LINKÖPING UNIVERSITY
Institutionen för datavetenskap
Patrick Lambrix / Huanyu Li

Tentamen

Advanced Data Models and Databases

2023

Grades:

For a pass grade all questions need to be answered and almost all correctly answered.

Instructions:

- This is an *individual* exam. If you have questions, ask the course leader.
- Send your answers to Patrick Lambrix via internal mail or e-mail.
- You can send answers as soon as you have finished them, i.e., no need to wait until you have answered everything.
- Deadline: February 1, 2024.
- If there are references to papers in the questions, then links to the papers will be on the course home page.

Question 1: Information Retrieval

Assume that we have two documents in our document base. Document 1 contains 'enzyme' 5 times, 'gene' 10 times, 'protein' 0 times and 'signal' 8 times. Document 2 contains 'enzyme' 0 times, 'gene' 0 times, 'protein' 7 times and 'signal' 1 time.

(i) Assume that we use the vector model for information retrieval. Assume that we are only interested in the words 'gene', 'enzyme', 'protein' and 'signal'.
1. Explain tf and idf in the vector model.
2. In which cases is a weight w_{ij} in a document vector equal to 0?
3. Give the document representations for Document 1 and Document 2 according to the tf-idf model.

(ii) Assume that we use the boolean model for information retrieval. Assume that we are only interested in the words 'gene', 'enzyme', 'protein' and 'signal'.
1. Give the document representations for Document 1 and Document 2 according to the Boolean model.
2. Represent the query for all documents containing gene or protein, but not signal. Compute then the completed DNF (disjunctive normal form) of the query.

Question 2: Description Logics

Define the following concepts using description logics:

C1: animals that have at least 4 legs and all foods they eat are wheat plants
C2: animals that have at least 2 legs and all foods they eat are plants

Does C2 subsume C1, i.e. C1 isa C2? Prove your answer using a tableau algorithm. (NOTE: Use the Baader, Nutt chapter in the reading list.)

Question 3: DB vs KB

Describe the difference between open-world assumption and closed-world assumption. Give examples.

Question 4: Ontology Engineering

(i) Give 4 different kinds of matchers for ontology alignment. For each kind of matcher give an example and explain briefly what it does.

(ii) Given the following axioms in an ontology: $A \sqsubseteq B$; $A \sqsubseteq E$; $B \sqsubseteq C$; $B \sqsubseteq D$; $C \sqsubseteq F$; $D \sqsubseteq F$; $F \sqsubseteq G$; $E \sqsubseteq G$; $A \sqsubseteq \neg G$.

Debug the ontology. Compute MIPS and MUPS.

Question 5. Semi-structured data

(i) Below, a (simplified) description of the biological data source PIR-PSD, is given. This data source contains information about protein sequences. Each entry in the data source has information about the entry (identification numbers, dates of creation and update), names of the protein sequence and the parts that it contains, the organism from which the sequence was taken, references to the literature describing the sequence, references to other related databases and the protein sequence. The simplified PIR-PSD schema is given as a DTD. Draw a strong data guide for PIR-PSD. Use the OEM model.

```
<!-- ************************************************************************
PIR-International Protein Sequence Database (PSD)
************************************************************************ -->

<!-- ProteinEntry: the root element. -->
<!ELEMENT ProteinEntry (header,protein,organism,reference*,sequence)>

<!-- header: database information. -->
<!ELEMENT header (uid,accession*,created_date,seq-update_date)>

<!ELEMENT accession (#PCDATA)> <!-- accession number -->
<!ELEMENT created_date (#PCDATA)> <!-- date (DD-MMM-YYYY) -->
<!ELEMENT seq-update_date (#PCDATA)> <!-- date (DD-MMM-YYYY) -->

<!-- protein: the protein-names. -->
<!ELEMENT protein(name,alt-name*,contains*)>

<!ELEMENT name (#PCDATA)> <!-- protein name -->
<!ELEMENT alt-name (#PCDATA)> <!-- alternate protein name -->
<!ELEMENT contains (#PCDATA)> <!-- activity name -->

<!-- organism: identification of the biological source. -->
<!ELEMENT organism (source,common-name?,note*)>

<!ELEMENT source (#PCDATA)> <!-- source name -->
<!ELEMENT common-name (#PCDATA)> <!-- common name -->

<!-- reference -->
<!ELEMENT reference (refinfo,note*)>

<!-- refinfo: identification of the literature source. -->
<!ELEMENT refinfo (authors,citation,title?,xrefs?)>

<!ELEMENT authors (author+)> <!-- list of authors -->
```

```
<!ELEMENT author (#PCDATA)> <!-- author name -->
<!ELEMENT citation (#PCDATA)> <!-- citation name -->
<!ELEMENT title (#PCDATA)> <!-- title text -->

<!ELEMENT xrefs (xref+)> <!-- cross-references -->
<!ELEMENT xref (db,uid)> <!-- a cross-reference -->
<!ELEMENT db (#PCDATA)> <!-- database tag -->

<!-- sequence: the amino acid sequence. -->
<!ELEMENT sequence (#PCDATA)> <!-- amino acid symbols and
 punctuation -->

<!-- General elements. Elements that can be contained in several
 other elements. -->
<!ELEMENT note (#PCDATA)> <!-- note text -->
<!ELEMENT uid (#PCDATA)> <!-- entry identifier -->
```

(ii) In figures 1 and 2 (simplified and slightly modified) data from a SwissProt data source is given. (The dashed nodes in the figures refer to nodes in the other figure. Both figures together make up the data source.) Draw a strong data guide for this data. Use the OEM model.

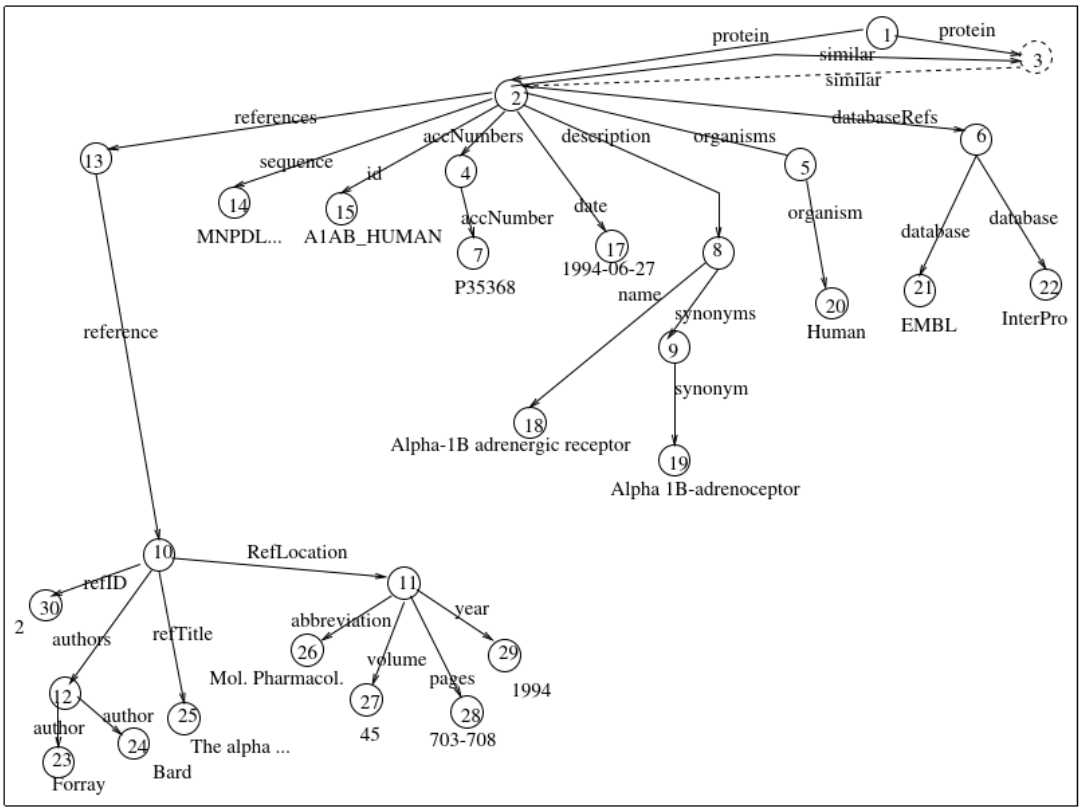

Fig. 1. SwissProt db - 1

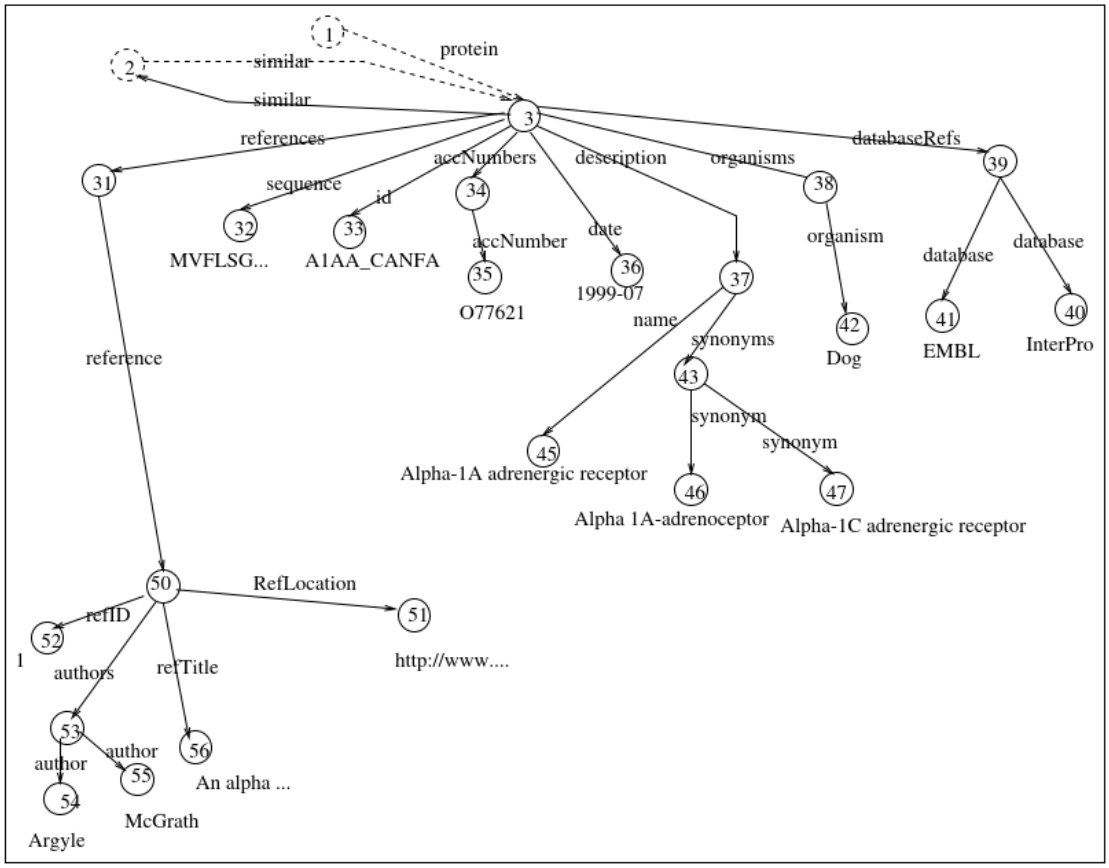

Fig. 2. SwissProt db - 2

Question 6. XML

Assume an XML document 'data.xml' that is valid for the DTD in Question 5(i).

(i) Write an XPath expression that selects the citation names of all literature sources that have a cross-reference to a database called DBLP.

(ii) Write an XPath expression that selects every reference of which the last author is Adam Smith.

(iii) Write an XQuery expression that lists the accession numbers of all protein entries that have the same creation date as some entry for the protein with accession number NM_000784. Accession number NM_000784 must not be in the list.

Question 7. RDF and SPARQL

Consider the following set of RDF triples (prefix declarations omitted).

```
ex:game_1 a ex:Game  .
ex:game_1 ex:publisher ex:Nintendo .
ex:game_1 ex:platform ex:Nintendo-Switch .
ex:game_1 ex:title "The Legend of Zelda: Tears of the Kingdom" .
ex:game_1 ex:release_date  "2023-05-12" .
ex:game_1 ex:mode ex:Single-Player .
ex:game_1 ex:series "The Legend of Zelda" .

ex:game_2 a ex:Game  .
ex:game_2 ex:publisher ex:Bandai-Namco-Entertainment .
ex:game_2 ex:platform ex:PlayStation-4 .
ex:game_2 ex:platform ex:PlayStation-5 .
ex:game_2 ex:platform ex:Windows .
ex:game_2 ex:platform ex:Xbox .
ex:game_2 ex:title "Elden Ring" .
ex:game_2 ex:release_date  "2022-02-25" .
ex:game_2 ex:mode ex:Single-Player .
ex:game_2 ex:mode ex:Multi-Player .

ex:game_3 a ex:Game  .
ex:game_3 ex:publisher ex:CD-Projekt .
ex:game_3 ex:title "The Witcher 3: Wild Hunt" .
ex:game_3 ex:release_date  "2015-05-19" .
ex:game_3 ex:mode ex:Single-Player .
ex:game_3 ex:series "The Witcher" .

ex:Nintendo                   ex:founded_date "1889-09" .
ex:Bandai-Namco-Entertainment ex:founded_date "2006-03" .
ex:CD-Projekt                 ex:founded_date "1994-05" .
```

(i) What is the result of evaluating the following SPARQL query (prefix declaration omitted) over the given set of RDF triples? Represent the result in a tabular form.

```
SELECT ?g WHERE {
  ?g a ex:Game .
  ?g ex:mode ex:Single-Player .
  ?g ex:mode ex:Multi-Player .
}
```

(ii) What is the result of evaluating the following SPARQL query (prefix declaration omitted) over the given set of RDF triples? Represent the result in a tabular form.

```
SELECT ?g ?s ?d WHERE {
  ?g a ex:Game .
  ?g ex:release_date ?d .
  OPTIONAL {
    ?g ex:series ?s .
  }
}
```

(iii) Write a SPARQL query for RDF data such as the triples given above to retrieve the URI of every game released after 2020.

(iv) Write a SPARQL query for RDF data such as the triples given above to retrieve games that belong to at least two platforms.

Question 8. NoSQL

Represent all of the information captured by the aforementioned set of RDF triples both as a key-value database and a document database.

Question 9. HDFS, Map/Reduce

(i) Explain how the Map/Reduce programming model works.

(ii) Explain how HDFS distributing files.

(iii) Assume we have distributed daily precipitation data of each city in Sweden for the year 2022, show how to write a map/reduce program to get the yearly precipitation for each city (write pseudo code, specify key-value formats at each step such as map and reduce). The example format of the data is as follows (in a tabular form).

```
Date       City        Precipitation (mm)
---------------------------------------
2022-01-01  Linköping       0.0
2022-01-01  Norrköping      0.5
2022-01-02  Linköping       0.3
...
```

Question 10. Graph databases

(i) Explain the two sophisticated graph partitioning methods and give examples for which they are suitable.

(ii) The diagram in Figure 3 shows a Google Pregel example for a graph processing problem with 3 supersteps. Please complete the diagram by specifying vertex states and communication during each superstep. (The number in a vertex means that the number is generated by the vertex's computation and will be sent as a message if needed in a superstep).

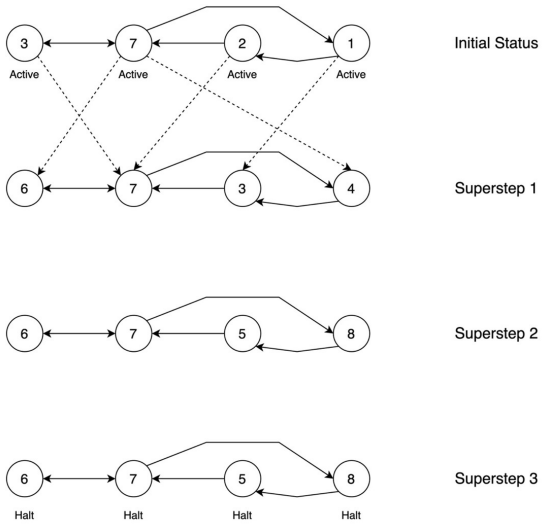

Fig. 3. Google Pregel example.

(iii) Explain in what cases, such a Google Pregel method has limitations.

Question 11. Integration of data sources

(i) *Global model*

Draw a global domain model based on your data guides from question 5. Use the information in figure 4 about which concepts in SwissProt match to concepts in PIR-PSD. Assume that concepts with the same name in the two data sources match. When labeling, give priority to SwissProt terminology.

(ii) *Global as view and local as view*

Discuss the global as view approach versus the local as view approach regarding the mappings between the global model and the content of the local sources as well as regarding query processing. *Exemplify* using examples based on the scenario in question 5 and question 11(a). (If needed, you may extend the scenario or the data source descriptions.)

SwissProt concept	PIR-PSD concept
id	uid
accNumbers	accession
date	created_date
description	protein
synonym	alt_name
refTitle	title
refLocation	citation
databaseRefs	xrefs
database	db

Fig. 4. Matching concepts.