A Prototype Natural Language Interface to a Large Complex Knowledge Base, the Foundational Model of Anatomy

Gregory Distelhorst, **Vishrut Srivastava, Cornelius Rosse, MD, DSc,**
and **James F. Brinkley, MD, PhD**

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

ABSTRACT

We describe a constrained natural language interface to a large knowledge base, the Foundational Model of Anatomy (FMA). The interface, called GAPP, handles simple or nested questions that can be parsed to the form, subject-relation-object, where subject or object is unknown. With the aid of domain-specific dictionaries the parsed sentence is converted to queries in the StruQL graph-searching query language, then sent to a server we developed, called OQAFMA, that queries the FMA and returns output as XML. Preliminary evaluation shows that GAPP has the potential to be used in the evaluation of the FMA by domain experts in anatomy.

INTRODUCTION

Recent work in knowledge representation has generated knowledge bases of considerable size. Representation, however, is only half of the problem. This knowledge must be made available to other agents, human and otherwise. The goal of the work described in this report is to create a prototype interface for one such knowledge base: the Foundational Model of Anatomy (FMA)[1]. We describe a program, named GAPP, which takes natural language (NL) questions as input, translates them into an existing query-language, and returns the result of the query as an XML document.

Although NL question-answering is still in the research stage[2], a semantically constrained domain of inquiry can greatly simplify the problem[3]. GAPP is constrained by 1) only providing access to knowledge that is explicitly represented in the FMA, 2) providing a bare minimum of inferential ability, 3) assuming that all sentences directed towards GAPP are questions, and 4) focusing on what seem to be the most intuitive way to ask such questions.

In the remainder of this paper we present the basic features of GAPP. We then illustrate the kinds of questions GAPP can process, and evaluate its performance. Our preliminary evaluation shows that, in spite of the above constraints, GAPP is able to process a sufficiently rich set of NL questions that it is likely to be useful as an interface for domain experts when evaluating the FMA.

GAPP AND THE FMA

The FMA is a frame-based ontology that represents declarative knowledge about the structural organization of the human body. Its *Anatomy Taxonomy* links over 67,000 anatomical concepts into a single class subsumption hierarchy through the *isa* relationship. The *Anatomical Structural Abstraction* (ASA), a second component of the FMA, relates these concepts to one another through more than 1.4 million instances of over 100 kinds of structural relationships.

GAPP is a question-answering system developed for querying the FMA. It is built as an adaptation of Apple Pie Parser (APP), a program for parsing English sentences into their syntactic structures[4]. GAPP was designed to process questions that resemble the queries accepted by *Emily*, another application we have developed through which structured queries can be submitted to the FMA via a graphical user interface[5]. In *Emily*, graphical queries consist of three elements: Subject, Relationship, and Object, any one of which can be unknown. In GAPP, an English language sentence is resolved into these three regions. To accomplish this, GAPP exploits the regularities of English syntax, along with pattern-matching phrase and word combinations that might define the relationship component. This is a three-step process: syntactic parse, pattern-matching, and construction of the query. APP generates the syntactic structure of each incoming question. For any English sentence, APP returns a parenthesized and labeled syntactic structure. For example, the sentence “What gives blood to the prostate?” returns:

```
(S (NP what) (VP gives (NP blood) (PP to (NP the prostate))) ?)
```

Note that APP gives a simplified syntactic structure, which GAPP simplifies even further. GAPP converts adjectival phrases and adverbial phrases to prepositional phrases (PP). Although linguistically unnatural, this simplification is quite useful. We rarely need to discern between these types of phrases, and they often occur in the same position in similar questions.

The syntactic structure of the sentence allows GAPP to break the question into three parts. First, the top-level noun-phrase (NP) is marked as an anatomical entity. GAPP decides later whether this NP is a known or unknown entity. GAPP next assumes that the relationship begins at the sentence-level verb-phrase (VP). To decide where the relationship region ends, GAPP uses a user-defined file: the parse dictionary. Using the complete VP as its search term, GAPP reads this dictionary to see how much of the structure can be considered part of the relationship between entities (without crossing into the other anatomical entity). Once GAPP defines the relationship region, it marks the remaining NP as the second anatomical entity. The parse dictionary entry for the VP above looks like this:

```
<1-VP>
:give, gives
  <2-NP>
  :blood, arterial supply, blood-supply = PP [ART SUPPLY] 2
  :motor supply, primary motor supply = PP [MOTOR SUPPLY] 1
  :segmental contribution = PP [SEG CONTRIBUTION] 2
  </2-NP>
</1-VP>
```

The bracketed tags represent phrase labels in the structure; <1-VP> denotes a top-level verb-phrase. The comma-separated list of words following the colon lists search terms for the VP. Here, GAPP matches on the word “gives”. Next, GAPP finds the label for an embedded NP: <2-NP>. It then searches for and finds the word “blood” inside of that NP. There are no additional phrase labels within the entry for “blood”. This tells GAPP that it has reached the end of the relationship region of the question. The search returns the entry: PP [ART SUPPLY] 2.

What does this entry mean? The PP indicates that the last anatomical entity is contained in a PP. The entry [ART SUPPLY] is the semantic translation of the question; it corresponds to a relationship of arterial supply within the FMA. The particulars of this query are stored in another user-defined file. The number 2 is a binary variable (1 or 2), indicating the directionality of the relationship between the top-level NP and the NP located after the question-region. Now the sentence has been divided into three regions:

(S	(NP What) Entity 1	(VP gives (NP blood) (PP to Relationship	(NP the prostate) Entity 2	))) ?)

Next, GAPP distinguishes the known anatomical entity from the unknown one. This is a simple pattern-match on “what” or “which” in each NP. This gives the final structure:

(S	(NP What) Unknown	(VP gives (NP blood) (PP to Relationship	(NP the prostate) Known	)))?)

The final step of the process is query-generation. GAPP was designed to convert questions into database queries in the StruQL query language, a database language designed specifically for queries over graphs. Each StruQL query consists of a WHERE clause that generates a set of variable bindings and a CREATE clause that outputs data generated in the WHERE clause. GAPP generates StruQL queries from templates provided in another user-defined file, the relationship dictionary. An entry for the relationship [ART SUPPLY] might be:

```
rel: [ART SUPPLY]
1
in: [KN]->"Preferred name"|"Synonyms"."name"|"Latin name (TA)"->[IN],
query: [KN]->"supplies"->[UNK],
out: [UNK]->"Preferred name"."name"->[OUT],
/1

2
in: [KN]->"Preferred name"|"Synonyms"."name"|"Latin name (TA)"->[IN],
query: [KN]->"arterial supply"->[UNK],
out: [UNK]->"Preferred name"."name"->[OUT],
/2
```

The entry is headed by [ART SUPPLY], denoting the FMA relationship of arterial supply. Following this are open and close tags for 1 (recall that relationships are directional—1 or 2 indicates which side of the relationship is known and which unknown). The next three elements represent different sections of the query itself. The line preceded by in: is an expression dealing with the input from the question, query: finds entities fulfilling a certain relationship to the input, and out: produces the names of the output concepts. These lines are separated largely for the use of nested queries (see Nested Queries section). Note also that each line begins and ends with a bracketed variable-name. These are virtual variables, representing known entity [KN], unknown entity [UNK], input text [IN] and output text [OUT]. These variables are also used for the generation of complex queries.

While the entries for 1 and 2 look quite similar, the query: line differentiates them. Query 1 uses the relationship “supplies”, while query 2 uses “arterial supply”. These are inverse relationships in the FMA;

that is, query 1 looks for an organ that receives arterial supply from the known entity, while query 2 searches for an artery that supplies the known entity.

From the relationship dictionary entry, GAPP assigns real variable names to the virtual variables. To do this, it searches backward from output variables, attempting to find the relationship that created the output. GAPP finally puts these variables into a CREATE clause, the part of the query that generates output:

```
WHERE
Y->"Preferred name"|"Synonyms"."name"|"Latin name (TA)"->"Prostate",
Y->"arterial supply"->X,
X->"Preferred name"."name"->arterial_supply,
CREATE
Output(arterial_supply);
```

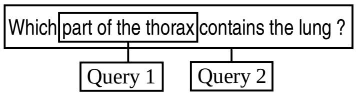
GAPP names the output variable the same as the query relationship, in this case “arterial_supply”. Note that the line containing “Prostate” has a complicated search path. This is because the FMA stores many names for each anatomical entity. The search path above checks preferred name, synonyms, and Latin names for the term “Prostate”.

This query is then sent to OQAFMA, a server we developed that supports StruQL queries to the FMA[6]. The above query gives the following output:

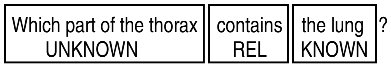
```
<results>
  <Output>
    <arterial_supply>Prostatic part of left middle rectal
artery</arterial_supply>
  </Output>
  <Output>
    <arterial_supply>Prostatic part of right middle rectal
artery</arterial_supply>
  </Output>
  <Output>
    <arterial_supply>Prostatic part of left inferior gluteal
artery</arterial_supply>
  </Output>
  <Output>
    <arterial_supply>Prostatic part of right inferior gluteal
artery</arterial_supply>
  </Output>
  <Output>
    <arterial_supply>Prostatic part of left inferior vesical
artery</arterial_supply>
  </Output>
  <Output>
    <arterial_supply>Prostatic part of right inferior vesical
artery</arterial_supply>
  </Output>
</results>
```

A web interface to GAPP is available at: http://sig.biostr.washington.edu/projects/gapp/. In response to a user question typed in a web form the interface outputs both the generated StruQL query and the query results.

Nested Question Processing. While many queries resemble the ones above, others span multiple relationships in the FMA. For example: “Which part of the thorax contains the lung?” This query uses two relationships to generate the output set: “part of” and “contains”. Such queries require more processing of the input—they can be considered nested queries:



This question is modeled in GAPP by performing the internal query (Query 1) first. The output of the internal query then serves as the input for the top-level query (Query 2). GAPP creates such a query by generating the top-level query, and revising it after encountering the internal query. For its initial analysis, GAPP would determine the following:



This generates a partial query:

```
Y->"Preferred name"|"Synonyms"."name"|"Latin name (TA)"->"Lung",
Y->"contained in"->X,
X->"Preferred name"."name"->[OUT],
```

This query is not yet complete—GAPP still needs to constrain the output to the parts of the thorax. To complete the query, GAPP further analyzes the NP “Which part of the thorax”. It looks for substructure that indicates a query within the NP. This module finds the phrase “part of”, and generates a second partial query for the NP:

```
0 M->"Preferred name"|"Synonyms"."name"|"Latin name (TA)"->"Thorax",
1 M->"generic part"+->X,
2 X->"Preferred name"."name"->[OUT],
```

The two partial queries are combined to give the final query:

```
WHERE
Y->"Preferred name"|"Synonyms"."name"|"Latin name (TA)"->"Lung",
M->"Preferred name"|"Synonyms"."name"|"Latin name (TA)"->"Thorax",
Y->"contained in"->X,
M->"generic part"+->X,
X->"Preferred name"."name"->contained_in,
CREATE
Output(contained_in);
```

The results returned by OQAFMA are:

```
<results>
  <Output>
    <contained_in>Thoracic cavity</contained_in>
  </Output>
</results>
```

PRELIMINARY EVALUATION

As designed, GAPP should be capable of answering all questions of the format Unknown-Relationship-Known provided that the appropriate entries exist in the two user-defined dictionaries. We tested GAPP with a battery of 198 questions covering varied grammatical forms. These questions were designed by developers of GAPP and the FMA, working backwards from the relationships represented in the FMA. That is, we tested the questions that GAPP was designed to answer.

GAPP's response was considered correct if it generated the correct StruQL query, not if the StruQL query returned the correct results. GAPP can only ask the correct question of the FMA—it has no knowledge of anatomy itself. The FMA is an evolving resource, and some relationships remain unmodeled. Many questions were scored as correct for generating the correct query, even when the FMA did not provide correct results. The GAPP web interface provides both the StruQL query and results of the query, allowing evaluators to examine GAPP's performance independent of the FMA. At the time of writing all of the example questions provided in this paper both generated the correct query and returned correct results.

GAPP's queries were evaluated by anatomists from the Structural Informatics Group. Judging the queries was fairly unambiguous, since all questions were very specific, and limited to the domain of physical anatomical relationships. GAPP's performance was divided between simple and nested questions.

Simple Questions. GAPP is very successful on simple questions. Simple questions are those that have one known entity, one unknown entity and one relationship between them. This includes questions in the active voice, such as "What gives blood to the right lung?" GAPP generated the correct query for 47 of 52 of these questions. The errors were due to erroneous parses from Apple Pie Parser, such as: "The biceps brachii gives motor supply to what?"

```
(S (NPL the biceps brachii) (VP gives (SBAR (SS (NPL motor) (VP supply
(PP to (NPL what)))))) ?)
```

Simple "is" Questions. These questions are grouped together, as their syntactic parses from APP are most similar. These include questions like: "What is the muscular origin of the flexor carpi ulnaris?" For 66 of 69 questions, GAPP generated the correct query. The three problematic sentences were again caused by erroneous parses by Apple Pie Parser.

Simple Wh-movement Questions. These questions involve syntactic movement of a question-word to the front of the sentence, such as: "What is the right lung surrounded by?" These questions are markedly more difficult for GAPP, causing problems for 11 of the 33 questions in this battery. Unlike previous problems with syntactic parsing, these problems are caused by limitations of GAPP. It presently lacks the ability to parse certain questions with Wh-movement from inside an NP or PP (which, in GAPP, includes adjectival and adverbial phrases as well).

```
(S (NP What) (VP is (NP the heart) (NP (NP a part) (PP of)) ) ?)
(S (NP What) (VP is (NP (NP the heart) (PP adjacent (PP to)) )) ?)
(S (NP what) (VP are (NP (NP the lungs) (PP posterior (PP to)) )) ?)
```

Nested Questions. Nested questions require multiple FMA relationships to be answered. These come in several formats. For nested questions, GAPP generated correct queries for 34 of 35 example sentences. This spans many formats of nested question. Some have an embedded Complement Phrase headed by "that": "What is the branch of the celiac trunk that gives blood to the stomach? "

```
(S (NP What) (VP is (NP (NP the branch) (PP of (NP the celiac trunk)))
(SBAR that (SS (VP gives (NP blood) (PP to (NP the stomach)))))) ?)
```

Such "that" questions can often be rephrased so the embedded query is located in an NP: "Which branches of the celiac trunk give blood to the stomach?"

```
(S (NP (NP Which branches) (PP of (NP the celiac trunk))) (VP give (NP
blood) (PP to (NP the stomach))) ?)
```

Nested questions can also have Wh-movement. As with simple questions, these are also markedly more difficult for GAPP. However, because of the problems for simple questions, the question battery lacks any questions that will obviously cause failure (see the above examples of problems for simple questions). Questions already known to fail are omitted from testing.

Constrained-NP Nested Questions. In the final set of questions the queried element is constrained to some category: bone, artery, organ, etc. The seven tested questions all generated the correct query, although the OQAFMA server was not always able to process them. Some examples that produced answers from the FMA at the time of this writing were: "Which arteries supply the stomach?" and "Which bones are part of the skull?"

Response Time. The time to retrieve XML results for a question from the question battery was approximately 5-10 seconds. However, for arbitrary queries GAPP's queries occasionally required a long processing time (>>10 seconds), even though generation of the StruQL query is very quick (<30 ms). One reason for the lengthy retrieval times is that GAPP queries always search multiple name paths in the knowledge-base, such that anatomical entities can be indexed by their synonyms, Latin names, etc. This query is the union of a few large database relations, which increases the processing time. We are exploring ways to reduce the response time.

DISCUSSION

In its current state, GAPP has the potential to offer a convenient way for anatomists familiar with its limitations to evaluate the FMA. In order to extend its capabilities, we need to generate a large quantity of potential end-user questions. These questions would guide the addition and editing of GAPP's dictionary-entries. That is, they would at once extend GAPP's coverage of varied questions and test whether the program is flexible enough to adapt to the many ways people use language. The best standard for the program will ultimately be how useful a naïve user finds it. To this end, the GAPP web interface keeps a log of all user queries.

GAPP needs a slightly more robust syntactic parser. Since GAPP is written for the output formatted by APP, we might take a more consistent parser and reformat its output to match APP. This would fix many of the bad parses in the input data.

One clear concern is that of scalability. Can GAPP ever be complex enough to cover the huge range of questions of a naïve user? With regard to anatomical terminology, GAPP benefits greatly from the FMA. If they exist, the FMA includes synonyms and Latin equivalents of anatomical terms. Thus, GAPP need not translate "Gullet" to "Esophagus" or "Cor" to "Heart"; the FMA contains this information itself.

Perhaps the biggest improvement to GAPP would be to provide more user feedback for failed questions. Both lay-users and evaluators of the FMA would benefit by knowing whether a query failed because GAPP generated an incorrect query or because the FMA does not yet model the queried information.

Even in its current state of development our preliminary evaluation suggests that GAPP offers a method for anatomy experts to evaluate the evolving content of the FMA. Evaluation is an integral part of developing the FMA, and doing so requires the input of many anatomists. By allowing natural language queries to the FMA, GAPP provides an intuitive way for these experts to browse the knowledge of the FMA. Although NL interfaces to databases in general have not yet been successful[7], we believe that the highly constrained nature of this particular application should allow it to be of practical use, not only for the anatomy domain, but also for other constrained areas of knowledge representation.

ACKNOWLEDGMENTS

Supported by NIH grants LM06822 and DC/MH02310, and contract LM03528.

REFERENCES

1. Rosse C, Mejino LVJ. A reference ontology for bioinformatics: The Foundational Model of Anatomy. J Biomed Inform. 2003 In press
2. Katz B. From Sentence Processing to Information Access on the World Wide Web. AAAI Spring Symposium on Natural Language Processing for the World Wide Web. 1997.
3. Kittredge R. Sublanguages. American Journal of Computational Linguistics 8(2), 1982: 79-84
4. Sekine S, Grishman R. A Corpus-based Probabilistic Grammar with Only Two Non-terminals. Fourth International Workshop on Parsing Technology. 1995: 216-223
5. Shapiro LG, Chung E, Detwiler LT, Mejino JLV, Agoncillo AV, Brinkley JF, Rosse C. A query interface for evaluating relationships in a large biomedical knowledge base, the Foundational Model of Anatomy. Submitted 2003.
5. Fernandez M, Florescu D, Levy A, Suciu D. A query language for a web-site management system. SIGMOD Record 1997 Sep; 26(3):4–11.
6. Mork P, Suciu D, Brinkley JF, Rosse C. A declarative query interface for large semantic networks. Proceedings AMIA Annual Symposium. 2002: 1110.
7. Androutsopoulos I, Ritchie GD, Thanish P. Natural language interfaces to databases - an introduction. In: Natural Language Engineering; 1995. p. 29-81.