National University of Singapore
School of Computing
CS3243 Introduction to AI

Tutorial 7: Logical Agents I

Issued: Week 9 Discussion in: Week 10

Important Instructions:

1. **Tutorial Assignment 7** consists of **Question 4** from this tutorial.
2. Your solution(s) must be TYPE-WRITTEN, though diagrams may be hand-drawn.
3. You must submit your solution(s) via **Canvas > Assignments**, satisfying the deadlines:
   - Pre-tutorial 7 submission by **Week 9 Sunday, 2359 hrs**.
   - Post-tutorial 7 submission by **Week 10 Friday, 2359 hrs**.
4. You must make both submissions for your assignment score to be counted.

Refer to **Appendix A** for notes on Knowledge Bases, and **Appendix B** for Propositional Logic Laws.

1. Verify the following logical equivalences. Cite the equivalence law used with each step of your working (refer to Appendix B for a list of these laws).

   (a) $\neg(p \lor \neg q) \lor (\neg p \land \neg q) \equiv \neg p$.

   (b) $(p \land \neg(\neg p \lor q)) \lor (p \land q) \equiv p$.

2. Victor would like to invite three friends, Alice, Ben, and Cindy to a party, but must satisfy the following constraints:

   C1. Cindy attends only if Alice does not attend.
   C2. Alice attends if either Ben or Cindy (or both) attend.
   C3. Cindy attends if Ben does not attend.

   Victor would like to know who will come to the party, and who will not. Help Victor by expressing each of the above three constraints in propositional logic, and then, using these constraints, determine who will attend his party.

3. Consider the following knowledge base (KB).

   R1. All Fire Trucks are red.
   R2. All Fire Trucks are vehicles.
   R3. All vehicles have four wheels.

   (a) Assume that an inference algorithm, $\boldsymbol{A}_1$, that takes the query sentence "a Ferrari is a red vehicle", infers the statement "a Ferrari is a Fire Truck". Determine which of the following properties **does not** apply to $\boldsymbol{A}_1$.

   Option 1: Complete.
   Option 2: Sound.
   Option 3: Both above.

(b) Assume that an inference algorithm, $\boldsymbol{A}_2$, is given the query sentence "a Ferrari is a red vehicle". Determine which of the following properties **would guarantee** that $\boldsymbol{A}_2$ would infer the sentence “a Ferrari has four wheels”.

Option 1: Completeness.

Option 2: Soundness.

Option 3: Both above are required.

(c) Determine if the following statement is True or False. *Justify your answer*.

“Two agents with the same knowledge base but different inference engines, both of which are complete and sound, will always behave in the same way.”

4. In each of the cases given below, a knowledge base (**KB**) and query (α) are specified. For each case, use **Truth Table Enumeration** to determine if **KB** $\vDash \alpha$. In other words, do the following.
   i. Write down all possible true/false assignments for all the variables present in associated **KB**.
   ii. For each such assignment, determine (1) if the **KB** is true/false, and (2) if α is true/false.
   iii. Finally, determine if $M(\textbf{KB}) \subseteq M(\alpha)$, i.e., determine if the models of the KB are a subset of the model of α.

   (a) **KB**: $(x_1 \lor x_2) \land (x_1 \Rightarrow x_3) \land \neg x_2$
   α: $x_3 \lor x_2$

   (b) **KB**: $(x_1 \lor x_3) \land (x_1 \Rightarrow \neg x_2)$
   α: $\neg x_2$

Appendix A: Notes on Knowledge Bases

A knowledge base (KB) corresponds to a set of logical rules that models what the agent knows. These rules are written using a certain language (or *syntax*) and uses a certain truth model (or *semantics*) to determine when a certain statement is *True* or *False*. In propositional logic sentences are defined as follows.

1. Atomic Boolean variables are sentences.
2. If S is a sentence, then so is $\neg S$.
3. If S_1 and S_2 are sentences, then so are:
   a. $S_1 \land S_2$, i.e., "S_1 and S_2".
   b. $S_1 \lor S_2$, i.e., "S_1 or S_2".
   c. $S_1 \Rightarrow S_2$, i.e., "S_1 implies S_2".
   d. $S_1 \Leftrightarrow S_2$, i.e., "S_1 holds if and only if S_2 holds".

We say that a logical statement α models β ($\alpha \vDash \beta$) if β holds whenever α holds. In other words, if $M(\alpha)$ is the set of all value assignments to variables in α for which α holds *True*, then $M(\alpha) \subseteq M(\beta)$.

An inference algorithm $\boldsymbol{A}$ is one that takes as input a knowledge base (**KB**) and a query α and decides whether α is derived from **KB**, written as $\mathbf{KB} \vdash_{\boldsymbol{A}} \alpha$. $\boldsymbol{A}$ is *sound* if $\mathbf{KB} \vdash_{\boldsymbol{A}} \alpha$ implies that $\mathbf{KB} \vDash \alpha$; $\boldsymbol{A}$ is *complete* if $\mathbf{KB} \vDash \alpha$ implies that $\mathbf{KB} \vdash_{\boldsymbol{A}} \alpha$.

Appendix B: Propositional Logic Laws

De Morgan's Laws	$\neg(p \lor q) \equiv \neg p \land \neg q$	$\neg(p \land q) \equiv \neg p \lor \neg q$
Idempotent laws	$p \lor p \equiv p$	$p \land p \equiv p$
Associative laws	$(p \lor q) \lor r \equiv p \lor (q \lor r)$	$(p \land q) \land r \equiv p \land (q \land r)$
Commutative laws	$p \lor q \equiv q \lor p$	$p \land q \equiv q \land p$
Distributive laws	$p \lor (q \land r) \equiv (p \lor q) \land (p \lor r)$	$p \land (q \lor r) \equiv (p \land q) \lor (p \land r)$
Identity laws	$p \lor False \equiv p$	$p \land True \equiv p$
Domination laws	$p \lor True \equiv True$	$p \land False \equiv False$
Double negation law	$\neg(\neg p) \equiv p$	
Complement laws	$p \lor \neg p \equiv True \lor \neg False \equiv True$	$p \land \neg p \equiv False \land \neg True \equiv False$
Absorption laws	$p \lor (p \land q) \equiv p$	$p \land (p \lor q) \equiv p$
Conditional identities	$p \Rightarrow q \equiv \neg p \lor q$	$p \Leftrightarrow q \equiv (p \Rightarrow q) \land (q \Rightarrow p)$