

TDT4136 Introduction to Artificial Intelligence

Lecture 8: First Order Logic

Chapter 8 in the textbook

Pinar Öztürk

Norwegian University of Science and Technology

2021

Syntax of FOL (Predicate) logic

Semantics of FOL

Quantifiers

Next week: Inference in FOL (chapter 9)

Pros and Cons of propositional logic

PROs:

- Propositional logic is declarative: pieces of syntax correspond to facts
- Propositional logic allows partial information (unlike most data structures and databases)
- Propositional logic is compositional: meaning of $B_{1,1} \wedge P_{1,2}$ is derived from meaning of $B_{1,1}$ and of $P_{1,2}$
- Meaning in propositional logic is context-independent (unlike natural language, where meaning depends on context)

CON:

- Propositional logic has very limited expressive power (unlike natural language)

Limitations of propositional logic

- Some knowledge is hard or impossible to encode in the propositional logic.
- Two cases that are specially hard to represent:
 - Statements about similar objects, relations
 - Statements referring to groups of objects.

Limitations of propositional logic - Example 1

- **Statements referring to groups of objects required to be enumerated**
- Example: Assume we want to express *Every student likes vacation*:
 - John likes vacation \wedge
 - Mary likes vacation \wedge
 - Ann likes vacation \wedge
 -
- Problem:** KB grows large
- Possible solution: ??**
- *All students like vacation.*

Limitations of propositional logic - Example 2: Seniority domain

- Statements about similar objects and relations need to be enumerated
- Assume we have:
 - Stig is older than Sissel
 - Sissel is older than Paul
 - Stig is older than Sissel \wedge Sissel is older than Paul \implies Stig is older than Paul
- We can derive *Stig is older than Paul*

Example on Limitations of propositional logic - Example 2 cont.

- Assume we add *Hanne is older than Sissel* into the KB
- The current KB now:
 - Stig is older than Sissel*
 - Sissel is older than Paul*
 - $\text{Stig is older than Sissel} \wedge \text{Sissel is older than Paul} \implies \text{Stig is older than Paul}$
 - Hanne is older than Sissel*
- What else do we need to have in the KB in order to derive *Hanne is older than Paul*?

Example on Limitations of propositional logic - Example 2 cont.

- The current KB:
Stig is older than Sissel
Sissel is older than Paul
Stig is older than Sissel \wedge *Sissel is older than Paul* \implies *Stig is older than Paul*
Hanne is older than Sissel
- What else do we need to have in the KB in order to derive *Hanne is older than Paul*?

We need:
Hanne is older than Sissel \wedge *Sissel is older than Paul* \implies *Hanne is older than Paul*

What is the problem?

Problem? KB grows large

Possible solution: ??

PersA is older than PersB \wedge PersB is older than PersC \implies
PersA is older than PersC

Limitations of Propositional Logic -example 3: Wumpus

- Consider the statement "If there is breeze in a square, there must be pit in an adjacent square"
- In propositional logic we need 16 sentences (one for each square) to represent this statement (for 4x4 grid):
 - $B_{1,1} \implies P_{1,2} \vee P_{2,1}$
 - $B_{1,2} \implies P_{1,2} \vee P_{1,3} \vee P_{2,2}$
 - ...
 - ...
- **We want to be able to say this in one single sentence.**

How to say it in one sentence

- Our statement above refers to 2 types objects (pit and square). The square has the property to be breezy. The relationship between a square and pit is adjacency, i.e., neighbourhood.
- In FOL, this statement is represented by means of the following formula - instead of 16 sentences in propositional logic,:

$\forall \text{ square, adjacent}(\text{square}, \text{pit}) \implies \text{breezy}(\text{square})$

First Order Logic (FOL)

- More expressive than propositional logic. While PL assumes that the world contains facts, FOL assumes the world contains
 - Objects: trees, people, numbers, movies, Trump, maps, colours, hypotheses, Wumpus....
 - Relations: square, smelly, brother of, older than, owns, has colour, adjacent to....
 - Functions: brother-of, colour-of, adjacent-to,....
- Representing objects, their properties, relations and statements about them.

Syntax of FOL - elements

- Constants: NTNU, KingHarald, 5, ...
 - Predicates: Brother, $>$, $=$, ...
 - Functions: Sqrt, LeftLegOf
 - Variables: x , y , a , b , ...
 - Connectives: \neg , \wedge , \vee , \rightarrow , \leftrightarrow
 - Quantifiers: \forall , \exists
-
- First order logic or "predicate calculus" introduces **variables** and **quantifiers** to refer to objects in the world, their relations, group of objects, and to express general rules.

Functions versus Relations

- Functions are a way of referring to individuals indirectly, e.g.,
 - brother-of (Janne) and Edvard would refer to the same object/individual if Janne's brother is the person named Edvard.
- Relations hold among objects
 - Brother(Janne,Edvard) is true if Edvard is Janne's brother
 - Unary, binary, n-ary relations

- Atomic sentence: $\text{predicate}(\text{term}_1, \dots, \text{term}_n)$
Term: constant, or variable or function($\text{term}_1, \dots, \text{term}_m$)
- Complex sentences: Composed of atomic sentences using connectives

$$\neg S, S_1 \vee S_2, S_1 \wedge S_2, S_1 \implies S_2, S_1 \Leftrightarrow S_2$$

Examples:

- $\text{Brother}(\text{JonSnow}, \text{AryaStark}) \implies \text{Sister}(\text{AryaStark}, \text{JonSnow})$
- $\text{Childish}(\text{Trump}) \vee \text{BestStudents}(\text{Students}(\text{NTNU}), \text{Norway})$

Universal Quantifiers

- Quantifications express properties of collections of objects.
- \forall : "For all"
- $\forall \langle \text{variable} \rangle \langle \text{sentence} \rangle$

We can state the following: $\forall x P(x)$

English translation: "for all values of x , $P(x)$ is true"

Example: $P(x) : x+1 \geq x$

English translation: "for all values of x , $x+1 \geq x$ is true"

Universal quantification

- Everyone at NTNU is smart:

$$\forall x \text{ } At(x, NTNU) \implies Smart(x)$$

- $\forall x \text{ } P$ is true in a model m iff P is true with x being **each** possible object in the model

- Equivalent to the **conjunction** of **instantiations** of P

$$\begin{aligned} & (At(KingJohn, NTNU) \implies Smart(KingJohn)) \\ \wedge & (At(Richard, NTNU) \implies Smart(Richard)) \\ \wedge & (At(NTNU, NTNU) \implies Smart(NTNU)) \\ \wedge & \dots \end{aligned}$$

Existential quantification

- \exists : "There exist a/some"
- $\exists \langle \text{variables} \rangle \langle \text{sentence} \rangle$
- Someone at NTNU is smart:
 $\exists x \text{ At}(x, \text{NTNU}) \wedge \text{Smart}(x)$
- $\exists x P$ is true in a model m iff P is true with x being **some** possible object in the model
- Equivalent to the **disjunction** of **instantiations** of P
 - $(\text{At}(\text{KingJohn}, \text{NTNU}) \wedge \text{Smart}(\text{KingJohn}))$
 - $\vee (\text{At}(\text{Richard}, \text{NTNU}) \wedge \text{Smart}(\text{Richard}))$
 - $\vee (\text{At}(\text{NTNU}, \text{NTNU}) \wedge \text{Smart}(\text{NTNU}))$
 - $\vee \dots$

Example for comparison of Propositional Logic and FOL

Primitives in Propositional Logic. Ski-race example.¹

Objects

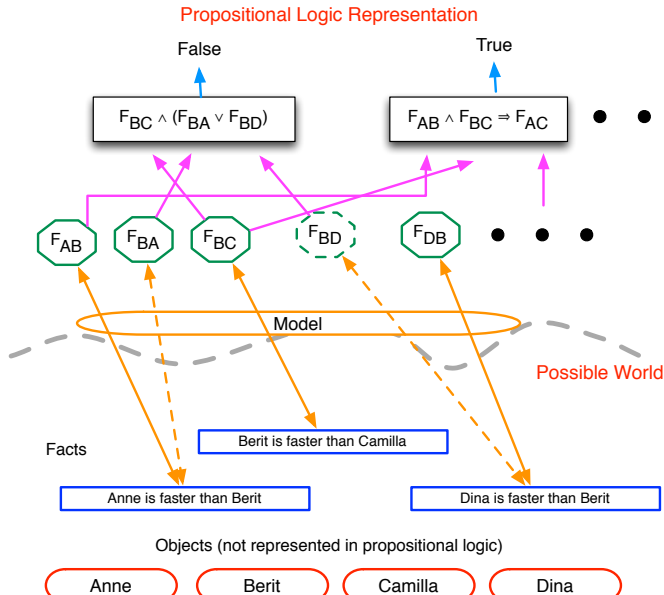
- Anne (A), Berit (B), Camilla (C), Dina (D)
- These are not actually represented in propositional logic.
- Only True-or-False facts **about** them are represented.
- Objects alone do not have a truth value, whereas all primitives in propositional logic do.

Propositional Symbols

- F_{AB} (Anne is faster than Berit) F_{BA} (Berit is faster than Anne), .. F_{AC}
- These have truth values and are the primitive terms.
- Their logical combinations into sentences (representing specific facts or general rules) also have truth values.

¹Thanks to Keith Downing for this example

Example- Ski-race in Propositional Logic



Example- Ski Race in FOL

Objects

- Anne, Berit, Camilla, Dina
- These are now represented in the logic, even though they still have no truth value.

Functions

- $\text{best10K}(\text{person}) \rightarrow \text{time}$. Mapping from athlete to their best 10K time.
- $\text{rank}(\text{person}) \rightarrow \text{integer}$. Mapping from athlete to their seeding in the competition.
- $\text{start}(\text{person}) \rightarrow \text{integer}$. Mapping from athlete to start order in the race, where slowest start first.
- These have no truth value and map one primitive object (person) to another (number).

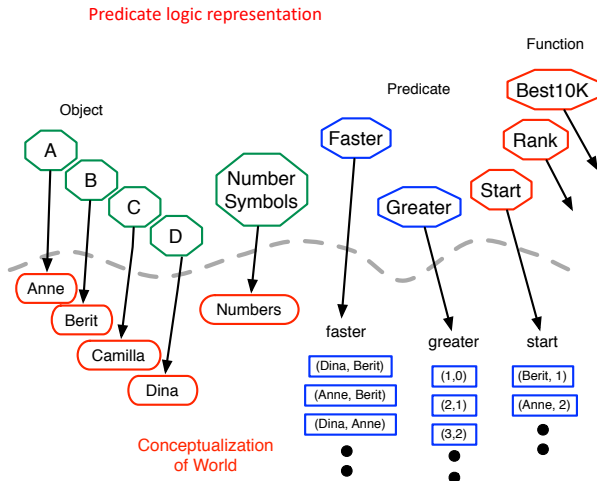
Example- Ski Race in FOL

Relations

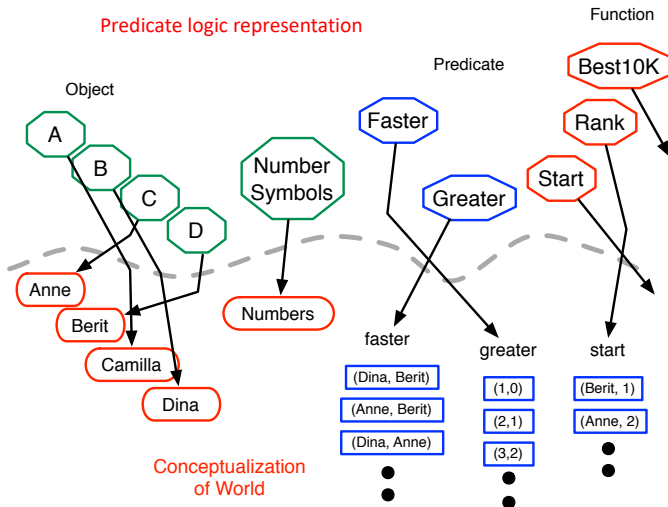
- $\text{greater}(X,Y) \rightarrow \{\text{True}, \text{False}\}$. Is number X greater than number Y?
- $\text{faster}(X,Y) \rightarrow \{\text{True}, \text{False}\}$. Is athlete X faster than athlete Y?
- These always have a truth value.
- These are often viewed as explicit lists of tuples, one list for each TRUE relation. So in one possible world, **faster** is represented by:
 $\{ (\text{anne}, \text{berit}), (\text{anne}, \text{camilla}), (\text{dina}, \text{anne}), (\text{dina}, \text{camilla}), (\text{camilla}, \text{berit}), (\text{dina}, \text{berit}) \}$

Example- Interpretations in First-Order Logic

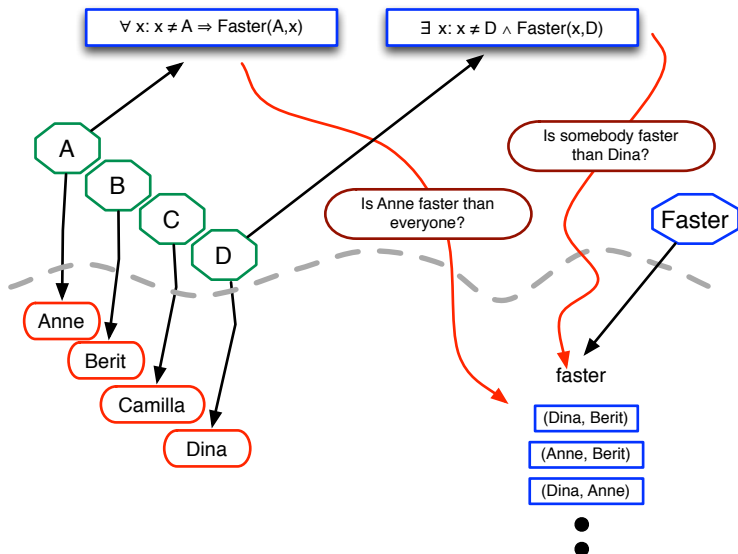
Interpretation = Mapping from constant, function and predicate symbols of the representation to the conceptualization.



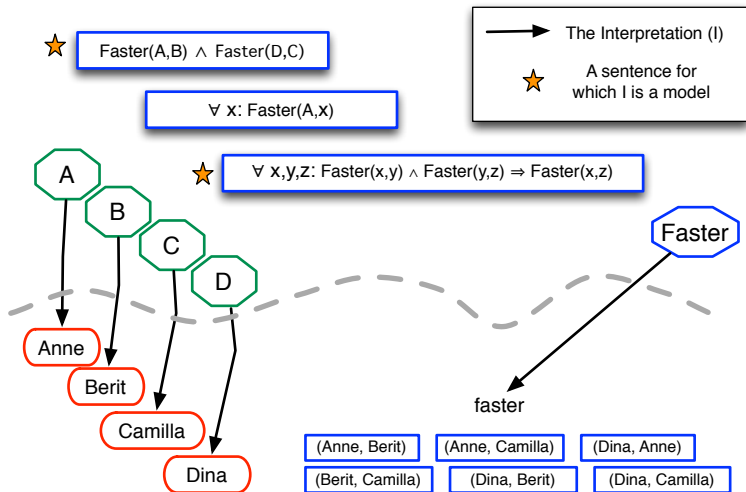
Example- Another Legal Interpretation



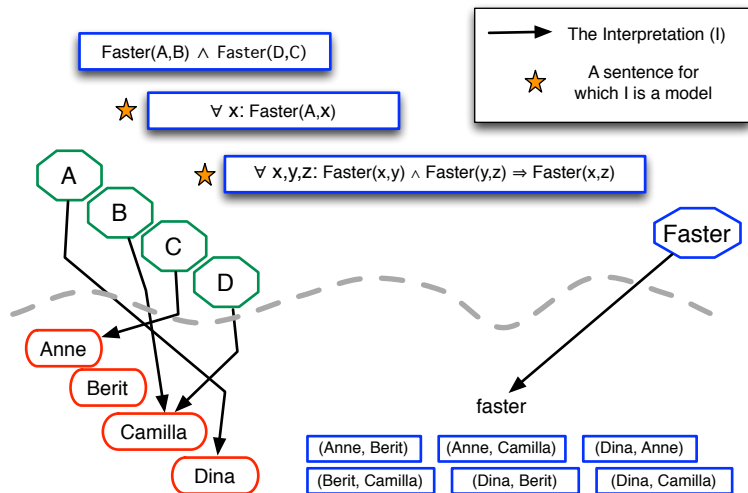
Example - Evaluating Sentences with Quantified Variables



Example - Ski Models, Interpretation 1



Example - Ski Models for Interpretation 2



Common mistakes with Quantifiers

- Typically, \implies is the main connective with \forall
- Common mistake: using \wedge as the main connective with \forall :
Let us represent this sentence: "Everybody at NTNU is smart"

$$\forall x \text{ } At(x, NTNU) \wedge Smart(x)$$

- Correct? No, it means "Everyone is at NTNU and everyone is smart"

Another common mistake to avoid

- Typically, \wedge is the main connective with \exists
- Common mistake: using \implies as the main connective with \exists :
Let us represent this sentence: "There is a smart person at NTNU" –
Correct?

$$\exists x \text{ } At(x, NTNU) \implies Smart(x)$$

- No, because it becomes true if there is anyone who is not at NTNU!

Connections between \forall and \exists

- All statements made with one quantifier can be converted into equivalent statements with the other quantifier by using negation.
- Remember De Morgan's Rule-
 - $P \wedge Q \equiv (\neg(\neg P \vee \neg Q))$
 - $P \vee Q \equiv (\neg(\neg P \wedge \neg Q))$
 - $\neg(P \wedge Q) \equiv \neg P \vee \neg Q$
 - $\neg(P \vee Q) \equiv \neg P \wedge \neg Q$
- Generalized De Morgan's rules
 - $\forall x P(x) \equiv \neg \exists x (\neg P(x))$
 - $\exists x P(x) \equiv \neg \forall x (\neg P(x))$
 - $\neg \forall x P(x) \equiv \exists x (\neg P(x))$
 - $\neg \exists x P(x) \equiv \forall x (\neg P(x))$

Connections between \forall and \exists

- Conversion/negation of a quantifier means to represent the quantified statement in terms of the "other" quantifier.
- Negation rules/laws:
 - $\neg \exists \equiv \forall \neg$
 - $\neg \forall \equiv \exists \neg$
 - $\neg \forall \neg \equiv \exists$
 - $\neg \exists \neg \equiv \forall$

Multiple Quantifiers

- more complex sentences can be formulated by multiple variables and by nesting quantifiers
- variables must be introduced by quantifiers, and belong to the innermost quantifier that mention them
- examples

$\forall x, y \text{ Parent}(x, y) \implies \text{Child}(y, x)$

$\forall x \text{ Human}(x) \forall y \text{ Loves}(x, y)$

$\exists x \text{ Human}(x) \forall y \text{ Loves}(x, y)$

$\exists x \text{ Human}(x) \forall y \text{ Loves}(y, x)$

Multiple Quantifiers -2

- "For all x , there exists a y such that $P(x,y)$ "
 - $\forall x \exists y P(x,y)$
 - Example: $\forall x \exists y (x + y = 0)$
- "There exists an x such that for all y $P(x,y)$ is true"
 - $\exists x \forall y P(x,y)$
 - Example: $\exists x \forall y (x * y = 0)$

Order of Quantifiers

- Reversing the order of the same type of quantifiers does not change the truth value of a sentence

$\forall x \forall y P(x, y)$ is the same as $\forall y \forall x P(x, y)$

- $\exists x \forall y$ *and* $\forall x \exists y$ are not equivalent!

$\forall x \exists y \text{ Loves}(x, y)$

$\exists y \forall x \text{ Loves}(x, y)$

Example -Order of Quantifiers

- $\forall x \exists y \text{ Loves}(x, y)$
 - Everyone in the world loves someone.
 - With parentheses: $\forall x (\exists y \text{ Loves}(x, y))$
 - y is inside the **scope** of x
- $\exists x \forall y \text{ Loves}(x, y)$
 - There is a person who loves everyone in the world.
 - The same person x loves everybody.
 - y is inside the scope of x

More examples- Order of Quantifiers

- $\exists y \forall x \text{Loves}(x, y)$
 - There is someone whom everybody likes
 - Everybody likes the same y.
 - x is inside the scope of y.
- What about "There is a puppy that likes every woman."

More examples - Order of Quantifiers

- What about "There is a puppy that loves every woman."
- $\exists p \text{Puppy}(p) \wedge (\forall w \text{Woman}(w) \implies \text{Loves}(p, w))$

Negating multiple Quantifiers

- Recall negation rules for single quantifiers:

$$\neg \forall x P(x) = \exists x \neg P(x)$$

$$\neg \exists x P(x) = \forall x \neg P(x)$$

- You change the quantifier(s), and negate what it's quantifying:

$$\neg(\forall x \exists y P(x, y)) \equiv \exists x \neg \exists y P(x, y) \equiv \exists x \forall y \neg P(x, y)$$

- FOL is called first-order because it allows quantifiers to range over objects (terms) but not properties, relations, or functions applied to those objects.
- Second-order logic allows quantifiers to range over predicates and functions as well:
 - $\forall x \forall y [(x = y) \leftrightarrow (\forall P P(x) \leftrightarrow P(y))]$
Means that two objects are equal if and only if they have exactly the same properties.
 - $\forall F \forall G [(F = G) \leftrightarrow (\forall x F(x) = G(x))]$
Means that two functions are equal if and only if they have the same value for all possible arguments.

- Perception

$\forall s, b, g, u, c, t \text{ Percept}([\text{Stench}, b, g, u, c], t) \implies \text{Stench}(t)$

$\forall s, b, g, u, c, t \text{ Percept}([s, \text{Breeze}, g, u, c], t) \implies \text{Breez}(t)$

$\forall s, b, g, u, c, t \text{ Percept}([s, b, \text{Glitter}, u, c], t) \implies \text{Glitter}(t)$

$\forall s, b, g, u, c, t \text{ Percept}([s, b, g, \text{Bump}, c], t) \implies \text{Bump}(t)$

$\forall s, b, g, u, c, t \text{ Percept}([s, b, g, u, \text{Scream}], t) \implies \text{Scream}(t)$

where s is square, t is timestep.

Wumpus in first-order logic

- Represent a square as an object s with coordinate functions $x(s)$ and $y(s)$.
- Represent square adjacency with a predicate

$$\begin{aligned}\forall s, q \text{ Adjacent}(s, q) &\Leftrightarrow x(s) = x(q) \wedge y(s) = y(q) + 1 \\ &\vee x(s) = x(q) \wedge y(s) + 1 = y(q) \\ &\vee y(s) = y(q) \wedge x(s) = x(q) + 1 \\ &\vee y(s) = y(q) \wedge x(s) + 1 = x(q)\end{aligned}$$

Reflex Agent in the Wumpus World

- rules that directly connect percepts to actions
$$\forall s, b, g, u, c, t \text{ Percept}([s, b, \text{Glitter}, u, c], t) \implies \text{Action}(\text{Grab}, t)$$
- requires a rule for each different combination of percepts - every possible state
- can be simplified by intermediate predicates
$$\begin{aligned}\forall s, b, g, u, c, t \text{ Percept}([\text{Stench}, b, g, u, c], t) &\implies \text{Smelly}(t) \\ \forall s, b, g, u, c, t \text{ Percept}([s, \text{Breeze}, g, u, c], t) &\implies \text{Breezy}(t) \\ \forall s, b, g, u, c, t \text{ Percept}([s, b, \text{Glitter}, u, c], t) &\implies \text{AtGold}(t) \\ \forall s, b, g, u, c, t \text{ Percept}([s, b, g, \text{Bump}, c], t) &\implies \text{Bump}(t) \\ \forall s, b, g, u, c, t \text{ Percept}([s, b, g, u, \text{Scream}], t) &\implies \text{Scream}(t) \\ \forall t \text{ AtGold}(t) &\implies \text{Action}(\text{Grab}, t) \\ \dots\end{aligned}$$

Reflex Wumpus Agent

- Assume the agent has the following percept statement at time 5:
Percept [noe , none , glitter , none , none , 5]
- We want to query for an appropriate action. Find an action (*ask* the KB): ASK ($\exists a \text{ Action}(a, 5)$)
- 1. $\forall s, b, g, u, c, t [\text{Percept}(\text{stench}, b, g, u, c, t) \implies \text{Smelly}(t)]$
 $\forall s, b, g, u, c, t [\text{Percept}(s, b, \text{glitter}, u, c, t) \implies \text{AtGold}(t)]$
...
- 2. Step: Choice of action
 $\forall t [\text{AtGold}(t) \implies \text{Action}(\text{grab}, t)]$
 $\text{AtGold}(5) \implies \text{Action}(\text{grab}, 5)$
- Our reflex agent does not know when it should climb out of the cave (with gold) and cannot avoid an infinite loop.

Deducing hidden properties

- Properties of locations :
 - $\forall x, t \text{ } At(Agent, x, t) \wedge Smell(t) \implies Smelly(x)$
 - $\forall x, t \text{ } At(Agent, x, t) \wedge Breeze(t) \implies Breezy(x)$
where x is square, t time.
- Squares are breezy near a pit. **Definition** for the *Breezy* predicate (x and y are squares):
$$\forall y \text{ } Breezy(y) \Leftrightarrow \exists x \text{ } Pit(x) \wedge Adjacent(x, y)$$
- Propositional logic would require an axiom for each square!
- **Diagnostic** rule - infer cause from effect
$$\forall y \text{ } Breezy(y) \implies \exists x \text{ } Pit(x) \wedge Adjacent(x, y)$$
- **Causal** rule - infer effect from cause
$$\forall x, y \text{ } Pit(x) \wedge Adjacent(x, y) \implies Breezy(y)$$