Mining Text-Attributed Graphs with LLMs

BOWEN JIN
COMPUTER SCIENCE
UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN

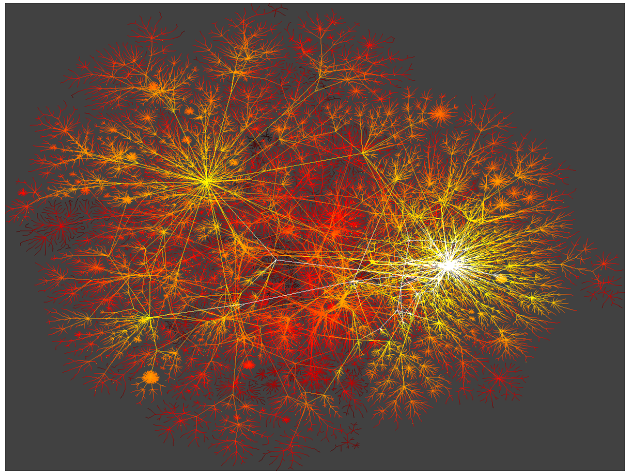
MARCH 15, 2025

Outline

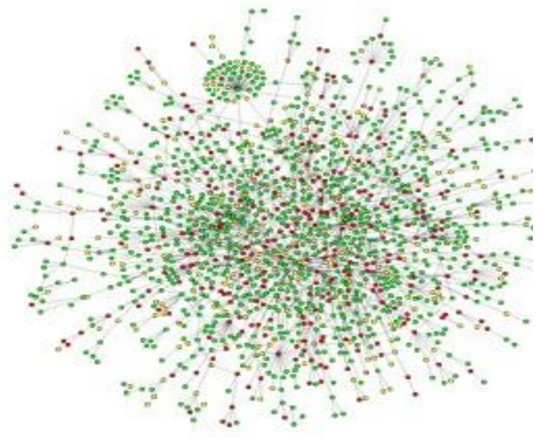
- **Motivation:** Why Mining Text-attributed Graphs?
- **Content:** Mining Text-attributed Graphs with Language Models
  - Representation learning with language models on text-attributed graphs
  - Language model pretraining text-attributed graphs
  - Large language model reasoning on text-attributed graphs

Ubiquitous Graphs

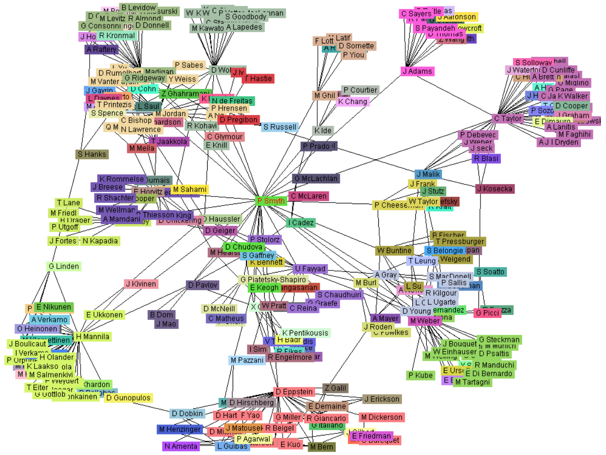
- Graphs and substructures: Chemical compounds, visual objects, circuits, XML
- Biological networks
- Bibliographic networks: DBLP, ArXiv, PubMed, …
- Social networks: Facebook >100 million active users
- World Wide Web (WWW): > 3 billion nodes, > 50 billion arcs
- Cyber-physical networks



World-Wide Web



Yeast protein interaction network

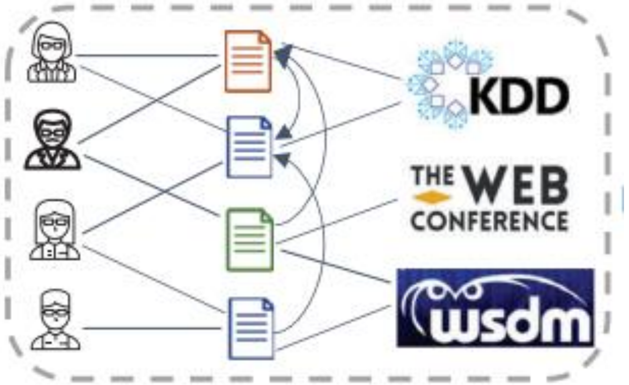


Co-author network

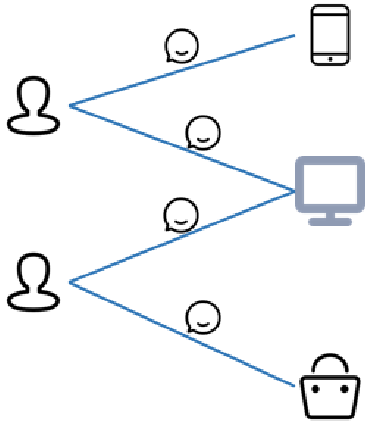
Social network sites

Text-attributed Graphs

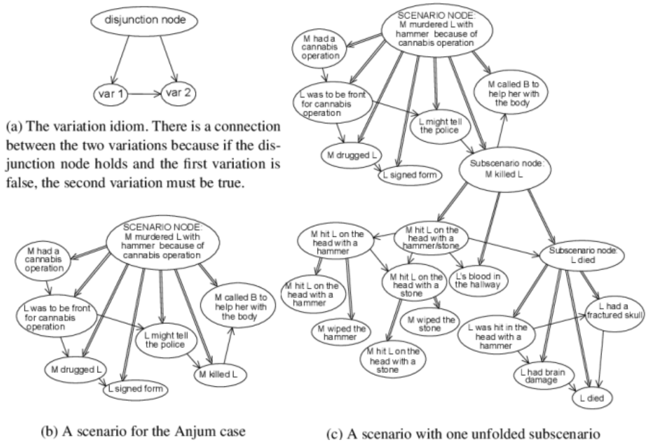
- Text-attributed Graphs
  - A graph with some nodes or edges associated with **text**.
  - Also called text-rich graphs.
  - E.g., Academic Network, User-review-Item Network, Legal-case Network



Academic network



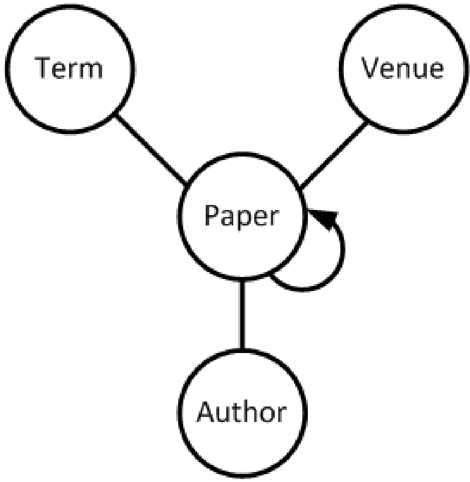
E-commerce network



Legal case network

Mining Text-attributed Graphs

- Text-attributed information networks contain rich semantic information and structure information.
  - Semantics: Ex. In an academic network, we can infer the topic of a paper from its title and abstract.
  - Structure: Ex. In an e-commerce network, two items frequently connected to the same user (co-viewed) can have similar functions.
- In a text-attributed information network, some nodes/edges can contain textual information, while others might not.
  - Ex. In an academic network, papers are associated with paper title/abstract; authors are not associated semantic-rich text.
  - Ex. In an e-commerce network, there might be text (review) between the item and user if the user leaves a review.



Examples of Text-attributed Graphs

- **Bibliographic information networks:** DBLP, ArXive, PubMed
  - Node types: *paper (P), venue* (V), *author (A), and term (T)*
  - *Edge type: authors write papers, venues publish papers, papers contain terms*
  - *Text-rich nodes: paper (title/abstract)*
- **E-commerce item network:** Amazon, Taobao
  - Node types: *item (I), brand (B), …*
  - *Edge* types: items *co-viewed with* items, items *co-purchased with* items, items *belong to* brands
  - *Text-rich nodes: item (title/description), brand (name)*
- **Legal case networks:**
  - Node types: *cases, laws, academic papers, …*
  - *Edge* types: cases *cited by* cased, cased *interpret* laws, cases *explained by* academic papers
  - Text-rich nodes: cases (content), laws (content), academic papers (content)

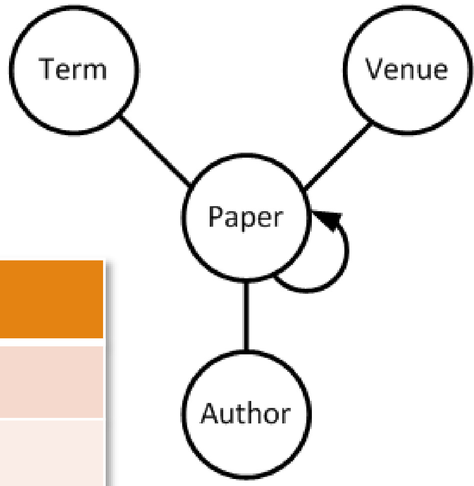
Examples of Text-attributed Graphs

- **E-commerce user-item network:** Amazon, Taobao
  - Node types: user (U), *item (I), brand (B), …*
  - *Edge* types: items *co-viewed with* items, items *co-purchased with* items, items *belong to* brands, items *purchased by* users, items *carted by* users, …
  - *Text-rich nodes: item (title/description), brand (name)*
  - *Text-rich edges:* items *reviewed by* users

- **Social networks:** twitter, Instagram
  - Node types: *users, posts, tags, …*
  - *Edge* types: posts *written by* users, posts *liked by* users, users *messaged by* users, posts *associated with* tags, …
  - Text-rich nodes: posts (content)
  - Text-rich edges: users *messaged by* users

What Can be Mined from Text-attributed Graphs?

- A raw text corpus can be derived from its “parent” text-attributed graph
  - Ex. Paper corpus from the original academic networks
- Text-attributed networks carry richer info. than the raw text corpus
- Text-attributed nodes & links imply more semantics, leading to richer discovery
- Ex.: DBLP: A Computer Science bibliographic database (network)



26 Yizhou Sun, Jiawei Han, Charu C. Aggarwal, Nitesh V. Chawla: When will it happen?: relationship prediction in heterogeneous information networks. WSDM 2012: 663-672

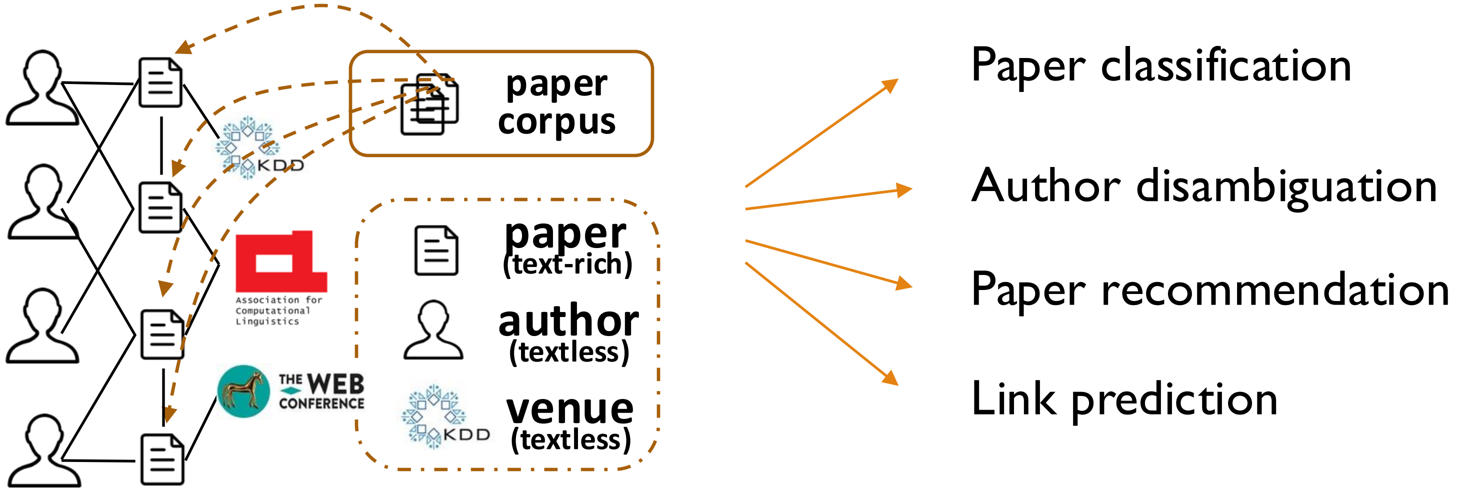
Knowledge hidden in DBLP Network	Mining Functions
Who are the **leading** researchers on Web search?	Ranking
Who are the **peer** researchers of Jure Leskovec?	Similarity Search
Whom **will** Christos Faloutsos **collaborate with**?	Relationship Prediction
Which **relationships** are most **influential** for an author to decide her topics?	Relation Strength Learning
How was the field of Data Mining **emerged** or **evolving**?	Network Evolution
Which authors are **rather different** from his/her peers in IR?	Outlier/anomaly detection

Outline

- **Motivation:** Why Mining Text-attributed Graphs?
- **Content:** Mining Text-attributed Graphs with Language Models
  - Representation learning with language models on text-attributed graphs
  - Language model pretraining text-attributed graphs
  - Large language model reasoning on text-attributed graphs

Representation learning with language models on text-attributed networks

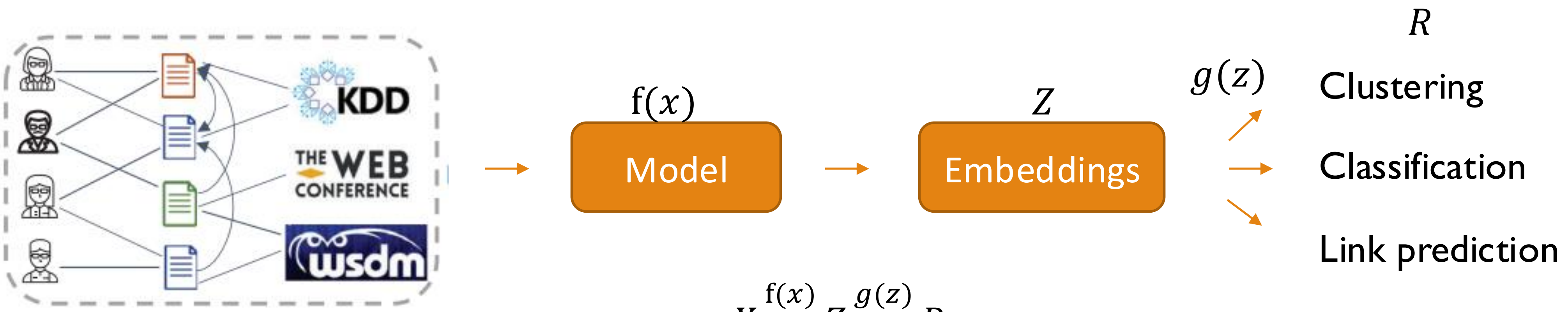
- Given a text-attributed network, people are interested in various tasks.
  - Node classification, link prediction, and node clustering.
  - E.g., academic network
    - Automatically classify each paper.
    - Find the authors of a new paper.
    - Provide paper recommendation.



Representation learning with language models on text-attributed networks

- Given a text-attributed network, people are interested in various tasks.
  - Node classification, link prediction, and node clustering.

- Learn representations for nodes/edges which can be utilized in various tasks.
  - Textual information & structure information



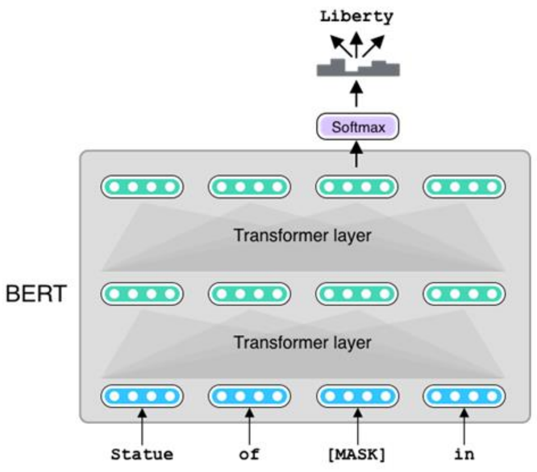
$$X \xrightarrow{f(x)} Z \xrightarrow{g(z)} R$$

Representation learning with language models on text-attributed networks

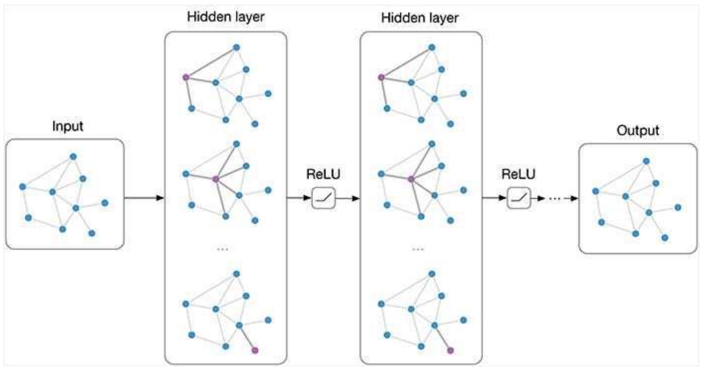
- Given a text-attributed network, people are interested in various tasks.
  - Node classification, link prediction, and node clustering.

- Learn representations for nodes/edges which can be utilized in various tasks.
  - Textual information & structure information

How to have a unified model?



Language models



Graph Neural Networks

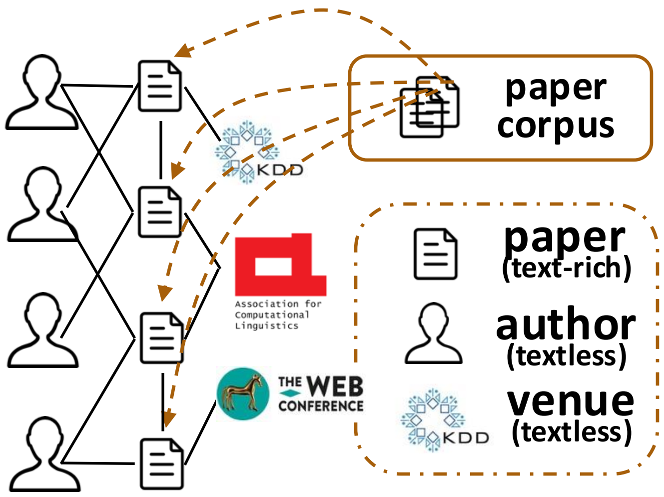
Representation learning with language models on text-attributed networks

- Heterformer [1] (KDD 2023)
  - Graph-empowered Transformer
  - Heterogeneous text-attributed networks
- Edgeformers [2] (ICLR 2023)
  - Graph-empowered Transformers
  - Textual-edge networks
- METERN [3] (NeurIPs 2024 GLFrontiers)
  - Multiplex text-attributed network

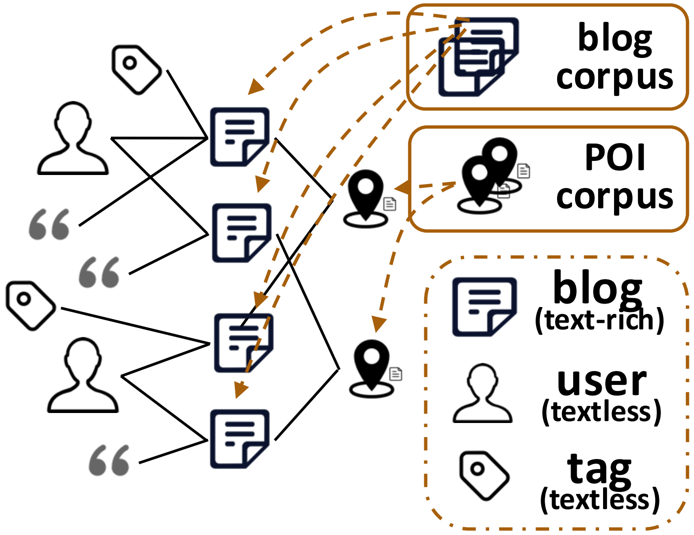
[1] Heterformer: Transformer-based Deep Node Representation Learning on Heterogeneous Text-Rich Networks. KDD 2023.
[2] Edgeformers: Graph-Empowered Transformers for Representation Learning on Textual-Edge Networks. ICLR 2023.
[3] Learning Multiplex Representations on Text-Attributed Graphs with One Language Model Encoder. Arxiv 2023.

Heterformer: Transformer-based Deep Node Representation Learning on Heterogeneous Text-Rich Networks (KDD23)

- Heterogeneous text-rich networks are ubiquitously utilized to model real-world data
  - Text-rich.
  - Heterogeneous.
  - E.g., Academic Networks, Social Media Networks



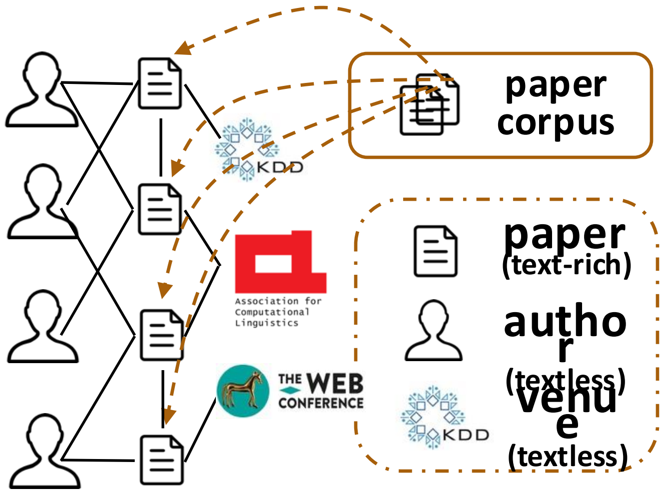
Academic Network



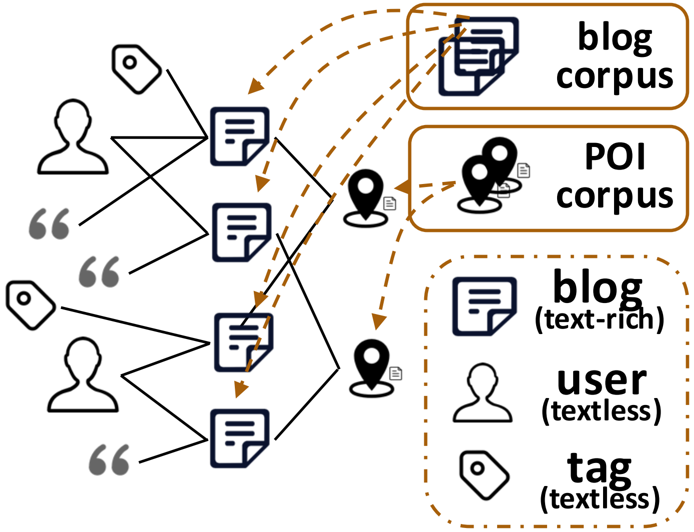
Social Media Network

Heterformer: Transformer-based Deep Node Representation Learning on Heterogeneous Text-Rich Networks (KDD23)

- Heterogeneity in those text-rich networks
  - Presence or absence of text.
  - Diversity of types.



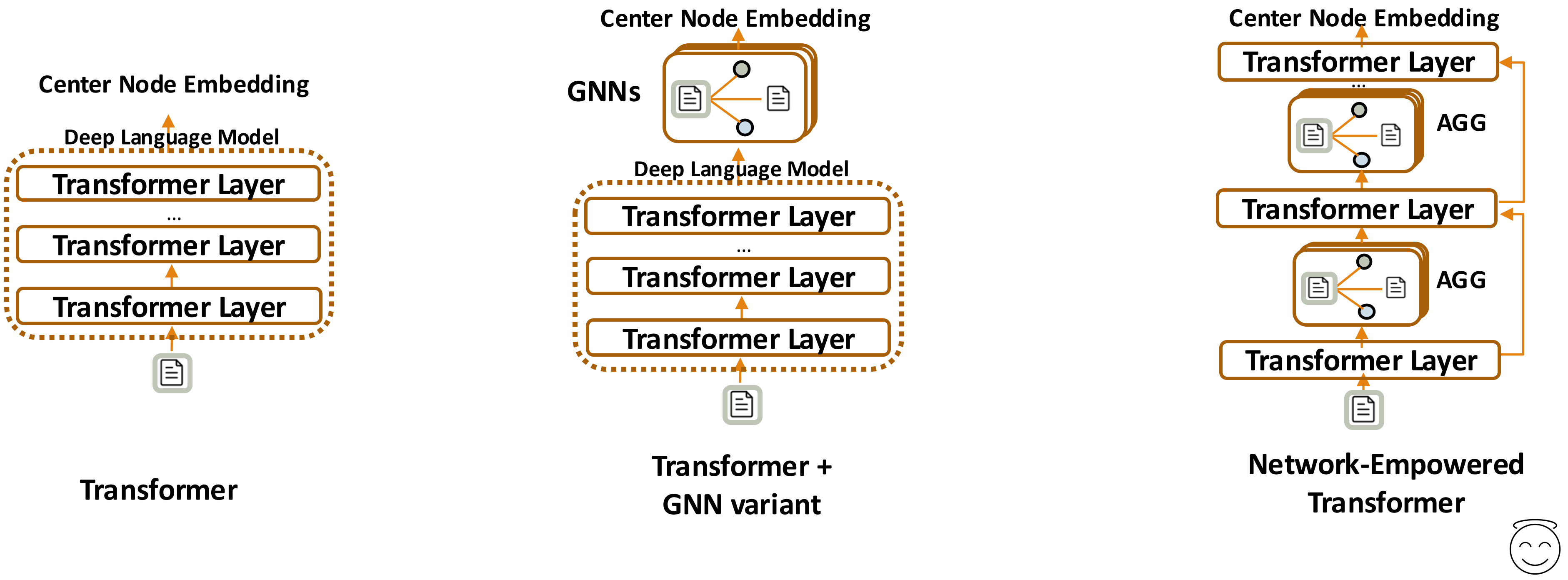
Academic Network



Social Media Network

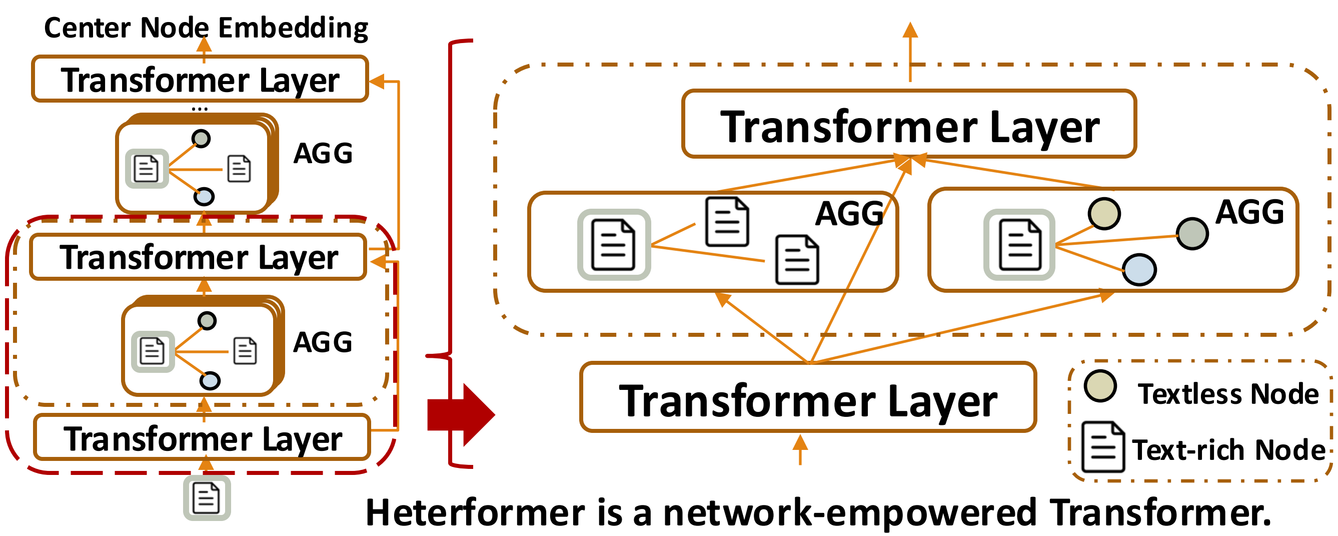
Heterformer: Transformer-based Deep Node Representation Learning on Heterogeneous Text-Rich Networks (KDD23)

- Overall framework
  - Transformers + GNN vs. Network-Empowered Transformer



Heterformer: Transformer-based Deep Node Representation Learning on Heterogeneous Text-Rich Networks (KDD23)

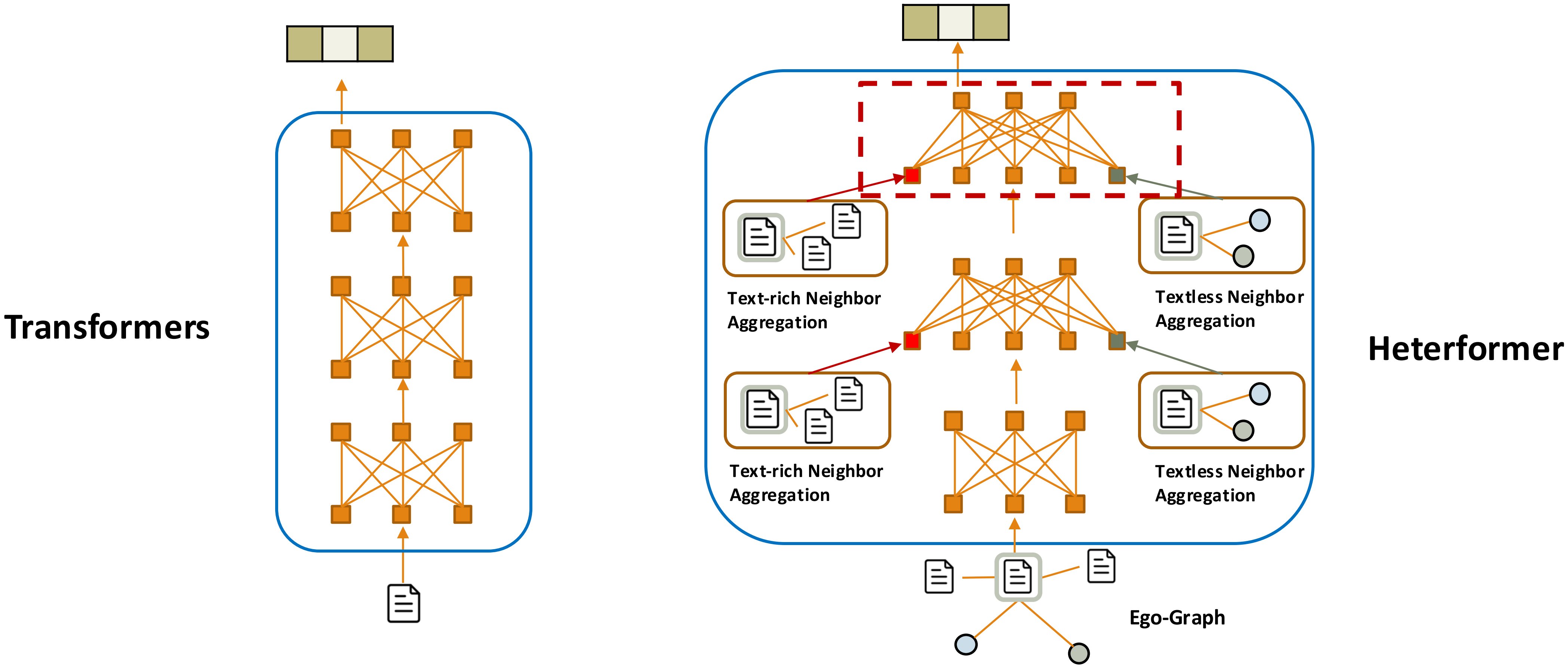
- Overall framework
  - Heterformer: a network-empowered Transformer.
  - Unifying text semantic encoding and network signal capturing.



Heterformer is a network-empowered Transformer.

Heterformer: Transformer-based Deep Node Representation Learning on Heterogeneous Text-Rich Networks (KDD23)

- Text-Rich Node Encoding
  - Network-aware node text encoding with virtual neighbor tokens.
  - Multi-head attention-based heterogeneous neighbor aggregation.



Heterformer: Transformer-based Deep Node Representation Learning on Heterogeneous Text-Rich Networks (KDD23)

- Textless Node Encoding
  - Node type heterogeneity-based representation

$$\boldsymbol{h}_{v_p}^{(l)} = \mathbf{W}_{\phi_i}^{(l)} \boldsymbol{h}_{v_p}^{(0)}, \quad \text{where } \phi(v_p) = \phi_i, \;\; \phi_i \in \mathcal{A}_{\mathrm{TL}}.$$

Node type heterogeneity

  - Textless node embedding warm up
    - A great number of textless nodes will introduce a great number of randomly initialized parameters into the model -> underfitting.
    - Warm up to give textless node embeddings good initializations.

$$\min_{\boldsymbol{h}_{v_p}^{(l)}} \mathcal{L}_w = \sum_{\substack{v_p \in \mathcal{V} \\ \phi(v_p) \in \mathcal{A}_{\mathrm{TL}}}} \sum_{v_u \in \widehat{N}_{v_p}} -\log \frac{\exp(\bar{\boldsymbol{h}}_{v_u}^{\top} \boldsymbol{h}_{v_p}^{(l)})}{\exp(\bar{\boldsymbol{h}}_{v_u}^{\top} \boldsymbol{h}_{v_p}^{(l)}) + \sum_{v_u'} \exp(\bar{\boldsymbol{h}}_{v_u'}^{\top} \boldsymbol{h}_{v_p}^{(l)})},$$

Heterformer: Transformer-based Deep Node Representation Learning on Heterogeneous Text-Rich Networks (KDD23)

- Model Training
  - Unsupervised training objective

$$\max_{\Theta} O = \prod_{\substack{v_i \in \mathcal{V} \\ \phi(v_i) \in \mathcal{A}_{\text{TR}}}} \prod_{\substack{v_j \in N_{v_i} \\ \phi(v_j) \in \mathcal{A}_{\text{TR}}}} p(v_j | v_i; \Theta),$$

$$p(v_j | v_i; \Theta) = \frac{\exp(\boldsymbol{h}_{v_j}^\top \boldsymbol{h}_{v_i})}{\sum_{v_u \in \mathcal{V}, \phi(v_u) \in \mathcal{A}_{\text{TR}}} \exp(\boldsymbol{h}_{v_u}^\top \boldsymbol{h}_{v_i})},$$

  - Negative sampling

$$\min_{\Theta} \mathcal{L} = \sum_{\substack{v_i \in \mathcal{V} \\ \phi(v_i) \in \mathcal{A}_{\text{TR}}}} \sum_{\substack{v_j \in N_{v_i} \\ \phi(v_j) \in \mathcal{A}_{\text{TR}}}} -\log \frac{\exp(\boldsymbol{h}_{v_j}^\top \boldsymbol{h}_{v_i})}{\exp(\boldsymbol{h}_{v_j}^\top \boldsymbol{h}_{v_i}) + \sum_{v'_u} \exp(\boldsymbol{h}_{v'_u}^\top \boldsymbol{h}_{v_i})}.$$

Heterformer: Transformer-based Deep Node Representation Learning on Heterogeneous Text-Rich Networks (KDD23)

- Datasets:
  - DBLP
    - CS papers from 1990 to 2020.
  - Twitter
    - POI-related tweets in LA and NY.
  - Goodreads
    - Books listed in Goodreads

Dataset	Node	Edge
DBLP	# paper*: 3,597,191 # venue: 28,638 # author: 2,717,797	# paper-paper: 36,787,329 # venue-paper: 3,633,613 # author-paper: 10,212,497
Twitter	# tweet*: 279,694 # POI*: 36,895 # hashtag: 72,297 # user: 76,398 # mention: 24,089	# tweet-POI: 279,694 # user-tweet: 195,785 # hashtag-tweet: 194,939 # mention-tweet: 50,901
Goodreads	# book*:1,097,438 # shelves: 6,632 # author: 205,891 # format: 768 # publisher: 62,934 # language code: 139	# book-book: 11,745,415 # shelves-book: 27,599,160 # author-book: 1,089,145 # format-book: 588,677 # publisher-book: 591,456 # language code-book: 485,733

Heterformer: Transformer-based Deep Node Representation Learning on Heterogeneous Text-Rich Networks (KDD23)

- Link prediction

	Method	DBLP PREC	DBLP MRR	DBLP NDCG	Twitter PREC	Twitter MRR	Twitter NDCG	Goodreads PREC	Goodreads MRR	Goodreads NDCG
	MeanSAGE	0.7019	0.7964	0.8437	0.6489	0.7450	0.7991	0.6302	0.7409	0.8001
	BERT	0.7569	0.8340	0.8726	0.7179	0.7833	0.8265	0.5571	0.6668	0.7395
Homo GNN	BERT+MeanSAGE	0.8131	0.8779	0.9070	0.7201	0.7845	0.8275	0.7301	0.8167	0.8594
	BERT+MAXSAGE	0.8193	0.8825	0.9105	0.7198	0.7845	0.8276	0.7280	0.8164	0.8593
	BERT+GAT	0.8119	0.8771	0.9063	0.7231	0.7873	0.8300	0.7333	0.8170	0.8593
	GraphFormers	0.8324	0.8916	0.9175	0.7258	0.7891	0.8312	0.7444	0.8260	0.8665
Hetero GNN	BERT+RGCN	0.7979	0.8633	0.8945	0.7111	0.7764	0.8209	0.7488	0.8303	0.8699
	BERT+HAN	0.8136	0.8782	0.9072	0.7237	0.7880	0.8306	0.7329	0.8174	0.8597
	BERT+HGT	0.8170	0.8814	0.9098	0.7153	0.7800	0.8237	0.7224	0.8112	0.8552
	BERT+SHGN	0.8149	0.8785	0.9074	0.7218	0.7866	0.8295	0.7362	0.8195	0.8613
	GraphFormers++	0.8233	0.8856	0.9130	0.7159	0.7799	0.8236	0.7536	0.8328	0.8717
	Heterformer	**0.8474***	**0.9019***	**0.9255***	**0.7272***	**0.7908***	**0.8328***	**0.7633***	**0.8400***	**0.8773***

Heterformer: Transformer-based Deep Node Representation Learning on Heterogeneous Text-Rich Networks (KDD23)

- Node Classification

Table 3: Transductive text-rich node classification.

Method	DBLP		Goodreads	
	Micro-F1	Macro-F1	Micro-F1	Macro-F1
BERT	0.6119	0.5476	0.8364	0.7713
BERT+MaxSAGE	0.6179	0.5511	0.8447	0.7866
BERT+MeanSAGE	0.6198	0.5522	0.8420	0.7826
BERT+GAT	0.5943	0.5175	0.8328	0.7713
GraphFormers	0.6256	0.5616	0.8388	0.7786
BERT+HAN	0.5965	0.5211	0.8351	0.7747
BERT+HGT	0.6575	0.5951	0.8474	0.7928
BERT+SHGN	0.5982	0.5214	0.8345	0.7737
GraphFormers++	0.6474	0.5790	0.8516	0.7993
Heterformer	**0.6695***	**0.6062***	**0.8578***	**0.8076***

Table 4: Inductive text-rich node classification.

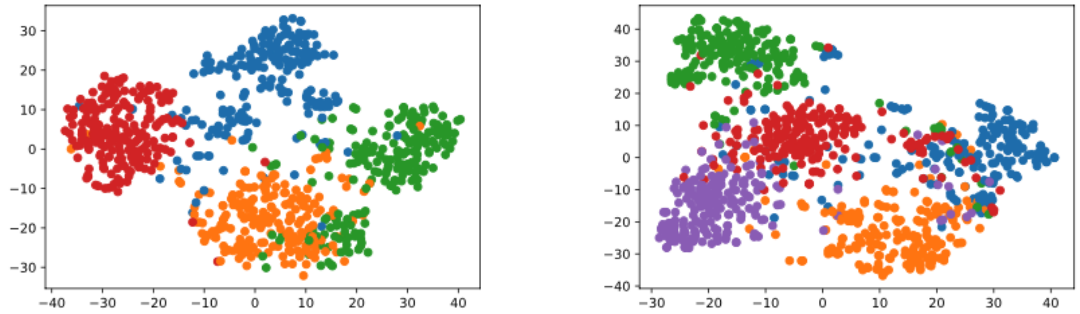
Method	DBLP		Goodreads	
	Micro-F1	Macro-F1	Micro-F1	Macro-F1
BERT	0.5996	0.5318	0.8122	0.7371
BERT+MaxSAGE	0.6117	0.5435	0.8368	0.7749
BERT+MeanSAGE	0.6129	0.5431	0.8350	0.7721
BERT+GAT	0.5879	0.5150	0.8249	0.7590
GraphFormers	0.6197	0.5548	0.8330	0.7683
BERT+HAN	0.5948	0.5165	0.8279	0.7626
BERT+HGT	0.6467	0.5835	0.8390	0.7798
BERT+SHGN	0.5955	0.5202	0.8280	0.7626
GraphFormers++	0.6386	0.5696	0.8427	0.7848
Heterformer	**0.6600***	**0.5976***	**0.8507***	**0.7977***

Heterformer: Transformer-based Deep Node Representation Learning on Heterogeneous Text-Rich Networks (KDD23)

- Node Clustering

Table 6: Node clustering.

Method	DBLP		Goodreads	
	NMI	**ARI**	**NMI**	**ARI**
BERT	0.2570	0.3349	0.2325	0.4013
BERT+MaxSAGE	0.2615	0.3490	0.2205	0.4173
BERT+MeanSAGE	0.2628	0.3488	**0.2449**	**0.4329**
BERT+GAT	0.2598	0.3419	0.2408	0.4185
GraphFormers	0.2633	0.3455	0.2362	0.4139
BERT+HAN	0.2568	0.3401	0.2391	0.4266
BERT+HGT	0.2469	0.3392	0.2427	0.4296
BERT+SHGN	0.2589	0.3431	0.2373	0.4171
GraphFormers++	0.2566	0.3432	0.2372	0.4211
Heterformer	**0.2707***	**0.3639***	0.2429	0.4199



(a) DBLP (b) Goodreads

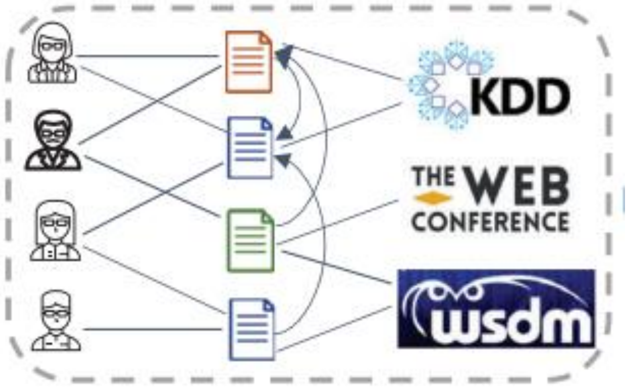
Figure 3: Embedding visualization.

Outline

- **Motivation:** Why Mining Text-attributed Graphs?
- **Content:** Mining Text-attributed Graphs with Language Models
  - Representation learning with language models on text-attributed graphs
  - Language model pretraining text-attributed graphs
  - Large language model reasoning on text-attributed graphs

Why do we need language model pretraining on network?

- Given a text-rich network, people are interested in various downstream tasks
  - Document/node classification, document retrieval and link prediction
  - E.g., academic network
    - Automatically classify each paper
    - Find the authors of a new paper
    - Provide paper recommendation



Academic network

Paper classification

Author disambiguation

Paper recommendation

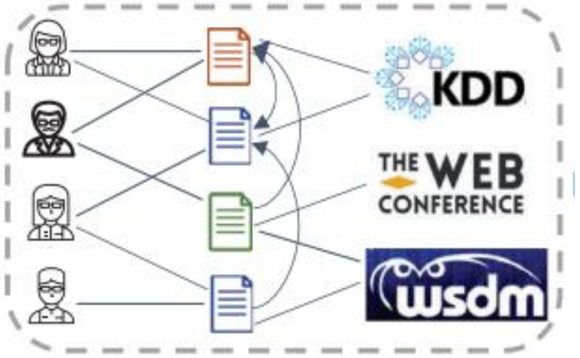
Link prediction

Why do we need language model pretraining on network?

- Given a text-rich network, people are interested in various downstream tasks
  - Document/node classification, document retrieval and link prediction

- Text-rich network contains rich unsupervised semantic information
  - Alleviate human labeling burden for downstream tasks

PLM →



→ G-Adapted PLM

Retrieval

Reranking

Classification

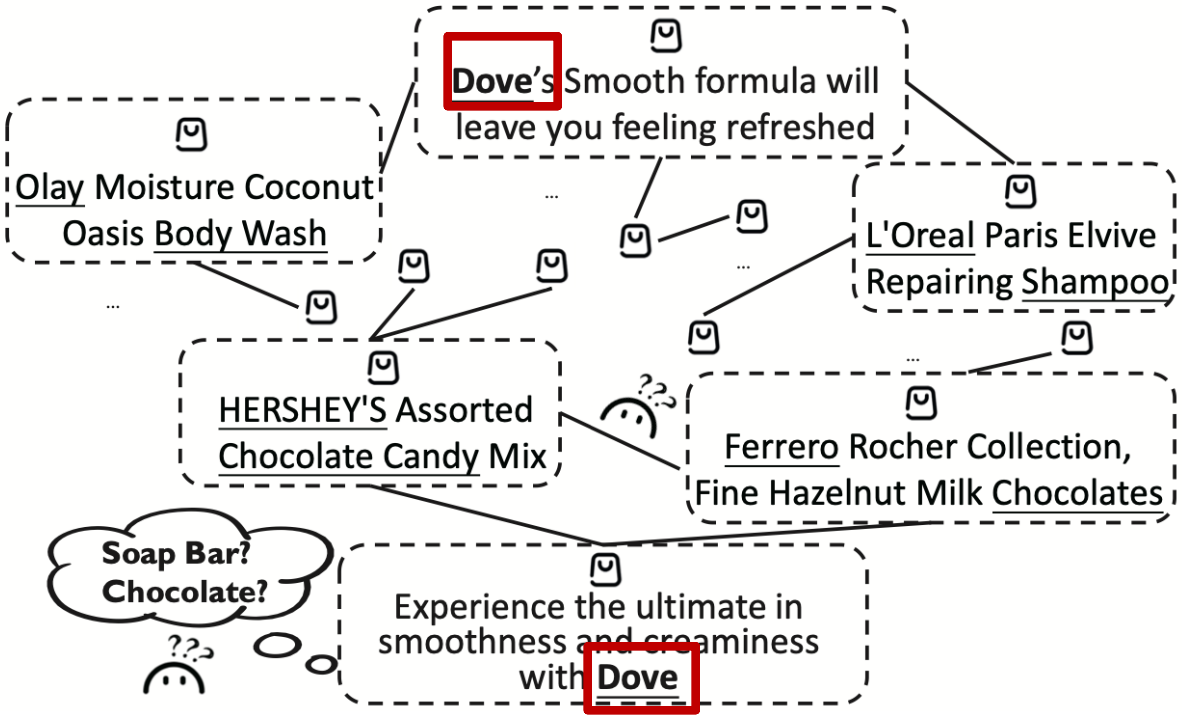
Link prediction

Pretraining on a Text-rich Network G

Finetuning on downstream tasks

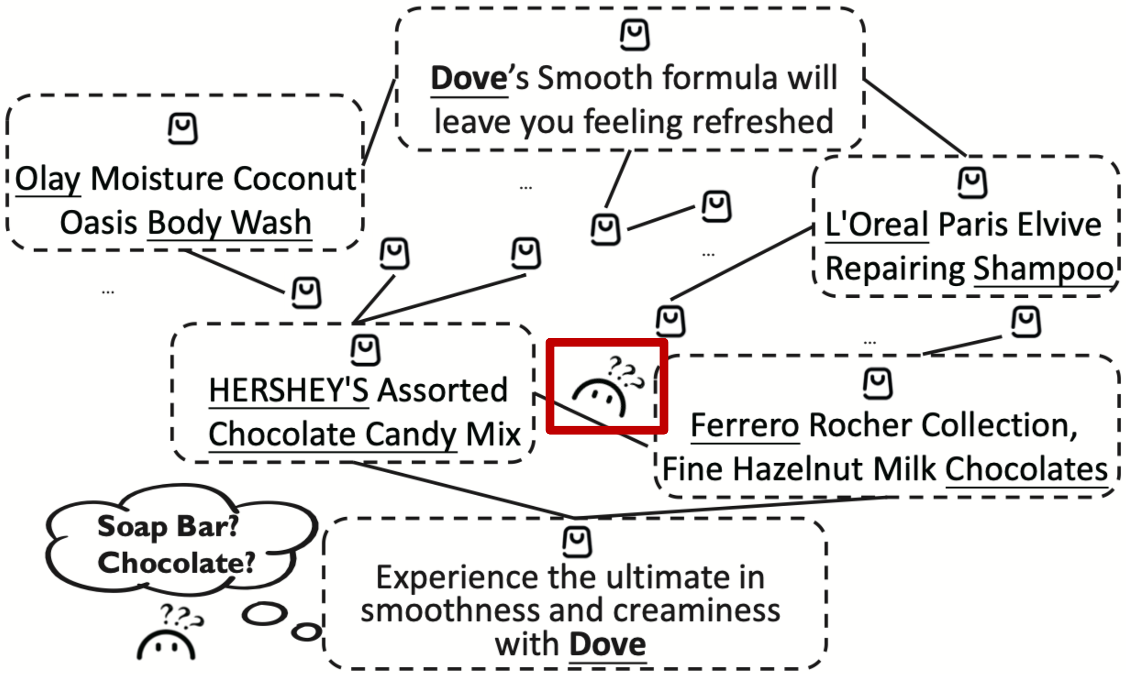
Patton: Language Model Pretraining on Text-Rich Networks (ACL 2023)

- How to design pretraining strategies to help LMs extract unsupervised semantic information from the network?
  - Motivation 1: On token-level, documents can help facilitate the understanding of tokens.



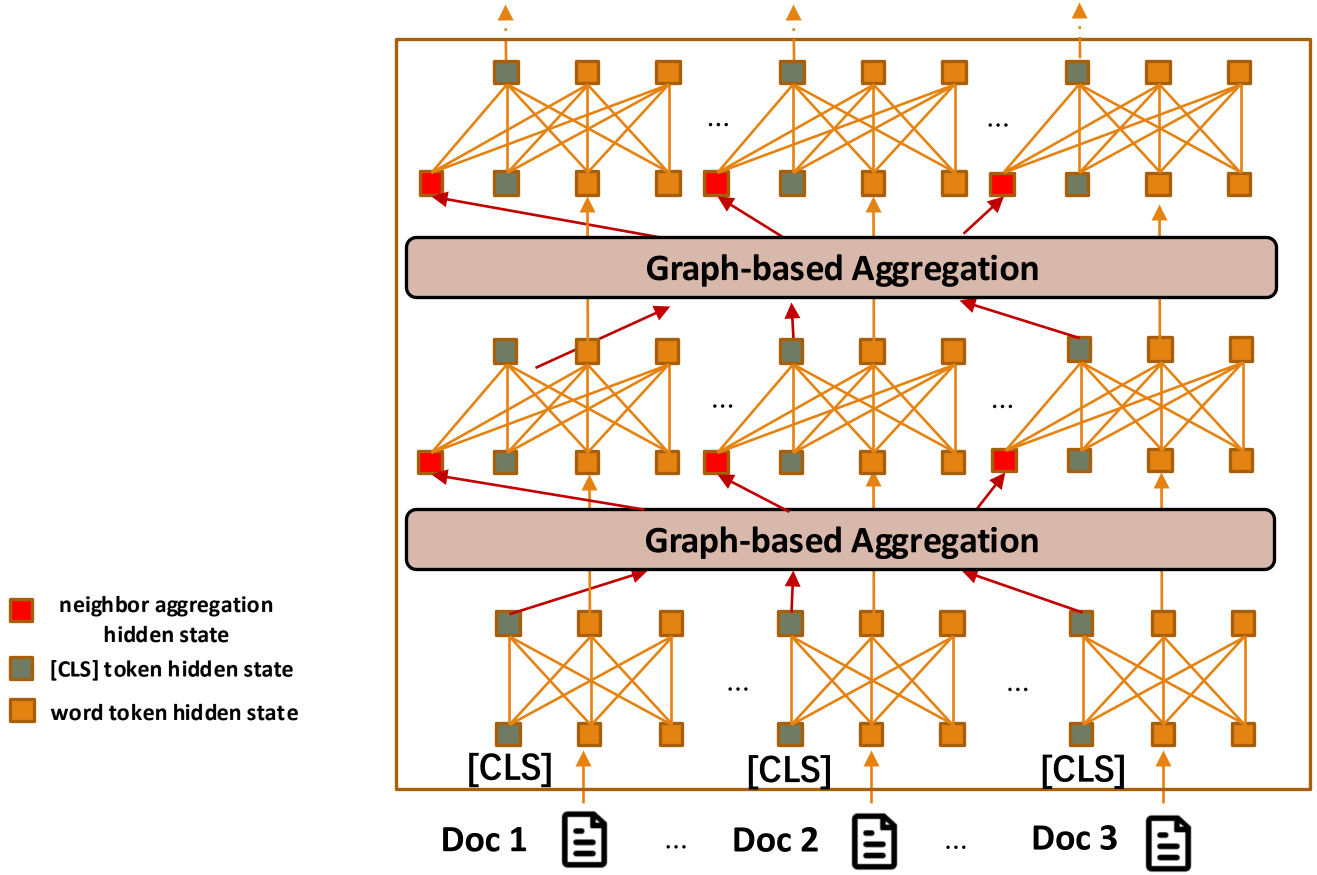
Patton: Language Model Pretraining on Text-Rich Networks (ACL 2023)

- How to design pretraining strategies to help LMs extract unsupervised semantic information from the network?
  - Motivation 2: On document-level, the two connected nodes can have quite related overall textual semantics.



Patton: Language Model Pretraining on Text-Rich Networks (ACL 2023)

- Mode architecture
  - GraphFormers: Graph-empowered Transformer architecture



$$z_x^{(l)} = \mathrm{GNN}(\{H_y^{(l)}\texttt{[CLS]} | y \in N_x\}), \quad (1)$$

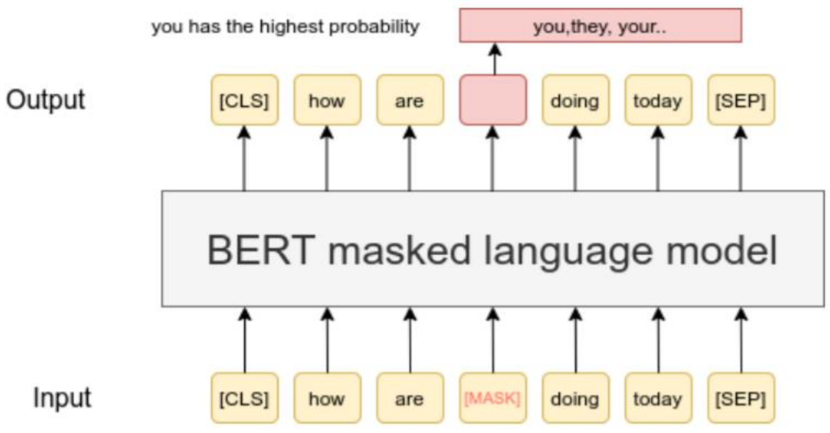
$$\widetilde{H}_x^{(l)} = \mathrm{Concate}(z_x^{(l)}, H_x^{(l)}), \quad (2)$$

$$\widetilde{H}_x^{(l)\prime} = \mathrm{LN}(H_x^{(l)} + \mathrm{MHA}_{asy}(\widetilde{H}_x^{(l)})), \quad (3)$$

$$H_x^{(l+1)} = \mathrm{LN}(\widetilde{H}_x^{(l)\prime} + \mathrm{MLP}(\widetilde{H}_x^{(l)\prime})), \quad (4)$$

Patton: Language Model Pretraining on Text-Rich Networks (ACL 2023)

- Pretraining strategy 1: Network-contextualized masked language modeling
  - Original masked language modeling
    - BERT, domain adaptation
    - The semantics of each token can be reflected by its contexts.



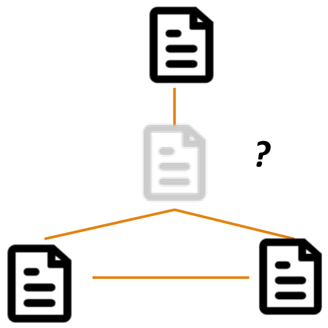
$$\mathcal{L}_{\text{MLM}} = -\sum_{i \in M_t} \log p(w_i | \boldsymbol{H}_i),$$

  - Ours
    - In node MLM -> Network contextualized MLM
    - Use both in-node text context and neighbor node context to conduct masked token prediction
    - Facilitate the LM to understand both in-node token correlation and network-contextualized text semantic relatedness

$$\mathcal{L}_{\text{NMLM}} = -\sum_{i \in M_t} \log p(w_i | \boldsymbol{H}_x, \boldsymbol{z}_x),$$

Patton: Language Model Pretraining on Text-Rich Networks (ACL 2023)

- Pretraining strategy 2: Masked Node Prediction
  - We dynamically hold out a subset of nodes from the network ($M_v \subseteq V$), mask them, and train the LM to predict the masked nodes based on the adjacent network structure.
  - LM will absorb document semantic hints hidden inside the network structure.



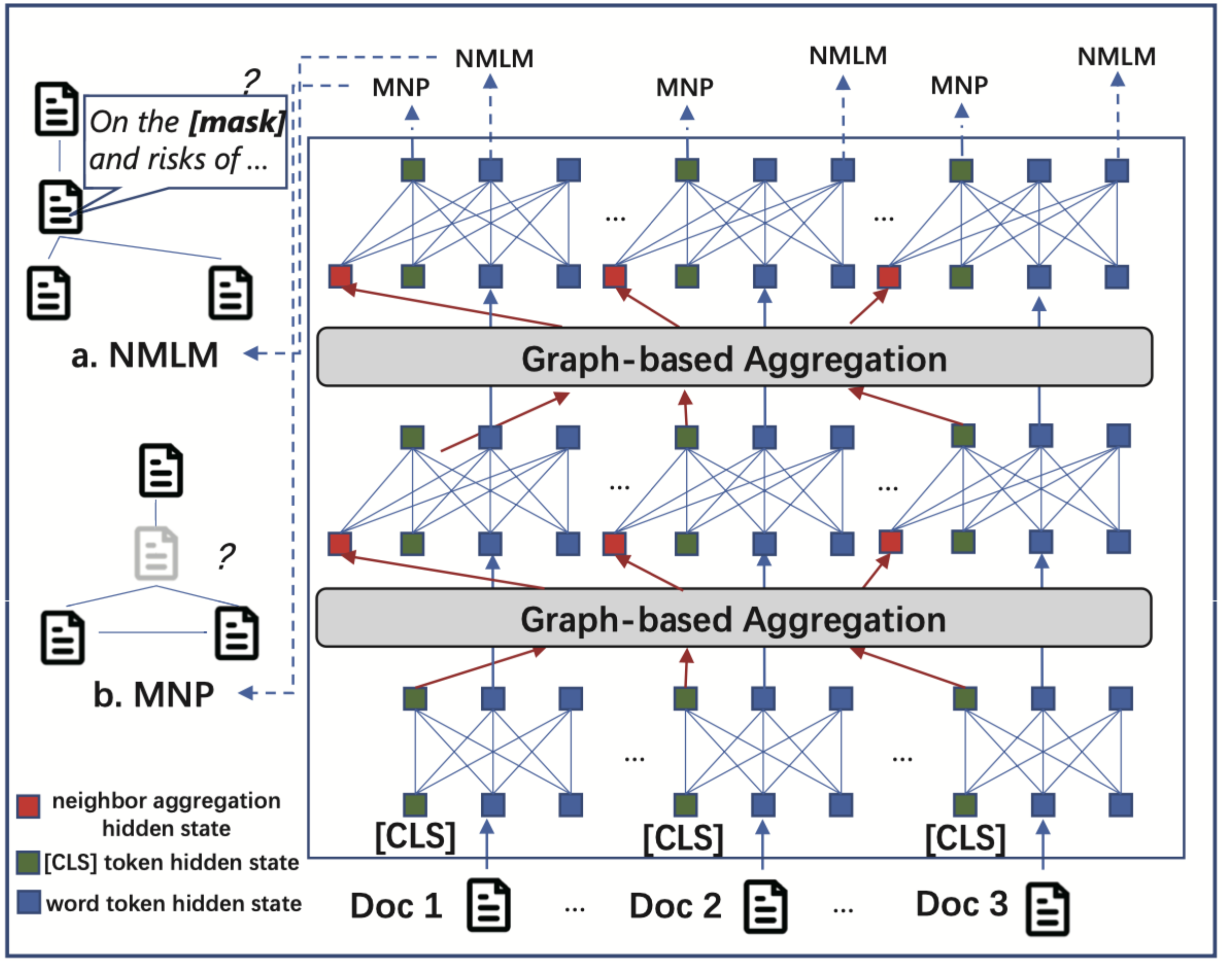
$$\mathcal{L}_{\text{MNP}} = -\sum_{v_j \in M_v} \log p(v_j | \boldsymbol{G}_{v_j})$$

- Directly optimizing masked node prediction is computationally expensive
  - Representations for all candidates/neighboring nodes
- We prove that masked node prediction can be theoretically transferred to a computationally cheaper pairwise link prediction task.

$$\begin{aligned}
& \prod_{v_{\texttt{[MASK]}} \in M_v} p(v_{\texttt{[MASK]}} = v_i | v_k \in N_{v_{\texttt{[MASK]}}}) \\
\propto & \prod_{v_{\texttt{[MASK]}} \in M_v} p(v_k \in N_{v_{\texttt{[MASK]}}} | v_{\texttt{[MASK]}} = v_i) \\
= & \prod_{v_{\texttt{[MASK]}} \in M_v} \prod_{v_k \in N_{v_{\texttt{[MASK]}}}} p(v_k | v_{\texttt{[MASK]}} = v_i) \\
= & \prod_{v_{\texttt{[MASK]}} \in M_v} \prod_{v_k \in N_{v_{\texttt{[MASK]}}}} p(v_k \longleftrightarrow v_i)
\end{aligned}$$

Patton: Language Model Pretraining on Text-Rich Networks (ACL 2023)

- Joint pretraining



$$\mathcal{L}_{\text{NMLM}} = -\sum_{i \in M_t} \log p(w_i | \boldsymbol{H}_x, \boldsymbol{z}_x)$$

$$\mathcal{L}_{\text{MNP}} = -\sum_{v_j \in M_v} \sum_{v_k \in N_{v_j}} \log p(v_j \leftrightarrow v_k)$$

$$= -\sum_{v_j \in M_v} \sum_{v_k \in N_{v_j}} \log \frac{\exp(\boldsymbol{h}_{v_j}^\top \boldsymbol{h}_{v_k})}{\exp(\boldsymbol{h}_{v_j}^\top \boldsymbol{h}_{v_k}) + \sum_{u'} \exp(\boldsymbol{h}_{v_j}^\top \boldsymbol{h}_{v'_u})}$$

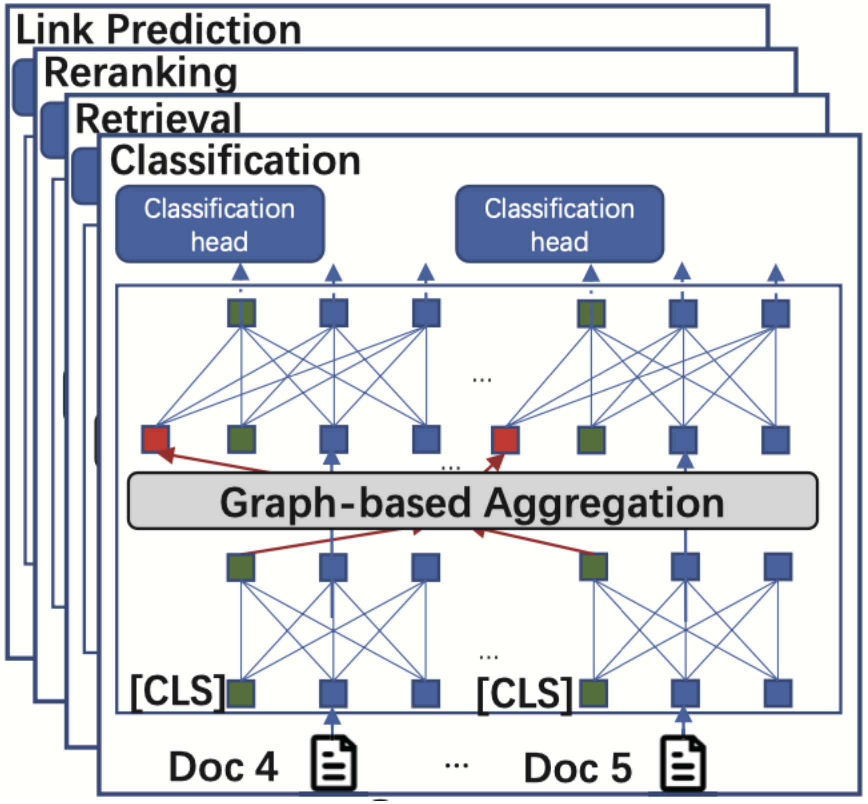
$$\mathcal{L} = \mathcal{L}_{\text{NMLM}} + \mathcal{L}_{\text{MNP}}$$

Patton: Language Model Pretraining on Text-Rich Networks (ACL 2023)

- Finetuning

  - Texts in the network (thus with neighbor info)
    - Feed both node text sequence and neighbor text sequences

  - Texts not in the network (neighbor info not available)
    - Feed text sequence and leave neighbor text sequences blank



Patton: Language Model Pretraining on Text-Rich Networks (ACL 2023)

- Datasets
  - MAPLE
    - Mathematics, Geology, Economy
    - Academic network
  - Amazon
    - Cloth, Sports
    - E-commerce network
- Downstream tasks
  - Classification
  - Retrieval
  - Reranking
  - Link prediction

Field of Study	#Nodes	#Edges	#Fine-Class	#Coarse-Class
Mathematics	490,551	2,150,584	14,271	18
Geology	431,834	1,753,762	7,883	17
Economics	178,670	1,042,253	5,205	40
Clothes	889,225	7,876,427	2,771	9
Sports	314,448	3,461,379	3,034	16

Patton: Language Model Pretraining on Text-Rich Networks (ACL 2023)

- Classification

Table 2: Experiment results on Classification. We show the mean$_{\text{std}}$ of three runs for all the methods.

Method	Mathematics		Geology		Economy		Clothes		Sports	
	Macro-F1	**Micro-F1**	**Macro-F1**	**Micro-F1**	**Macro-F1**	**Micro-F1**	**Macro-F1**	**Micro-F1**	**Macro-F1**	**Micro-F1**
BERT	$18.14_{0.07}$	$22.04_{0.32}$	$21.97_{0.87}$	$29.63_{0.36}$	$14.17_{0.08}$	$19.77_{0.12}$	$45.10_{1.47}$	$68.54_{2.25}$	$31.88_{0.23}$	$34.58_{0.56}$
GraphFormers	$18.69_{0.52}$	$23.24_{0.46}$	$22.64_{0.92}$	$31.02_{1.16}$	$13.68_{1.03}$	$19.00_{1.44}$	$46.27_{1.92}$	$68.97_{2.46}$	$43.77_{0.63}$	$50.47_{0.78}$
SciBERT	$23.50_{0.64}$	$23.10_{2.23}$	$29.49_{1.25}$	$37.82_{1.89}$	$15.91_{0.48}$	$21.32_{0.66}$	-	-	-	-
SPECTER	$23.37_{0.07}$	$29.83_{0.96}$	$30.40_{0.48}$	$38.54_{0.77}$	$16.16_{0.17}$	$19.84_{0.47}$	-	-	-	-
SimCSE (unsup)	$20.12_{0.08}$	$26.11_{0.39}$	$38.78_{0.19}$	$38.55_{0.17}$	$14.54_{0.26}$	$19.07_{0.43}$	$42.70_{2.32}$	$58.72_{0.34}$	$41.91_{0.85}$	$59.19_{0.55}$
SimCSE (sup)	$20.39_{0.07}$	$25.56_{0.00}$	$25.66_{0.28}$	$33.89_{0.40}$	$15.03_{0.53}$	$18.64_{1.32}$	$52.82_{0.87}$	$75.54_{0.98}$	$46.69_{0.10}$	$59.19_{0.55}$
LinkBERT	$15.78_{0.91}$	$19.75_{1.19}$	$24.08_{0.58}$	$31.32_{0.04}$	$12.71_{0.12}$	$16.39_{0.22}$	$44.94_{2.52}$	$65.33_{4.34}$	$35.60_{0.33}$	$38.30_{0.09}$
BERT.MLM	$23.44_{0.39}$	$31.75_{0.58}$	$36.31_{0.36}$	$48.04_{0.69}$	$16.60_{0.21}$	$22.71_{1.16}$	$46.98_{0.84}$	$68.00_{0.84}$	$62.21_{0.13}$	$75.43_{0.74}$
SciBERT.MLM	$23.34_{0.42}$	$30.11_{0.97}$	$36.94_{0.28}$	$46.54_{0.40}$	$16.28_{0.38}$	$21.41_{0.81}$	-	-	-	-
SimCSE.in-domain	$25.15_{0.09}$	$29.85_{0.20}$	$38.91_{0.08}$	$48.93_{0.14}$	$18.08_{0.22}$	$23.79_{0.44}$	$57.03_{0.20}$	$80.16_{0.31}$	$65.57_{0.35}$	$75.22_{0.18}$
PATTON	$\mathbf{27.58}_{0.03}$	$\mathbf{32.82}_{0.01}$	$39.35_{0.06}$	$48.19_{0.15}$	$19.32_{0.05}$	$25.12_{0.05}$	$\mathbf{60.14}_{0.28}$	$\mathbf{84.88}_{0.09}$	$\mathbf{67.57}_{0.08}$	$\mathbf{78.60}_{0.15}$
SciPATTON	$27.35_{0.04}$	$31.70_{0.01}$	$\mathbf{39.65}_{0.10}$	$\mathbf{48.93}_{0.06}$	$\mathbf{19.91}_{0.08}$	$\mathbf{25.68}_{0.32}$	**-**	**-**	**-**	**-**
w/o NMLM	$25.91_{0.45}$	$27.79_{2.07}$	$38.78_{0.19}$	$48.48_{0.17}$	$18.86_{0.23}$	$24.25_{0.26}$	$56.68_{0.24}$	$80.27_{0.17}$	$65.83_{0.28}$	$76.24_{0.54}$
w/o MNP	$24.79_{0.65}$	$29.44_{1.50}$	$38.00_{0.73}$	$47.82_{1.06}$	$18.69_{0.59}$	$25.63_{1.44}$	$47.35_{1.20}$	$68.50_{2.60}$	$64.23_{1.53}$	$76.03_{1.67}$

Patton: Language Model Pretraining on Text-Rich Networks (ACL 2023)

- Retrieval

Table 3: Experiment results on Retrieval. We show the mean$_{\text{std}}$ of three runs for all the methods.

Method	Mathematics		Geology		Economy		Clothes		Sports	
	R@50	R@100	R@50	R@100	R@50	R@100	R@50	R@100	R@50	R@100
BM25	20.76	24.55	19.02	20.92	19.14	22.49	15.76	15.88	22.00	23.96
BERT	$16.73_{0.17}$	$22.66_{0.18}$	$18.82_{0.39}$	$25.94_{0.39}$	$23.95_{0.25}$	$31.54_{0.21}$	$40.77_{1.68}$	$50.40_{1.41}$	$32.37_{1.09}$	$43.32_{0.96}$
GraphFormers	$16.65_{0.12}$	$22.41_{0.10}$	$18,92_{0.60}$	$25.94_{0.39}$	$24.48_{0.36}$	$32.16_{0.40}$	$41.77_{2.05}$	$51.26_{2.27}$	$32.39_{0.89}$	$43.29_{1.12}$
SciBERT	$24.70_{0.17}$	$33.55_{0.31}$	$23.71_{0.89}$	$30.94_{0.95}$	$29.80_{0.66}$	$38.66_{0.52}$	-	-	-	-
SPECTER	$23.86_{0.25}$	$31.11_{0.31}$	$26.56_{1.05}$	$34.04_{1.32}$	$31.26_{0.15}$	$40.79_{0.11}$	-	-	-	-
SimCSE (unsup)	$17.91_{0.26}$	$23.19_{0.29}$	$20.45_{0.20}$	$26.82_{0.26}$	$25.83_{0.23}$	$33.42_{0.28}$	$44.90_{0.35}$	$54.76_{0.38}$	$38.81_{0.35}$	$49.30_{0.44}$
SimCSE (sup)	$20.29_{0.41}$	$26.23_{0.51}$	$22.34_{0.49}$	$29.63_{0.55}$	$28.07_{0.38}$	$36.51_{0.37}$	$44.69_{0.59}$	$54.70_{0.77}$	$40.31_{0.43}$	$50.55_{0.41}$
LinkBERT	$17.25_{0.30}$	$23.21_{0.47}$	$17.14_{0.75}$	$23.05_{0.74}$	$22.69_{0.30}$	$30.77_{0.36}$	$28.66_{2.97}$	$37.79_{3.82}$	$31.97_{0.54}$	$41.77_{0.67}$
BERT.MLM	$20.69_{0.21}$	$27.17_{0.25}$	$32.13_{0.36}$	$41.74_{0.42}$	$27.13_{0.04}$	$36.00_{0.14}$	$52.41_{1.71}$	$63.72_{1.79}$	$54.10_{0.81}$	$63.14_{0.83}$
SciBERT.MLM	$20.65_{0.21}$	$27.67_{0.32}$	$31.65_{0.71}$	$40.52_{0.76}$	$29.23_{0.67}$	$39.18_{0.73}$	-	-	-	-
SimCSE.in-domain	$24.54_{0.05}$	$31.66_{0.09}$	$33.97_{0.07}$	$44.09_{0.19}$	$28.44_{0.31}$	$37.81_{0.27}$	$61.42_{0.84}$	$72.25_{0.86}$	$53.77_{0.22}$	$63.73_{0.30}$
PATTON	$27.44_{0.15}$	$34.97_{0.21}$	$34.94_{0.23}$	$45.01_{0.28}$	$32.10_{0.51}$	$42.19_{0.62}$	$\mathbf{68.62}_{0.38}$	$\mathbf{77.54}_{0.19}$	$\mathbf{58.63}_{0.31}$	$\mathbf{68.53}_{0.55}$
SciPATTON	$\mathbf{31.40}_{0.52}$	$\mathbf{40.38}_{0.66}$	$\mathbf{40.69}_{0.52}$	$\mathbf{51.31}_{0.48}$	$\mathbf{35.82}_{0.69}$	$46.05_{0.69}$	-	-	-	-
w/o NMLM	$30.85_{0.14}$	$39.89_{0.23}$	$39.29_{0.07}$	$49.59_{0.11}$	$35.17_{0.31}$	$\mathbf{46.07}_{0.20}$	$65.60_{0.26}$	$75.19_{0.32}$	$57.05_{0.14}$	$67.22_{0.12}$
w/o MNP	$22.47_{0.07}$	$30.20_{0.15}$	$31.28_{0.89}$	$40.54_{0.97}$	$29.54_{0.36}$	$39.57_{0.57}$	$60.20_{0.73}$	$69.85_{0.52}$	$51.73_{0.41}$	$60.35_{0.78}$

Patton: Language Model Pretraining on Text-Rich Networks (ACL 2023)

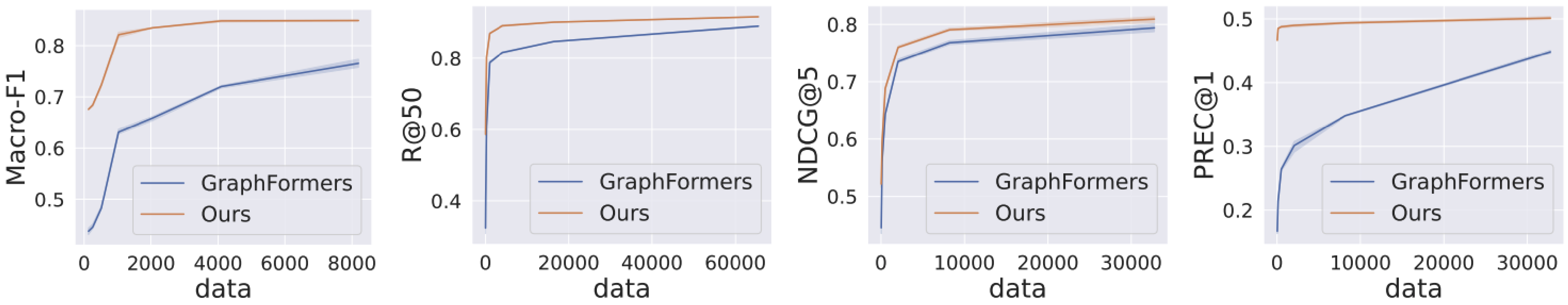
- Link prediction

Table 5: Experiment results on Link Prediction. We show the mean$_{\text{std}}$ of three runs for all the methods.

Method	Mathematics		Geology		Economy		Clothes		Sports	
	PREC@1	MRR	PREC@1	MRR	PREC@1	MRR	PREC@1	MRR	PREC@1	MRR
BERT	$6.60_{0.16}$	$12.96_{0.34}$	$6.24_{0.76}$	$12.96_{1.34}$	$4.12_{0.08}$	$9.23_{0.15}$	$24.17_{0.41}$	$34.20_{0.45}$	$16.48_{0.45}$	$25.35_{0.52}$
GraphFormers	$6.91_{0.29}$	$13.42_{0.34}$	$6.52_{1.17}$	$13.34_{1.81}$	$4.16_{0.21}$	$9.28_{0.28}$	$23.79_{0.69}$	$33.79_{0.66}$	$16.69_{0.36}$	$25.74_{0.48}$
SciBERT	$14.08_{0.11}$	$23.62_{0.10}$	$7.15_{0.26}$	$14.11_{0.39}$	$5.01_{1.04}$	$10.48_{1.79}$	-	-	-	-
SPECTER	$13.44_{0.5}$	$21.73_{0.65}$	$6.85_{0.22}$	$13.37_{0.34}$	$6.33_{0.29}$	$12.41_{0.33}$	-	-	-	-
SimCSE (unsup)	$9.85_{0.10}$	$16.28_{0.12}$	$7.47_{0.55}$	$14.24_{0.89}$	$5.72_{0.26}$	$11.02_{0.34}$	$30.51_{0.09}$	$40.40_{0.10}$	$22.99_{0.07}$	$32.47_{0.06}$
SimCSE (sup)	$10.35_{0.52}$	$17.01_{0.72}$	$10.10_{0.04}$	$17.80_{0.07}$	$5.72_{0.26}$	$11.02_{0.34}$	$35.42_{0.06}$	$46.07_{0.06}$	$27.07_{0.15}$	$37.44_{0.16}$
LinkBERT	$8.05_{0.14}$	$13.91_{0.09}$	$6.40_{0.14}$	$12.99_{0.17}$	$2.97_{0.08}$	$6.79_{0.15}$	$30.33_{0.56}$	$39.59_{0.64}$	$19.83_{0.09}$	$28.32_{0.04}$
BERT.MLM	$17.55_{0.25}$	$29.22_{0.26}$	$14.13_{0.19}$	$25.36_{0.20}$	$9.02_{0.09}$	$16.72_{0.15}$	$42.71_{0.31}$	$54.54_{0.35}$	$29.36_{0.09}$	$41.60_{0.05}$
SciBERT.MLM	$22.44_{0.08}$	$34.22_{0.05}$	$16.22_{0.03}$	$27.02_{0.07}$	$9.80_{0.00}$	$17.72_{0.01}$	-	-	-	-
SimCSE.in-domain	$33.55_{0.05}$	$46.07_{0.07}$	$24.56_{0.06}$	$36.89_{0.11}$	$16.77_{0.10}$	$26.93_{0.01}$	$\mathbf{60.41}_{0.03}$	$\mathbf{71.86}_{0.06}$	$49.17_{0.04}$	$63.48_{0.03}$
PATTON	$70.41_{0.11}$	$80.21_{0.04}$	$44.76_{0.05}$	$57.71_{0.04}$	$57.04_{0.05}$	$68.35_{0.04}$	$58.59_{0.12}$	$70.12_{0.12}$	$46.68_{0.09}$	$60.96_{0.23}$
SciPATTON	$\mathbf{71.22}_{0.17}$	$\mathbf{80.79}_{0.10}$	$\mathbf{44.95}_{0.24}$	$\mathbf{57.84}_{0.25}$	$\mathbf{57.36}_{0.26}$	$\mathbf{68.71}_{0.31}$	-	-	-	-
w/o NMLM	$71.04_{0.13}$	$80.60_{0.07}$	$44.33_{0.23}$	$57.29_{0.22}$	$56.64_{0.25}$	$68.12_{0.16}$	$60.30_{0.03}$	$71.67_{0.07}$	$\mathbf{49.72}_{0.06}$	$\mathbf{63.76}_{0.04}$
w/o MNP	$63.06_{0.23}$	$74.26_{0.11}$	$33.84_{0.60}$	$47.02_{0.65}$	$44.46_{0.03}$	$57.05_{0.04}$	$49.62_{0.06}$	$61.61_{0.01}$	$36.05_{0.20}$	$49.78_{0.25}$

Patton: Language Model Pretraining on Text-Rich Networks (ACL 2023)

- How pretraining help the model?
  - Finetune data size study



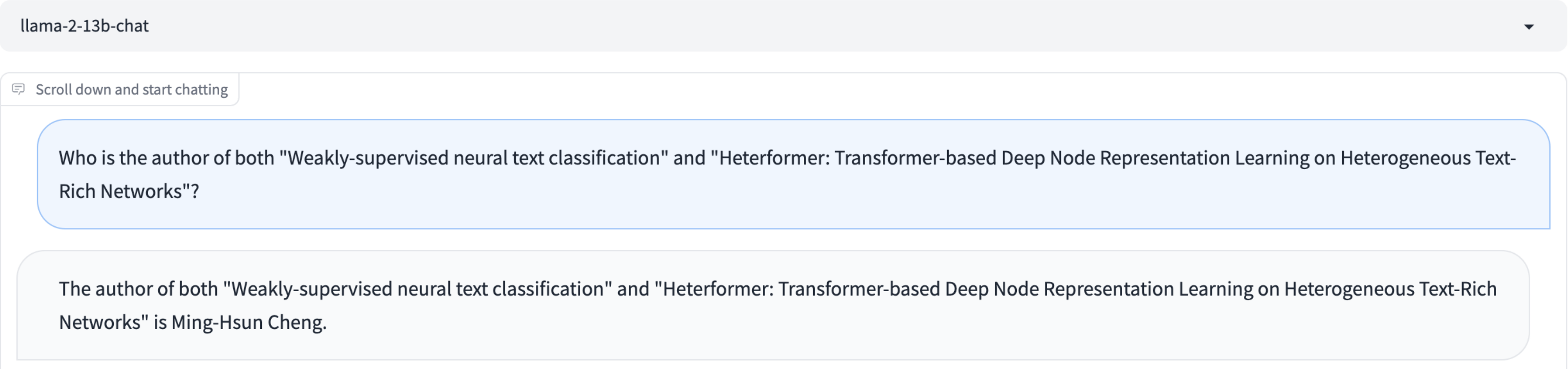
(a) Classification (b) Retrieval (c) Reranking (d) Link Prediction

Outline

- **Motivation:** Why Mining Text-attributed Graphs?
- **Content:** Mining Text-attributed Graphs with Language Models
  - Representation learning with language models on text-attributed graphs
  - Language model pretraining text-attributed graphs
  - Large language model reasoning on text-attributed graphs

Graph Chain-of-Thought: Augmenting Large Language Models by Reasoning on Graphs (ACL 2024)

- ❑ **Motivation**
  - ❑ Large language models suffer from hallucination and misinformation.

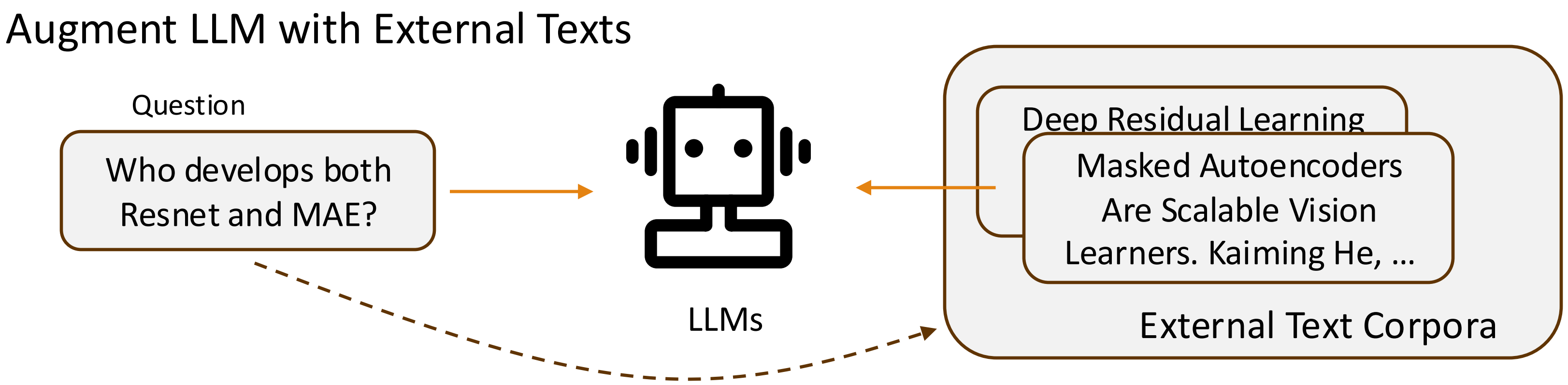


Hallucinating!

Graph Chain-of-Thought: Augmenting Large Language Models by Reasoning on Graphs (ACL 2024)

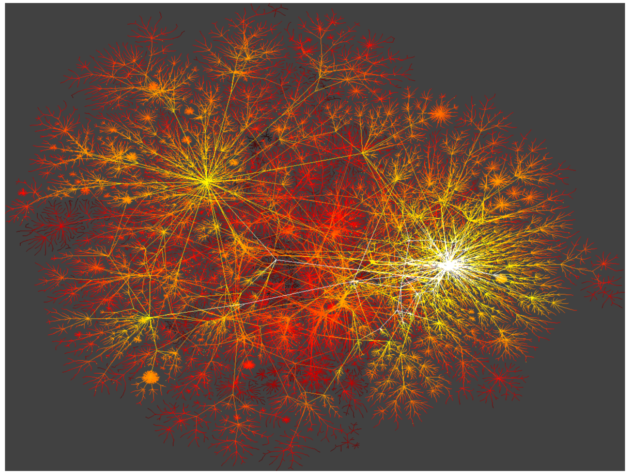
Motivation

- Existing works propose to augment LLMs with individual text units retrieved from external knowledge corpora to alleviate the issue (RAG).

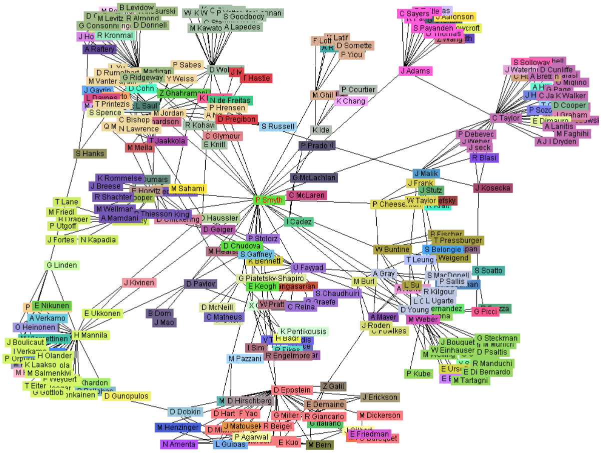


Graph Chain-of-Thought: Augmenting Large Language Models by Reasoning on Graphs (ACL 2024)

- **Motivation**
  - However, in many domains, texts are interconnected which form a (text-attributed) graph.
    - Legal case opinions are linked by citation relationships.
    - Web pages are connected by hyperlinks (Common Crawl).



World-Wide Web



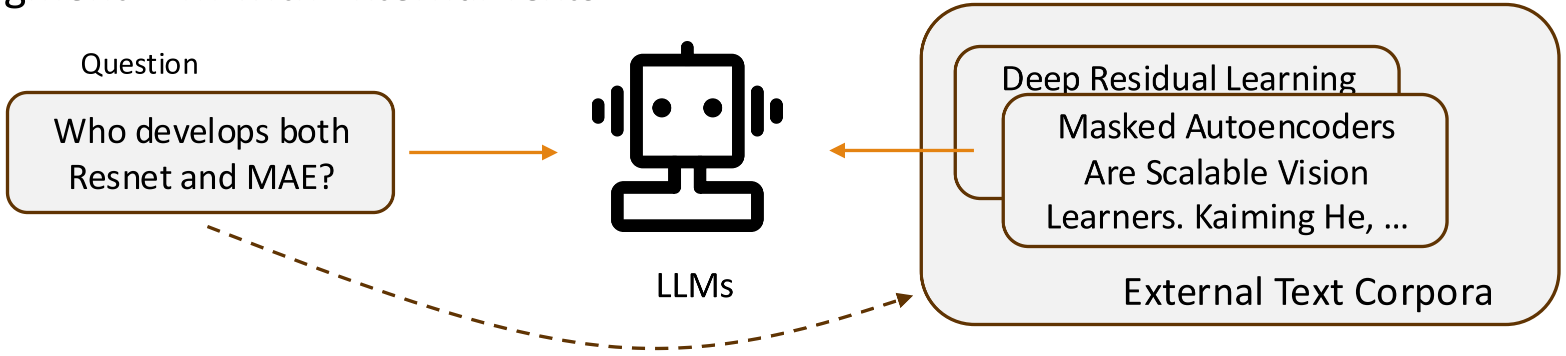
Co-author network

Graph Chain-of-Thought: Augmenting Large Language Models by Reasoning on Graphs (ACL 2024)

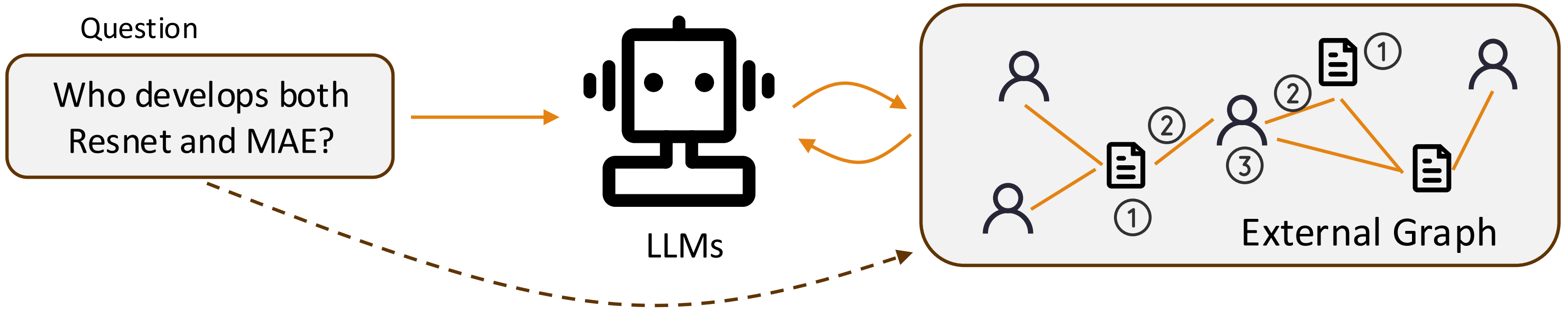
❑ Motivation

- This motivates us to explore the problem of augmenting LLMs with external graphs.

Augment LLM with External Texts

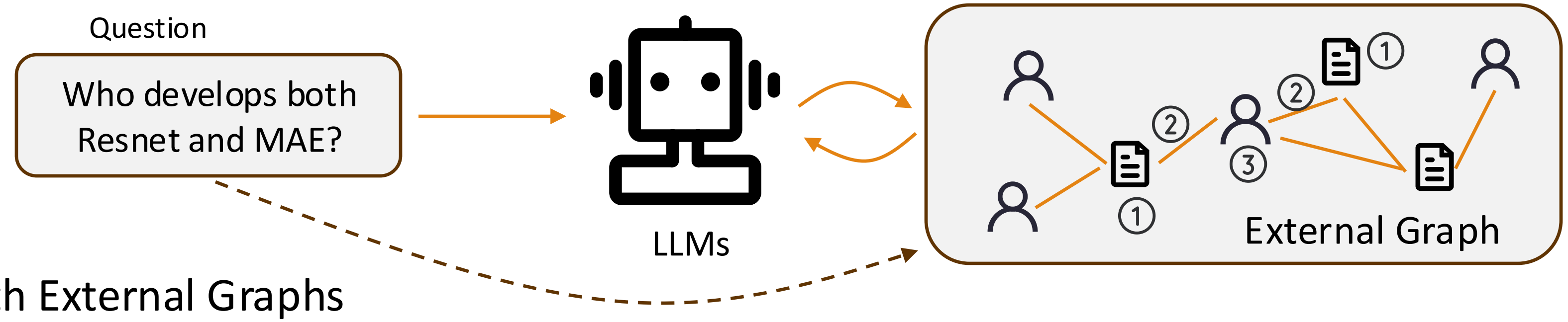


Augment LLM with External Graphs



Graph Chain-of-Thought: Augmenting Large Language Models by Reasoning on Graphs (ACL 2024)

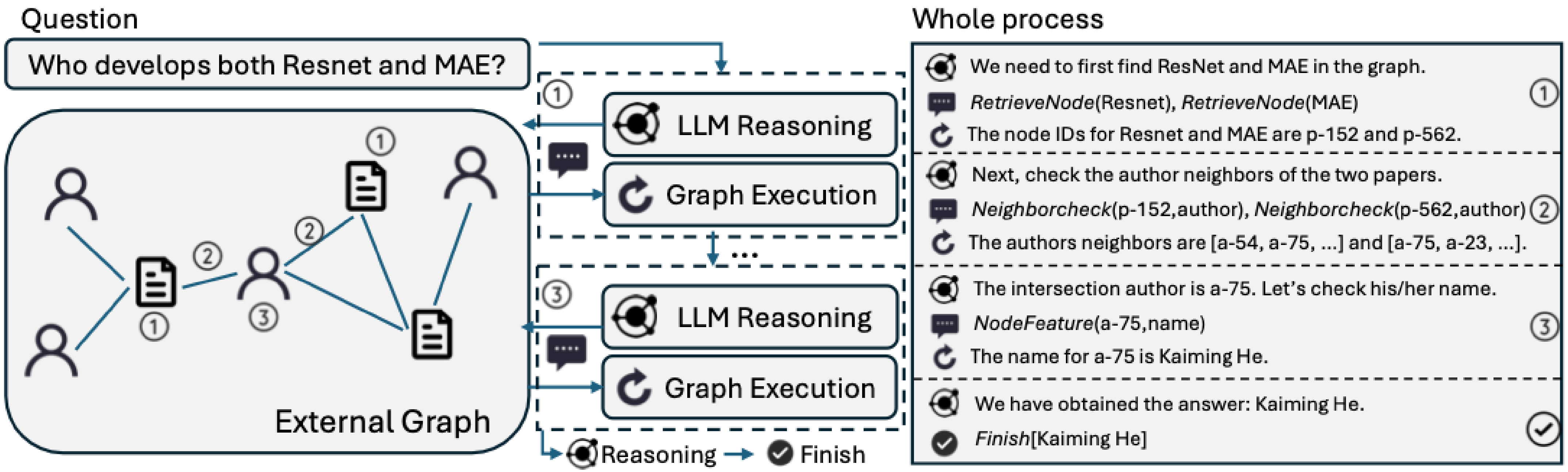
- **Can RAG be directly adopted for LLMs on graphs?**
  - Structure context:
    - Retrieval augmentation can find individual nodes/texts from the graphs.
    - However, knowledge on the graph also lies in the structure which cannot be captured by single nodes.
  - Graph size explosion:
    - It is feasible to convert local subgraph structure into text descriptions as the input contexts to LLMs.
    - However, the size of the local subgraph increases exponentially as the hop number increases.
    - It will result in an excessively long context sequence and cause LLM to be lost in the middle.



Augment LLM with External Graphs

Graph Chain-of-Thought: Augmenting Large Language Models by Reasoning on Graphs (ACL 2024)

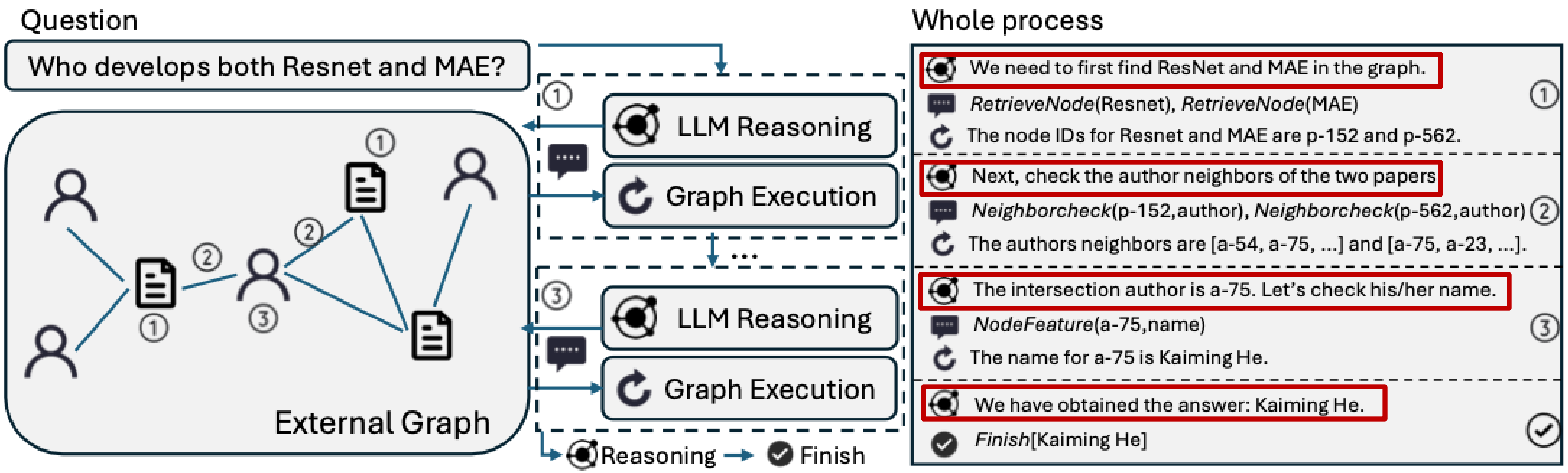
- **Framework**
  - Iterative reasoning, interaction and execution.



Graph Chain-of-Thought: Augmenting Large Language Models by Reasoning on Graphs (ACL 2024)

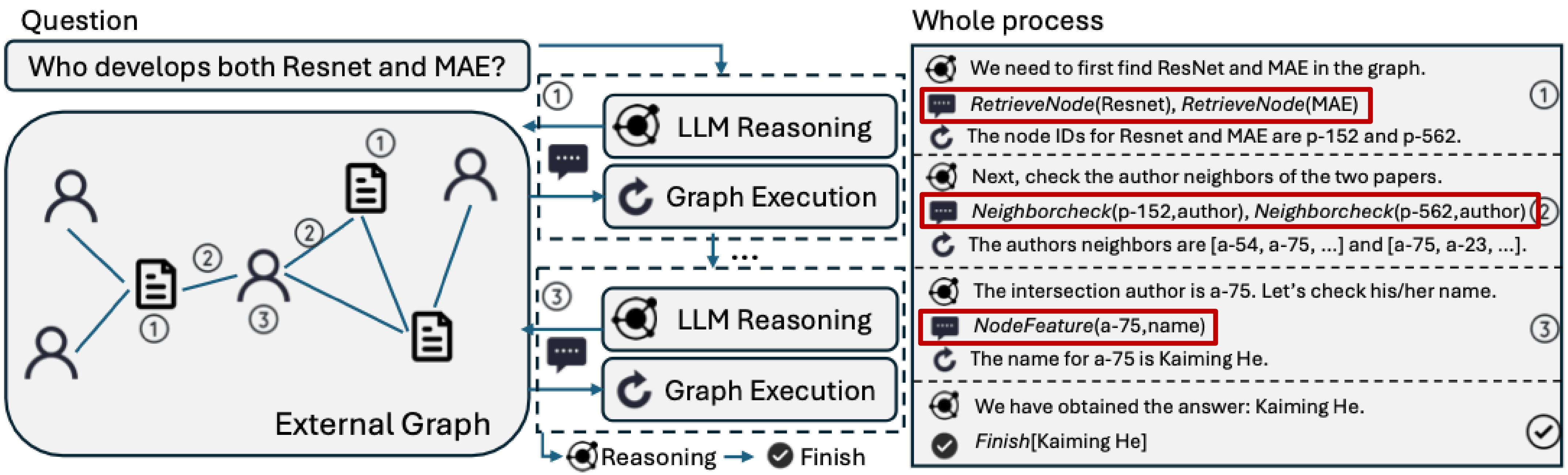
❑ LLM reasoning

- ❑ LLM conduct reasoning on what further external information from graph is needed.
- ❑ If the question is answerable with the current contexts from graphs.



Graph Chain-of-Thought: Augmenting Large Language Models by Reasoning on Graphs (ACL 2024)

- **Interaction between LLMs and graphs**
  - Let LLMs know how to interact with the graphs and fetch relevant information.

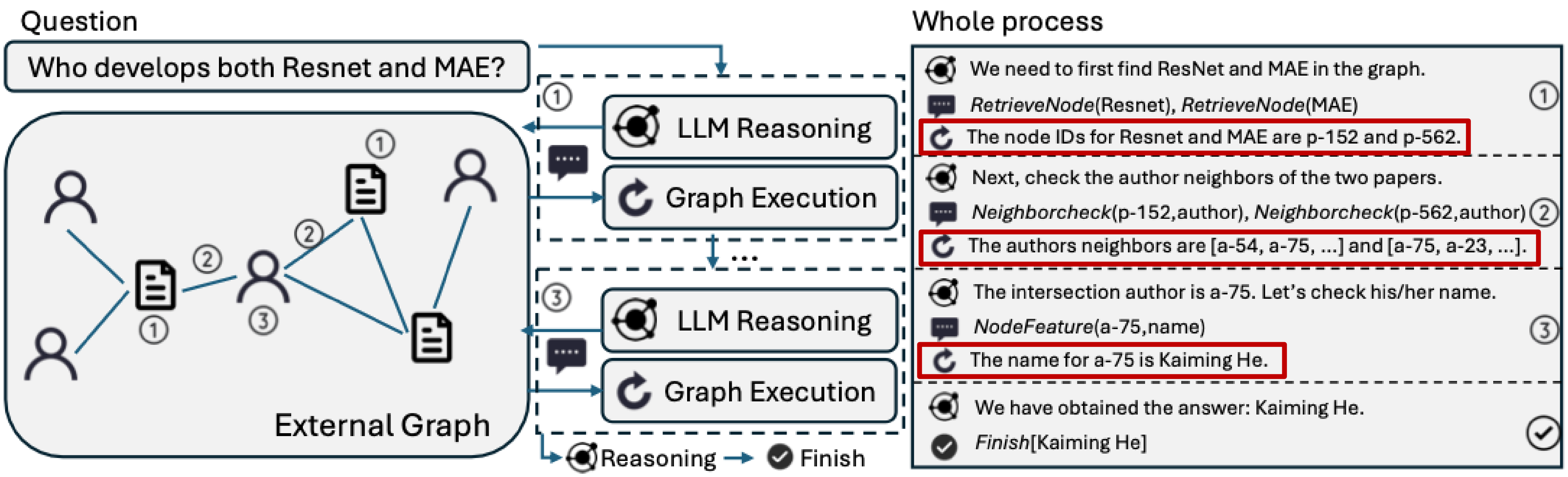


Graph Chain-of-Thought: Augmenting Large Language Models by Reasoning on Graphs (ACL 2024)

- **Interaction between LLMs and graphs**
  - We pre-define four graph functions to cover both the semantic and structure information on graphs:
    - RetrieveNode(Text): Identify related nodes in the graph with semantic search.
    - NodeFeature(NodeID, FeatureName): Extract the textual feature information for a specific node.
    - NeighborCheck(NodeID, NeighborType): Return the neighboring information for a specific node.
    - NodeDregree(NodeID, NeighborType): Return the degree of a specific neighbor type for a node.

Graph Chain-of-Thought: Augmenting Large Language Models by Reasoning on Graphs (ACL 2024)

- **Execution on graphs**
  - Call the functions and fetch relevant information from the graph.



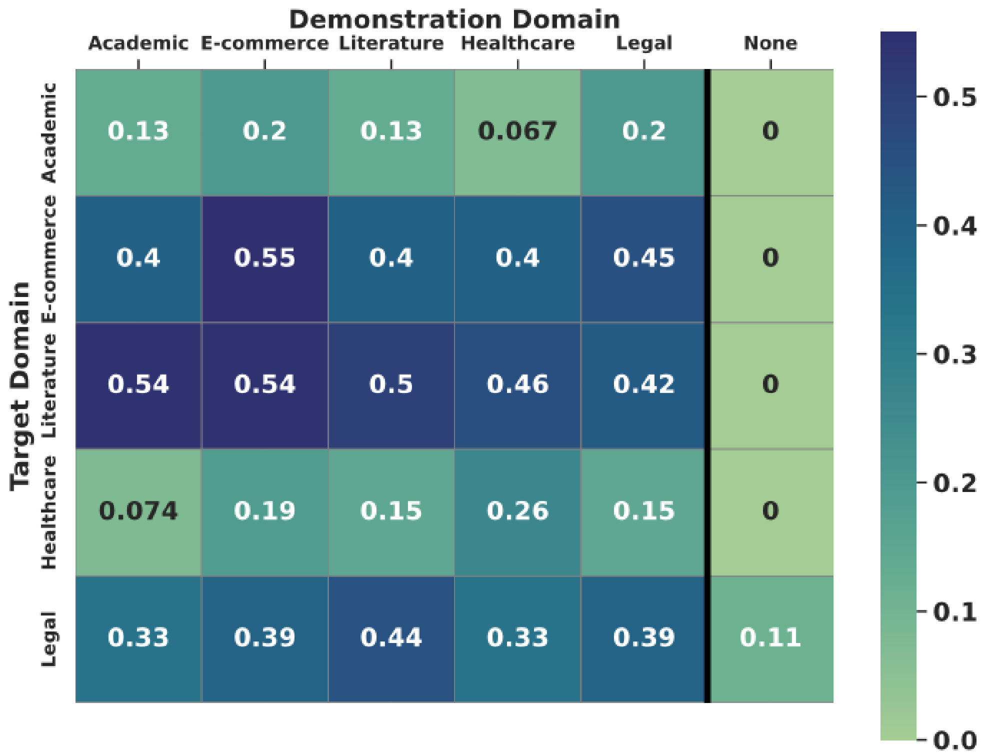
Graph Chain-of-Thought: Augmenting Large Language Models by Reasoning on Graphs (ACL 2024)

Overall performance

	Model	Academic		E-commerce		Literature		Healthcare		Legal	
		R-L	GPT4score	R-L	GPT4score	R-L	GPT4score	R-L	GPT4score	R-L	GPT4score
Base	LLaMA-2-13b-chat	8.13	8.03	7.01	12.00	5.32	20.83	5.25	13.70	15.97	16.11
	Mixtral-8x7b	9.02	8.14	12.54	18.00	7.50	22.50	3.88	20.00	12.74	16.11
	GPT-3.5-turbo	6.05	12.80	9.18	23.50	10.43	26.67	5.83	14.44	10.51	20.00
Text RAG	LLaMA-2-13b-chat	8.69	8.52	9.23	12.50	7.61	20.00	1.44	5.93	15.37	16.67
	Mixtral-8x7b	8.44	8.02	23.14	29.50	13.35	27.92	3.22	16.67	19.69	25.00
	GPT-3.5-turbo	5.83	9.91	14.06	20.00	10.04	20.83	4.57	8.52	18.14	23.89
Graph RAG	LLaMA-2-13b	22.01	22.97	12.48	20.00	9.25	20.00	2.97	4.81	17.98	17.22
	Mixtral-8x7b	27.77	31.20	32.87	37.00	20.08	33.33	8.66	15.19	23.48	25.56
	GPT-3.5-turbo	18.45	26.98	17.52	28.00	14.94	24.17	8.69	14.07	18.66	22.22
	GRAPH-COT	**31.89**	**33.48**	**42.40**	**44.50**	**41.59**	**46.25**	**22.33**	**28.89**	**30.52**	**28.33**

Graph Chain-of-Thought: Augmenting Large Language Models by Reasoning on Graphs (ACL 2024)

❑ How important are the demonstrations for Graph-CoT?



- ❑ LLMs suffer if given insufficient instructions (only graph definition and interaction function definitions).
- ❑ In-domain demonstrations (diagonal) perform quite well, and Graph-CoT is overall robust to demonstration domain-shift.

Graph Chain-of-Thought: Augmenting Large Language Models by Reasoning on Graphs (ACL 2024)

- **How different LLMs perform in Graph-CoT?**

Model	GPT4score
GRAPH-COT	
w. LLaMA-2-13b-chat	16.04
w. Mixtral-8x7b	36.46
w. GPT-3.5-turbo	36.63
w. GPT-4	46.28

- An LLM with more advanced instruction following ability and reasoning ability (i.e., GPT-4) can contribute to better performance in Graph-CoT.

Graph Chain-of-Thought: Augmenting Large Language Models by Reasoning on Graphs (ACL 2024)

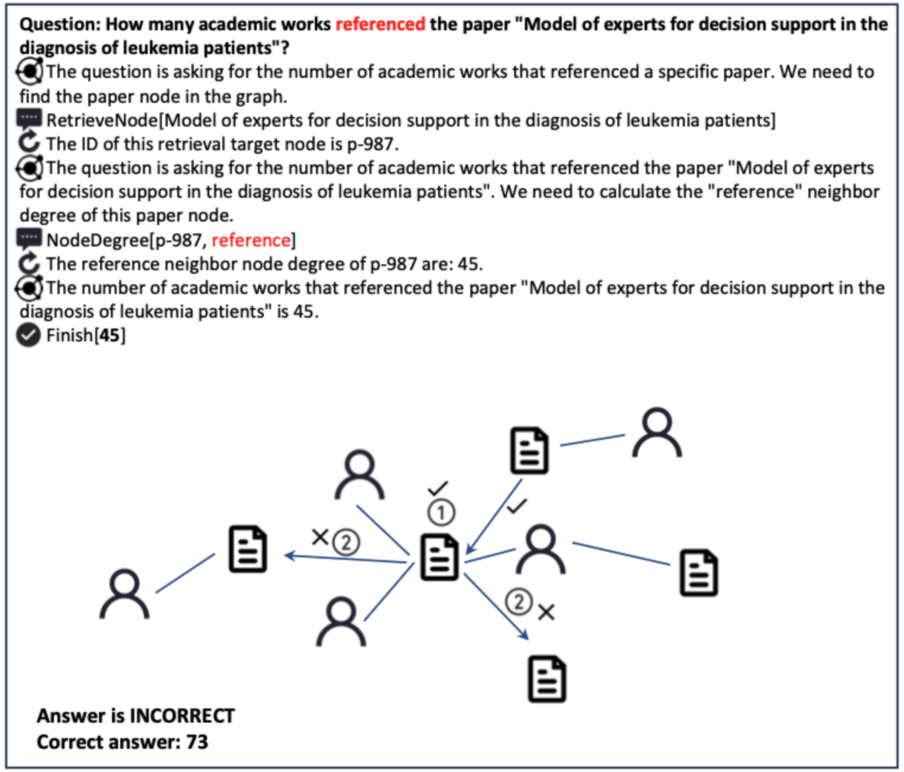
❑ Graph RAG vs Graph-CoT

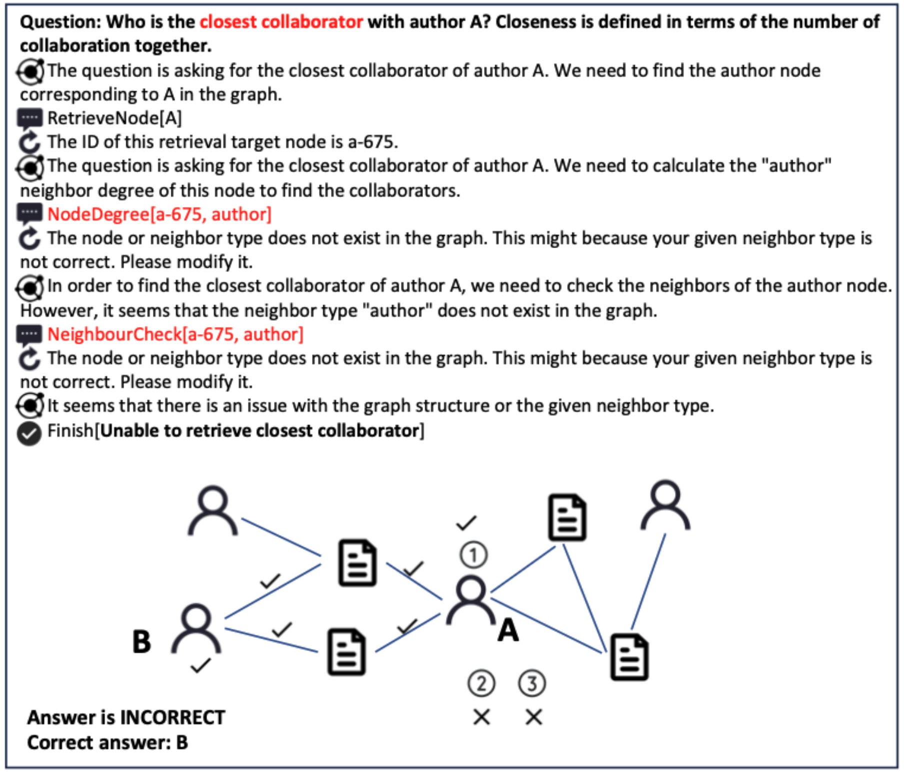
Model	**GPT4score**
GPT-3.5-turbo	19.48
+ node retrieval	16.63
+ 1-hop subgraph retrieval	*23.09*
+ 2-hop subgraph retrieval	22.12
+ GRAPH-COT	**36.29**

- ❑ Retrieving 1-hop ego-graph performs the best but still underperforms Graph-CoT.
- ❑ The number of nodes/texts grow exponentially as the hop number grows linearly.
- ❑ A large-hop ego-graph will lead to a super long context -> lost in the middle.

Graph Chain-of-Thought: Augmenting Large Language Models by Reasoning on Graphs (ACL 2024)

❑ Case Study





- ❑ The LLM sometimes refers to the occurrence of the word rather than understanding its semantic meaning.
- ❑ The LLM sometimes misunderstands the structure of the graph, resulting in interaction failures.

Thank you!
Q/A