Text-Attributed Knowledge Graph Enrichment with Large Language Models for Medical Concept Representation

Mohsen Nayebi Kerdabai, Arya Hadizadeh Moghaddam,
Chen Chen, Dongie Wang, Zijun Yao*

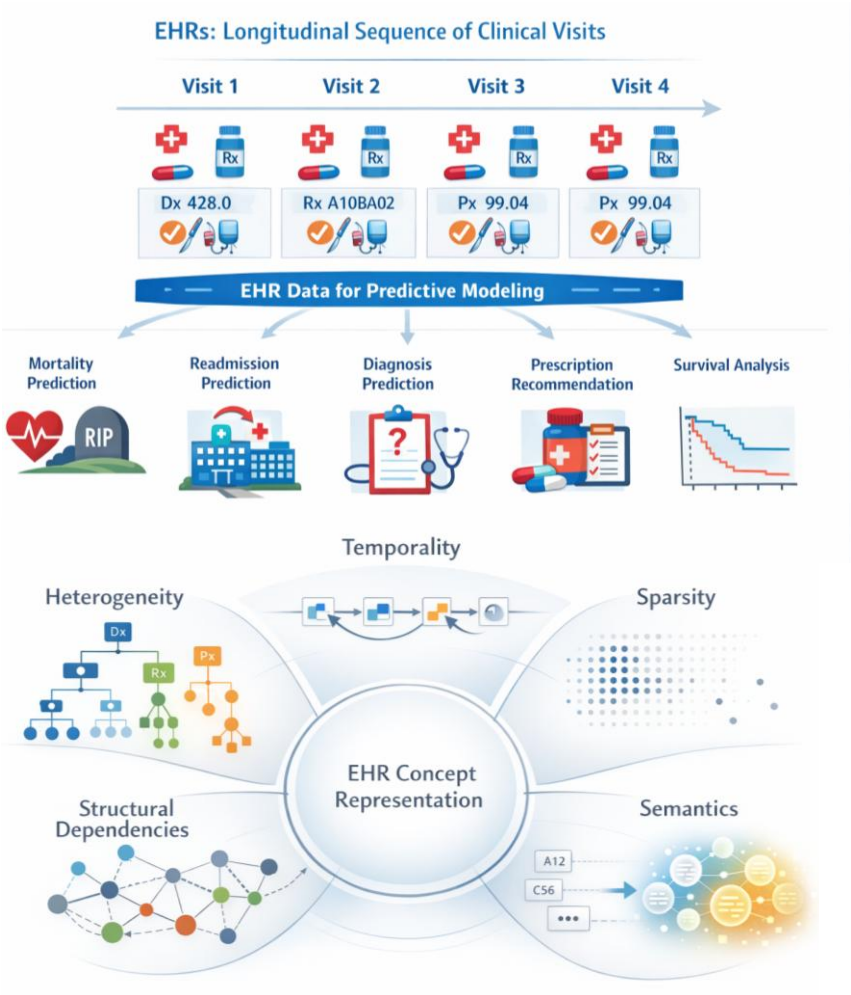
Outline

- **Background and Motivation**
- Proposed Method
- Evaluation
- Conclusion

Electronic Health Records

- **Structured EHRs** represent a patient's medical history as a longitudinal sequence of clinical visits.

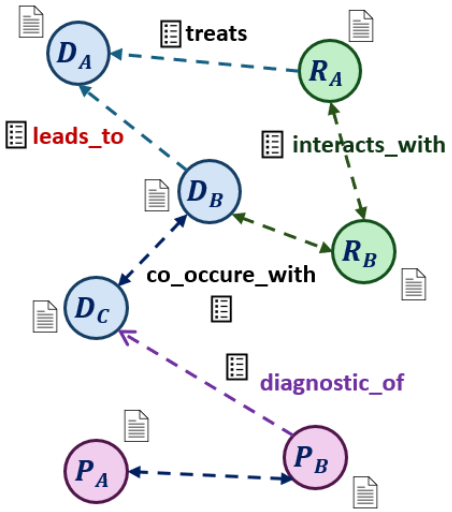
- Each visit contains sets of **standard codes**:
  - **Diagnosis (Dx)** --> ICD-9 428.0 (Congestive heart failure)
  - **Medication (Rx)** --> ATC A10BA02 (Metformin)
  - **Procedure (Px)** --> ICD-9 Proc. 99.04 (Packed cell transfusion)
- EHRs are sparse, high-dimensional, Heterogenous, etc.
- Medical codes are the **Building Blocks** of EHRs
  - Rich code representations are critical for robust clinical predictions.



Motivation

- Rich representation => we need to capture 1) relational dependencies 2) semantic meaning
- Existing ontology resources are limited:
  - **Hierarchical ontologies** mainly capture **within-type parent–child structure**
  - **Relational ontologies** often have **missing or incomplete cross-type links**
- Medical KGs must include **typed semantic connections**:
  - A **dx–dx** edge may mean comorbidity, causality, or progressionA **dx–rx** edge may mean treatment, contraindication
- At the same time, **raw medical codes are only identifiers** and do not carry enough clinical meaning.

We need to construct a **heterogeneous medical knowledge graph** with both **typed relations** and **rich semantics**.



Motivation

- LLMs for KG Construction
  - must be: (1) **evidence-conditioned, (2) type-aware, (3) clinically consistent**
- Raw codes/connections lack semantics
  - A medical code does not fully express: (1) **indication (2) mechanism (3) role in care, (4) clinical context**
  - We need **node**-level and **edge**-level **textual semantics**, not just connectivity.
- Text and graph are hard and inefficient to jointly model
  - **GNNs** are strong at **relational structure**
  - **LLMs** are strong at **rich text semantics**
  - Learning from a **text-attributed graph** requires aligning both effectively in one framework.

Contributions:

- Constructed an **evidence-grounded heterogeneous medical KG** from EHR-derived evidence and **type-constrained LLM prompting.**
- Enriched the KG into a **text-attributed graph** with **node descriptions** and **edge rationales**
- Jointly learned concept embeddings through **LLM–GNN co-learning Strategy**

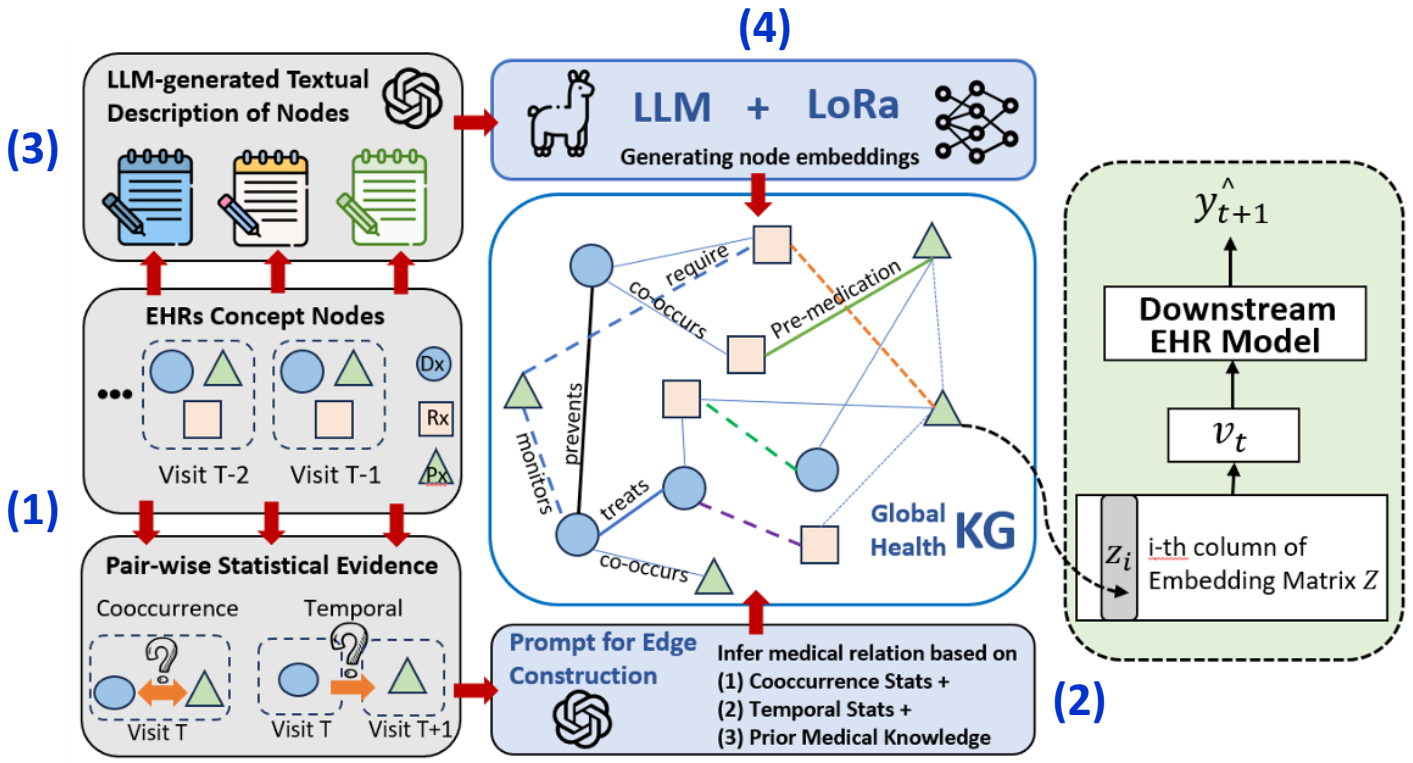
Outline

- Background and Motivation
- **Proposed Method**
- Evaluation
- Conclusion

Proposed Method

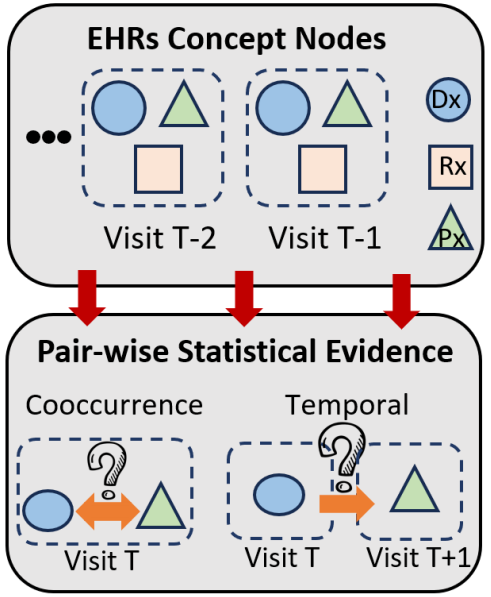
We present **CoMed, a joint KG-LLM Medical Concept Encoder:**

1. **Extract intra-visit co-occurrence and next-visit transition evidence**
2. **Use type-constrained, evidence-based LLM prompting to assign directed relation types**
3. **Enrich nodes with LLM-generated concept descriptions and edges with relation metadata**
4. **Jointly train a LoRA-tuned LLaMA encoder and a relation-aware GNN**



Proposed Method - KG Construction

Step 1: Extract intra-visit co-occurrence and next-visit transition evidence



- Goal: Build evidence-grounded candidate code pairs from EHRs
- Computing two types of empirical signals:
  - **Intra-visit co-occurrence**: codes appearing together within the same visit
  - **Next-visit transition**: codes appearing from Visit t → Visit t+1
- Pair-level statistics: For each directed code pair $(c_i \rightarrow c_j)$,we compute:
  - **Smoothed conditional probability**
  - **Pointwise Mutual Information PMI**
  - **Chi-square p-value**
- Filtering: (1) sufficient support, (2) strong association, (3) statistically significance
- This yields a set of **well-supported, clinically meaningful candidate pairs** for the next step.

Proposed Method - KG Construction

Step 2: Evidence-Supported LLM Prompting for Medical Relation Induction

- Goal: Assign each statistically supported code pair a clinically meaningful semantic relation using an LLM.
- Type-constrained relation pool
- The relation inventory covers semantic categories such as:
  - Causality
  - Temporal progression
  - Therapeutic intent
  - Diagnostic usage
  - Workflow structure
  - Safety / contraindication

LLM Prompt

Task
You are a medical reasoning expert. Infer the most plausible semantic relation between two clinical concepts C1, C2 using medical knowledge first, then statistical signals.

Code Pair (canonical order)
code1: **<C1>=<C1 NAME>**, type= **<C1 TYPE>**, P(C1)=**<P(C1)>**
code2: **< C2>=< C2 NAME >**, type= **<C2 TYPE>**, P(C2)=**<P(C2)>**

Statistical Evidence
Co-occurrence Metrics (same visit)
- 12 (C1→C2): P12 = **<CO_P_12>**, PMI12 = **<CO_PMI_12>**

Temporal Metrics (next visit)
- 12 (C1→C2): tP12 = **<TEMP_P_12>**, tPMI12 = **<TEMP _PMI_12>**

Metrics Glossary
a detailed description of the statistical metrics

Candidate Relationship Labels
The list of candidate relations + description for **<C1 TYPE , C2 TYPE>**

Confidence Calibration
- 90–100: strong clinical + statistical support
- 70–89: strong clinical or statistical support
- 50–69: plausible but limited/conflicting
- <50: weak or contradictory

Decision Rules
- Use clinical knowledge first, then statistics.
- Orient the KG triple correctly:
  - if edge_orientation ∈ {12, symmetric, none}
    → ["**<C1>**", "<predicted_relation>", "**<C2>**"]
  - if edge_orientation = 21
    → ["**<C2>**", "<predicted_relation>", "**<C1>**"]
- Return exactly one relationship decision per pair.
- ...

Output Format (JSON only)
```
{
 "predicted_relationship": "<exactly_ONE_of_AllowedLabels>"
 "KG_triple": ["<HEAD>", "<predicted_relationship>", "<TAIL>"]
 "reasoning": "~50–60 words: clinical rationale first."
 "confidence": <integer 0–100>
}
```

Proposed Method - KG Construction

Step 2: Evidence-Supported LLM Prompting for Medical Relation Induction

- **Prompt Input:**
  - Code IDs and concept names
  - Intra-visit and next-visit statistical evidence
  - Allowed candidate relations for that code-type pair
  - Decision rules and confidence calibration guidance

- **LLM output: a strict structured Json:**
  - One relation label
  - An oriented KG triple
  - A confidence score
  - A short clinical rationale grounded in the evidence

- **Result: Evidence-grounded semantic edge induction for the medical KG**

LLM Prompt

Task
You are a medical reasoning expert. Infer the most plausible semantic relation between two clinical concepts C1, C2 using medical knowledge first, then statistical signals.

Code Pair (canonical order)
code1: **<C1>=<C1 NAME>**, type= **<C1 TYPE>**, P(C1)=**<P(C1)>**
code2: **< C2>=< C2 NAME >**, type= **<C2 TYPE>**, P(C2)=**<P(C2)>**

Statistical Evidence
Co-occurrence Metrics (same visit)
- 12 (C1→C2): P12 = **<CO_P_12>**, PMI12 = **<CO_PMI_12>**

Temporal Metrics (next visit)
- 12 (C1→C2): tP12 = **<TEMP_P_12>**, tPMI12 = **<TEMP _PMI_12>**

Metrics Glossary
a detailed description of the statistical metrics

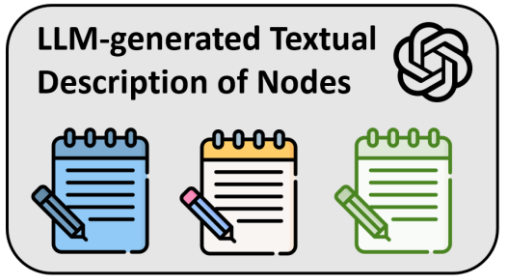
Candidate Relationship Labels
The list of candidate relations + description for **<C1 TYPE , C2 TYPE>**

Confidence Calibration
- 90–100: strong clinical + statistical support
- 70–89: strong clinical or statistical support
- 50–69: plausible but limited/conflicting
- <50: weak or contradictory

Decision Rules
- Use clinical knowledge first, then statistics.
- Orient the KG triple correctly:
  - if edge_orientation ∈ {12, symmetric, none}
    → ["**<C1>**", "<predicted_relation>", "**<C2>**"]
  - if edge_orientation = 21
    → ["**<C2>**", "<predicted_relation>", "**<C1>**"]
- Return exactly one relationship decision per pair.
- ...

Output Format (JSON only)
```
{
 "predicted_relationship": "<exactly_ONE_of_AllowedLabels>"
 "KG_triple": ["<HEAD>", "<predicted_relationship>", "<TAIL>"]
 "reasoning": "~50–60 words: clinical rationale first."
 "confidence": <integer 0–100>
}
```

Proposed Method – KG Construction



Step 3: Enrich nodes with LLM-generated concept descriptions and edges with relation metadata

- **Node-level enrichment:**
  - Use a type-specific LLM prompt to generate a concise clinical description
  - These descriptions are attached to nodes as **text attributes**.
- **Edge-level enrichment:** For each candidate code pair, combine:
  - Induced relation label
  - LLM free-text rationale
  - EHR-derived co-occurrence and transition statistics
- Step 1-3 steps turns statistically supported code pairs into a **text-attributed heterogeneous Knowledge Graph**.

LLM Prompt for Node Description

You are a medical reasoning assistant. Write **one dense paragraph (~250 words) giving most important/accurate clinical context about a medical concept.**

Code
- id: <CODE_ID> - label: <CODE_LABEL> - type: <CODE_TYPE> - P(code):<P(CODE)>

Focus (ideal information to provide, if exists)
{guidance} for type: <CODE_TYPE>

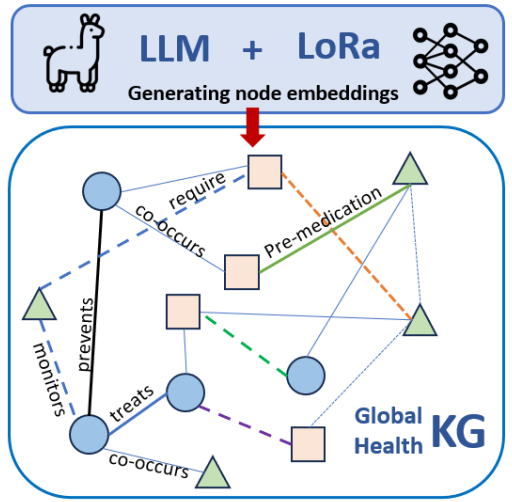
Constraints
- Write one paragraph (~250 words); prefer well-established clinical knowledge.
- Widely accepted, class-level clinical facts appropriate to the label/type.
- Avoid numeric dosing, exact thresholds, or hospital-specific policies.

Forbidden content
- Specific product names, URLs, dosing, numeric lab thresholds
- Speculative claims, local statistics, or unverified niche practices.

Output (JSON only)
{
"code_id": "<code_id>",
"description": "<one dense paragraph ~250 words; no lists/URLs/doses/thresholds>"
}

Proposed Method – CoLearning of LLM and GNN

- Goal: Learn unified medical concept embeddings by combining:
  - **LLM-based semantic knowledge**
  - **KG-based relational structure**
- How it works
  - A **LoRA-tuned LLM text encoder** converts each node's textual description into an semantic embedding
  - A **heterogeneous GNN** then propagates information over the KG.
- Practical training strategy
  - Running the LLM over all medical codes is expensive
  - MedCo uses a coverage-aware sampling update schedule:
    - Phase 1: prioritize least-updated codes
    - Phase 2: mix least-updated and frequent codes
  - This improves coverage while keeping training efficient.



A LLaMA-based text encoder

$$\mathbf{z}_i^{\text{text}} = f_\theta(s_i) \in \mathbb{R}^{d_L},$$
$$\mathbf{h}_i^{(0)} = \mathbf{W}_{\tau(i)}\, \mathbf{z}_i^{\text{text}} \in \mathbb{R}^d.$$

GNN process

$$\begin{aligned}
\mathbf{u}_{ji}^{(\ell)} &= \mathrm{MSG}^{(\ell)}\left(\mathbf{h}_i^{(\ell)}, \mathbf{h}_j^{(\ell)}, \mathbf{e}_{ji}, r_{ji}\right), \\
\mathbf{m}_i^{(\ell)} &= \mathrm{AGG}^{(\ell)}\left(\left\{\mathbf{u}_{ji}^{(\ell)} : j \in \mathcal{N}(i)\right\}\right), \\
\mathbf{h}_i^{(\ell+1)} &= \mathrm{UPDATE}^{(\ell)}\left(\mathbf{h}_i^{(\ell)}, \mathbf{m}_i^{(\ell)}\right).
\end{aligned}$$

Proposed Method – Plug-In into Downstream Task

Integrating MedCo into EHR Models

- MedCo can be used as a plug-in medical concept encoder for any EHR model.
- Outputs dynamic, ontology-informed embeddings tailored for clinical tasks.
- No changes needed to the core architecture of existing EHR models.

Embedding for code i

Raw input feature for medical concept i at level L

$$\mathbf{Z} = [\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_{N^{(L)}}] \leftarrow \text{LINKO}(\mathbf{x}_1^{(L)}, \mathbf{x}_2^{(L)}, \ldots, \mathbf{x}_{N^{(L)}}^{(L)})$$

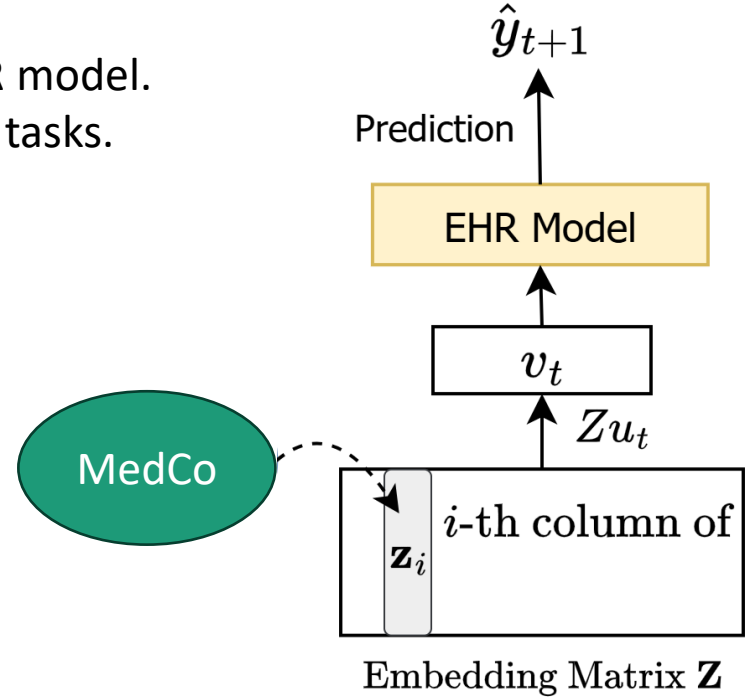
$$\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_t = \mathbf{Z}[\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_t]$$

$$\mathbf{h}_p = \text{Model}(\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_t)$$

$$\hat{\mathbf{y}}_{t+1} = \text{Sigmoid}(\mathbf{W}\mathbf{h}_p + \mathbf{b})$$

Visit representation for time t

Patient representation



Outline

- Background and Motivation
- Proposed Method
- **Evaluation**
- Conclusion

Evaluation - Experimental Setup

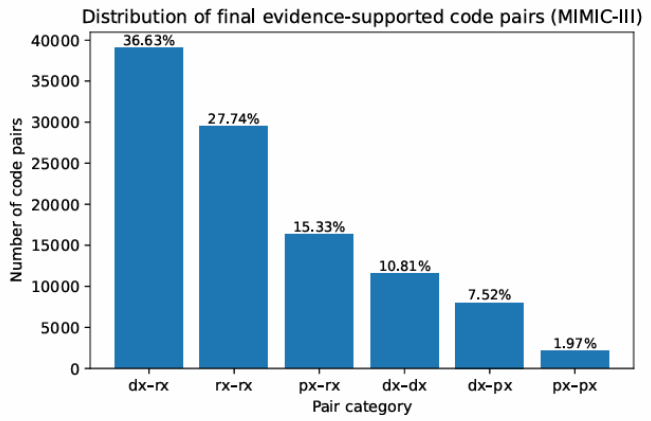
- **Dataset**
  - MIMIC-III, MIMIC-IV
- **Medical Coding**
  - Drug codes: Anatomical Therapeutic Chemical (ATC) classification
  - Diagnosis/Procedure : The Clinical Classification System (CCS)
- **Task**
  - Sequential Diagnosis Prediction: Given a patient's visit sequence $X_t = \{V_1, V_2, \ldots, V_t\}$, the objective is to predict diagnosis codes for the next visit V_{t+1}
- **Evaluation Metrics**
  - AUPRC
  - Acc@k
  - F1-score

Metric	MIMIC-III	MIMIC-IV
# Patients	7,515	18,829
# Visits (samples)	12,430	25,028
# Labels/sample	12.31	10.56
# Unique conditions (ICD)	515	562
# Conditions/sample	26.85	59.50
# Drugs/sample	70.10	118.16
# Unique drugs	471	510
# Procedures/sample	6.49	5.41
# Unique procedures	280	322

Evaluation – KG Analysis

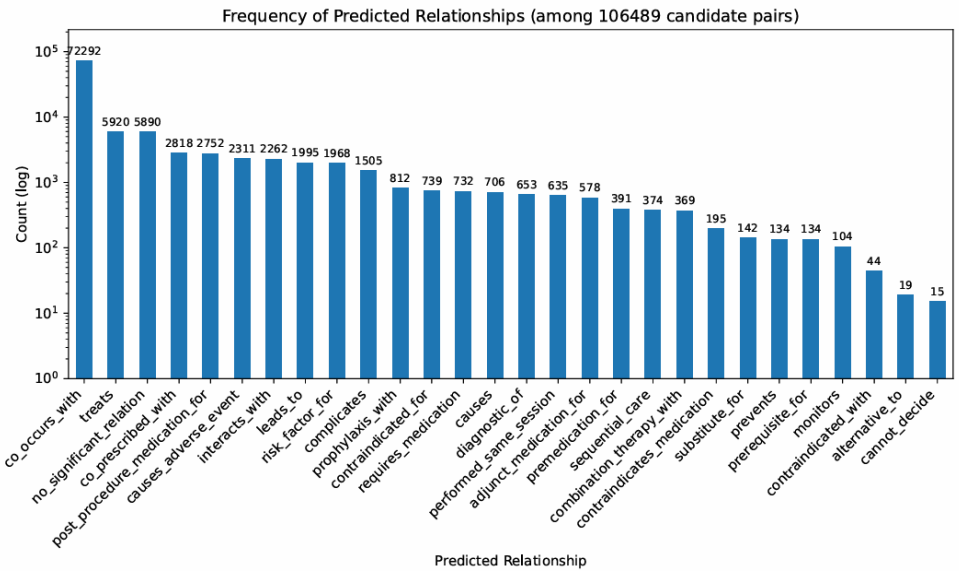
Evidence-Supported KG Construction

- Built from **100K+ candidate code pairs**
- Dominant pair types:
  - **dx–rx (36.6%)** → strong diagnosis–treatment links
  - **rx–rx (27.7%)** → medication co-prescription patterns
  - **px–rx (15.3%)** → procedure–medication interactions



Distribution of Predicted Relations

- Most frequent relations:
  - co_occurs_with, treats, co_prescribed_with
- Long-tail of clinically meaningful relations:
  - e.g., *prevents*, *contraindicated_with*, *monitors*



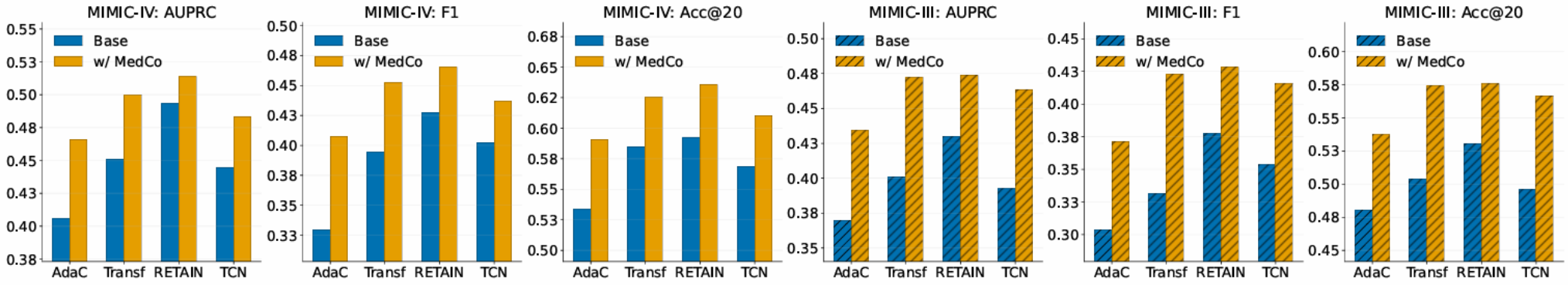
Evaluation – KG Analysis

Clinical Expert Validation

- **50 edges audited** by 2 clinicians
- **Rating scale:** 1 (wrong) → 5 (correct)
- **Overall score: 4.84 ± 0.29**
- Perfect scores (5.0):
  - *treats*, *risk_factor_for*, *post_procedure_medication_for*
- Lowest (still strong):
  - *no_significant_relation* (4.40) → context-sensitive

Predicted relationship	Mean	Std
causes_adverse_event	4.70	0.27
co_occurs_with	4.80	0.45
co_prescribed_with	4.80	0.27
complicates	4.90	0.22
interacts_with	4.90	0.22
leads_to	4.90	0.22
no_significant_relation	4.40	0.42
post_procedure_medication_for	5.00	0.00
risk_factor_for	5.00	0.00
treats	5.00	0.00
Overall	**4.84**	**0.29**

Evaluation - RQ1: Plug-in Enhancement Evaluation



RQ1: Does our method enhance the performance of EHR models as a complementary concept encoder?

- Integrated **MedCo** as a plug-in encoder into multiple EHR models
- Consistently improved downstream prediction performance
- Demonstrated strong **model-agnostic generalization** across architectures
- Validated LINKO's effectiveness in **enhancing medical concept representations**

Evaluation - RQ2: Baseline Comparison

RQ2: How does MedCo perform compared to existing medical code encoders?

- Base model is Transformer, other rows represent Base with added medical concept encoders
- Compare against:
  - **Base Transformer**
  - **Ontology-based encoders**: GRAM, MMORE, KAME, HAP
  - **KG-based methods:** ADORE, KAMPNet, GraphCare, LINKO
- **MedCo** consistently **outperforms**
- Label category performance shows **robustness for rare and low-frequency medical codes.**

	Model	General Performance					Label Category Performance (AUPRC)			
		AUPRC	F1	Acc@15	Acc@20	Acc@30	0–25%	25–50%	50–75%	75–100%
MIMIC-III	Base	41.00	33.16	47.20	50.40	58.80	40.60	47.30	72.80	78.40
	GRAM	41.70	34.60	48.60	51.90	59.80	42.20	50.10	74.10	79.30
	MMORE	42.60	35.30	49.20	52.60	60.40	42.80	51.20	74.80	80.00
	KAME	42.18	35.07	48.97	52.06	60.06	41.84	49.76	74.06	79.14
	G-BERT	42.47	35.38	49.26	52.33	60.27	42.06	50.04	74.24	79.28
	HAP	42.36	35.17	49.11	52.18	60.18	41.95	49.92	74.13	79.24
	ADORE	42.58	35.43	49.33	52.47	60.32	42.18	50.09	74.29	79.31
	KAMPNet	43.06	35.96	49.88	53.07	60.86	42.63	50.83	74.77	79.89
	GraphCare	43.35	35.46	52.76	56.00	62.75	44.80	58.16	70.97	65.72
	LINKO	44.91	38.20	52.30	55.62	62.80	44.80	55.20	**76.43**	83.41
	CoMed	**47.21**	**42.28**	**54.20**	**57.40**	**64.39**	**47.67**	**66.29**	76.02	**86.70**
MIMIC-IV	Base	45.10	39.10	54.80	58.10	64.70	46.80	53.60	53.90	77.20
	GRAM	46.30	40.00	55.60	58.90	65.70	47.70	54.60	54.80	77.90
	MMORE	46.90	40.60	56.10	59.40	66.10	48.20	55.10	55.70	78.40
	KAME	45.97	39.66	55.27	58.57	65.26	47.16	54.24	54.28	77.63
	G-BERT	46.37	40.07	55.68	58.96	65.66	47.58	54.73	54.87	78.04
	HAP	46.33	39.88	55.57	58.92	65.58	47.52	54.57	54.83	77.92
	ADORE	46.52	40.23	55.84	59.09	65.79	47.74	54.84	55.06	78.13
	KAMPNet	47.08	40.96	56.46	59.77	66.47	48.26	55.57	56.16	78.68
	GraphCare	46.90	40.12	55.89	59.14	65.71	48.31	**60.31**	73.70	66.33
	LINKO	48.14	42.05	57.78	61.21	67.97	49.61	56.87	57.58	80.27
	CoMed	**50.00**	**45.25**	**59.22**	**62.54**	**68.71**	**52.01**	57.70	**59.35**	**81.19**

Evaluation - RQ3: Ablation Study

RQ3: What is the impact of each component of LINKO on performance?

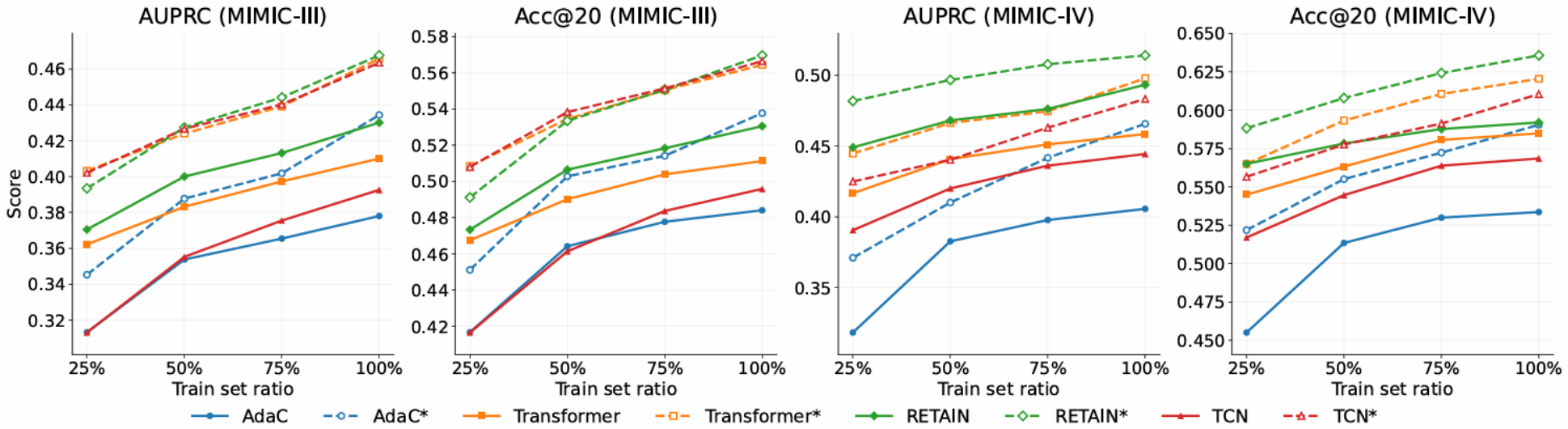
- Removing any key component of MedCo leads to **performance degradation**
  - Adding the induced KG gives the largest improvement.
  - Adding edge features gives further gains
  - Adding LLM-based node descriptions further improves performance, showing that textual semantics complement graph structure
  - Enabling LoRA fine-tuning gives the best final results, indicating the benefit of adapting the LLM to the cohort and task.

	Model	PRAUC	F1	Acc@20
MIMIC-IV	Base (Transformer)	45.10	39.10	58.10
	Base + KG	48.55	42.60	60.90
	Base + KG + Edge feat.	48.72	42.85	61.10
	Base + KG + Edge feat. + LLM (freeze)	49.38	43.71	61.47
	Base + KG + Edge feat. + LLM (LoRA)	**50.00**	**45.25**	**62.54**
MIMIC-III	Base (Transformer)	41.00	33.16	50.40
	Base + KG	45.79	38.90	55.60
	Base + KG + Edge feat.	45.91	39.20	55.85
	Base + KG + Edge feat. + LLM (freeze)	46.10	40.45	56.44
	Base + KG + Edge feat. + LLM (LoRA)	**47.21**	**42.28**	**57.40**

Evaluation – RQ4: Data Insufficiency Analysis

RQ4: How does our method mitigate data insufficiency limitations and rare disease predictions?

- Evaluating LINKO under varying training dataset sizes to simulate limited-data scenarios, it consistently outperforms baselines in data-scarce settings.
- In the Plug-in Enhancement setting, MedCo significantly improves EHR model performance even with reduced training data.



Outline

- Background and Motivation
- Proposed Method
- Evaluation
- **Conclusion**

Conclusion

We present MedCO, an LLM empowered graph learning framework for medical concept representation.

- Built an evidence-grounded heterogeneous medical KG using type-constrained, evidence-based LLM prompting
- Used LLMs to induce typed relations and generate node/edge semantics and Convert the KG into a text-attributed graph
- Jointly learned embeddings with LoRA-tuned LLM + heterogeneous GNN
- Achieved the best downstream results on MIMIC-III / IV

Thank you

mohsen.nayebi@ku.edu