Formulating queries for assessing clinical trial eligibility

Deryle Lonsdale[1], Clint Tustison[1], Craig Parker[2], and David W. Embley[2]

[1] Brigham Young University Department of Linguistics
[2] Brigham Young University Department of Computer Science

Abstract. This paper introduces a system that processes clinical trials using a combination of natural language processing and database techniques. We process web-based clinical trial recruitment pages to extract semantic information reflecting eligibility criteria for potential participants. From this information we then formulate a query that can match criteria against medical data in patient records. The resulting system reflects a tight coupling of web-based information extraction, natural language processing, medical informatic approaches to clinical knowledge representation, and large-scale database technologies. We present an evaluation of the system and future directions for further system development.

1 Background and overview

Researchers design information extraction systems to perform various tasks, and these tasks require various levels of linguistic processing. Some systems are only concerned with parsing out the extracted information and therefore only require the use of a syntactic parser. Others need more in-depth processing and include a semantic component that can give some meaning to the extracted information. Yet other systems are dependent on real-world knowledge and require a pragmatic component to relate the data gathered from the system to outside information.

One area receiving recent attention is the medical domain. Much of the natural language processing (NLP) research done with medical literature has involved developing systems that extract different types of relationships from text. For example, NLP techniques have been used on Medline[3] abstracts to extract information on genes, proteins, acronyms, and molecular binding relationships.

For its part, the field of medical informatics has produced large-scale resources, largely in database format, that specify the vast knowledge required for medical research and patient services. Highly specialized tools for representing clinical information and patient data have also been developed. Unfortunately, there has been only a modest amount of crossover between the NLP and medical informatics fields. The topic of information extraction is a salient one for demonstrating how applications can leverage the developments from both fields.

This paper[4] describes our approach to identification, extraction, and query formulation of information regarding medical clinical trials. Figure 1 shows an overview of the system. In Step 1, extraction and formula generation, we extract patient criteria from a web-based natural language description of qualifications for clinical

[3] See http://www.medlineplus.gov.

[4] This work was partially funded under National Science Foundataion Information and Intelligent Systems grant IIS-0083127. See also www.deg.byu.edu.

trial participants, and create predicate logic expressions (PLE's) that reflect the semantic content of the text. In Step 2, code generation, the system processes parsed criteria and their PLE's. The system then attempts to map the criteria to concepts in an electronic medical record. For the criteria that map successfully, the system outputs appropriate logic for computing whether or not a patient meets each criterion.

In Step 3, eligibility assessment, the system evaluates the eligibility of a potential participant by executing the logic generated in Step 2 against that patient's electronic medical record. The system produces a generated report that can help a clinician make an informed decision about whether to further evaluate the patient for enrollment in the clinical trial.

In Section 2 we describe Step 1 of the system, which involves the NLP component. Section 3 describes the subsequent medical records database query component. We then discuss the system evaluation in Section 4. Finally, we sketch ways the system could be enhanced in the future to provide better results.

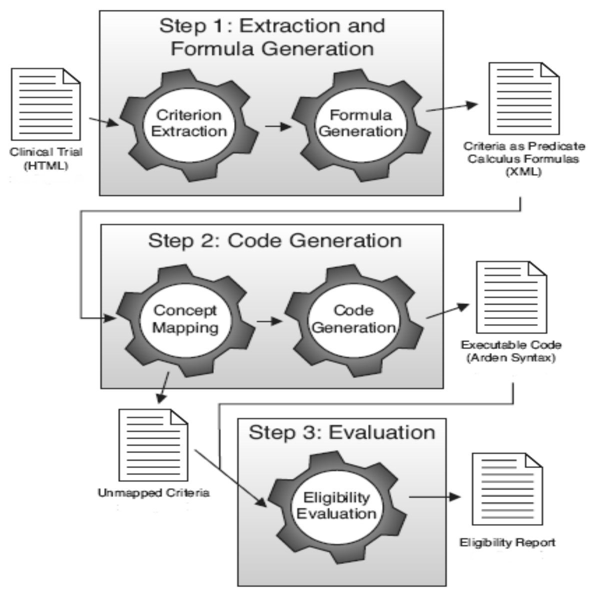


Fig. 1. Stages of processing in the system with data formats (input, intermediate, and output).

2 Extraction and formula generation

The domain that our system addresses is clinical trials, which medical professionals use as a tool to assess diagnostic and therapeutic agents and procedures. Such trials require voluntary human subjects to undergo the new treatments or receive experimental medications. With the increasing cost of bringing experimental new drugs to the public, there is a crucial need for improving and automating access to the information in clinical trials including the directed recruitment of experimental participants, which is otherwise costly and labor-intensive.

In this section we first discuss the web corpus we have targeted. Then we sketch the first stage of the system—how the pertinent text is processed by the NLP components of the system.

2.1 The corpus: clinical trials

From 1997 to 1999 the U.S. National Library of Medicine (NLM) and the National Institutes of Health (NIH) developed an online repository of clinical trials (McCray, 2000). This repository currently contains about 25,000 trials which are sponsored by various governmental and private organizations[5]; the repository receives about 8,000,000 page views per month[6].

Providers develop web pages for the clinical trials website using a simple user interface[7] including a text box for the eligibility criteria. No format restrictions are currently enforced on the text, though some boilerplate material can be entered (e.g. patient ages and gender) via dropdown boxes.

Each trial in the online repository comprises a series of sections that contain specific information regarding the trial that is useful to providers and patients. Figure 2 shows a sample web page for an individual clinical trial and the hierarchy of different components it contains.

For this paper we extract information from one section of the web page: the Eligibility section. This section contains a listing of the requirements that a person must satisfy in order to participate in the trial. For example, nearly every eligibility section specifies the patient age and also the gender.

Each web page undergoes two levels of preprocessing: (i) locating, retrieving, and converting the Eligibility section to an XML format with each item embedded in `<criterion>` tags; and (ii) manipulating the natural language text of some criteria to enable further processing. Often eligibility criteria are expressed telegraphically, for example with elided subjects or as standalone noun phrases. Parsing works best on full sentences, but only a small percentage have eligibility criteria structured as complete sentences. For elided subjects, a dummy subject and verb (i.e. *A criterion equals...*) are prepended to the criterion.

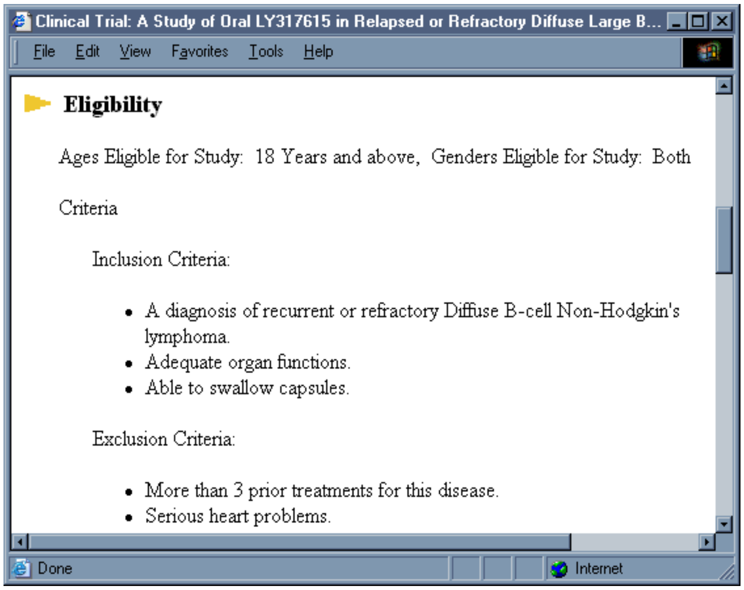
In other instances the first word in the criterion needs to be nominalized in order to produce a grammatical sentence. For example, the criterion *able to swallow capsules* is reformulated as *an ability to swallow capsules*, and then the dummy subject and verb are prepended.

Figure 2 shows an example clinical trials web page, its corresponding XML version, and the linguistically-annotated rendition of its eligibility criteria.

2.2 Deriving syntactic and semantic information

The next step in the process involves using a syntactic parser to process the natural language criteria and produce a corresponding syntactic representation. We

[5] See http://www.clinicaltrials.gov.
[6] See http://www.clinicaltrials.gov/ct/info/about.
[7] See http://prsinfo.clinicaltrials.gov/elig.html.



(a) ***Clinical trial web page NCT00042666.***

```
<criteria trial="http://www.clinicaltrials.gov/ct/show/NCT00042666">
 <criterion>
  <text>Eligibility</text>
  <text val="1">Ages Eligible for Study: 18 Years and above,</text>
 </criterion>
 <criterion>
  <text>Eligibility</text>
  <text val="2">Genders Eligible for Study: Both</text>
 </criterion>
  ... (ADDITIONAL CRITERIA) ...
 <criterion>
  <text>Eligibility</text>
  <text>Criteria</text>
  <text>Exclusion Criteria:</text>
  <text val="6">More than 3 prior treatments for this disease.</text>
 </criterion>
 <criterion>
  <text>Eligibility</text>
  <text>Criteria</text>
  <text>Exclusion Criteria:</text>
  <text val="7">Serious heart problems.</text>
 </criterion>
</criteria>
```

(b) ***Criteria annotated with XML tags.***

1. *A criterion equals* `an age greater than 18 years.`
2. *A criterion equals* `both genders.`
3. *A criterion equals* `a diagnosis of recurrent or refractory Diffuse B-Cell Non-Hodgkin’s lymphoma.`
4. *A criterion equals* `adequate organ functions.`
5. *A criterion equals* `an ability to swallow capsules.`
6. *A criterion equals* `more than 3 prior treatments for this disease.`
7. *A criterion equals* `serious heart problems.`

(c) ***Criteria with linguistic elements added.***

Fig. 2. Portion of clinical trial NCT00042666 and preprocessed versions of eligibility criteria.

use the link grammar (LG) parser (Sleator and Temperley, 1991). We chose this tool because of its open-source availability, efficiency, robustness in the face of ungrammaticality and out-of-vocabulary words, and flexibility[8].

The system reads in a .txt file containing each criterion (as extracted from the XML file described above) on a separate line in the file and parses each sentence individually. Because of structural ambiguities in English, a single input sentence might produce multiple parses; in this project, we only consider the highest-scored parse for subsequent processing. Figure 3 shows how a parse of *A criterion equals serious heart problems.* would be represented syntactically by the LG parser. Different labeled links connect the words in the sentence in a way that expresses their dependencies. These links are the key to the next step, extracting the semantic meaning from the syntactic output.

Once syntactic parsing of a sentence has been completed, the sentence is analyzed by the syntax-to-semantics conversion engine. This is a component (that was previously developed for other applications) specifically designed to take the output from the LG parser and convert its content to PLE's (though other semantic formats are also supported by the system).

The engine is built on Soar[9], a rule-based symbolic intelligent agent architecture that uses a goal-directed, operator-based approach to problem solving (Newell, 1994). Several dozen pertinent rules have been developed to interpret the LG parse links and convert their associated words to logical predicates and their associated arguments. Variables are generated for predicates to specify with appropriate arity which referents the predicates refer to.

For example, the parsed sentence *A criterion equals serious heart problems.* would yield the PLE "criterion(N2) & serious(N6) & heart_problems(N6) & equals(N2,N6). Note that the dummy subject and verb, which were added for parsing purposes, are present in the PLE. For this reason, a postprocessing stage removes this extraneous information. Then the resulting PLE is placed in the abovementioned XML file.

Figure 3 illustrates the parse, its PLE, and the XML file after the NL processing stages have finished.

3 Query generation

Once the source web page has undergone the NL processing techniques described above, the resulting extracted information feeds a database query stage to match them with patient medical records. In this section we can only briefly mention the technologies germane to the task at hand; more details are available elsewhere (Parker, 2005).

3.1 The target

Medical information systems manage patient information for a wide variety of tasks including patient care, administration (e.g. billing), research, and regulatory reporting. Coded medical vocabularies have been developed in order to ensure consistency, computability, and sharability. Often they are conceptually based

8 See http://www.link.cs.cmu.edu/link.

9 Freely available at http://sitemaker.umich.edu/soar.

```
    +----------------------------Xp--------------------------+
    |                        +-------------Op------------+     |
    +-----Wd-----+           |        +---------A--------+     |
    |     +--Ds--+----Ss----+         |        +----AN---+     |
    |     |      |           |        |        |         |     |
LEFT-WALL a criterion.n equals.v serious.a heart.n problems.n .
```

```
       LEFT-WALL      Xp      <---Xp---->  Xp         .
 (m)   LEFT-WALL      Wd      <---Wd---->  Wd         criterion.n
 (m)   a              Ds      <---Ds---->  Ds         criterion.n
 (m)   criterion.n    Ss      <---Ss---->  Ss         equals.v
 (m)   equals.v       O       <---Op---->  Op         problems.n
 (m)   serious.a      A       <---A----->  A          problems.n
 (m)   heart.n        AN      <---AN---->  AN         problems.n
       .              RW      <---RW---->  RW         RIGHT-WALL
```

(a) ***Link grammar output for a criterion's sentential form.***

~~criterion(N2) &~~ serious(N6) & heart_problems(N6) ~~&~~
~~equals(N2,N6)~~.

serious(N6) & heart_problems(N6).

(b) ***Predicate logic expressions before and after postprocessing.***

```
<criteria trial="http://www.clinicaltrials.gov/ct/show/NCT00042666">
 <criterion>
  <text>Eligibility</text>
  <text val="1">Ages Eligible for Study: 18 Years and above,</text>
  <pred val="1">age(N4) &amp; quantification(N5,greater_than)
                        &amp; measurement(N4,N5) &amp; units(N5,years)
                        &amp; magnitude(N5,18)</pred>
 </criterion>
 <criterion>
  <text>Eligibility</text>
  <text val="2">Genders Eligible for Study: Both</text>
  <pred val="2">both_genders(N4)</pred>
 </criterion>
... (ADDITIONAL CRITERIA) ...
 <criterion>
  <text>Eligibility</text>
  <text>Criteria</text>
  <text>Exclusion Criteria:</text>
  <text val="7">Serious heart problems.</text>
  <pred val="7">serious(N6) &amp; heart_problems(N6)</pred>
 </criterion>
</criteria>
```

(c) ***XML file with tagged predicate logic expressions added.***

Fig. 3. Final result of natural language processing stages.

and have associated lexicons or vocabularies which are sometimes hierarchical in nature. For example, the SNOMED-CT (Spackman and Campbell, 1998) coded vocabulary has a code “254837009” that represents the concept “breast cancer”.

Representing patient data usually requires more information than simple concepts. A data model called a detailed clinical model defines relationships between coded concepts or (other data values) and information of clinical interest. For example, a detailed clinical model might define a diagnosis in terms of a type and a subject/person, so that a statement “The patient has breast cancer.” could be encoded with the diagnosis type from SNOMED-CT as described above, and the subject/person with the relevant patient ID number. Detailed clinical models thus combine coded concepts into meaninful expressions of a higher-order nature. We make extensive use of both coded concepts and detailed clinical models in the concept mapping process shown in Step 2 in Figure 1.

The target electronic medical record for this project is Intermountain Health Care’s Clinical Data Repository (CDR)[10]. The CDR makes extensive use of coded vocabularies; it also defines detailed clinical models using Abstract Syntax Notation One (ASN.1) (Huff et al., 1998), an ISO standard for describing electronic messages (8824-1, 2002), including binary and XML encodings for many different application areas ranging from telecommunications to genome databases.

All coded concepts in the CDR are drawn from IHC’s Healthcare Data Dictionary (HDD) (Rocha et al., 1994), a large coded vocabulary (over 800,000 concepts with over 4 million synonyms). The names of all the detailed clinical models used in the CDR and the fields they contain are defined as concepts in the HDD.

The CDR comprises a database and its associated services. Besides providing a common access mechanism (for security, auditing, and error handling), the services crucially provide for handling of detailed clinical models as the basis for information access and retrieval. For example, an application can pass an instance of a detailed clinical model to the services, which will then return relevant instances of other detailed clinical models.

One of the outputs of Step 2 in Figure 1 is executable logic in Arden Syntax format (Hripcsak et al., 1990), an ANSI standard for handling medical data. Arden Syntax is written in units called medical logic modules (MLMs). Each MLM contains the logic necessary for making one medical decision. One category of information in an MLM defines knowledge required for making clinical decisions; this category is what we use in this project. The most significant slots in this category are the data slot and the logic slot. The data slot contains mappings of symbols used in an MLM to data in the target electronic medical record. The logic slot, as its name implies, contains the logic that operates on the data.

Finally, since electronic medical records vary widely in content and structure across applications, it has been useful to use an abstraction called the virtual medical record (VMR) (Parker et al., 2004). This assures that any number of healthcare organizations can write, maintain, and share clinical decision logic no matter what the structure of their own repositories. For eligibility criteria we use a small subset of VMR attributes called observations.

[10] See http://www.3m.com/us/healthcare/his/products/records/data_repository.jhtml. Intermountain Health Care (IHC) is a regional, nonprofit, integrated health system based in Salt Lake City, UT. The CDR is the result of a joint development effort between IHC and 3M Health Information Systems.

3.2 Concept mapping

The process outlined in Step 2 of Figure 1 takes the XML file described above as input. It attempts to map each criterion to concepts and data structures in the target electronic medical record. For each successful mapped criterion we generate executable code for determining if any patients meet the criterion.

Since IHC's CDR stores clinical data as instances of clinical models with coded concepts, and since all coded concepts are in the HDD, the mapping task involves matching words and phrases from the eligibility criteria to concepts in the HDD that represent either names or values in detailed clinical models.

The concept mapping portion of the system thus iterates through each criterion, attempting to map it to coded concepts from the HDD used in the CDR's detailed clinical models. The system uses multiple matching strategies executed sequentially, and once a match is found, subsequent matches are not sought. Seven decision points formulate the matching strategy; we sketch each below.

(1) Execute special case handling. We use string comparisons and regular expression matching for processing predictable boilerplate material (e.g. age and gender). (2) Match the raw text of a criterion to concepts in the database, in case subsequent processing does not succeed. Note that these two steps do not require predicate logic expressions, and thus are executed for every criterion. The remaining steps, however, are executed only for criteria that are successfully parsed into predicate calculus formulas.

(3) Match predicate names to the HDD. For example, the criterion "heart disease" yields the formula: heart(x) & disease(x). In this stage the mapper retrieves the best coded concept from the HDD that includes both predicate names. (4) Match the predicate with a measurement. Measurements are extracted as predicates; they include magnitudes, units, and other information. Here the criterion "LDL-C 130-190 mg/dL" is successfully matched to a query that searches LDL-C measurements (a valid HDD concept) in medical records and returns those within the acceptable range.

If the full matches above are not possible, partial matching is then tried. (5) Match name-value pairs. The predicate names are processed to find possible name-value pair relationships. For example, the criterion "diagnosis of appendicitis" does not map to a single concept in the HDD, but it does map to concepts in the CDR. Furthermore, the HDD recognizes "diagnosis" as a valid name for a clinical observation, and "appendicitis" as a valid value. We thus combine them to form a name-value pair. (6) Match a conjunction/disjunction. Often criteria are conjoined, and in such cases we process all elements. For example, the elements of the criterion "Hyperthyroidism or hypothyroidism" are mapped separately and then related with the relevant operation (conjunction or disjunction). (7) Partial match. The best possible match with all available predicate names is attempted, preferring nouns over other parts of speech. Thus, for example, a criterion "active neoplasms" would not match on the predicate "active" but would on the other one, "neoplasm". This heuristic is generally useful, though not always correct. For example, in the concept "renal disease," the adjective "renal" is more useful than the noun "disease".

3.3 Code generation

The second stage of Step 2 is code generation, where we generate executable code from the output of the concept mapping process. The code that we generate for this project is an Arden Syntax MLM (Medical Logic Module) that specifies VMR queries for data access[11]. The process has two steps.

The first step takes place in tandem with the mapping process described above. Each database mapping for a criterion spawns a related VMR query. Abstracting away from the details, this process can be summarized as a rather straightforward conversion from and to nested attribute/value structures.

The second and subsequent step in generating code involves creating the Arden Syntax MLM. For our database query we only use a small subset of the possible MLM slots (most of which are meant for human perusal). To generate the query, we iterate through the criteria, generating an Arden Syntax "read" statement when a mapping to the target electronic medical record is possible.

Assessing the applicability an encoded criterion involves the straightforward querying of electronic patient records. A report summarizes for the clinician which criteria parsed and matched the stated values. Figure 4 shows an Arden Syntax VMR query and a sample eligibility report.

```
Criterion1 := READ {
  <VMRQuery class="Observation">
      <value op="equals">
           <cd code="1450395" displayName="heart disease"/>
      </value>
  </VMRQuery>
```

(a) ***A sample Arden Syntax read statement containing a VMR query.***

Eligibility Report	
Header	
Title of Trial	A Study of Oral LY317615 in Relapsed or Refractory Diffuse & Large B-Cell Non-Hodgkin's Lymphoma
Patient Name	J. Doe
Medical Record #	1234567
Eligibility Summary	
Criteria met	6
Mapped Criteria for which eligibility could not be determined	7
Criteria not mapped	5
Total criteria	18
Criterion Detail	
Criterion 1	
...	
Criterion 3	
Criterion	LDL-C 130-190 mg/dL
Mapped	Yes
Status	Patient meets this criterion
...	
Criterion 11	
Criterion	Heart disease
Mapped	Yes
Status	Unable to determine if patient meets this criterion

(b) ***Portion of sample eligibility report.***

Fig. 4. Results for query generation and assessment stages.

[11] Generating code in a different language would only require an appropriate reimplementation of the generator interface.

4 Evaluation results

We recently carried out an end-to-end system performance evaluation. We randomly chose one hundred unseen clinical trials from www.clinicaltrials.gov and ran them through Steps 1 and 2 in Figure 1. Afterwards we manually inspected each report, comparing them to the generated queries, and characterizing their success or failure. We tallied these results numerically, and a summary appears in Figure 5.
The 85 parsable trials varied in size and complexity, having from 3 to 71 criteria per trial. They also varied widely in subject matter, covering conditions from cancer to infertility to gambling. Two main factors contributed to the failure of 15 trials: some had unexpected special characters (e.g. the HTML character “ü” representing the umlat u character), and others had sentences of such complexity that the parser failed.

Trials evaluated	100
Trials successfully completing Steps 1 & 2	85
Criteria extracted	1545
Criteria parsed into logical forms	473
Criteria parsed but not mapped into queries	49
Queries generated	520
Completely correct queries	140
Other useful queries	113
Technically correct queries	4
Incorrect queries	263

Fig. 5. Results from end-to-end system evaluation.

These 85 trials yielded 1,545 eligibility criteria; logical forms were successfully created for 473 of these criteria. All but 49 of these yielded queries, and another 96 queries could be generated without logical forms, so a total of 520 queries were formulated. Of these, 140 completely and exactly represented their original eligibility criteria. Another 113 of the queries were not entirely correct or complete but still yielded useful information for clinician decision-making. Four queries were technically well-formed based on the logical form though did not reflect the intent of the original criteria. In total, 257 queries were either completely correct, usefully correct, or technically correct. The remaining 263 queries were neither correct nor useful in determining eligibility.

5 Discussion and future work

Our experimental system demonstrates that some degree of automatic evaluation of eligibility criteria is feasible. In its current state, the system generated useful queries for about half of the number of criteria that had formulas. Several types of improvements are possible, and we highlight a few of them below.

One problematic issue has been the consistent authoring of parsable natural language statements by data providers. Tighter editorial controls could help solve this problem. A solution less intrusive to the users would be to develop collection of medical knowledge in the form of potentially reusable ontologies and axioms that could be used to assist in bridging the gap.

So far we have done little to customize the LG parser for our purposes, and we foresee improving it in at least three ways: (i) extending the range of acceptable grammatical structures; (ii) refining the parse scoring algorithm to return the most plausible parse; and (iii) integrating it with a large-scale medical lexicon as others have done (Szolovits, 2003). Currently the semantics engine only handles a limited number of syntactic structures—far less than those provided by the LG parser—and we have not as yet experimented with the semantic engine's inherent machine learning capabilities either.

In several cases the system correctly mapped the name portion of a pair, but incorrectly mapped the value portion, rendering the query incorrect. For example, consider the criterion *blood products or immunoglobulins within 6 months prior to entering the study*. The system found a mapping to an appropriate concept, "blood products used"; it also found a mapping to the valid concept "months". However, the latter is not a permissible value for the former, so processing failed. If appropriate constraint checking could mediate name-value pairings, the system would be able to more gracefully reformulate such instances.

The synonyms supplied by the HDD produced frequent successes, but occasional ambiguity proved problematic. The system mapped the abbreviation "PCP"to the drug "phencyclidine", whereas the trial intended "pneumocystic carinii pneumonia". It also mapped PG to "phosphatidyl glycerol" whereas the trial used it in an ad-hoc fashion for "pathological gambling".

Often unsuccessful queries reflected an absence of relevant concepts from the HDD. This is not unexpected, given the domain's focus on experimental medications. We could use additional sources of clinical concepts such as the National Library of Medicine's Unified Medical Language System (Lindberg, 1990) or a database of experimental drugs. New concepts, though, would not be helpful unless patient records contain such concepts, which is unlikely.

Several queries provided partial information that was useful, but could not fully assess eligibility. For example, the system mapped the criterion "uterine papillary serous carcinoma", to the concept "papillary carcinoma". Matching "papillary carcinoma" in a patient's record does not necessarily satisfy the criterion, but it could suggest further action by a clinician.

With some criteria a match will never be possible. EMR's typically do not store patient information that would reflect such criteria as "plans to become pregnant during the study" or "male partners of women who are pregnant".

Criteria we missed could be evaluated based on data in the EMR, by adding further inferencing with external knowledge. For example, "meets psychiatric diagnostic criteria for depression" requires the system to know what these diagnostic criteria are before this criterion can be evaluated.

Another possibility for improving the system include mapping criteria to more VMR classes than just the observation class. This would facilitate more accurate queries against information such as procedures, demographics, and medications.

Bibliography

[8824-1, 2002]8824-1, I. S. I. (2002). Information technology—Abstract Syntax Notation One (ASN.1): Specification of basic notation. Technical report. ITU-T Recommendation X.680.

[Hripcsak et al., 1990]Hripcsak, G., Clayton, P. D., Pryor, T. A., Haug, P., Wigertz, O. B., and v. d. Lei, J. (1990). The Arden Syntax for Medical Logic Modules. In Miller, R. A., editor, *Proceedings of the Fourteenth Annual Symposium on Computer Applications in Medical Care*, pages 200–204. IEEE Computer Society Press, Washington D. C.

[Huff et al., 1998]Huff, S. M., Rocha, R. A., Solbrig, H. R., Barnes, M. W., Schrank, S. P., and Smith, M. (1998). Linking a medical vocabulary to a clinical data model using Abstract Syntax Notation 1. *Methods of Information in Medicine*, 37(4-5):440–452.

[Lindberg, 1990]Lindberg, C. (1990). The Unified Medial Language System (UMLS) of the National Library of Medicine. *Journal of the American Medical Reccord Association*, 61(5):40–42.

[McCray, 2000]McCray, A. T. (2000). Better access to information about clinical trials. *Annals of Internal Medicine*, 133(8):609–614.

[Newell, 1994]Newell, A. (1994). *Unified Theories of Cognition*. Harvard University Press.

[Parker, 2005]Parker, C. (2005). Generating medical logic modules for clinical trial eligibility. Master's thesis, Brigham Young University.

[Parker et al., 2004]Parker, C. G., Rocha, R. A., Campbell, J. R., Tu, S. W., and Huff, S. M. (2004). Detailed clinical models for sharable, executable guidelines. *Medinfo*, 11(Pt 1):145–148.

[Rocha et al., 1994]Rocha, R. A., Huff, S. M., Haug, P. J., and Warner, H. R. (1994). Designing a controlled medical vocabulary server: the VOSER project. *Computers and Biomedical Research*, 27(6):472–507.

[Sleator and Temperley, 1991]Sleator, D. and Temperley, D. (1991). Parsing English with a link grammar. Technical Report CMU-CS-91-196, Carnegie Mellon University.

[Spackman and Campbell, 1998]Spackman, K. A. and Campbell, K. E. (1998). Compositional concept representation using snomed: Towards further convergence of clinical terminologies. In *Proceedings of the American Medical Informatics Association Annual Symposium*, pages 740–744.

[Szolovits, 2003]Szolovits, P. (2003). Adding a medical lexicon to an English parser. In *Proceedings of the American Medical Informatics Association Annual Symposium*, pages 639–643.