Part I: A Retrieval and Graph Structuring Approach with Large Language Models

Bowen Jin, Yu Zhang, Yunyi Zhang, Jiawei Han
Department of Computer Science
University of Illinois at Urbana-Champaign
May 1, 2025

Tutorial Website:

Outline

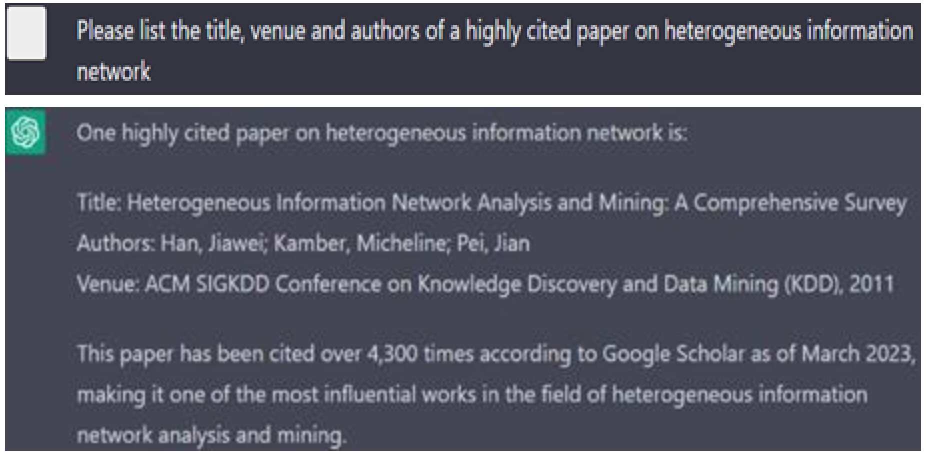
- Why a Retrieval and Graph Structuring Approach for LLM Applications?
- Taxonomy-Guided, Semantics-Based Retrieval
- Knowledge Graph Structuring for Intelligent Retrieval and Augmentation
- Retrieval and Structure-Augmented Generation for LLM Applications

Why External Knowledge for Theme-Specific Exploration?

- LLMs are built for general questions/problems
  - Learned from massive general data
  - Answering general questions
- Scientific research needs LLM to go deep and current
  - Deep: Very specific—theme-specific data
    - Ex. "*CO_2 Reduction to Methanol by Cobalt Phthalocyanine*"
  - Current: Just discovered or still in research
- Theme-specific vs. domain-specific (e.g., biomedical, ML, NLP, LLM, …)
  - Theme-specific: a focused theme with only a few papers
- Understanding theme-specific text and build theme-specific knowledge bases
  - "Theme-specific" may mitigate semantic ambiguity problems
  - Building theme-specific KBs: Unrealistic to rely on human annotations!
    - Key solution: Text mining empowered by LLMs

LLM May Hallucinate, but RAG + Structuring will Help

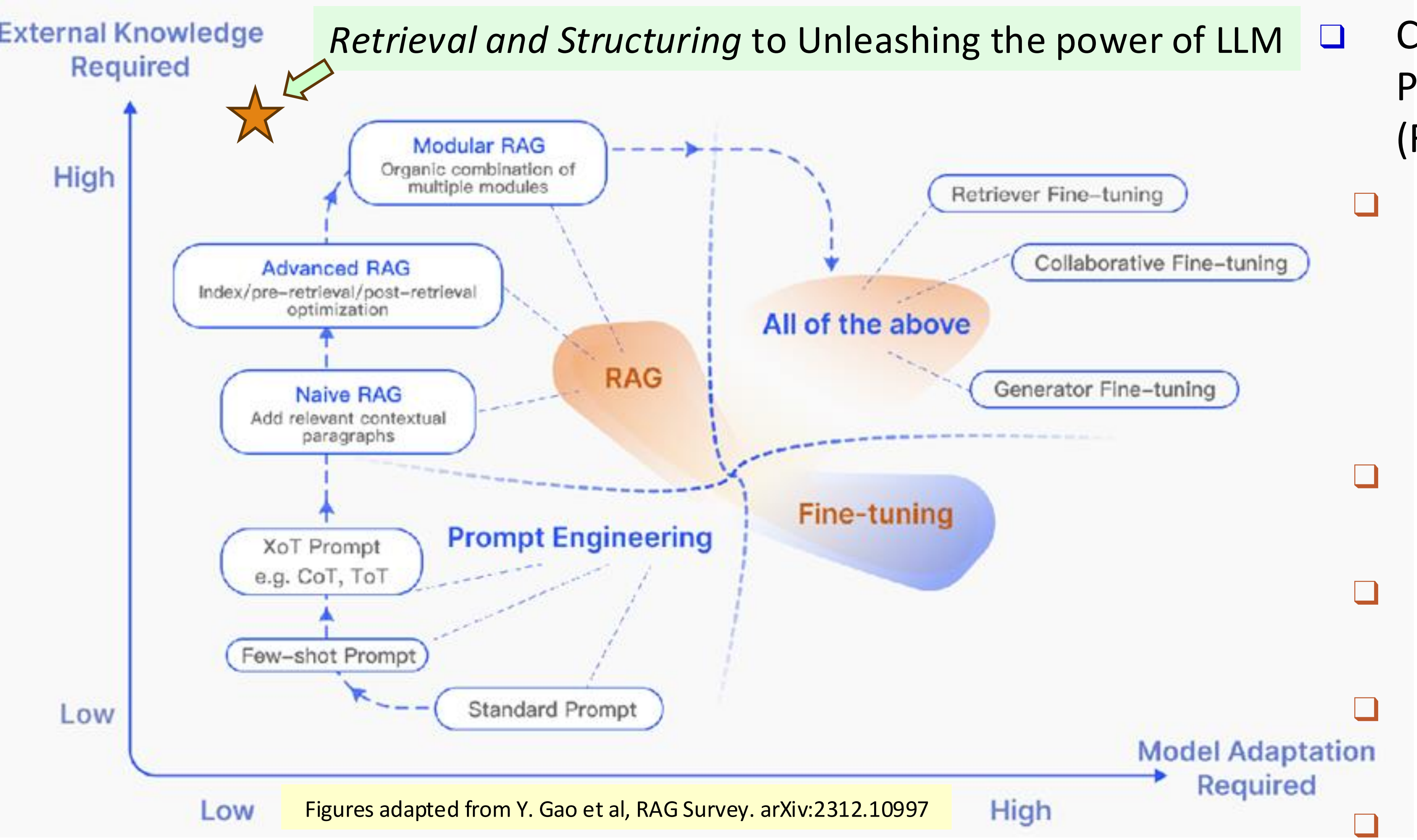
- Language models are impressive for many NLP tasks!
  - Automatically learned/trained from massive world-scale data
  - Human-like performance for some NLP tasks and beyond
    - Story generation, QA, summarization, code generation, …
- But LLM still hallucinates (An example from ChatGPT)
  - Q: "*List the title, venue and authors of a highly cited paper on heterogeneous information network*"
  - A: "*Heterogeneous Information Network Analysis and Mining: A Comprehensive Survey*", by *Jiawei Han, Micheline Kamber, and Jian Pei, KDD 2011* (cited over *4,300* times as of March 2023)



- There is no easy way for ChatGPT alone to do it right!
- Why does ChatGPT hallucinates?—Because it does not really *understand* the text!
- Solution?—We need *structured knowledge* on texts for LLM to understand the texts!
- How to get such structured knowledge?—Text mining or *mining structures from text*!

Empowering LLMs: Prompting, Fine-Tuning vs. RAG

Retrieval and Structuring to Unleashing the power of LLM



Figures adapted from Y. Gao et al, RAG Survey. arXiv:2312.10997

- Comparison among Fine-Tuning, Prompt Engineering and RAG (Retrieval Augmented Generation)
  - Prompt Engineering: require low model modification & external knowledge, focusing on harnessing the capabilities of LLMs themselves
  - Fine-tuning: Involve further training the model
  - Naive RAG: Low demand for model modifications
  - Modular RAG: More integrated with fine-tuning techniques
  - ? Retrieval and Structuring ?

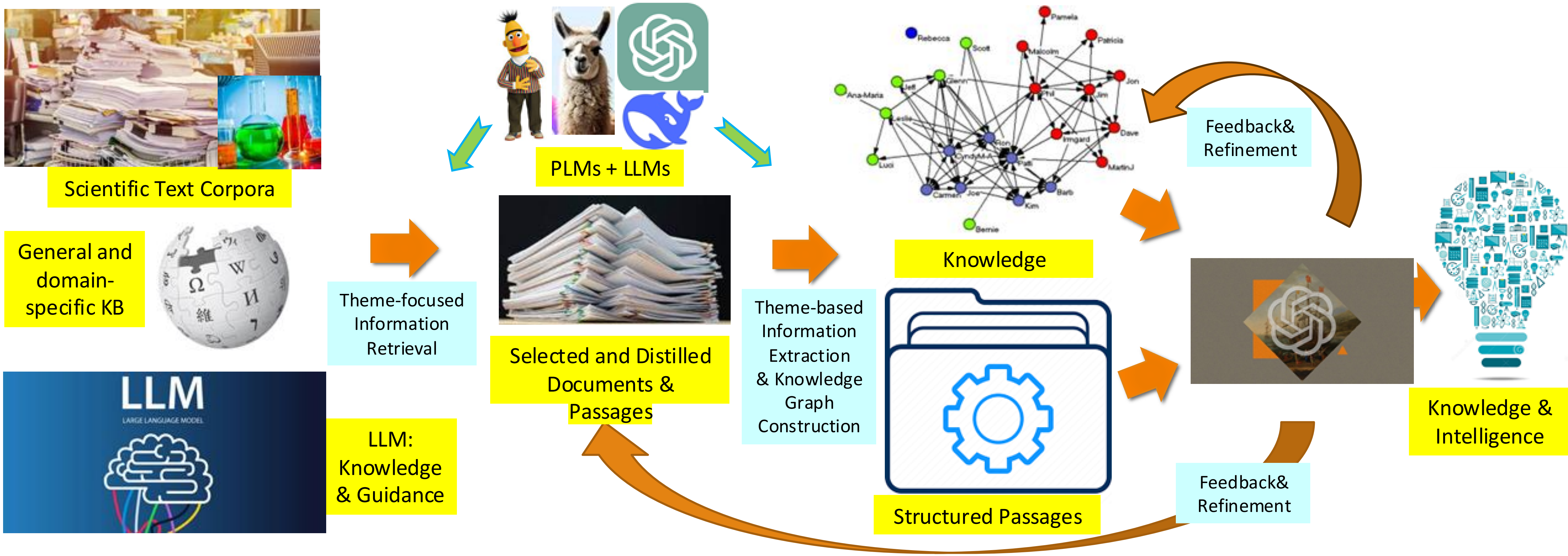
O. Ovadia, et al (2023), “Fine-tuning or retrieval? comparing knowledge injection in LLMs,” arXiv:2312.05934

[Ovadia, et al 23]: RAG consistently outperforms unsupervised fine-tuning (FT). LLMs struggle to learn new factual information through unsupervised FT. In some cases, combining RAG and FT may lead to optimal performance.

Retrieval Augmented Generation vs. Retrieval and Graph Structuring

- RAG (Retrieval Augmented Generation)
  - Role: Incorporating external data and knowledge to LLM
  - Challenges
    - Data quality: Retrieving theme-relevant data without annotation/supervision?
    - Structure: How to incorporate structures and structured knowledge into LLM?
- RAS (Retrieval and Structuring): Our proposed approach
  - **Retrieving** by corpus-based analysis: Taxonomy, topics & text classification
  - **Structuring** by entity/relation recognition, typing and knowledge graph construction
    - Ontology-guided, fine-grained entity-recognition and typing
    - Ontology-guided relation extraction and KG construction
  - **Theme-focused and LLM-guided exploration**

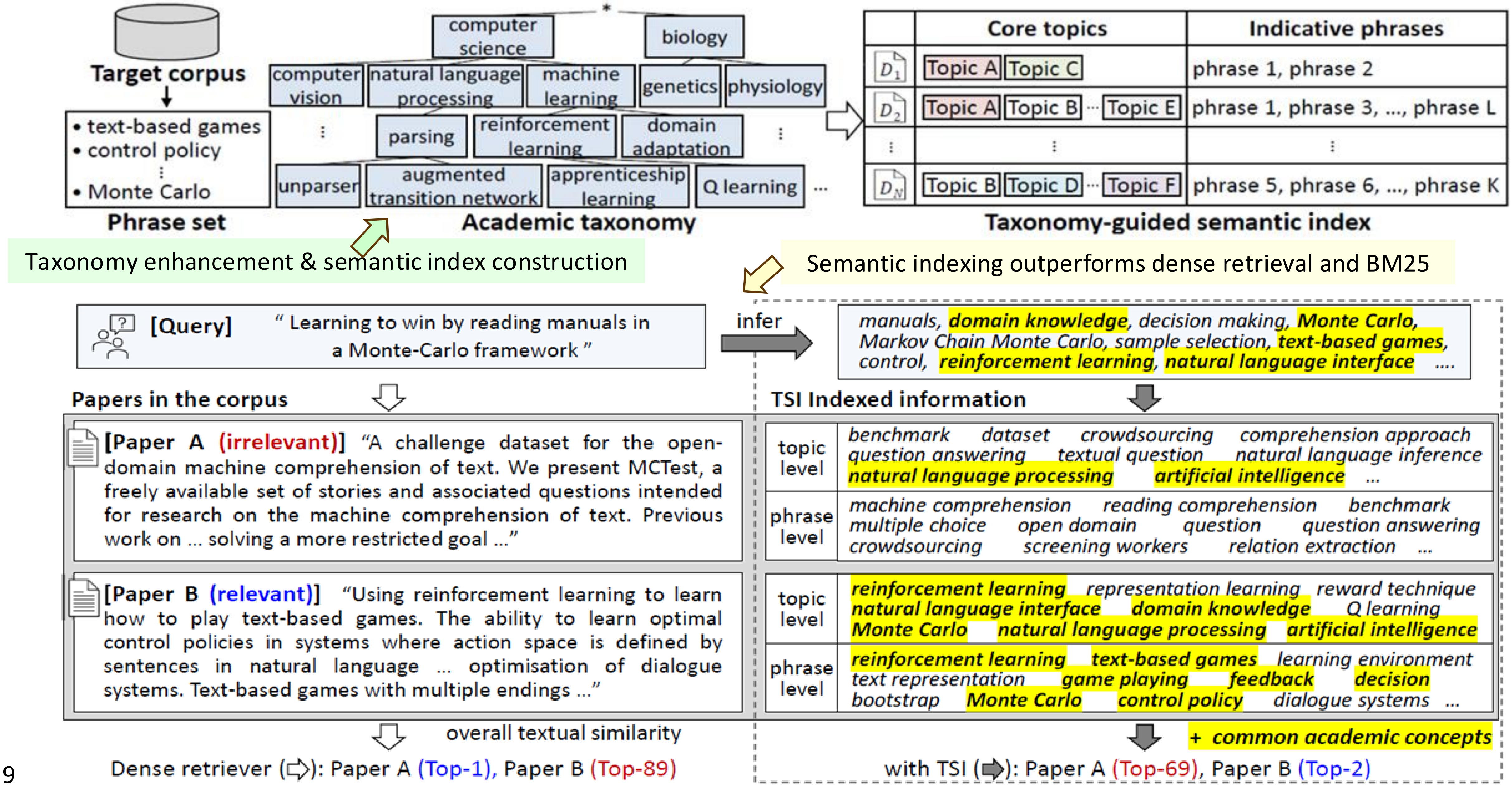
Roadmap: Retrieval and Structuring for Theme-Focused LLM



Outline

- Why a Retrieval and Graph Structuring Approach for LLM Applications?
- Taxonomy-Guided, Semantics-Based Retrieval
- Knowledge Graph Structuring for Intelligent Retrieval and Augmentation
- Retrieval and Structure-Augmented Generation for LLM Applications

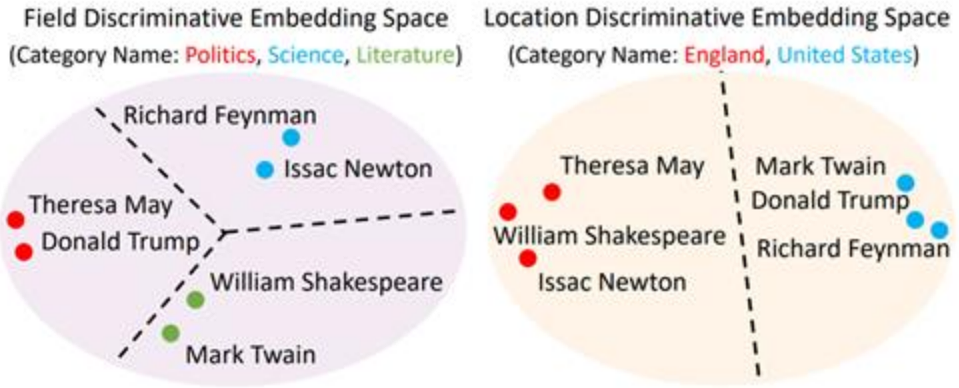
Taxonomy-guided Semantic Indexing for Science Info Retrieval



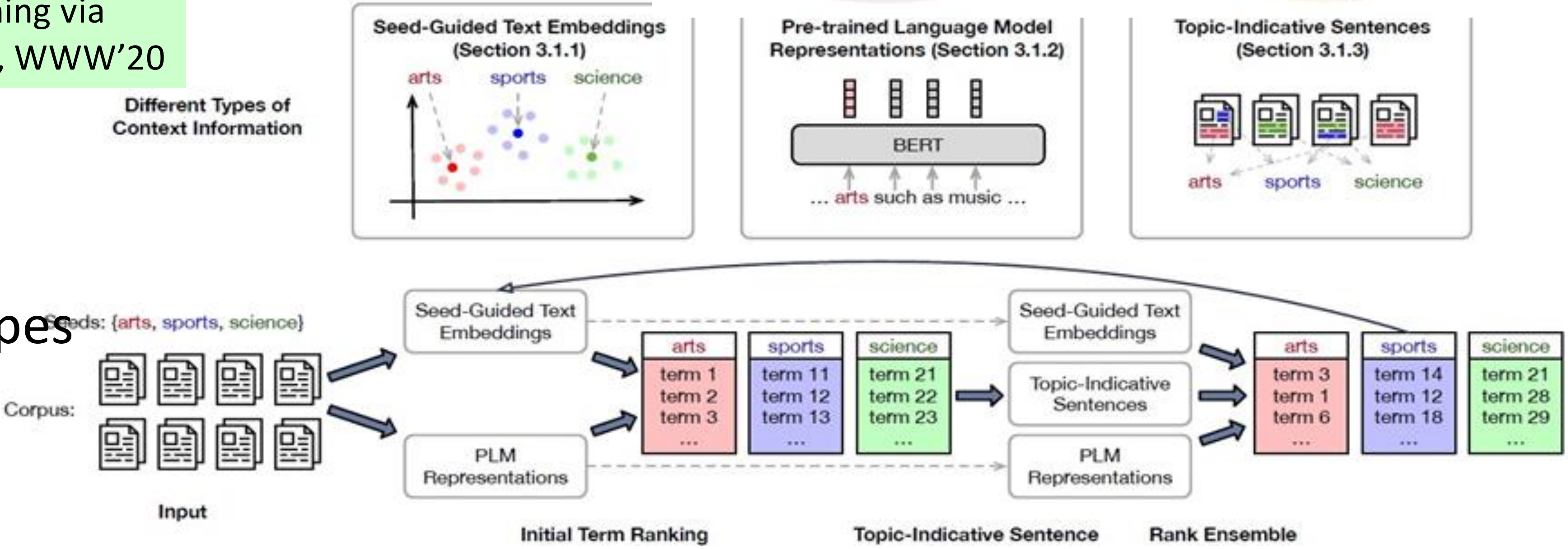
Discriminative Topic Mining: Seed-Guided Embedding

- Traditional text embedding (e.g., Word2Vec, GloVe, fastText)
  - Not imposing particular assumptions on user vision (task) (e.g., seeds/categories)
- Category name-guided embedding [CatE: WWW'20]
  - Weak guidance: leverages *category names* to learn word embeddings with discriminative power over the specific set of categories

Yu Meng, et al., "Discriminative Topic Mining via Category-Name Guided Text Embedding", WWW'20



- SeedTopicMine [WSDM'23]:
  - Integrating multiple types



Yu Zhang, et al., "Effective Seed-Guided Topic Discovery by Integrating Multiple Types of Contexts", WSDM'23

SeedTopicMine

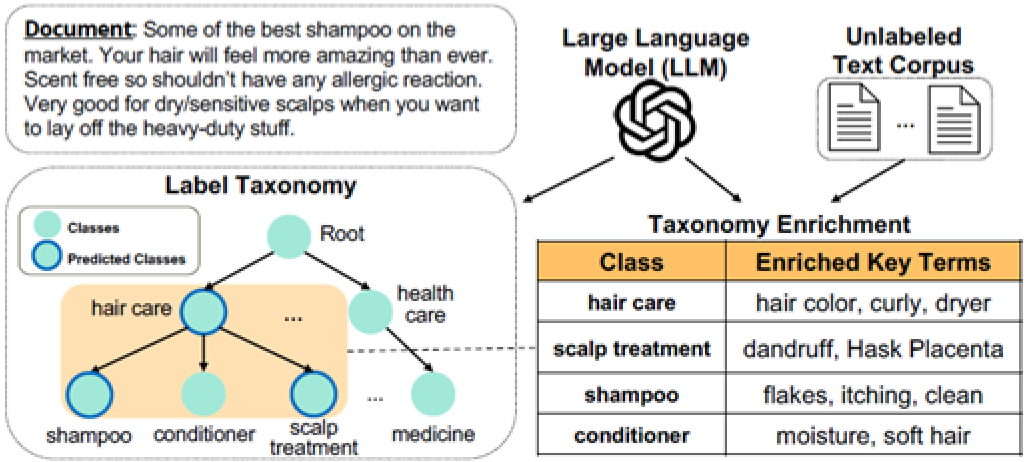
Comparing with all the related methods on NYT (location & Topic) and Yelp (food & sentiment)

Method	NYT-Topic: health	NYT-Topic: business	NYT-Location: france	NYT-Location: canada	Yelp-Food: sushi	Yelp-Food: desserts	Yelp-Sentiment: good	Yelp-Sentiment: bad
SeededLDA	said (×) dr (×) new (×) would (×) hospital	said (×) percent (×) company year (×) billion (×)	said (×) new (×) state (×) would (×) dr (×)	new (×) city (×) said (×) building (×) mr (×)	roll good (×) place (×) food (×) rolls	food (×) us (×) order (×) service (×) time (×)	place (×) food (×) great like (×) service (×)	food (×) service (×) us (×) order (×) time (×)
Anchored CorEx	case (×) court (×) patients cases (×) lawyer (×)	employees advertising media (×) businessmen commerce	school (×) students (×) children (×) education (×) schools (×)	market (×) percent (×) companies (×) billion (×) investors (×)	rolls roll sashimi fish (×) tempura	also (×) really (×) well (×) good (×) try (×)	definitely (×) prices (×) strip (×) selection (×) value (×)	one (×) would (×) like (×) could (×) us (×)
KeyETM	team (×) game (×) players (×) games (×) play (×)	percent (×) japan (×) year (×) japanese (×) economy	city (×) state (×) york (×) school (×) program (×)	people (×) year (×) china (×) years (×) time (×)	sashimi rolls roll fish (×) japanese	food (×) great (×) place (×) good (×) service (×)	great delicious amazing excellent tasty	food (×) place (×) service (×) time (×) restaurant (×)
CatE	public health health care medical hospitals doctors	diversifying (×) clients (×) corporate investment banking executives	french corsica spain (×) belgium (×) de (×)	alberta british columbia ontario manitoba canadian	freshest fish (×) sashimi nigiri ayce sushi rolls	delicacies (×) sundaes savoury (×) pastries custards	tasty delicious yummy chilaquiles (×) also (×)	unforgivable frustrating horrible irritating rude
SeedTopicMine	medical hospitals hospital public health patients	companies businesses corporations firms corporate	french paris philippe (×) french state frenchman	canadian quebec montreal toronto ottawa	maki rolls sashimi ayce sushi revolving sushi nigiri	cheesecakes croissants pastries breads (×) cheesecake	great excellent fantastic delicious amazing	terrible horrible awful lousy shitty

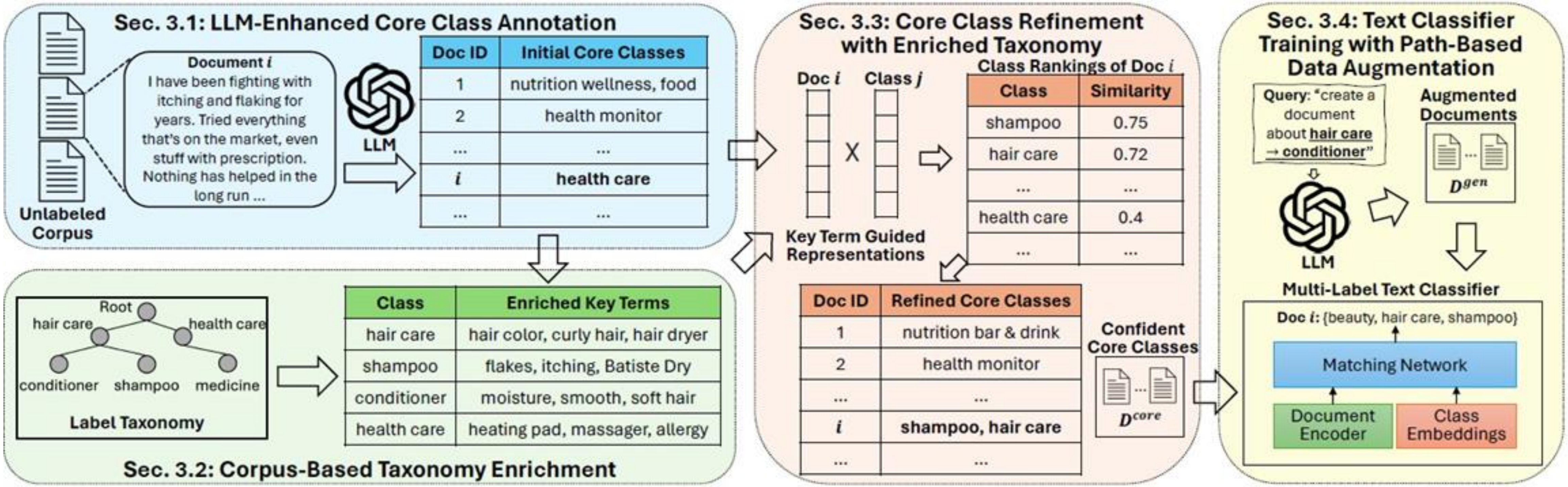
Comparing with CatE on more fine-grained terms

Dataset	Method	Lower-ranked Terms
NYT-Topic	CatE	**sports**: baseball, football, clubs (×), tennis, coaches, amateur (×), n.b.a, handball
	SeedTopicMine	**sports**: coaches, athletics, players, championships, sportsman, olympians, sporting events, tournament
	CatE	**politics**: rhetoric (×), constituencies (×), vitriolic (×), passivity (×), unprincipled (×), polarized (×), philosophically (×), worldview (×)
	SeedTopicMine	**politics**: democratic, parties, conservative coalition, elected, liberal, electoral, leaders (×), political alliance
Yelp-Food	CatE	**desserts**: churros, chocolate, omelettes (×), crepes, truffles (×), fondue (×), sweets, breakfasts (×)
	SeedTopicMine	**desserts**: candied, scones, truffles (×), tarts, crepes, coffees (×), doughnuts, candies
	CatE	**seafood**: oysters, softshell, paella, fishes, octopus, mussel, mackerel, crawfish
	SeedTopicMine	**seafood**: lobster, clam, seafood, crawfish, blue crab, imitation crab, jumbo shrimp, sardines

TELEClass: Taxonomy Enrichment and LLM-Enhanced Hierarchical Text Classification with Minimal Supervision



- Task: *Classifying documents into 10^2 to 10^3 classes, without human annotation*?
- Automatically enrich the label taxonomy with class-indicative topical terms mined from the corpus to facilitate classifier training
- Use LLMs for both data annotation and creation tailored for the hierarchical label space



TELEClass: Performance Study and Cost for Text Classification

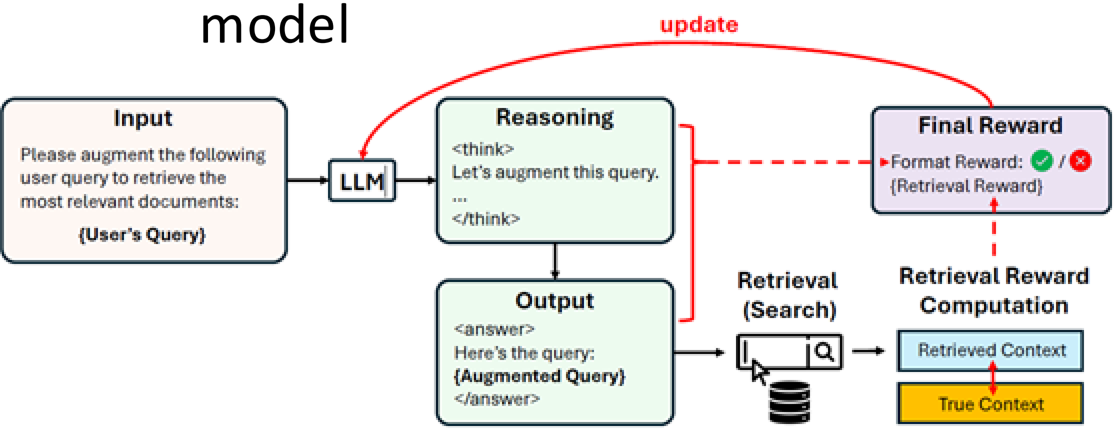
Supervision Type	Methods	Amazon-531				DBPedia-298			
		Example-F1	P@1	P@3	MRR	Example-F1	P@1	P@3	MRR
Zero-Shot	Hier-0Shot-TC†	0.4742	0.7144	0.4610	–	0.6765	0.7871	0.6765	–
	GPT-3.5-turbo	0.5164	0.6807	0.4752	–	0.4816	0.5328	0.4547	–
Weakly-Supervised	Hier-doc2vec†	0.3157	0.5805	0.3115	–	0.1443	0.2635	0.1443	–
	WeSHClass†	0.2458	0.5773	0.2517	–	0.3047	0.5359	0.3048	–
	TaxoClass-NoST†	0.5431	0.7918	0.5414	0.5911	0.7712	0.8621	0.7712	0.8221
	TaxoClass†	0.5934	0.8120	0.5894	0.6332	0.8156	0.8942	0.8156	0.8762
	TELEClass	**0.6330**	**0.8439**	**0.6269**	**0.6664**	**0.8684**	**0.9293**	**0.8684**	**0.8880**
Fully-Supervised		0.8843	0.9524	0.8758	0.9085	0.9786	0.9945	0.9786	0.9826

Methods	Amazon-531					DBPedia-298				
	Example-F1	P@1	P@3	Est. Cost	Est. Time	Example-F1	P@1	P@3	Est. Cost	Est. Time
GPT-3.5-turbo	0.5164	0.6807	0.4752	$60	240 mins	0.4816	0.5328	0.4547	$80	400 mins
GPT-3.5-turbo (level)	0.6621	0.8574	0.6444	$20	800 mins	0.6649	0.8301	0.6488	$60	1,000 mins
GPT-4‡	0.6994	0.8220	0.6890	$800	400 mins	0.6054	0.6520	0.5920	$2,500	1,000 mins
TELEClass	0.6330	0.8439	0.6269	<$1	3 mins	0.8684	0.9293	0.8684	<$1	7 mins

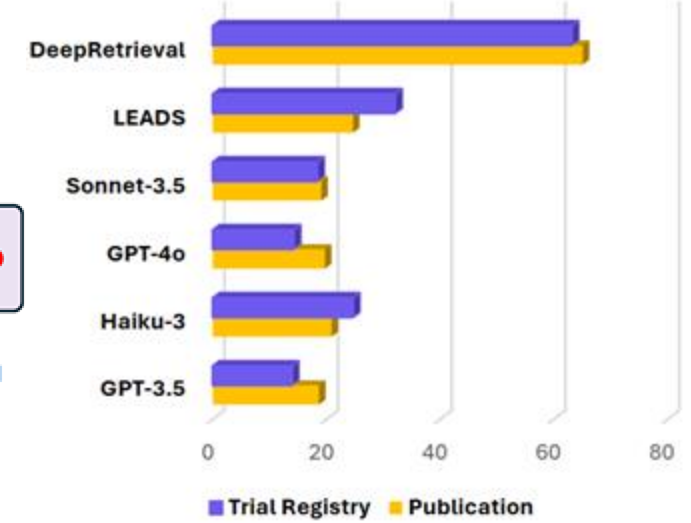
Yunyi Zhang, et al., "TELEClass: Taxonomy Enrichment and LLM-Enhanced Hierarchical Text Classification with Minimal Supervision", WWW'25

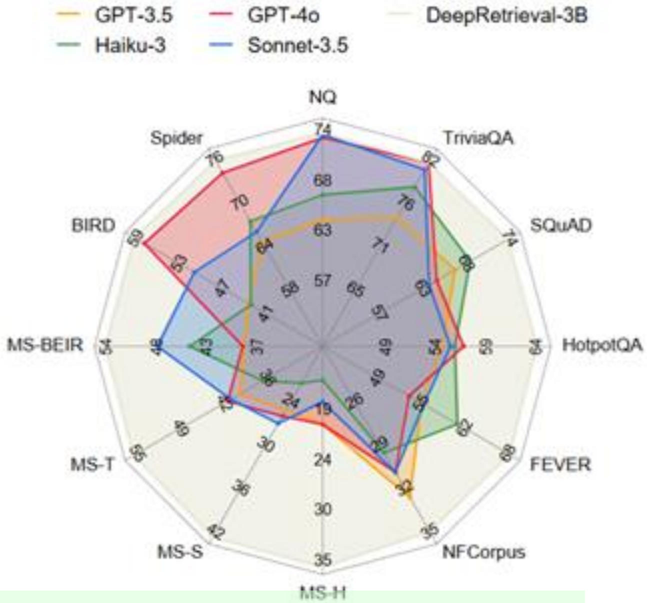
DeepRetrieval w. LLMS via RL

- DeepRetrieval
  - Based on an input user query, the LLM generates an augmented query, which is used to retrieve documents.
  - Format reward and retrieval reward are both computed as the feedback to update the model



	Literature Search		Evidence-Seeking Retrieval								
	Publication	CinicalTrials	NQ			TriviaQA			SQuAD		
	Recall	Recall	H@1	H@5	H@20	H@1	H@5	H@20	H@1	H@5	H@20
Original Query	10.36	18.01	21.9	43.8	63.0	48.2	66.3	76.4	**36.5**	**57.4**	**71.1**
GPT-3.5	11.67	9.42	24.3	46.0	63.9	45.8	64.3	74.2	31.6	52.4	66.6
w/o reasoning	18.68	13.94	25.2	47.5	66.3	47.5	66.8	76.7	33.9	54.9	69.5
GPT-4o	17.59	16.25	**35.8**	57.5	72.2	**59.6**	**73.3**	**80.5**	30.4	49.9	64.4
w/o reasoning	19.72	14.26	29.1	56.2	69.3	53.4	70.1	78.7	33.0	52.2	66.7
Claude-3-Haiku	11.26	10.10	26.2	48.6	66.4	48.8	67.9	77.7	33.3	54.1	68.4
w/o reasoning	20.92	24.68	25.0	48.1	65.5	49.0	67.7	77.3	33.2	54.3	68.8
Claude-3.5-Sonnet	20.94	18.33	35.7	**57.1**	**72.5**	57.1	71.7	79.7	28.5	48.1	63.5
w/o reasoning	19.08	18.41	37.2	56.9	72.7	60.8	73.8	80.6	30.3	49.8	64.7
Mistral$_{7B\text{-}Inst}$	7.18	8.08	26.9	48.8	66.0	50.0	66.7	75.9	27.7	46.6	61.6
LEADS$_{7B}$ (SFT)	**24.68**	**32.11**	-	-	-	-	-	-	-	-	-
Qwen2.5$_{3B\text{-}Inst}$	6.59	6.09	25.0	45.8	63.4	44.4	61.2	70.9	28.4	46.4	61.3
w/o reasoning	9.46	7.97	23.8	45.3	64.0	46.0	64.4	74.2	32.3	52.8	66.8
DeepRetrieval$_{3B}$	65.07	63.18	**35.5**	57.5	72.7	**58.4**	**73.2**	80.6	38.5	59.4	72.9
w/o reasoning	**51.90**	**53.31**	26.9	48.8	66.9	52.0	69.4	77.7	**37.8**	**58.0**	**72.5**





P. Jiang , et al., "DeepRetrieval: Hacking Real Search Engines and Retrievers with Large Language Models via Reinforcement Learning," ArXiv2503

*Our **DeepRetrieval** outperforms GPT-4o, etc. for search on PubMed & ClinicalTrial.gov, and other tasks with a wide margin.*

Theme-focused Information Distillation by Multi-granularity Text Analysis

- A document may still contain many less relevant passages
  - Document-level analysis may dilute the theme-focused analysis and lead to too much data to effectively augment an LLM
- Cross-passage analysis
  - Individual passages, looking irrelevant, when interlinked, may become important
  - Co-reference analysis
  - Multi-hop dense retrieval and multi-passage BERT outperform local normalization techniques
  - Combining graph-, classification-, and set search-based methods

Outline

- Why a Retrieval and Graph Structuring Approach for LLM Applications?
- Taxonomy-Guided, Semantics-Based Retrieval
- Knowledge Graph Structuring for Intelligent Retrieval and Augmentation
- Retrieval and Structure-Augmented Generation for LLM Applications

OntoType: Ontology-Guided Entity Typing

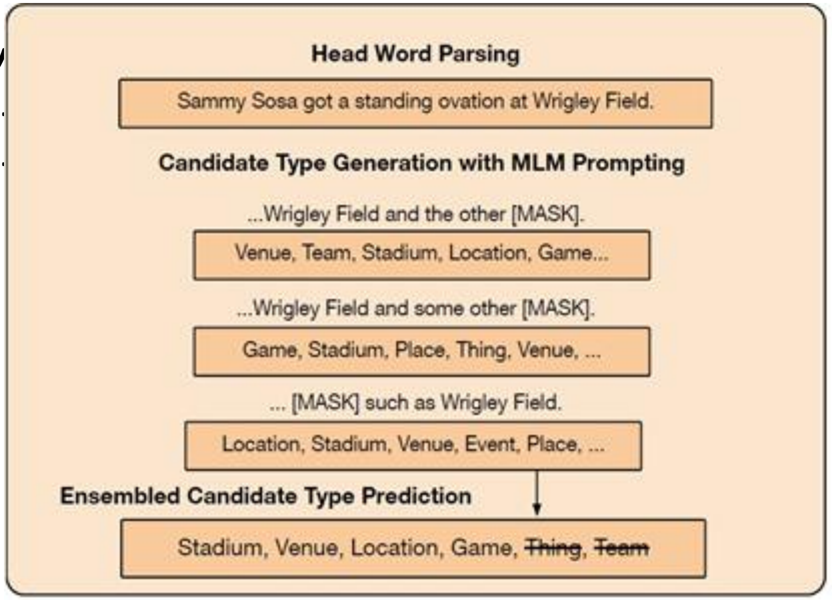
- Fine-grained entity typing (FET): Assigns entities in text with context-sensitive, fine-grained semantic types
  - Ex. *Sammy Sosa* [Person/Player] got a standing ovation at *Wrigley Field* [Location/Building/Stadium]
- Challenges of weak supervision based on masked language model (MLM) prompting
  - A prompt generates a set of tokens, some likely vague or inaccurate, leading to erroneous typing
  - Not incorporate the rich structural information in a given, fine-grained type ontology
- OntoType: Ontology-guided, Annotation-Free, Fine-Grained Entity Typing
  - Ensemble multiple MLM prompting results to generate a set of type candidates
  - Progressively refine type resolution, from coarse to fine, following the type

Tanay, Komarlu, et al., "ONTOTYPE: Ontology-Guided and Pre-Trained Language Model Assisted Fine-Grained Entity Typing", KDD 2024

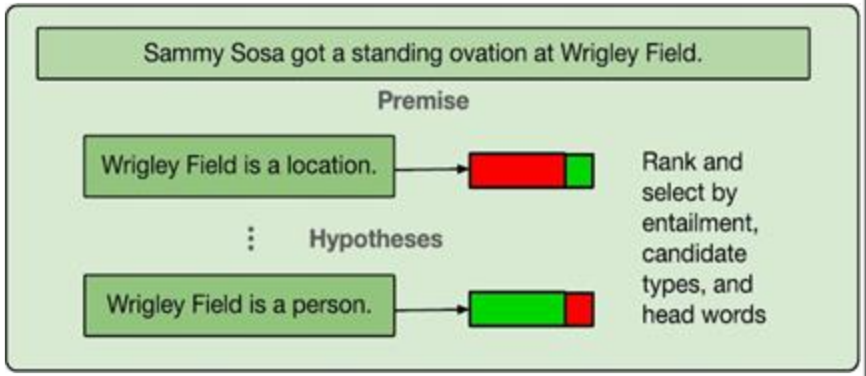
OntoType: Ontology-Guided Entity Typing

- Ex. *Sammy Sosa* [Person/Player] got a standing ovation at *Wrigley Field* [Location/Building/Stadium]
- Candidate type generation
  - Multiple MLM prompting + ensembled candidate type prediction
  - Ex. Stadium, venue, location, games, ~~things, teams~~
- High-level type alignment by entailment (local context + NLI)
- Progressively refine type resolution, from coarse to fine, following the type ontology

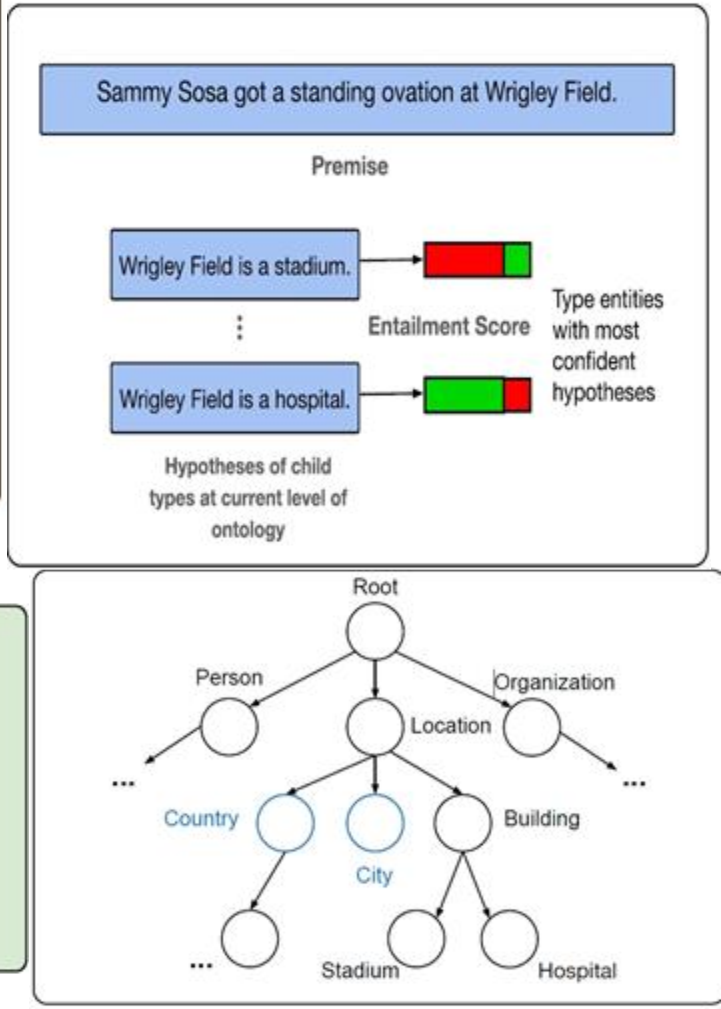
Step 1: Candidate type generation



Step 2: high-level type alignment



Step 3: Top-down, progressive refinement, with type ontology



Zero-Shot Entity Typing Leads to High Performance

- Use 3 benchmark FET datasets: NYT, Ontonotes, and FIGER

Datasets	Ontonotes	FIGER	NYT
# of Types	89	113	125
# of Documents	300k	3.1M	295k
# of Entity Mentions	242K	2.7M	1.18M
# of Train Mentions	223K	2.69M	701K
# of Test Mentions	8,963	563	1,010

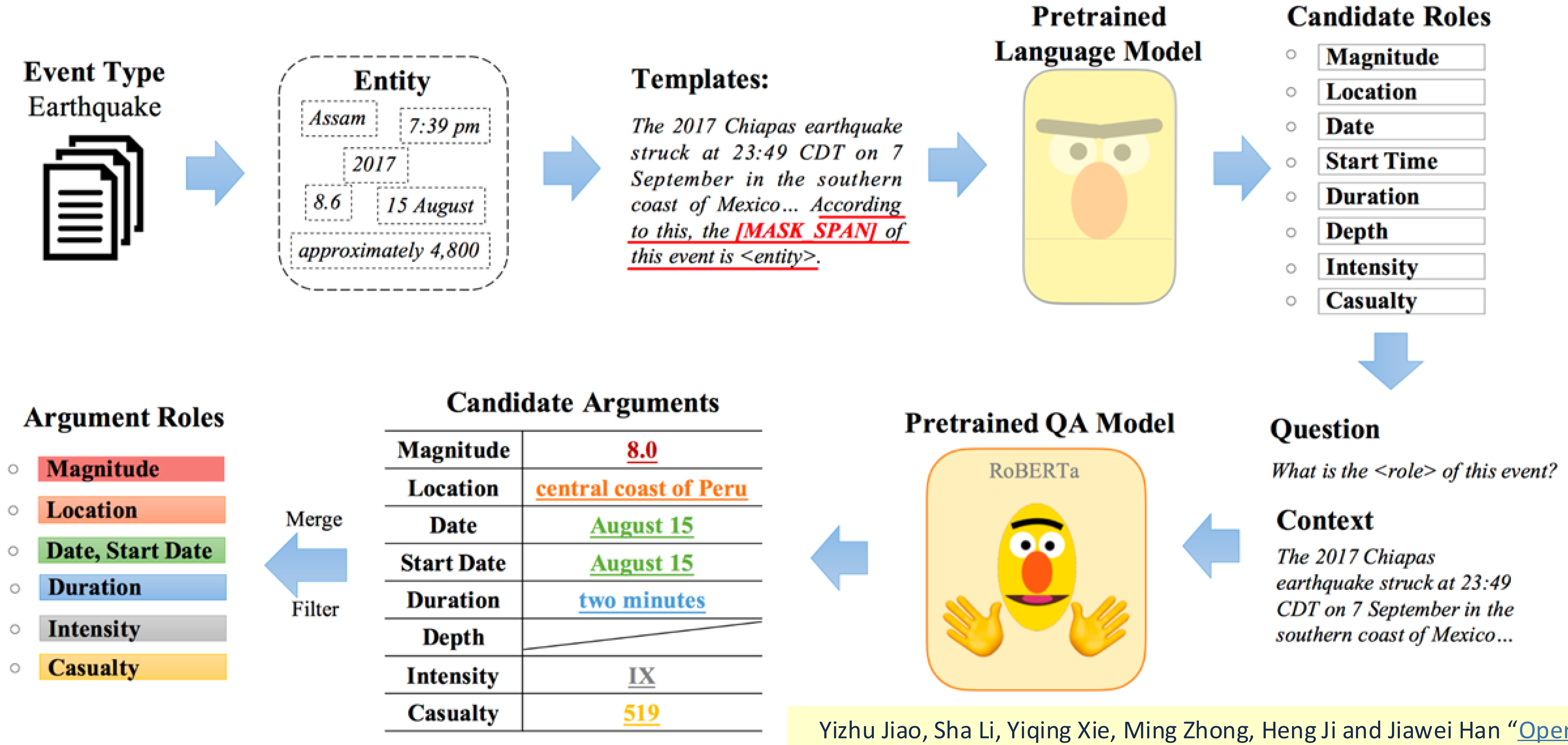
Case Study: MZET vs. ZOE vs. OntoType

MZET	**US President Joe Biden** \Person\Politician was one of many foreign leaders to speak with President Zelensky, and he "pledged to continue providing **Ukraine** \Location with the support needed to defend itself, including advanced air defence systems", **the White House** \Location\Building said.
ZOE	**US President Joe Biden** \Person\Politician was one of many foreign leaders to speak with President Zelensky, and he "pledged to continue providing **Ukraine** \Location\Country with the support needed to defend itself, including advanced air defence systems", the **White House** \Location\Building said.
ONTOTYPE	**US President Joe Biden** \Person\Politician\President was one of many foreign leaders to speak with President Zelensky, and he "pledged to continue providing **Ukraine** \Location\Country with the support needed to defend itself, including advanced air defence systems", **the White House** \Organization\Government said.

- Compare with supervised and 0-shot methods:

Settings	Model	NYT Acc	NYT Mi-F1	NYT Ma-F1	FIGER Acc	FIGER Mi-F1	FIGER Ma-F1	Ontonotes Acc	Ontonotes Mi-F1	Ontonotes Ma-F1
Weak Supervision with Human Annotations	UFET [3]	-	-	-	-	-	-	59.5	71.8	76.8
	BERT-MLMET [4]	-	-	-	-	-	-	67.44	80.35	85.44
	LITE [13]	-	-	-	66.2	74.7	80.1	68.2	81.4	86.6
	NFETC-SSL [32]	-	-	-	71.2	80.2	81.9	64.4	74.3	79.7
Distant Supervision via KBs	DZET [20]	27.3	53.1	51.6	28.5	56.0	55.1	23.1	28.1	27.6
	ZOE [39]	62.1	73.7	76.9	**58.8**	**71.3**	74.8	50.7	60.8	66.9
Transfer Learning	OTyper [35]	46.4	65.7	67.3	47.2	67.2	69.1	31.8	36.0	39.1
	MZET [36]	30.7	58.2	56.7	31.9	57.9	55.5	33.7	43.7	42.3
Annotation-Free	ChatGPT[22]	47.3	59.1	54.3	51.7	65.3	58.3	27.7	37.5	32.6
	ONTOTYPE + Original Ontology	**69.6**	**78.4**	**82.8**	49.1	67.4	**75.1**	**65.7**	**73.4**	**81.5**
	ONTOTYPE + Modified Ontology	-	-	-	51.1	68.9	**77.2**	-	-	-

RolePred: Argument Role Prediction [EMNLP'22]



Yizhu Jiao, Sha Li, Yiqing Xie, Ming Zhong, Heng Ji and Jiawei Han "Open-Vocabulary Argument Role Prediction for Event Extraction", EMNLP'22

RolePred: Candidate Role Generation

- Predict candidate role names for named entities by casting it as a prompt-based in-filling task
  - Prompt Construction: (using Generation Model : T5)
    - *Context*. According to this, the ⟨MASK SPAN⟩ of this Event Type is Entity.
  - Ex. *The 1964 Alaskan earthquake, also known as the Great Alaskan earthquake, occurred at 5:36 PM AKST on Good Friday, March 27.* According to this, the ⟨MASK SPAN⟩ of this earthquake is 5:36 PM.
    - ⟨MASK SPAN⟩ is expected to be filled with *time* (or *start time*) as the argument role
  - Considering the entity's general semantic type: person, location, number, etc., we slightly alter the prompt to fluently and naturally support the unmasking argument roles

Prompt design for different entities

Entity Type	Prompt
PERSON	*According to this, Entity play the role of ⟨MASK SPAN⟩ in this Event Type.*
LOCATION	*According to this, the ⟨MASK SPAN⟩ is Entity in this Event Type.*
NUMBER	*According to this, the number of ⟨MASK SPAN⟩ of this Event Type is Entity.*
OTHER TYPES	*According to this, the ⟨MASK SPAN⟩ of this Event Type is Entity.*

RolePred: Candidate Argument Extraction

- Formulate the argument extraction problem into question-answering task
  - Input: follow a standard BERT-style format (Model: BERT based pretrained QA model)
    - [CLS] What is the Event Role in this Event Type event? [SEP] Document [SEP]
  - Ex. [CLS] What is the *casualty* in this *pandemic* event? [SEP] *The COVID-19 pandemic is an ongoing global pandemic of coronavirus disease. It's estimated that the worldwide total number of deaths has exceeded five million …* [SEP]
    - The argument is expected to be five million

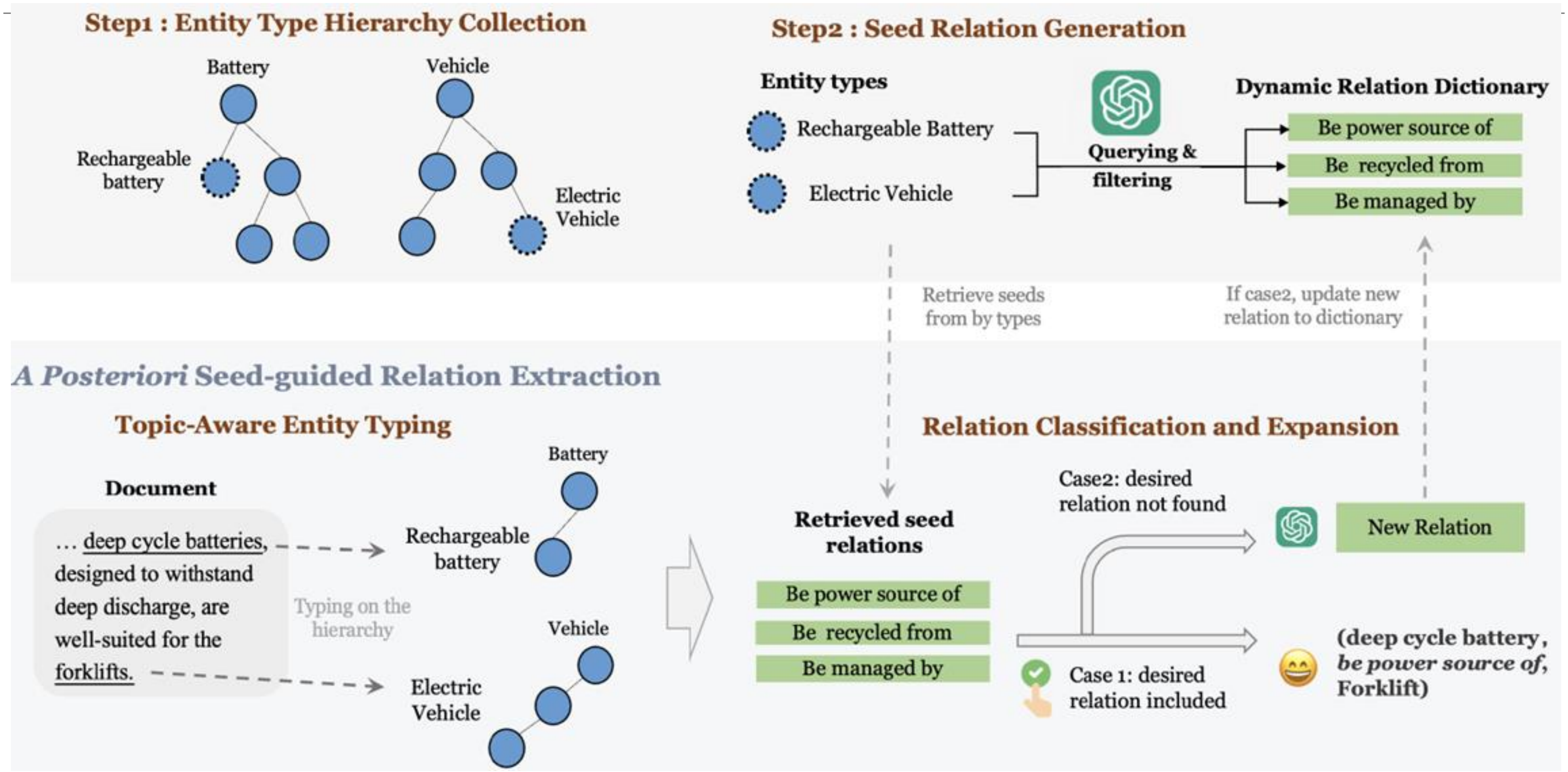
Datasets	# EvTyp.	# RoleTyp.	# Doc.	# ArgScat.
ACE2005	33	35	599	1
KBP2016	18	20	169	1
KBP2017	18	20	167	1
MUC-4	4	5	1,700	4.0
WikiEvents	50	59	246	2.2
RAMS	139	65	3,993	4.8
RoleEE	50	143	4,132	7.1

Dataset statistics

Models	Hard Matching			Soft Matching		
	Precision	Recall	F1	Precision	Recall	F1
LiberalEE	0.1342	0.2613	0.1773	0.3474	0.5340	0.4209
VASE	0.0926	0.1436	0.1125	0.2581	0.4274	0.3218
ODEE	0.1241	0.3076	0.1768	0.3204	0.4862	0.3862
CLEVE	0.1363	0.2716	0.1815	0.3599	0.5712	0.4415
ROLEPRED (BERT)	0.2128	0.4582	0.2906	0.4188	0.6896	0.5211
ROLEPRED (T5)	**0.2552**	**0.6461**	**0.3659**	**0.4591**	**0.7079**	**0.5570**
- RoleMerge	0.2233	0.6962	0.3381	0.4234	0.7677	0.5457
- RoleMerge - RoleFilter	0.1928	0.6582	0.2983	0.4188	0.7084	0.5264
Human	0.6098	0.8270	0.7020	0.7365	0.8732	0.7990

Argument Role Prediction

Open Relation Extraction for Automated Theme KG Construction



Linyi Ding, Jinfeng Xiao, Sizhe Zhou, Chaoqi Yang, Jiawei Han, “Topic-Oriented Open Relation Extraction with A Priori Seed Generation”, in EMNLP’24

Automated Theme-Specific Knowledge Graph Construction

Table 1: Statistics of Datasets

Datasets	Documents	Entities	Relations	Triples
EVB	20	130	64	330
HAI	20	142	77	425

ThemeKG: Application in Question-Answering

Question	Which countries support Hamas or condemn Israel in the Hamas attack on Israel in Oct 2023?
Vanilla GPT4	I'm sorry, but as of my knowledge cutoff date in march 2023, i do not have information on specific events that occurred in october 2023.
RAG+GPT4	In the **Middle East** and **North Africa**, most countries either condemned Israel or offered full-throated support to Hamas. **North Korea** is also mentioned as condemning Israel.
TKG+GPT4	1. **Iran**, 2. **Persian Gulf countries**, 3. **North Korea**, 4. most **Middle East countries**, 5. most **North Africa countries**

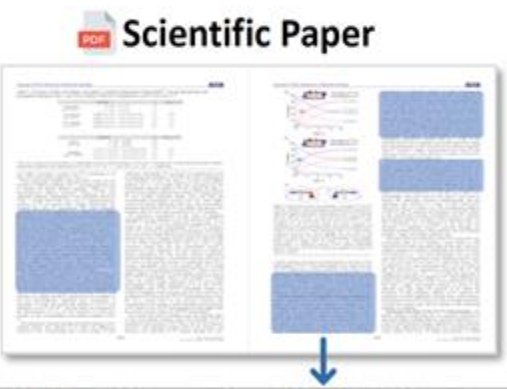
Table 2: Comparison with baselines on KG construction.

Dataset	Method	Entity Metric			Triple Metric			Theme Metric
		Recall	Precision	F1-score	Recall	Precision	F1-score	Coherence
EVB	OpenIE [37]	0.62	0.36	0.46	0.13	0.24	0.17	0.46
	REBEL [24]	0.22	**0.80**	0.35	0.11	**0.80**	0.19	0.80
	IMoJIE [31]	0.44	0.49	0.46	0.26	0.45	0.33	0.78
	KG-GPT [47]	0.72	0.69	0.70	0.67	0.64	0.65	0.95
	GPT-4 [1]	0.68	0.71	0.69	0.64	0.65	0.64	**0.97**
	TKGCon (w/o ontology)	/	/	/	0.67	0.57	0.62	0.92
	TKGCon	**0.92**	**0.80**	**0.86**	**0.78**	0.73	**0.75**	**0.97**
HAI	OpenIE [37]	0.52	0.28	0.36	0.17	0.22	0.19	0.35
	REBEL [24]	0.16	0.87	0.27	0.15	0.75	0.25	0.75
	IMoJIE [31]	0.33	0.39	0.36	0.25	0.31	0.28	0.83
	KG-GPT [47]	0.84	0.79	0.81	0.72	0.69	0.70	0.91
	GPT-4 [1]	0.82	0.80	0.83	0.70	0.72	0.71	**0.93**
	TKGCon (w/o ontology)	/	/	/	0.75	0.62	0.68	0.88
	TKGCon	**0.90**	**0.88**	**0.89**	**0.81**	**0.75**	**0.78**	0.92

- Construction of theme-specific knowledge graphs **automatically**
- **Method:** Convert **open-vocabulary relation extraction** to **relation classification**
- **We are exploring the application of Theme KG in chemistry, material science and geographic science**

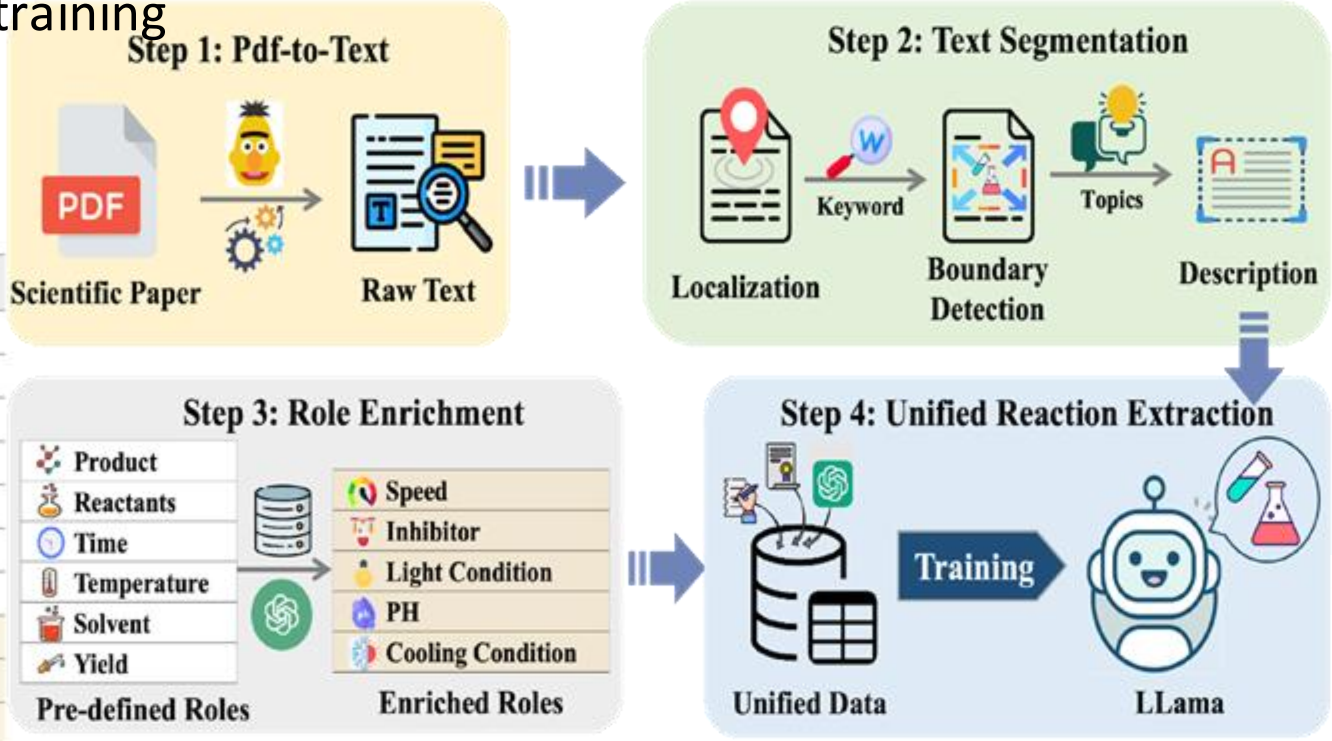
Reaction Miner: Chemical Reaction Info Extraction from Text Data

- Automatic extraction of chemical reaction information (e.g., reactants, catalyst, temperature, etc.)
- Role enrichment: Use RolePredict to extract new reaction roles and generate synthetic data corresponding to each role based on GPT-4 for training

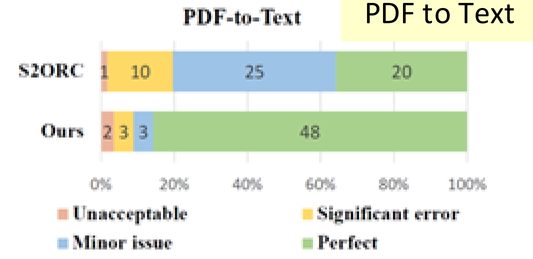


Chemical Reaction

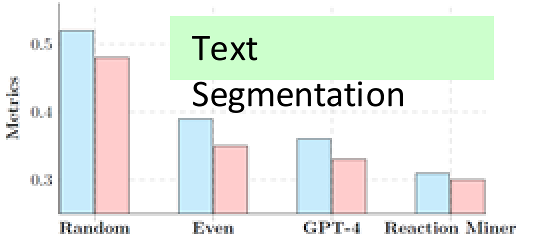
Product	2-chlorophenol
Reactants	phenol, oxalyl chloride
Time	3 hours
Temperature	10 °C
Solvent	NaOH solution
Yield	78% (2-chlorophenol)
Workup Reagent	6 M HCl
Speed	500 rpm
Inhibitor	BHT
Light Condition	dark
PH	13.2
Cooling Condition	ice bath

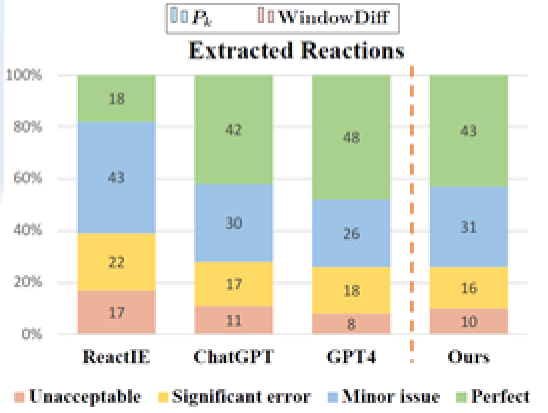


PDF to Text



Text Segmentation

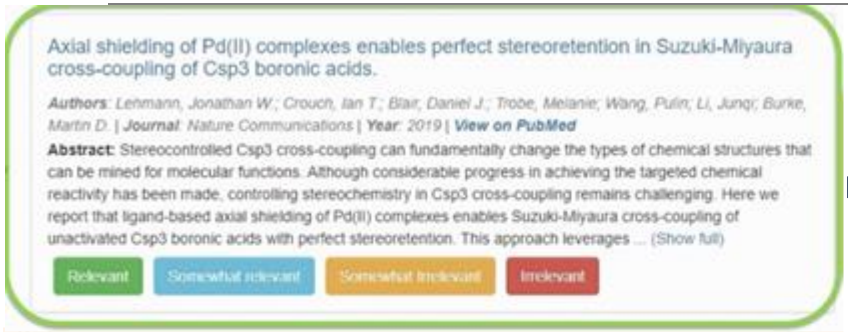


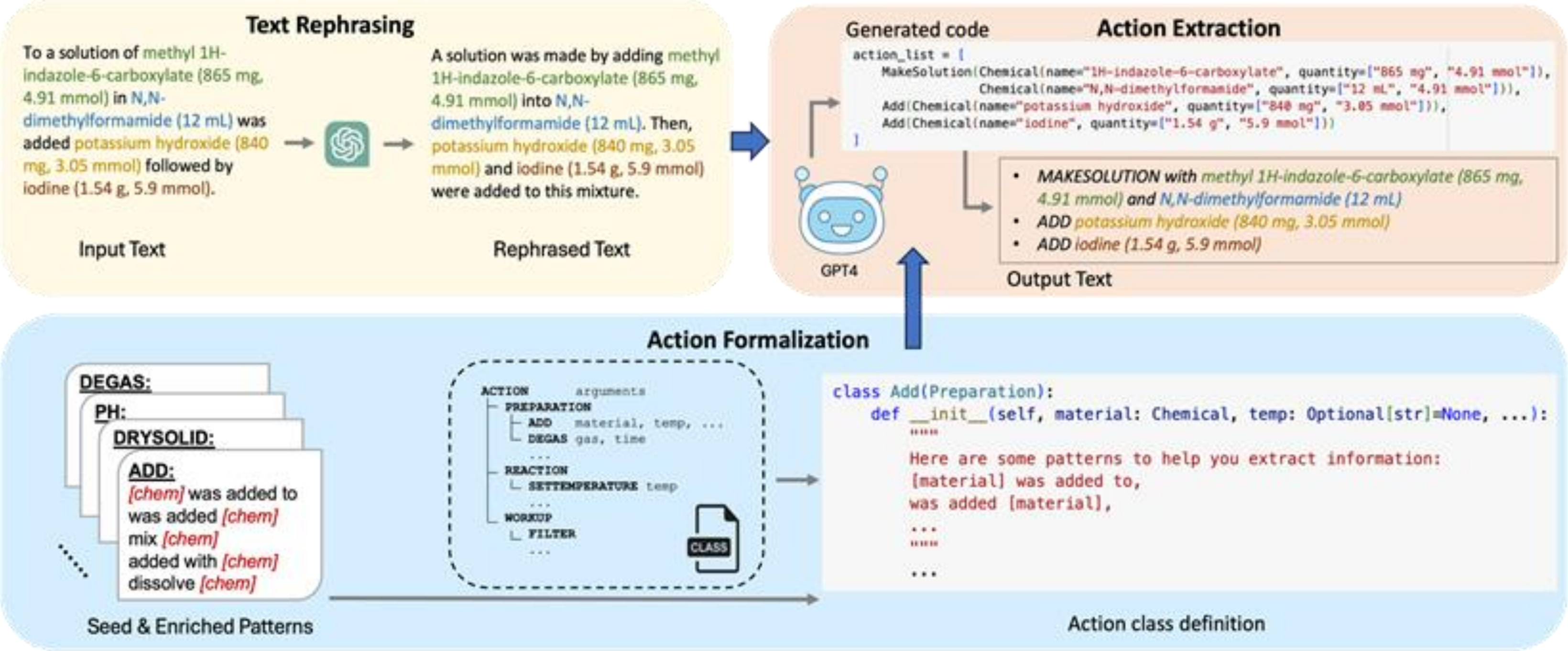


Performance study: ReactionMiner outperforms GPT4 on extraction quality

Ming Zhong, Siru Ouyang, Yizhu Jiao, Priyanka Kargupta, Leo Luo, et al., "Reaction Miner: An Integrated System for Chemical Reaction Extraction from Textual Data" EMNLP'2023

ActionIE: Action Extraction from Scientific Literature with Programming Languages





1. ADD 2,3-dichloropyridine
2. ADD K2CO3
3. ADD DMSO
4. MAKESOLUTION with DMSO and benzyl mercaptan
5. ADD SLN dropwise at 100 °C over 600 s
6. STIR for 3600 s at 100 °C
7. ADD water
8. EXTRACT with dichloromethane
9. COLLECTLAYER organic
10. DRYSOLUTION over sodium sulfate
11. CONCENTRATE
12. PURIFY
13. YIELD 2-benzylthio-3-chloropyridine

Workflow

1. Text Rephrasing: Rephrasing the paragraphs for better information retrieval.
2. Pattern Enrichment: Mining undiscovered and significant action patterns automatically.
3. Action Extraction: Automatic extraction of chemical reaction actions with code generation.

Models	BLEU	Levenshtein Similarity	Precision	Recall	F1	Graph Matching Similarity
			Results for PatentAction *(Avg Length: 158.24)*			
Supervised Methods						
Paragraph2Actions	**0.8511**	0.8927	0.9017	0.9034	0.8985	0.8003
ChemTrans	-	-	0.5927	0.4325	0.4866	-
Few-shot Methods (10-shot)						
Galactica-6.7b	0.0051	0.1336	0.3526	0.2705	0.2732	0.2921
GPT-4	0.4280	0.6822	0.7537	0.7758	0.7458	0.7923
ACTIONIE	0.8237	**0.9018**	**0.9126**	**0.9198**	**0.9101**	**0.8136**
- Patterns	0.6829	0.8070	0.8458	0.8220	0.8218	0.8074

Xianrui Zhong, Yufeng Du, Siru Ouyang, Ming Zhong, et al., "ActionIE: Action Extraction from Scientific Literature with Programming Languages", ACL'2024

Outline

- Why a Retrieval and Graph Structuring Approach for LLM Applications?
- Taxonomy-Guided, Semantics-Based Retrieval
- Knowledge Graph Structuring for Intelligent Retrieval and Augmentation
- Retrieval and Structure-Augmented Generation for LLM Applications

StructRAG: Motivation and Methodology

- How to better leverage LLMs to transform scattered information into various structure formats
  - hybrid information structuring mechanism: different tasks require different knowledge structure representations for more precise reasoning

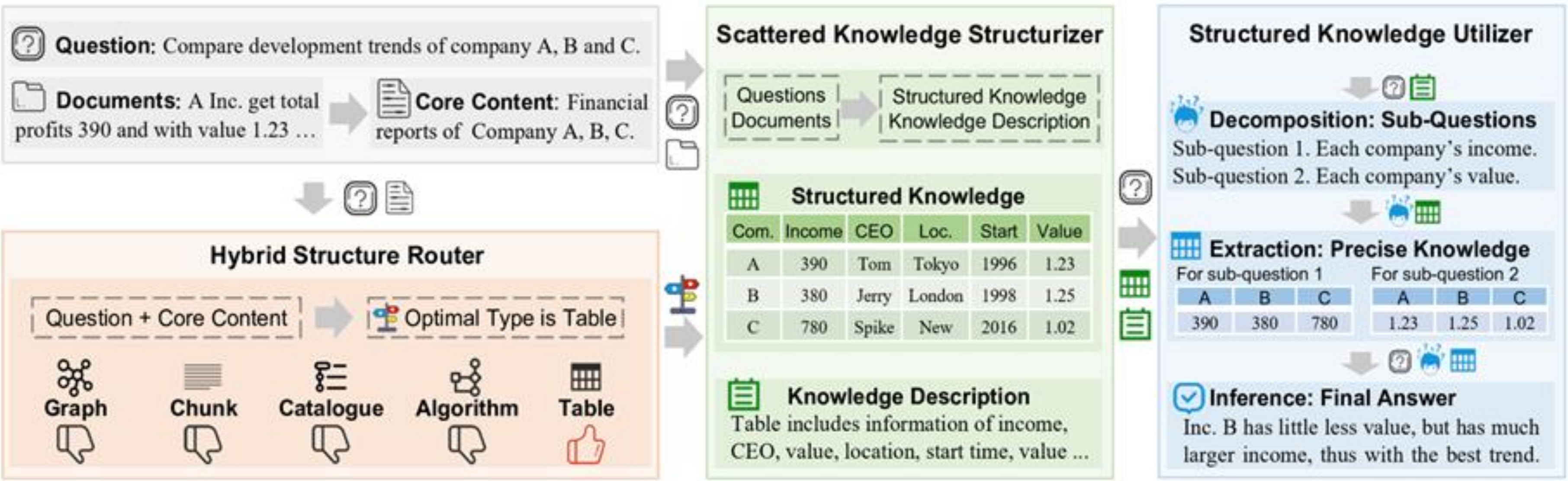


Figure 1: The overview of StructRAG framework, including an hybrid structure router to select the optimal structure type based on task requirements, a scattered knowledge structurizer to convert raw documents into structured knowledge, and a structured knowledge utilizer to decompose complex question and then effectively using the structured knowledge to infer the final answer.

- **Hybrid Structure Router:** select the most optimal structure type from five candidate structure types.
- **Scattered Knowledge Structurizer:** extracts the textual knowledge scattered across raw documents for reconstruction.
- **Structured Knowledge Utilizer:** LLM-based knowledge utilizer to facilitate question decomposition, precise knowledge extraction, and final answer inference.

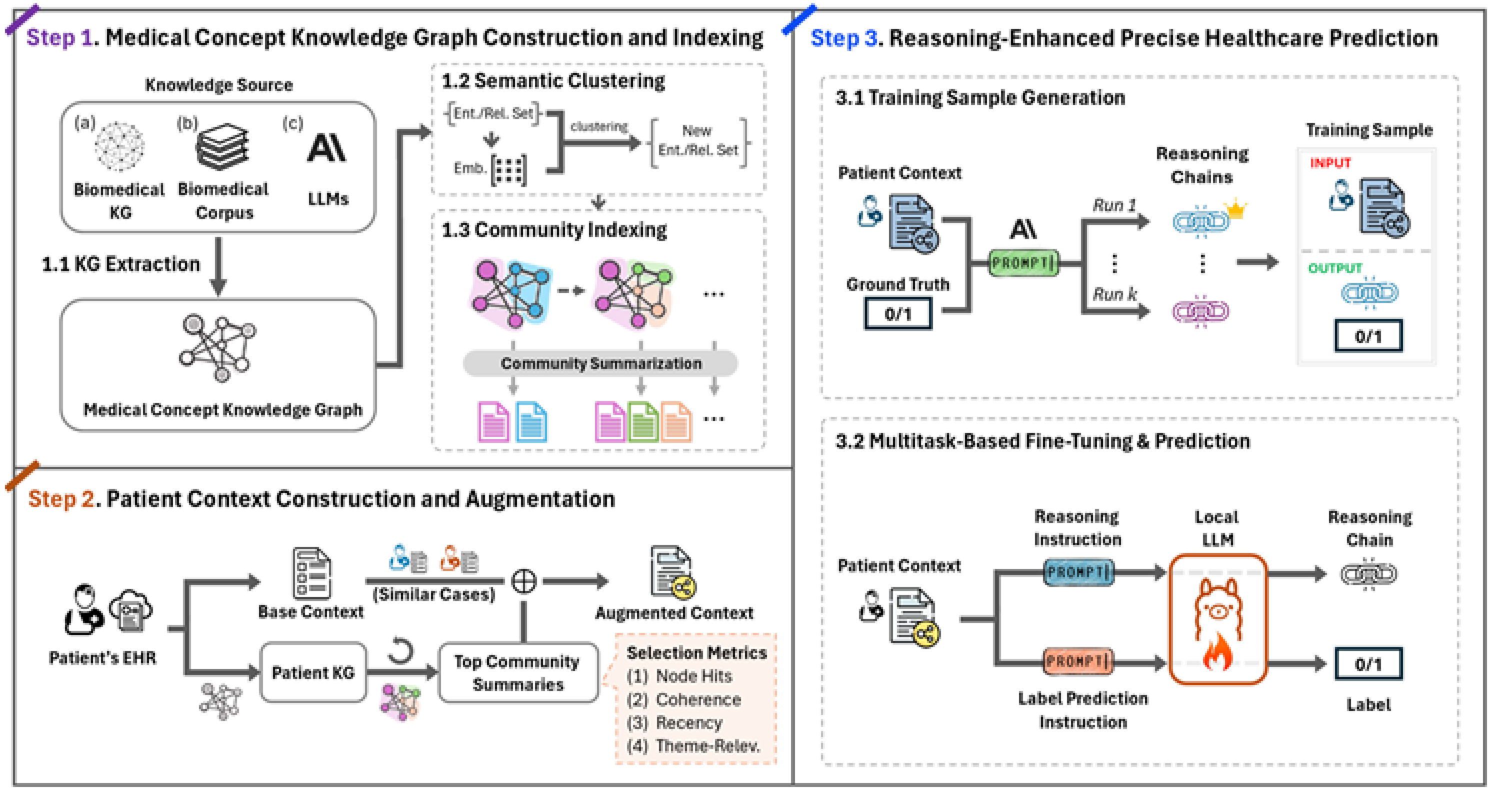
Li et al., "StructRAG: Boosting Knowledge Intensive Reasoning of LLMs via Inference-time Hybrid Information", ICLR 2025.

StructRAG: Experiments and Analyses

Method	Spot.		Comp.		Clus.		Chain.		Overall	
	LLM Score	EM	LLM Score	EM	LLM Score	EM	LLM Score	EM	LLM Score	EM
Set 1 (10K-50K Tokens)										
Long-context (Yang et al., 2024a)	68.49	**0.55**	60.60	0.37	47.08	0.08	**70.39**	**0.36**	60.11	0.29
RAG (Lewis et al., 2020)	51.08	0.35	44.53	0.27	37.96	0.05	53.95	0.35	46.11	0.23
RQ-RAG (Chan et al., 2024)	72.31	0.54	48.16	0.05	47.44	0.07	58.96	0.25	53.51	0.17
GraphRAG (Edge et al., 2024)	31.67	0.00	27.60	0.00	40.71	0.14	54.29	0.43	40.82	0.18
StructRAG (Ours)	**74.53**	0.47	**75.58**	**0.47**	**65.13**	**0.23**	67.84	0.34	**69.43**	**0.35**
Set 2 (50K-100K Tokens)										
Long-context (Yang et al., 2024a)	64.53	0.43	42.60	0.21	38.52	0.05	51.18	0.20	45.71	0.17
RAG (Lewis et al., 2020)	66.27	**0.46**	46.28	0.31	38.95	0.05	46.15	**0.22**	45.42	0.19
RQ-RAG (Chan et al., 2024)	57.35	0.35	50.83	0.16	42.85	0.03	47.60	0.10	47.09	0.10
GraphRAG (Edge et al., 2024)	24.80	0.00	14.29	0.00	37.86	0.00	46.25	0.12	33.06	0.03
StructRAG (Ours)	**68.00**	0.41	**63.71**	**0.36**	**61.40**	**0.17**	**54.70**	0.19	**60.95**	**0.24**
Set 3 (100K-200K Tokens)										
Long-context (Yang et al., 2024a)	46.99	0.27	37.06	0.13	31.50	0.02	35.01	0.07	35.94	0.09
RAG (Lewis et al., 2020)	**73.69**	**0.55**	42.20	0.27	32.78	0.02	37.65	0.13	42.60	0.18
RQ-RAG (Chan et al., 2024)	50.50	0.13	44.62	0.00	36.98	0.00	36.79	0.07	40.93	0.05
GraphRAG (Edge et al., 2024)	15.83	0.00	27.40	0.00	42.50	0.00	43.33	0.17	33.28	0.04
StructRAG (Ours)	68.62	0.44	**57.74**	**0.35**	**58.27**	**0.10**	**49.73**	**0.13**	**57.92**	**0.21**
Set 4 (200K-250K Tokens)										
Long-context (Yang et al., 2024a)	33.18	0.16	26.59	0.08	29.84	**0.01**	25.81	0.04	28.92	0.06
RAG (Lewis et al., 2020)	52.17	**0.24**	24.60	0.10	26.78	0.00	17.79	0.00	29.29	0.07
RQ-RAG (Chan et al., 2024)	29.17	0.08	40.36	0.00	26.92	0.00	34.69	0.00	31.91	0.01
GraphRAG (Edge et al., 2024)	17.50	0.00	26.67	0.00	20.91	0.00	33.67	0.33	23.47	0.05
StructRAG (Ours)	**56.87**	0.19	**55.62**	**0.25**	**56.59**	0.00	**35.71**	**0.05**	**51.42**	**0.10**

KARE: Need Construction of Theme-Specific KGs

- LLMs may produce hallucinations or incorrect information due to a lack of specialized medical knowledge in healthcare domain, due to
  - Retrieve seemingly related but not insightful information
  - Leverage knowledge graph with graph community retrieval is largely unexplored



Algorithm 1 Dynamic Graph Retrieval and Augmentation

Input: Set of communities $\mathcal{C}$, patient graph G_p, base context $\mathcal{B}_p$, desired number of summaries N
Output: Augmented patient context $\mathcal{A}_p$
Initialize $S_p \leftarrow \emptyset$
Initialize hit counts $H(v) \leftarrow 0$ for each node $v \in V_p^{\text{direct}}$
while $|S_p| < N$ **do**
  Compute Relevance(C_k) for all $C_k \in \mathcal{C}$ using Eq. 3
  Select $C_{\text{best}} \leftarrow \arg\max_{C_k \in \mathcal{C}} \text{Relevance}(C_k)$
  Add $S_{C_{\text{best}}}$ to S_p: $S_p \leftarrow S_p \cup S_{C_{\text{best}}}$
  For each $v \in V_{C_{\text{best}}} \cap V_p^{\text{direct}}$, $H(v) \leftarrow H(v) + 1$
  Remove C_{best} from $\mathcal{C}$: $\mathcal{C} \leftarrow \mathcal{C} \setminus C_{\text{best}}$
end
Augment patient context: $\mathcal{A}_p = \mathcal{B}_p \oplus S_p$
return $\mathcal{A}_p$

Jiang et al., "Reasoning-Enhanced Healthcare Predictions with Knowledge Graph Community Retrieval", ICLR 2025

Experiment Setting: Task, Data and Metrics

- Tasks: EHR-based prediction
  - Mortality Prediction: Estimates mortality outcome for next visit (Patient's survival status during visit x_t)
  - Readmission Prediction: Predicts if patient will be readmitted within σ days (σ is set to 15 in this study)

Table 1: Statistics of pre-processed EHR datasets. "#": "the number of", "/ patient": "per patient".

	MIMIC-III-Mort.			**MIMIC-III-Read.**			**MIMIC-IV-Mort.**			**MIMIC-IV-Read.**		
	Train	Valid	Test	Train	Valid	Test	Train	Valid	Test	Train	Valid	Test
# Patients (Samples)	7730	991	996	7730	991	996	8018	996	986	8029	958	1013
# Visits / Patient	1.56	1.60	1.61	1.56	1.60	1.61	1.26	1.30	1.21	1.26	1.28	1.25
# Conditions / Patient	23.27	23.92	25.89	23.27	23.92	25.89	14.34	15.30	13.59	13.62	14.21	13.21
# Procedures / Patient	6.22	6.56	7.17	6.22	6.56	7.17	2.96	3.08	2.84	2.89	2.96	2.81
# Medications / Patient	54.79	55.77	63.73	54.79	55.77	63.73	30.66	32.86	28.40	28.74	30.61	27.59

- Datasets: Use the publicly available MIMIC-III (v1.4) and MIMIC-IV (v2.0) EHR datasets
  - Use PyHealth (Yang et al., 2023a) for preprocessing, …
- Evaluation Metrics: Four key metrics:
  - Accuracy: Overall correct predictions across both outcomes
  - Macro-F1: A balanced measure, crucial for the imbalanced datasets
  - Sensitivity: Model's ability to identify patients at risk of mortality or readmission
  - Specificity: Identify patients unlikely to experience these outcomes, helping avoid unnecessary

Performance Comparison on MIMIC-III Dataset

Results are averaged by multiple runs. asterisk (∗): important for handling imbalanced datasets.

		MIMIC-III							
		Mortality Prediction (pos = 5.42%)				Readmission Prediction (pos = 54.82%)			
Type	Models	Accuracy	Macro F1*	Sensitivity*	Specificity	Accuracy	Macro F1	Sensitivity	Specificity
ML	GRU (Chung et al., 2014)	92.7	50.7	3.7	97.8	62.2	61.5	68.9	54.0
	Transformer (Vaswani et al., 2017)	92.7	51.9	5.6	97.6	58.8	58.2	65.0	51.3
	RETAIN (Choi et al., 2016)	92.4	50.6	3.7	97.6	59.1	56.9	74.9	40.0
	GRAM (Choi et al., 2017)	92.4	50.2	5.2	95.2	61.8	60.4	74.9	46.4
	Deepr (Nguyen et al., 2016)	91.9	51.0	3.7	98.2	62.6	62.1	66.7	57.6
	TCN (Bai et al., 2018)	91.6	53.2	9.3	96.4	63.4	62.7	70.7	54.7
	ConCare (Ma et al., 2020b)	94.6	48.6	0.0	**100.0**	59.2	59.0	61.5	56.4
	AdaCare (Ma et al., 2020a)	90.6	54.1	9.1	97.6	61.6	60.5	70.8	50.3
	GRASP (Zhang et al., 2021)	93.7	49.9	1.9	98.9	61.3	59.5	74.9	44.8
	StageNet (Gao et al., 2020)	90.5	50.5	5.6	95.4	60.5	60.0	65.1	54.9
	KerPrint (Yang et al., 2023b)	92.4	52.2	9.8	94.7	63.5	62.1	68.0	56.1
LM+ML	GraphCare (Jiang et al., 2024a)	94.9	58.3	17.2	97.1	65.4	64.1	70.3	57.8
	RAM-EHR (Xu et al., 2024)	94.4	59.6	14.8	98.9	64.8	63.5	74.7	52.4
	EMERGE (Zhu et al., 2024a)	94.1	57.7	13.2	98.4	63.7	62.0	68.0	55.9
LLM	Zero-shot (LLM: Claude 3.5 Sonnet)								
	w/ EHR context only	89.5	50.4	6.4	94.4	54.3	35.4	98.9	0.2
	w/ Classic RAG[a]	89.9	51.2	10.2	92.8	53.2	34.6	91.2	1.4
	w/ KARE-augmented context[b]	92.3	54.6	14.2	94.6	56.3	43.8	93.9	10.6
	Few-Shot (LLM: Claude 3.5 Sonnet)								
	w/ exemplar only (N=2)[c]	88.7	49.5	5.6	93.4	52.7	42.2	87.0	11.1
	w/ exemplar only (N=4)	88.0	49.2	5.6	92.7	53.6	44.7	84.0	15.7
	w/ EHR-CoAgent[d] (Cui et al., 2024)	87.4	51.7	13.0	91.8	55.2	46.1	78.2	20.1
	w/ KARE-augmented context	91.5	53.5	13.7	94.0	57.1	49.3	75.5	27.2
	KARE (ours)	**95.3**	**64.6**	**24.7**	98.3	**73.9**	**73.7**	**76.7**	**70.7**

RepoGraph: Background and Motivation

- Real-world software engineering often extends beyond single function or self-contained code files:
  - navigating complex structured code bases
  - understanding intricate dependencies between code file
  - ensuring that changes integrate seamlessly without introducing new issues

A perfect testbed for RAS in engineering domain!

Ouyang et al., "RepoGraph: Enhancing AI Software Engineering with Repository-level Coding Graph", ICLR 2025

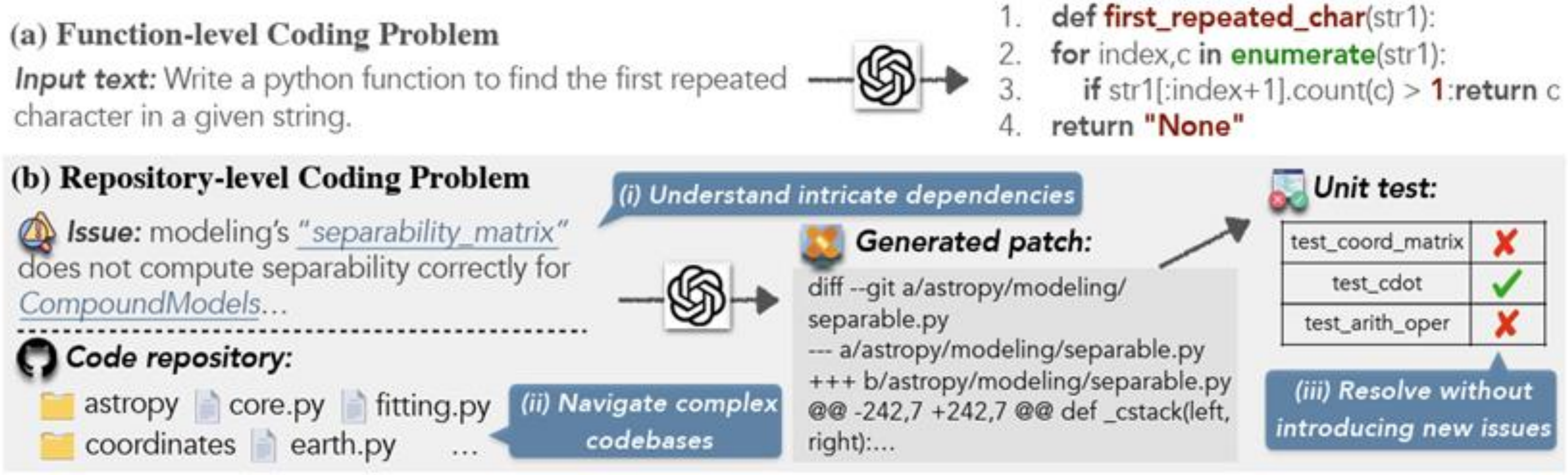


Figure 1: The illustration of *(a) a function-level coding problem* from HumanEval (Chen et al., 2021) and *(b) a repository-level coding problem* from SWE-Bench (Jimenez et al., 2024).

Table 1: Comparison between our approach REPOGRAPH and existing methods for representing the repository on various aspects. ***RepoUnderstander (Ma et al., 2024) and CodexGraph (Liu et al., 2024) are concurrent works to ours.**

Model	Line-level	File-level	Repo-level
DraCo	✗	✓	✗
Aider	✓	✗	✗
RepoUnderstander*	✗	✓	✓
CodexGraph*	✗	✓	✓
REPOGRAPH	✓	✓	✓

RepoGraph: Methodology

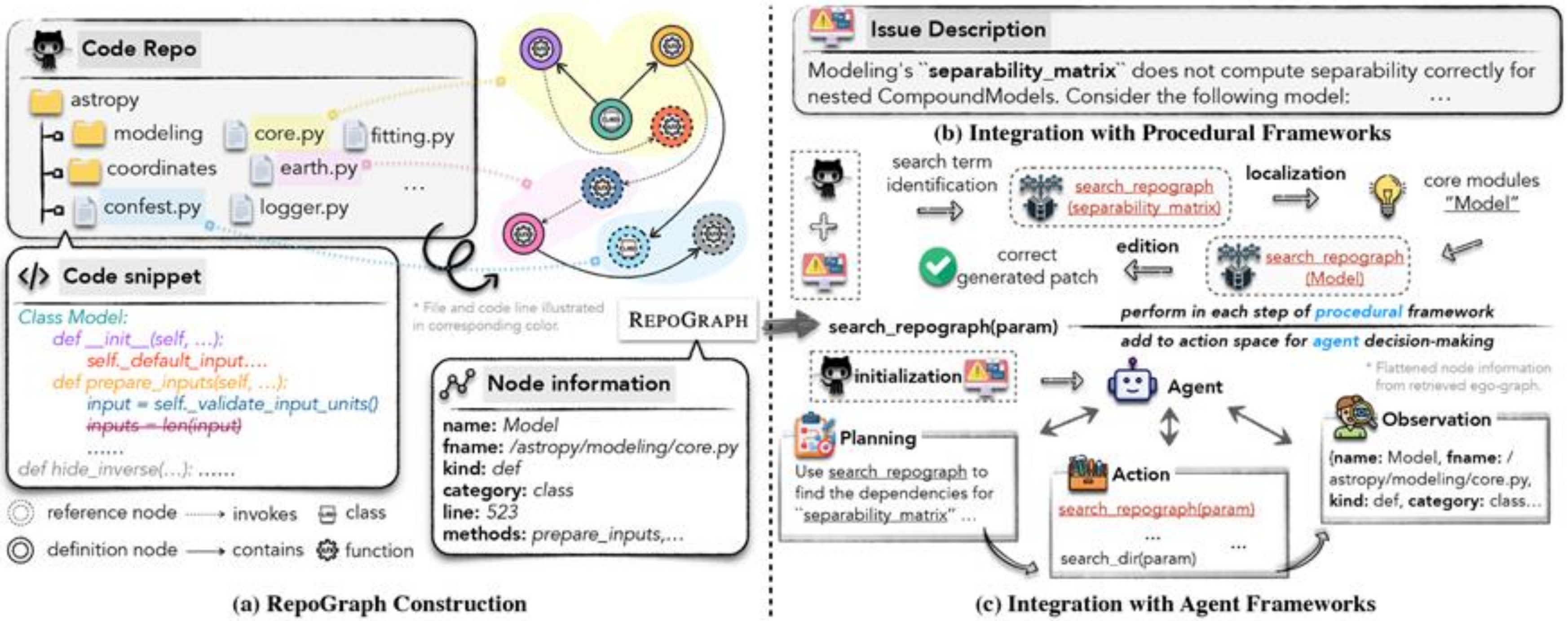


Figure 2: An in-depth illustration of *(a) the construction*, *(b) the integration with procedural frameworks*, and *(c) the integration with agent frameworks* of REPOGRAPH. Given a code repository, we first utilize AST to construct $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$, where $\mathcal{G}$ consists of "*reference*" and "*definition*" node, $\mathcal{E}$ includes "*invoke*" and "*contain*" relations (files and code lines shown in corresponding color). The constructed REPOGRAPH are then used in procedural frameworks by adding sub-retrieval results into each step, and agent frameworks by adding graph retrieval as an additional action "search_repograph". A simplified version can be found in Figure 10.

- Graph construction comprises of three steps: [step 1] - code line parsing using static analysis tools; [step 2] - project-dependent relation filtering; and [step 3] - graph organization
- Utility includes integration with procedural and agent frameworks, making RepoGraph versatile

RepoGraph: Experiments and Analyses

Table 2: Results of REPOGRAPH with open-source baselines in two research lines, including procedural and agent frameworks. Numbers of accuracy-related metrics are directly taken from the leaderboard, while the cost-related ones are computed from the corresponding trajectories[5].

Methods	LLM	Accuracy: resolve	Accuracy: # samples	Accuracy: patch apply	Avg. Cost: $ cost	Avg. Cost: # tokens
Procedural frameworks						
RAG	GPT-4	2.67	8	29.33	$0.13	11,736
+REPOGRAPH	GPT-4	5.33 ↑2.66	16 ↑8	47.67 ↑18.34	$0.17	15,439
Agentless	GPT-4o	27.33	82	97.33	$0.34	42,376
+REPOGRAPH	GPT-4o	29.67 ↑2.34	89 ↑7	98.00 ↑0.67	$0.39	47,323
Agent frameworks						
AutoCodeRover	GPT-4	19.00	57	83.00	$0.45	38,663
+REPOGRAPH	GPT-4	21.33 ↑2.33	64 ↑7	86.67 ↑3.67	$0.58	45,112
SWE-agent	GPT-4o	18.33	55	87.00	$2.53	498,346
+REPOGRAPH	GPT-4o	20.33 ↑2.00	61 ↑6	90.33 ↑3.33	$2.69	518,792

- RepoGraph brings consistent performance gain for all combinations of frameworks and LLM model bases.
- Performance gain brought by RepoGraph is slightly larger on procedural frameworks than agent ones.
- Performance gain brought by RepoGraph does not rely on more costs.

Table 5: Results on the subset of CrossCodeEval with GPT-4o and Deepseek-Coder-V2-Lite-Instruct as the backbone LLMs.

Methods	Code Match EM	Code Match ES	Identifier Match EM	Identifier Match F1
Deepseek-Coder	10.2	57.3	16.6	49.1
+REPOGRAPH	19.7	67.8	29.3	58.9
GPT-4o	10.5	59.6	16.8	47.9
+REPOGRAPH	28.7	68.9	36.0	61.3

- RepoGraph brings significant benefit to open-source LLMs, on traditional coding tasks.

- The context included by RepoGraph is comprehensive.
- Node and edges grow exponentially when k increases. Flattening the graph increases the tokens. Trade-off of token context comprehensiveness and the ability of LLMs to deal with it.

Table 4: The number of nodes, edges, and tokens of REPOGRAPH and its variants. For different retrieval and integration variants, we report the average number on the test set. "summ." refers to the summarized version by LLMs of the retrieved ego-graph.

Metrics	REPOGRAPH	1-hop + flatten	1-hop + summ.	2-hop + flatten	2-hop + summ.
# Nodes	1419.3	11.6	11.6	54.5	54.5
# Edges	26392.1	37.1	37.1	89.9	89.9
# tokens	-	2310.7	717.5	10505.3	1229.2
resolve rate	-	29.67	28.33	26.00	28.67

Table 3: Percentage of problems for accurate edition localizations with respect to file, function, and line levels. All the numbers are computed from the corresponding generated patches.

Methods	LLM	*file*	*function*	*line*
RAG	GPT-4	47.3	23.3	12.7
+REPOGRAPH	GPT-4	51.7 ↑4.4	25.3 ↑2.0	14.3 ↑1.6
Agentless	GPT-4o	68.7	51.0	34.3
+REPOGRAPH	GPT-4o	74.3 ↑5.6	54.0 ↑3.0	36.7 ↑2.4
Agent frameworks				
AutoCodeRover	GPT-4	62.3	42.3	29.0
+REPOGRAPH	GPT-4	69.0 ↑4.7	45.7 ↑3.4	31.7 ↑2.7
SWE-agent	GPT-4o	61.7	46.3	32.3
+REPOGRAPH	GPT-4o	67.3 ↑5.6	49.3 ↑3.0	35.0 ↑2.7

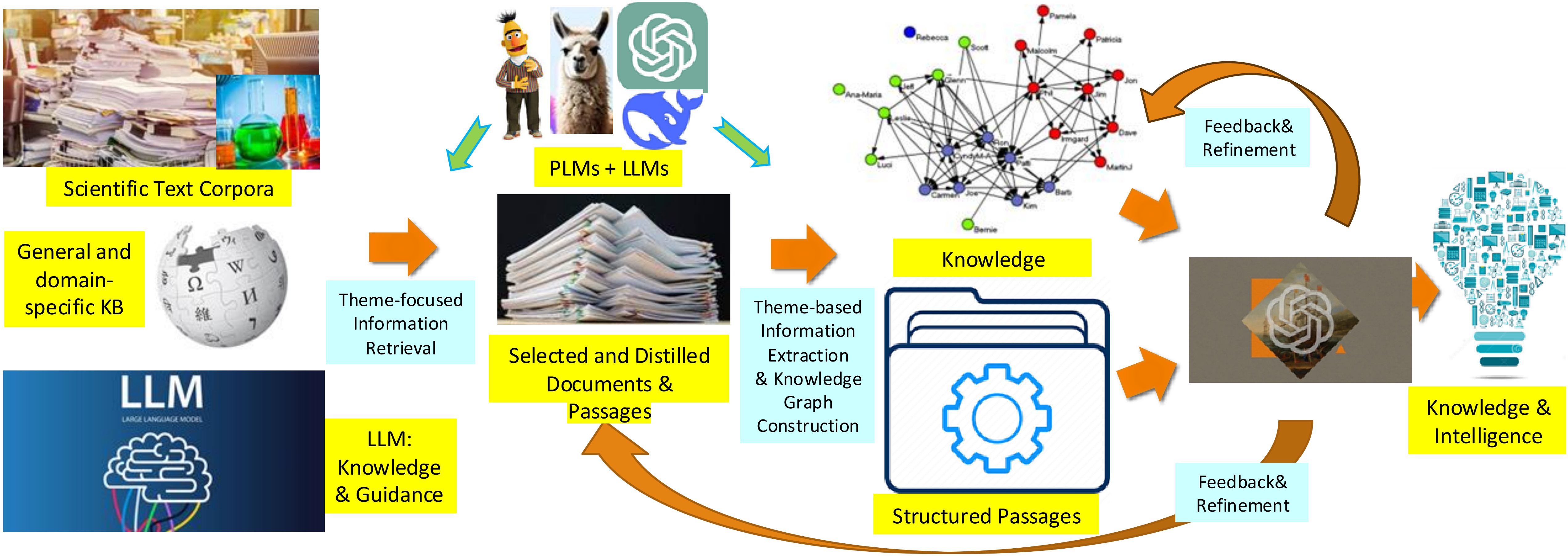
- Recall improves at all granularities; the improvement at finer granularity is relatively smaller.

Outline

- Why a Retrieval and Graph Structuring Approach for LLM Applications?
- Taxonomy-Guided, Semantics-Based Retrieval
- Knowledge Graph Structuring for Intelligent Retrieval and Augmentation
- Retrieval and Structure-Augmented Generation for LLM Applications
- Conclusion

Conclusions

- Theme-specific LLM will enhance the power of LLM for scientific exploration
- RAS (Retrieval and Structuring) will integrate external knowledge and unleash the power of LLM
- Integrating knowledge graphs and powerful LLMs may drive groundbreaking scientific progress



References

- Linyi Ding, Jinfeng Xiao, Sizhe Zhou, Chaoqi Yang, Jiawei Han, “Topic-Oriented Open Relation Extraction with A Priori Seed Generation”, in Proc. 2024 Conf. on Empirical Methods in Natural Language Processing (EMNLP’24), Nov. 2024
- Pengcheng Jiang, Cao Xiao, Minhao Jiang, Parminder Bhatia, Taha Kass-Hout, Jimeng Sun, Jiawei Han, "Reasoning-Enhanced Healthcare Predictions with Knowledge Graph Community Retrieval", ICLR’2025
- Yizhu Jiao, Sha Li, Yiqing Xie, Ming Zhong, Heng Ji and Jiawei Han “Open-Vocabulary Argument Role Prediction for Event Extraction”, EMNLP’22
- Seongku Kang, Shivam Agarwal, Bowen Jin, Dongha Lee, Hwanjo Yu, and Jiawei Han, “Improving Retrieval in Theme-specific Applications using a Corpus Topical Taxonomy”, WWW’24
- SeongKu Kang, Yunyi Zhang, Pengcheng Jiang, Dongha Lee, Jiawei Han, Hwanjo Yu, “Taxonomy-guided Semantic Indexing for Academic Paper Search”, EMNLP’24
- Tanay Komarlu, Minhao Jiang, Xuan Wang, Jiawei Han, "OntoType: Ontology-Guided and Pre-Trained Language Model Assisted Fine-Grained Entity Typing", KDD'24
- Yu Meng, Jiaxin Huang, Guangyuan Wang, Zihan Wang, Chao Zhang, Yu Zhang and Jiawei Han, “Discriminative Topic Mining via Category-Name Guided Text Embedding”, WWW’20
- Yu Meng, Jiaxin Huang, Yu Zhang, Jiawei Han, "Generating Training Data with Language Models: Towards Zero-Shot Language Understanding"", NeurIPS’22
- Siru Ouyang, Jiaxin Huang, Pranav Pillai, Yunyi Zhang, Yu Zhang, and Jiawei Han, "Ontology Enrichment for Effective Fine-grained Entity Typing", KDD’24
- Yu Zhang, Yunyi Zhang, Martin Michalski, Yucheng Jiang, Yu Meng, and Jiawei Han, “Effective Seed-Guided Topic Discovery by Integrating Multiple Types of Contexts”, WSDM’23
- Yunyi Zhang, Ruozhen Yang, Xueqiang Xu, Jinfeng Xiao, Jiaming Shen, Jiawei Han, "TELEClass: Taxonomy Enrichment and LLM-Enhanced Hierarchical Text Classification with Minimal Supervision", WWW’25