MEDINFO 2004
M. Fieschi et al. (Eds)
Amsterdam: IOS Press

A Relation-Centric Query Engine for the Foundational Model of Anatomy

Landon T. Detwiler, Emily Chung, Ann Li, José L.V. Mejino Jr., Augusto V. Agoncillo, James F. Brinkley, Cornelius Rosse, Linda G. Shapiro

Structural Informatics Group, Departments of Biological Structure, Medical Education and Biomedical Informatics, and Computer Science and Engineering, University of Washington, Seattle, WA 98195

Abstract

The Foundational Model of Anatomy (FMA), a detailed representation of the structural organization of the human body, was constructed to support the development of software applications requiring knowledge of anatomy. The FMA's focus on the structural relationships between anatomical entities distinguishes it from other current anatomical knowledge sources. We developed Emily, a query engine for the FMA, to enable users to explore the richness and depth of these relationships. Preliminary analysis suggests that Emily is capable of correctly processing real world anatomical queries provided they have been translated into a constrained form suitable for processing by the query engine.

Keywords:

Anatomy, Information Retrieval, Knowledge Bases.

Introduction

Empowering software agents to reason in a complex domain requires a precise, complete, and logically organized knowledge representation, upon which accurate inference can be performed. One of the largest knowledge sources, within the anatomy domain, is the University of Washington's Foundational Model of Anatomy (FMA)[1]. The FMA is intended to facilitate the development of this sort of intelligent software agent.

With substantial portions of the FMA instantiated, the database backend now contains more than 1.5 million records. Computer-based tools are required to navigate and query these data. The querying of such a large and complex model, however, presents nontrivial challenges. Although a database query language for the FMA is evolving[2], it is not suitable for anatomists, most students, nor health care providers. A prototype for a natural language interface to the FMA has also been developed[3], however, it does not yet constrain user queries to those requiring only anatomical knowledge retrievable from the model. We developed *Emily*, a relation-centric[1] graphical query engine for the Foundational Model of Anatomy, to circumvent these difficulties.

Our objectives are to describe *Emily*, illustrate the kinds of queries it can handle, and present a preliminary evaluation of the inference engine. Before addressing these objectives, however, we briefly introduce the FMA and discuss two complementary approaches for its navigation.

Foundational Model of Anatomy

The FMA is a formal description of the structural organization of the human body. Built using the Protégé-2000 knowledge representation system[4], the FMA is a frame-based ontology. Its backbone is the *Anatomy Taxonomy* (AT), an inheritance class subsumption hierarchy for anatomical entities. The subclass/superclass relations represented in the AT are just two of 167 relationship types currently contained in the FMA. Relationship types are instantiated to assign attributes to the anatomical concepts of the AT, and define the structural relationships between them. These relationships, represented as slots of a concept frame, constitute the *Anatomical Structural Abstraction* (ASA) component of the FMA. Some examples of ASA relationships that interconnect anatomical entities are *boundary, part, branch, tributary, containment, connection, continuity, attachment,* and *adjacency. Emily* was specifically designed for exploring these inter-entity relationships.

Navigating the FMA

The Foundational Model of Anatomy knowledge base can be viewed as a complex, highly connected, network in which nearly 70,000 anatomical concepts (a subset of the over 180,000 frames) are interrelated by over 580,000 relationship instances (a subset of the over 1.5 million relationship slot values, the remainder of which link anatomical concepts to primitive values such as textual strings or Boolean values). The concepts correspond to the nodes of the network and the relationships between concepts form the edges connecting these nodes. For example, `Abdominal cavity` and `Small intestine` are two nodes of the network, and the relationship *contains* is a directed edge from `Abdominal cavity` to `Small intestine`. Given the complexity of this network, the efficient retrieval of information becomes a major issue.

Browsing, in the FMA, is concept-centric. Users can only retrieve information on a per concept basis, and navigation is constrained by the model's inherent semantic structure. Because the FMA is a frame-based model, every concept frame is fully spec-

---

[1] We define relation-centric queries as those pertaining to relationships between first-class objects. In the case of the FMA, such first-class objects are restricted to classes of anatomical structures.

ified by a collection of slot values. Therefore, by browsing the FMA, it is possible to retrieve all of the information that is represented *explicitly* in the model. Such a task, however, is laborious, particularly by a user not familiar with the anatomy domain and the model itself.

Querying, an alternative mode of information retrieval, allows users to interrogate the model unconstrained by its organizational structure. Does this free the user from the need to be familiar with the model? Not entirely. If we present users with a natural language query interface, allowing them to pose arbitrary queries, the results will likely be frustrating because the queries are bound to contain concepts and relationships not represented in the FMA. We developed *Emily* with the objective of eliminating such frustrations by constraining the concepts and relationships that can be incorporated in a query. We illustrate with examples that such a constrained approach allows for the generation of not only direct answers, but also query results that are *not represented explicitly* in the knowledge base.

We focus on the set of *relation-centric queries* (a subset of all possible queries), whose elements are queries pertaining to the relationships between concepts. Such a relation-centric query is "What is the relationship between the small intestine and the wall of the duodenum?" A user, attempting to browse for the answer to this question, might start from the frame of the `Small intestine` and then navigate through many nodes before finding the `Wall of duodenum`. When the `Wall of duodenum` has been reached, the user will have to remember the path he or she navigated in order to answer the original question. By contrast, submitting this question through the *Emily* query engine generates the result: "`Small intestine` *has part* `Duodenum`, which *has part* `Wall of duodenum`" without further user navigation.

Emily

We refer to *Emily* as relation-centric because it is specifically intended to explore the structural relations between anatomical entities. We first provide an overview of *Emily's* implementation, followed by the kinds of queries that *Emily* can process, and then illustrate the graphical interface for submitting these queries and obtaining their results.

Implementation

Emily is built as a Java application on top of the Protégé 2000 knowledge-base library (Figure 1). *Emily* enables users to easily construct suitably constrained queries to the FMA, through its graphical user interface (Figure 2). The *Emily* engine translates these queries into method calls to the Protégé 2000 library. The Protégé 2000 library methods perform all of the FMA database information retrieval. *Emily* then translates the returned results into the appropriate form for display, to the user, within its interface. Users can pose *basic queries* that involve a single structural relationship between two anatomical entities and *compound queries* that involve more than one relationship.

Basic queries

A basic query is a triplet of the form: (Subject, Relation, Object). The Subject and Object elements can be *Unknown* or selected from the Anatomical Taxonomy. Likewise, the Relation element can be *Unknown*, or selected from the structural relationships of the ASA. Because most of the relations have transitive closures, there are two possible forms for each relationship: direct and closure. For example, the *continuous with* relationship allows the user to submit queries for *is continuous with (directly)*, which would return only those entities directly continuous with the given entity, and for *is continuous with*, which would return those entities continuous with the given entity in the closure sense. The latter set refers to the entities directly continuous with the given entity, the entities with which each of those entities are continuous, and so on, all the way to the terminus of a continuous structure. Users can specify all three fields of a query, or choose any field to leave as *Unknown*, but a query cannot contain more than one *Unknown* or the query would not be sufficiently specified.

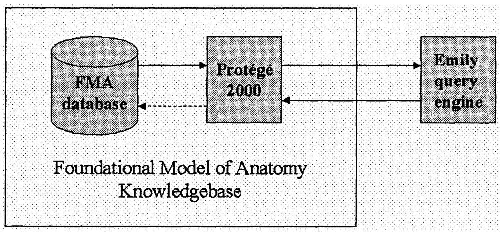


Figure 1 - A simplified view of Emily's architecture

The basic form allows for a variety of queries, for example:

1. `Ileum` *is continuous with (directly)* Unknown
2. `Ileum` *is continuous with* Unknown
3. Unknown *is contained in* `Abdominal cavity`
4. `Ileum` *is part of* `Small intestine`
5. `Small intestine` Unknown `Gut`
6. `Abdominal cavity` Unknown `Ileum`

Queries 1-3 each have one *Unknown* and should return a set of zero or more anatomical entities, each of which is a valid substitution for the *Unknown*. For queries with no *Unknown*, such as query 4, *Emily* returns simply *yes* or *no* as appropriate (*yes* in the case of query 4). Query 5 contains an unknown relationship between `Small intestine` and `Gut`. The response should be that `Small intestine` *is part of* `Gut`. Query 6 asks for the unknown relationship between the `Abdominal cavity` and the `Ileum`. This query requires *Emily* to navigate through the database in order to produce the response that the `Abdominal cavity` *contains* `Small intestine`, which *has part* `Ileum`.

Compound queries

In addition to basic queries, *Emily* allows users to pose questions involving more than one relationship. These compound queries can be formulated in two different ways:

1. Sets of Linked Queries

Queries may generate or use variables whose values are sets of anatomical entities. Two queries can be linked by a common variable. For example, the query "Unknown *is part of (directly)* `Small intestine`" generates the set {`Wall of small intestine`, `Lumen of small intestine`, `Duodenum`, `Jejunum`, `Ileum`}, which is then automatically assigned to a variable, such as *U1*. For a second query, "U1 *is continuous with (directly)* Unknown", *Emily* searches for the *is continuous with (directly)* relationship between each element of *U1* and other anatomical entities. The result is a tree structure that contains elements of U1 at the top level and the entities that satisfy the query as children of these top-level entities. For the above query, the tree structure (in which the indention denotes *continuous with*) contains the following information:

```
Wall of small intestine
    Wall of stomach
    Wall of large intestine
Lumen of small intestine
    Cavity of stomach
    Lumen of large intestine
Duodenum
    Pylorus
    Jejunum
Jejunum
    Duodenum
    Ileum
Ileum
    Jejunum
    Cecum
```

In addition to displaying this result tree, *Emily* creates another variable, *U2*, to which it assigns the set {`Wall of stomach`, `Wall of large intestine`, `Cavity of stomach`, `Lumen of large intestine`, `Pylorus`, `Jejunum`, `Duodenum`, `Ileum`, `Cecum`}, representing the set of leaves in the tree.

2. Boolean Combinations of Queries

Sets of results produced by several different queries can be combined via Boolean expressions. For example, the query

Unknown *is continuous with (directly)* `Ileum`
AND NOT
Unknown *is part of (directly)* `Small intestine`

executes the two separate queries and combines their results to produce a set containing the single element `Cecum` which is continuous with the `Ileum`, but is not a part of the `Small intestine`.

Unknown relation queries

Queries regarding an unknown relationship require special consideration because there may be a vast number of indirect connections between any two anatomical concepts. Because *Emily* cannot return all relational sequences connecting two concepts, it returns the first one it finds according to a three-fold search strategy (we assume that the most direct relation is the most desirable). *Emily* will first search for direct and closure relationships. Next, it searches for pre-defined relational sequences determined to be significant. One such sequence is identified by the following regular expression, *(has part)*contains*, which indicates a sequence of any number of *has part* relationships followed by a single *contains* relationship. Finally, if no direct, closure, or pre-defined sequence is found, *Emily* will resort to a depth-limited breadth–first search. Our current system searches all possible chains containing up to 4 relationships before abandoning the search. Although this limit was chosen for reasons of computational complexity, in practice relational chains longer than this are seldom informative.

The graphical user interface

We have developed a graphical user interface for *Emily*, shown in Figure 2, which gives users a simple way to enter both basic and compound queries. Basic queries are entered using the top portion of the interface, which is divided into three sections corresponding to the three fields of a query (Subject, Relation, Object). In both the Subject and Object sections the user can select anatomical entities from hierarchical trees. These trees contain the terms for all anatomical entities in the Anatomy Taxonomy. The desired ASA relationship can be selected from the center section. Each of the three sections provides an *Unknown* option. Once the query is specified, a *Query* button initiates the processing of the query by the *Emily* engine. Queries are translated into appropriate calls to the Protégé API and the results are formatted for display in the lower portion of the interface.

There are a number of other useful features included in the *Emily* interface. Users can either browse for an anatomical term or type one in and allow the system to locate it for them. In the latter case, *Emily* has the capacity to translate common synonyms, in English or Latin (or in limited cases other non-English languages), into the FMA's preferred concept name. Users can see a history of all the queries from a given session, and they can click on the result sets to review previous results. These features, along with its query processing capabilities, are likely to make *Emily* a useful tool for querying the Foundational Model of Anatomy and for learning about basic anatomical relationships.

Reliability of Emily

In order to evaluate the *Emily* query engine, we tested whether it could correctly process queries we selected from two published compendia of anatomy exam questions[5,6]. We restricted the questions to structurally-based multiple choice items and excluded those questions that require functional or developmental knowledge, not presently included in the FMA. The following is typical of the selected questions: "The coronary sinus receives each of the following vessels EXCEPT the: **a**. Great cardiac vein; **b**. Middle cardiac vein; **c**. Anterior cardiac vein; **d**. Small cardiac vein; **e**. Posterior vein of the left ventricle." This question can be transcribed into a single query: "`Coronary sinus` *has tributary (directly)* Unknown" for which *Emily* returns the result set {`Great cardiac vein`, `Posterior vein of left ventricle`, `Middle cardiac vein`, `Small cardiac vein`, `Oblique vein of left atrium`}, indicating choice **c** as the exception. *Emily* consistently returned results corresponding to the published key, for all questions which could be translated into the required form, provided the required anatomical entities and relationships were present in the FMA. Questions that could not be translated into the triplet form included those that were not relation-centric (i.e. "What color is blood?") and those involving reified relationships.[1] While the FMA contains reified relationships, handling them is left as a future augmentation of *Emily*.

Following this useful, but somewhat ad hoc test, we performed a more formal evaluation of *Emily's* reliability. We considered the first 100 questions from each of the 1st seven chapters from a published source of anatomy questions[5]. From these 700 questions, we again excluded those requiring knowledge outside of the domain covered by the FMA. This reduction yielded 412 candidate questions. From these candidate questions, 100 were chosen, at random, for processing by the *Emily* query engine. The chosen questions were translated into *Emily* queries by domain experts.

Those questions which *Emily* could not answer were grouped according to the reasons for their failure. Many such failures were a result of the incompleteness of the underlying anatomy model. Since the evaluation of the FMA is outside of the scope of this report, we shall not discuss it here. However, it is important to note that the failures were not caused by deficiencies in the *Emily* application. We postulated that *Emily* could correctly answer these questions provided the appropriate data were entered into the FMA. Therefore, we selected several such failed queries, entered the missing data, and then successfully posed them to the *Emily* query engine. This procedure suggested that our postulate was correct; the failures of only 10% of the randomly selected queries could be directly attributed to *Emily.* In each of these cases, the cause of the failure was the same, *Emily's* inbility to handle reified relationships.

Query efficiency

In order to evaluate *Emily's* performance, we chose a set of 10 representative queries, of varying degree of difficulty, and timed their evaluation. All efficiency tests were performed with both the *Emily* application and a local MySQL DBMS, containing the FMA database, running on a 2.60GHz Intel Pentium 4 PC with 1GB of RAM. Each query was chosen because it is indicative of the efficiency of a particular class of queries.

The evaluation queries were repeated multiple times. Simple queries as well as Boolean combinations of simple queries generally took less than a second (one execution of one of the simple evaluation queries took 2.67 seconds). Unknown relation queries for which the solution could be reached via a direct relationship or via the transitive closure of a single relationship were also very quick, in all cases executing in less than 3 seconds. However, unknown relation queries that required *Emily* to resort to a graph search were significantly slower. In the worst case *Emily* took as long as 3 minutes and 19 seconds to evaluate a query[1].

A notable observation from the performance evaluation of unknown relation queries is that execution time is dependent on the branching factor of the subject entity, but not of the object entity. This is significant because two similar queries with their subjects and objects transposed could have significantly different execution times.

Discussion and conclusion

We have shown in the initial evaluation of *Emily* that it can process queries from real world anatomical discourse, provided the questions are first translated into the "subject, relation, object" form. This translation also requires describing anatomical concepts and relationships in terms consistent with those represented in the FMA. The search function helps with this translation by utilizing the FMA's *synonym* and *non-English equivalent* attributes. A shortcoming of the *Emily* program, highlighted by our evaluation, is its inability to handle reified relationships. This extension is slated as a future augmentation to the *Emily* interface.

Emily's performance is quite efficient in most cases. All evaluation queries, for which the relation element was specified, were answered within a few seconds. While unknown relation queries require more complex graph traversal and may take much longer to answer, over 3 minutes for our worst evaluation case , the processing time is limited by the branching factor of the tree and our maximum depth limit.

Emily's ability to return correct answers to anatomy exam questions suggests that it will be a useful query tool. Furthermore, incorporation of evolving versions of *Emily* into educational applications promises to raise the quality of computer-based anatomy programs to a cognitively higher level. Although such programs are currently rich in image-based information, symbolic information is limited to image annotations and English language text[7].

Acknowledgements

This work was funded by National Library of Medicine grants LM06822 and LM06316.

References

[1] Rosse C, Mejino JVL. A reference ontology for bioinformatics: the Foundational Model of Anatomy. *J Biomed Inform* 2003;36:478-500.

[2] Mork P, Brinkley JF, Rosse C. OQAFMA querying agent for the Foundational Model of Anatomy: providing flexible and efficient access to a large semantic network. *J Biomed Inform* 2003;36:501-517.

[3] Distelhorst G, Srivastava V, Rosse C, Brinkley JF. A Prototypical Natural Language Interface to a Complex Knowledge Base: the Foundational Model of Anatomy. *Proceedings AMIA* 2003; 200-204.

---

1. An example of a question involving a reified relationship is the following, "Is the heart continuous with, posteriorly, the left superior pulmonary vein?" In this sentence the adverb 'posteriorly' applies to the continuity relationship, making it more specific. *Emily* can identify that the left superior pulmonary vein is continuous with the heart, but it cannot currently make use of the additional coordinate attribute on the relationship.

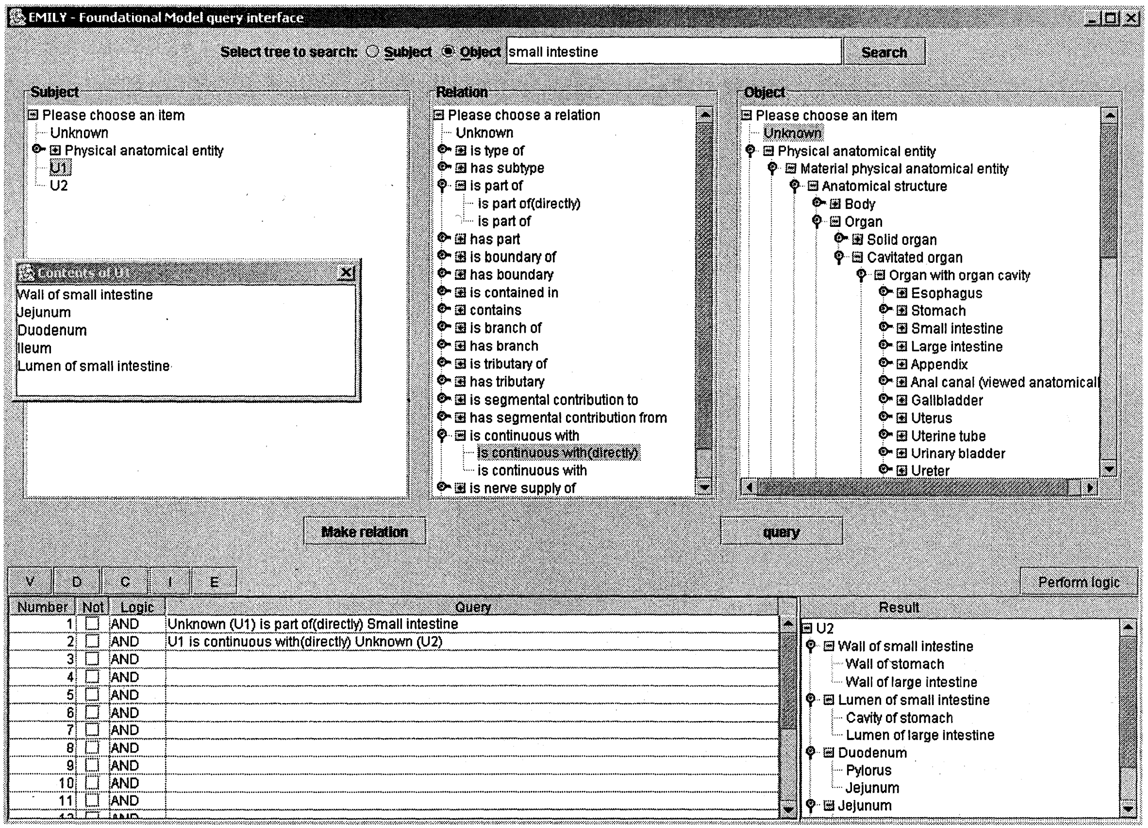
1. The worst case scenario for an unknown relation query occurs when there is no relationship found. This case requires the maximum amount of graph search that *Emily* allows. The actual size of this search space depends on the branching factor of the subject entity and a maximum depth limit presently set to allow 4 levels of graph traversal

[4] Noy NF, Fergerson RW, Musen MA. The knowledge model of Protégé 2000: combining interoperability and flexibility. *Proceedings of the 12th International Conference on Knowledge Engineering and Knowledge Management* (EKAW-2000); Juan-les-Pins France, Springer; 2000.

[5] Fitzgerald MJT, Golden JP. *Anatomy 1600 Multiple Choice Questions*. London: Butterworth & Co. LTD.; 1973.

[6] Wilson JL. *Anatomy 700 Questions: A USMLE Step I Review*. 10th Edition. Connecticut: Appleton and Lange; 1995.

[7] Kim S, Brinkley JF, Rosse C. A profile of on-line anatomy information resources: design and instructional implications. *Clin Anat* 2003;16:55-71.



small intestine

Contents of U1

Wall of small intestine
Jejunum
Duodenum
Ileum
Lumen of small intestine

Number	Not	Logic	Query
1	☐	AND	Unknown (U1) is part of(directly) Small intestine
2	☐	AND	U1 is continuous with(directly) Unknown (U2)
3	☐	AND	
4	☐	AND	
5	☐	AND	
6	☐	AND	
7	☐	AND	
8	☐	AND	
9	☐	AND	
10	☐	AND	
11	☐	AND	

Result

- U2
  - Wall of small intestine
    - Wall of stomach
    - Wall of large intestine
  - Lumen of small intestine
    - Cavity of stomach
    - Lumen of large intestine
  - Duodenum
    - Pylorus
    - Jejunum
  - Jejunum

Figure 2 - The Emily graphical user interface after processing the queries Unknown is part of (directly) Small intestine (the results of which form the set UI) and UI is continuous with (directly) Unknown (U2). In the lower right-handcorner the result set, U2, is shown as a tree, in which elements of UI form the first layer, and the continuities of each are displayed as indentations. The figure also shows a small pop-up window for reviewing the results of a previous query (in this case the set UI)