

CSCE 580: Introduction to AI

Lecture 4: Formal Representation and Logic

PROF. BIPLAV SRIVASTAVA, AI INSTITUTE

29TH AUG 2024

Carolinian Creed: “I will practice personal and academic integrity.”

Credits: Copyrights of all material reused acknowledged

Organization of Lecture 4

- Introduction Segment
 - Recap of Lecture 3
 - Clarification on Class Logistics
 - Programming exercise: resume
- Main Segment
 - Representing World Knowledge
 - Logic (Propositional)
 - Inferencing (Propositional)
 - Code setup for AIMA examples
- Concluding Segment
 - Course Project Discussion
 - About Next Lecture – Lecture 5
 - Ask me anything

Introduction Section

Recap of Lecture 3

- Data preparation
- Knowledge representation/ graph
- Ontology

Class Logistics

Where to look for information

- a) **instructor personal website**: course description, schedule
- b) **Course Github**: for code; slides is a courtesy
- c) **Blackboard**: for posting grades. Video recorded and shared as a courtesy.

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?

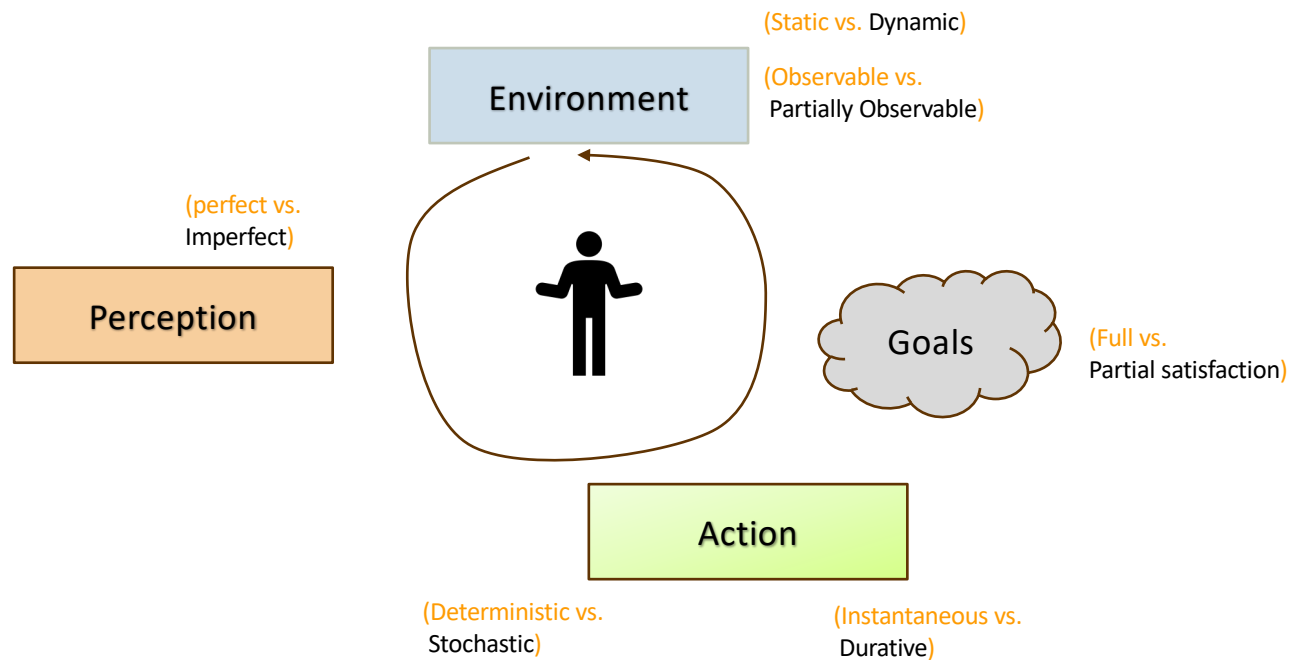
Sample Code for Reference:

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

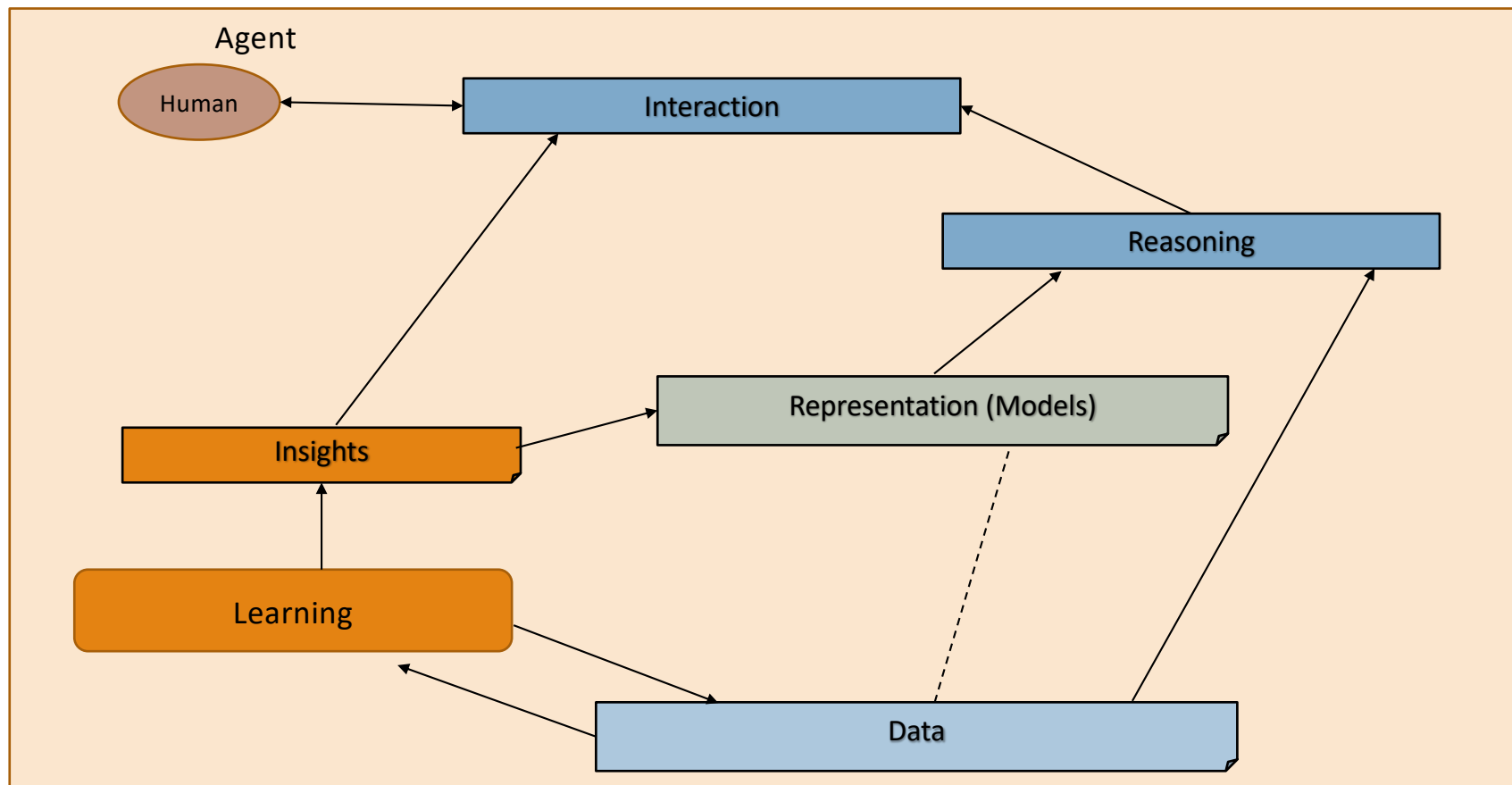
Resume Exercise - Report

- 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

Intelligent Agent Model



Relationship Between Main AI Topics



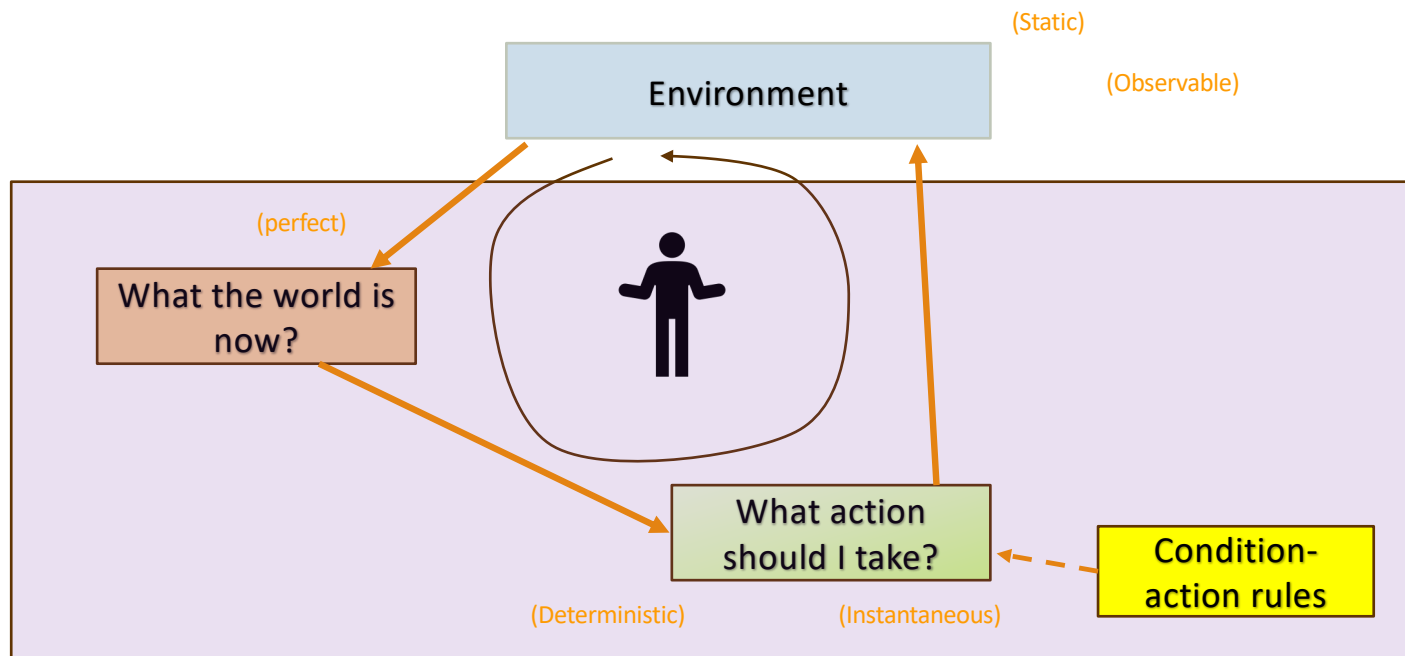
Where We Are in the Course

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 – Simple Knowledge Based



Logic – Basic Idea

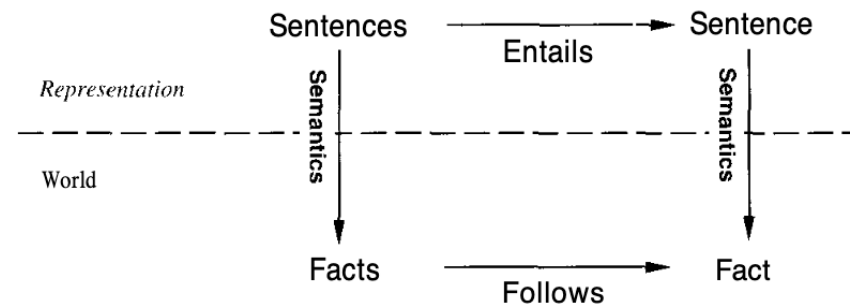
- Starting with **true assumptions**, a knowledge-based system (automaton) to draw **true conclusions**
- Logic consists of three components
 - Syntax -- allowable sentences
 - Semantics -- determining truth of sentence given a model (assignment of values to sentences)
 - Inference Procedure -- rules to draw conclusion given a set of sentences

Formal Logic – 1/3

- An automaton for manipulating symbols and drawing conclusions

- Consists of a knowledge base with:

- a set of true statements (sentences). Sentences have
 - Syntax
 - Semantics – compositional property
- Proof theory: a set of rules for deducing the entailments / interpretations of the sentences



- Properties of sentences

- **Valid:** A sentence is **valid** or necessarily true if and only if it is true under all possible interpretations in all possible worlds. Also called a **tautology**
- **Satisfiable:** A sentence is satisfiable if and only if there is some interpretations in some possible worlds where it is true.

Credits:

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

Propositional Logic

- Sentences: assertions about the world
 - Atomic sentence: propositional symbol
 - ClassToday – whether there is a class today: Yes, No
 - A -- any fact of interest: Yes, No
 - True -- always true
 - False -- always false

Propositional Logic - Syntax

Sentence — *AtomicSentence* | *ComplexSentence*

AtomicSentence — *True* | *False*

| *P* | *Q* | *R* | ...

ComplexSentence — (*Sentence*)

| *Sentence* *Connective* *Sentence*

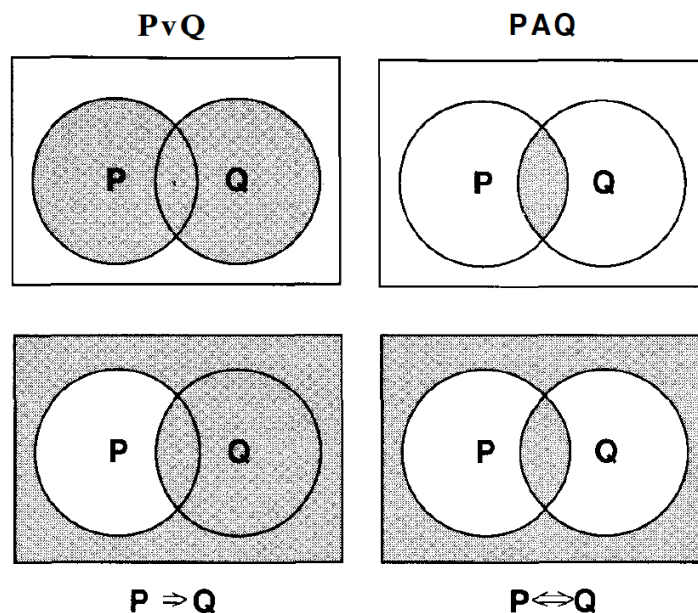
| \neg *Sentence*

Connective — *A* | *V* | \Leftrightarrow | \Rightarrow

BNF (Backus-Naur Form) grammar
of sentences in propositional logic

Source: Russell & Norvig, AI: A Modern Approach

Propositional Logic - Semantics



Model of sentence: Any world in which a sentence is true (under a particular interpretation)

α	β	\neg	$\alpha \vee \beta$	$\neg \beta \vee \neg$	$\alpha \vee \neg$
False	False	False	False	True	False
False	False	True	False	True	True
False	True	False	True	False	False
False	True	True	True	True	True
True	False	False	True	True	True
True	False	True	True	True	True
True	True	False	True	False	True
True	True	True	True	True	True

Truth table to prove soundness of inference

Source: Russell & Norvig, AI: A Modern Approach

Inference Procedure

- Given a knowledge base (KB), generate new sentences α that are entailed by KB
 - $KB \models \alpha$
- Given KB and α , report whether or not α is entailed by KB
 - $KB \models \alpha$
-

Propositional Logic

Inference Procedures

- ◇ **Modus Ponens or Implication-Elimination:** (From an implication and the premise of the implication, you can infer the conclusion.)

$$\frac{a \Rightarrow \beta, \quad a}{\beta}$$

- ◇ **And-Elimination:** (From a conjunction, you can infer any of the conjuncts.)

$$\frac{\alpha_1 \wedge \alpha_2 \wedge \dots \wedge \alpha_n}{\alpha_i}$$

- ◇ **And-Introduction:** (From a list of sentences, you can infer their conjunction.)

$$\frac{\alpha_1, \alpha_2, \dots, \alpha_n}{\alpha_1 \wedge \alpha_2 \wedge \dots \wedge \alpha_n}$$

- ◇ **Or-Introduction:** (From a sentence, you can infer its disjunction with anything else at all.)

$$\frac{\alpha_i}{\alpha_1 \vee \alpha_2 \vee \dots \vee \alpha_n}$$

- ◇ **Double-Negation Elimination:** (From a doubly negated sentence, you can infer a positive sentence.)

$$\frac{\neg\neg\alpha}{\alpha}$$

- ◇ **Unit Resolution:** (From a disjunction, if one of the disjuncts is false, then you can infer the other one is true.)

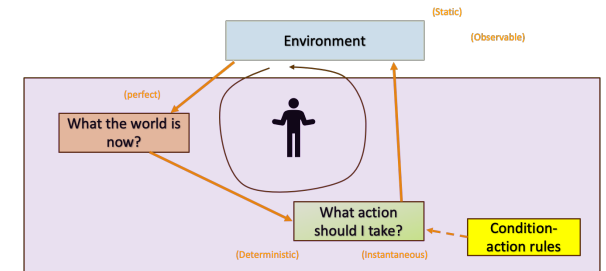
$$\frac{a \vee \beta, \quad \neg\beta}{a}$$

- ◇ **Resolution:** (This is the most difficult. Because 0 cannot be both true and false, one of the other disjuncts must be true in one of the premises. Or equivalently, implication is transitive.)

$$\frac{a \vee \beta, \quad \neg\beta \vee \gamma}{a \vee \gamma} \quad \text{or equivalently} \quad \frac{\neg\alpha \Rightarrow \beta, \quad \beta \Rightarrow \gamma}{\neg\alpha \Rightarrow \gamma}$$

Source: Russell & Norvig, AI: A Modern Approach

KB Agent Procedure



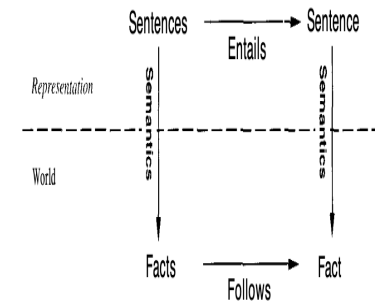
function KB-AGENT(*percept*) **returns** an *action*
static: *KB*, a knowledge base
t, a counter, initially 0, indicating time

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

Source: Russell & Norvig, AI: A Modern Approach

Formal Logic – 2/3

- Levels at which sentences are encoded
 - Epistemic (also called knowledge): what agents knows or believes
 - Logical: how sentences are encoded to allow inferencing. E.g., symbols
 - Executional: how sentences are encoded during execution. E.g., vectors, symbols
- Properties of sentences
 - **Valid**: A sentence is **valid** or necessarily true if and only if it is true under all possible interpretations in all possible worlds. Also called a **tautology**
 - **Satisfiable**: A sentence is satisfiable if and only if there is some interpretations in some possible worlds where it is true.



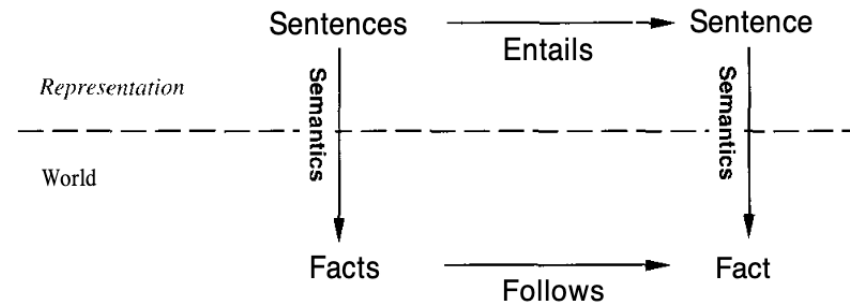
Credits:

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

Formal Logic – 3/3

- 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 - 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

Issue: What about hierarchy among courses?

Representation - 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
- 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.

LowerThan_Course_A1_CourseA2
LowerThan_Course_A2_CourseA3
LowerThan_Course_B1_CourseB2
LowerThan_Course_B2_CourseB3
LowerThan_Course_C1_CourseC2
LowerThan_Course_AC_CourseC3

How to Tackle Course Selection Problem ?

- Represent the world as sentences in KB
 - Update KB based on scenarios
- Solve problems about courses selection scenarios
 - Pose problems as queries to KB
 - Interpret answers // reason with the world

```
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 ← t + 1  
  return action
```

Source: Russell & Norvig, AI: A Modern Approach

Major Types of Reasoning

- Inference: From premises to conclusions
 - Major types
 - **Deduction**: deriving logical conclusions from premises known or assumed to be true
 - **Induction**: deriving from particular premises to a universal conclusion.
 - **Abduction**: from an observation, find the most likely conclusion from the observations
- Usage
 - Deduction is useful to build knowledge bases from parts
 - Induction: to generalize
 - Abduction is a good source for hypothesis / priors in Bayesian learning

Setting Up for AIMA Code

- AI resources
<https://github.com/biplav-s/course-ai-tai-f23/blob/main/sample-code/AI-Resources.md>
- Setting up for Python code from AIMA book
<https://github.com/biplav-s/course-ai-tai-f23/tree/main/sample-code/ai-book-samples>

Exercise and Code

- Logical Reasoning
 - From Book: AI – A Modern Approach, <https://github.com/aimacode/aima-python/blob/master/logic.ipynb>

1,4	2,4	3,4	4,4
1,3 W!	2,3	3,3	4,3
1,2 S OK	2,2 OK	3,2	4,2
1,1 V OK	2,1 B V OK	3,1 P!	4,1

A = Agent
B = Breeze
G = Glitter, Gold
OK = Safe square
P = Pit
S = Stench
V = Visited
W = Wumpus

Source: Russell & Norvig, AI: A Modern Approach

Examples of Agents

Agent Type	Percepts	Actions	Goals	Environment
Medical diagnosis system	Symptoms, findings, patient's answers	Questions, tests, treatments	Healthy patient, minimize costs	Patient, hospital
Satellite image analysis system	Pixels of varying intensity, color	Print a categorization of scene	Correct categorization	Images from orbiting satellite
Part-picking robot	Pixels of varying intensity	Pick up parts and sort into bins	Place parts in correct bins	Conveyor belt with parts
Refinery controller	Temperature, pressure readings	Open, close valves; adjust temperature	Maximize purity, yield, safety	Refinery
Interactive English tutor	Typed words	Print exercises, suggestions, corrections	Maximize student's score on test	Set of students

Source: Russell & Norvig, AI: A Modern Approach

Lecture 4: Summary

- We talked about
 - Knowledge-based agents
 - Logic (Propositional)
 - Inferencing (Propositional)

Concluding Section

Course Project

Discussion: Projects

- New: two projects, or
 - Project 1: model assignment
 - Project 2: single problem/ llm based solving / fine-tuning/ presenting result
- Old
 - Single problem of choice
 - Three sprints; solution to problem available via a chatbot that is built
 - Compare result with ChatGPT

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

About Next Lecture – Lecture 5

Lecture 5: Representing Knowledge

- Logic – First Order
- Inferencing