Ontology
Database Online
Database Normalization

BMI701 Introduction of Biomedical Informatics
Lab Session 3

Wei-Hung Weng

September 18, 2016

HMS DBMI — MGH LCS

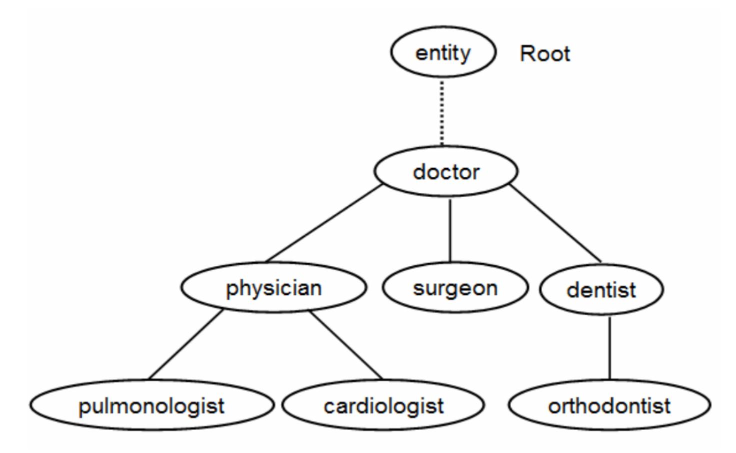
HARVARD
MEDICAL SCHOOL

MASSACHUSETTS
GENERAL HOSPITAL

Some Medical Databases

- MIMIC
  - Intensive care database
- ClinicalTrials.gov
- CDC.gov
- Medicare.gov
- CMS.gov
- National Practitioner Data Bank (NPDB)
- PubMed
- Web of Science

What Is Ontology?



Liu, 2012

Medical Ontology

- SNOMED-CT (for all medical terms)
- RxNorm (for medication)
- MeSH (for all biomedical terms)
- ICD-10 (for disease categorization)
  - `W22.02XD: Walked into lamppost, subsequent encounter.`
  - `W59.29XS: Other contact with turtle, sequel.`
  - `V97.33XD: Sucked into jet engine, subsequent encounter.`
  - Some bizarre codes
- FMA (for anatomy)
- HPO (for rare diseases)

Interconnectivity

- Upper level connection
- UMLS Metathesaurus
- Make sure you already have UTS account
- Two versions per year (now 2016AA)
- Concept unique identifier (CUI)
  - `C0031511|...|SNOMEDCT_US|OAS|154555009|Phaeochr...`
  - `C0031511|...|SCTSPA|PT|85583005|feocromocitoma`

Semantic Connection

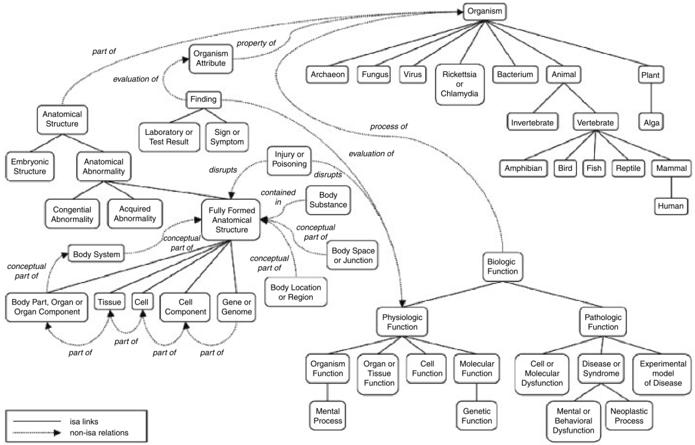


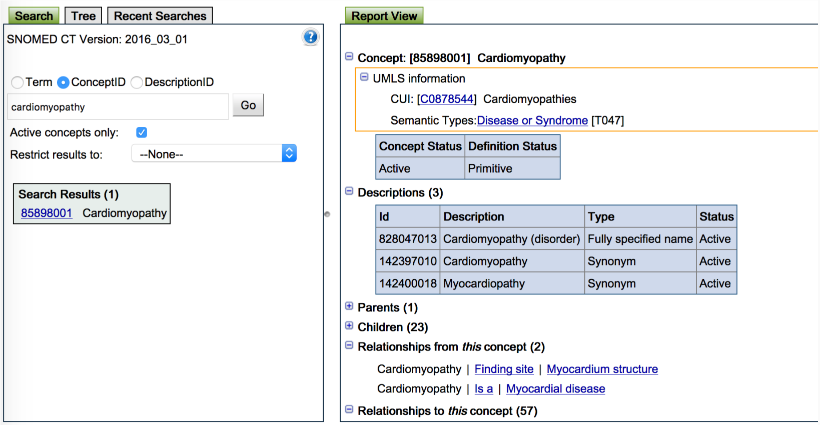
Figure 1. A portion of the UMLS semantic network

McCray, 2003

Some Medical Ontology

- BioPortal ontology repository
- UMLS
  - UTS web application for UMLS
- SNOMED
  - UTS web application for SNOMED
- RxNorm
  - RxNav (Web application for RxNorm)
- LOINC
- Human Phenotype Ontology
  - For rare, congenital diseases

Ontology in MySQL and R



SNOMED CT Version: 2016_03_01

○ Term ● ConceptID ○ DescriptionID

cardiomyopathy

Active concepts only: ☑

Restrict results to: --None--

Search Results (1)

85898001 Cardiomyopathy

⊟ **Concept: [85898001] Cardiomyopathy**

⊟ UMLS information

CUI: [C0878544] Cardiomyopathies

Semantic Types: Disease or Syndrome [T047]

Concept Status	Definition Status
Active	Primitive

⊟ **Descriptions (3)**

Id	Description	Type	Status
828047013	Cardiomyopathy (disorder)	Fully specified name	Active
142397010	Cardiomyopathy	Synonym	Active
142400018	Myocardiopathy	Synonym	Active

⊞ **Parents (1)**

⊞ **Children (23)**

⊟ **Relationships from *this* concept (2)**

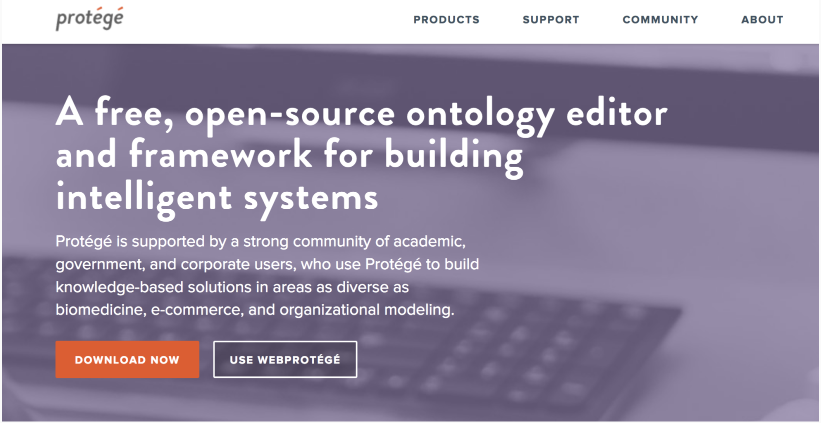
Cardiomyopathy | Finding site | Myocardium structure

Cardiomyopathy | Is a | Myocardial disease

⊟ **Relationships to *this* concept (57)**

- Loading SNOMED into MySQL
- github.com/ckbjimmy/bmi701lab/blob/master/lab03.R

Want to Edit or Create Your Own Ontology?



A free, open-source ontology editor and framework for building intelligent systems

Protégé is supported by a strong community of academic, government, and corporate users, who use Protégé to build knowledge-based solutions in areas as diverse as biomedicine, e-commerce, and organizational modeling.

Protégé

Making Database Online

- Using Amazon RDS
- Create MySQL instance on RDS
  - Remember your username, password, and check the RDS address

-
  ```
  mysqldump -host=localhost -user=root DB_NAME |
  mysql --host=YOUR_RDS_ADDRESS --user=YOUR_RDS_USER
  --password YOUR_RDS_PW DB_NAME
  ```

Using `RMySQL` to Check the Online Database

- `library(RMySQL)`
- `con <- dbConnect(MySQL(), user="YOUR_RDS_USER", password="YOUR_RDS_PW", dbname="DB_NAME", host="YOUR_RDS_ADDRESS")`
- `dbListTables(con)`
- `dbGetQuery(con, "select * from TABLE_NAME")`

Database Normalization

- Normalization theory
- Simply to say, the rules to divide your database
  - From a big table to several small tables
- Purpose
  - Minimizing data redundancy
  - Reducing data size
  - Eliminating anomalies during data insertion/update/deletion
  - Easy to maintain
- 1NF → 2NF → 3NF → BCNF (Boyce-Codd Normal Form) → 4NF → 5NF → ...
- BCNF is enough for most cases
- Column dependency

Column Dependency

Course ID	Name	Score
BMI701	Adam	A
STAT115	Shirley	A+
CS109	Wei-Hung	A-
...	...	...

- `Score` does not make sense if we remove `Course ID` and `Name` → `Score` is dependent on both `Course ID` and `Name`

First Normal Form (1NF)

- Normalizing step by step
- The foundation of database normalization in RDB
- Expanding the table
- Rules
  - Ensure that there is a primary key (PK)
  - Contains only atomic values
  - No repeating groups

First Normal Form (1NF)

Date	Name	Working Hour
Sep 9, Sep 10	Adam	8
Sep 12	Zak	12
Sep 14	Zak	4

- `Date` is not atomic

Course ID	Name	Score
BMI701	Adam	A+
	Husky	B-
STAT115	Shirley	A
CS109	Wei-Hung	A
	Mike	A-

- `Name` and `Score` are not atomic

First Normal Form (1NF)

Date	**Name1**	**Name2**	**Working Hour**
Sep 9	Adam	Rachel	8
Sep 12	Zak	Alexa	12
Sep 14	Zak	Adam	4

- `Name1` and `Name2` are repeating groups

First Normal Form (1NF)

Date	Name	Working Hour
Sep 9	Adam	8
Sep 10	Adam	8
Sep 12	Zak	12
Sep 14	Zak	4

Course ID	Name	Score
BMI701	Adam	A+
BMI701	Husky	B-
STAT115	Shirley	A
CS109	Wei-Hung	A
CS109	Mike	A-

First Normal Form (1NF)

Date (PK)	Name (PK)	Working Hour
Sep 9	Adam	8
Sep 9	Rachel	8
Sep 12	Zak	12
Sep 12	Alexa	12
...	...	...

- Saving the duplicated or repeated items to different records (with PK)

Second Normal Form (2NF)

- So many redundant data after 1NF
- Removing ”partial (functional) dependency”
- Rules
  - Following 1NF
  - All non-key attributes should be fully functional dependent on the primary key

Second Normal Form (2NF)

CID	**CName**	**CInstr**	**SID**	**SName**	**Score**
PK			**PK**		
BMI701	Intro of BMI	Adam	1234	James	A+
BMI701	Intro of BMI	Adam	2834	Husky	B-
STAT115	Bioinformatics	Shirley	2834	Husky	A
CS109	Data Sci	Peter	9877	Wei-Hung	A
CS109	Data Sci	Peter	9572	Mike	A-

- Partial dependency
  - `Student Name` is dependent on `Student ID`
  - `Course Name` and `Instructor` is dependent on `Course ID`

Second Normal Form (2NF)

- Problem?
  - Adding: What if a 2nd year student Josh, who doesn't need to take any course?
  - Updating: What if we want to change the course name? → Need to replace all values (inefficient!)
  - Deleting: What if James want to drop BMI701? His data will disappear
- Solution
  - Breaking the big table into multiple small tables
  - Three tables in our case

Second Normal Form (2NF)

CID **PK**	**SID** **PK**	**Score**
BMI701	1234	A+
BMI701	2834	B-
STAT115	2834	A
CS109	9877	A
CS109	9572	A-

Second Normal Form (2NF)

CID	**CName**	**CInstr**
PK		
BMI701	Intro of BMI	Adam
STAT115	Bioinformatics	Shirley
CS109	Data Sci	Peter

SID	**SName**
PK	
1234	James
2834	Husky
9877	Wei-Hung
9572	Mike

Third Normal Form (3NF)

- Data loss
- Removing ”transitive dependency”
- Rules
  - Following 1NF & 2NF
  - No transitive functional dependency (what’s this!?)
    - e.g. A → B & B → C, then A and C are transitive dependency

Third Normal Form (3NF)

CID PK	CName	InstrID	CInstr
BMI701	Intro of BMI 1	001	Adam
BMI702	Intro of BMI 2	001	Adam
STAT115	Bioinformatics	002	Shirley
CS109	Data Sci	003	Peter

- `InstrID` depends on `CID`
- `CInstr` depends on `CID`
- `CInstr` also depends on `InstrID` → Eliminating this transitive dependency!

Third Normal Form (3NF)

CID PK	CName	InstrID
BMI701	Intro of BMI 1	001
BMI702	Intro of BMI 2	001
STAT115	Bioinformatics	002
CS109	Data Sci	003

InstrID PK	CInstr
001	Adam
002	Shirley
003	Peter

Boyce-Codd Normal Form (BCNF) (optional)

- Only do BCNF if you have multiple PKs in the table
- Rules
  - Following 1NF & 2NF & 3NF
  - PK doesn't depend on other attribute

Student	**Problem**	**Mentor**
PK	**PK**	
Wei-Hung	ML	Pete
Wei-Hung	NLP	Alexa
David	ML	Jesse
Josh	NLP	Alexa

- Student, Problem → Mentor
- Mentor → Problem

Boyce-Codd Normal Form (BCNF) (optional)

- Changing PK
- Separating the relation

Student	**Mentor (PK)**
Wei-Hung	Pete
Wei-Hung	Alexa
David	Jesse
Josh	Alexa

Problem	**Mentor (PK)**
ML	Pete
NLP	Alexa
ML	Jesse

Problems of Normalization

- No need to do 3NF or BCNF everytime
- Lossless decomposition
  - Student, Problem → Mentor (disappeared)
  - Denormalization to 3NF
- Too many tables → ↓ system performance
- Or dividing the BCNF tables if there are columns merely used → put them into another table
- Denormalization
  - Disk is cheap
  - Space-time trade-off
  - Array/JSON

Take Home Message

- SNOMED-CT, RxNorm, MeSH, ICD-10
- UMLS Metathesaurus
- Use `RMySQL` to play with ontology
- NF: PK, atomic, no repeating groups → removing partial dependency → removing transitive dependency
- Use AWS RDS to upload your MySQL database
- Contact
  - Github repository
  - ckbjimmy@gmail.com
  - Linkedin: Wei-Hung Weng