Logic and AI (First-order and Propositional logics)

Dr. Animesh Chaturvedi
Assistant Professor: IIIT Dharwad
Young Researcher: Heidelberg Laureate Forum
Postdoc: King's College London & The Alan Turing Institute
PhD: IIT Indore MTech: IIITDM Jabalpur

Logic and AI

- Would like our AI to have knowledge about the world, and logically draw conclusions from it
- Search algorithms generate successors and evaluate them, but do not "understand" much about the setting
- Example question: is it possible for a chess player to have all her pawns and 2 queens?
    - Search algorithm could search through tons of states to see if this ever happens, but…

Syntax

- What do well-formed sentences in the knowledge base look like?
- A BNF grammar:
- *Symbol* $\rightarrow$ P, Q, R, …,
- *Sentence* $\rightarrow$ True | False | *Symbol* | NOT(*Sentence*) | (*Sentence* AND *Sentence*) | (*Sentence* OR *Sentence*) | (*Sentence* => *Sentence*)
- We will drop parentheses sometimes, but formally they really should always be there

Semantics

- A model specifies which of the proposition symbols are true and which are false
- Given a model, I should be able to tell you whether a sentence is true or false
- Truth table defines semantics of operators:
- Given a model, can compute truth of sentence recursively with these

a	b	NOT(a)	a AND b	a OR b	a => b
false	false	true	false	false	true
false	true	true	false	true	true
true	false	false	false	true	false
true	true	false	true	true	true

Basic terms

- A **valid sentence** or **tautology** is a sentence that is True under all interpretations, no matter what the world is actually like or how the semantics are defined. Example: "It's raining or it's not raining."
- An **inconsistent sentence** or **contradiction** is a sentence that is False under all interpretations. The world is never like what it describes, as in "It's raining and it's not raining."
- **P entails Q**, written P |= Q, means that whenever P is True, so is Q. In other words, all models of P are also models of Q.
- TwoIsAnEvenNumber OR ThreeIsAnOddNumber
  - is true (not exclusive OR)
- TwoIsAnOddNumber => ThreeIsAnEvenNumber
  - is true (if the left side is false it's always true)

All of this is assuming those symbols are assigned their natural values…

Tautologies

- A sentence is a tautology if it is true for any setting of its propositional symbols
- (P OR Q) OR (NOT(P) AND NOT(Q)) is a tautology

P	Q	P OR Q	NOT(P) AND NOT(Q)	(P OR Q) OR (NOT(P) AND NOT(Q))
false	false	false	true	true
false	true	true	false	true
true	false	true	false	true
true	true	true	false	true

Logical equivalences

- Two sentences are logically equivalent if they have the same truth value for every setting of their propositional variables
- P OR Q and NOT(NOT(P) AND NOT(Q)) are logically equivalent
- Tautology = logically equivalent to True

P	Q	P OR Q	NOT(NOT(P) AND NOT(Q))
false	false	false	false
false	true	true	true
true	false	true	true
true	true	true	true

Famous logical equivalences

- (a OR b) ≡ (b OR a) *commutativity*
- (a AND b) ≡ (b AND a) *commutativity*
- ((a AND b) AND c) ≡ (a AND (b AND c)) *associativity*
- ((a OR b) OR c) ≡ (a OR (b OR c)) *associativity*
- NOT(NOT(a)) ≡ a *double-negation elimination*
- (a => b) ≡ (NOT(b) => NOT(a)) *contraposition*
- (a => b) ≡ (NOT(a) OR b) *implication elimination*
- NOT(a AND b) ≡ (NOT(a) OR NOT(b)) *De Morgan*
- NOT(a OR b) ≡ (NOT(a) AND NOT(b)) *De Morgan*
- (a AND (b OR c)) ≡ ((a AND b) OR (a AND c)) *Distributivity*
- (a OR (b AND c)) ≡ ((a OR b) AND (a OR c)) *Distributivity*

Reasoning patterns

- Obtain new sentences directly from some other sentences in knowledge base according to reasoning patterns
- If we have sentences a and a => b , we can correctly conclude the new sentence b
  - This is called modus ponens
- A **Horn sentence** or **Horn clause** has the form:

  $P1 \wedge P2 \wedge P3 \ldots \wedge Pn \rightarrow Qm$ *where* $n>=0$, m *in* $\{0,1\}$
- If we have a AND b , we can correctly conclude a
- All of the logical equivalences from before also give reasoning patterns

$$\boxed{(P \rightarrow Q) = (\neg P \vee Q)}$$

Propositional logic (zeroth-order logic

- **Logical constants**: true, false
- **Propositional symbols**: P, Q, S, … (**atomic sentences**)
- Wrapping **parentheses**: (…)
- Sentences are combined by **connectives**:

$\wedge$ …and	[conjunction]
$\vee$ …or	[disjunction]
$\Rightarrow$…implies	[implication / conditional]
$\Leftrightarrow$..is equivalent	[biconditional]
$\neg$ …not	[negation]

- **Literal**: atomic sentence or negated atomic sentence

Examples of PL sentences

- User defines a set of propositional symbols, like P and Q.
- User defines the **semantics** of each propositional symbol:
  - P means "It is hot"
  - Q means "It is humid"
  - R means "It is raining"
- $(P \wedge Q) \rightarrow R$
  "If it is hot and humid, then it is raining"
- $Q \rightarrow P$
  "If it is humid, then it is hot"
- A better way:
  - Hot = "It is hot"
  - Humid = "It is humid"
  - Raining = "It is raining"

Propositional logic (PL)

- A simple language useful for showing key ideas and definitions
- A sentence (well formed formula) is defined as follows:
  - A symbol is a sentence
  - If S is a sentence, then $\neg$S is a sentence
  - If S is a sentence, then (S) is a sentence
  - If S and T are sentences, then (S $\vee$ T), (S $\wedge$ T), (S $\rightarrow$ T), and (S $\leftrightarrow$ T) are sentences
  - A sentence results from a finite number of applications of the above rules

A BNF grammar of sentences in propositional logic

```
S := <Sentence> ;
<Sentence> := <AtomicSentence> | <ComplexSentence> ;
<AtomicSentence> := "TRUE" | "FALSE" |
                    "P" | "Q" | "S" ;
<ComplexSentence> := "(" <Sentence> ")" |
          <Sentence> <Connective> <Sentence> |
          "NOT" <Sentence> ;
<Connective> := "NOT" | "AND" | "OR" | "IMPLIES" | "EQUIVALENT" ;
```

Modus ponens

- The form of a modus ponens argument resembles a syllogism, with two premises and a conclusion:
  - If P, then Q.
  - P.
  - Therefore, Q.
- It is a deductive argument form and rule of inference. It can be summarized as "P implies Q. P is true. Therefore Q must also be true."

Modus ponens Example

- for all x: Loves(John, x)
  - John loves every thing
- for all y: (Loves(y, Jane) => FeelsAppreciatedBy(Jane, y))
  - Jane feels appreciated by every thing that loves her
- Can infer from this:
  - FeelsAppreciatedBy(Jane, John)
- Here, we used the substitution {x/Jane, y/John}
  - Note we used different variables for the different sentences
- General UNIFY algorithms for finding a good substitution

Sound rules of inference

- Here are some examples of sound rules of inference
  - *A rule is sound if its conclusion is true whenever the premise is true*
- Each can be shown to be sound using a truth table

RULE	PREMISE	CONCLUSION
Modus Ponens	$A, A \rightarrow B$	B
And Introduction	A, B	$A \wedge B$
And Elimination	$A \wedge B$	A
Double Negation	$\neg\neg A$	A
Unit Resolution	$A \vee B, \neg B$	A
Resolution	**$A \vee B, \neg B \vee C$ $A \vee C$**	

Soundness of modus ponens

A	**B**	**A → B**	**OK?**
True	True	True	√
True	False	False	√
False	True	True	√
False	False	True	√

Example

- Consider the problem of representing the following information:
  - Every person is mortal.
  - Confucius is a person.
  - Confucius is mortal.
- How can these sentences be represented so that we can infer the third sentence from the first two?
- In PL we have to create propositional symbols to stand for all or part of each sentence.
  - P = "person"; Q = "mortal"; R = "Confucius"
  - so the above 3 sentences are represented as:
  - $P \rightarrow Q$; $R \rightarrow P$; $R \rightarrow Q$
- Although the third sentence is entailed by the first two, we needed an explicit symbol, R, to represent an individual, Confucius, who is a member of the classes "person" and "mortal"
- To represent other individuals we must introduce separate symbols for each one, with some way to represent the fact that all individuals who are "people" are also "mortal"

Propositional logic is a weak language

- Hard to identify "individuals" (e.g., Mary, 3)
- Can't directly talk about properties of individuals or relations between individuals (e.g., "Bill is tall")
- Generalizations, patterns, regularities can't easily be represented (e.g., "all triangles have 3 sides")
- First-Order Logic (abbreviated FOL or FOPC) is expressive enough to concisely represent this kind of information

FOL adds relations, variables, and quantifiers, e.g.,

- *"Every elephant is gray"*: $\forall$ x (elephant(x) $\rightarrow$ gray(x))
- *"There is a white alligator"*: $\exists$ x (alligator(X) ^ white(X))

Limitations and Problems of propositional logic

- No notion of objects
- No notion of relations among objects
- Some English statements are hard to model in propositional logic.
- Zeroth-order logic (Propositional Logic) is
  - first-order logic without variables or quantifiers.

First-order logic

- First-order logic (FOL) models the world in terms of
  - **Objects,** which are things with individual identities
  - **Properties** of objects that distinguish them from other objects
  - **Relations** that hold among sets of objects
  - **Functions,** which are a subset of relations where there is only one “value” for any given “input”
- Examples:
  - Objects: Students, lectures, companies, cars …
  - Relations: Brother-of, bigger-than, outside, part-of, has-color, occurs-after, owns, visits, precedes, …
  - Properties: blue, oval, even, large, …
  - Functions: father-of, best-friend, second-half, one-more-than …

Reasoning about many objects at once

- Variables: x, y, z, … can refer to multiple objects
- New operators "for all" and "there exists"
  - Universal quantifier and existential quantifier
- for all x: CompletelyWhite(x) => NOT(PartiallyBlack(x))
  - Completely white objects are never partially black
- there exists x: PartiallyWhite(x) AND PartiallyBlack(x)
  - There exists some object in the world that is partially white and partially black

Inference rules

- **Logical inference** is used to create new sentences that logically follow from a given set of predicate calculus sentences (KB).
- An inference rule is **sound** if every sentence X produced by an inference rule operating on a KB logically follows from the KB. (That is, the inference rule does not create any contradictions)
- An inference rule is **complete** if it is able to produce every expression that logically follows from (is entailed by) the KB. (Note the analogy to complete search algorithms.)

Limitations of First-Order Logic

- We are **not** allowed to reason in general about relations and functions
- We need higher-order logic (which is more powerful).
- "If a property is inherited by children, then for any thing, if that property is true of it, it must also be true for any child of it"
- Axioms: basic facts about the domain, our "initial" knowledge base
- Theorems: statements that are logically derived from axioms

Thinking Rationally

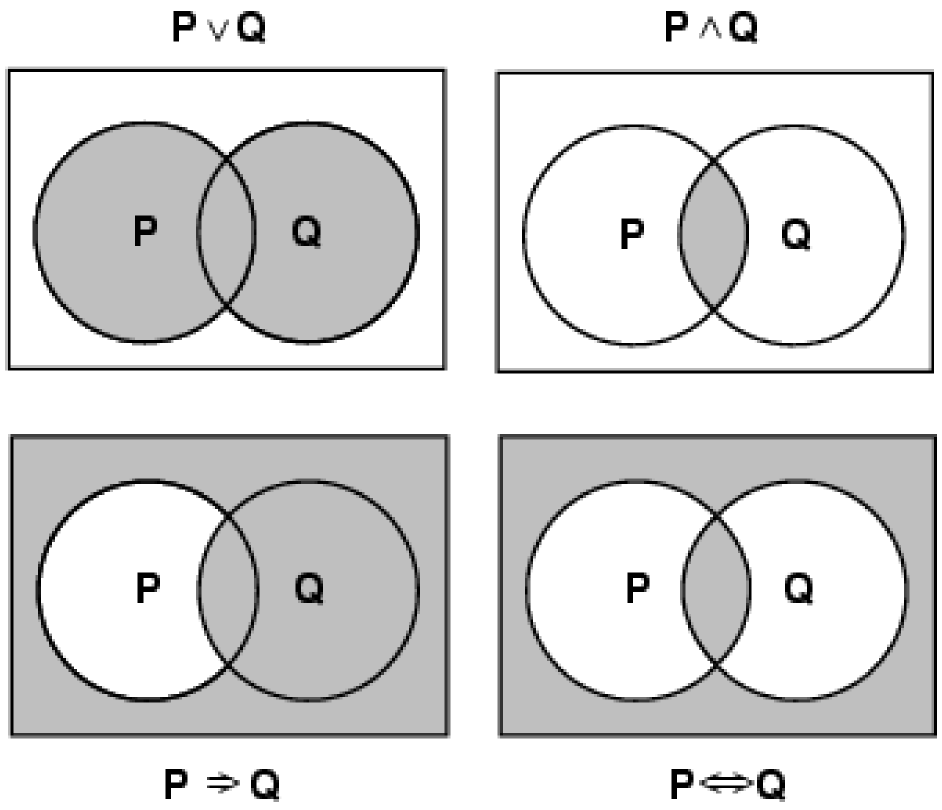
- **Right Thinking:** "arguments structures that always gave correct conclusions given correct premises"
- **Syllogism:** "Socrates is a man; all men are mortal therefore Socrates is mortal."
- **Field of Logic:** Laws of thought were supposed to govern the operation of the mind.
  - **Logical notation** to find solution to a problem.
  - Finding solution to all kinds of things in the world and the relations between them.
- **Issue 1:** Not easy to take informal knowledge and state it in the formal terms required by Logical notation, particularly when the knowledge is less than 100% certain.
- **Issue 2:** There is a big difference between being able to solve a problem "in principle'" and doing so in practice. Means proposing algorithm and it's coding are different problems.
- Power of the representation and reasoning systems

Machine Acting V/S Thinking?

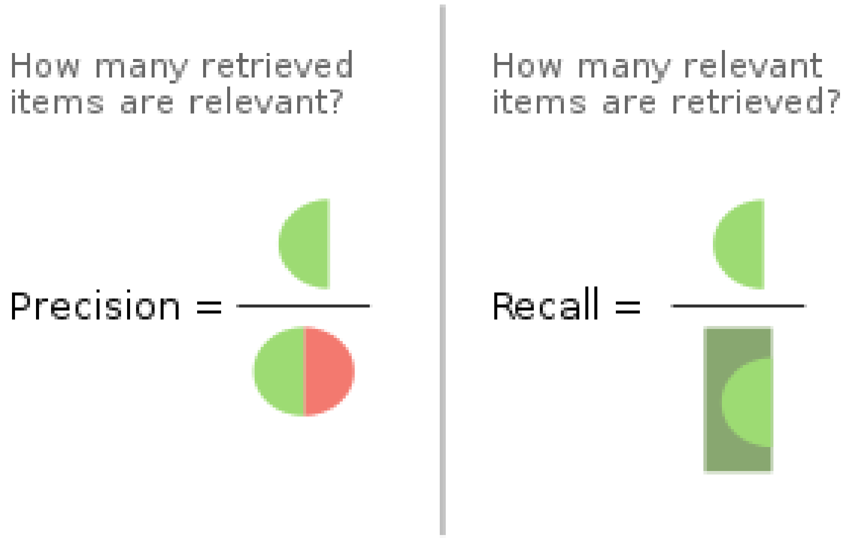
- Weak AI Hypothesis vs. Strong AI Hypothesis
  - Weak Hyp: machines could act as if they are intelligent
  - Strong Hyp: machines that act intelligent have to think intelligently too
- Hypothesis:
  - "a supposition or **propos**ed explanation made on the basis of limited evidence as a starting point for further investigation"
  - "a **propos**ition made as a basis for reasoning, without any assumption of its truth"
- Machine learning:
  - "algorithms to build model using data for training, which makes machine capable enough to make **predic**tions or decisions without being explicitly programmed"
- What is Machine learning Hypothesis?

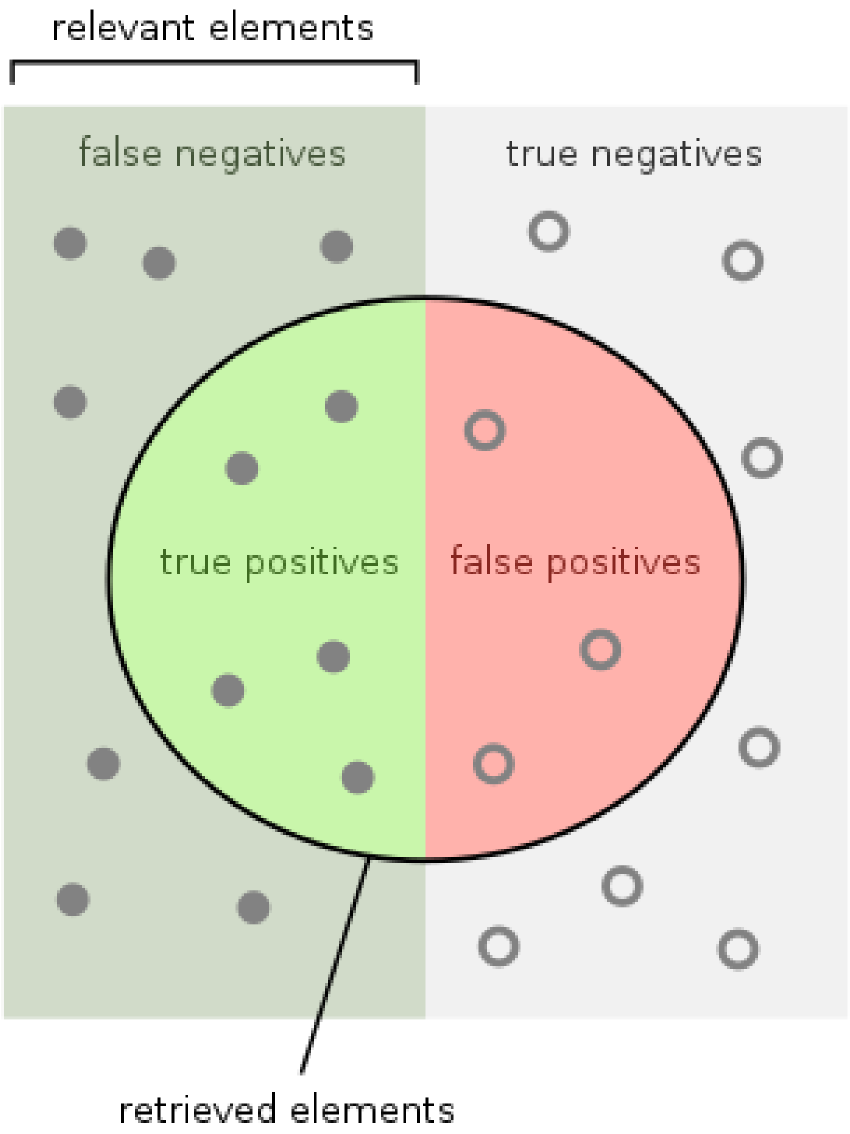
Models of complex sentences

- The meaning or **semantics** of a sentence determines its **interpretation**.
- Given the truth values of all symbols in a sentence, it can be "evaluated" to determine its **truth value** (True or False).
- A **model** for a KB is a "possible world" (assignment of truth values to propositional symbols) in which each sentence in the KB is True.



Precision and Recall





Precision and Recall

- A set of retrieved documents
  - (e.g. the list of documents produced by a web search engine for a query)
- A set of relevant documents
  - (e.g. the list of all documents on the internet that are relevant for a certain topic)

$$\text{precision} = \frac{|\{\text{relevant documents}\} \cap \{\text{retrieved documents}\}|}{|\{\text{retrieved documents}\}|}$$

$$\text{recall} = \frac{|\{\text{relevant documents}\} \cap \{\text{retrieved documents}\}|}{|\{\text{relevant documents}\}|}$$

Logic and Artificial Intelligence

1. Logic and Artificial Intelligence
2. John McCarthy and Common Sense Logicism
3. Nonmonotonic Reasoning and Nonmonotonic Logics
4. Reasoning about Action and Change
5. Causal Reasoning
6. Spatial Reasoning
7. Reasoning about Knowledge
8. Towards a Formalization of Common Sense
9. Logical Approaches to Natural Language and Communication
10. Taxonomic Representation and Reasoning
11. Contextual Reasoning
12. Prospects for a Logical Theory of Practical Reason

Stanford Encyclopedia of Philosophy

Logic and Artificial Intelligence

Summary

- The process of deriving new sentences from old one is called **inference**.
  - **Sound** inference processes derives true conclusions given true premises
  - **Complete** inference processes derive all true conclusions from a set of premises
- A **valid sentence** is true in all worlds under all interpretations
- If an implication sentence can be shown to be valid, then—given its premise—its consequent can be derived
- Different logics make different **commitments** about what the world is made of and what kind of beliefs we can have regarding the facts
  - Logics are useful for the commitments they do not make because lack of commitment gives the knowledge base engineer more freedom
- **Propositional logic** commits only to the existence of facts that may or may not be the case in the world being represented
  - It has a simple syntax and simple semantics. It suffices to illustrate the process of inference
  - Propositional logic quickly becomes impractical, even for very small worlds

References

- Vincent Conitzer, Logic CPS 270: Artificial Intelligence http://www.cs.duke.edu/courses/fall08/cps270/
- Vincent Conitzer, First-Order Logic CPS 270: Artificial Intelligence http://www.cs.duke.edu/courses/fall08/cps270/
- Some material adopted from notes and slides by Tim Finin, Marie desJardins, Andreas Geyer-Schulz and Chuck Dyer
- Wikipedia contents
  - https://en.wikipedia.org/wiki/Precision_and_recall
  - https://en.wikipedia.org/wiki/Confusion_matrix
- Stuart Russel, and Peter Norvig. "Artificial intelligence: A modern approach. Third edit." Upper Saddle River, New Jersey 7458 (2015). http://aima.cs.berkeley.edu/

ขอบคุณ

Thai

Grazie

Italian

תודה רבה

Hebrew

धन्यवादः

Sanskrit

ಧನ್ಯವಾದಗಳು

Kannada

Ευχαριστώ

Greek

Thank You

English

Gracias

Spanish

Спасибо

Russian

Obrigado

Portuguese

شكراً

Arabic

https://sites.google.com/site/animeshchaturvedi07

Merci

French

多謝

Traditional Chinese

धन्यवाद

Hindi

Danke

German

多谢

Simplified Chinese

நன்றி

Tamil

Tamil

ありがとうございました

Japanese

감사합니다

Korean