Artificial Intelligence

The Very Idea

Vasant G. Honavar
Dorothy Foehr Huck and J. Lloyd Huck Chair in Biomedical Data Sciences and Artificial Intelligence
Professor of Data Sciences, Informatics, Computer Science, Bioinformatics & Genomics and Neuroscience
Director, Artificial Intelligence Research Laboratory
Director, Center for Artificial Intelligence Foundations and Scientific Applications
Associate Director, Institute for Computational and Data Sciences
Pennsylvania State University

vhonavar@psu.edu
http://faculty.ist.psu.edu/vhonavar
http://ailab.ist.psu.edu

Agents that reason

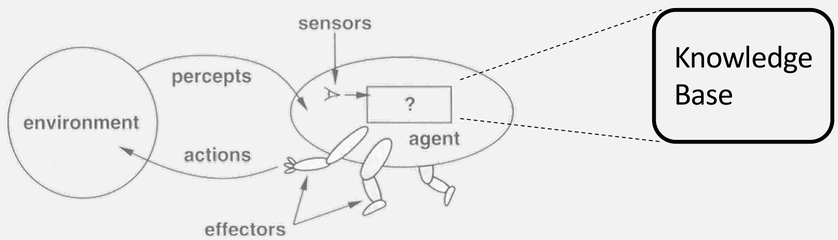
- With rule-based knowledge representation, we can perform rule-based reasoning
- However, the rules are heuristic in nature, and the conclusions need not be logically sound
- In logic-based systems, conclusions derived from the axioms universally hold, and provably correct (if the underlying inference algorithm is sound)
- Logic based systems can be made more or less expressive based on the type of logic used

Logic and AI

- ``Civilization advances by extending the number of important operations which we can perform without thinking of them. ”
  — Alfred North Whitehead
- ``It is unworthy of excellent men to lose hours like slaves in the labor of calculation which could safely be relegated to anyone else if machines were used’’.
  — Gottfried Leibniz
- ``He that cannot reason is a fool. He that will not is a bigot. He that dare not is a slave.’’
  — Andrew Carnegie

Deliberative Agents

- Can represent and reason with knowledge
- Exhibit logical rationality
- Derive conclusions that logically follow from the facts and only those that logically follow from the facts

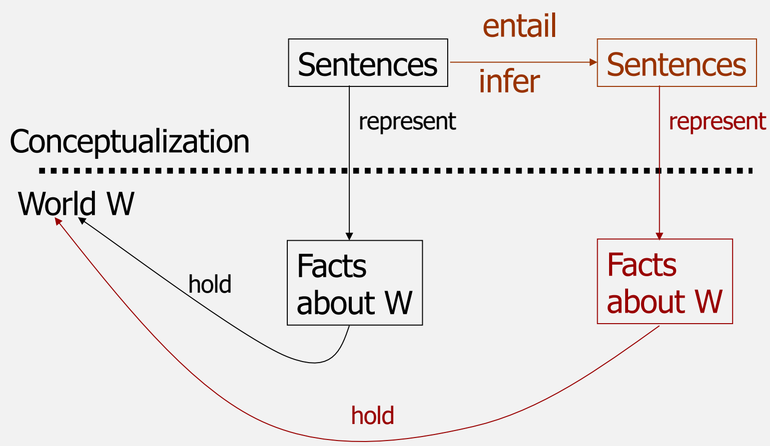


Knowledge representation (KR) is a surrogate

A declarative knowledge representation

- Encodes facts that are true in the world into sentences
- Reasoning is performed by manipulating sentences according to sound rules of inference
- The results of inference are sentences that correspond to facts that are true in the world
- The correspondence between facts that hold in the world and sentences that describe the world gives meaning to the representation
- Allows agents to substitute thinking for acting in the world
  - Known facts: The coffee is hot; coffee is a liquid; a hot liquid will burn your tongue;
  - Inferred fact: Coffee will burn your tongue

The nature of representation



Logic as a Knowledge Representation Formalism

Logic is a declarative language to:

- Assert sentences representing facts that hold in a real or imagined world *W* (these sentences are given the value true)
- Deduce the true/false values to sentences representing other aspects of *W*
- We shall see that Logical reasoning = computation
- Anticipated by Leibnitz, Hilbert
  - Can all truths be reduced to calculation?
  - Is there an effective procedure for determining whether or not a conclusion is a logical consequence of a set of facts?

Propositional Logic - Syntax

Propositional logic is a formal language with syntax and semantics

- Syntax refers to the structure or form of the sentences
- Semantics refers to the meaning of sentences

Syntax

- Basic units – propositions, e.g., $A, B, Tall, Short, Rich, Poor$ that can be True or False
- Propositions have no intrinsic meaning
- Logical connectives
  - $\wedge$ or logical AND
  - $\vee$ or logical OR
  - $\neg$ or logical negation or NOT
  - $\equiv$ or logical equivalence
  - $\rightarrow$ or logical implication

Propositional Logic - Syntax

Valid sentences include:

- Basic sentences
  - Propositions, e.g., $A, B, Rich, Poor$ that can be True or False
- Sentences that combine other sentences using logical connectives $\wedge$, $\vee$, $\neg$, $\rightarrow$
- If S_1 and S_2 are sentences, so are
  - $\neg S_1$, $\neg S_2$
  - $S_1 \wedge S_2$
  - $S_1 \vee S_2$
  - $S_1 \rightarrow S_2$

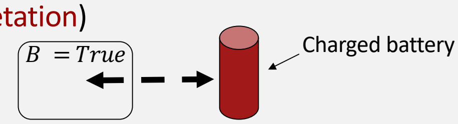
Note that this is a recursive definition

- We use extra-linguistic symbols like parenthesis to disambiguate e.g., $(A \wedge B) \vee (\neg B \wedge C)$

Propositional Logic - Semantics

A proposition (sentence)

- does not have intrinsic meaning
- gets its meaning from correspondence with properties of the world (interpretation)



e.g., proposition B denotes the fact that battery is charged

- There are two possible worlds – one in which battery is charged and one in which it is not
- The proposition B is True or False in a real or imagined world
- B is true in the world in which the battery is charged and false in the world in which it is not charged

Propositional Logic - Semantics

- Meaning of Logical connectives
  - $A \wedge B$ is True if both A and B are True
  - $A \vee B$ is True if A is True or B is True, or both A and B are True
  - $\neg A$ is True if and only if A is False and
  - $\neg A$ is False if and only if A is True,
  - $A \rightarrow B$ is equivalent to $\neg A \vee B$
    - $A \rightarrow B$ is True whenever A is False or B is True

A	B	$A \wedge B$
False	False	False
False	True	False
True	False	False
True	True	True

A	B	$A \vee B$
False	False	False
False	True	True
True	False	True
True	True	True

A	$\neg A$
False	True
True	True

A couple of notes about implication

- Unlike $\wedge$ and $\vee$, $\rightarrow$ is not commutative
  - $p \rightarrow q$ is not the same as $q \rightarrow p$
- The meaning of logical implication is not quite the same as the conversational meaning we assign to implication
  - $Study \rightarrow Pass$
  - If the antecedent is true, $\rightarrow$ has the usual meaning
  - If antecedent is false, then the implication is true regardless of the truth or falsity of the conclusion
  - In conversation when we say p implies q we suggest a causal relationship between p and q
  - Why? By design, $p \rightarrow q \equiv \neg p \vee q$
  - By design, the truth or falsity of a compound sentence is completely determined by the truth or falsity of the components of the sentence and not any other extraneous information

What can we infer in propositional logic?

- Propositional logic provides the machinery for us to determine
  - Whether or not some conclusion follows logically from a given set of assertions (facts or assumptions)
  - Provided both the conclusion and facts/assumptions are sentences in propositional logic
  - What does it mean for a conclusion to logically follow from a set of assertions?
- We shall see that Reasoning = computation
- Anticipated by Leibnitz, Hilbert
  - Can all truths be reduced to calculation?
  - Is there an effective procedure for determining whether or not a conclusion is a logical consequence of a set of facts?

Model theoretic or Tarskian Semantics

- Consider a logic with only two propositions:
  - $Rich, Poor$
  - denoting Tom is rich and Tom is poor respectively
- A model *M* is a subset of the set *A* of atomic sentences or propositions in the language
- Given this logic, we have

$$A = \{Rich, Poor\}$$

- The models correspond to all possible subsets of A

$$M_0 = \{\quad\}$$
$$M_1 = \{Rich\}$$
$$M_2 = \{Poor\}$$
$$M_3 = \{Rich, Poor\}$$

- The models denote possible worlds, that is, the possible states of affairs that one can describe or imagine in this logic

Exercise

$$M_0 = \{\quad\}$$
$$M_1 = \{Rich\}$$
$$M_2 = \{Poor\}$$
$$M_3 = \{Rich, Poor\}$$

Identify the models where the following sentences are true

$$Rich$$
$$Rich \vee Poor$$
$$Rich \wedge Poor$$
$$Rich \Rightarrow \neg Poor$$
$$\neg Rich \vee \neg Poor$$

Exercise

$$M_0 = \{\quad\}$$
$$M_1 = \{Rich\}$$
$$M_2 = \{Poor\}$$
$$M_3 = \{Rich, Poor\}$$

Identify the models where the following sentences are true

$$\begin{array}{rll} Rich & \text{is } True \text{ in} & M_1, M_3 \\ Rich \vee Poor & \text{is } True \text{ in} & M_1, M_2, M_3 \\ Rich \wedge Poor & \text{is } True \text{ in} & M_3 \\ Rich \Rightarrow \neg Poor & \text{is } True \text{ in} & M_0, M_1, M_2 \\ \neg Rich \vee \neg Poor & \text{is } True \text{ in} & M_0, M_1, M_2 \end{array}$$

Model theoretic or Tarskian Semantics

- The possible worlds are

$$M_0 = \{\quad\}, M_1 = \{Rich\}, M_2 = \{Poor\}, M_3 = \{Rich, Poor\}$$

- By a model *M* we mean the state of affairs in the world in which
  - every atomic sentence that is in *M* is *true* and
  - every atomic sentence that is not in *M* is *false*
- In M_0 Tom is neither rich nor poor
- In M_1 Tom is rich
- In M_2 Tom is poor
- In M_3 Tom is both rich and poor

Model theoretic or Tarskian Semantics

- We have

$$A = \{Rich, Poor\}$$

- The possible worlds are

$$M_0 = \{\quad\}$$
$$M_1 = \{Rich\}$$
$$M_2 = \{Poor\}$$
$$M_3 = \{Rich, Poor\}$$

- In M_0 Tom is neither rich nor poor
- In M_1 Tom is rich
- In M_2 Tom is poor
- In M_3 Tom is both rich and poor
- How could this be?

Model theoretic or Tarskian Semantics

- The possible worlds are
  $M_0 = \{\quad\}, M_1 = \{Rich\}, M_2 = \{Poor\}, M_3 = \{Rich, Poor\}$
- By a model *M* we mean the state of affairs in the world in which
  - every atomic sentence that is in *M* is *true* and
  - every atomic sentence that is not in *M* is *false*
- In M_0 Tom is neither rich nor poor $Rich$ is False and $Poor$ is False
- In M_1 Tom is rich: $Rich$ is True, $Poor$ is False
- In M_2 Tom is poor: $Poor$ is True, $Rich$ is False
- In M_3 Tom is both rich and poor: $Rich$ is True and $Poor$ is True
- How could this be?
- Because the propositions *Rich, Poor* have no intrinsic meaning!
- They get their meaning from correspondence with the states of the world

Model theoretic or Tarskian Semantics

- The possible worlds are

$$M_0 = \{\ \ \}, M_1 = \{Rich\}, M_2 = \{Poor\}, M_3 = \{Rich, Poor\}$$

- What if we wanted to ensure that the meaning of $Rich$ and *Poor* are mutually exclusive?
  - We must assert that Tom cannot be both rich and poor: $\neg(Rich \wedge Poor)$
- What if we wanted to assert that Tom cannot be neither rich nor poor?
  - We must assert that: $Rich \vee Poor$
- Hence, if we want to talk about possible worlds in which Tom is rich or poor and ensure that our assertions align with their intuitive meanings, we must constrain their meanings by the additional assertions $\neg(Rich \wedge Poor)$, $Rich \vee Poor$

Model theoretic or Tarskian Semantics

- The possible worlds are

$$M_0 = \{\quad\}, M_1 = \{Rich\}, M_2 = \{Poor\}, M_3 = \{Rich, Poor\}$$

- What if we wanted to ensure that the meaning of $Rich$ and *Poor* are mutually exclusive?
  - We must assert that Tom cannot be both rich and poor: $\neg(Rich \wedge Poor)$
- What if we wanted to assert that Tom cannot be neither rich nor poor?
  - We must assert that: $Rich \vee Poor$
- Hence, if we want to ensure that our logical assertions align with their intuitive meanings, we restrict their meanings by the additional assertions $\neg(Rich \wedge Poor)$, $Rich \vee Poor$

Some laws of propositional logic

- **Commutative law.**

  $p \wedge q \equiv q \wedge p$

  $p \vee q \equiv q \vee p$

- **Associative law**

  $(p \wedge q) \wedge r \equiv p \wedge (q \wedge r)$

  $(p \vee q) \vee r \equiv p \vee (q \vee r)$

- **Distributive law**

  $p \wedge (q \vee r) \equiv (p \wedge q) \vee (p \wedge r)$

  $p \vee (q \wedge r) \equiv (p \vee q) \wedge (p \vee r)$

- **De Morgan's Laws**

  $\neg(p \wedge q) \equiv (\neg p) \vee (\neg Q)$

  $\neg(p \vee q) \equiv (\neg p) \wedge (\neg q)$

- **Identity**

  $p \wedge \top \equiv p$

  $p \vee \bot \equiv p$

- **Tautology**

  $p \vee \neg p \equiv \top$

  Contradiction

  $p \wedge \neg p \equiv \bot$

What does it mean for a conclusion to logically follow from a given set of assertions?

- First, note that any set of assertions can be combined using $\wedge$ to obtain a single equivalent sentence
  - ``$A \wedge B$ is True and $\neg C \vee D$ is True '' is equivalent to
  - $(A \wedge B) \wedge (\neg C \vee D)$ is True
- Hence, it suffices to consider what it means for one sentence, say q, to logically follow from another, say, p

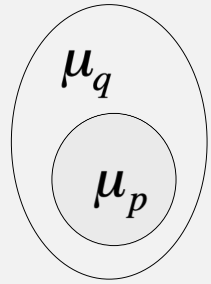
Proof Theory: Logical Entailment

- What does it mean for q to logically follow from p?
- We say that p entails q (written as $p \vDash q$) if q holds in every model in which p hold

$\mu_q =$ set of models in which q holds

$\mu_p =$ set of models in which p holds

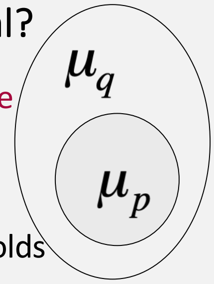
$p \models q$ if it is the case that $\mu_p \subseteq \mu_q$



To find out if $p \vDash q$, we can

- Enumerate μ_p, the set of models in which p holds
- Enumerate μ_q, the set of models in which q holds
- Check if $\mu_p \subseteq \mu_q$
- Note that entailment $\vDash$ is akin to what we conversationally mean by implication which is different from $\rightarrow$

What does it mean to be logically rational?



- Infer only those conclusions from one's knowledge base that are sanctioned by logical entailment
- To find out if $p \vDash q$, we can
  - Enumerate μ_p, the set of models in which p holds
  - Enumerate μ_q, the set of models in which q holds
  - Check if $\mu_p \subseteq \mu_q$
- Suppose you know that being human implies being mortal
- Then you find out that you are human
- Is it rational for you to believe that you are mortal?

What does it mean to be logically rational?

- Infer only those conclusions from one's knowledge base that are sanctioned by logical entailment
  - Suppose you know that being human implies being mortal
  - Then you find out that you are human
  - Is it rational for you to conclude that you are mortal?

What does it mean to be logically rational?

- Suppose you know that being human implies being mortal
- Then you find out that you are human
- Is it rational for you to conclude that you are mortal?

Let us construct a logic to find out

- Let H denote being human
- Let M denote being mortal
- Knowledge base: $H \rightarrow M, H$
- We need to check whether $H \wedge (H \rightarrow M) \vDash M$

How can we tell if $H \wedge (H \rightarrow M) \vDash M$?

Enumerate the models

$M_0 = \{\ \}, M_1 = \{H\}\ M_2 = \{M\}, M_3 = \{H, M\}$

Let p be the sentence $H \wedge (H \rightarrow M)$ and q be the sentence M

$\mu_H =$ the set of models in which H holds = $\{M_1, M_3\}$

$\mu_{H \rightarrow M}$ = the set of models in which $H \rightarrow M$ holds

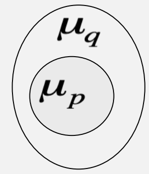
= the set of models in which $\neg H \vee M$ holds

$= \mu_{\neg H} \cup \mu_M = \{M_0, M_2\} \cup \{M_2, M_3\} = \{M_0, M_2, M_3\}$

$\mu_{H \wedge (H \rightarrow M)} = \mu_H \cap \mu_{H \rightarrow M} = \{M_1, M_3\} \cap \{M_0, M_2, M_3\} = M_3 = \mu_p$

$\mu_M = \{M_2, M_3\} = \mu_q$

How can we tell if $H \wedge (H \rightarrow M) \vDash M$?



Enumerate the models

$M_0 = \{\,\}, M_1 = \{H\}\ M_2 = \{M\}, M_3 = \{H, M\}$

Let p be the sentence $H \wedge (H \rightarrow M)$ and q be the sentence M

$\mu_p = M_3$

$\mu_q = \{M_2, M_3\}$

Clearly, $\mu_p \subseteq \mu_q$

Hence $p \vDash q$

Therefore $H \wedge (H \rightarrow M) \vDash M$

That is, given H and $H \rightarrow M$, it is logically rational to conclude M

What did we just do?

We just proved that

$H \wedge (H \rightarrow M) \models M$

- Note that we never really made use of the fact that H and M denote being human and being mortal respectively
- So long as our knowledge base has two sentences of the form α and $\alpha \rightarrow \beta$ hold, logic permits us to conclude that β holds as well
- This yields a logically sound rule of inference that we can mechanically apply to any knowledge base:

  Given $\alpha, \alpha \rightarrow \beta$, infer β
- This is the rule called Modus Ponens that Aristotle had introduced but without solid justification which we now have, thanks to Tarski

Validity and satisfiability, equivalence

- A sentence is valid if it is true in *all* models,
  - e.g., *True*, $A \vee \neg A$, $A \rightarrow A$, $(A \wedge (A \rightarrow B)) \rightarrow B$
- A sentence is satisfiable if it is true in *some* model
  - e.g., $A \vee B, C$
- A sentence is unsatisfiable if it is true in *no* models
  - e.g., $A \wedge \neg A$
- A useful result for proof by contradiction
  - $KB \vDash s$ if and only if $(KB \wedge \neg s)$ is unsatisfiable
- Two sentences are logically equivalent iff they are true in same set of models or $\alpha \equiv \beta$ iff $\alpha \vDash \beta$ and $\beta \vDash \alpha$.

Logical Rationality

- A logical agent A with a knowledge base KB_A is justified in inferring q if it is the case that $KB_A \vDash q$
- How can the agent A decide whether in fact $KB_A \vDash q$?
  - Model checking
    - Enumerate μ_{KB_A} i.e., all the models in which KB_A holds
    - Enumerate μ_q i.e., all the models in which q holds
    - Check whether $\mu_{KB_A} \subseteq \mu_q$
  - Inference algorithm based on inference rules
    - We saw one such inference rule that is provably sound:
      - Given $\alpha, \alpha \rightarrow \beta$, infer β

Searching for proofs: inference

- An inference rule $\{\alpha, \beta\} \vdash \gamma$ consists of
  - 2 sentence patterns α and β called the premises and
  - one sentence pattern γ called the conclusion
- Note the difference between $\models$ and $\vdash$
  - $\models$ is a semantic notion
  - $\vdash$ is a syntactic pattern matching procedure
- If α and β match two sentences of KB then
  - the corresponding sentence of the form γ can be inferred according to the rule
- Given one or more sound inference rules and a knowledge base *KB*
  - inference is the process of successively applying inference rules to *KB*
  - Each rule application adds its conclusion to *KB*
- This could involve forward chaining or backward chaining

Generalized Modus Ponens

$$\{\alpha, \beta\} \vdash \gamma$$

$$\{\, \overbrace{p \rightarrow q}\,, p \,\} \vdash q$$

$$\frac{a_1 \wedge a_2 \wedge a_3 \cdots \wedge a_{i-1} \wedge a_i \wedge a_{i+1} \cdots a_m \rightarrow q \qquad a_i}{a_1 \wedge a_2 \wedge a_3 \cdots \wedge a_{i-1} \wedge a_{i+1} \cdots a_m \rightarrow q}$$

Generalized Modus Ponens

$$\{\alpha, \beta\} \vdash \gamma$$

$$\{\, p \rightarrow q \,, p \,\} \vdash q$$

Generalized Modus Ponens

$$\frac{\begin{array}{c} a_1 \wedge a_2 \wedge a_3 \cdots \wedge a_{i-1} \wedge a_i \wedge a_{i+1} \cdots a_m \rightarrow q \\ b_1 \wedge b_2 \wedge \cdots \wedge b_n \rightarrow a_i \end{array}}{a_1 \wedge a_2 \wedge a_3 \cdots \wedge a_{i-1} \wedge a_{i+1} \cdots a_m \wedge b_1 \wedge b_2 \wedge \cdots \wedge b_n \rightarrow q}$$

Example: Inference using Modus Ponens

KB:

$$BatteryOK \wedge BulbsOK \rightarrow HeadlightsWork$$

$$BatteryOK \wedge StarterOK \wedge \neg EmptyGasTank \rightarrow EngineStarts$$

$$EngineStarts \wedge \neg FlatTire \wedge HeadlightsWork \rightarrow CarOK$$

$$BatteryOK, BulbsOK, StarterOK\ , \neg EmptyGasTank, \neg FlatTire$$

Query:

$CarOK?$

Example: Forward-chaining using Modus Ponens

$$BatteryOK \wedge BulbsOK \rightarrow HeadlightsWork$$
$$BatteryOK, BulbsOK$$

$$HeadlightsWork$$

$$BatteryOK \wedge StarterOK \wedge \neg EmptyGasTank \rightarrow EngineStarts$$
$$BatteryOK, StarterOK, \neg EmptyGasTank$$

$$EngineStarts$$

$$EngineStarts \wedge \neg FlatTire \wedge HeadlightsWork \rightarrow CarOK$$
$$EngineStarts, \neg FlatTire, HeadlightsWork$$

$$CarOK$$

Exercise: Use backward chaining to prove $CarOK$

KB:

$$BatteryOK \wedge BulbsOK \rightarrow HeadlightsWork$$

$$BatteryOK \wedge StarterOK \wedge \neg EmptyGasTank \rightarrow EngineStarts$$

$$EngineStarts \wedge \neg FlatTire \wedge HeadlightsWork \rightarrow CarOK$$

$$BatteryOK, BulbsOK, StarterOK, \neg EmptyGasTank, \neg FlatTire$$

Query:

$CarOK?$

Not all inference rules are sound

- *Modus ponens*

  $\alpha \rightarrow \beta, \alpha \vdash \beta$

Modus ponens derives only inferences sanctioned by entailment

Modus ponens is sound

- *Loony tunes*

  $Friday \vdash \beta$

Loony tunes can derive inferences that are not sanctioned by entailment

Loony tunes is not sound

Soundness and Completeness of an inference rule $\vdash$

- We write $p \vdash q$ to denote that that p can be inferred from q using the inference rule $\vdash$

An inference rule $\vdash$ is said to be

- Sound if whenever $p \vdash q$, it is also the case that $p \models q$
- Complete if whenever $p \models q$, it is also the case that $p \vdash q$

Soundness and Completeness of an inference rule ⊢

- We can show that *modus ponens* is sound, but *not* complete unless the KB is *Horn* i.e., the KB can be written as a collection of sentences of the form
- $a_1 \wedge a_2 \wedge a_3 \ldots a_{i-1} \wedge a_i \wedge a_{i+1} \wedge a_{i+2} \ldots \wedge a_m \rightarrow b$
- Where each a_i and b are atomic sentences

Unsound inference rules are not necessarily useless!

Abduction (Charles Peirce) is not sound, but useful in diagnostic reasoning or hypothesis generation

$$\frac{\begin{array}{l} p \Rightarrow q \\ q \end{array}}{p}$$

$$\frac{\begin{array}{l} BlockedArtery \Rightarrow HeartAttack \\ HeartAttack \end{array}}{BlockedArtery}$$

Constructing proofs

- Finding proofs can be cast as a search problem
- Search can be
  - forward (forward chaining) to derive *goal* from *KB*
  - or backward (backward chaining) from the *goal*
- Searching for proofs
  - Involves repeated application of applicable inference rules.
  - Is sound if it uses only sound inference rules
  - Can be more efficient than enumerating models in practice with the use of suitable heuristics
- Propositional logic is monotonic
  - Inference steps can only add inferred facts
  - An inferred fact once added is never deleted
  - A theorem once proven can never be disproven (barring error in proof)

Soundness and Completeness

- An inference algorithm starts with the KB and applies applicable inference rules until the desired conclusion is reached
- An inference algorithm is sound if it uses a sound inference rule
- An inference algorithm is complete if
  - It uses a complete inference rule and
  - a complete search procedure

Completeness of Modus Ponens for Propositional Logic

- Modus Ponens is not complete for Propositional Logic
- Suppose that all classes at some university meet either Mon/Wed/Fri or Tue/Thu.
- The AI course meets at 4 PM in the afternoon
- Jane has volleyball practice Thursdays and Fridays at that time.
- Can Jane take AI?

$$1. MWFAI4pm \lor TRAI4pm$$
$$2\ TRAI4pm \land JaneBusyR4pm \rightarrow JaneConflictAI$$
$$3. MWFAI4pm \land JaneBusy4pm \rightarrow JaneConflictAI$$
$$4. JaneBusyR4pm$$
$$5. JaneBusyF4pm$$

Completeness of Modus Ponens for Propositional Logic

- Modus Ponens is not complete for Propositional Logic
- Can Jane take AI?

$$1.\, MWFAI4pm \;\vee TRAI4pm$$
$$2\; TRAI4pm \wedge JaneBusyR4pm \rightarrow JaneConflictAI$$
$$3.\, MWFAI4pm \wedge JaneBusy4pm \rightarrow JaneConflictAI$$
$$4.\, JaneBusyR4pm$$
$$5.\, JaneBusyF4pm$$

- Of course not!
- Try proving this using Modus Ponens
- You can't!
- Why?

Completeness of Modus Ponens for Propositional Logic

$$1. MWFAI4pm \lor TRAI4pm$$
$$2\ TRAI4pm \land JaneBusyR4pm \rightarrow JaneConflictAI$$
$$3. MWFAI4pm \land JaneBusy4pm \rightarrow JaneConflictAI$$
$$4. JaneBusyR4pm$$
$$5. JaneBusyF4pm$$

We can use Modus Ponens to establish

$$2\&4: TRAI4pm \rightarrow JaneConflictAI$$
$$3\&4: MWFAI4pm \rightarrow JaneConflictAI$$

But Modus Ponens can't take us further to conclude $JaneConflictAI$!

- Modus Ponens is not complete for Propositional Logic (except in the restricted case when the KB is Horn)
- However, we can generalize Modus Ponens to obtain a sound and complete inference rule for Propositional Logic

Proof

- The proof of a sentence α from a set of sentences *KB* is the derivation of α obtained through a series of applications of sound inference rules to *KB*

Soundness and Completeness of Forward Chaining

- An inference algorithm starts with the KB and applies applicable inference rules until the desired conclusion is reached
- An inference algorithm is sound if it uses a sound inference rule
- An inference algorithm is complete if
  - It uses a complete inference rule and
  - a complete search procedure
- Forward chaining using Modus Ponens is sound and complete for Horn knowledge bases (i.e., knowledge bases that contain only Horn clauses)

Forward vs. backward chaining

- FC is data-driven, automatic, unconscious processing,
  - Akin to day dreaming…
- May do lots of work that is irrelevant to the goal
- BC is goal-driven, appropriate for problem-solving
  - e.g., Where are my keys? How do I get into a PhD program?
- The run time of FC is linear in the size of the KB.
- The run time of BC can be, in practice, much less than linear in size of *KB*

Resolution principle

Resolution is sound and complete for propositional *KB*

Given

$\neg a_1 \vee \ldots \vee \neg a_{i-1} \vee \neg a_i \vee \neg a_{i+1} \vee \ldots \vee \neg a_M \vee q_1 \vee q_2 \ldots \vee q_N$

$b_1 \vee \ldots \vee b_L \vee c_1 \vee \ldots \vee c_{j-1} \vee c_j \vee c_{j+1} \ldots \vee c_K$

If $a_i = c_j$ then we can conclude:

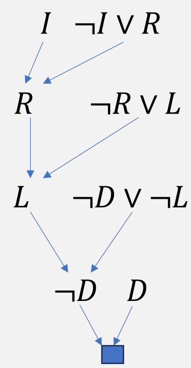
$\neg a_1 \ldots a_{i-1} \vee \neg a_{i+1} \vee \ldots \vee \neg a_M \vee q_1 \vee q_2 \ldots \vee q_N \vee b_1 \vee \ldots \vee b_L \vee c_1 \vee \ldots \vee c_{j-1} \vee c_{j+1} \ldots \vee c_K$

Applying resolution

- Transform *KB* into an *equivalent* Conjunctive normal form (CNF)
  - Each sentence in *KB* is a disjunction of literals or their negations using known logical equivalences
  - *KB* is a conjunction of disjunctions
- Any propositional KB can be converted into CNF

Example: Applying resolution

- Given KB: $I, D, \neg R \vee L, \neg D \vee \neg L$
- Negated query: $\neg I \vee R$



Transformation to Clause Form (CNF)

Example:

$(A \vee \neg B) \rightarrow (C \wedge D)$

1. Eliminate $\rightarrow$

   $\neg(A \vee \neg B) \vee (C \wedge D)$
2. Reduce scope of $\neg$ using De Morgan's laws

   $(\neg A \wedge B) \vee (C \wedge D)$
3. Distribute $\vee$ over $\wedge$

   $(\neg A \vee (C \wedge D)) \wedge (B \vee (C \wedge D))$

   $(\neg A \vee C) \wedge (\neg A \vee D) \wedge (B \vee C) \wedge (B \vee D)$

KB in the form of a set of clauses or conjunction of disjunctions (CNF):

$\{\neg A \vee C, \neg A \vee D, B \vee C, B \vee D\}$

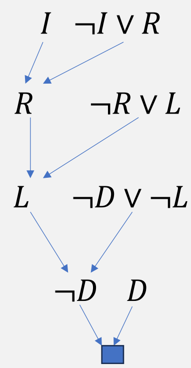
Proof

- The proof of a sentence α from a set of sentences *KB* is the derivation of α obtained through a series of applications of sound inference rules to *KB*
- $KB \models \alpha$ if and only if $\{KB, \neg\alpha\}$ is unsatisfiable

  (contradiction, $T \rightarrow F$, ■, empty sentence)
- Proving α from *KB* is equivalent to deriving a contradiction from *KB* augmented with the negation of α

Example: Applying resolution

- Given KB: $I, D, \neg R \vee L, \neg D \vee \neg L$
- Negated query: $\neg I \vee R$



Example: Applying Resolution

- Suppose that all classes at some university meet either Mon/Wed/Fri or Tue/Thu. The AI course meets at 4 PM in the afternoon, and Jane has volleyball practice Thursdays and Fridays at that time.
- Does Jane have a conflict with AI? Assume not.

$$1. MWFAI4pm \lor TRAI4pm$$
$$2\ TRAI4pm \land JaneBusyR4pm \rightarrow JaneConflictAI$$
$$3. MWFAI4pm \land JaneBusyF4pm \rightarrow JaneConflictAI$$
$$4. JaneBusyR4pm$$
$$5. JaneBusyF4pm$$
$$6. \neg JaneConflictAI$$

Example: Applying Resolution

$1.\ MWFAI4pm \lor TRAI4pm$

$2\ \neg TRAI4pm \lor \neg JaneBusyR4pm \lor JaneConflictAI$

$3.\ \neg MWFAI4pm \lor \neg JaneBusy4pm \lor JaneConflictAI$

$4.\ JaneBusyR4pm$

$5.\ JaneBusyF4pm$

$6.\ \neg JaneConflictAI$

Proof ———————————————————————————————————— —

$2, 4.\ \neg TRAI4pm \lor JaneConflictAI$

$3, 5.\ \neg MWFAI4pm \lor JaneConflictAI$

$1, (2,4).\ MWFAI4pm \lor JaneConflictAI$

$(3,5), (1, (2,4).\ JaneConflictAI \lor JaneConflictAI$

$6, ((3,5), (1, (2,4)).\ JaneConflictAI$

$6, (6, ((3,5), \left(1, (2,4)\right)).\ \blacksquare$

Exercise: Prove $CarOK$ using resolution

KB:

$$BatteryOK \wedge BulbsOK \rightarrow HeadlightsWork$$

$$BatteryOK \wedge StarterOK \wedge \neg EmptyGasTank \rightarrow EngineStarts$$

$$EngineStarts \wedge \neg FlatTire \wedge HeadlightsWork \rightarrow CarOK$$

$$BatteryOK, BulbsOK, StarterOK, \neg EmptyGasTank, \neg FlatTire$$

Query:

$CarOK?$