CSC486/2502 Test 1

When: Friday October 27
12:10 – 1:00 pm (50 minutes duration)
Please arrive at noon so that we can start promptly at 12:10

Where: BA026 (here)

What: Chapters 1-9 (Everything up to and including description logic)


Rough Format: 4-5 questions
- one question with definitions (1-2 lines)
- one question with short answers
- two or three questions where you may have to axiomatize something and/or prove something

Can I post old midterms?: Sorry – no I cannot.
Like the assignments, the material is not mine to post.
Review

1. Introduction
2. The language of first-order logic
3. Expressing knowledge
4. Resolution
5. Horn logic
6. Procedural control of reasoning
7. Rules in production systems
8. Object oriented representation
9. Structured descriptions
Key Concepts

1. **Introduction**
   - What is: knowledge, representation, reasoning
   - Why: knowledge, representation, reasoning
   - What is entailment
   - knowledge level vs. symbol level

2. **The language of first-order logic**
   - How about using FOL for KR
   - Syntax of FOL
   - Semantics of FOL (interpretations, denotations, ..)
   - Entailment
   - Knowledge-based systems
Key Concepts

3. **Expressing knowledge**
   - How to axiomatize a domain
   - Entailment examples
   - What we can and cannot encode easily in FOL
Key Concepts

4. Resolution
   • Deductive reasoning in a language as close as possible to FOL
   • CNF and DNF
   • Conversion to CNF (propositional case, first-order case)
   • Skolemization
   • Resolution rule of inference
   • Procedure for entailment
   • Relating the symbol level back up to the knowledge level
   • Soundness and completeness (resolution is *refutation*-complete!)
   • Answer extraction
   • Undecidability
   • Resolution is difficult…what can we do about it?
     1. Keep the search as general as possible
     2. Avoid needless search (e.g., irrelevant resolvents)
Key Concepts

4. **Resolution**
   - Resolution is difficult...what can we do about it?
     1. Keep the search as general as possible
     2. Avoid needless search (e.g., irrelevant resolvents)
   - More specifically:
     - Herbrand Theorem (Herbrand base, Herbrand universe)
     - MGU
     - Resolution strategies
     - Discussion of SAT
Key Concepts

We saw how hard it was to reason w/ FOL. Let’s try see how far we can get by \textit{restricting the language} and/or \textit{controlling inference} in some way. Is our language expressive enough still? Is reasoning still sound and complete?

5. **Horn logic**
   - What is a horn clause (positive, negative, definite)
   - SLD resolution, why it’s still sound and refutation-complete
   - Prolog uses SLD resolution. Why no answer predicate needed.
   - Backward-chaining and problem w/backward chaining (infinite loop)!
   - Forward-chaining and FO undecidability
Key Concepts

Restricting the language and/or controlling inference

6. **Procedural control of reasoning**

How to express knowledge to provide control for the simple case of backward-chaining Horn reasoning. Basically all the tricks that Prolog gives the user to control backward-chaining search, many of which are realized by the Prolog cut operator.

- Algorithm design (e.g., Fibonacci numbers, extra predicates, accumulators, etc.)
- Ordering goals
- Commiting to a branch of the search space (using cut)
- If-then-else (a version of committing)
- controlling backtracking
- Negation as failure (an important concept)
- Dynamic DBs (if-needed, if-added, if-removed)
Key Concepts

Restricting the language and/or controlling inference

7. **Rules in production systems**

Now we’re going to restrict our language and control inference by limiting ourselves to directional rules. In the last chapters we emphasized backward-chaining, here we will emphasize forward-chaining over rules.

- Goal vs. data-directed reasoning
- Forward vs. back-chaining
- Which is better suited for each and how to make one do the other
- Production systems
- Working memory
- Pattern specification and pattern matching
- Conflict resolution and combination
- Rete algorithm
- Production system applications
Restricting the language and/or controlling inference

8. **Object oriented representation**

All KR to date was flat. Let’s group facts and rules in terms of the kinds of objects they pertain to, i.e. what the knowledge is about. This section discussed *frames*, a procedural KR formalism that is OO in this way.

- Basic Frame Language (generic frame, instance frame, slots, …)
- Procedures and defaults
- IS-A and (defeasible!!) inheritance
- Reasoning with frames
- Example of frames in action
Key Concepts

Restricting the language and/or controlling inference

9. **Structured descriptions**

Again from sentences to objects, but this time in a more declarative rather than procedural way. Description logic/terminological logic. A subset of FOL that is often decidable. Great for defining vocabularies.

- (atomic & non-atomic) concepts, roles, constant
- Syntax and semantics of the DL language (extensions)
- The 3 sentences you can express and use to construct queries
- Entailment and reasoning, the 2 different queries
- Symbol-level: Computing subsumption
- Normalization & structure matching
- Taxonomies (implicit rather than explicit as in frame languages)
- Computing classification
- Most specific subsumer, most general subsumee
- Using the taxonomic structure
- Extensions and applications