

Једно решење претраге релационих база података уз помоћ Graph RAG система

One Approach to Relational Database Search Using a Graph RAG System

Игор Илић, Факултет техничких наука, Нови Сад

Област – РАЧУНАРСТВО И АУТОМАТИКА

Кратак садржај – Релационе базе података су и даље ослонац пословних информационих система, али рад са њима захтева познавање SQL-а. Иако су велики језички модели значајно приближили могућност постављања упита на природном језику, претварање тих питања у поуздане и прецизне одговоре над структурираним подацима остаје изазов. Чисто текстуални RAG (енгл. *Retrieval Augmented Generation*) често занемарује шеме, кључеве и ограничења, док је директно превођење природно језичког упита у SQL уз помоћ LLM-ова крхко и тешко провекливо. Овај рад представља GraphRAG (енгл. *Graph Retrieval Augmented Generation*) решење проблема, којим се омогућава природним језиком приступ релационим подацима.

Кључне речи: Релациона база података, граф база података, векторска база података, велики језички модели, вештачка интелигенција

Abstract – *Relational databases remain the backbone of business information systems, but working with them requires knowledge of SQL. While Large Language Models have significantly brought closer the possibility of querying using natural language, turning those questions into reliable and precise answers from relational data remains a challenge. Pure textual RAG (Retrieval Augmented Generation) often ignores schemas, keys, and constraints, while the direct translation of a natural language query into SQL using LLMs is fragile and difficult to verify. This paper presents the GraphRAG (Graph Retrieval Augmented Generation) solution to the problem, which enables natural language access to relational data.*

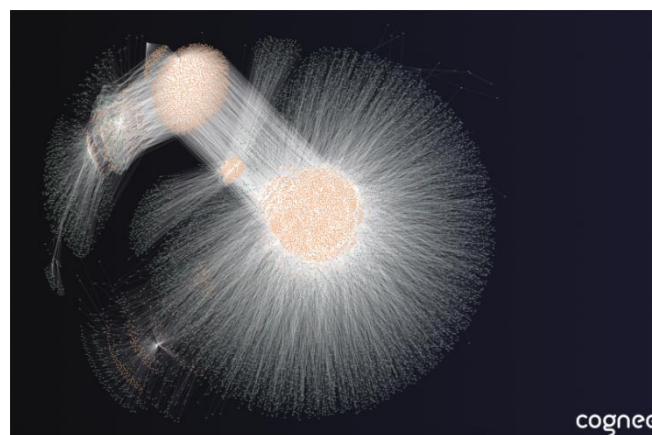
Keywords: *Relational Databases, Graph Databases, Vector Databases, Large Language Models, Artificial Intelligence*

НАПОМЕНА: Овај рад проистекао је из мастер рада чији ментор је био др Иван Каштелан, ред. проф.

1. УВОД

Релационе базе података су и даље ослонац пословних информационих система, али рад са њима захтева

познавање SQL-а, шема и логике спајања табела. Иако су велики језички модели (енгл. *Large Language Models, LLM*) значајно приближили могућност постављања упита на природном језику, претварање тих питања у поуздане одговоре над структурираним подацима остаје изазов. Чисто текстуални RAG (енгл. *Retrieval Augmented Generation*) занемарује шеме, кључеве и ограничења, док је директно превођење природно језичког упита у SQL уз помоћ LLM-ова крхко. Мигрирањем релационих података у GraphRAG (енгл. *Graph Retrieval Augmented Generation*) систем, омогућиће се природно језичким приступ релационим подацима. GraphRAG систем спаја граф претрагу са векторском семантичком претрагом како би се премостио јаз између природног језика и релационе структуре. Предложени дизајн избегава крхкост директне генерације SQL синтаксе тако што утемељује LLM у структурираним доказима извученим из изворне SQL базе. На слици 1. можемо да видимо резултат миграције релационе базе у граф базу за GraphRAG системе.



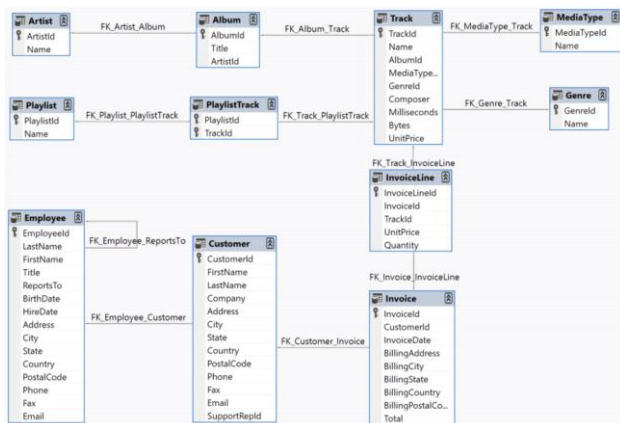
Слика 1. Граф визуализација мигриране релационе базе података у GraphRAG систем

2. ТЕОРИЈСКЕ ОСНОВЕ

2.1. Релационе базе података

Релационе базе података заснивају се на математичким темељима теорије скупова и предикатне логике првог

реда. Концепт је формализовао Едгар Ф. Код 1970. године [1]. Користе се за управљање организованим подацима у различитим секторима, као што су финансије (банкарство, управљање рачунима), електронска трговина (наруџбине, купци, производи), здравство (медицински картони пацијената), логистика (праћење пошљици) и управљање залихама (праћење нивоа залиха). На слици 2. можемо видети пример шеме података музичке радње у једној релационој бази, иста ова база података ће бити коришћена за верификацију резултата касније у раду.



Слика 2. Пример шеме релационе базе података

2.2. Граф базе података

Граф базе података моделује податке као чворове (ентитете) и везе (релације) са својствима, у оквиру *property graph* парадигме. *Cypher* је декларативни језик за подударење патерна (енгл. *pattern matching*) и модификацију таквих графова. Предности укључују експлицитно моделирање веза, интуитивне упите над путањама и добру читљивост. Граф базе су природна подлога за GraphRAG системе јер експлицитно моделују везе, омогућавају ефикасне *multi-hop* претраге и нуде алгоритме који подижу квалитет и објашњивост контекста.

2.3. Векторске базе података

Дигитална трансформација и експоненцијални раст неструктурираних података — попут текста, слика, аудио и видео записа — поставили су нове изазове пред традиционалне системе за управљање подацима. Конвенционалне релационе базе података показале су се недовољним за руковање сложеним семантичким претрагама које захтевају савремене апликације вештачке интелигенције. Векторска база података је специјализована врста базе података дизајнирана за индексирање и управљање високодимензионалним векторским уграђивањима (енгл. *embeddings*) [2]. За разлику од традиционалних база података које се ослањају на прецизно подударење вредности (нпр. SQL упити *WHERE*), векторске базе података омогућавају претрагу сличности (енгл. *Similarity search*). Оне проналазе податке који су семантички слични упиту, а не нужно они који садрже идентичне кључне речи. Срце сваке векторске базе су векторска уграђивања. То су нумерички прикази (вектори) који претварају сложене, неструктуриране податке у низове

бројева у високодимензионалном простору. Ови бројеви хватају семантичко значење и односе између података.

2.4. Велики језички модели

Вештачка интелигенција постала је катализатор револуционарних промена у начину на који машине разумеју и генеришу људски језик. Велики језички модели (енгл. *Large Language Models, LLM*) представљају врхунац деценија истраживања у области обраде природног језика (енгл. *Natural Language Processing, NLP*) и дубоког учења. Велики језички модели су врста неуронских мрежа темељених на архитектури трансформера [3], обучених на масовним количинама текстуалних података, који могу разумети, генерисати и манипулисати људским језиком на софистициран начин.

2.5. Retrieval Augmented Generation

Retrieval-Augmented Generation је техника за побољшање тачности и поузданости генеративних AI модела помоћу информација добијених из специфичних и релевантних извора података. Ова спољашња база знања може укључивати интерне документе компаније, истраживачке радове или било који други корпус структурираних или неструктурираних података. Основна идеја је да се генеративне способности LLM-а прошире механизмом за проналажење информација, омогућавајући моделу да даје одговоре који су не само кохерентни, већ и утемељени на проверљивим, ажурним подацима. Ово се постиже дељењем докумената у сегменте, векторски уграђивање тих сегмената па потом у времену претраге издвајањем најрелевантнијих сегмената из свих докумената.

2.6. Graph Retrieval Augmented Generation

Из недостатака адекватног повезивања информација из различитих сегмената докумената у RAG системима нови приступ настаје под именом GraphRAG [4]. GraphRAG је свеобухватни назив за различите системе који интегришу граф базе података за свој систем меморије за велике језичне моделе. Овај рад је заснован на *Cognee* GraphRAG пројекту отвореног кода. GraphRAG решава изазове адекватног повезивања информација тако што користи графове знања као основу за претрагу информација. Графови знања су структурирани као мреже ентитета (чворови) и њихових међусобних релација (везе), што GraphRAG-у омогућава семантичко претраживање које прати логичке везе, уместо да се ослања искључиво на статичку векторску сличност.

3. КОНЦЕПТ РЕШЕЊА

GraphRAG системи нису направљени са нативном подршком за релационе базе, намењени су фундаментално за обраду и процесуирање текстуалних података попут докумената. Ти документи су анализирани и процесуирани уз помоћ LLM-ова за релативне концепте и ентитете. Те информације се чувају уз сегменте и потом користе као контекст за

будуће упите од корисника (енгл. *User Query*). У случају релационих база, анализа и процесуирање података уз помоћ LLM-ова нису ни могући ни потребни јер је свака информација нужна и већ сведена на сам минимум корисности. Направљен је систем директног мапирања података из релационе базе у податке који GraphRAG системи могу и знају да користе. Мигрирани подаци су сачувани у релационој бази у граф бази података, па су потом ови подаци векторски уграђени (енгл. *Embedding*) и прикладно сачувани у векторску базу података. Овакав систем је направљен унутар *Cognee* GraphRAG система као проточна структура (енгл. *custom pipeline*) која служи само за мигрирање релационих база у *Cognee* GraphRAG систем [5].

3.1. Мапирање ентитета

TableType, *TableRow* и *ColumnValue* чине троделну, онтолошки утемељену репрезентацију релационих података у граф-парадигми. Они наслеђују метамодел *DataPoint* који је дефинисан у *Cognee* систему, метамодел *DataPoint* омогућава директно мапирање елемената у графу са елементима у векторској бази и омогућава дефинисање тачно шта од информација треба бити прослеђено LLM-у током упита корисника. Сваки наследник метамодела *DataPoint*-а треба да дефинише адекватне вредности у пољима:

1. *metadata* – У овом пољу се спецификује на основу које вредности треба радити векторско уграђивање, очекиван улаз за вредност је конкретно поље из модела (на пример, *description* поље)
2. *description* – У овом пољу се спецификује које текстуалне информације треба приложити LLM-у приликом давања одговора на упит корисника, када је конкретни чвор део релевантних резултата претраге.

Наслеђивањем *DataPoint*-а на овај начин нам је омогућено коришћење *Cognee* метода за креирање GraphRAG репрезентације, односно прављењем сопствене онтолошке репрезентације података и коришћењем *Cognee* система претраге података. Када је мапирање завршено додани су новокреирани чворови и везе у саму граф базу података. *Cognee* систем има своју апстракцију за комуникацију са граф и векторским базама података, има јасно дефинисане интерфејсе који су исти за све различите моделе који су наследили *DataPoint* и за све подржане типове база података.

Систем врши пресликавања ентитета описана у Табели 1. Систем додатно спаја ентитете у везе у графу на основу начина дефинисаних у Табели 2.

Табела 1. Мапирање релационих модела на GraphRAG моделе

Релациони модел	GraphRAG модел
Табела	<i>TableType</i> чвор
Ред у табели	<i>TableRow</i> чвор
Примарни кључ	Јединствени идентификатор <i>TableRow</i> чвора

Вредност колумне у реду у табели	<i>ColumnValue</i> чвор
----------------------------------	-------------------------

Табела 2. Мапирање веза у GraphRAG систему на основу релационих односа

Ентитет извор	Ентитет циљ	Име веза	Опис везе
<i>TableRow</i>	<i>TableType</i>	<i>is_part_of</i>	Припадност реда његовој основној табели.
<i>ColumnValue</i>	<i>TableRow</i>	Име колоне	Вредност у колони конкретних редова у табели.
<i>TableRow</i>	<i>TableRow</i>	Име колоне страног кључа	Везу између редова кроз страни кључ.

3.2. Векторско уграђивање

Након мапирања података из релационе базе у чворове и везе у граф базу ти подаци су векторски уграђени и сачувани у векторску базу података. Ово је једноставно урађено у *Cognee* систему коришћењем већ постојећих метода:

- *index_data_points* – ради мапирање чворова у векторску базу података.
- *index_graph_edges* – ради мапирање веза у векторску базу података.

Позивањем ових функција добијене су четири табеле у којима се чувају векторски уграђене вредности. Конкретно:

- *ColumnValue_properties*
- *TableRow_properties*
- *TableType_name*
- *EdgeType_relationship_name*

3.3. Претрага по упиту корисника

Cognee методе претраге аутоматски раде са мигрираном релационом базом. Сама претрага је тренутно написана тако да ради пројекцију граф и векторске базе у RAM меморију, одатле се прави листа свих триплета (два чвора и веза између њих) и сваки триплет се бодује за релевантност на основу следеће формуле:

$$R = d_{\epsilon_1} + d_{\epsilon_2} + d_v$$

где је: d_{ϵ_1} - векторска косинусна удаљеност чвора 1 од упита корисника, d_{ϵ_2} – векторска косинусна удаљеност чвора 2 од упита корисника, а d_v - векторска косинусна удаљеност везе од упита корисника.

Векторска косинусна удаљеност је мера сличности између два вектора у вишедимензионалном простору. Она мери косинус угла између та два вектора. Од листе свих триплета узима се *top_k* број триплета са најмањим резултатом укупне удаљености од корисниковог упита. Листа релевантних триплета се претвара у текст. Ово је постигнуто тако што *DataPoint* модели у себи имају поље *description* које се директно претвара у опис чвора за LLM, у овом пољу је специфицирано тачно шта представља који податак

и шта се у њему садржи. Ова вредност је динамички генерисања током процеса миграција. На пример током миграција *ColumnValue* чвора опис је подешен да буде:

"column from relational database table={table name}. Column name={key} and value={value}. This column has the following ID: {column_node_id}"

Из ових информација на корисничко питање LLM успева успешно да разазна вишеслојно порекло информација и њихово значење и релације и да да адекватан одговор базиран само на приложеном контексту.

4. РЕЗУЛТАТИ

У наставку је описан резултат миграције релационе базе података једне фиктивне музичке радње [6]. Шема ове релационе базе се може видети на слици 2. На слици 1 се види визуализација граф базе настале овом миграцијом. Граф база настала од ове миграције садржи 19437 чворова и 57818 веза. Миграција је трајала 27 минута и 5 секунди. Величина меморије датотеке *Kuzi* граф базе настале миграцијом је 63MB. Величина датотеке *LanceDB* векторске базе настале од ове миграције је 246,6MB. Величина иницијалне *SQLite* релационе базе за миграње је 1.1MB. После миграција података могуће је радити природно језичке упите везане за податке који се налазе у релационој бази података. Питања попут: "Које су фактуре од муштерије *Leonie Köhler*", "Које су све песме део плејлисте *Heavy Metal Classic*", "Које су све муштерије из Прага?" су све примери питања која се успешно и конзистентно одговарају. Приликом верификације резултата над миграцијом релационом базом музичке продавнице постављено је питање да се пронађу фактуре везане за сваку муштерију појединачно и резултати су верификовани поређењем резултата SQL претраге за исте информације, а за сваку муштерију су успешно пронађене његове фактуре без халуцинација и изузетака. Докле год тачан одговор на питање не захтева више резултата него што је доступно контексту дефинисаним *top_k* бројем релевантних триплета. Са друге стране одговори на комплекснија питања попут "Који су све радници исте националности као муштерија *Leonie Köhler*" *Cognee* претрага не може да одговори јер питање о националности је потребно експлицитно да се постави. Није могуће квалитетне имплицитне претраге правити због лимитација LLM контекста, односно јер неће довољно релевантних информација бити нађено за такво имплицитно питање.

5. ЗАКЉУЧАК

На примеру јавно доступне базе музичке радње демонстриран је комплетан процес: од дефинисања онтологије и процеса миграције, преко изградње графа знања и векторског индекса, до практичне употребе система за претрагу и одговор на питања. Исправност имплементације верификована је комбинацијом аутоматизованих тестова и ручног увида.

Практичне импликације су двоструке: миграцијом релационе базе у GraphRAG систем омогућава се интуитивнији приступ подацима из релационих извора путем природног језика, али уз реалне трошкове у погледу простора и времена миграције и претраге.

Тренутно се ти трошкови чини превише великим за конзистентно и лако коришћење на комерцијалном нивоу и нивоу свакодневног коришћења, али имплементација може бити интересантна у истраживачком пољу са аспекта омогућавања природно језичке анализе података који се касније морају верификовати да се потврди тачност резултата.

6. ЛИТЕРАТУРА

- [1] Codd, E. F. (1970) "A Relational Model of Data for Large Shared Data Banks." *Communications of the ACM* 13 (6): pp. 377-387. <https://doi.org/10.1145/362384.362685>.
- [2] Grohe, M. (2020). "word2vec, node2vec, graph2vec, x2vec: Towards a theory of vector embeddings of structured data", in *proceedings of the 39th ACM SIGMOD-SIGACT-SIGAI symposium on principles of database systems*, pp. 1-16.
- [3] Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. (2017) "Attention is all you need", *Advances in neural information processing systems* 30.
- [4] Jonathan Larson, Steven Truitt (2024). "GraphRAG: Unlocking LLM discovery on narrative private data", <https://www.microsoft.com/en-us/research/blog/graphrag-unlocking-llm-discovery-on-narrative-private-data/> (приступано 10.11.2025.)
- [5] Cognee Open Source project. <https://github.com/topoteretes/cognee/tree/main> (приступано 10.11.2025.)
- [6] Chinook (n.d.) Music store relational database. <https://github.com/lerocha/chinook-database/tree/master> (приступано 10.11.2025.)

Кратка биографија:



Игор Илић, рођен у Новом Саду, 04.11.1996. године.

Основне студије завршио 2019. године, смер Рачунарство и аутоматика, на Факултету техничких наука, Нови Сад.

Контакт: igorilic03@yahoo.com