CMSC 471: Intro to AI

Propositional and First-Order Logic

Many slides courtesy Tim Finin

Big Idea

- Drawing reasonable conclusions from a set of data (observations, beliefs, etc.) seems key to intelligence
- Logic is a powerful and well-developed approach to this & highly regarded by people
- Logic is also a strong formal system that computers can use (cf. John McCarthy's work)
- We can solve some AI problems by represent-ing them in logic and applying standard proof techniques to generate solutions

AI Use Cases for Logic

Logic has many use cases even in a time dominated by deep learning, including these examples:

- Modeling and using knowledge
- Allowing agents to develop complex plans to achieve a goal and create optimal plans
- Defining and using semantic knowledge graphs such as schema.org and Wikidata
- Adding features to neural network systems

Knowledge-Based Agents: Big Idea

- Drawing reasonable conclusions from a set of data (observations, beliefs, etc.) seems key to intelligence
- Logic is a powerful and well-developed approach to this & highly regarded by people
- Logic is also a strong formal system that computers can use (cf. John McCarthy)
- We can solve some AI problems by representing them in logic and applying standard proof techniques to generate solutions

Inference in People

- People can do logical inference, but are not always very good at it
- Reasoning with negation and disjunction seems particularly difficult
- But, people seem to employ many kinds of reasoning strategies, most of which are neither *complete* nor *sound*

Question #1

Here is a simple puzzle

Don’t try to solve it -- listen to your intuition

Question #1

Here is a simple puzzle

Don’t try to solve it -- listen to your intuition

- A bat and ball cost $1.10
- The bat costs one dollar more than the ball
- How much does the ball cost?

Question #1

Here is a simple puzzle

Don’t try to solve it -- listen to your intuition

- A bat and ball cost $1.10
- The bat costs one dollar more than the ball
- How much does the ball cost?

The ball costs $0.05

Question #2

Try to determine, as quickly as you can, if the argument is logically valid. Does the conclusion follow the premises?

Question #2

Try to determine, as quickly as you can, if the argument is logically valid. Does the conclusion follow the premises?

- **All roses are flowers**
- **Some flowers fade quickly**
- **Therefore some roses fade quickly**

Question #2

Try to determine, as quickly as you can, if the argument is logically valid. Does the conclusion follow the premises?

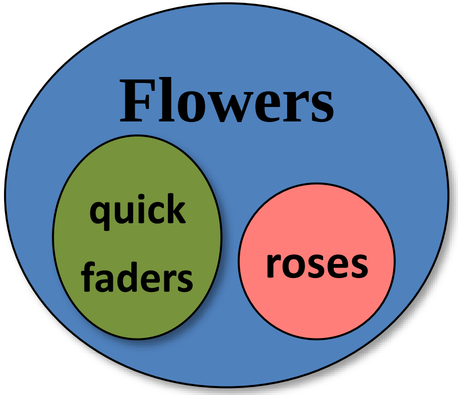
- All roses are flowers
- Some flowers fade quickly
- Therefore some roses fade quickly

It is possible that there are no roses among the flowers that fade quickly

Question #2

Try to determine, as quickly as you can, if the argument is logically valid. Does the conclusion follow the premises?

- All roses are flowers
- Some flowers fade quickly
- Therefore some roses fade quickly



It is possible that there are no roses among the flowers that fade quickly

Question #3

It takes 5 machines 5 minutes to make 5 widgets

How long would it take 100 machines to make 100 widgets?

Question #3

It takes 5 machines 5 minutes to make 5 widgets

How long would it take 100 machines to make 100 widgets?

- **100 minutes or 5 minutes?**

Question #3

It takes 5 machines 5 minutes to make 5 widgets

How long would it take 100 machines to make 100 widgets?

- 100 minutes or 5 minutes?

5 minutes

Wason Selection Task

- I have a pack of cards; each has a *letter* written on one side and a *number* on the other
- I claim the following rule is true:

  If a card has a *vowel* on one side, then it has an *even number* on the other

- Which cards should you turn over in order to decide whether the rule is true or false?

E	4	T	7

Wikipedia

Wason Selection Task

- Wason (1966) showed that people are bad at this task
- To disprove rule P=>Q, find a situation in which P is true but Q is false, i.e., show P^~Q
- To disprove **vowel => even**, find a card with a vowel and an odd number
- Thus, turn over the cards showing **vowels** and those showing **odd numbers**

$\boxed{E}$	4	T	$\boxed{7}$

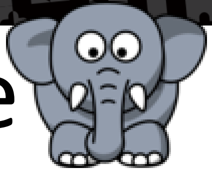
Wason Selection Task

- This version is easier for people, as shown by [Griggs & Cox, 1982]
- You are the bouncer in a bar; which of these people do you verify given the rule: *You must be 21 or older to drink beer.*

beer	**coke**	**22**	**20**

Perhaps easier because it’s more familiar or because people have special strategies to reason about certain situations, such as cheating in a social situation

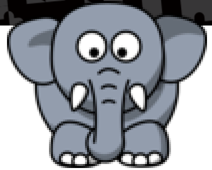
Negation in Natural Language



- We often model the meaning of natural language sentences as a logic statements
- This maps these into equivalent statements
  - All elephants are gray
  - No elephant are not gray
- Double negation is common in informal language: th*at won't do you no good*

As a way to state a negative more strongly

Negation in Natural Language



- It’s not just informal language actually
- What does this mean:

  we cannot underestimate the importance of logic

- Does it mean logic is important or not?
- See the LanguageLog blog misnegation archive for lots of real-world examples

Logic as a Methodology

Even if people don’t use formal logical reason-ing for solving a problem, logic might be a good approach for AI for a number of reasons

- Airplanes don’t need to flap their wings
- Logic may be a good implementation strategy
- Solution in a formal system can offer other benefits, e.g., letting us prove properties of the approach

• See neats vs. scruffies

Knowledge-based agents

- Knowledge-based agents have a knowledge base (KB) and an inference system
- KB: a set of representations of facts believed true
- Each individual representation is called a **sentence**
- Sentences are expressed in a **knowledge represent-ation language**
- The agent operates as follows:
  1. It **TELL**s the KB what it perceives
  2. It **ASK**s the KB what action it should perform
  3. It performs the chosen action

Architecture of a KB agent

- **Knowledge Level**
  - Most abstract: describe agent by what it knows
  - Ex: Autonomous vehicle knows Golden Gate Bridge connects San Francisco with the Marin County
- **Logical Level**
  - Level where knowledge is encoded into *sentences*
  - Ex: **links**(GoldenGateBridge, SanFran, MarinCounty)
- **Implementation Level**
  - Software representation of sentences, e.g.
    `(links goldengatebridge sanfran marincounty)`

Does your agent have complete knowledge?

- Closed world assumption (CWA): the lack of knowledge is assumed to mean it's false
- Open world assumption: no such assumption is made

Q: Why would we ever make a closed world assumption?

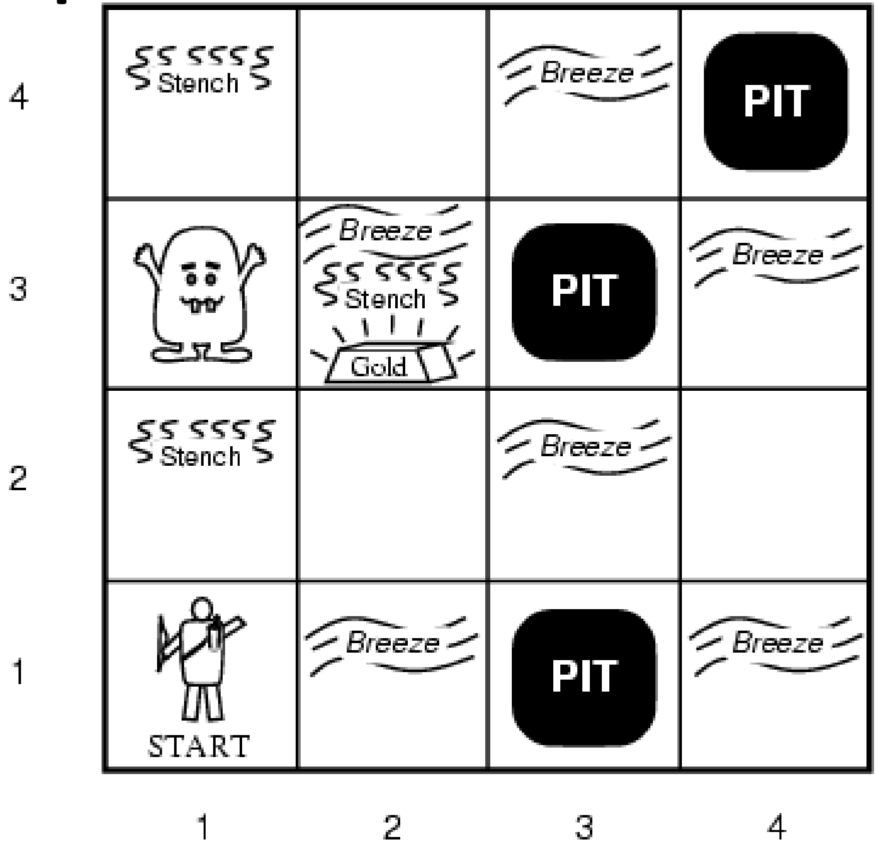
Wumpus World environment

- Based on Hunt the Wumpus computer game
- Agent explores cave of rooms connected by passageways
- Lurking in a room is the *Wumpus*, a beast that eats any agent that enters its room
- Some rooms have *bottomless pits* that trap any agent that wanders into the room
- Somewhere is a heap of gold in a room
- Goal: collect gold & exit w/o being eaten

AIMA’s Wumpus World

The agent always starts in the field [1,1]

Agent’s task is to find the gold, return to the field [1,1] and climb out of the cave



Agent in a Wumpus world: Percepts

- The agent perceives
  - **stench** in square containing Wumpus and in adjacent squares (not diagonally)
  - **breeze** in squares adjacent to a pit
  - **glitter** in the square where the gold is
  - **bump**, if it walks into a wall
  - Woeful **scream** everywhere in cave, if Wumpus killed
- Percepts given as five-tuple, e.g., if stench and breeze, but no glitter, bump or scream:

  [Stench, Breeze, None, None, None]
- Agent cannot perceive its location, e.g., (2,2)

Wumpus World Actions

- **go forward**
- **turn right** 90 degrees
- **turn left** 90 degrees
- **grab**: Pick up object in same square as agent
- **shoot**: Fire arrow in direction agent faces. It continues until it hits & kills Wumpus or hits outer wall. Agent has one arrow, so only first shoot action has effect
- **climb:** leave cave, only effective in start square
- **die:** automatically and irretrievably happens if agent enters square with pit or living Wumpus

Wumpus World Goal

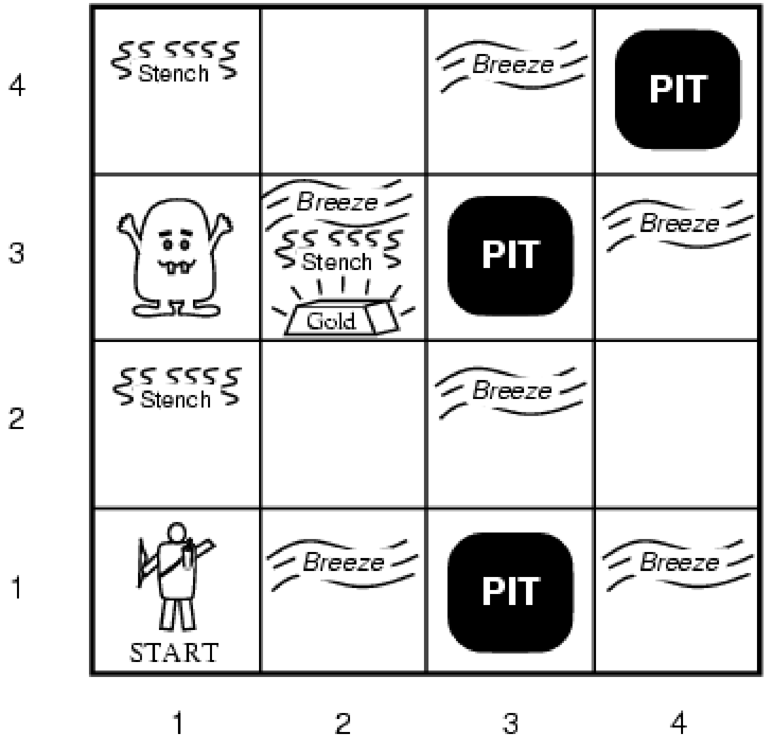
Agent’s goal is to find the gold and bring it back to the start square as quickly as possible, without getting killed

- 1,000 point reward for climbing out of cave with gold
- 1 point deducted for every action taken
- 10,000 point penalty for getting killed

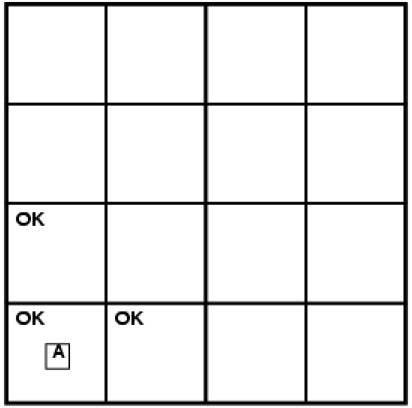
AIMA's Wumpus World

The agent always starts in the field [1,1]

Agent's task is to find the gold, return to the field [1,1] and climb out of the cave



Exploring a wumpus world



label	fact
A	agent
B	breeze
G	glitter
OK	safe cell
P	pit
S	stench
W	wumpus

We label cells with facts agent learns about them as it moves through world

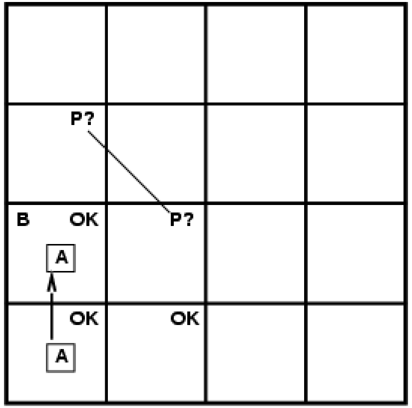
The Hunter's first step

(a) (b)

Since agent is alive and perceives neither breeze nor stench at [1,1], it **knows** [1,1] and its neighbors are OK

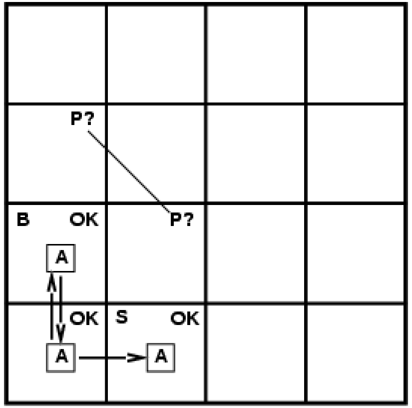
Moving to [2,1] is a **safe move** that reveals a breeze but no stench, **implying** that Wumpus isn't adjacent but one or more pits are

Exploring a wumpus world



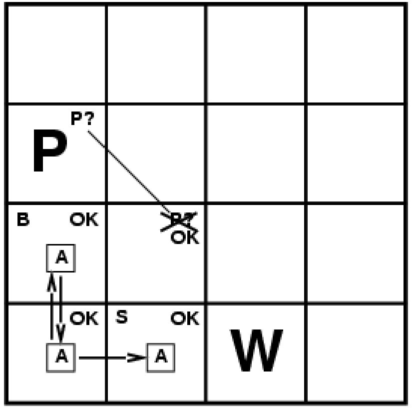
A	agent
B	breeze
G	glitter
OK	safe cell
P	pit
S	stench
W	wumpus

Exploring a wumpus world



A	agent
B	breeze
G	glitter
OK	safe cell
P	pit
S	stench
W	wumpus

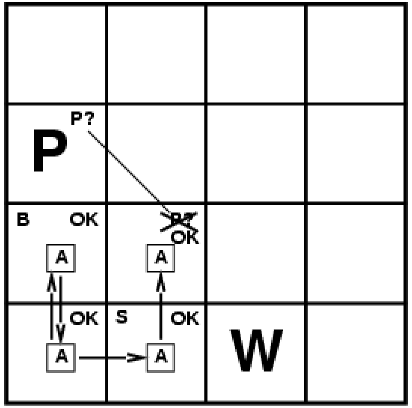
Exploring a wumpus world



A	agent
B	breeze
G	glitter
OK	safe cell
P	pit
S	stench
W	wumpus

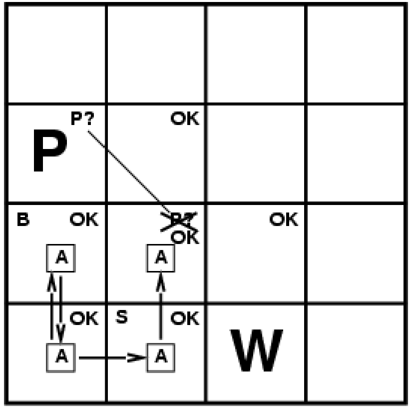
No stench in (1,2) => Wumpus not in (2,2)
No breeze in (2,1) => no pit in (2,2) => pit in (1,3)

Exploring a wumpus world



A	agent
B	breeze
G	glitter
OK	safe cell
P	pit
S	stench
W	wumpus

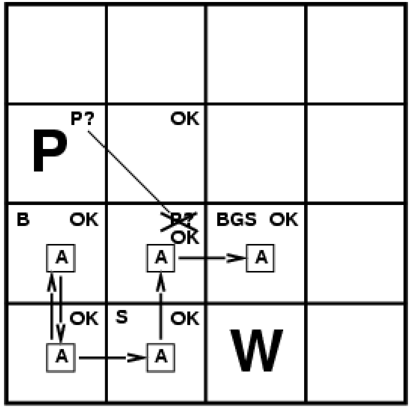
Exploring a wumpus world



A	agent
B	breeze
G	glitter
OK	safe cell
P	pit
S	stench
W	wumpus

Going to (2,2) is the only “safe” move

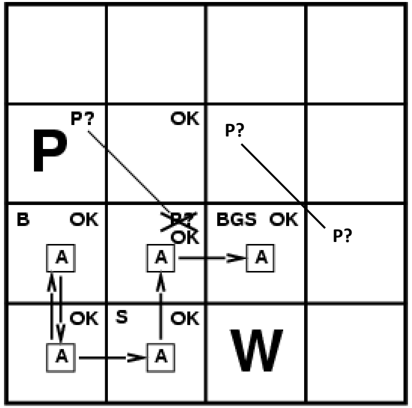
Exploring a wumpus world



A	agent
B	breeze
G	glitter
OK	safe cell
P	pit
S	stench
W	wumpus

Going to (2,3) is a “safe” move

Exploring a wumpus world



A	agent
B	breeze
G	glitter
OK	safe cell
P	pit
S	stench
W	wumpus

Found gold! Now find way back to (1,1)

Logic in general

- **Logics** are formal languages for representing information so that conclusions can be drawn
- **Syntax** defines the sentences in the language
- **Semantics** define the "meaning" of sentences
  - i.e., define **truth** of a sentence in a world

E.g., the language of arithmetic

- $x+2 \geq y$ is a sentence; $x2+y > \{\}$ is not a sentence
- $x+2 \geq y$ is true iff the number $x+2$ is no less than the number y
- $x+2 \geq y$ is true in a world where $x = 7, y = 1$
- $x+2 \geq y$ is false in a world where $x = 0, y = 6$
- $x+1 > x$ is true for all numbers x

Entailment

- Entailment: one thing follows from another
- KB $\models \alpha$
- Knowledge base *KB* entails sentence α iff α is true in *all possible worlds* where *KB* is true

- A **possible world where KB is true** can contain additional facts as long as they don't contradict anything in the KB
- E.g.: 'what we know today' + there's lif on Venus!

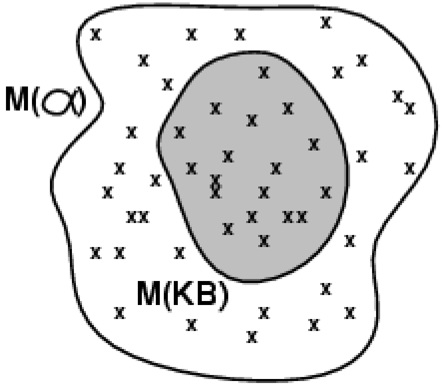
Entailment

- Entailment: one thing follows from another
- KB $\models$ α
- Knowledge base *KB* entails sentence α iff α is true in *all possible worlds* where *KB* is true
  – E.g., the KB containing "UMBC won" and "JHU won" entails "Either UMBC won or JHU won"
  – E.g., x+y = 4 entails x = 4 - y
  – Entailment is a relationship between (sets of) sentences (i.e., syntax) that is based on **semantics**

Models

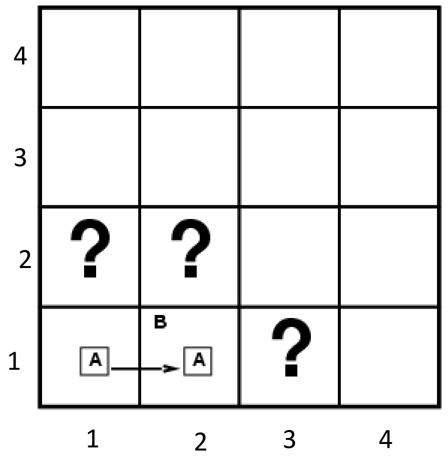
- Logicians talk of models: formally structured worlds w.r.t which truth can be evaluated
- *m* is a model of sentence α if α is true in *m*

  Lots of other things might or might not be true or might be unknown in *m*
- *M(α)* is the set of all models of α
- Then KB $\models$ α iff *M(KB)* $\subseteq$ *M*(α)
  - *KB* = UMBC and JHU won
  - α = UMBC won
  - Then KB $\models$ α



Entailment in the Wumpus World

- Situation after detecting nothing in [1,1], move right, breeze in [2,1]
- Possible models for *KB* assuming only pits and restricting cells to {(1,3)(2,1)(2,2)}
- Two observations: ~B11, B12
- Three more propositional variables variables: P13, P21, P22
- ⇒ 8 possible models consistent with observations

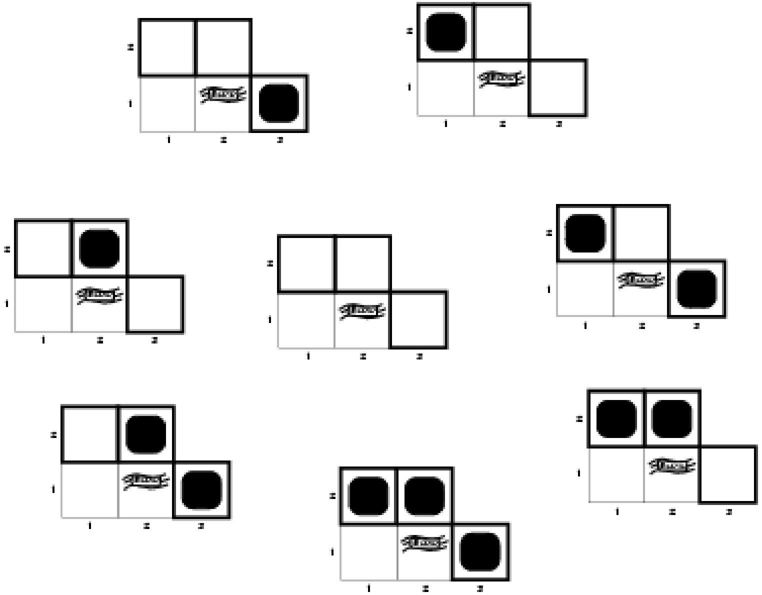


B11: breeze in (1,1)
P13: pit in (1,3)

Wumpus models

P13	P21	P22
F	F	F
F	F	T
F	T	F
F	T	T
T	F	F
T	F	T
T	T	F
T	T	T

Each row is a possible world



Some of these are inconsistent with the observed facts

Wumpus World Rules (1)

- **If a cell has a pit, then a breeze is observable in every adjacent cell**
- In propositional calculus we can not have rules with variables (e.g., forall X…)

P11 => B21

P11 => B12

P21 => B11

P21 => B22 …

If a pit in (1,1) then a breeze in (2,1), …

these also follow

~B21 => ~P11

~B12 => ~P11

~B11 => ~P21

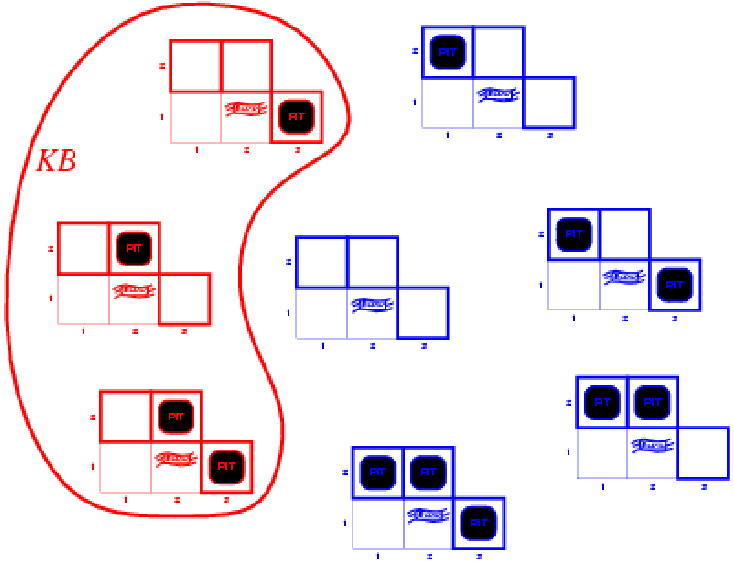
~B22 => ~P21

…

Wumpus models

Only **three** of the possible models are consistent with what we know

Any of the three might be the way the world really is.



***KB* = wumpus-world rules + observations**

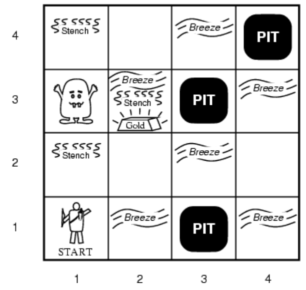
Wumpus World Rules (2)

- **Cell safe if it has neither a pit nor wumpus**

  OK11 => ~P11 ∧ ~W11

  OK12 => ~P12 ∧ ~W12 …

OK11: (1,1) is safe
W11: Wumpus in (1,1)



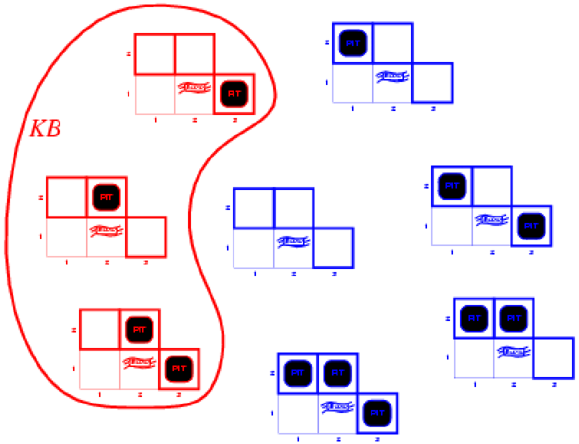
- From which we can derive the more useful “rules”

  P11 ∨ W11 => ~OK11

  P11 => ~OK11

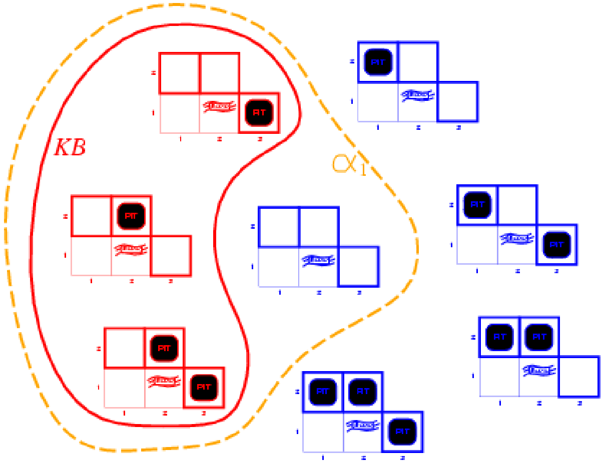
  W11 => ~OK11 …

Wumpus models



- *KB* = wumpus-world rules + observations

Wumpus models



- *KB* = wumpus-world rules + observations
- α_1 = “[1,2] is safe”
- *Since all models include* α_1
- *KB* $\models \alpha_1$, proved by **model checking**

Inference, Soundness, Completeness

- $KB \vdash_i \alpha$: sentence α can be derived (inferred) from *KB* by procedure *i*
- **Soundness:** *i* is sound if whenever $KB \vdash_i \alpha$, it is also true that $KB \models \alpha$
- **Completeness:** *i* is complete if whenever $KB \models \alpha$, it is also true that $KB \vdash_i \alpha$

- Preview: **first-order logic** is expressive enough to say almost anything of interest and has a **sound** and **complete** inference procedure

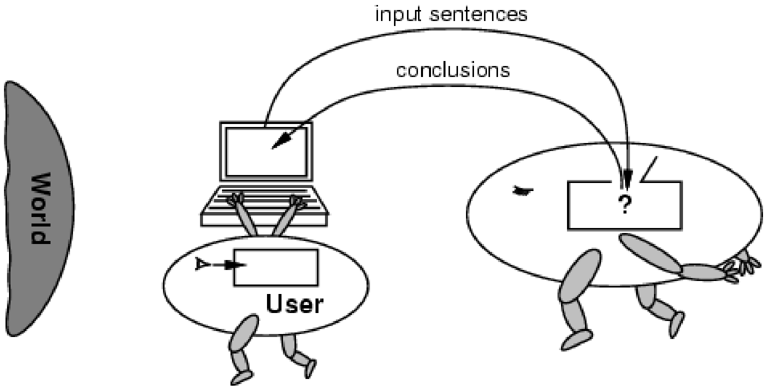
Soundness and completeness

- A *sound* inference method derives only entailed sentences
- A complete inference method can (eventually) derive any entailed sentence

- Analogous to the property of *soundness* and *completeness* in search

No independent access to the world

- Reasoning agents often get knowledge about world as a sequence of logical sentences and draw conclusions from them w/o independent access to the world
- Very important that the agents' reasoning is sound!
- Completeness is harder, but maybe less fundamental



Summary

- Intelligent agents need knowledge about world for good decisions
- Agent's knowledge stored in a knowledge base (KB) as **sentences** in a knowledge representation (KR) language
- Knowledge-based agents needs a **KB** & **inference mechanism**. They store sentences in KB, infer new sentences & use them to **deduce** which actions to take
- A **representation language** defined by its syntax & semantics, which specify structure of sentences & how they relate to facts of the world
- **Interpretation** of a sentence is fact to which it refers. If fact is part of the actual world, then the sentence is true

Propositional logic syntax

- Users specify
  - Set of propositional symbols (e.g., P,Q) whose values can be **True** or **False**
  - What each *means*, e.g.: P: "*It's hot*", Q: "*It's humid*"
- A sentence (well formed formula) is defined as:
  - Any symbol is a sentence
  - If S is a sentence, then **¬S** is a sentence
  - If S is a sentence, then **(S)** is a sentence
  - If S and T are sentences, then so are **(S ∨ T), (S ∧ T), (S → T),** and **(S ↔ T)**
  - A finite number of applications of the rules

Examples of PL sentences

- **Q**

  "It's humid"
- **Q → P**

  "If it's humid, then it's hot"
- **(P ∧ Q) → R**

  "If it's hot and it's humid, then it's raining"
- We're free to choose better symbols, e.g.:

  Hot for "It's hot"

  Humid for "It's humid"

  Raining for "It's raining"

Some terms

- Given the truth values of all symbols in a sentence, it can be ***evaluated*** to determine its **truth value** (True or False)
- We consider a **Knowledge Base** (KB) to be a set of sentences that are all True
- A **model** for a KB is a **possible world** – an assignment of truth values to propositional symbols that makes each KB sentence true

More terms

- A **valid sentence** or **tautology:** one that's **True** under all interpretations, no matter what the world is actually like or what the semantics is. Example: "It's raining or it's not raining" $(P \lor \neg P)$

- An **inconsistent sentence** or **contradiction:** a sentence that's **False** under all interpretations. The world is never like what it describes, as in "It's raining and it's not raining." $(P \land \neg P)$

Truth tables

Used to define meaning of logical connectives

[redacted]

Truth tables for the five logical connectives

P	$\neg P$
True	False
True	False
False	True
False	True

"not"

Truth tables

Used to define meaning of logical connectives

[redacted]

Truth tables for the five logical connectives

P	Q	$\neg P$	$P \wedge Q$
True	True	False	True
True	False	False	False
False	False	True	False
False	True	True	False

“and”

Truth tables

Used to define meaning of logical connectives

[redacted]

Truth tables for the five logical connectives

P	Q	$\neg P$	$P \wedge Q$	$P \vee Q$
True	True	False	True	True
True	False	False	False	True
False	False	True	False	False
False	True	True	False	True

(inclusive)
"or"

Truth tables

Used to define meaning of logical connectives

[redacted]

Truth tables for the five logical connectives

P	Q	$\neg P$	$P \wedge Q$	$P \vee Q$	$P \rightarrow Q$
True	True	False	True	True	True
True	False	False	False	True	False
False	False	True	False	False	True
False	True	True	False	True	True

implication
of q from p

Truth tables

Used to define meaning of logical connectives

[redacted]

Truth tables for the five logical connectives

P	Q	$\neg P$	$P \land Q$	$P \lor Q$	$P \rightarrow Q$	$P \leftrightarrow Q$
True	True	False	True	True	True	True
True	False	False	False	True	False	False
False	False	True	False	False	True	True
False	True	True	False	True	True	False

Bidirectional implication (aka, equivalence)
$(P \rightarrow Q) \land (Q \rightarrow P)$

Distribution of Negation

P	Q	$\neg P$	$P \vee Q$	$\neg P \wedge \neg Q$	$P \wedge Q$	$\neg P \vee \neg Q$
True	True	False	True	False	True	False
True	False	False	True	False	False	True
False	False	True	False	True	False	True
False	True	True	True	False	False	True

Examples

- What's the truth table of

$$\neg P \lor Q$$

P	Q	$\neg P$	$P \lor Q$
True	True	False	True
True	False	False	True
False	False	True	False
False	True	True	True

Examples

- What's the truth table of

$$\neg P \vee Q$$

P	Q	$\neg P$	$P \vee Q$	$\neg P \vee Q$
True	True	False	True	True
True	False	False	True	False
False	False	True	False	True
False	True	True	True	True

Examples

- What's the truth table of

$$\neg P \vee Q$$

P	Q	$\neg P$	$P \vee Q$	$\neg P \vee Q$	$P \rightarrow Q$
True	True	False	True	True	True
True	False	False	True	False	False
False	False	True	False	True	True
False	True	True	True	True	True

- What's the truth table of

$$(P \vee Q) \wedge \neg Q) \rightarrow P?$$

(Work it out on your own)

The implies connective: P→ Q

→ is a *logical connective*

- *P*→ *Q* is a **logical sentence** and has a truth value, i.e., is either **True** or **False**
- If the sentence is in a KB, it can be used by a rule (*Modus Ponens*) to infer that Q is True if P is True in the KB
- Given a KB where P=True and Q=True, we can derive/infer/prove that P→Q is True
- Note: P→ Q is equivalent to ~P∨Q

$P \rightarrow Q$

When is *P→Q* true? Check all that apply

- ❑ P=Q=true
- ❑ P=Q=false
- ❑ P=true, Q=false
- ❑ P=false, Q=true

$P \rightarrow Q$

When is *P→Q* true? Check all that apply

- ❑✓ P=Q=true
- ❑✓ P=Q=false
- ❑ P=true, Q=false
- ✓❑ P=false, Q=true

- We can get this from the truth table for →
- Note: in FOL it's much harder to prove that a conditional true, e.g., prime(x) → odd(x)

Knowledge Bases (KBs)

- **Literal**: a Boolean variable
- **Clause**: a disjunction of literals
  - If $l_1, \ldots, l_N$ are literals, then $l_1 \vee \cdots \vee \ l_N$ is a clause
  - Clauses don't need to contain *all* literals
- If a literal only appears with one polarity in any clauses it appears in (either as l_i or $\neg l_i$, but not both), then it's a **pure literal**

Knowledge Bases (KBs)

- A conjunction of **definite clauses**
- **Definite clause** (aka Strict Horn clause): a *body* implies a *head*
  - Form: $a_1 \wedge a_2 \wedge \cdots \wedge a_M \rightarrow h$
  - Body: $a_1 \wedge a_2 \wedge \cdots \wedge a_M$
  - Head: h
- If the body is empty, then the head is a *fact*

Q: Is
$A \vee B \vee \neg C$
a definite clause?

Knowledge Bases (KBs)

- A conjunction of **definite clauses**
- **Definite clause** (aka Strict Horn Clause): a *body* implies a *head*
  - Form: $a_1 \wedge a_2 \wedge \cdots \wedge a_M \rightarrow h$
  - Body: $a_1 \wedge a_2 \wedge \cdots \wedge a_M$
  - Head: h
- If the body is empty, then the head is a *fact*

Q: Is
$A \vee B \vee \neg C$
a definite clause?

A: No. Can you turn it into one?

Models for a KB

- KB: [P∨Q, P→R, Q→R]
- What are the sentences?

  s1: P∨Q

  s2: P→R

  s3: Q→R
- What are the propositional variables?

  P,Q, R
- What are the candidate models?

  1) Consider all **eight** possible assignments of T|F to P,Q, R

P	Q	R	s1	s2	s3
F	F	F	x	✓	✓
F	F	T	x	✓	✓
F	T	F	✓	✓	x
F	T	T	✓	✓	✓
T	F	F	✓	x	✓
T	F	T	✓	✓	✓
T	T	F	✓	x	x
T	T	T	✓	✓	✓

Here **x** means the model makes the sentence False and ✓means it doesn't make it False

Models for a KB

- KB: [P∨Q, P→R, Q→R]
- What are the sentences?
  s1: P∨Q
  s2: P→R
  s3: Q→R
- What are the propositional variables?
  P, Q, R
- What are the candidate models?
  1) Consider all possible assignments of T|F to P, Q, R
  2) Check truth tables for consistency, eliminating any row that does not make every KB sentence true

P	Q	R	s1	s2	s3
~~F~~	~~F~~	~~F~~	~~x~~	~~✓~~	~~✓~~
~~F~~	~~F~~	~~T~~	~~x~~	~~✓~~	~~✓~~
~~F~~	~~T~~	~~F~~	~~✓~~	~~✓~~	~~x~~
F	T	T	✓	✓	✓
~~T~~	~~F~~	~~F~~	~~✓~~	~~x~~	~~✓~~
T	F	T	✓	✓	✓
~~T~~	~~T~~	~~F~~	~~✓~~	~~x~~	~~x~~
T	T	T	✓	✓	✓

- Only 3 models are consistent with KB
- R true in all of them
- Therefore R is true and can be added to the KB

A simple example

The KB

$$P$$
$$Q \lor \neg R$$

The KB has 2 sentences.

The KB has 3 variables.

Models for the KB

P	Q	R	KB
T	T	F	T
T	T	T	T
T	F	F	T
T	F	T	F
F	T	F	F
F	T	T	F
F	F	T	F
F	F	F	F

The KB has 3 models. Each model has a value for every variable in the KB such every sentence evaluates to true.

Another simple example

The KB

$P \wedge Q$

$R \wedge \neg P$

The KB has 2 sentences.

The KB has 3 variables.

Models for the KB

P	Q	R

The KB has no models. There is no assignment of True or False to every variable that makes every sentence in the KB true