Propositional Logic: Semantics and an Example

CPSC 322 – Logic 2

Textbook §5.2
Lecture Overview

1. Recap: Syntax
2. Propositional Definite Clause Logic: Semantics
3. Using Logic to Model the World
4. Proofs
Propositional Definite Clauses: Syntax

Definition (atom)

An **atom** is a symbol starting with a lower case letter
Propositional Definite Clauses: Syntax

**Definition (atom)**
An **atom** is a symbol starting with a lower case letter.

**Definition (body)**
A **body** is an atom or is of the form $b_1 \land b_2$ where $b_1$ and $b_2$ are bodies.
# Propositional Definite Clauses: Syntax

## Definition (atom)
An **atom** is a symbol starting with a lower case letter.

## Definition (body)
A **body** is an atom or is of the form $b_1 \land b_2$ where $b_1$ and $b_2$ are bodies.

## Definition (definite clause)
A **definite clause** is an atom or is a rule of the form $h \leftarrow b$ where $h$ is an atom and $b$ is a body. (Read this as “$h$ if $b$.”)
Propositional Definite Clauses: Syntax

Definition (atom)
An atom is a symbol starting with a lower case letter.

Definition (body)
A body is an atom or is of the form $b_1 \land b_2$ where $b_1$ and $b_2$ are bodies.

Definition (definite clause)
A definite clause is an atom or is a rule of the form $h \leftarrow b$ where $h$ is an atom and $b$ is a body. (Read this as “$h$ if $b$."

Definition (knowledge base)
A knowledge base is a set of definite clauses.
Lecture Overview

1. Recap: Syntax
2. Propositional Definite Clause Logic: Semantics
3. Using Logic to Model the World
4. Proofs
Propositional Definite Clauses: Semantics

Semantics allows you to relate the symbols in the logic to the domain you’re trying to model.

Definition (interpretation)

An interpretation $I$ assigns a truth value to each atom.
Propositional Definite Clauses: Semantics

Semantics allows you to relate the symbols in the logic to the domain you’re trying to model.

**Definition (interpretation)**

An interpretation $I$ assigns a truth value to each atom.

We can use the interpretation to determine the truth value of clauses and knowledge bases:

**Definition (truth values of statements)**

- A body $b_1 \land b_2$ is true in $I$ if and only if $b_1$ is true in $I$ and $b_2$ is true in $I$.
- A rule $h \leftarrow b$ is false in $I$ if and only if $b$ is true in $I$ and $h$ is false in $I$.
- A knowledge base $KB$ is true in $I$ if and only if every clause in $KB$ is true in $I$. 
Models and Logical Consequence

Definition (model)
A **model** of a set of clauses is an interpretation in which all the clauses are *true*. 
Models and Logical Consequence

Definition (model)
A model of a set of clauses is an interpretation in which all the clauses are true.

Definition (logical consequence)
If $KB$ is a set of clauses and $g$ is a conjunction of atoms, $g$ is a logical consequence of $KB$, written $KB \models g$, if $g$ is true in every model of $KB$.

- we also say that $g$ logically follows from $KB$, or that $KB$ entails $g$.
- In other words, $KB \models g$ if there is no interpretation in which $KB$ is true and $g$ is false.
Example: Models

\[ KB = \begin{cases} 
  p \leftarrow q. \\
  q. \\
  r \leftarrow s.
\end{cases} \]

<table>
<thead>
<tr>
<th></th>
<th>( p )</th>
<th>( q )</th>
<th>( r )</th>
<th>( s )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( I_1 )</td>
<td>true</td>
<td>true</td>
<td>true</td>
<td>true</td>
</tr>
<tr>
<td>( I_2 )</td>
<td>false</td>
<td>false</td>
<td>false</td>
<td>false</td>
</tr>
<tr>
<td>( I_3 )</td>
<td>true</td>
<td>true</td>
<td>false</td>
<td>false</td>
</tr>
<tr>
<td>( I_4 )</td>
<td>true</td>
<td>true</td>
<td>true</td>
<td>false</td>
</tr>
<tr>
<td>( I_5 )</td>
<td>true</td>
<td>true</td>
<td>false</td>
<td>true</td>
</tr>
</tbody>
</table>

Which interpretations are models?
Example: Models

\[ KB = \begin{cases} 
  p \leftarrow q. \\
  q. \\
  r \leftarrow s. 
\end{cases} \]

<table>
<thead>
<tr>
<th>( I )</th>
<th>( p )</th>
<th>( q )</th>
<th>( r )</th>
<th>( s )</th>
</tr>
</thead>
</table>
| \( I_1 \) | true | true | true | true | is a model of \( KB \)  
| \( I_2 \) | false | false | false | false | not a model of \( KB \)  
| \( I_3 \) | true | true | false | false | is a model of \( KB \)  
| \( I_4 \) | true | true | true | false | is a model of \( KB \)  
| \( I_5 \) | true | true | false | true | not a model of \( KB \)  

Example: Models

\[ KB = \begin{cases} 
  p \leftarrow q. \\
  q. \\
  r \leftarrow s. 
\end{cases} \]

<table>
<thead>
<tr>
<th></th>
<th>( p )</th>
<th>( q )</th>
<th>( r )</th>
<th>( s )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( I_1 )</td>
<td>true</td>
<td>true</td>
<td>true</td>
<td>true</td>
</tr>
<tr>
<td>( I_2 )</td>
<td>false</td>
<td>false</td>
<td>false</td>
<td>false</td>
</tr>
<tr>
<td>( I_3 )</td>
<td>true</td>
<td>true</td>
<td>false</td>
<td>false</td>
</tr>
<tr>
<td>( I_4 )</td>
<td>true</td>
<td>true</td>
<td>true</td>
<td>false</td>
</tr>
<tr>
<td>( I_5 )</td>
<td>true</td>
<td>true</td>
<td>false</td>
<td>true</td>
</tr>
</tbody>
</table>

\( I_1 \) is a model of \( KB \)
\( I_2 \) is not a model of \( KB \)
\( I_3 \) and \( I_4 \) are models of \( KB \)
\( I_5 \) is not a model of \( KB \)

Which of the following is true?
- \( KB \models q, KB \models p, KB \models s, KB \models r \)
Example: Models

\[
KB = \begin{cases} 
p \leftarrow q. \\
q. \\
r \leftarrow s. 
\end{cases}
\]

<table>
<thead>
<tr>
<th></th>
<th>(p)</th>
<th>(q)</th>
<th>(r)</th>
<th>(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(I_1)</td>
<td>true</td>
<td>true</td>
<td>true</td>
<td>true</td>
</tr>
<tr>
<td>(I_2)</td>
<td>false</td>
<td>false</td>
<td>false</td>
<td>false</td>
</tr>
<tr>
<td>(I_3)</td>
<td>true</td>
<td>true</td>
<td>false</td>
<td>false</td>
</tr>
<tr>
<td>(I_4)</td>
<td>true</td>
<td>true</td>
<td>true</td>
<td>false</td>
</tr>
<tr>
<td>(I_5)</td>
<td>true</td>
<td>true</td>
<td>false</td>
<td>true</td>
</tr>
</tbody>
</table>

- \(I_1\) is a model of \(KB\)
- \(I_2\) not a model of \(KB\)
- \(I_3\) is a model of \(KB\)
- \(I_4\) is a model of \(KB\)
- \(I_5\) not a model of \(KB\)

Which of the following is true?

- \(KB \models q, \ KB \models p, \ KB \models s, \ KB \models r\)
- \(KB \not\models q, \ KB \not\models p, \ KB \not\models s, \ KB \not\models r\)
Lecture Overview

1. Recap: Syntax
2. Propositional Definite Clause Logic: Semantics
3. Using Logic to Model the World
4. Proofs
User’s view of Semantics

1. Choose a task domain: intended interpretation.
2. Associate an atom with each proposition you want to represent.
3. Tell the system clauses that are true in the intended interpretation: axiomatizing the domain.
4. Ask questions about the intended interpretation.
5. If $KB \models g$, then $g$ must be true in the intended interpretation.
6. The user can interpret the answer using their intended interpretation of the symbols.
Computer’s view of semantics

- The computer doesn’t have access to the intended interpretation.
  - All it knows is the knowledge base.
- The computer can determine if a formula is a logical consequence of KB.
  - If $KB \models g$ then $g$ must be true in the intended interpretation.
  - If $KB \not\models g$ then there is a model of $KB$ in which $g$ is false.
    This could be the intended interpretation.
Electrical Environment

Propositional Logic: Semantics and an Example
Representing the Electrical Environment

\[
\begin{align*}
\text{light}_l_1 & : \quad \text{live}_l_1 \leftarrow \text{live}_w_0 \\
\text{light}_l_2 & : \quad \text{live}_w_0 \leftarrow \text{live}_w_1 \land \text{up}_s_2 \\
\text{down}_s_1 & : \quad \text{live}_w_0 \leftarrow \text{live}_w_2 \land \text{down}_s_2 \\
\text{up}_s_2 & : \quad \text{live}_w_1, \leftarrow \text{live}_w_3 \land \text{up}_s_1 \\
\text{up}_s_3 & : \quad \text{live}_w_2 \leftarrow \text{live}_w_3 \land \text{down}_s_1 \\
\text{ok}_l_1 & : \quad \text{live}_l_2 \leftarrow \text{live}_w_4 \\
\text{ok}_l_2 & : \quad \text{live}_w_4 \leftarrow \text{live}_w_3 \land \text{up}_s_3 \\
\text{ok}_cb_1 & : \quad \text{live}_p_1 \leftarrow \text{live}_w_3 \\
\text{ok}_cb_2 & : \quad \text{live}_w_3 \leftarrow \text{live}_w_5 \land \text{ok}_cb_1 \\
\text{live}_w_5 & : \quad \text{live}_p_2 \leftarrow \text{live}_w_6 \\
\text{live}_w_6 & : \quad \text{live}_w_5 \leftarrow \text{live}_w_5 \land \text{ok}_cb_2 \\
\text{live}_w_5 & : \quad \text{live}_w_5 \leftarrow \text{live}_outside \\
\end{align*}
\]
Role of semantics

In user’s mind:
- \( l_2\text{-broken} \): light \( l_2 \) is broken
- \( sw_3\text{-up} \): switch is up
- \( power \): there is power in the building
- \( unlit_l_2 \): light \( l_2 \) isn’t lit
- \( lit_l_1 \): light \( l_1 \) is lit

In Computer:
\[
\begin{align*}
l_2\text{-broken} & \leftarrow sw_3\text{-up} \land power \land unlit_l_2. \\
sw_3\text{-up}. \\
power & \leftarrow lit_l_1. \\
unlit_l_2. \\
lit_l_1. \\
\end{align*}
\]

Conclusion: \( l_2\text{-broken} \)
- The computer doesn’t know the meaning of the symbols
- The user can interpret the symbols using their meaning
Lecture Overview

1. Recap: Syntax

2. Propositional Definite Clause Logic: Semantics

3. Using Logic to Model the World

4. Proofs
Proofs

- A **proof** is a mechanically derivable demonstration that a formula logically follows from a knowledge base.
- Given a proof procedure, \( KB \vdash g \) means \( g \) can be derived from knowledge base \( KB \).
- Recall \( KB \models g \) means \( g \) is true in all models of \( KB \).

**Definition (soundness)**

A proof procedure is **sound** if \( KB \vdash g \) implies \( KB \models g \).

**Definition (completeness)**

A proof procedure is **complete** if \( KB \models g \) implies \( KB \vdash g \).