Overview

⇒ Explicit Quantification

• Semantics

• Theorem Prover
Overview

- Predicate Calculus
  - variables
  - functions
  - negative & disjunctive
  - integrity constraints
  - quantifiers

© P. Heeman, 2020

CS560 Class 12: Explicit Quantification
Quantification

- So far, variables have been universally quantified at clause level
  \[ \forall X \forall Y \forall Z \quad \text{brothers}(X, Y) \leftarrow (\text{mother}(X, Z) \land \text{mother}(Y, Z) \land \neg (X = Y)) \]

  - Earlier in course, we argued that it was the same as follows:
  \[ \forall X \forall Y \quad \text{brothers}(X, Y) \leftarrow (\exists Z \quad \text{mother}(X, Z) \land \text{mother}(Y, Z) \land \neg (X = Y)) \]

  - Aside: If we assume UNA, equality is the same as unification

- But how can we capture “every boy loves a girl”
  - This means that for every boy, there exists a girl that the boy likes
    \[ \forall X \quad \text{boy}(X) \rightarrow (\exists Y \quad \text{girl}(Y) \land \text{likes}(X, Y)) \]
    \[ \forall X \quad (\exists Y \quad \text{girl}(Y) \land \text{likes}(X, Y)) \leftarrow \text{boy}(X) \]
    \[ \forall X \exists Y \quad (\text{girl}(Y) \land \text{likes}(X, Y) \leftarrow \text{boy}(X)) \]

- Need to explicitly deal with universal and existential quantifiers
Steps

• Step 1: Syntax
  - $\forall X$ and $\exists X$ can go around any ???

• Step 2: ??

• Step 3: ??
Order of Precedence

• Brackets tell you what arguments that infix operators apply to
  \[(2\times3) + 4 = 10\]
  \[2 \times (3+4) = 24\]

• If brackets are missing, use order of precedence
  - multiplication and division before add and subtract

• Same for FOPC
  - negation
  - and or implies
  - quantifiers
Overview

• Explicit Quantification

⇒ Semantics

• Theorem Prover
Semantics of Universal Quantification

- Textbook: $\forall X w$ is true in an interpretation $I$ iff $w$ is true in $I$ regardless of what object in the domain that $X$ is mapped to
  - $w$ probably has free variable $X$ in it
  - value of $X$ in $w$ determined by variable assignment
  - instead, explicitly enumerate over all possible assignments for $X$
  - $I_\rho(\forall X w)$ is true iff for all $d \in D$, $I_\rho'(w)$ is true where $\rho=\rho'$ except that $\rho'(X)=d$
What does it mean to say that $w$ is true for $I$?
- true if forall $\rho$ $I\rho$ makes $w$ true

What does it mean for $\forall X w$ to be true for $I$?
- true if forall $\rho$ $I\rho$ makes $\forall X w$ true
- true if forall $\rho$ forall $d \in D$ where $\rho$ maps $X$ to $d$ then $I\rho$ makes $w$ true
Facts about Universal Quantification

- Any expression with free variables is same as that expression but with the free variables universally quantified
  - Just making the universal quantification explicit

- Can prove that order of universals doesn’t matter:
  \( \forall X \forall Y w \) is true exactly when \( \forall Y \forall X w \) is true
Semantics of \textit{Exists}

- Textbook: $\exists Xw$ is true in an interpretation $I$ \textbf{iff} there is some individual in the domain, say $d$, such that if $X$ is mapped to $d$ then $w$ is true in $I$

- $I\rho(\exists Xw)$ is true \textbf{iff} there is a $d \in D$, such that $I\rho'(w)$ is true where $\rho=\rho'$ except that $\rho'(X)=d$
Examples

there exists a boy

there is a boy who is tall

there is a book that alice likes

there is a book that alice and tom like

there is a book that everyone loves

there is a book that every boy likes

everyone has a mother

every boy is tall
Overview

• Explicit Quantification
• Semantics

⇒ Theorem Prover
Existential Elimination

• Example: \( \exists Z \; \text{block}(Z) \)
  - We know that something is a block.
    In worst case, there isn’t even a name for it in our syntax.
    So let’s make up a new constant, say \( k \), and let \( k \) be that block.
  - Same with \( \exists Z \exists Y \; \text{hates}(Z, Y) \)

• Example: \( \forall Y \exists Z \; \text{hates}(Y, Z) \)
  - Who \( Z \) is depends on who \( Y \) is.
    Every different value of \( Y \) might have a different \( Z \).
  - If \( foe \) is a new function symbol, we can let \( foe(Y) \) be who \( Y \) hates.
    So we can infer \( \text{hates}(Y, foe(Y)) \)

• This process referred to as skolemization
  - \( foe(Y) \) is a skolem function
  - \( k \) is a skolem constant
Conversion to Clauses

- Convert each sentence into a set of clauses
  - with conjunction joining them
- Step 1: Eliminate \( \rightarrow \) by writing it in terms of \( \lor \), and \( \neg \)
- Step 2: Distribute negation so only applies to atomic sentences
  - \( \neg \neg \phi \) replaced by \( \phi \)
  - De’Morgans rules for distributing \( \neg \) over \( \lor \) and \( \land \)
    - \( \neg \forall v \phi \) replaced by \( \exists v \neg \phi \) and \( \neg \exists v \phi \) replaced by \( \forall v \neg \Phi \)
- Step 3: Rename variables so each quantifier has a unique variable
• **Step 4:** Replace existential quantifiers by skolem functions
  - Similar to existential instantiation
  - If inside of universal quantifier, its variable is considered free
    - $\exists X \, p(X)$ turns into $p(k)$ where $k$ not yet used
    - $\forall Y \, \exists X \, p(X, Y)$ turns into $p(f(Y), Y)$ where $f$ is a new function

• **Step 5:** Universals dropped
  - Doesn’t change meaning to drop outermost universals

• **Step 6:** Distribute $\lor$’s over $\land$’s
  - $\phi \lor (\psi \land \chi)$ replaced by $(\phi \lor \psi) \land (\phi \lor \chi)$
Example

• \( \forall X (\forall Y \ p(X, Y)) \rightarrow \neg (\forall Y \ q(X, Y) \rightarrow r(X, Y)) \)

• **Step 1:** Remove \( \rightarrow \) and \( \leftrightarrow \)
  \[ \forall X \neg (\forall Y \ p(X, Y)) \vee \neg (\forall Y \ q(X, Y) \rightarrow r(X, Y)) \]
  \[ \forall X \neg (\forall Y \ p(X, Y)) \vee \neg (\forall Y \ \neg q(X, Y) \vee r(X, Y)) \]

• **Step 2:** Distribute negation inwards
  \[ \forall X (\exists Y \ \neg p(X, Y)) \vee \neg (\forall Y \ \neg q(X, Y) \vee r(X, Y)) \]
  \[ \forall X (\exists Y \ \neg p(X, Y)) \vee (\exists Y \ \neg (\neg q(X, Y) \vee r(X, Y))) \]
  \[ \forall X (\exists Y \ \neg p(X, Y)) \vee (\exists Y \ (q(X, Y) \land \neg r(X, Y))) \]

• **Step 3:** Make variable names unique for each quantifier
  \[ \forall X (\exists Y \ \neg p(X, Y)) \vee (\exists Z \ (q(X, Z) \land \neg r(X, Z))) \]
Example Continued

• Step 4: Replace existential quantifiers by skolem functions
\[ \forall X \neg p(X, f(X)) \lor (q(X, g(X)) \land \neg r(X, g(X))) \]

• Step 5: Universals dropped
\[ \neg p(X, f(X)) \lor (q(X, g(X)) \land \neg r(X, g(X))) \]

• Step 6: Distribute \( \lor \)'s over \( \land \)'s
\[ (\neg p(X, f(X)) \lor q(X, g(X)) \land (\neg p(X, f(X)) \lor \neg r(X, g(X)))) \]
A Syntactic Proof

• Example:
  Horses are faster than dogs
  There is a greyhound that is faster than every rabbit
  (This is asserting that there is a greyhound and that it is faster than every rabbit)
  Harry is a horse
  Ralph is a rabbit
  Prove Harry is faster than ralph

• What world knowledge might we need?
A Syntactic Proof

• Example:
  Horses are faster than dogs
  There is a greyhound that is faster than every rabbit
  (This is asserting that there is a greyhound and that it is faster than every rabbit)
  Harry is a horse
  Ralph is a rabbit
  Prove Harry is faster than ralph

• What world knowledge might we need?
  Greyhounds are dogs
  If X is faster than Y and Y is faster than Z then X is faster than Z
Convert to Disjunctive Normal Form

\(\forall X \forall Y \text{horse}(X) \land \text{dog}(Y) \rightarrow \text{faster}(X, Y)\)
\(\forall X \forall Y \neg(\text{horse}(X) \land \text{dog}(Y)) \lor \text{faster}(X, Y)\)
\(\neg \text{horse}(X) \lor \neg \text{dog}(Y) \lor \text{faster}(X, Y)\)  \hspace{1cm} (1)

\(\exists X \text{greyhound}(X) \land (\forall Y \text{rabbit}(Y) \rightarrow \text{faster}(X, Y))\)
\(\exists X \text{greyhound}(X) \land (\forall Y \neg \text{rabbit}(Y) \lor \text{faster}(X, Y))\)
\(\text{greyhound}(\text{foe}) \land (\forall Y \neg \text{rabbit}(Y) \lor \text{faster}(\text{foe}, Y))\)
\(\text{greyhound}(\text{foe}) \land (\neg \text{rabbit}(Y) \lor \text{faster}(\text{foe}, Y))\)
\(\text{greyhound}(\text{foe}) \land \neg \text{rabbit}(Y) \lor \text{faster}(\text{foe}, Y)\)  \hspace{1cm} (2)

\text{horse}(\text{harry})  \hspace{1cm} (4)
\text{rabbit}(\text{ralph})  \hspace{1cm} (5)

\(\forall X \text{greyhound}(X) \rightarrow \text{dog}(X)\)
\(\neg \text{greyhound}(X) \lor \text{dog}(X)\)  \hspace{1cm} (6)

\(\forall X \forall Y \forall Z \text{faster}(X, Y) \land \text{faster}(Y, Z) \rightarrow \text{faster}(X, Z)\)
\(\forall X \forall Y \forall Z \neg(\text{faster}(X, Y) \land \text{faster}(Y, Z)) \lor \text{faster}(X, Z)\)
\(\forall X \forall Y \forall Z \neg \text{faster}(X, Y) \lor \neg \text{faster}(Y, Z) \lor \text{faster}(X, Z)\)
\(\neg \text{faster}(X, Y) \lor \neg \text{faster}(Y, Z) \lor \text{faster}(X, Z)\)  \hspace{1cm} (7)
Bottom-Up Proof

\[ \neg \text{horse}(X) \lor \neg \text{dog}(Y) \lor \text{faster}(X, Y) \]  
\[ \text{greyhound}(\text{foe}) \]  
\[ \neg \text{rabbit}(Y) \lor \text{faster}(\text{foe}, Y) \]  
\[ \text{horse}(\text{harry}) \]  
\[ \text{rabbit}(\text{ralph}) \]  
\[ \neg \text{greyhound}(X) \lor \text{dog}(X) \]  
\[ \neg \text{faster}(X, Y) \lor \neg \text{faster}(Y, Z) \lor \text{faster}(X, Z) \]  

\[ \neg \text{dog}(Y) \lor \text{faster}(\text{harry}, Y) \]  
\[ \neg \text{greyhound}(Y) \lor \text{faster}(\text{harry}, Y) \]  
\[ \text{faster}(\text{harry}, \text{foe}) \]  
\[ \neg \text{faster}(\text{foe}, Z) \lor \text{faster}(\text{harry}, Z) \]  
\[ \neg \text{rabbit}(Z) \lor \text{faster}(\text{harry}, Z) \]  
\[ \text{faster}(\text{harry}, \text{ralph}) \]  

(1) 
(2) 
(3) 
(4) 
(5) 
(6) 
(7) 
(8) 
(9) 
(10) 
(11) 
(12) 
(13)
Adding on to Datalog

• Equality
  + Axioms or Paramodulation

• Unique Name Assumption
  + Assume UNA and check if objects have different names
  + Delay atoms with inequality if they are not ground

• Negative and Disjunctive Knowledge
  + CNF with any number of positive or negative literals
  + Top down: negative ancestor rule
  + Bottom up: include answer clause in KB so can find disjunctive answers

• Complete Knowledge Assumption
  + So don’t have to list negative knowledge
  + Clark Completion
  + Negation as faiylre (delay atoms with ‘not’ if not ground)

• Existential Quantification
  + Convert to CNF with skolem functions/ constants