

基于电子病历系统关系数据库构建患者诊疗图谱

张百平

(广州医科大学附属第三医院 广州 510000)

〔摘要〕 介绍在电子病历 (Electronic Medical Record, EMR) 系统中使用图数据库的目的, 阐述使用 Neo4j 对 EMR 系统关系数据进行建模的全过程, 构建某医院门诊患者就诊记录多关系异构诊疗知识图谱, 实现 EMR 系统中关系数据库结构可视化, 并探索利用图数据库来挖掘数据管理和分析的潜力。

〔关键词〕 图数据库; 关系可视化; 电子病历; 知识图谱

〔中图分类号〕 R-058 〔文献标识码〕 A 〔DOI〕 10.3969/j.issn.1673-6036.2022.12.009

Constructing Patient Diagnosis and Treatment Knowledge Graph Based on Relational Database of Electronic Medical Record

System ZHANG Baiping, The Third Affiliated Hospital of Guangzhou Medical University, Guangzhou 510000, China

〔Abstract〕 The paper introduces the purpose of using graph database in Electronic Medical Record (EMR) system, expounds the whole process of using Neo4j to model the relational data of EMR system, constructs the multi-relational heterogeneous diagnosis and treatment knowledge graph of outpatient records of a hospital, realizes the visualization of the structure of relational database in EMR system, and explores the potential of using graph database to promote data management and analysis.

〔Keywords〕 graph database; visualisation of relations; Electronic Medical Record (EMR); knowledge graph

1 引言

从一个典型的医疗关系表结构中抽象出一个丰富的图结构, 图数据库方法可广泛利用这种图形表示。图数据库能够直接处理各种查询, 否则就需要在规范化关系表中执行深度连接操作。这些查询涉及医疗实体之间的关系, 可以用于协同过滤以及为医疗保健提供个性化开发服务。图数据库可以处理这类查询, 并提供直观的查询表示和对数据的访

问。作为非规范化表的一种形式, 图数据库可以避免生成和复制大量的表, 为医疗系统提供一个低成本的数据管理框架。

2 在电子病历中使用图数据库的目的

2.1 图数据库和关系数据库对比

关系数据库 (如 Oracle 和 MySQL) 在获取重复的表格数据方面表现出色。尽管名称中有“关系”一词, 但关系数据库在存储或表达存储的数据元素之间关系方面却效率很低。与关系数据库不同, 图数据库完全围绕数据关系构建。图数据库不将关系视为模式结构, 而将其视为数据。

对关系数据库与图数据库 (Neo4j) 在数据结构、

〔修回日期〕 2022-09-26

〔作者简介〕 张百平, 高级工程师, 博士研究生, 发表论文 3 篇。

数据模型特征和查询功能等方面进行比较^[1]，见表 1，发现当数据之间的关联度较高时，使用图数据库具有明显的优势。从数据模型、存储方式、特性和查询语言等方面分析多种 NoSQL（不仅是 SQL）数据库与关系数据库相比的优缺点^[2]，提出一个基于社交网络的微基准，通过测试 5 个不同的数据库（Dex、Neo4j、RDF-3X、Virtuoso 和 PostgreSQL）来比较不同数据管理模式^[3]。

表 1 图数据库与关系数据库对比

项目	关系数据库	图数据库
主要目标	管理和遍历连接的 数据	专注于结构化数据的存 储和检索
主要关注点	关注实体	关注实体、实体之间的 关系及其属性
关系表示形式	通过外键在逻辑上 进行连接	物理连接
数据层扩展 能力	不具备轻松横向扩 展的能力	可以线性横向扩展

2.2 电子病历中使用图数据库的优势

在图数据库中查找相关节点比在关系数据库中查找相关数据行更加有效。图数据模型的灵活性允许添加新的节点和关系，而原始数据保持不变。以图格式存储数据允许用户从不同角度查看相同数据和连接。数据和关系网络有助于找到最佳路径、模式匹配或提供建议。将数据从关系数据库移至图数据库可以利用所有图数据库提供的优势。

通常，门诊电子病历（Electronic Medical Record, EMR）数据由患者、疾病、医生、检验检查和药品组成。这些数据之间存在着许多联系。如果使用图数据库，可以更好地利用这些连接从海量医疗数据中做出更好的推断。要处理不断增长的连接数据，可以选择 Neo4j，这是一种针对关系管理而优化的非关系图数据库，以进行进一步的分析和决策。

3 基于 Neo4j 的电子病历系统关系数据建模

将数据从关系数据库映射到图数据库基本上是将

关系表示从一个数据库转换到另一个数据库，从表结构转换到图结构。关系数据库表中的数据实体是行（包含列），图数据库中的数据实体是节点（包含属性）。更具体地说，可以使用关系数据库模型的外键来构建边，从而将松散耦合的数据记录转换为高度有界的节点组。

3.1 建模特点

Neo4j 是一个属性图数据库，使用图的原生结构，即邻接表进行存储，便于快速处理关系^[4]。Neo4j 使用的是 Cypher 查询语言，Cypher 可以提供丰富、高效的表达式来描述关系查询^[5]。与其他图查询语言相比，Cypher 语言具有良好的发展前景^[6]。为了简化图分析的工作，笔者调查了几个图数据库。基于数据库对软件可追溯性的适用性审查^[7]，最终选择 Neo4j 作为研究对象。Cypher 对具有不同特征的数据进行操作，这些操作是在图的节点和边上执行的，而不是在表、元组以及与主键或外键的关系上执行。

Neo4j 图数据库建模是将任意域描述为节点和关系的连通图的过程。与传统关系型数据库不同，Neo4j 是一个无模式数据库，这意味着在添加数据之前不需要定义表和关系。一个节点可以有任何属性，任何节点都可以与其他节点相关联。Neo4j 数据库的数据模型在所包含的数据中是隐式的，Neo4j 数据建模是描述性的。Neo4j 的主要优点是对图数据库建模和查询的直观方式。使用 Neo4j 对门诊 EMR 关系型数据库进行图数据库建模，患者、疾病、药物及其之间关系将用 Neo4j 技术构建并存储在 Neo4j 中。

3.2 识别数据实体

第 1 步是识别数据实体类型，见图 1，有 4 种类型的数据实体：患者信息、药品字典、国际疾病分类（International Classification of Diseases, ICD）第 10 版、诊疗字典（LIS&RIS），每个表表示正在操作的不同类型的信息。

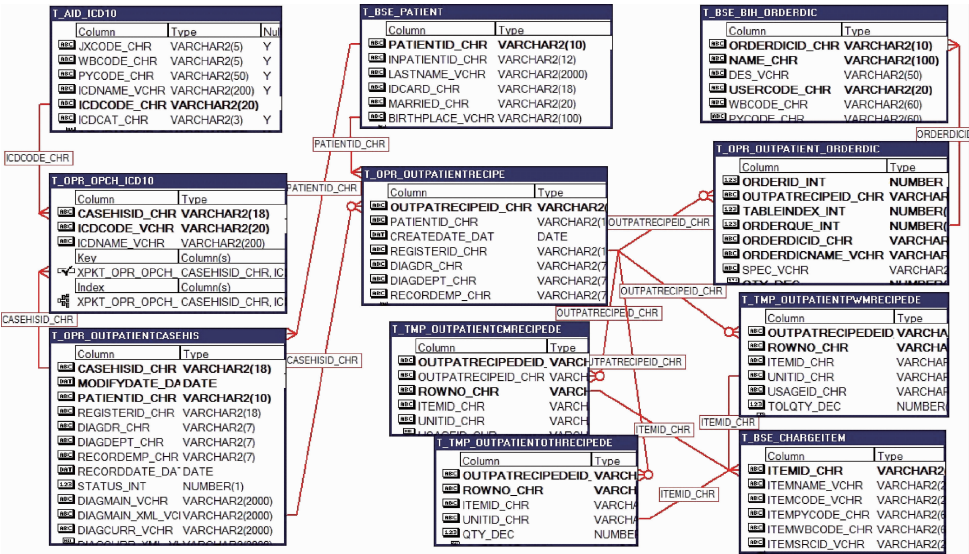


图 1 EMR 数据库关系数据模式的表结构逻辑关系

3.3 识别数据关系

接下来需要找到这些实体之间的语义关系。这一步不太明显，因为关系数据库并不是将关系作为一级公民来处理。患者单次就诊所产生的所有记录，包括检查处方、西药处方、中药处方、诊疗处方等都通过 CASEHISID_ CHR 这个就诊流水号直接或者间接串联起来。当正确标识数据实体和关系时，可以为每个数据实体创建节点，为每个关系创

建边。在图中找到并构建正确的边是一项关键活动。这些步骤影响数据结构有效响应最终用户问题的能力。包括将逻辑关系（外键）和隐式关系（公共属性）转换为物理存储在图数据结构中的显式边。使用 Neo4j（带标签的）属性图模型，其基本结构包括实体（节点）、属性（属性）、标签（类型），具有方向、开始节点和结束节点的关系（边）。最终识别到的关系类型，见表 2。

表 2 EMR 关系数据转换为图数据边（关系）

源表名	源表主键	源表外键	新增键	关系名	: STARTID	: ENIID
门诊病历 (T_ OPR_ OUTPATIENTCASEHIS)	CASEHISID_ CHR	PATIENTID_ CHR	PATIENTID ICD10_CODE_ CHR	Medical Record	ICD10_CODE_ CHR	PATIENTID
中药处方 (T_ TMP_ OUTPATIENTCMRECIPEDE)	OUTPATRECIPEID_ CHR	OUTPATRECIPEID_ CHR ITEMID_ CHR	PATIENTID ITEMID CASEHISID_ CHR	TCMP	PATIENTID	ITEMID
西药处方 (T_ TMP_ OUTPATIENTPWRECIPEDE)	OUTPATRECIPEID_ CHR	OUTPATRECIPEID_ CHR ITEMID_ CHR	PATIENTID ITEMID CASEHISID_ CHR	WMP	PATIENTID	ITEMID
诊疗处方 (T_ OPR_ OUTPATIENT_ ORDERDIC)	ORDERID_ INT	OUTPATRECIPEID_ CHR ORDERDICID_ CHR	PATIENTID CASEHISID_ CHR	DTP	PATIENTID	ORDERDICID_ CHR

3.4 构建 Neo4j 诊疗图谱

在 Intel（R）Xeon（R）6240 CPU@ 2.60 GHz（16 处理器）、RAM 32 GB、Windows Server 2012 R2 虚拟机服务器上，使用 neo4j - community - 4.3.6 构

建诊疗图谱。图数据库除了可以执行强大的连接操作之外，Neo4j 还允许用户创建索引。索引可以在节点或边的属性上创建，一个索引单元可以指向多个图模式单元，从而提高搜索效率。还可以在图上创建新的关系，见表 3。

表 3 新建关系、数量、代码及耗时

新建关系类型	Cypher 代码	数量（个）	耗时（毫秒）
NumofDiagnosis	MATCH (P: PatientInfo) - [MR: MedicalRecord] -> (ICD: ICD10)	320 262	90 753
	MATCH (P: PatientInfo) - [O: DTP] -> (L: LISRIS)		
	WHERE MR. CASEHISID_ CHR = O. CASEHISID_ CHR		
	WITH ICD, L, COUNT (*) AS N		
	CREATE (L) - [r: AuxiliaryDiagnosis { number: N }] -> (ICD)		
NumberofTCM	MATCH (P: PatientInfo) - [: TCMP] -> (M: Medicine)	257 909	1 275
	WITH M, P, COUNT (*) AS N		
	CREATE (P) - [r: NumberofTCM { number: N }] -> (M)		
NumberofWM	MATCH (P: PatientInfo) - [: WMP] -> (M: Medicine)	571 489	3 069
	WITH M, P, COUNT (*) AS N		
	CREATE (P) - [r: NumberofWM { number: N }] -> (M)		
NumberofDT	MATCH (P: PatientInfo) - [: DTP] -> (L: LISRIS)	602 345	2 765
	WITH L, P, COUNT (*) AS N		
	CREATE (P) - [r: NumberofDT { number: N }] -> (L)		
NumberofTreat	MATCH (P: PatientInfo) - [MR: MedicalRecord] -> (ICD: ICD10)	224 684	134 519
	MATCH (P: PatientInfo) - [OO: TCMP WMP] -> (M: Medicine)		
	WHERE MR. CASEHISID_ CHR = OO. CASEHISID_ CHR		
	WITH ICD, M, COUNT (*) AS N		
	CREATE (M) - [r: NumberofTreat { number: N }] -> (ICD)		
AuxiliaryDiagnosis	MATCH (P: PatientInfo) - [: MedicalRecord] -> (ICD: ICD10)	72 171	2 111
	WITH ICD, P, COUNT (*) AS N		
	CREATE (P) - [r: NumofDiagnosis { number: N }] -> (ICD)		

通过绘制表示数据中常见模式的图形片段，可以将模型可视化，见图 2。

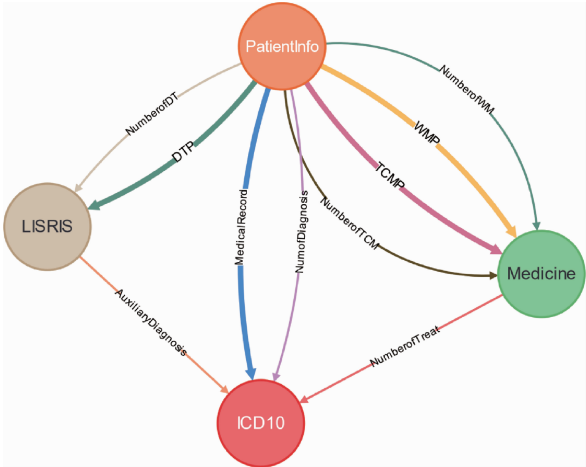


图 2 Neo4j 诊疗图谱节点及关系数据模式总览

经过以上环节的构建，获得患者诊疗图谱节点和关系基本情况，见表 4、表 5。患者诊疗图谱包含 270 871 个节点（4 类标签）和 6 091 656 个关系（10 种关系类型）。

表 4 诊疗图谱节点总览（个）

节点标签	采样数	属性平均数	关系平均数	关系最大数	总数
[Medicine]	513	118	493. 5	20 789	5 597
[PatientInfo]	21 515	69	26. 8	2 389	214 322
[LISRIS]	2 766	28	40. 2	8 273	27 872
[ICD10]	2 257	10	89. 6	82 944	23 080

表 5 诊疗图谱关系类型及数量（个）

关系类型	数量
MedicalRecord	963 303
TCMP	636 372
WMP	1 444 165
DTP	998 956
NumofDiagnosis	320 262
NumberofTCM	257 909
NumberofWM	571 489
NumberofDT	602 345
NumberofTreat	224 684
AuxiliaryDiagnosis	72 171

关系 MedicalRecord 记录着患者就诊时的详细信息，包括既往史、现病史、体温、发病时间、身份等基本情况。关系 TCMP 及 WMP 详细记录药物处

方的用法用量。NumofDiagnosis、NumberofTCM、NumberofWM、NumberofDT 分别表示患者在一段时间内所患某疾病次数、服用某中药次数、服用某西药次数及所做某检验检查项目次数。

4 知识图谱可视化分析应用

Neo4j 检索结果的速度比关系数据库快，比 MySQL 灵活。Neo4j 在结构类型查询方面的表现优于 MySQL，在使用整数数据时比 MySQL 慢。Neo4j 图数据库主要有 5 个构建块：节点、属性、关系、标签和数据浏览器。图数据库擅长遍历类型的查询，从而促进对关联概念链的探索。这种“连接发现”可以识别新的关系。Neo4j 不需要连接或索引就可以很容易地检索它的相邻节点或关系细节。下面以单个患者为例，在 Neo4j 图数据库中以图的形式展示患者 ID 为“0000001816”在某段时间的诊疗记录，可视化其在某段时间所患疾病，见图 3。所做检查及所用药物情况类似。

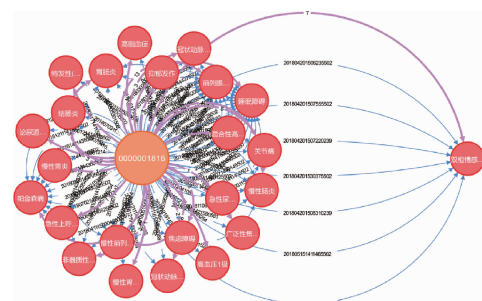


图 3 患者“0000001816”某段时间所患疾病情况

查询某些适应症药物的结果分析并可视化 EMR 中的规律与模式，见图 4。

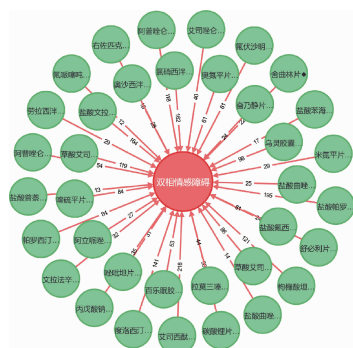


图 4 某段时期内用于治疗“双相情感障碍”的药物

归纳出对应于某一疾病或者某些疾病组合，医生们通常习惯于开哪些药。

基于医学数据的特点，使用属性图数据库 Neo4j 对 EMR 进行建模，实践表明，该模型比关系数据库模型具有更好的语义表达能力，能够更清晰地表达数据内部的语义关系，满足实际查询需求，并具有关系数据库模型无法实现的路径发现功能。

5 结语

大型医疗数据存储库是知识发现的来源。对这类存储库的有效分析通常需要对数据中的依赖关系有全面理解。知识图谱与任务无关，创建相同的知识图谱，可以用于许多不同的任务，甚至是机器学习之外的任务。知识图谱作为异构知识学习的默认数据模型，使用图数据库表示和查询疾病网络。

本文使用 Neo4j 对医疗数据进行建模，简洁直观地表达数据之间关系。使用图数据库有利于医疗数据分析和可视化，未来将添加机器学习机制，创建基于图数据库的医学推荐引擎。

参考文献

- 1 Miller J J. Graph Database Applications and Concepts with Neo4j [C]. Atlanta: Proceedings of the Southern Association for Information Systems Conference, 2013.
- 2 Nayak A, Poriya A, Poojary D. Type of NOSQL Databases and Its Comparison with Relational Databases [J]. International Journal of Applied Information Systems, 2013, 5 (4): 16 - 19.
- 3 Xia Y, Sun C. Property Graph Database Modeling and Application of Electronic Medical Record [C]. Harbin: 2018 Eighth International Conference on Instrumentation & Measurement, Computer, Communication and Control (IMCCC), 2018.
- 4 Lal M. Neo4j Graph Data Modeling [M]. Birmingham: Packt Publishing Ltd, 2015.
- 5 Rutgers P, Martella C, Voulgaris S, et al. Powerful and Efficient Bulk Shortest - path Queries: Cypher Language Extension & Giraph Implementation [C]. Redwood Shores: The Fourth International Workshop on Graph Data Management Experiences and Systems, 2016.
- 6 Holzschuher F, Peinl R. Performance of Graph Query Languages: Comparison of Cypher, Gremlin and Native Access in Neo4j [C]. Genoa: The Joint EDBT/ICDT 2013 Workshops, 2013.
- 7 Fette G, Kaspar M, Liman L, et al. Query Translation Between openEHR and i2b2 [C]. Hanover: EFMI - STC, 2019.