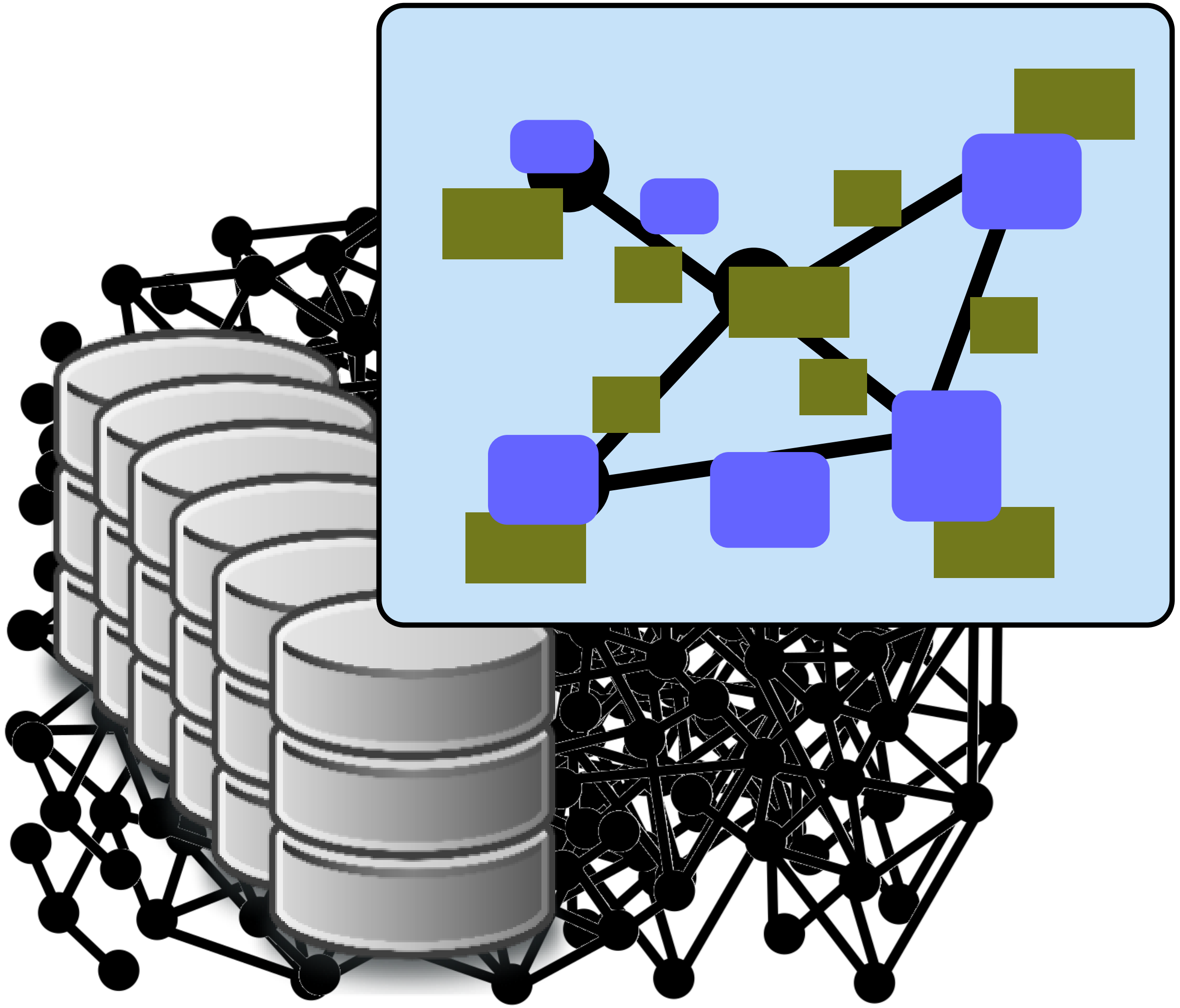
Neural network related operation, usually dense (e.g., MLP)

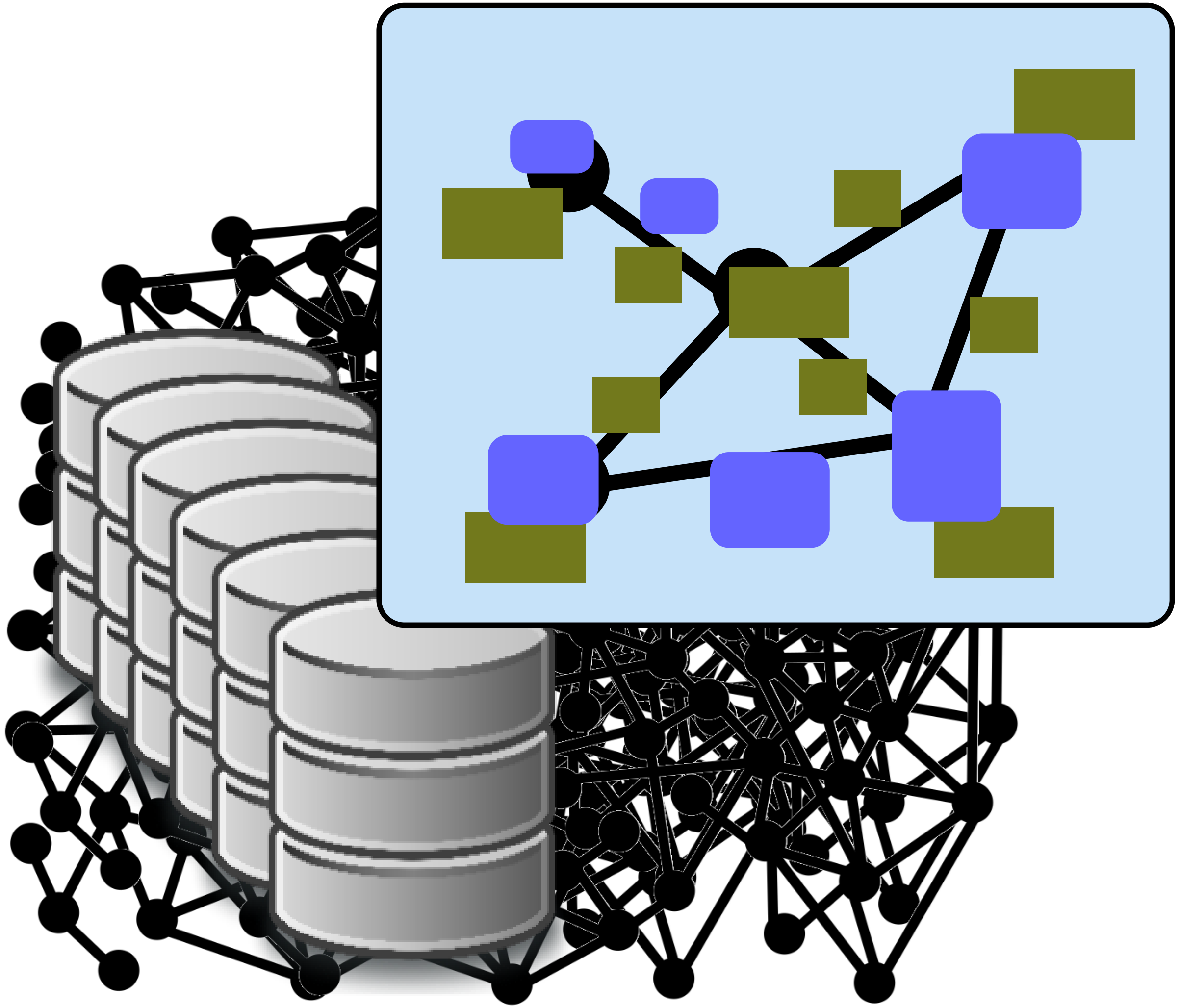
Normalization, non-linearity (optional)

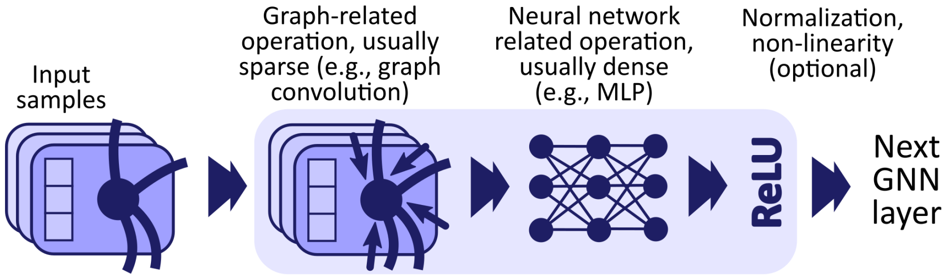


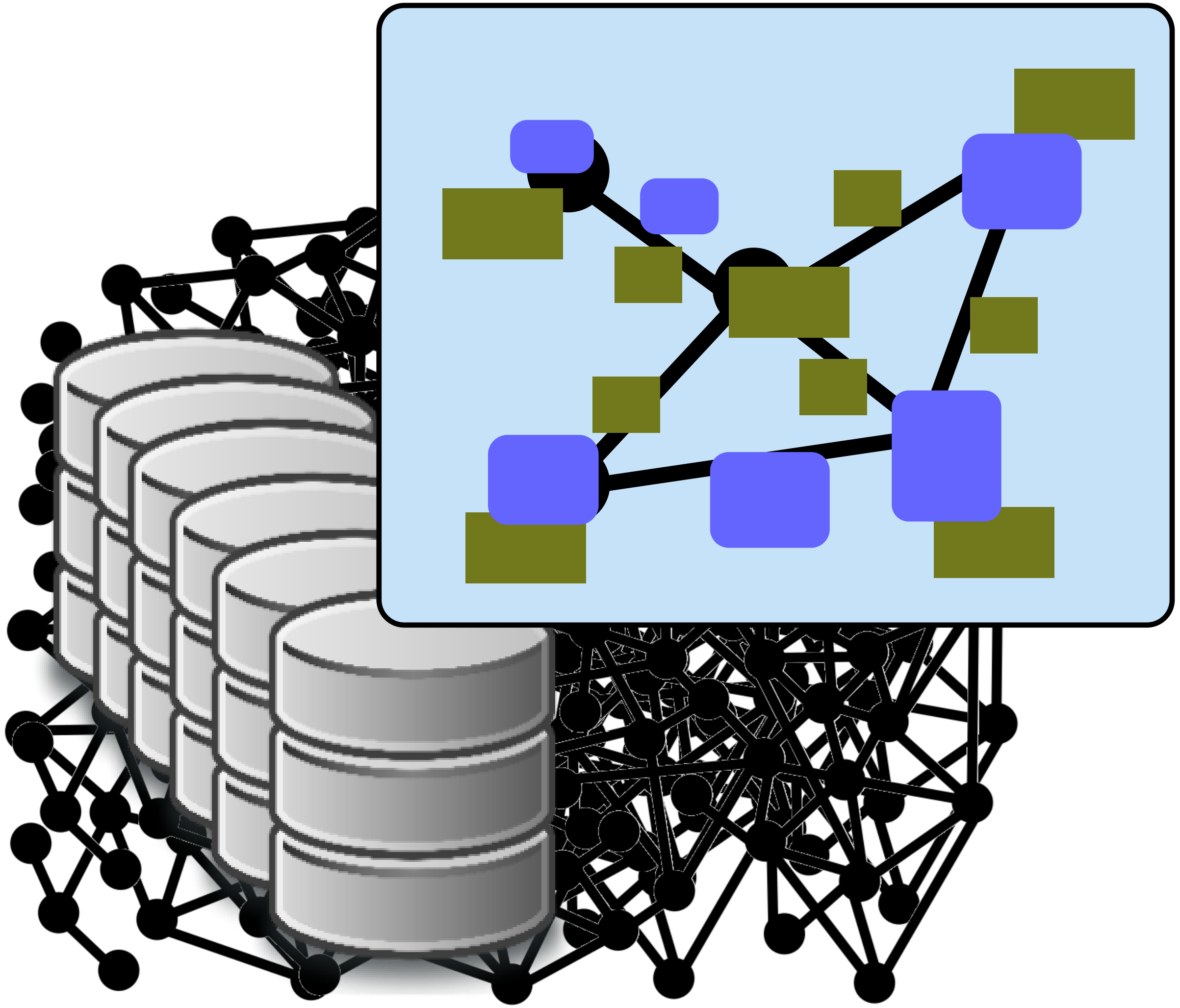
Next GNN layer

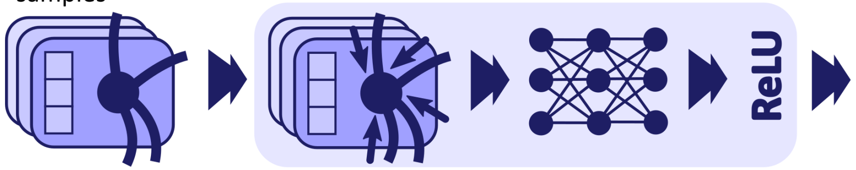




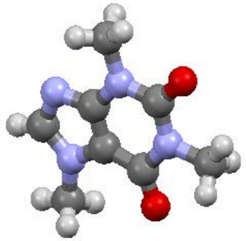


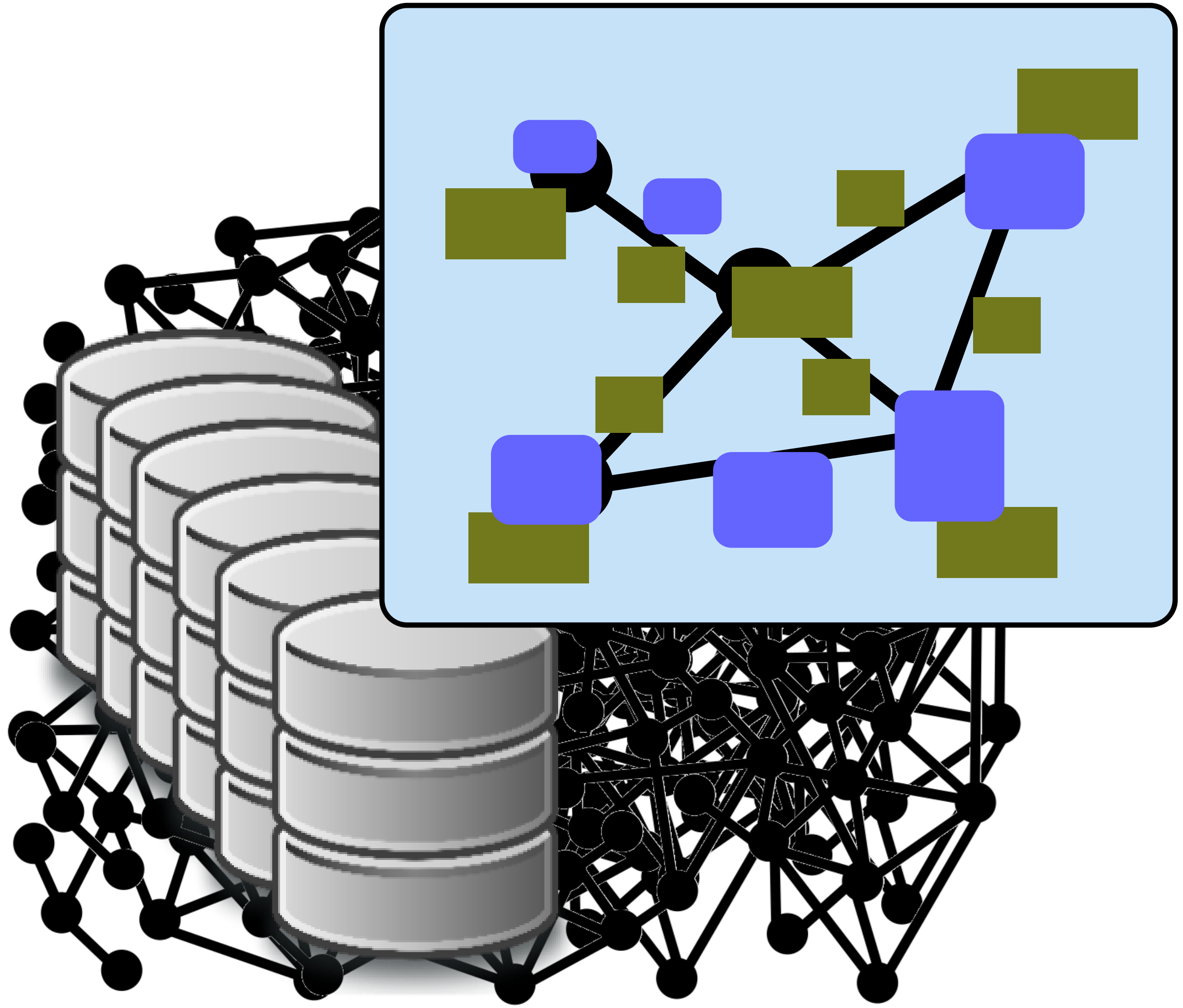

Input samples
Graph-related operation, usually sparse (e.g., graph convolution)
Neural network related operation, usually dense (e.g., MLP)
Normalization, non-linearity (optional)
ReLU
Next GNN layer

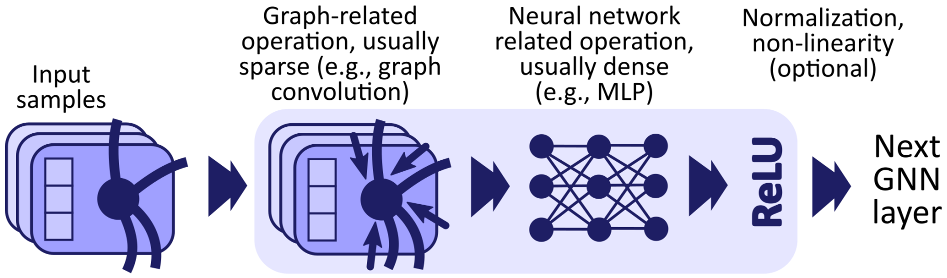




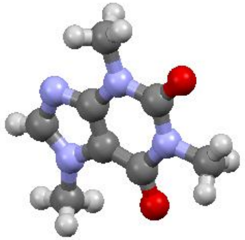
Classification or regression of graphs



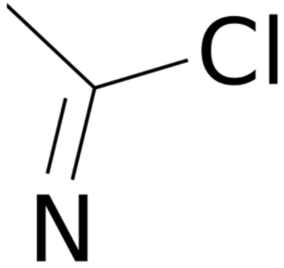


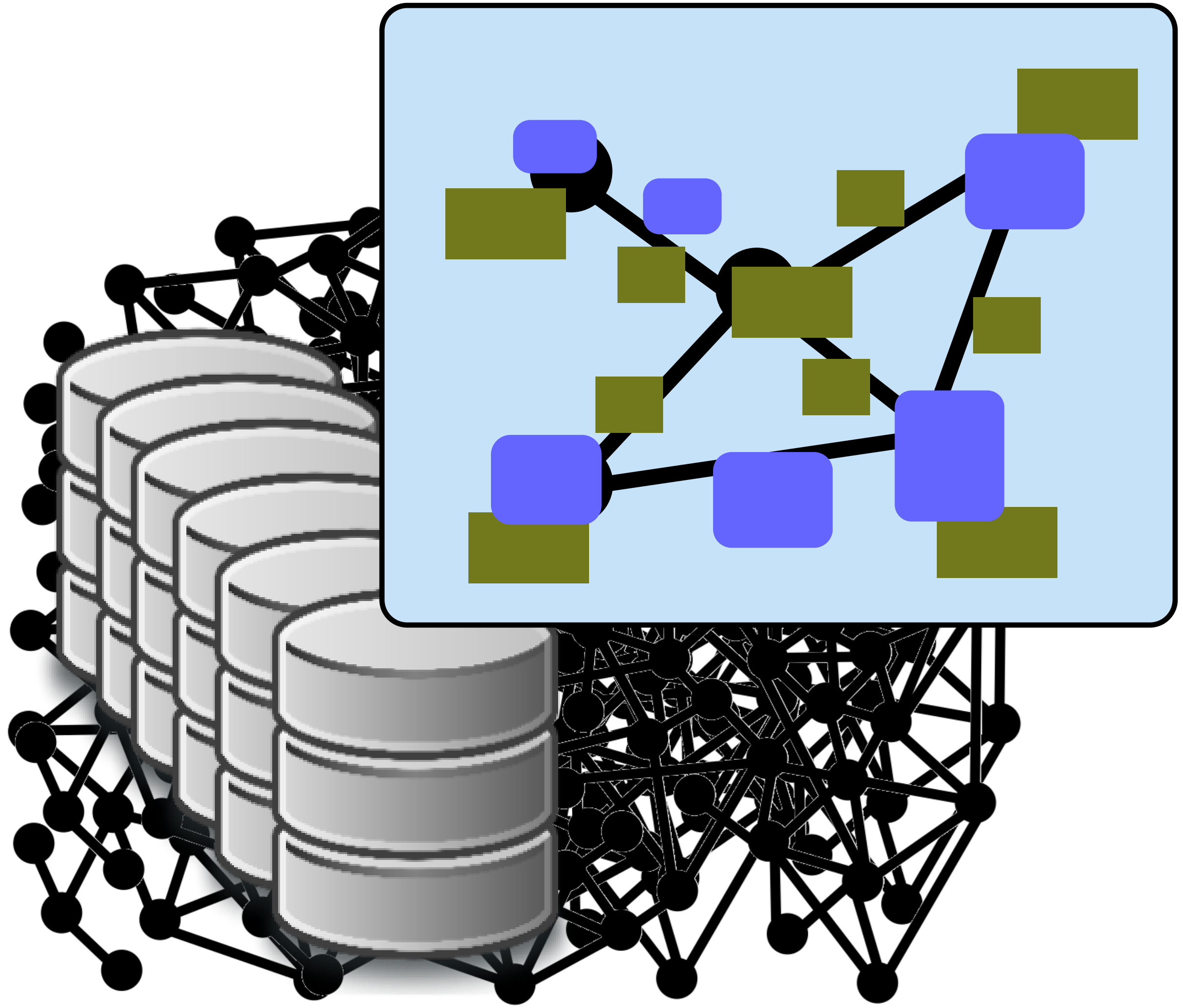


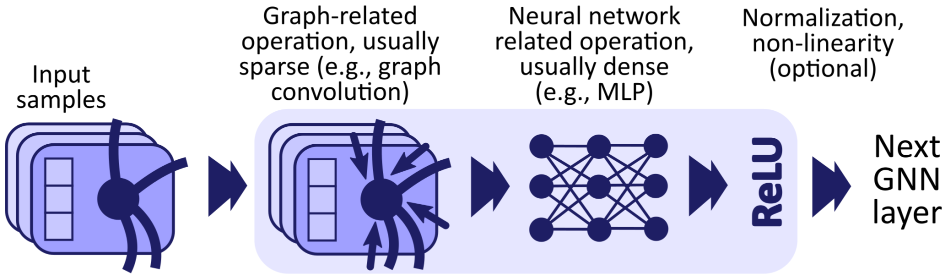
Classification or regression of graphs



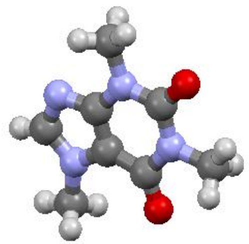
Classification or regression of edges



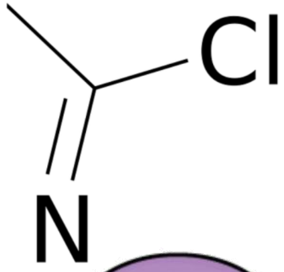




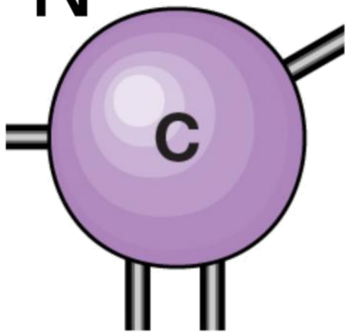
Classification or regression of graphs



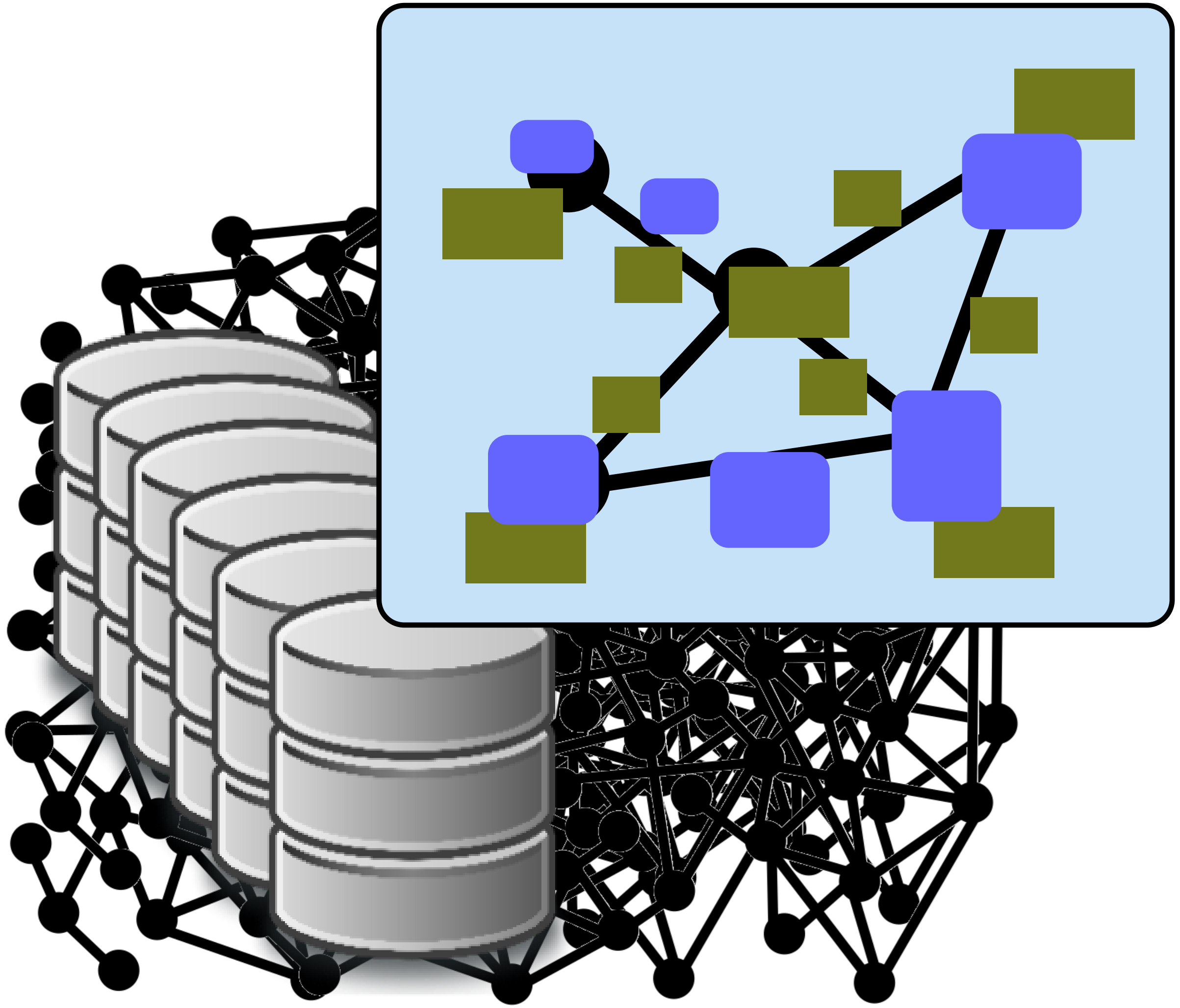
Classification or regression of edges



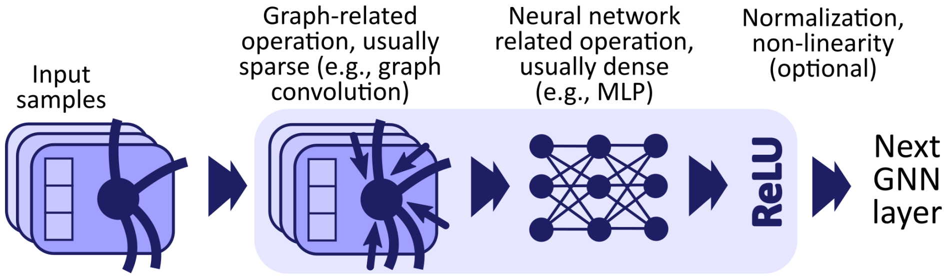
Classification or regression of nodes



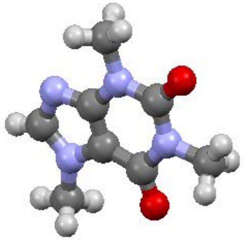
How to marry these two?



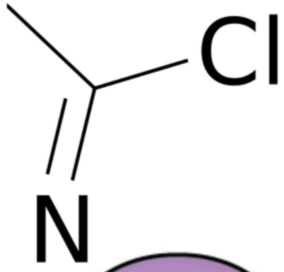
+



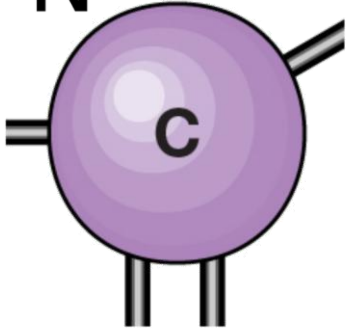
Classification or regression of graphs



Classification or regression of edges



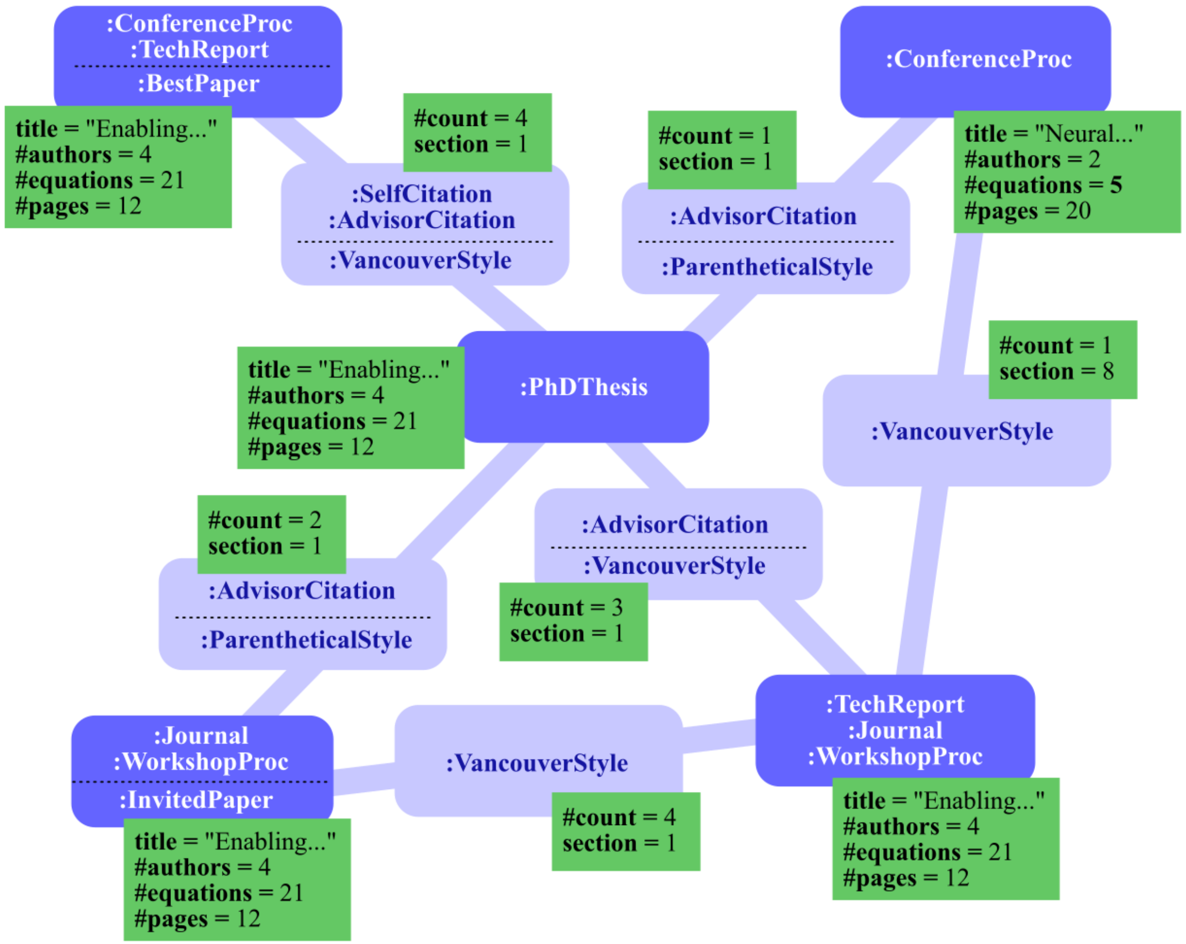
Classification or regression of nodes



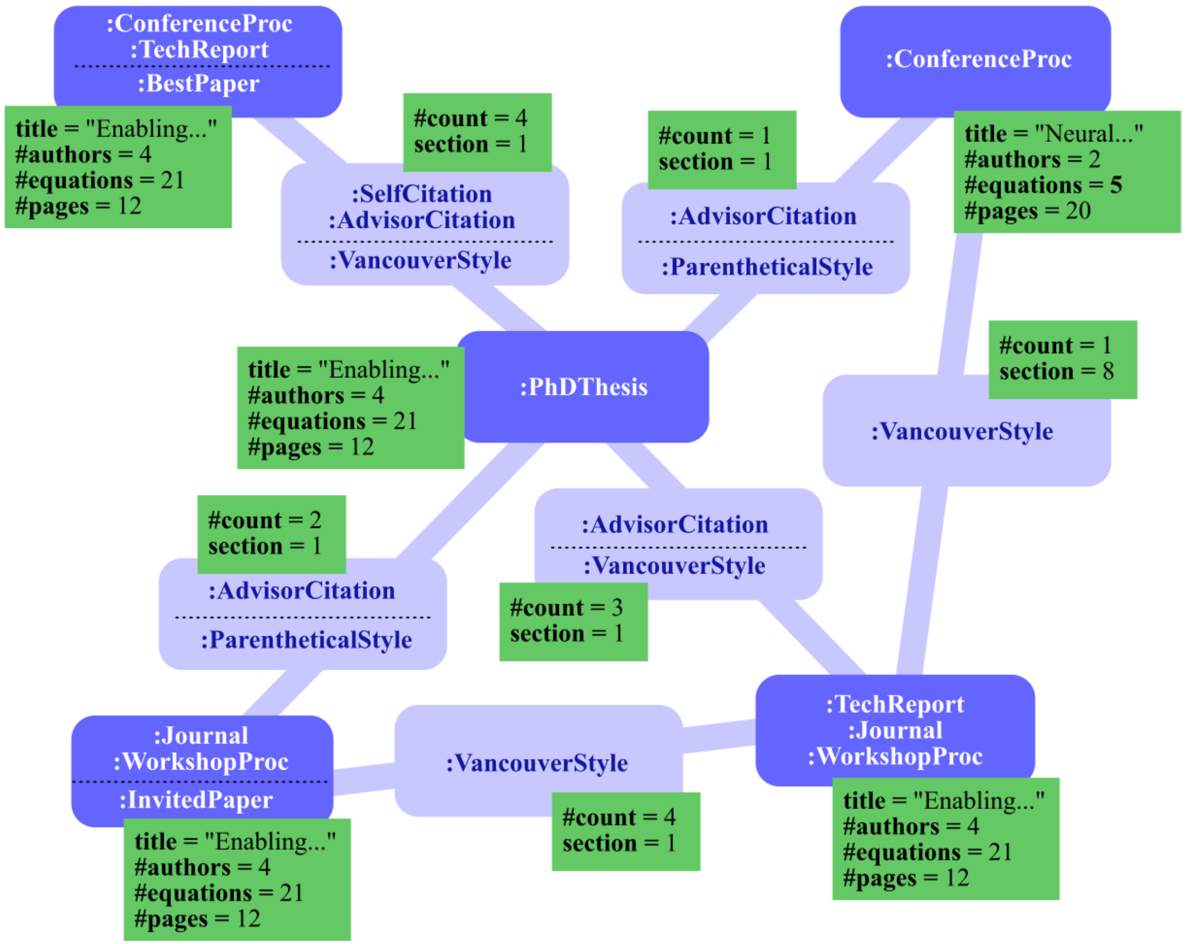
How to marry these two?

Main challenge: understanding how to merge these two conceptually

Marrying Graph Databases and GNNs

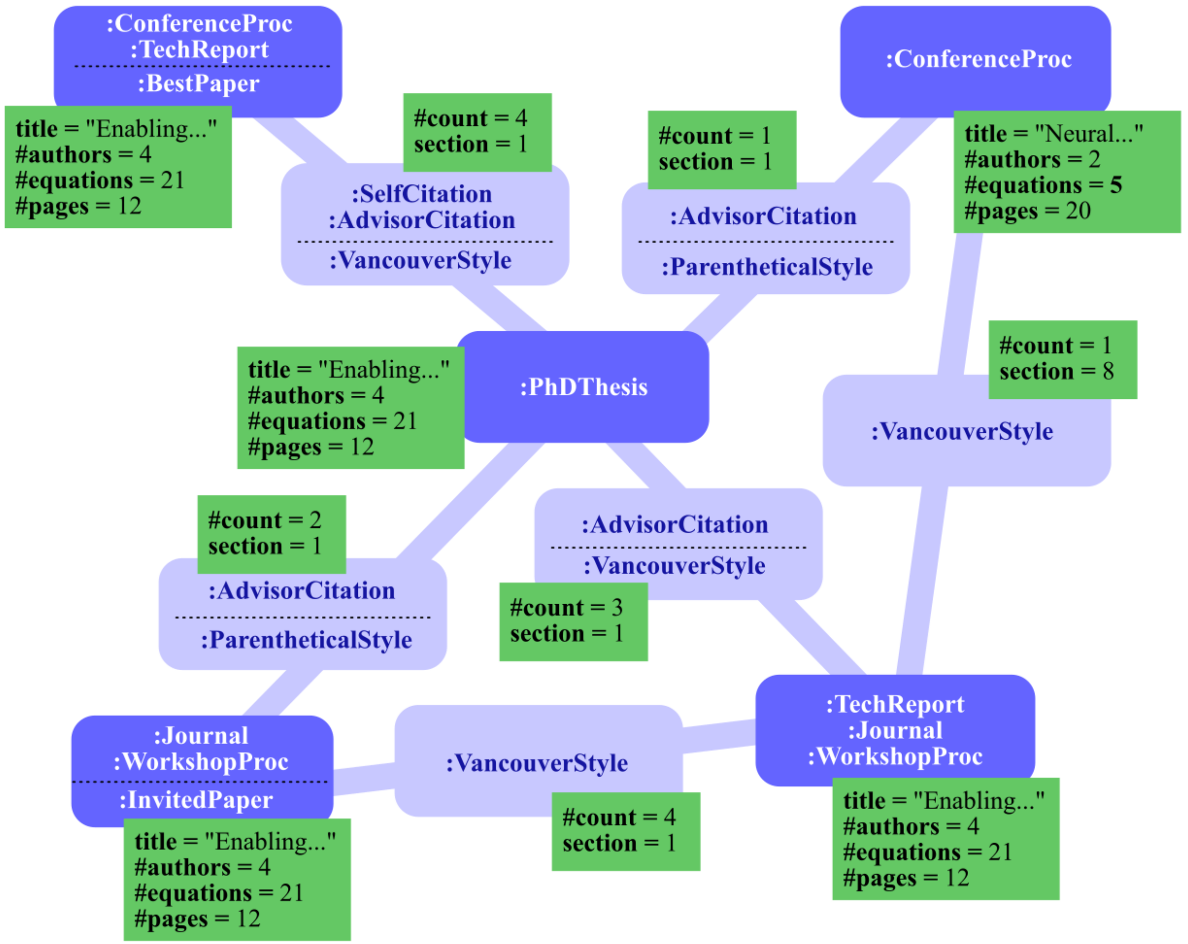


Marrying Graph Databases and GNNs



There is a neat way to think about the GNN workloads within the LPG model

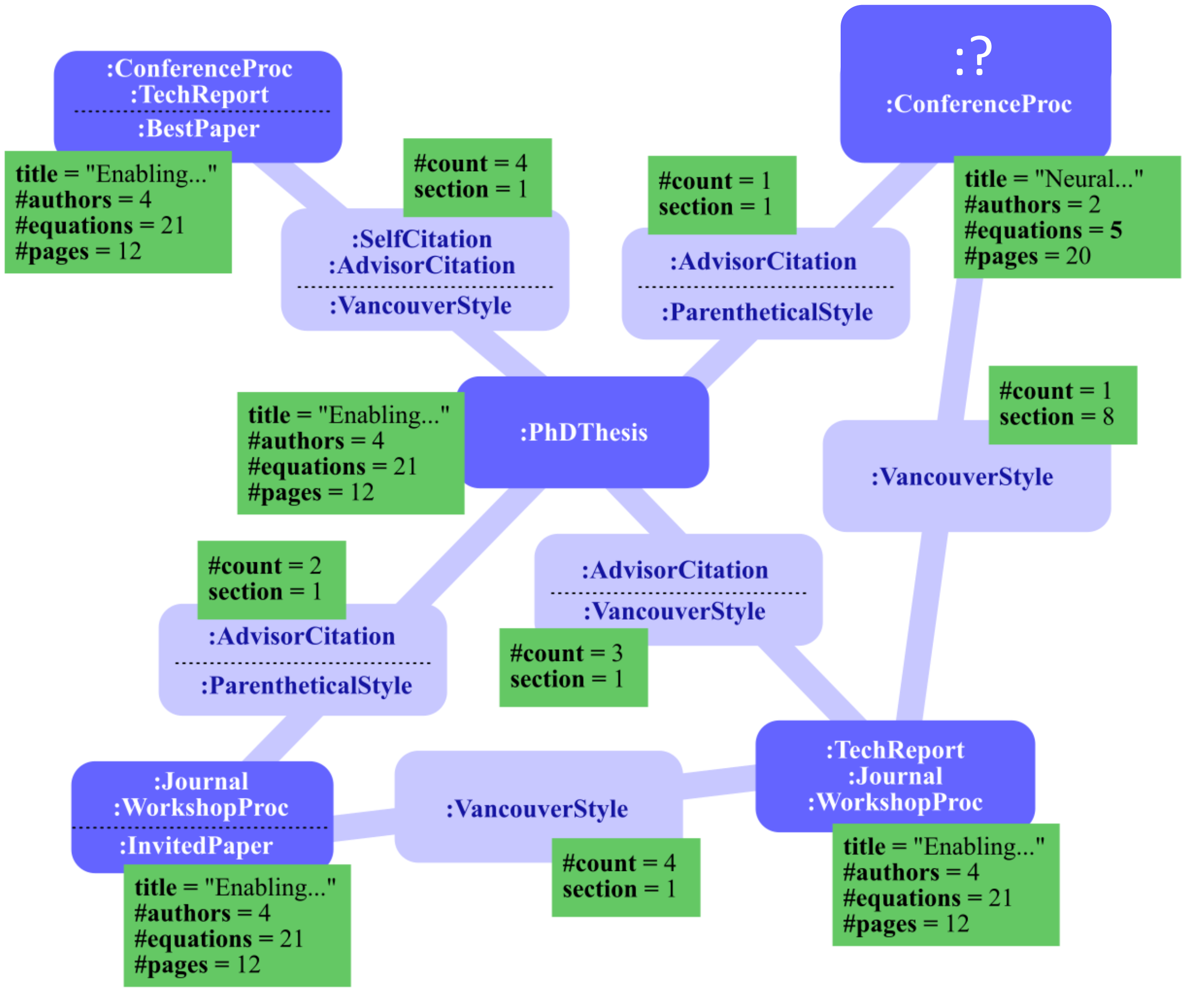
Marrying Graph Databases and GNNs



There is a neat way to think about the GNN workloads within the LPG model

Node/edge/graph classification becomes ***label prediction***

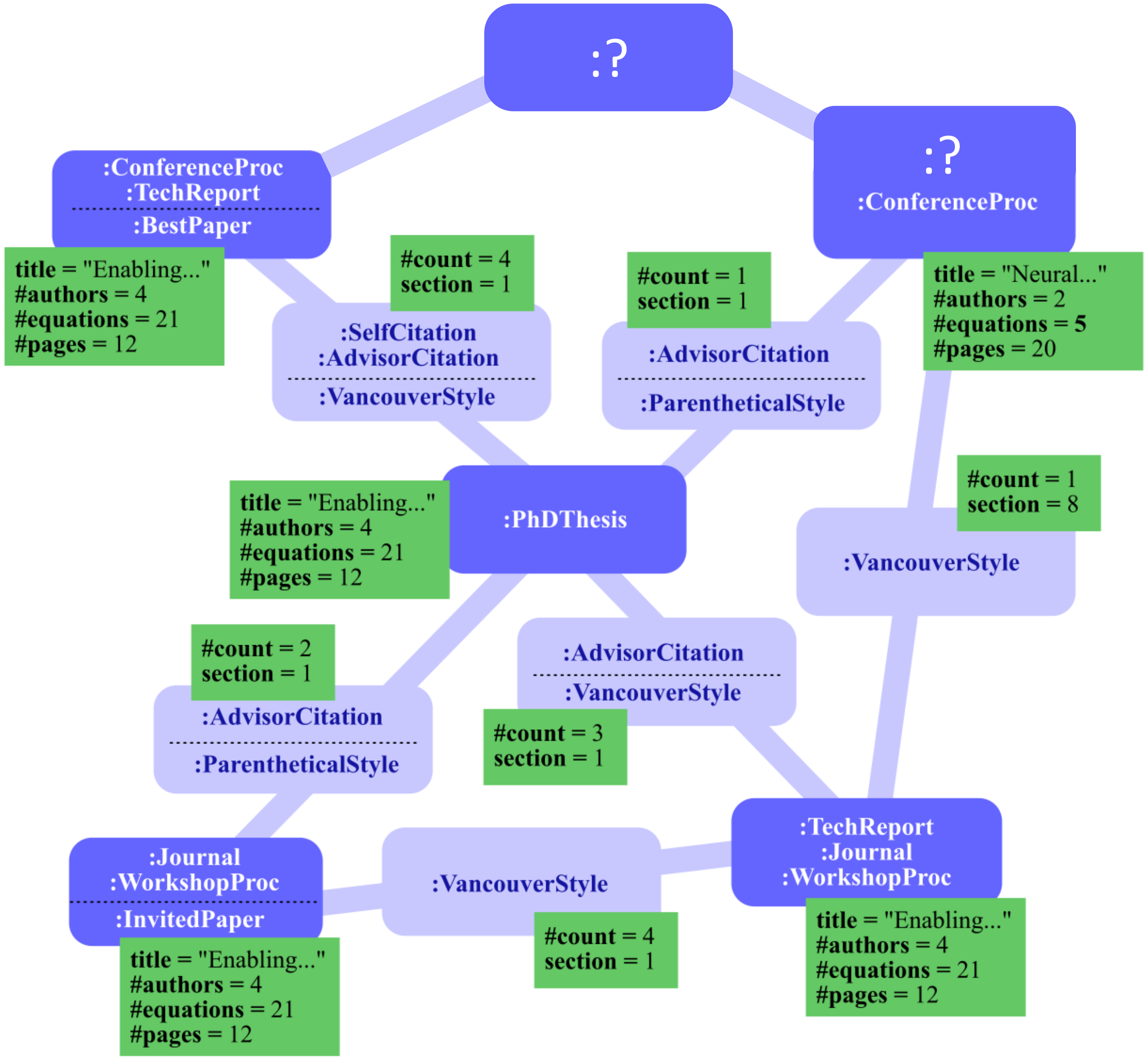
Marrying Graph Databases and GNNs



There is a neat way to think about the GNN workloads within the LPG model

Node/edge/graph classification becomes ***label prediction***

Marrying Graph Databases and GNNs



There is a neat way to think about the GNN workloads within the LPG model

Node/edge/graph classification becomes ***label prediction***

Marrying Graph Databases and GNNs

There is a neat way to think about the GNN workloads within the LPG model

Node/edge/graph classification becomes ***label prediction***

Marrying Graph Databases and GNNs

There is a neat way to think about the GNN workloads within the LPG model

Node/edge/graph classification becomes ***label prediction***

Node/edge/graph regression becomes ***property prediction***

Marrying Graph Databases and GNNs

There is a neat way to think about the GNN workloads within the LPG model

Node/edge/graph classification becomes ***label prediction***

Node/edge/graph regression becomes ***property prediction***

Marrying Graph Databases and GNNs

There is a neat way to think about the GNN workloads within the LPG model

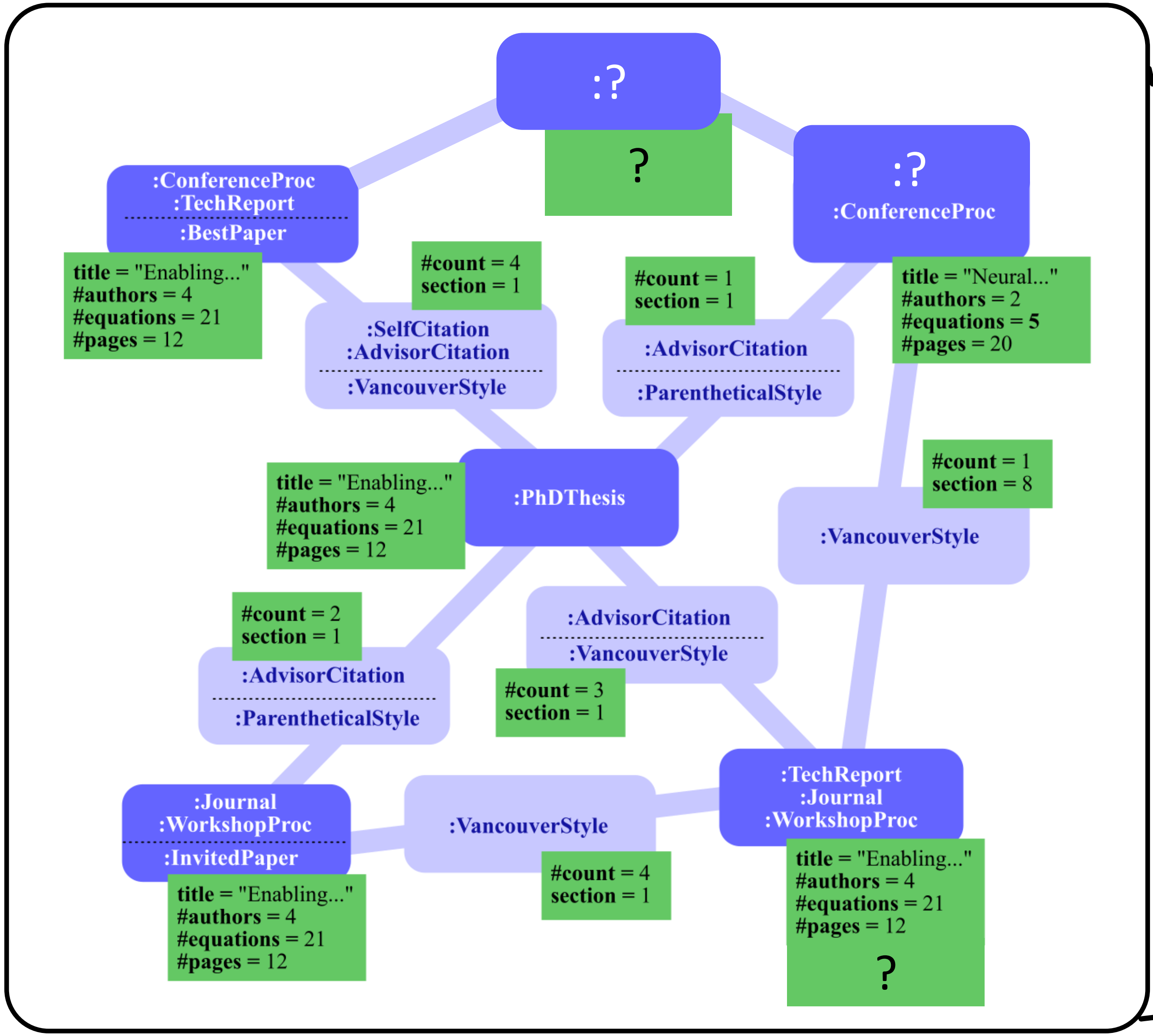
Node/edge/graph classification becomes ***label prediction***

Node/edge/graph regression becomes ***property prediction***

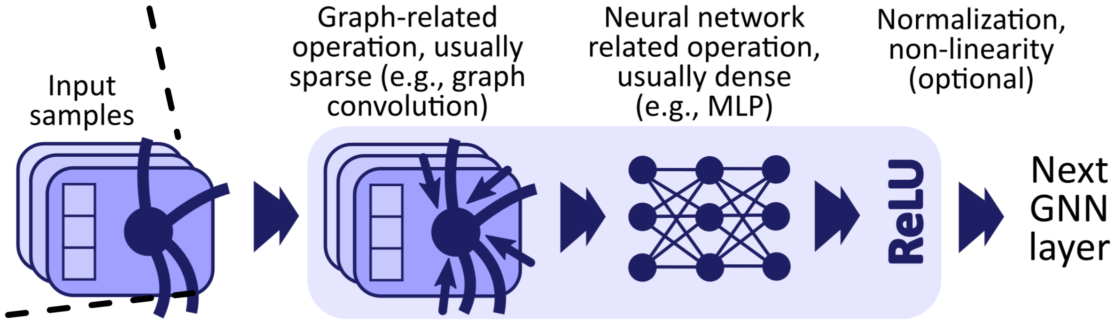
Marrying Graph Databases and GNNs

How do we enable these GNN workloads on the LPG graph datasets **while harnessing all the existing label/property information?**

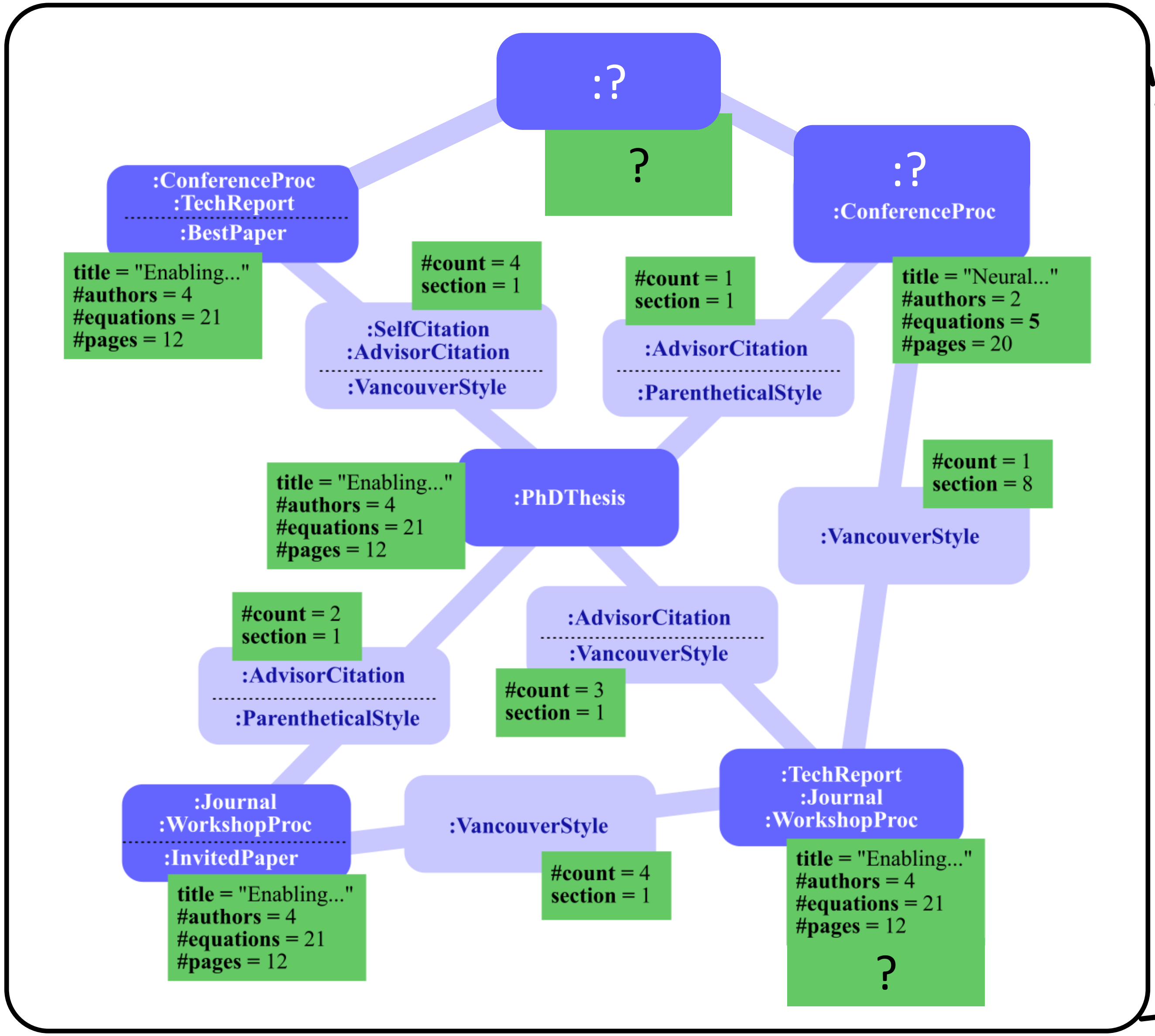
Marrying Graph Databases and GNNs



How do we enable these GNN workloads on the LPG graph datasets **while harnessing all the existing label/property information?**

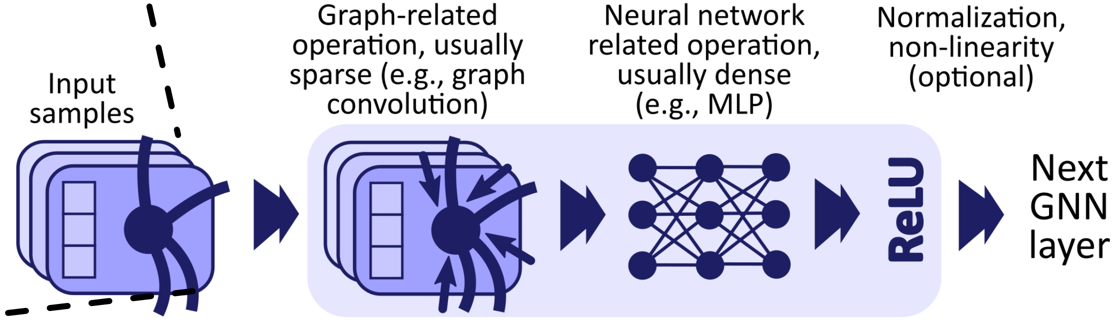


Marrying Graph Databases and GNNs



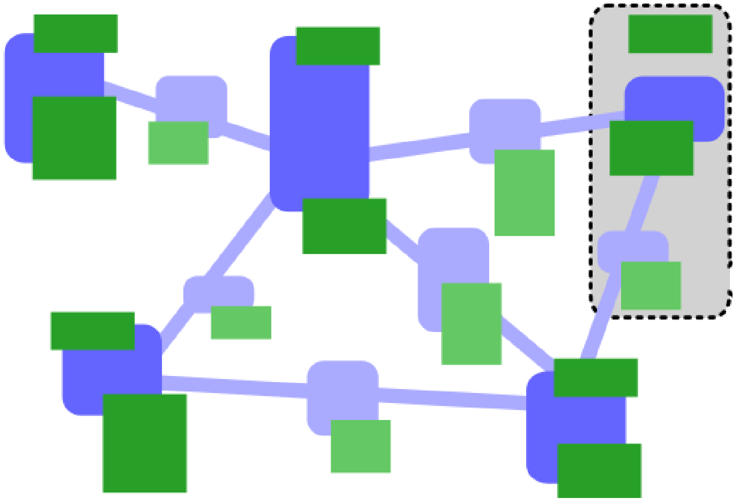
How do we enable these GNN workloads on the LPG graph datasets **while harnessing all the existing label/property information?**

We need the right encoder!

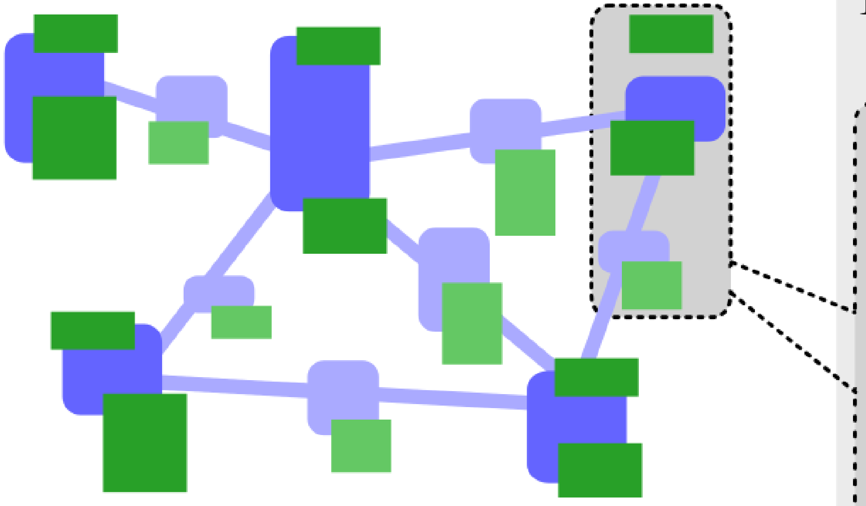


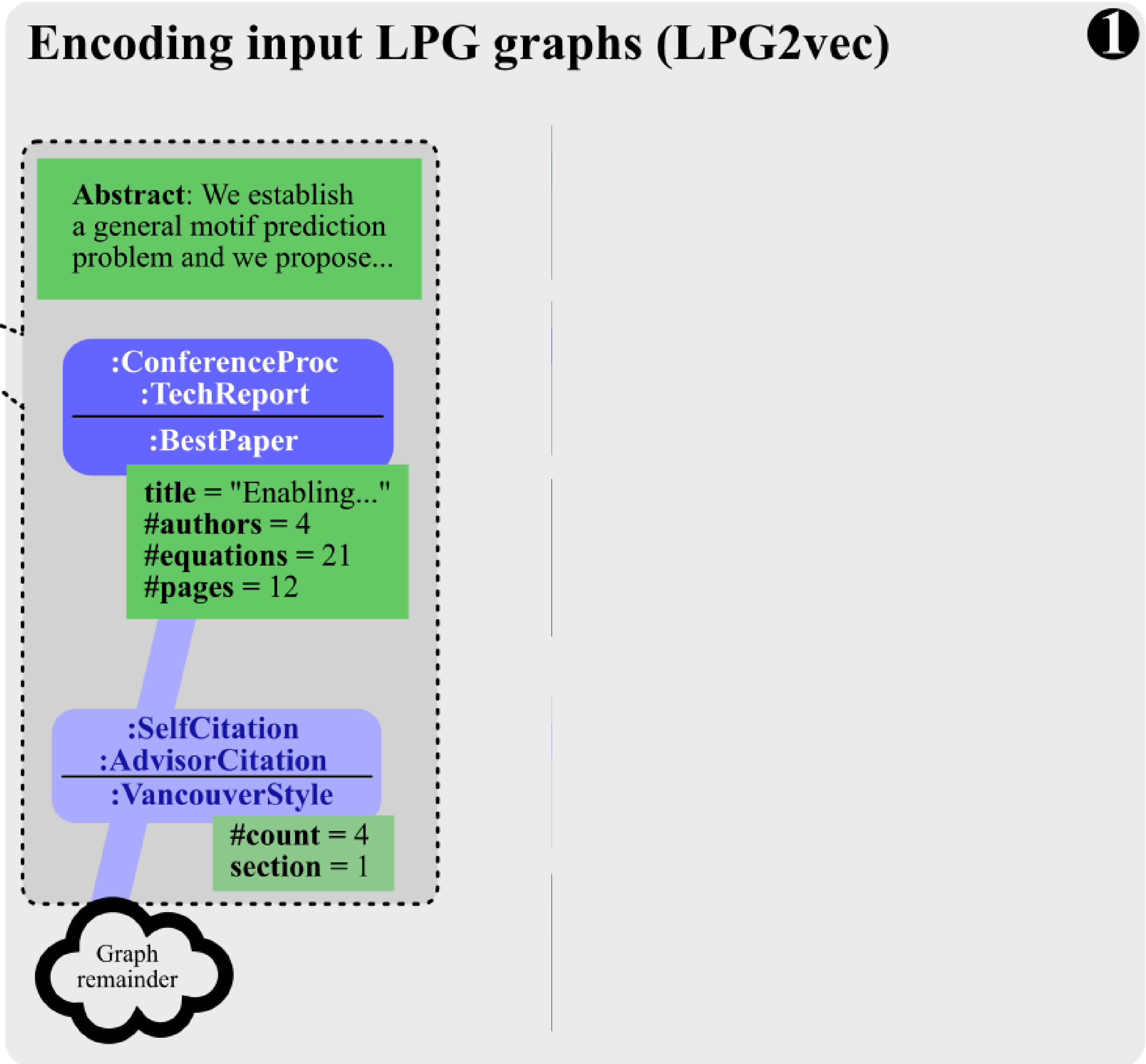
LPG2vec: Encoding LPG Datasets into a Format Digestible by GNNs Pipelines

LPG2vec: Encoding LPG Datasets into a Format Digestible by GNNs Pipelines

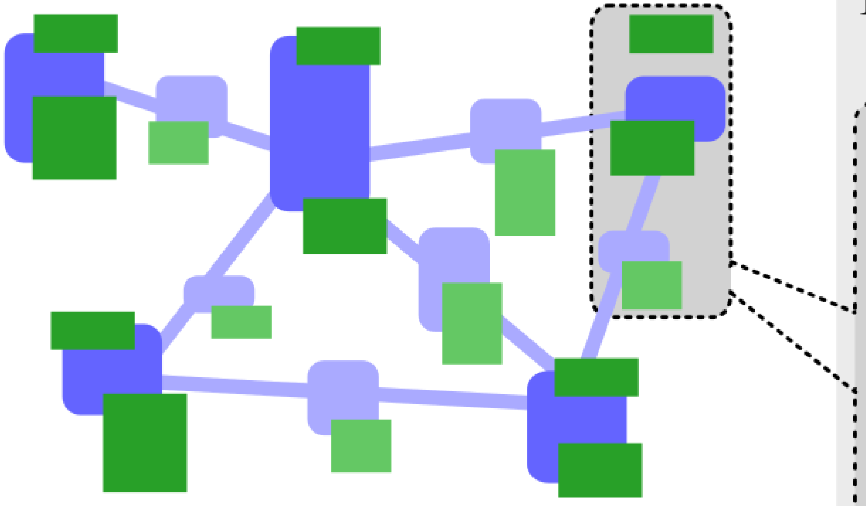


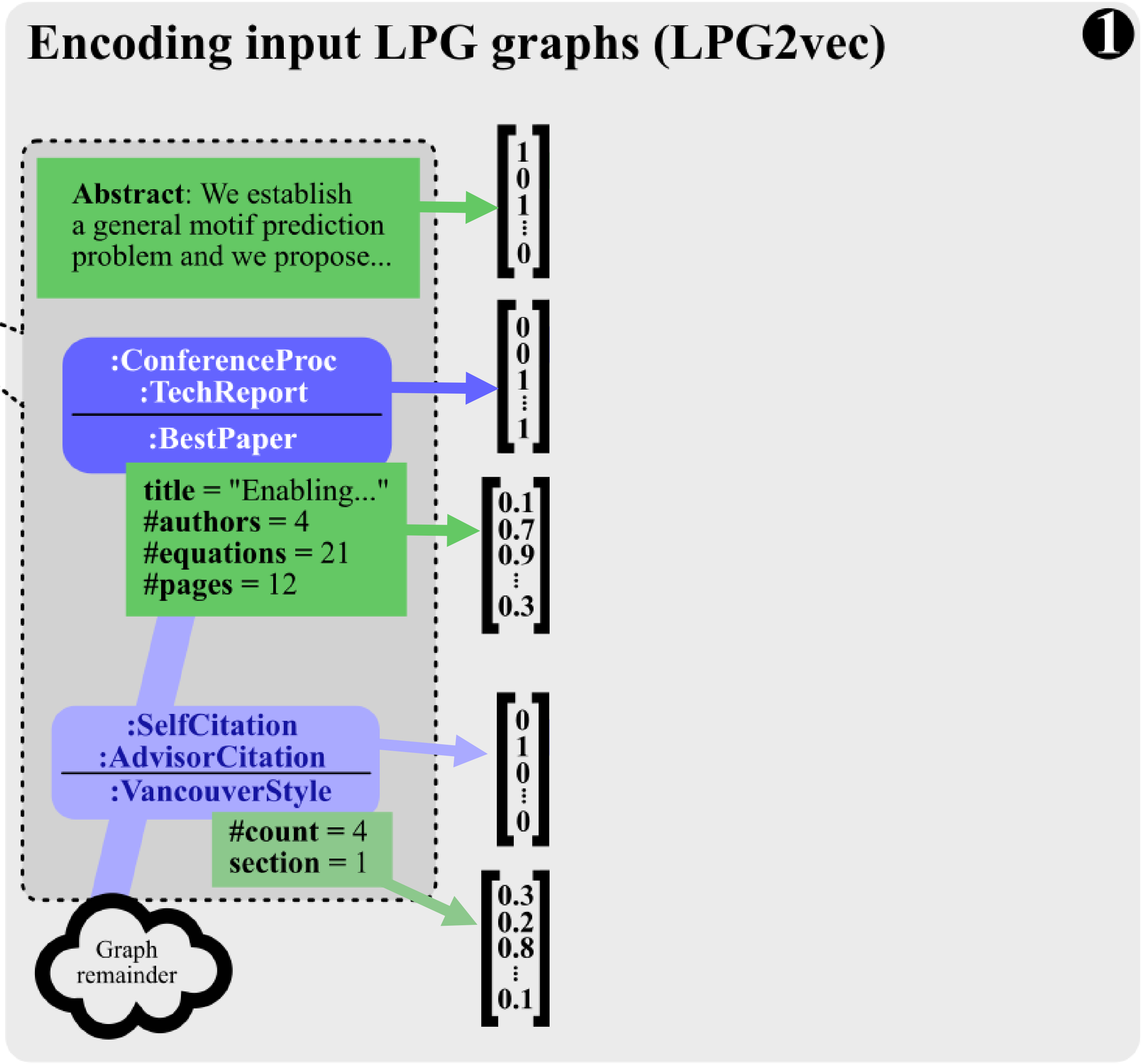
LPG2vec: Encoding LPG Datasets into a Format Digestible by GNNs Pipelines



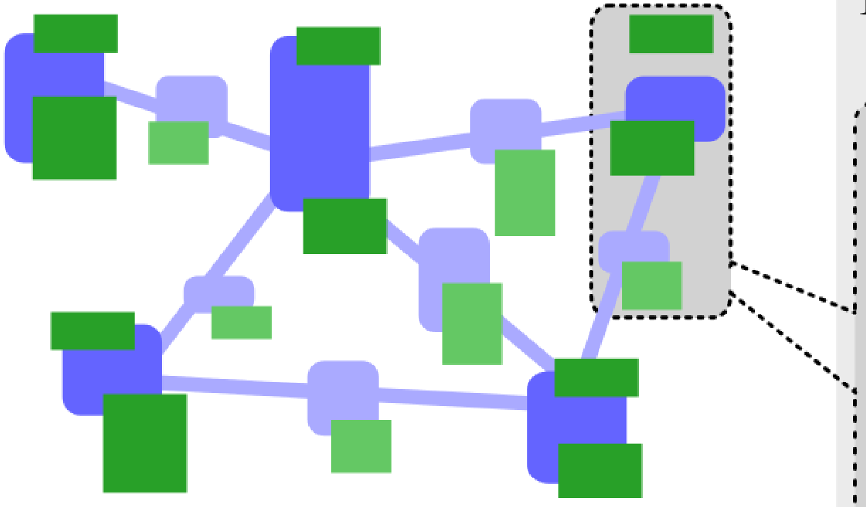


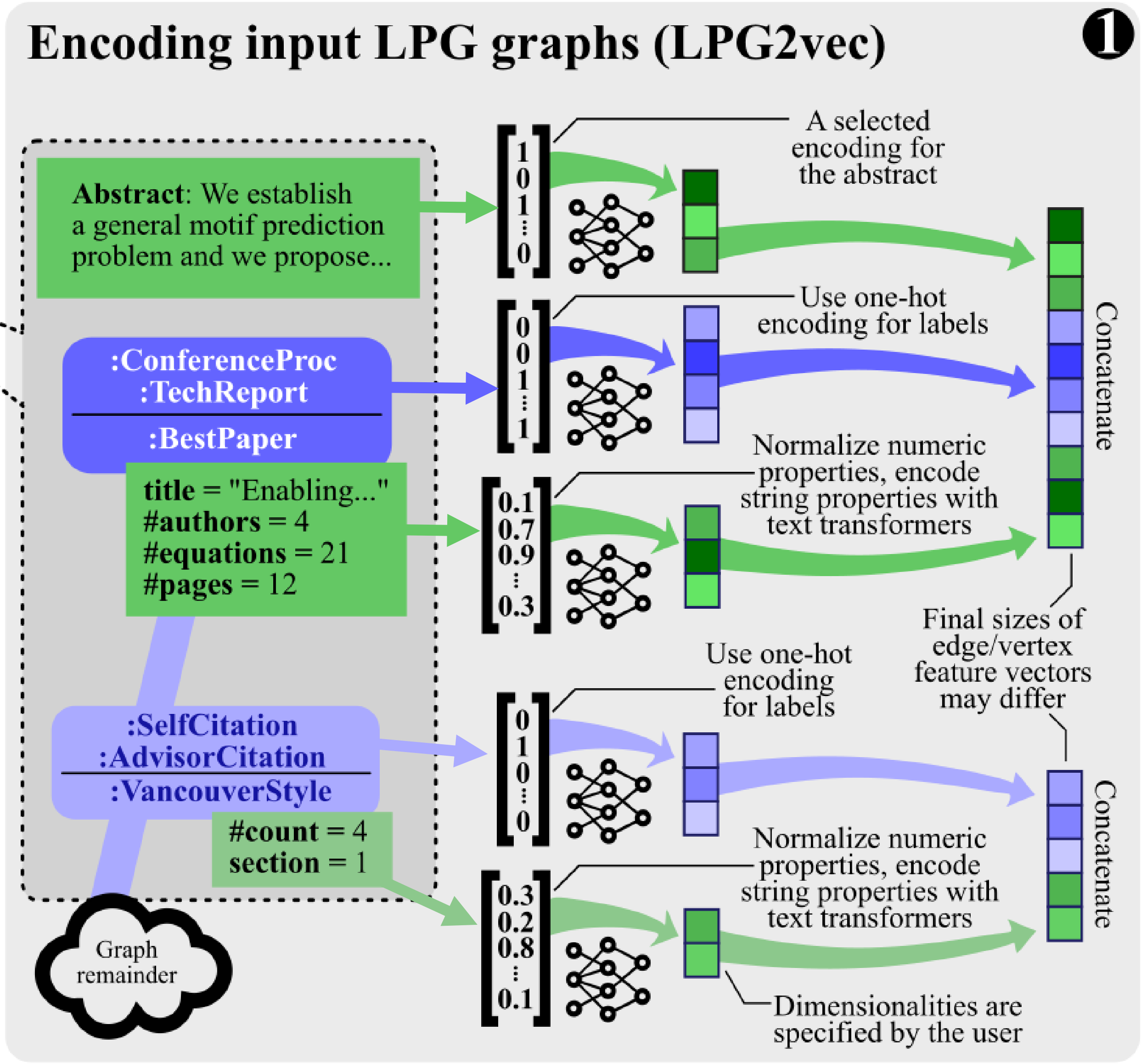
LPG2vec: Encoding LPG Datasets into a Format Digestible by GNNs Pipelines



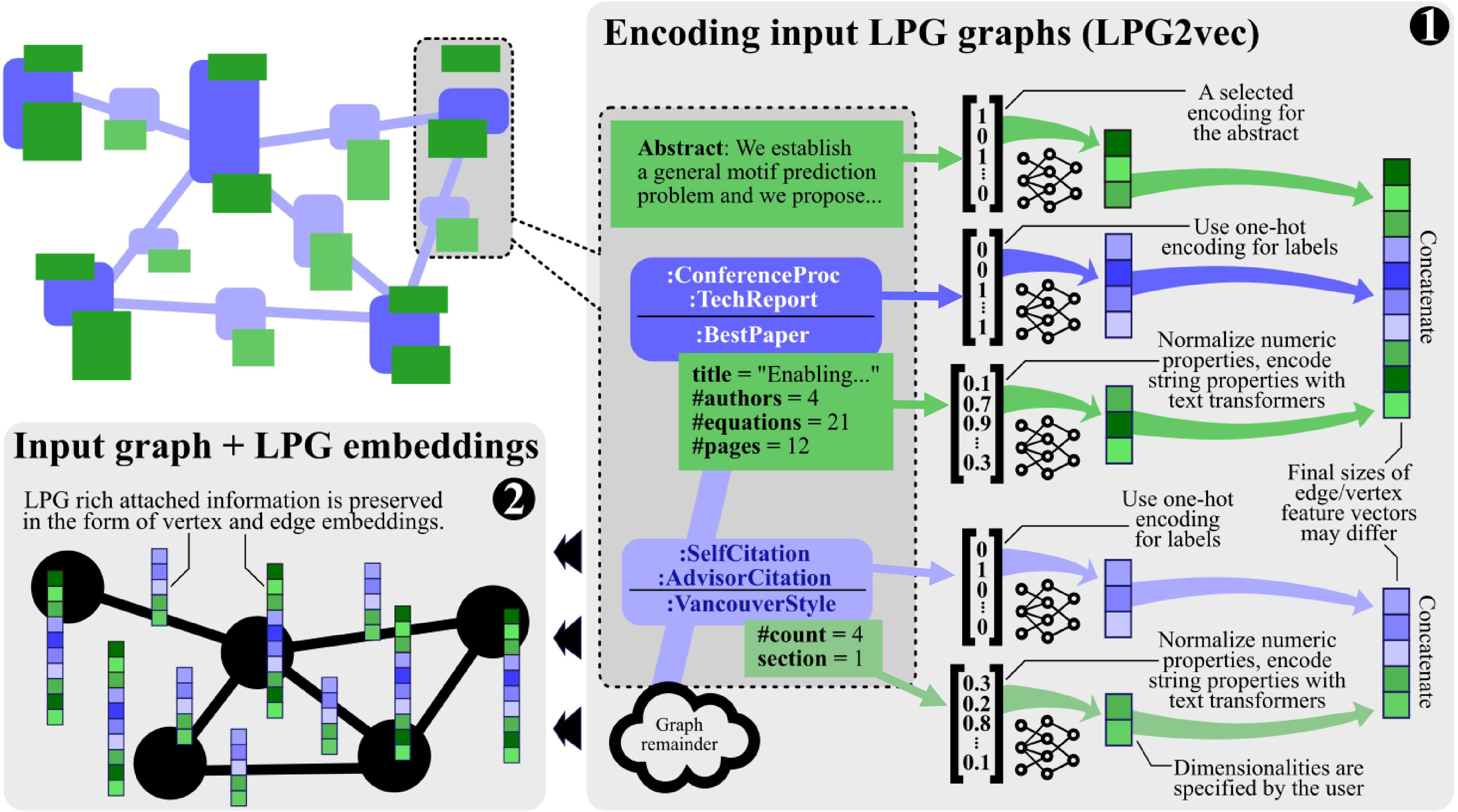


LPG2vec: Encoding LPG Datasets into a Format Digestible by GNNs Pipelines

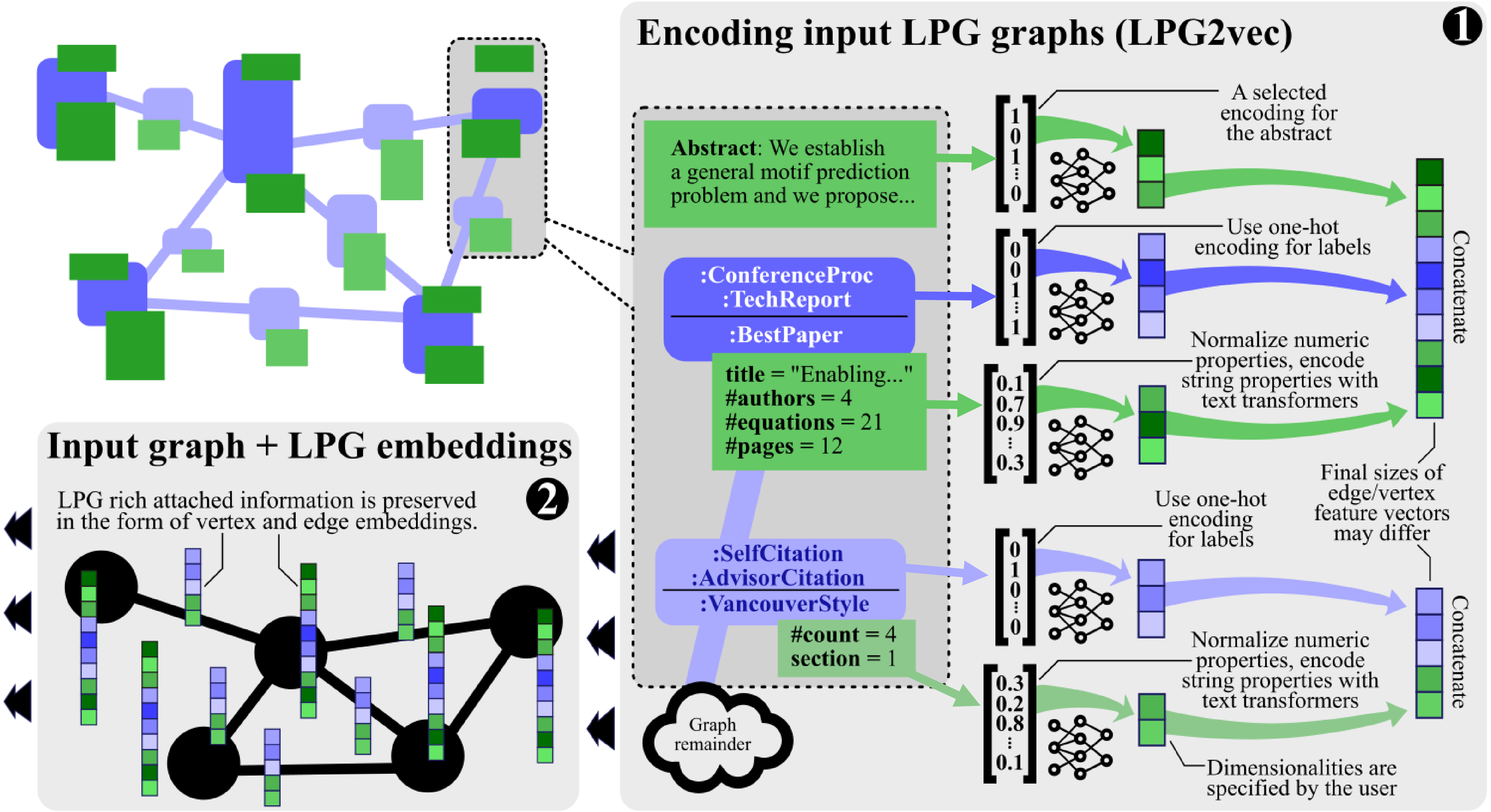




LPG2vec: Encoding LPG Datasets into a Format Digestible by GNNs Pipelines



LPG2vec: Encoding LPG Datasets into a Format Digestible by GNNs Pipelines



PyG

DEEP GRAPH LIBRARY

TF_GEOMETRIC

Evaluation: Used Machine & Objectives

Evaluation: Used Machine & Objectives

CSCS Cray Piz Daint & Ault
64GB – 2TB memory per server

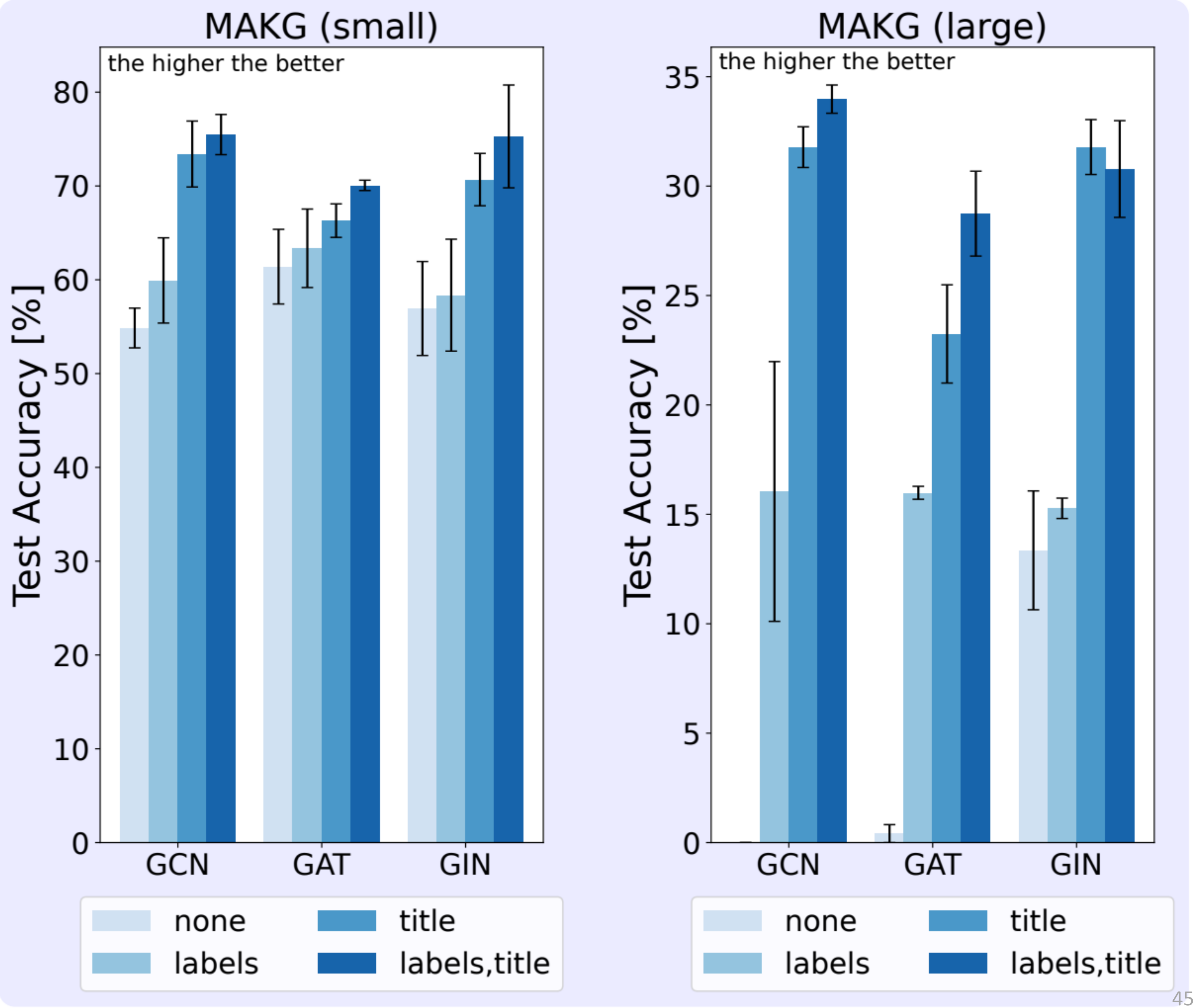
Evaluation: Used Machine & Objectives

CSCS Cray Piz Daint & Ault
64GB – 2TB memory per server

Main goal: show that **LPG2vec successfully harnesses the label and property information** from the LPG graph datasets to offer more accurate predictions in graph ML tasks

Node Classification, aka Label Prediction

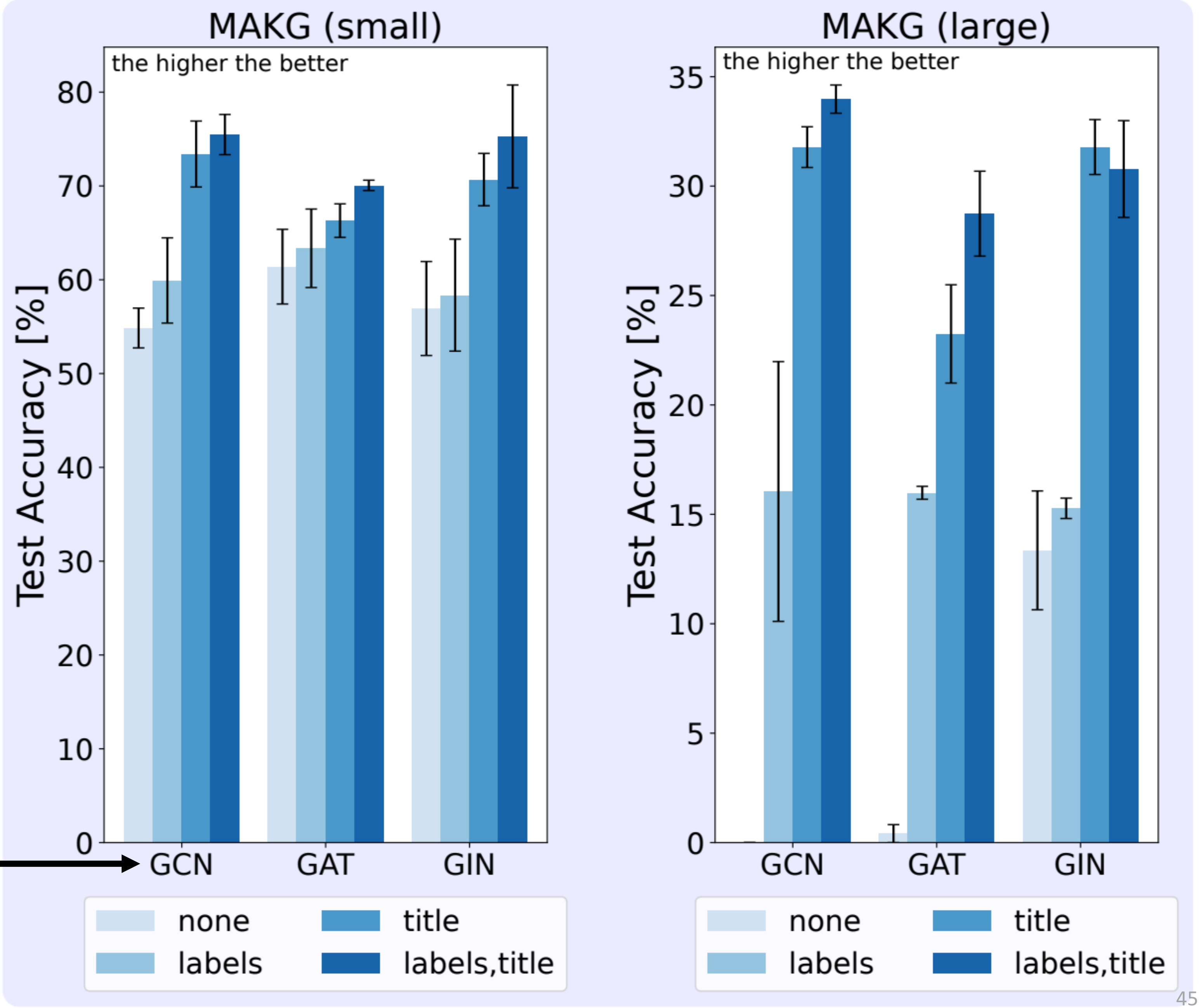
Task: predict the research area of the publication



Node Classification, aka Label Prediction

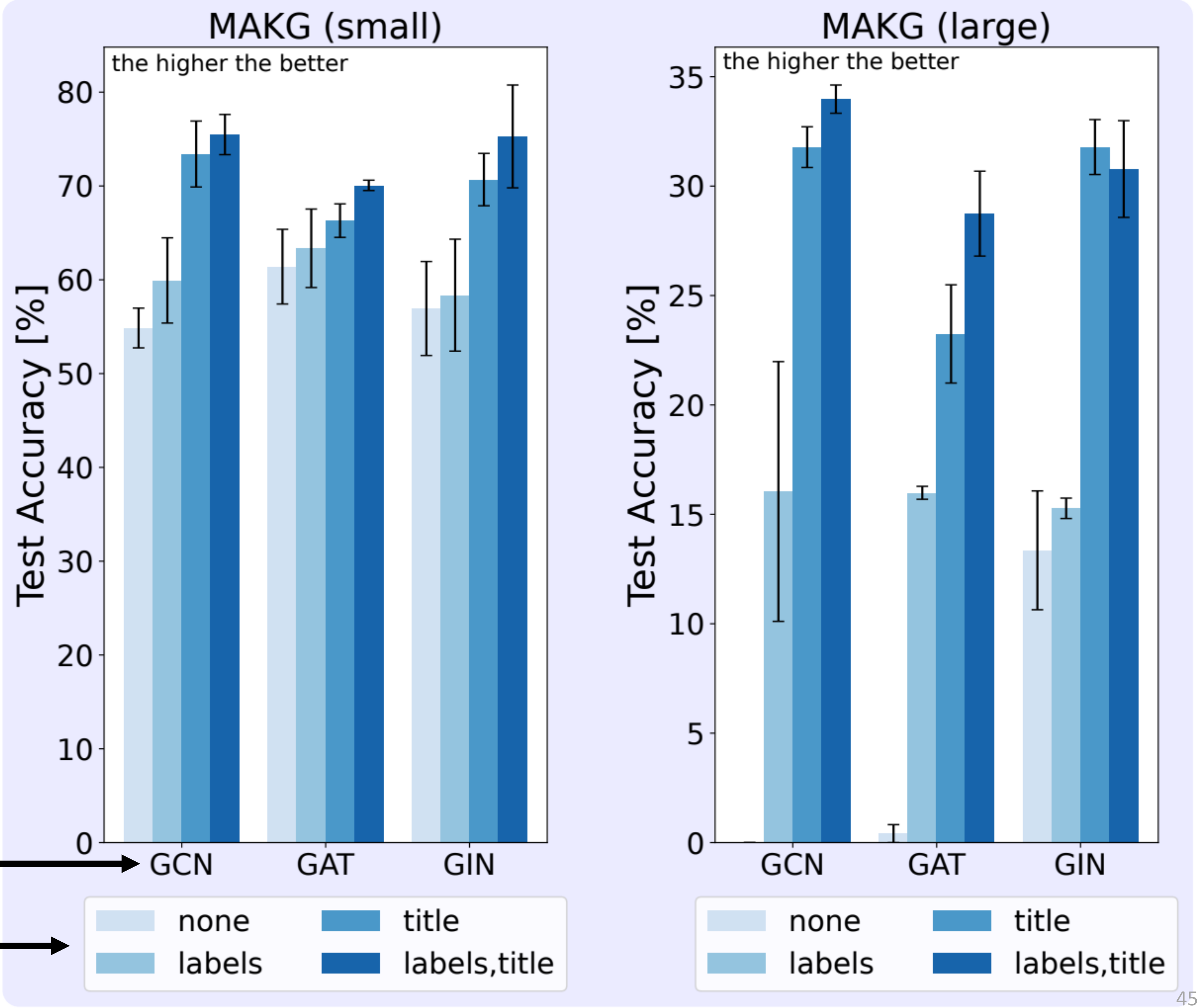
Task: predict the research area of the publication

What GNN model is considered? →



Node Classification, aka Label Prediction

Task: predict the research area of the publication



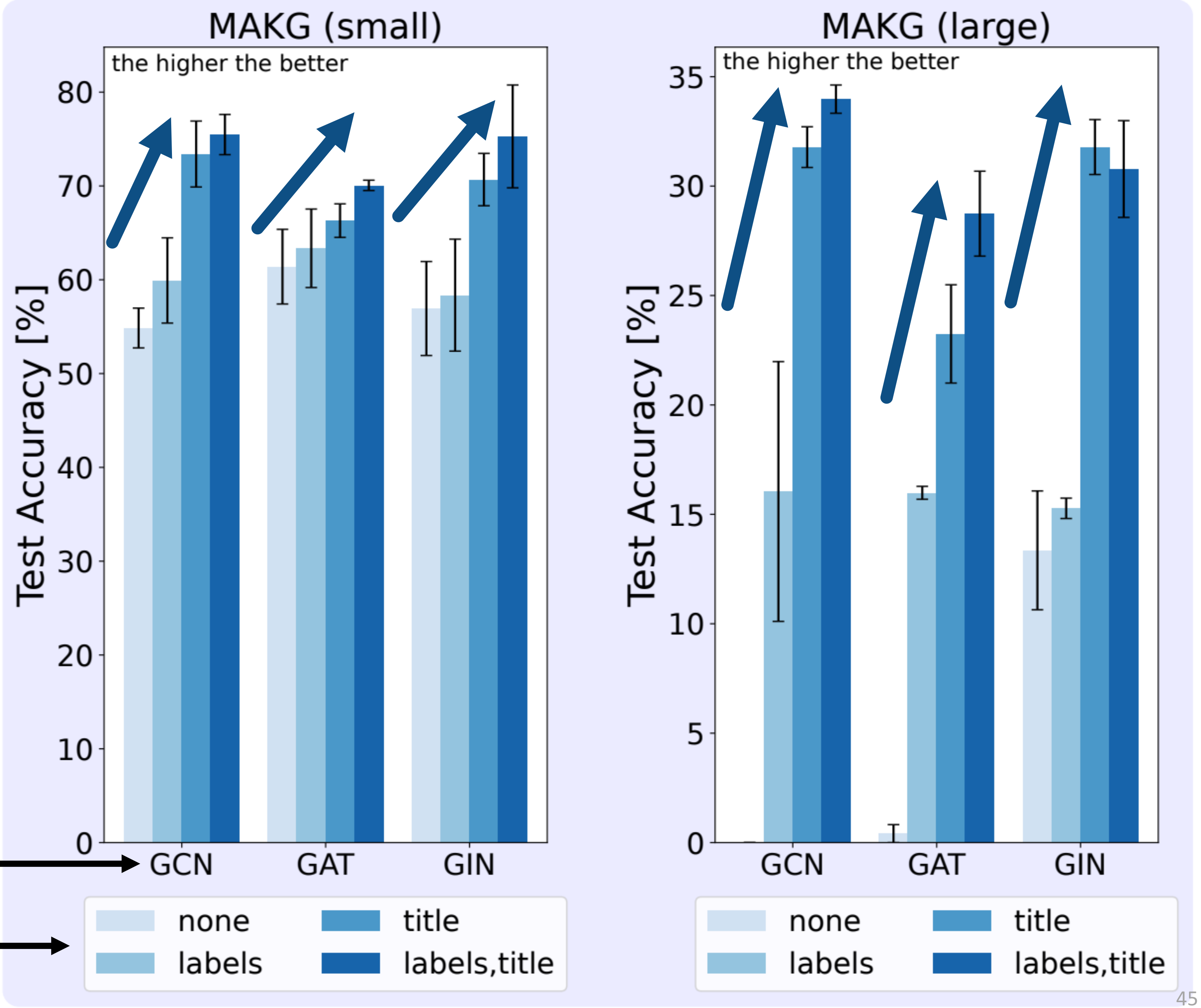
What GNN model is considered?

What label/property data is considered?

Node Classification, aka Label Prediction

Task: predict the research area of the publication

Better accuracy with labels/properties, it is important to use them both



What GNN model is considered?

What label/property data is considered?

Node Classification, aka Label Prediction

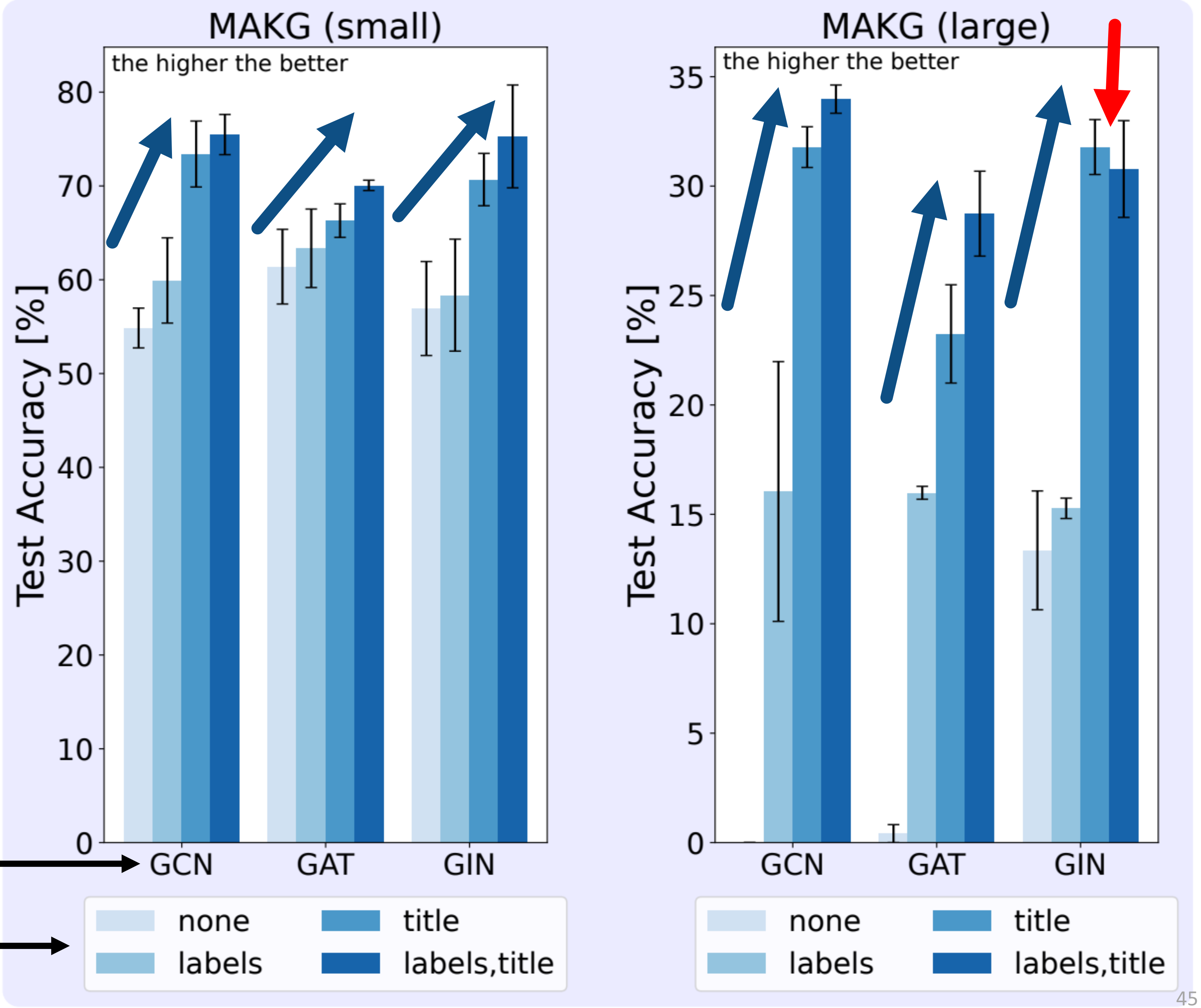
Task: predict the research area of the publication

Better accuracy with labels/properties, it is important to use them both

...not always?

What GNN model is considered?

What label/property data is considered?

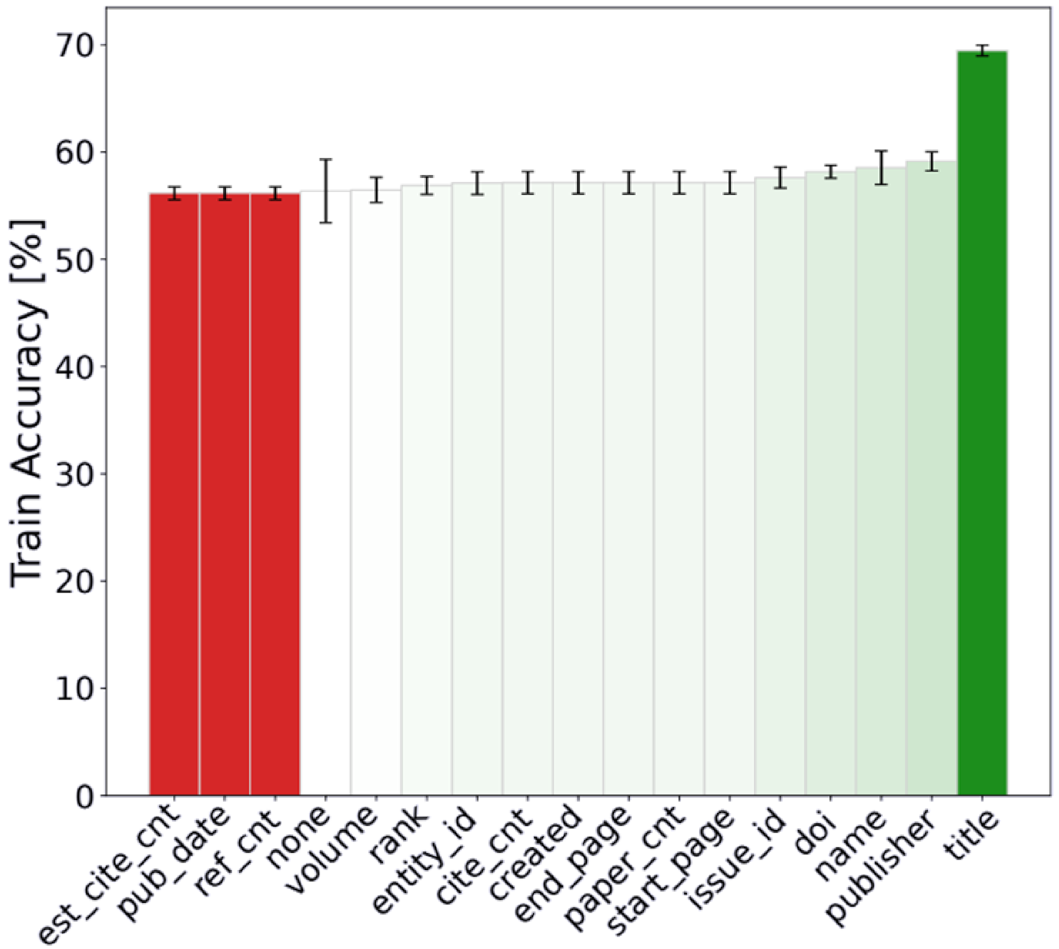


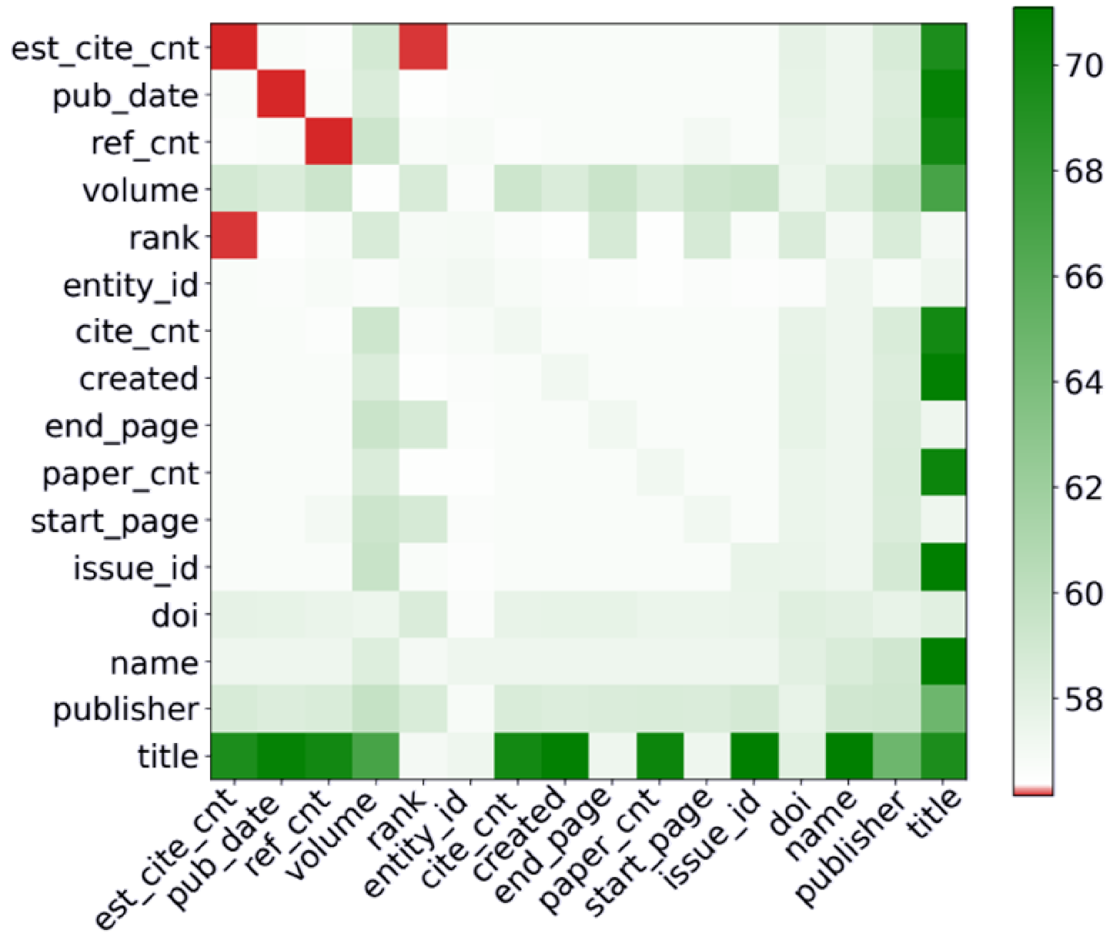
Node Classification, aka Label Prediction

Task: predict the research area of the publication

Node Classification, aka Label Prediction

Task: predict the research area of the publication

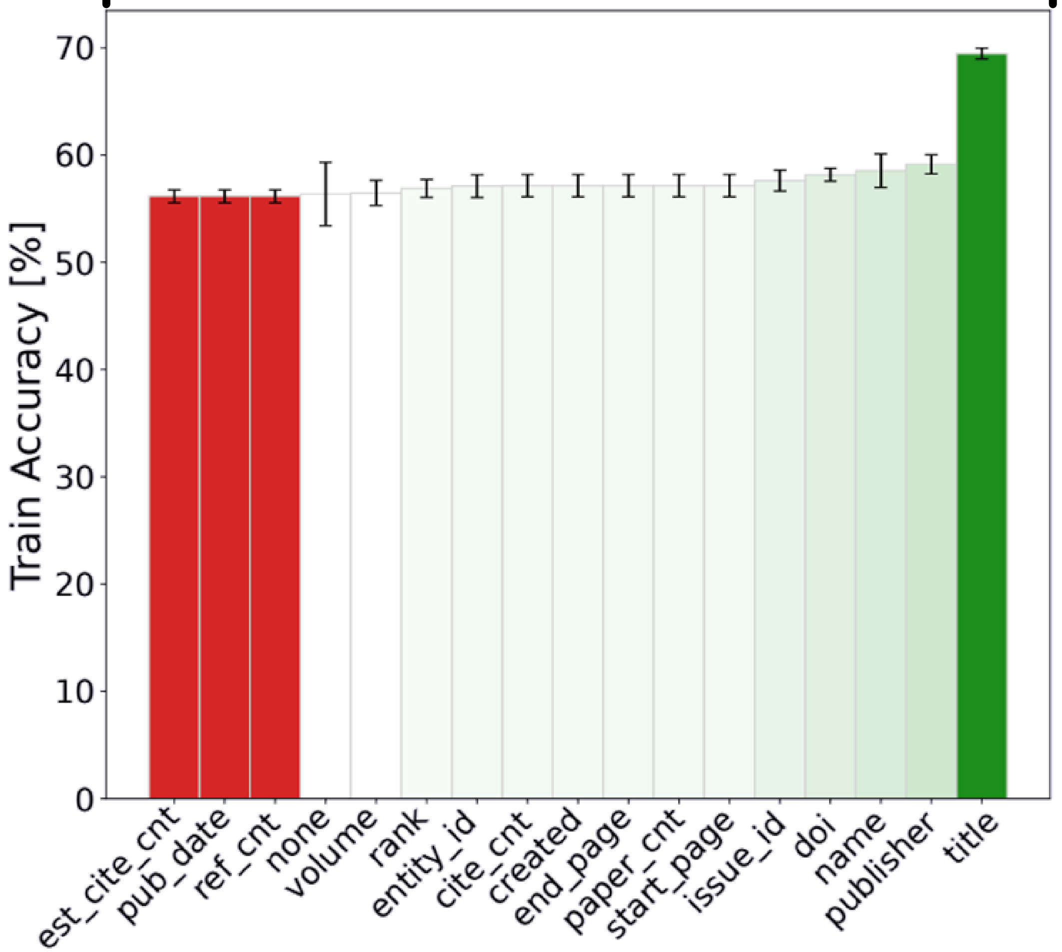


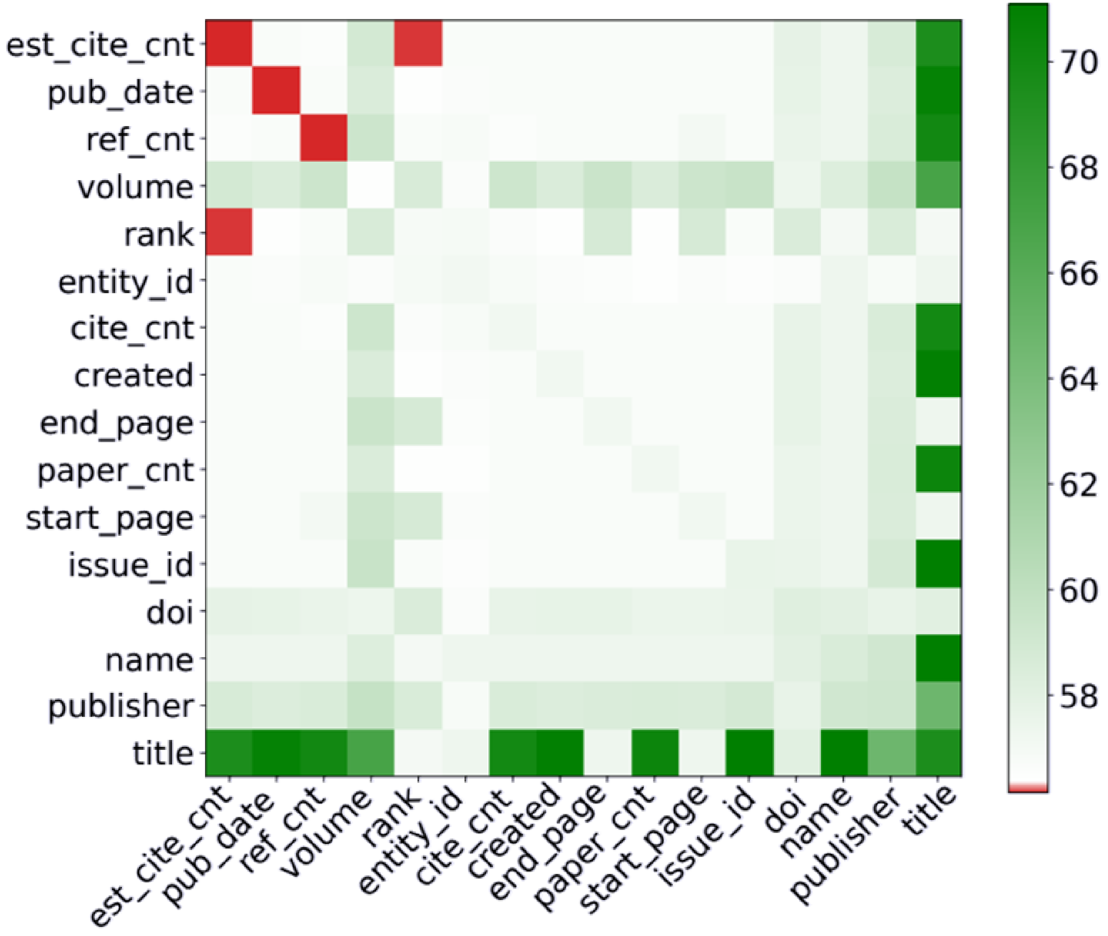


Node Classification, aka Label Prediction

Task: predict the research area of the publication

Impact from each label/property

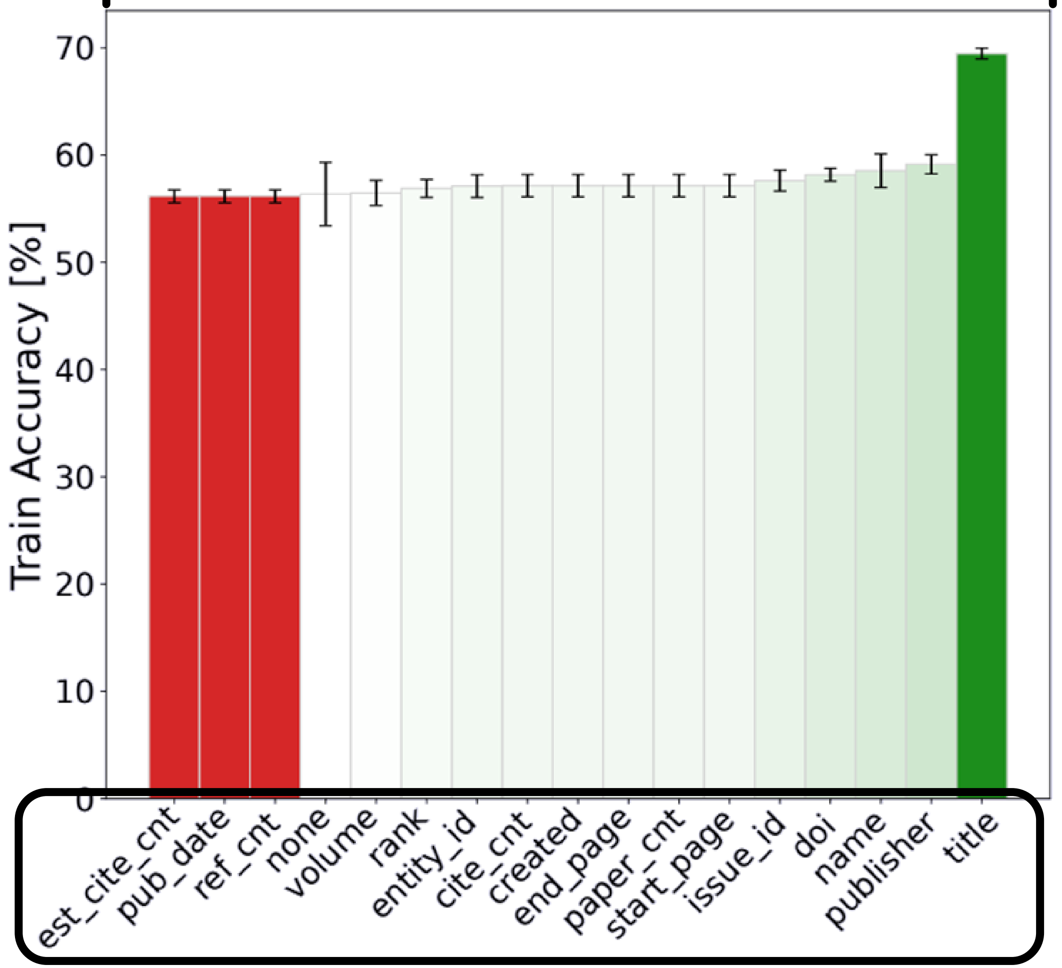


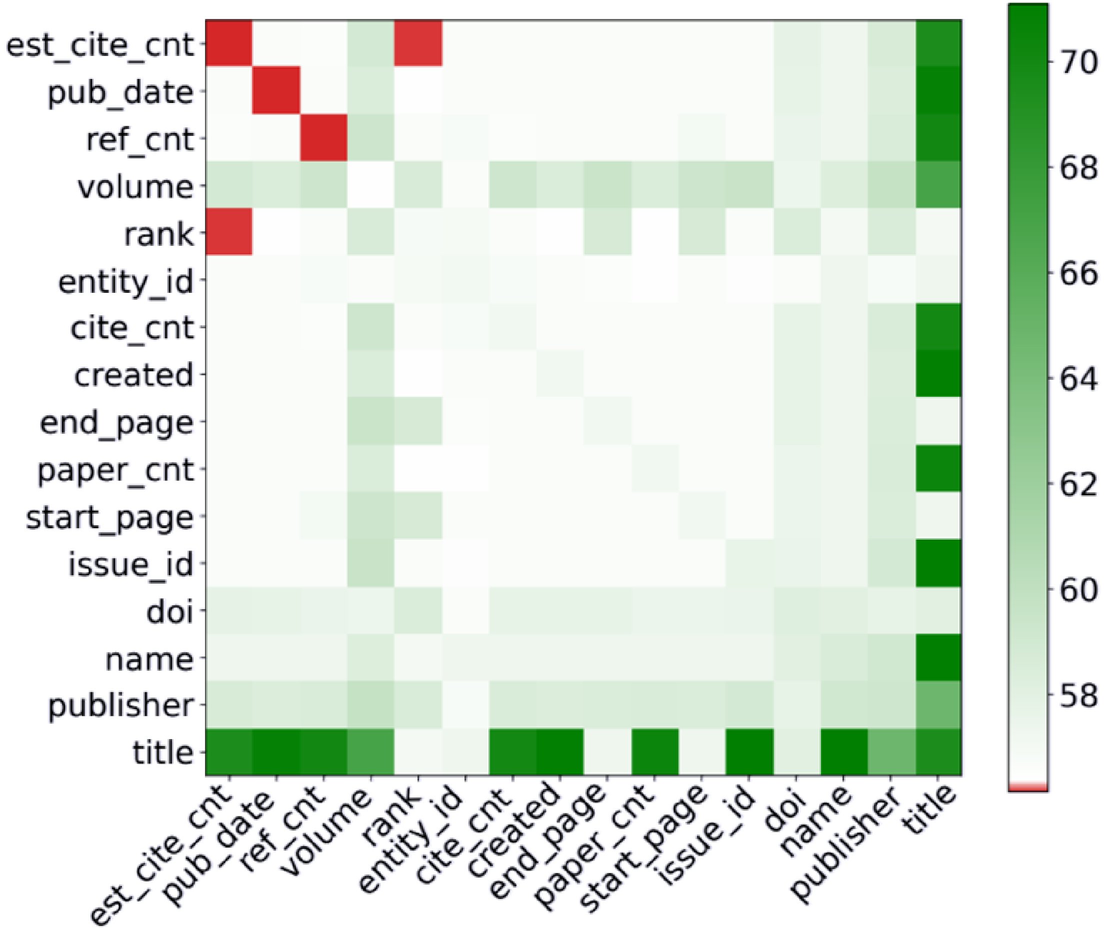


Node Classification, aka Label Prediction

Task: predict the research area of the publication

Impact from each label/property

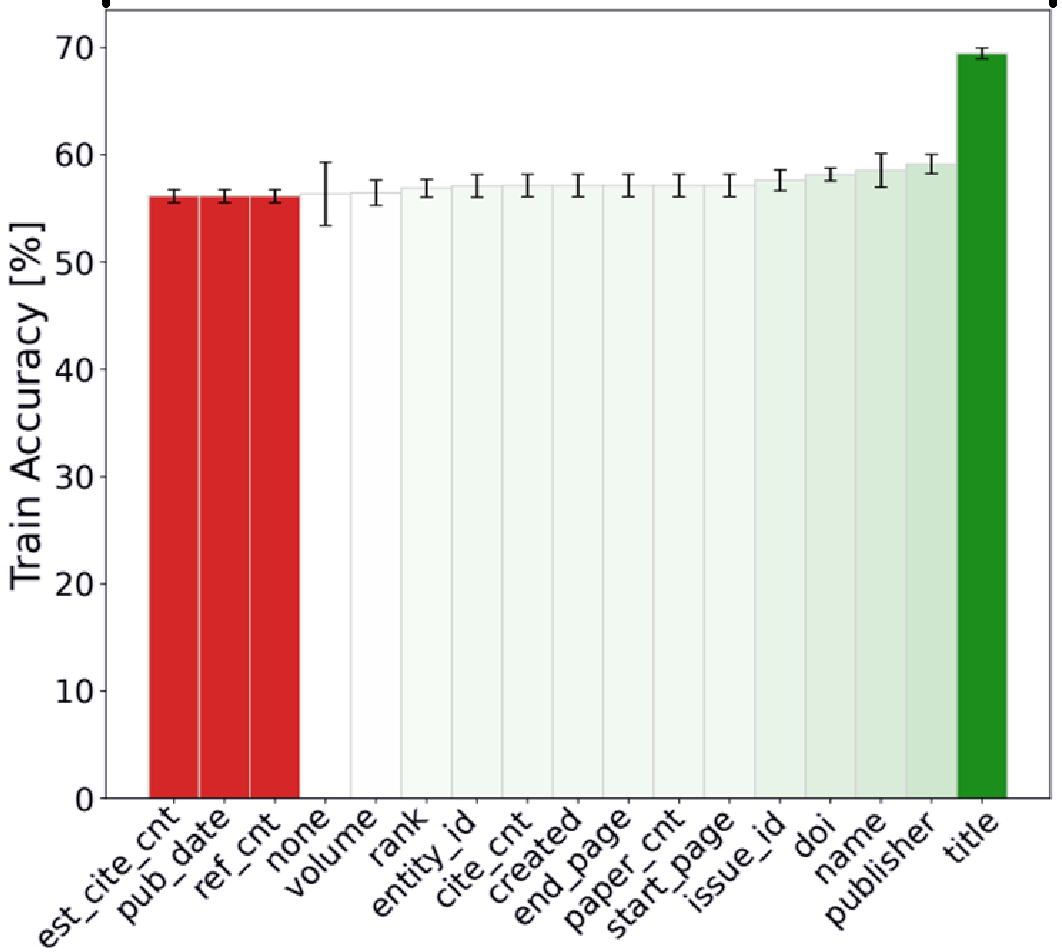


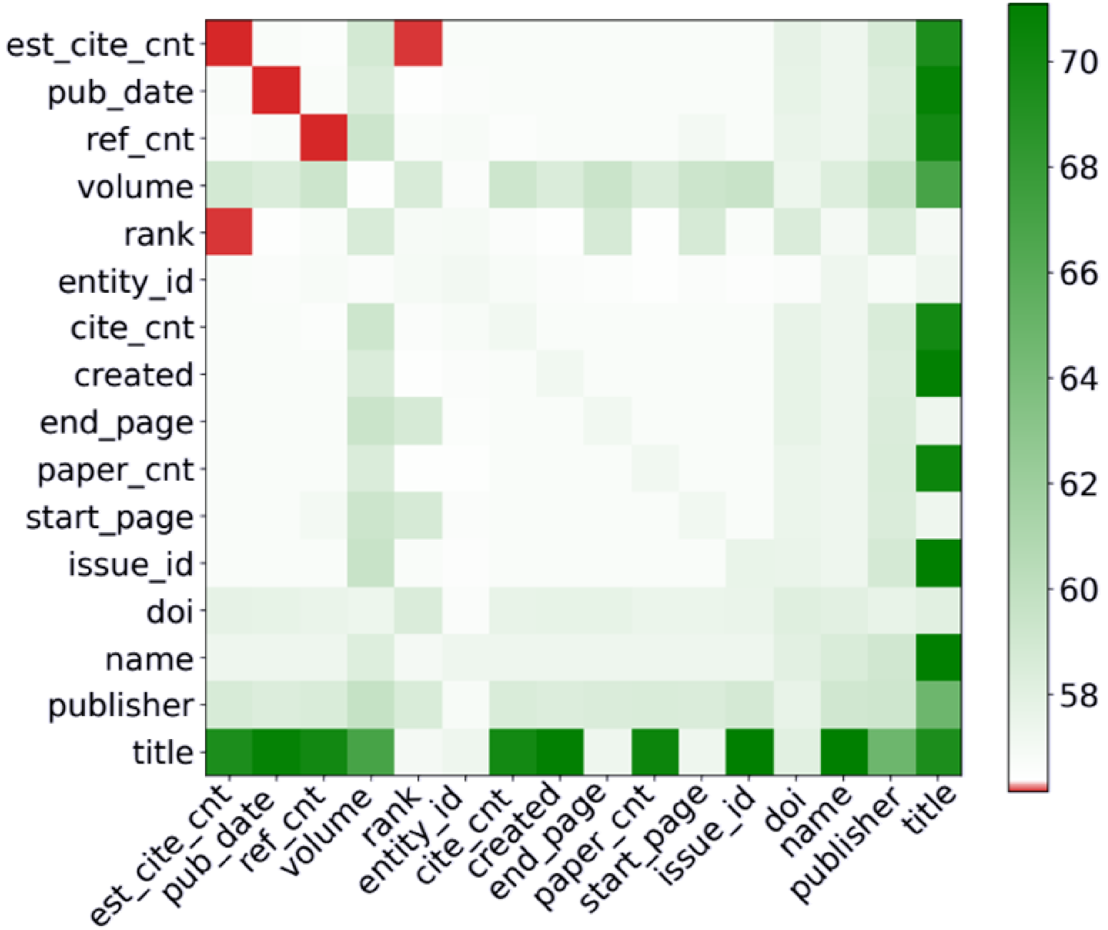


Node Classification, aka Label Prediction

Task: predict the research area of the publication

Impact from each label/property

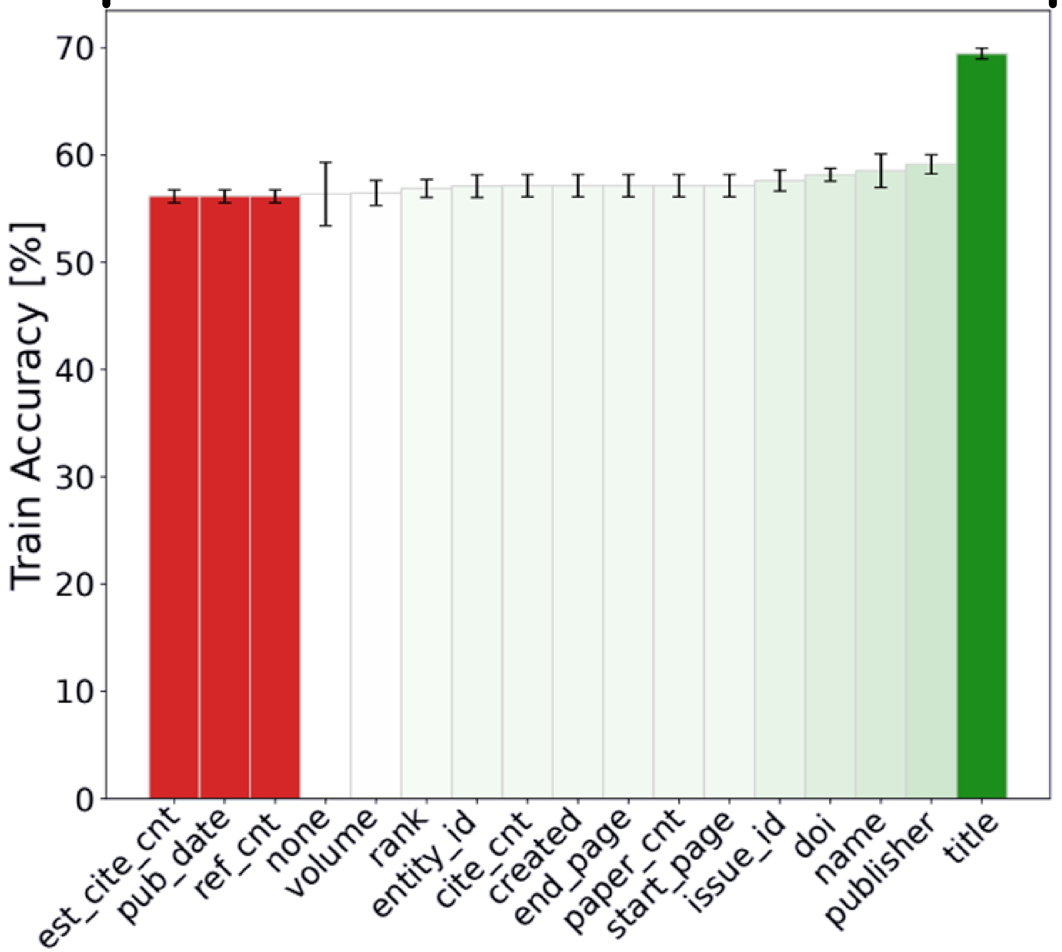




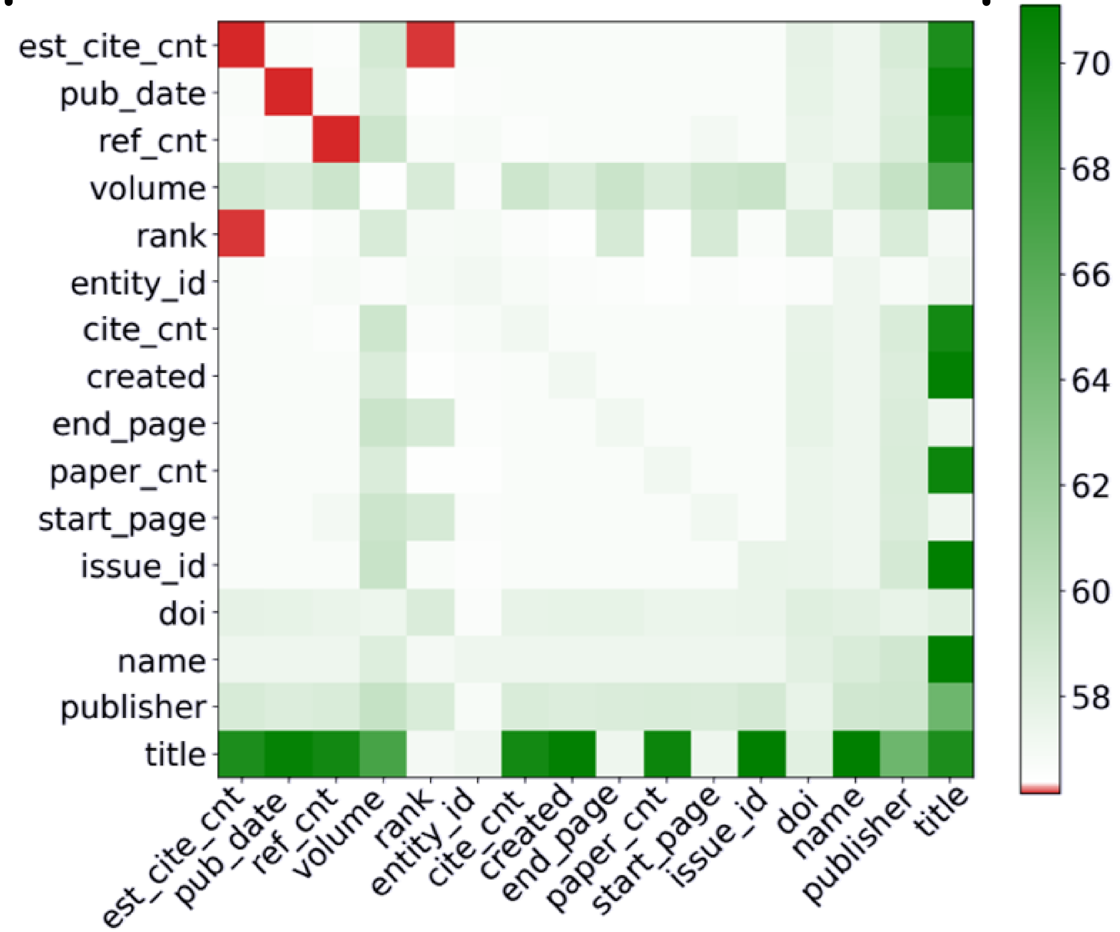
Node Classification, aka Label Prediction

Task: predict the research area of the publication

Impact from each label/property



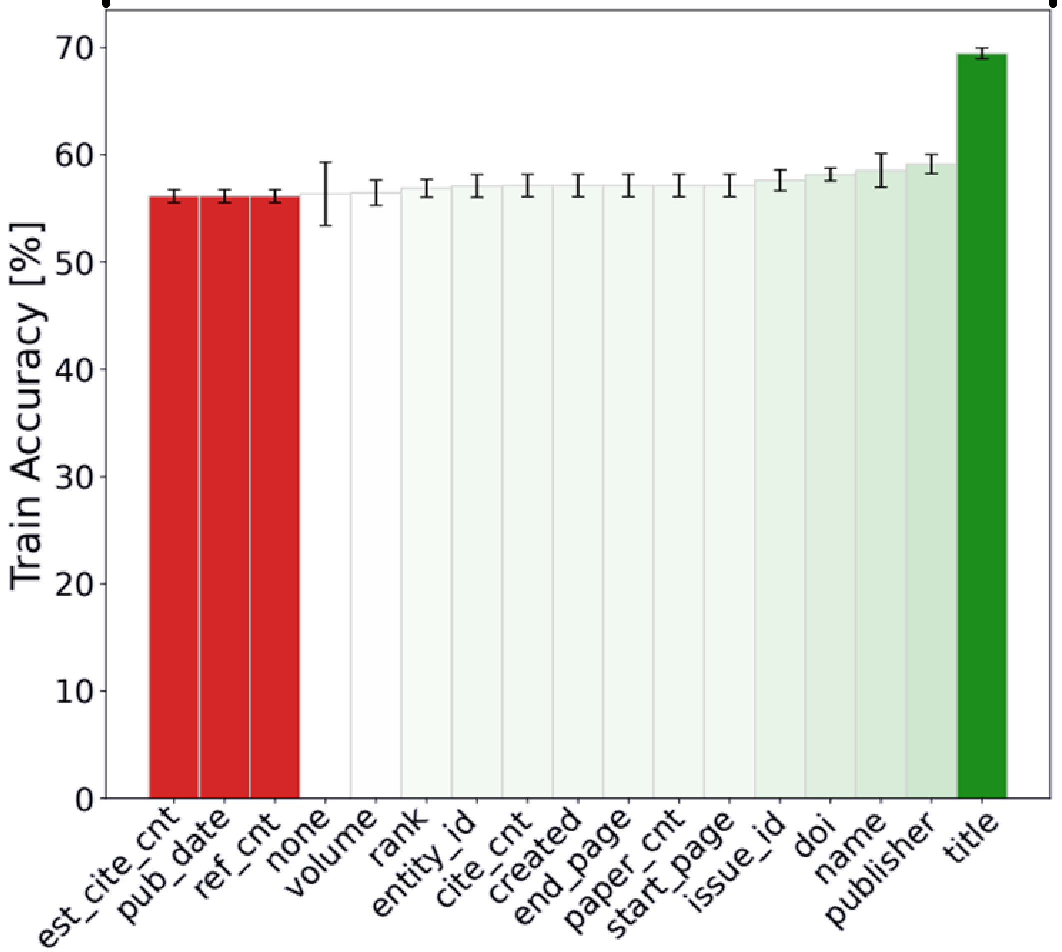
Impact from each label/property **pair**



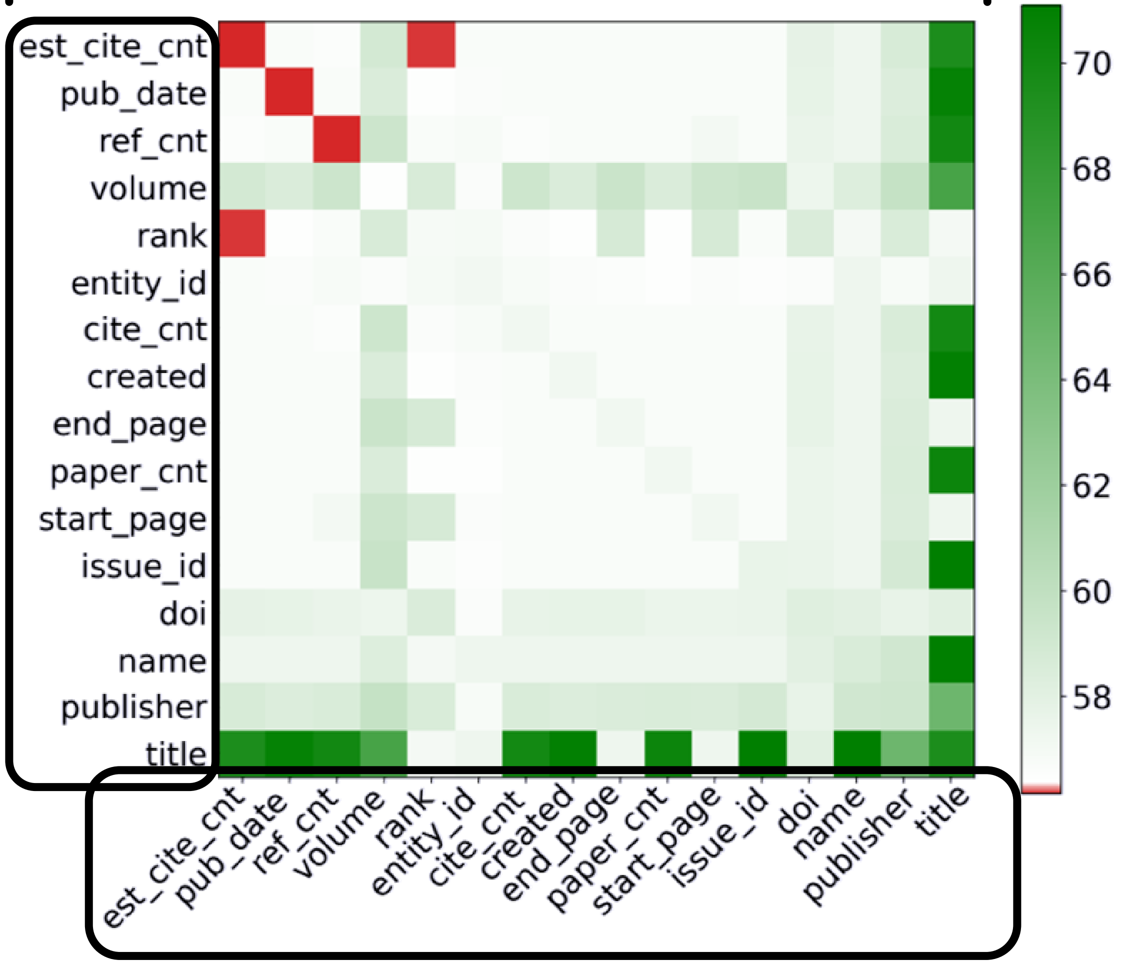
Node Classification, aka Label Prediction

Task: predict the research area of the publication

Impact from each label/property



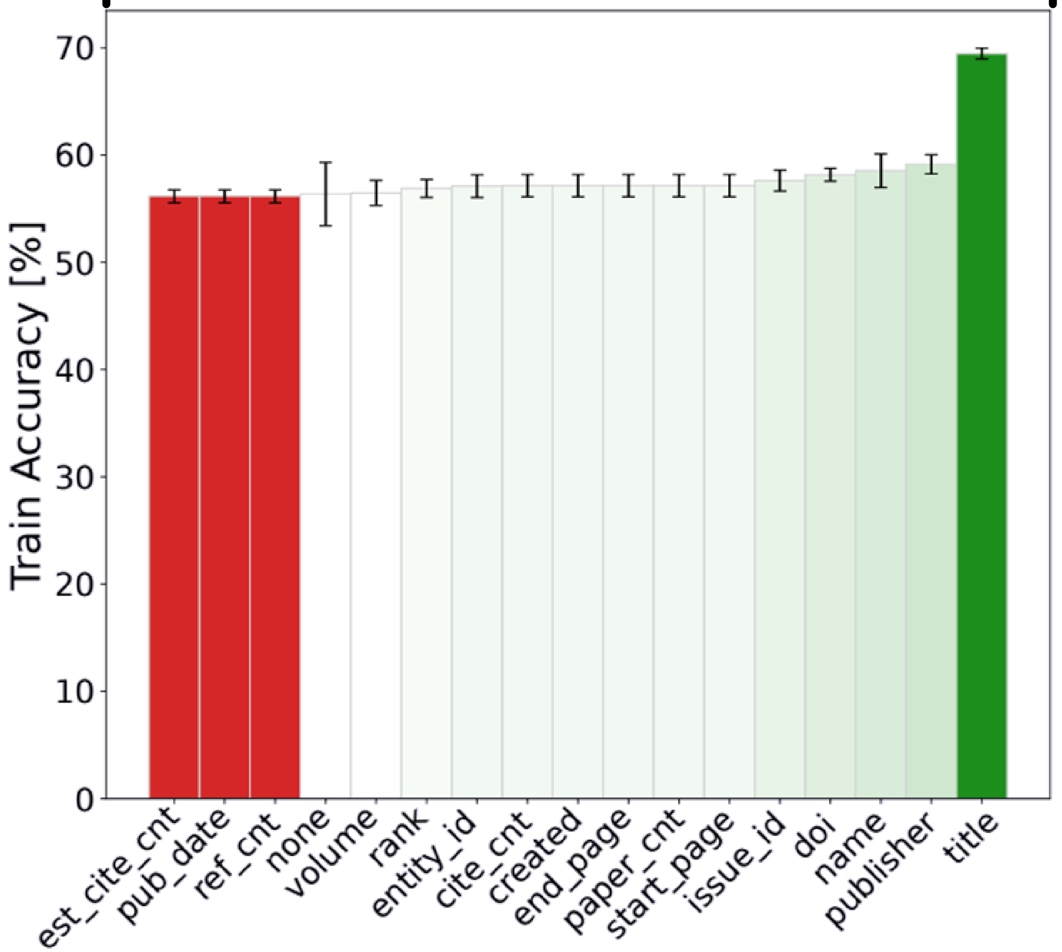
Impact from each label/property **pair**



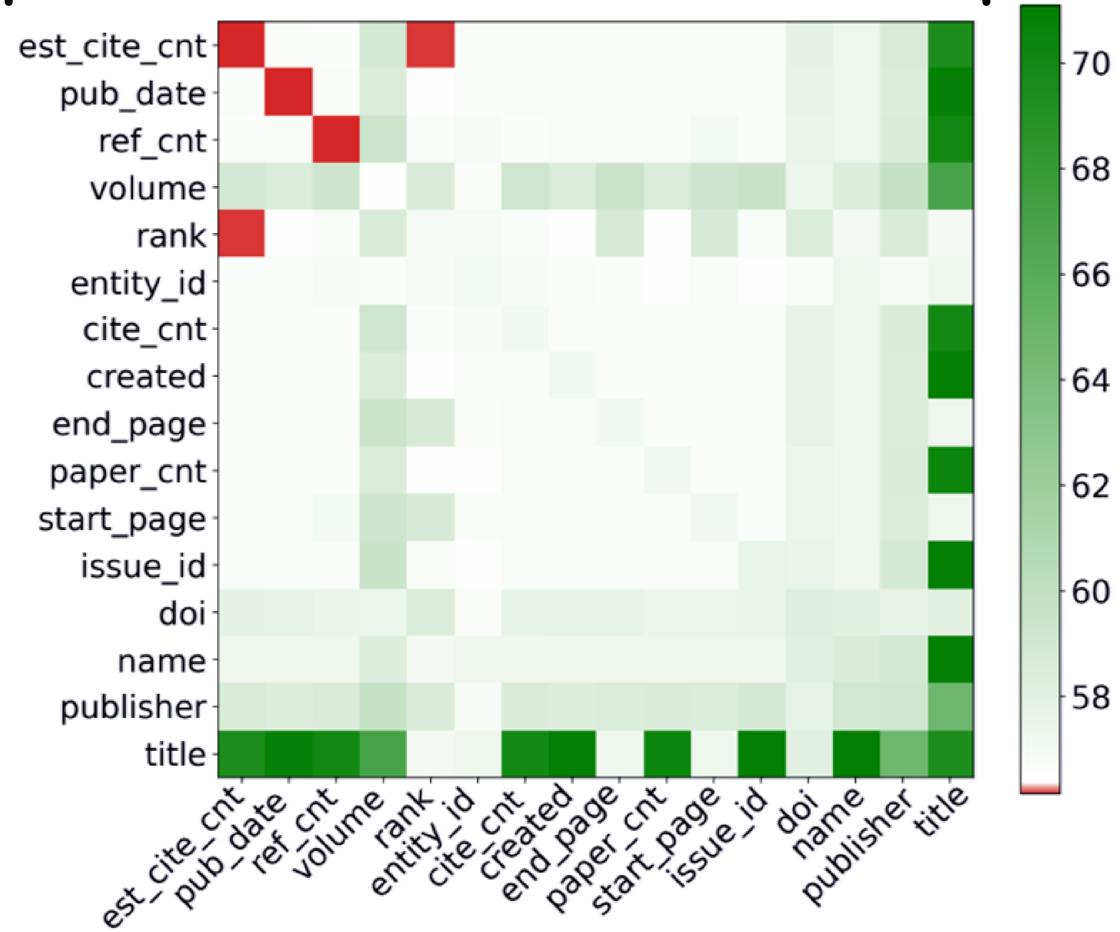
Node Classification, aka Label Prediction

Task: predict the research area of the publication

Impact from each label/property

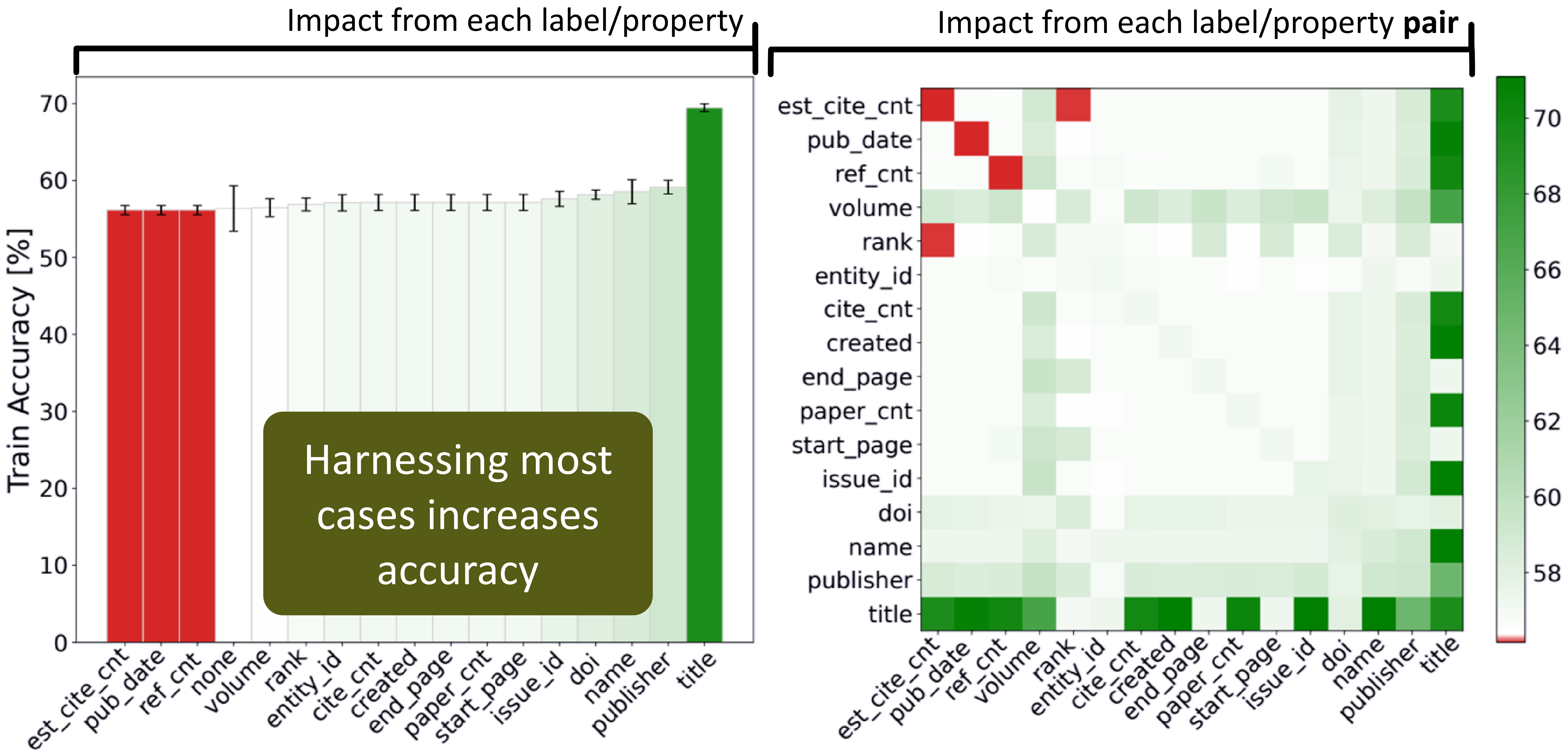


Impact from each label/property **pair**



Node Classification, aka Label Prediction

Task: predict the research area of the publication

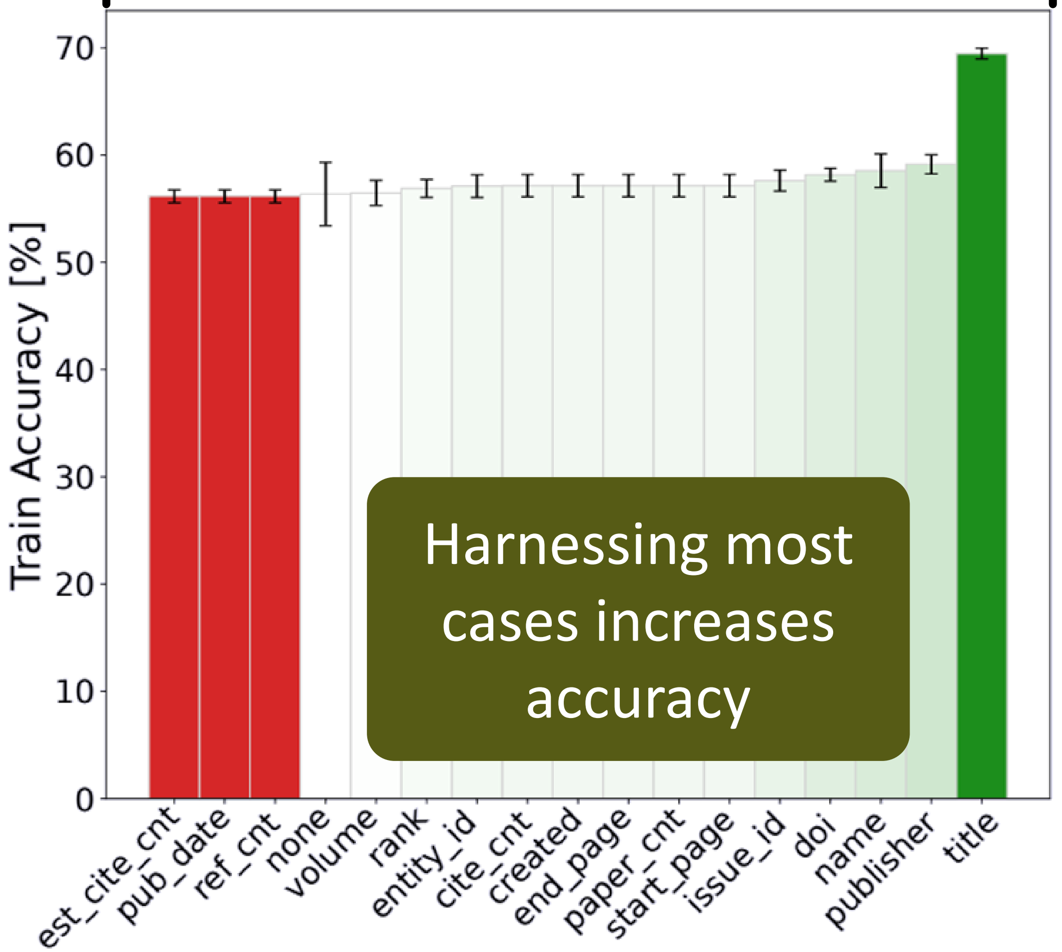


Node Classification, aka Label Prediction

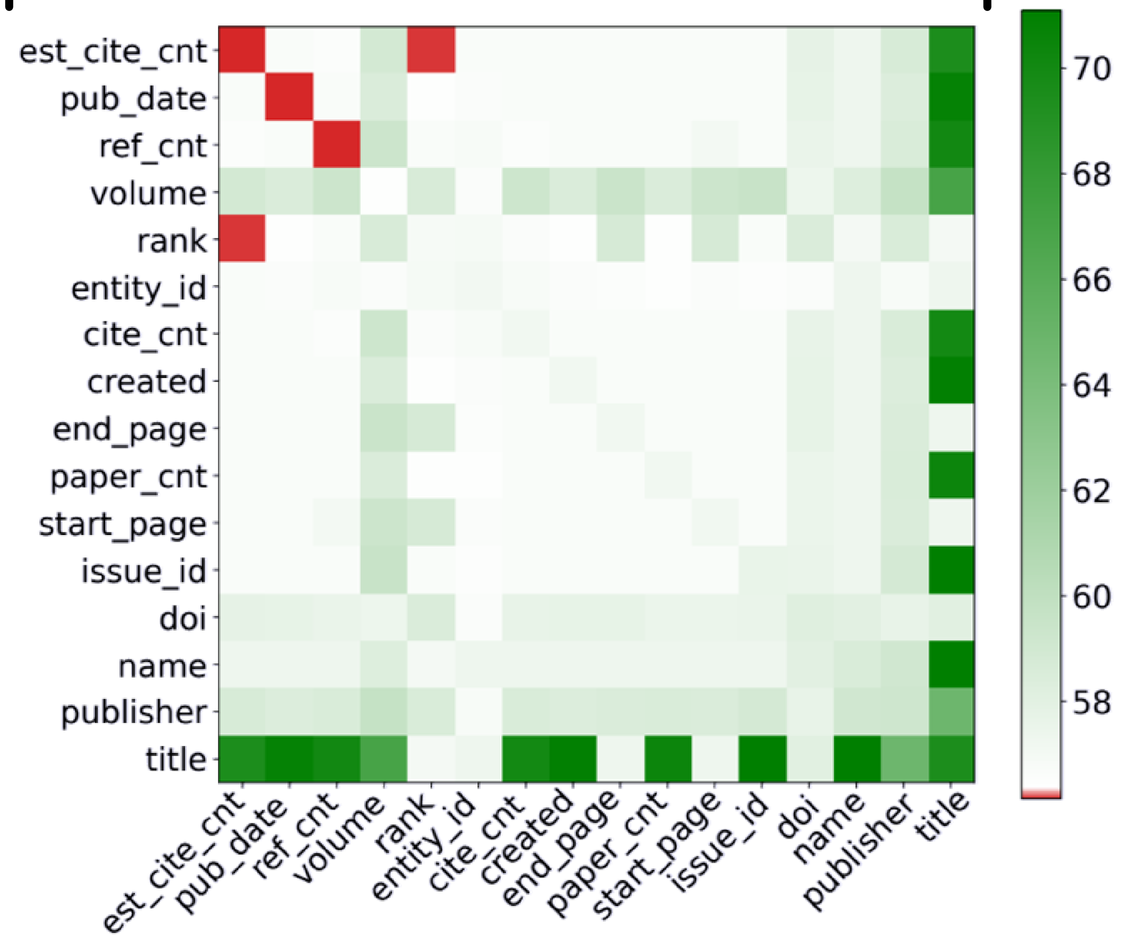
Task: predict the research area of the publication

In same cases, accuracy decreases

Impact from each label/property



Impact from each label/property **pair**



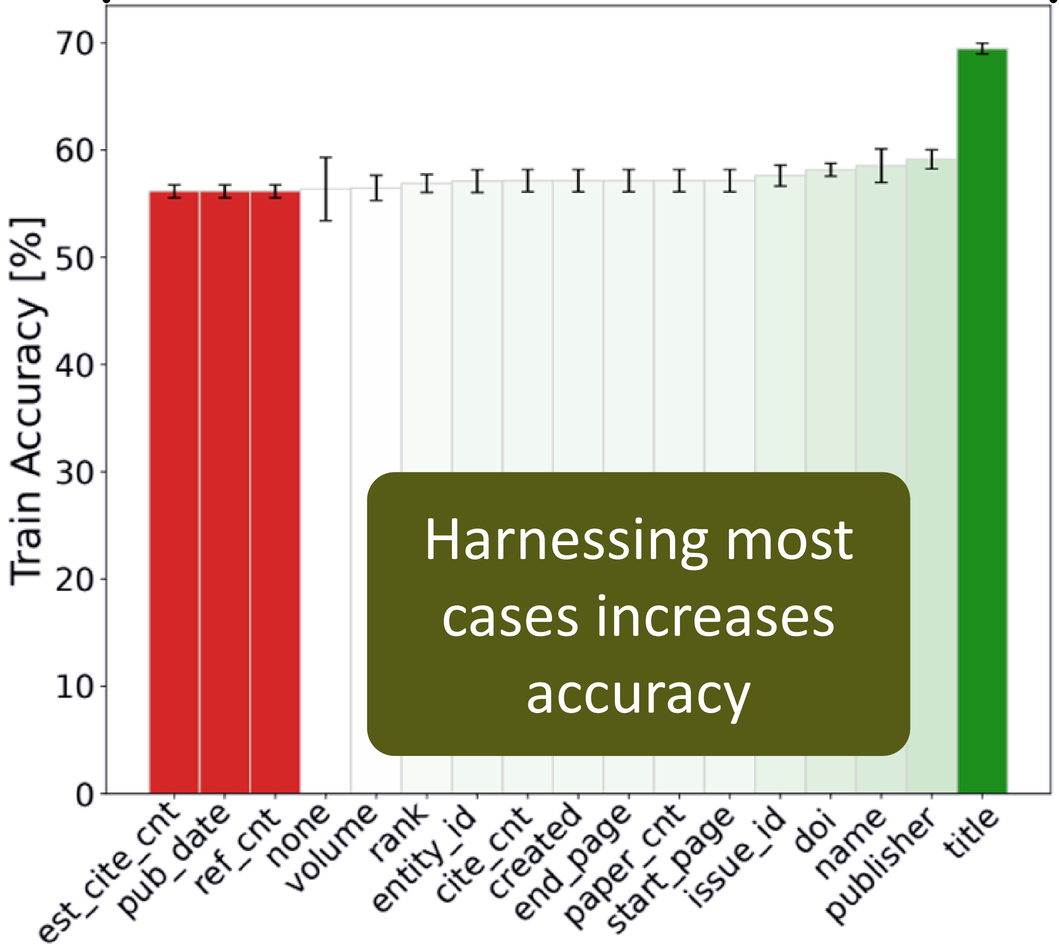
Node Classification, aka Label Prediction

Task: predict the research area of the publication

In same cases, accuracy decreases

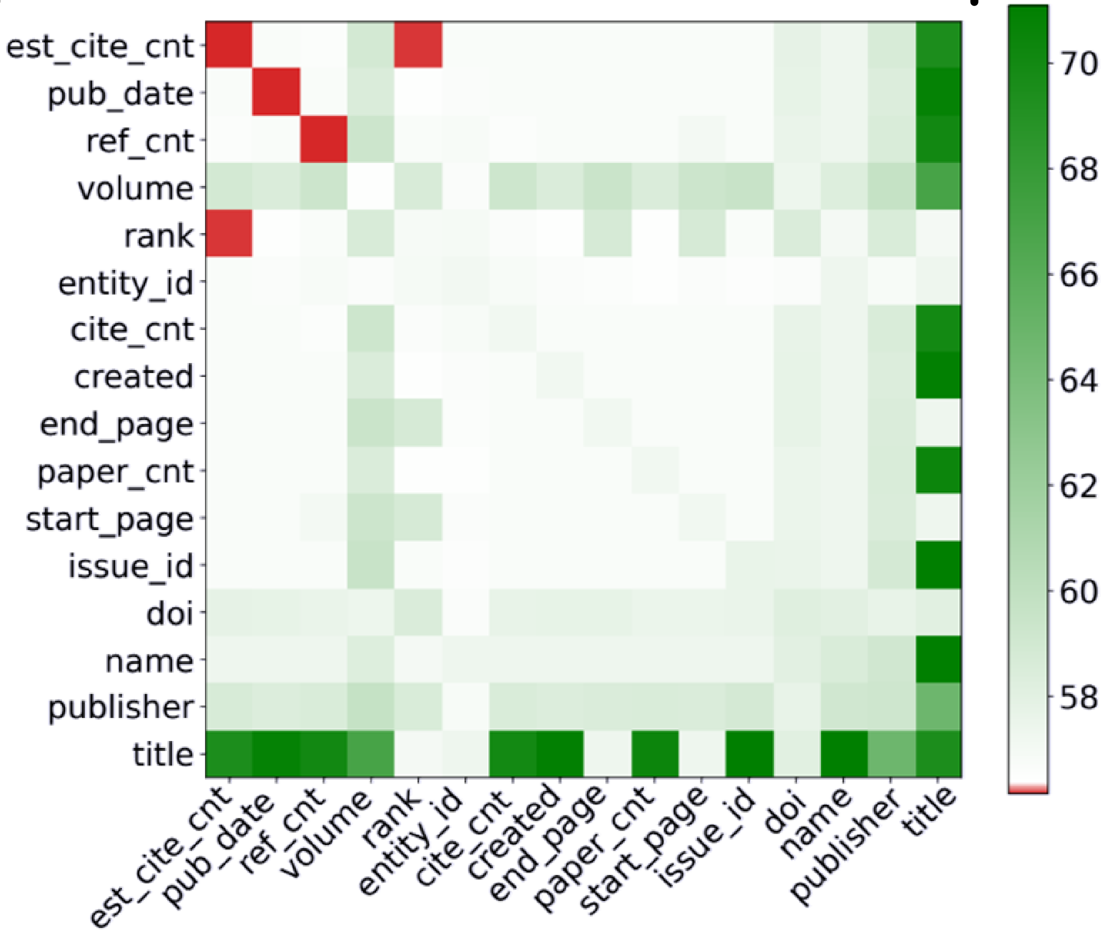
It is important to understand the data well and select the right encoded LPG information

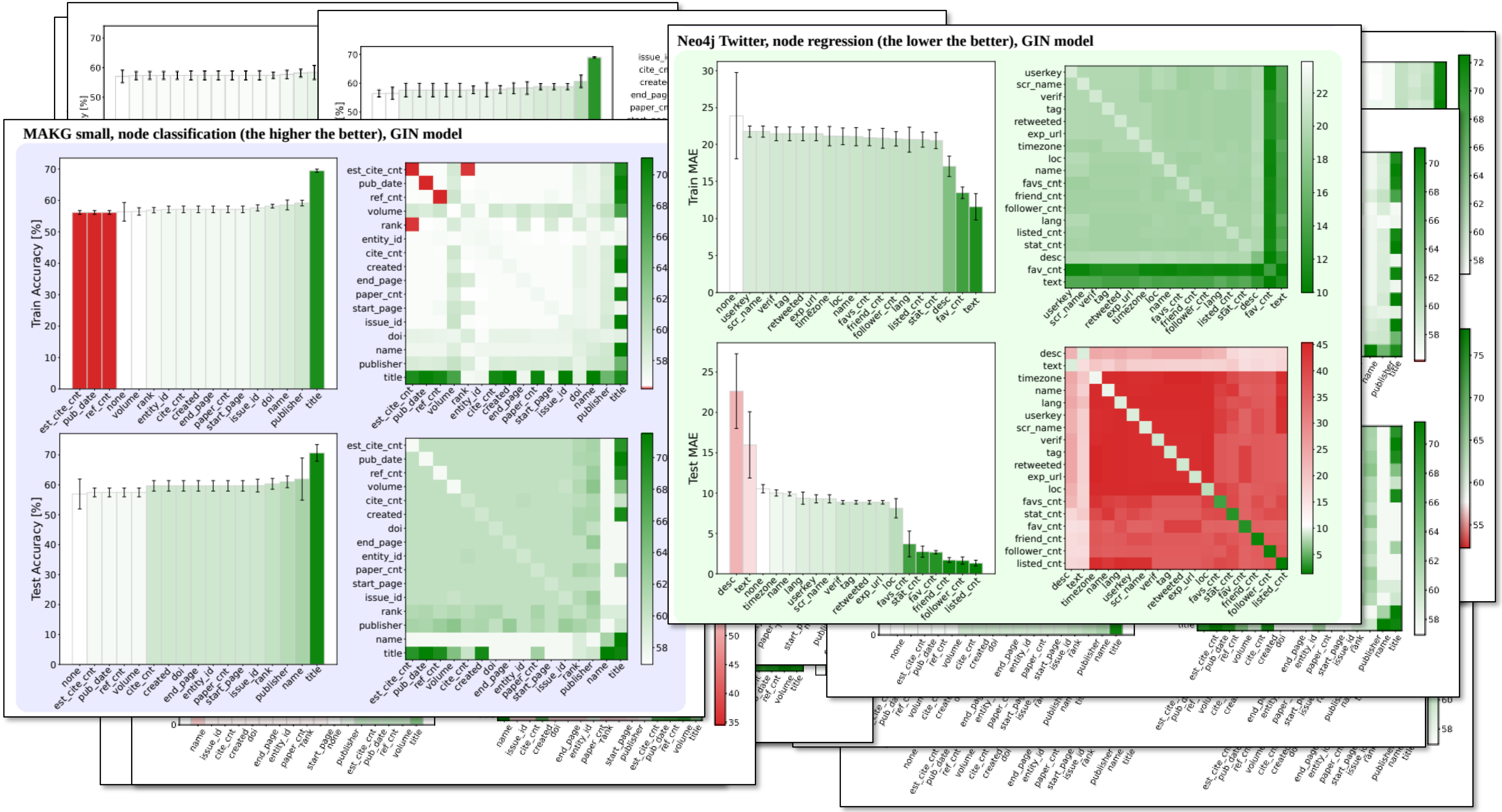
Impact from each label/property

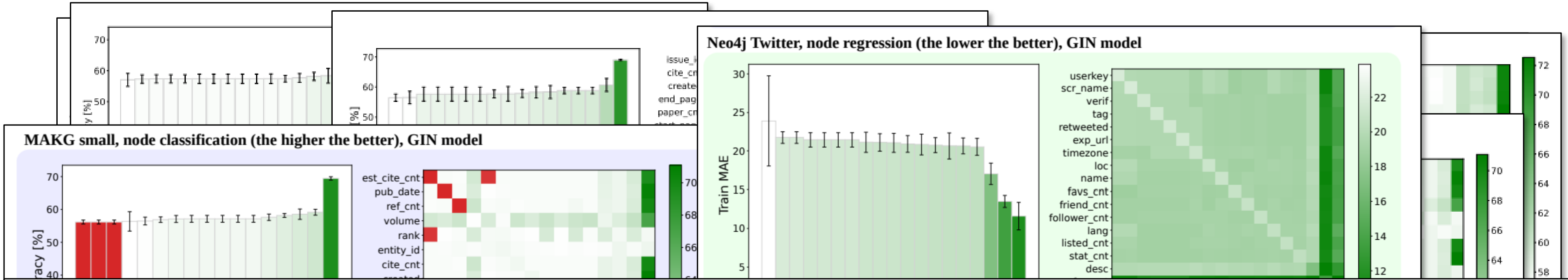


Harnessing most cases increases accuracy

Impact from each label/property **pair**




MAKG small, node classification (the higher the better), GIN model
Train Accuracy [%]
Test Accuracy [%]
Neo4j Twitter, node regression (the lower the better), GIN model
Train MAE
Test MAE



https://arxiv.org/abs/2209.09732 **@ LoG'22 (Learning on Graphs'22)**

Neural Graph Databases

Maciej Besta[1,†] Patrick Iff[1] Florian Scheidl[1] Kazuki Osawa[1] Nikoli Dryden[1]
Michal Podstawski[2,3] Tiancheng Chen[1] Torsten Hoefler[1,†]

[1]Department of Computer Science, ETH Zurich
[2]Warsaw University of Technology, Warsaw, Poland