CSCE 580: Introduction to AI

Lecture 5: Formal Represention and Logic

PROF. BIPLAV SRIVASTAVA, AI INSTITUTE

3RD SEP 2024

Carolinian Creed: "I will practice personal and academic integrity."

Credits: **Copyrights of all material reused acknowledged**

Organization of Lecture 5

- Introduction Segment
  - Recap of Lecture 4
- Main Segment
  - Logic – First Order
  - Inferencing
  - Representation in the Large: ConceptNet, Cyc
  - Trust Issues with Knowledge Representation
- Concluding Segment
  - Course Project Discussion
  - About Next Lecture – Lecture 6
  - Ask me anything

Resume Exercise – Programming: Word Processing

1. Read your resume in text and get list of words, call resume_words
2. Plot a histogram of top 100 words, i.e., bar graph of words and counts
3. Have a list of common words – stop words
   1. Articles: a, an, the
   2. Propositions, determiners
4. Remove stop words from resume_words
5. Plot a histogram of top 100 words, i.e., bar graph of words and counts
6. Observation: Notice any difference?

Resume Exercise – Programming: Word Tag Cloud

- Task 1
  - Take your resume
  - Create word tag cloud for your resume
- Task 2
  - Take all resumes in folder
  - Create word tag cloud for multiple (5 or more, even all) resumes
  - Inspect and compare images of tasks 1 and 2
- Observation: Notice any difference?

Sample Code for Reference:
https://github.com/biplav-s/course-nl-f22/blob/main/sample-code/l1-wordcloud/FirstLook_ClassSyllabusData.ipynb

Resume Exercise - Report

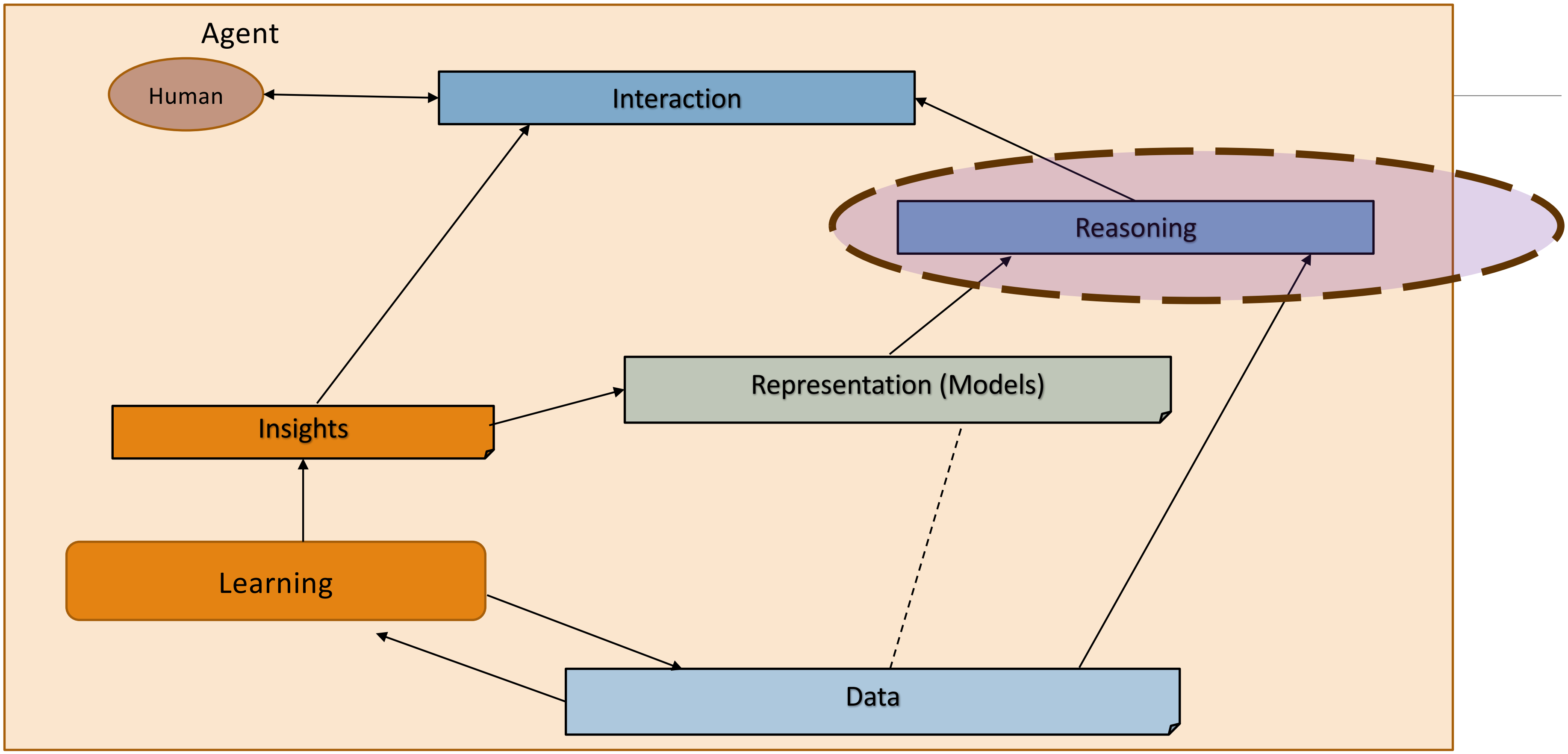
- T1: Put the code in your Github in sub-folder: “resume-exercise”
- T2: Document the finding on Github or Google drive as .pdf (.docx optional) (Create a Google doc in your Google drive called “Resume-Exercise”. In it,)
  - Show the two histograms
    - (top 100 and top-100 after stop words)
    - Comment on any difference you see
  - Show the two word tag clouds
    - (your resume and of the class)
    - Comment on any difference you see
- Do it by next Thursday – Sep 5, 2024

Introduction Section

Recap of Lecture 4

- Representing World Knowledge
- Logic (Propositional)
- Inferencing (Propositional)
- Code setup for AIMA examples

Relationship Between Main AI Topics



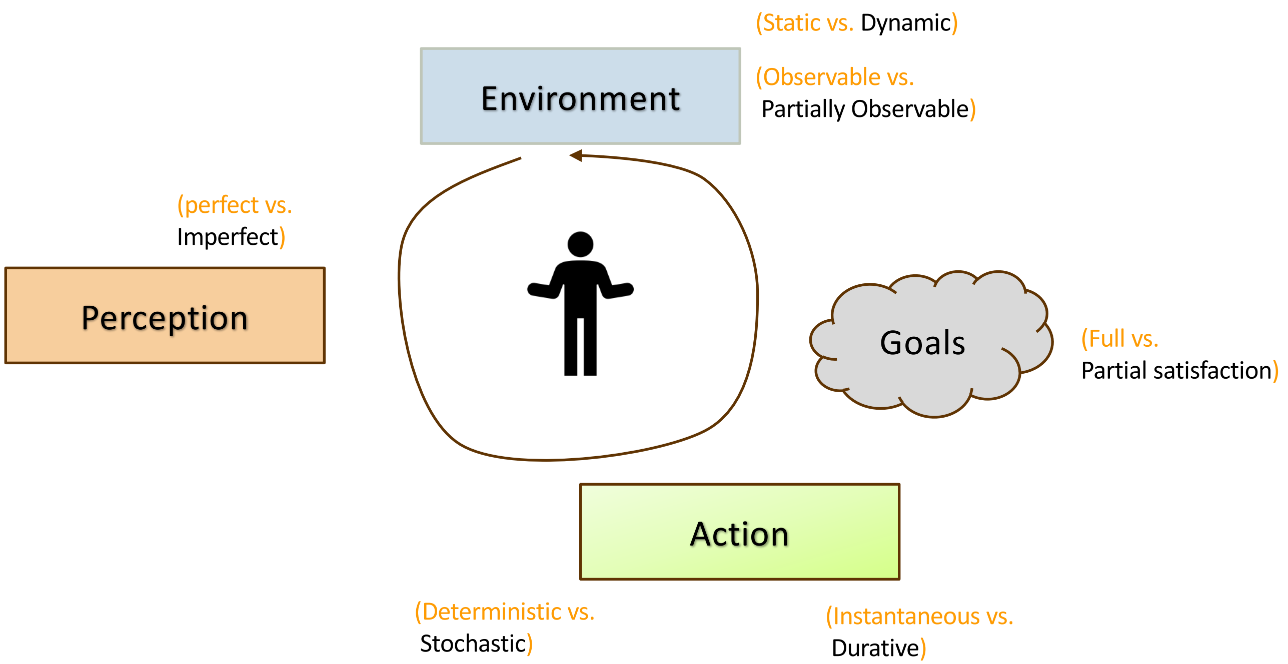
Where We Are in the Course

CSCE 580/ 581 – In This Course

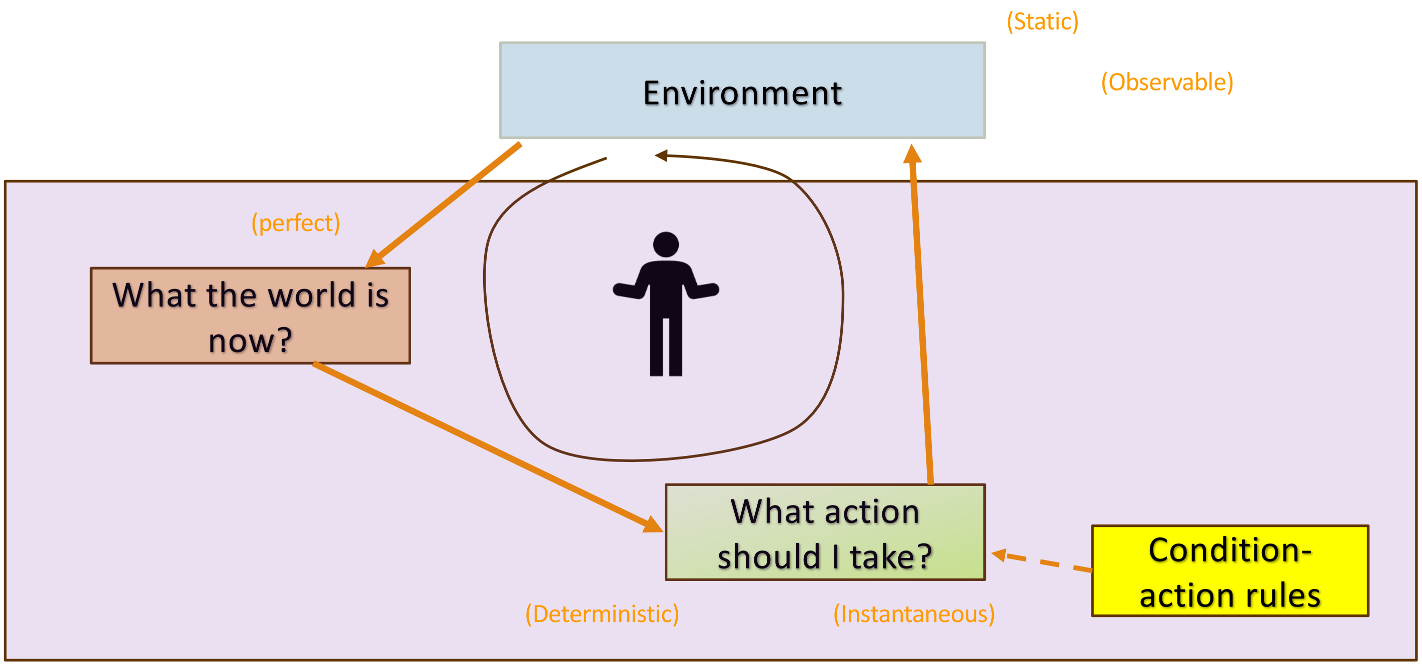
- Week 1: Introduction, Aim: Chatbot / Intelligence Agent
- Weeks 2-3: Data: Formats, Representation and the Trust Problem
- Week 4-5: Search, Heuristics - Decision Making
- Week 6: Constraints, Optimization – Decision Making
- Week 7: Classical Machine Learning – Decision Making, Explanation
- Week 8: Machine Learning - Classification
- Week 9: Machine Learning - Classification – Trust Issues and Mitigation Methods
- Topic 10: Learning neural network, deep learning, Adversarial attacks
- Week 11: Large Language Models – Representation, Issues
- Topic 12: Markov Decision Processes, Hidden Markov models - Decision making
- Topic 13: Planning, Reinforcement Learning – Sequential decision making
- Week 14: AI for Real World: Tools, Emerging Standards and Laws; Safe AI/ Chatbots

Main Section

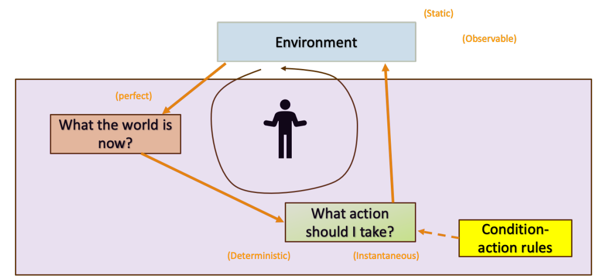
Intelligent Agent Model



Intelligent Agent – Simple Knowledge Based



KB Agent Procedure



function KB-AGENT(*percept*) **returns** an *action*
    static: *KB*, a knowledge base
        t, a counter, initially 0, indicating time

    TELL(*KB*, MAKE-PERCEPT-SENTENCE(*percept*, *t*))
    action ← ASK(*KB*, MAKE-ACTION-QUERY(*t*))
    TELL(*KB*, MAKE-ACTION-SENTENCE(*action*, *t*))
    $t \leftarrow t + 1$
    return *action*

Source: Russell & Norvig, AI: A Modern Approach

First Order Predicate Logic (FOPL)

Concepts

Constants: a, b, student123, teacher94
- Name of a specific object.

Variables: X, Y.
- Refer to an object without naming it.

Predicates: Father, Before
- Relationships between objects. May be many and may not be unique. Objects are specified as arguments (arity of a predicate).

Functions: father-of
- Mapping from objects to objects. Mapping must be present and be unique. Objects are specified as arguments (arity of a predicate).

Terms: dad-of(organism33), leftLeg(John)
- A logical expression that refers to an object

Atomic Sentences: in(dad-of(dog33), food6)
- Can be true or false
- Correspond to propositional symbols P, Q

Objects
Relations
Functions

Adapted from:
a) Dan Weld's AI course (CSE 573, Univ. of Washington)
b) Russell & Norvig, AI: A Modern Approach

FOPL - Syntax

Sentence	→	*AtomicSentence*
	|	*Sentence Connective Sentence*
	|	*Quantifier Variable,. . . Sentence*
	|	¬ *Sentence*
	|	*(Sentence)*
AtomicSentence	→	*Predicate(Term, . . .)* | *Term = Term*
Term	→	*Function(Term, . . .)*
	|	*Constant*
	|	*Variable*
Connective	→	$\Rightarrow$ | $\wedge$ | $\vee$ | $\Leftrightarrow$
Quantifier	→	$\forall$ | $\exists$
Constant	→	*A* | *X* | *John* | · · ·
Variable	→	a | *x* | *s* | · · ·
Predicate	→	*Before* | *HasColor* | *Raining* | · · ·
Function	→	*Mother* | *LeftLegOf* | · · ·

BNF (Backus-Naur Form) grammar
of sentences in FOPL

Source: Russell & Norvig, AI: A Modern Approach

Connectives and Quantifiers

Logical connectives: and, or, not, =>

Quantifiers:

- $\forall$: Forall
- $\exists$: There exists

Examples:

1. All students: $\forall$ **students**
2. All students are university members:
   $\forall$x Student(x) => UniversityMember(x)
   (For all x, if x is a student, then x is a UniversityMember)
3. A phone: **$\exists$x Phone(x)**
4. John has a phone:
   $\exists$x Phone(x) $\wedge$ Owns(John,x)
   (There exists a phone such that John owns it.)

Connections / Equivalences

$$
\begin{array}{lcl}
\forall x \ \neg P & = & \neg\exists x \ P \\
\neg\forall x \ P & = & 3x \ \neg P \\
\forall x \ P & = & \neg\exists x \ \neg P \\
\exists x \ \mathrm{P} & = & \neg\forall x \ \neg P
\end{array}
\qquad\qquad
\begin{array}{lcl}
\neg P \wedge \neg Q & = & \neg(P \vee Q) \\
\neg(P \wedge Q) & = & \neg P \vee \neg Q \\
P \wedge Q & = & \neg(\neg P \vee \neg Q) \\
P \vee Q & = & \neg(\neg P \wedge \neg Q)
\end{array}
$$

Derivable from De Morgan's law about sets:
(A ∪ B)' = A' ∩ B' and (A ∩ B)' = A' ∪ B'

Source: Russell & Norvig, AI: A Modern Approach

Comparing Syntax - FOPL and Propositional Logic

Sentence — *AtomicSentence* *ComplexSentence*

AtomicSentence — ***True*** | ***False***
| *P* *Q* *R* ...

ComplexSentence — (*Sentence*)
| *Sentence Connective Sentence*
| ¬*Sentence*

Connective — A | V | ⇔ | ⇒

Source: Russell & Norvig, AI: A Modern Approach

Sentence — *AtomicSentence*
| *Sentence Connective Sentence*
| *Quantifier Variable,. . . Sentence*
| ¬ *Sentence*
| (*Sentence*)

AtomicSentence — *Predicate*(*Term*, . . .) *Term* = *Term*

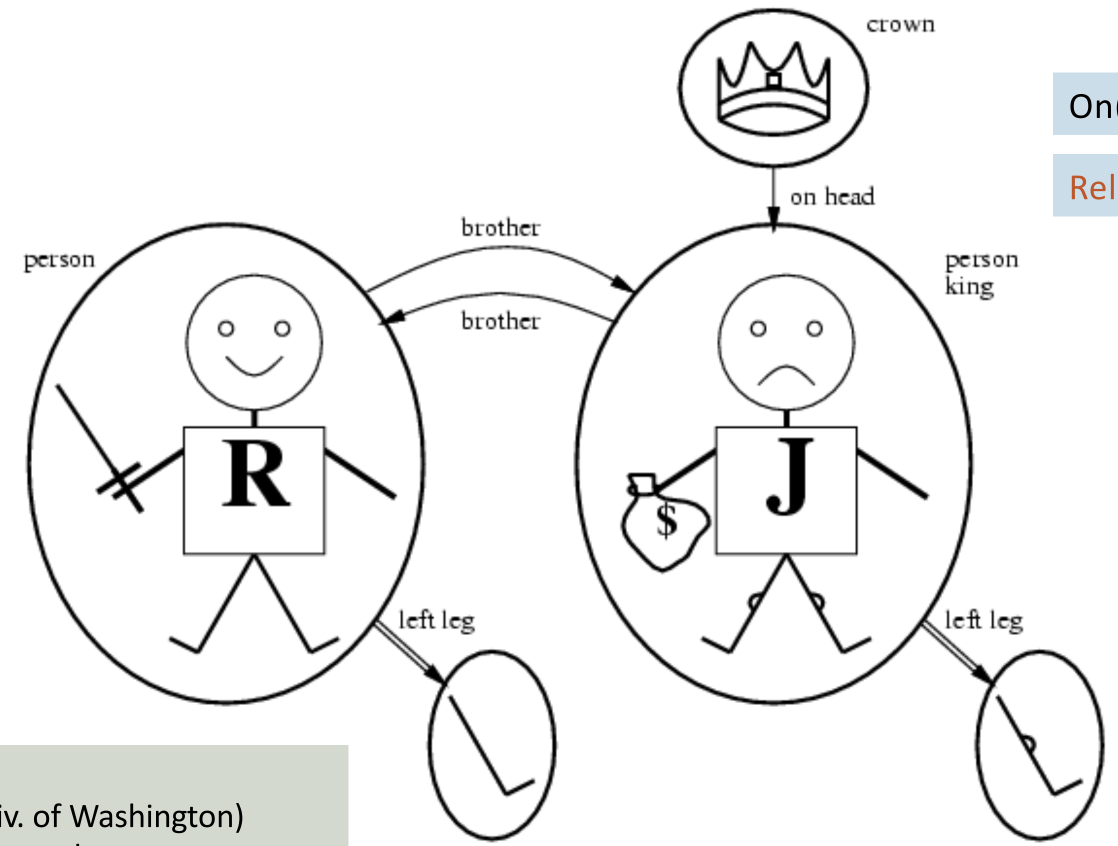
Term → *Function*(*Term*, . . .)
| *Constant*
\ *Variable*

Connective → ⇒ | A V | ⇔
Quantifier → V | 3
Constant → *A* \ *X*\ \ *John* | · · ·
Variable → a | *x* *s* • • •
Predicate → *Before* \ *HasColor* \ *Raining* \ · · ·
Function → *Mother* \ *LeftLegOf* \ · · ·

FOPL Semantics – Models and Interpretations

Richard, John
Constants

Leg(p,l)
Functions



On(x,y) King(p)
Relations

Adapted from:
a) Dan Weld's AI course (CSE 573, Univ. of Washington)
b) Russell & Norvig, AI: A Modern Approach

Interpretations - Mappings from Syntactic tokens → Model elements

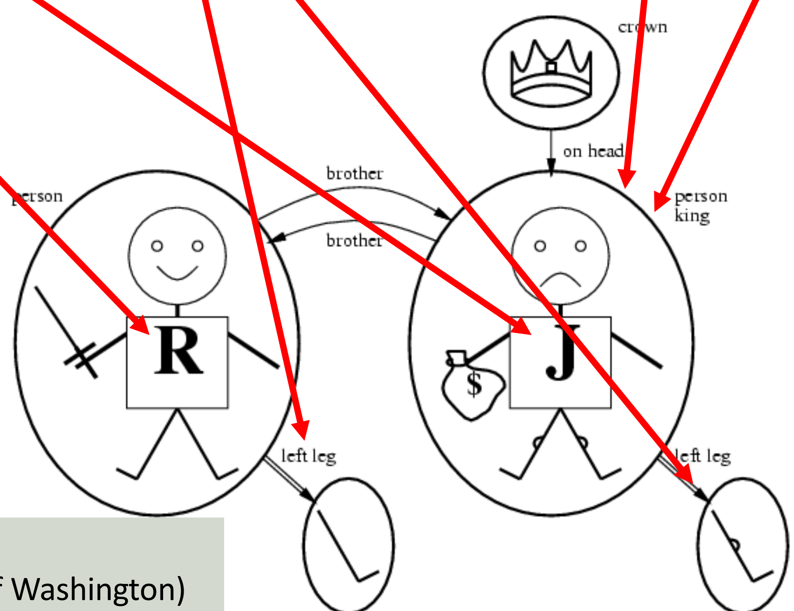
Richard John
Constants

Leg(p,l)
Functions

On(x,y) King(p)

Relations

Depiction of one possible interpretation



Adapted from:
a) Dan Weld's AI course (CSE 573, Univ. of Washington)
b) Russell & Norvig, AI: A Modern Approach

Satisfiability, Validity, & Entailment

- S is **valid** if it is true in all interpretations
- S is **satisfiable** if it is true in some interpretations
- S is **unsatisfiable** if it is false for all interpretations
- S1 **entails** S2 if forall interpretations where S1 is true, S2 is also true

Source: Dan Weld's AI course (CSE 573, Univ. of Washington

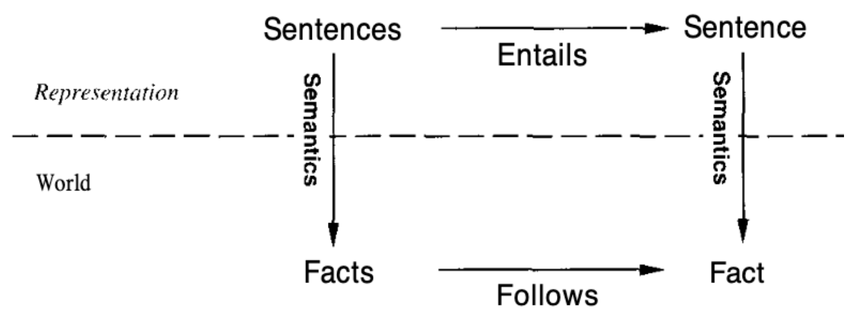
Comparing - Propositional Logic and FOPL

Ontology	Facts (P, Q)	Objects, Properties, Relations
Syntax	Atomic sentences Connectives	Variables & quantification Sentences have structure: terms father-of(mother-of(X)))
Semantics	Truth Tables	Interpretations *(Much more complicated)*
Inference Algorithm	DPLL, GSAT Fast in practice	Unification Forward, Backward chaining Prolog, theorem proving
Complexity	NP-Complete	Semi-decidable

Source: Dan Weld's AI course (CSE 573, Univ. of Washington

Formal Logic

- Properties of Logic System
  - **Soundness**: if it produces only true statements
  - **Completeness**: if it produces all true statements
  - **Consistency**: if it does not produce a sentence and its negation



Language	Ontological Commitment (What exists in the world)	Epistemological Commitment (What an agent believes about facts)
Propositional logic	facts	true/false/unknown
First-order logic	facts, objects, relations	true/false/unknown
Temporal logic	facts, objects, relations, times	true/false/unknown
Probability theory	facts	degree of belief 0…1
Fuzzy logic	degree of truth	degree of belief 0…1

Credits:
- Russell & Norvig, AI - A Modern Approach
- Deepak Khemani - A First Course in AI

Example: Course Selection

Example Situation – Course Selection

- A person wants to pass an academic program in two majors: A and B
- There are three subjects available: A, B and C, each with three levels (*1, *2, *3). There are thus 9 courses: A1, A2, A3, B1, B2, B3, C1, C2, C3
- To graduate, at least one course at beginner (*1) level is needed in major(s) of choice(s), and two courses at intermediate levels (*2) are needed
- Answer questions
  - Q1: How many minimum courses does the person have to take ?
  - Q2: Can a person graduate in 2 majors studying 3 courses only?
  - …

Representation – Propositional Example

- Domain Description: “There are three subjects: A, B and C, each with three levels (*1, *2, *3).”
- Representation
  - has_studied_courseA1: yes – student has taken course; no – student has not taken
  - has_studied_courseA2
  - has_studied_courseA3
  - has_studied_courseB1
  - has_studied_courseB2
  - has_studied_courseB3
  - has_studied_courseC1
  - has_studied_courseC2
  - has_studied_courseC3

LowerThan_Course_A1_CourseA2
LowerThan_Course_A2_CourseA3
LowerThan_Course_B1_CourseB2
LowerThan_Course_B2_CourseB3
LowerThan_Course_C1_CourseC2
LowerThan_Course_AC_CourseC3

- Previous statements set did not capture hierarchy between levels; new sentences would not have followed the reality in the world. Need more statements – LowerThan as shown.

Representation – FOPL Example

- Domain Description: “There are three subjects: A, B and C, each with three levels (*1, *2, *3).”
- Representation
  - has_studied (?x , ?y)
    - ?x: course name // A, B, C
    - ?y: course level // 1, 2, 3
  - lower_than _level(?x, ?y)
    - ? x: 1, 2
    - ?y: 2, 3

Revisiting Formal Representations: Ontologies

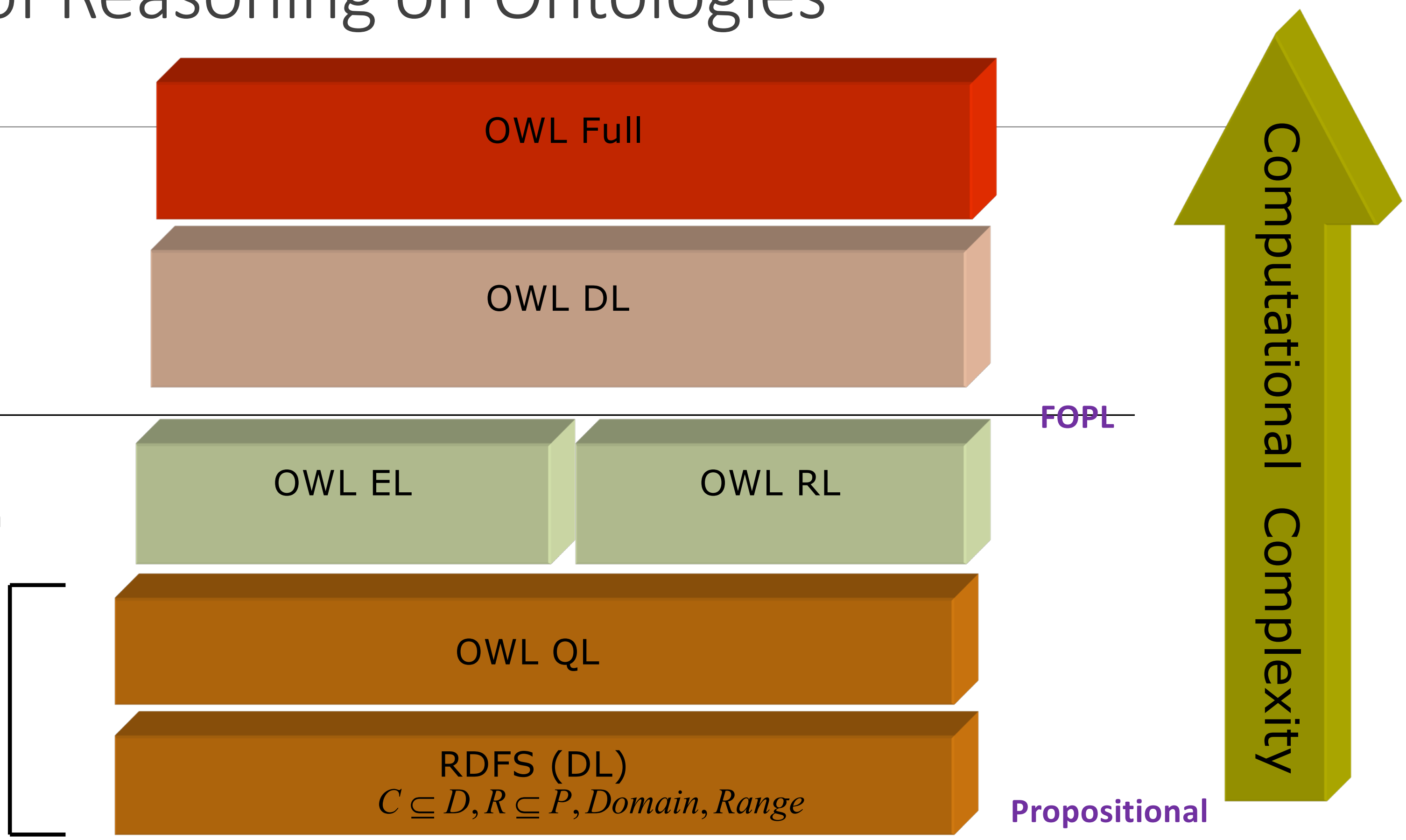
Challenge of Reasoning on Ontologies

- Undecidable

- Data Complexity open (NP-hard)
- Optimized tableau-based reasoners

- PTime-complete Data Complexity
- Efficient implementation on top of rule engine

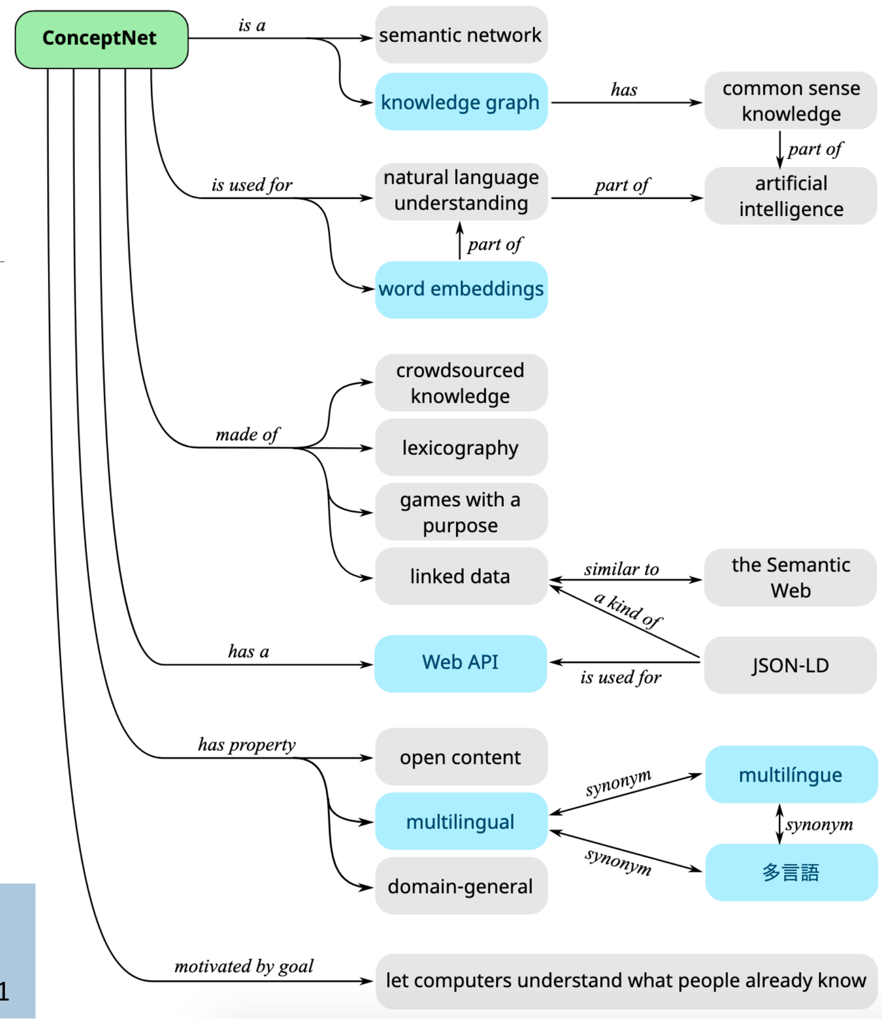
- LogSpace Data Complexity
- Efficient evaluation on top of RDBMS



Formal Representation in the Large

ConceptNet

- NLP focused graph knowledge graph that connects words and phrases of natural language with labeled edges.
- Concepts collected from experts, crowd-sourcing, and games with a purpose
- Supports multiple languages



Details: http://conceptnet.io/, https://github.com/commonsense/conceptnet5/wiki,
Paper: https://www.aaai.org/ocs/index.php/AAAI/AAAI17/paper/viewFile/14972/14051

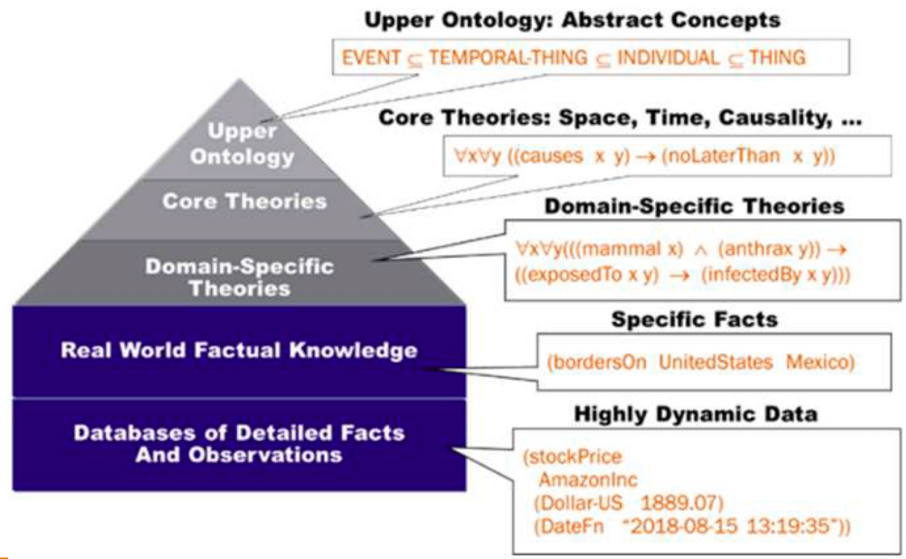
Demonstration - ConceptNet

Examples:

- Concepts:
  - Word: http://conceptnet.io/c/en/word ,
  - duck: http://conceptnet.io/c/en/duck
- Relationships:
  - http://conceptnet.io/s/resource/wordnet/rdf/3.1

Project CYC

- A large ontology to capture the world and human common sense
  - Doug Lenat lead team of computer scientists, computational linguists, philosophers, and logicians
  - Identify and formally axiomatize the tens of millions of rules about world
  - ~40 years effort by Cycorp
- Reasoners on the ontology to make decisions
  - 1000+ specialized reasoners



Details: https://www.cyc.com/

Source: Cyc White Paper

Cyc Details

- Ontology of about 1.5 million general concepts (e.g., taxonomically “placing” terms like eyes, sleep, night, person, unhappiness, hours, posture, being woken up, etc.);
- More than 25 million general rules and assertions involving those concepts
  - *“Most people sleep at night, for several hours at a time, lying down, with their eyes closed, they can be awakened by a loud noise but don’t like that, “*
- Domain-specific extensions to the common sense ontology and knowledge base
  - healthcare, intelligence, defense, energy, transportation and financial services.
- Promoting synergistic use of ontology and learning based approaches (now)
  - Cyc and LLM - https://arxiv.org/ftp/arxiv/papers/2308/2308.04445.pdf.

Source: White Paper – Cyc Technology Overview

Trust Issues With Data and Representation

- Data
  - Respecting data privacy; using open data whenever available
  - Diversity
  - Handling missing/ unknown data; standard representations
- Representation
  - Using appropriate logic
  - Having expressive facts in the knowledge base; reusing standards

Exercise and Code

- FOPL Reasoning
  - https://github.com/biplav-s/course-ai-tai-f23/blob/main/sample-code/Class5-fo-logic/ExploreFOLogic.ipynb

Source: Russell & Norvig, AI: A Modern Approach

Lecture 5: Summary

- We talked about
  - Logic – First Order
  - Inferencing
  - Representation in the Large: ConceptNet, Cyc
  - Trust Issues with Knowledge Representation

Concluding Section

Course Project

Discussion: Projects

- New: two projects
  - Project 1: model assignment
  - Project 2: single problem/ llm based solving / fine-tuning/ presenting result

Project Discussion

1. Go to Google spreadsheet against your name
2. Enter model assignment name and link from (http://modelai.gettysburg.edu/)

1. Create a private Github repository called "CSCE58x-Fall2024-<studentname>-Repo". Share with Instructor (biplav-s) and TA (vishalpallagani)
2. Create Google folder called "CSCE58x-Fall2024-<studentname>-SharedInfo". Share with Instructor (prof.biplav@gmail.com) and TA (vishal.pallagani@gmail.com)
3. Create a Google doc in your Google repo called "Project Plan" and have the following by next class (Sep 5, 2024)

Timeline
1. Title:
2. Key idea: (2-3 lines)
3. Data need:
4. Methods:
5. Evaluation:
6. Milestones
    1. // Create your own
7. Oct 3, 2024

Reference: Project 1 Rubric (30% of Course)

Assume total for Project-1 as 100

- **Project results** – 60%
  - Working system ? – 30%
  - Evaluation with results superior to baseline? – 20%
  - Went through project tasks completely ? – 10%
- **Project effort**s – 40%
  - Project report – 20%
  - Project presentation (updates, final) – 20%
- **Bonus**
  - Challenge level of problem – 10%
  - Instructor discretion – 10%
- **Penalty**
  - Lack of timeliness as per your milestones policy (right) - up to 30%

Milestones and **Penalties**

- Project plan due by Sep 5, 2024 **[-10%]**
- Project deliverables due by Oct 3, 2024 **[-10%]**
- Project presentation on Oct 8, 2024 **[-10%]**

About Next Lecture – Lecture 6

Lecture 6: Searching for Problem Solving

- Search
- Heuristics