NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

CZ3005 Artificial Intelligence

Week 8a – Logical Agent

Yu Han

han.yu@ntu.edu.sg

Nanyang Assistant Professor
School of Computer Science and Engineering
Nanyang Technological University

Second Half Lecturer

Nanyang Assistant Professor Yu Han

- Email: han.yu@ntu.edu.sg
- Office: N4-02c-109
- Website: http://hanyu.sg/

Course Overview (2nd Half)

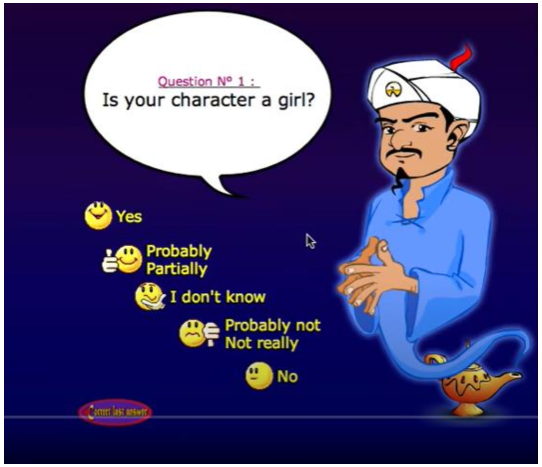
Week	Date	Venue: LT1A	Topic	Tutorial	Lab
8	09/10/2023	1	Logical Agent (a)		
8	10/10/2023	2	Logical Agent (b)		
9	16/10/2023	3	Propositional Logic (a)		
9	17/10/2023	4	Propositional Logic (b)		
10	23/10/2023	5	First-Order Logic (a)	Tutorial 4	One Lab session in Week 10/11 (Lab Attendance is NOT Compulsory)
10	24/10/2023	6	First-Order Logic (b)		
11	30/10/2023	7	Default Logic (a)		
11	31/10/2023	8	Default Logic (b)		
12	06/11/2023	9	Fuzzy Logic (a)	Tutorial 5	
12	07/11/2023	10	Fuzzy Logic (b)		
13	No Class				

Why Logic?

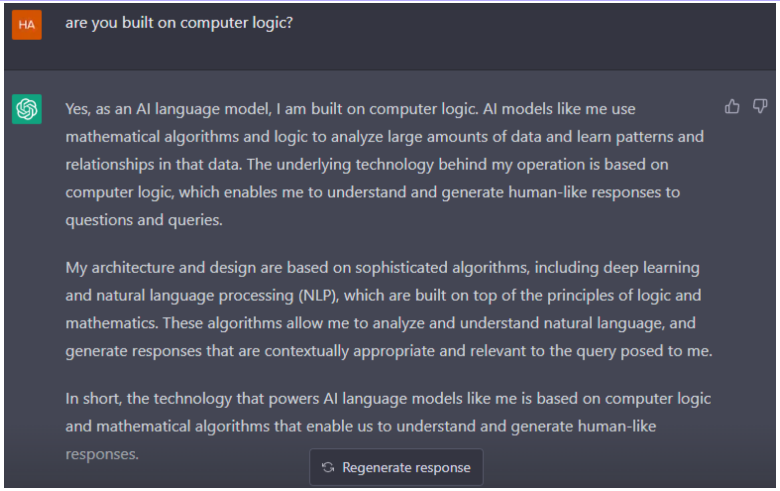
- The earliest way people imagined how to build AI system
- Still a relevant tool to guide today's AI with human knowledge
- Good interpretability (which deep learning-based AI lacks)

An Example Logical Agent

https://www.youtube.com/watch?v=MxI_tzOs6ME

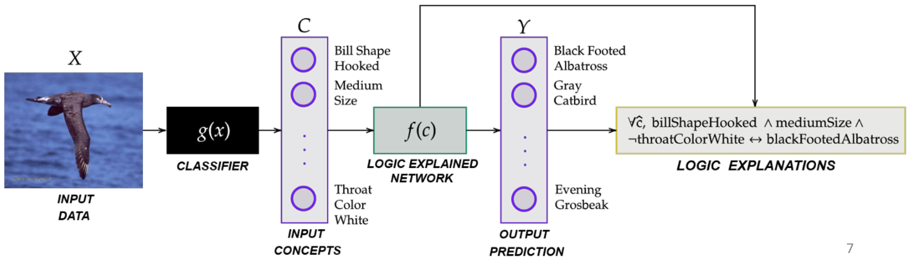


Why Logic?



Why Logic?

- New life for an old technology:
  - Deep Learning and Logic Programming are starting to merge today for building Explainable AI (XAI) systems.
  - Logic Explained Networks:

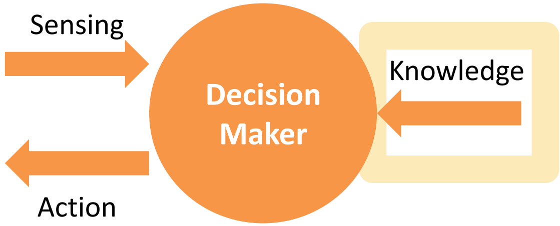


From Search to Logical Reasoning

- Representation of Knowledge and the Reasoning Processes are central to the entire field of AI

- A knowledge-based system (KBS) is a computer program that reasons and uses a knowledge base to solve complex problems.

Knowledge and Intelligence



The Knowledge-Based Approach

- Agents that ***know***
  - Achieve competence by being told new knowledge or by learning
  - Achieve adaptability by updating their knowledge
  - > *Knowledge representation*
    - State of the world, properties and evolution of the world; goals of the agent, actions and their effect
- Agents that ***reason***
  - Use knowledge to deduce course of actions
  - > *Knowledge inference*

Logic

Knowledge-Based Agents

- **Knowledge base (KB)**
  - Set of sentences i.e., representations of facts (DB)
  - Knowledge representation language
- Adding and querying knowledge
  - **Tell**: add a sentence to the KB
  - **Ask**: retrieve knowledge from the KB
    - Answers must *follow* from what has been **Tell**'ed (told)
- Inference mechanism
  - Determine what follows from the KB

Problem Formulation of KBS

- Knowledge Based System
  - States: Instances of the KB (sets of sentences)

    –> Use **Tell** to build the KB

    e.g. Tell(KB, "Smoke ⇒ Fire")
    Tell(KB, "Fire ⇒ Call_999")
    …
    Tell(KB, "Smoke") → there's smoke
  - Operators: Add / Infer a new sentence
  - Goal: Answer a query

    –> Use **Ask** to query the KB

    e.g. Ask(KB, "**?** Call_999")

A Generic Knowledge-Based Agent

```
function KB-Agent (percept) returns action
    static  KB,          // a knowledge base
            t            // a time counter, initially 0

    Tell (KB, Make-Percept-Sentence (percept, t))
    action ← Ask (KB, Make-Action-Query (percept, t))
    Tell (KB, Make-Action-Sentence (action, t))
    t ← t + 1
    return action
```

– > 3 steps: interpretation, inference, execution

– > KB: background knowledge (observed)
+ acquired information (deduced)

Levels of Knowledge

Human context is obvious to human.

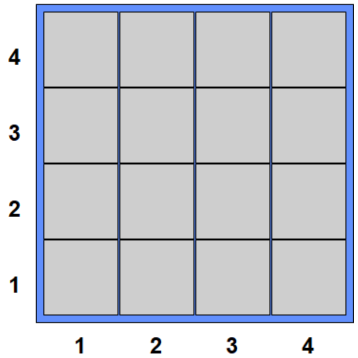
Logical encoding of knowledge

Tell

Ask

- **Epistemological level**
  - Declarative description of knowledge
    - e.g. facts: "there is smoke in the kitchen", "it is not warm enough"
      rules: "if there is smoke then there must be a fire"
- **Logical level**
  - Logical encoding of knowledge (into sentences)
    - e.g. facts: Smoke; rules: Implies(Smoke, Fire)
- **Implementation level**
  - Physical representation of knowledge (sentences)
    - e.g. - the string "Implies(Smoke, Fire)", or
      - a "1" entry in a 2-dimensional array: Implies[X,Y]

Example: the Wumpus World

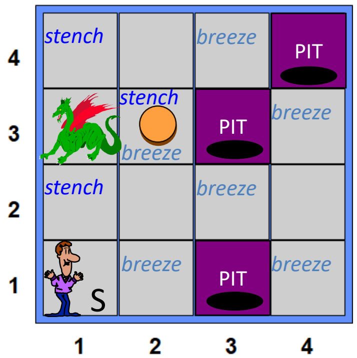


RL: don't know rules & explore

Rules are obvious, logic is used to solve this problem.

Example: the Wumpus World

- Problem description (PAGE)
  - Environment
    - Grid of squares, walls;
    - Agent, gold, pits, wumpus.
  - Goal
    - Find the gold, return to S at [1,1].
  - Percepts
    - A list of 5 symbols, e.g. [Stench, Breeze, Glitter, Bump, Scream];
    - Agent's location *not* perceived.
  - Actions
    - Go-Forward, Turn-Left, Turn-Right, Grab, Shoot (1 arrow only), Climb.

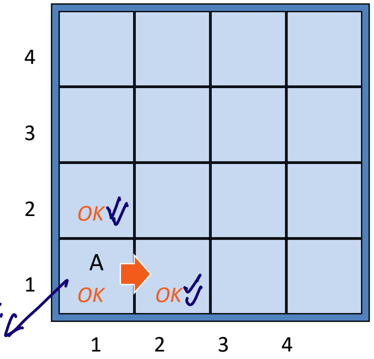


The Wumpus World

- Problem description (cont'd)
  - Initial state
    - Agent at [1,1]; gold, pits and wumpus in random squares.
  - Path-cost
    - Climbing out with the gold: +1000 (without: 0) • Each action: –1
    - Getting killed (pit or wumpus): –10000
  - Knowledge
    - "In all squares adjacent to the one where the wumpus is, the agent will perceive a stench."
    - "In all squares adjacent to a pit, the agent will perceive a breeze."
    - In the square where the gold is, the agent will perceive a glitter."
    - When walking into a wall, the agent will perceive a bump."
    - When the wumpus is killed, the agent will perceive a scream."

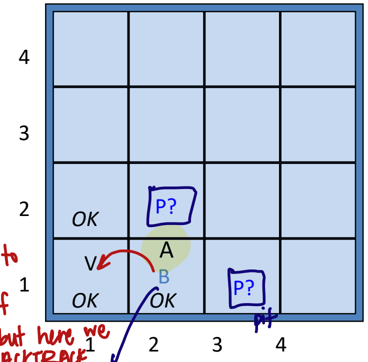
Acting and Reasoning in the Wumpus World

(0) Initial state
[nil, nil, nil, nil, nil]



Initial state: nothing sensed.

(1) after {F}
[nil, Breeze, nil, nil, nil]



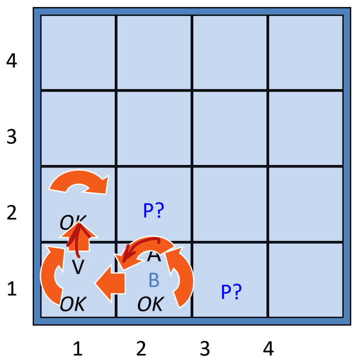
RL will tell us to take leap of faith but here we BACKTRACK

feel a Breeze

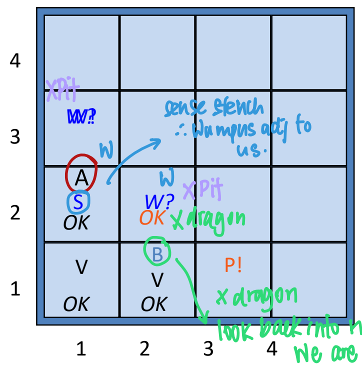
A = Agent G = Glitter, Gold
B = Breeze OK = Safe square
P = Pit V = Visited
S = Stench W = Wumpus

Acting and Reasoning in the Wumpus World

(1) after {F}
[nil, Breeze, nil, nil, nil]



(6) after {F, L, L, F, R, F}
[Stench , nil, nil, nil, nil]

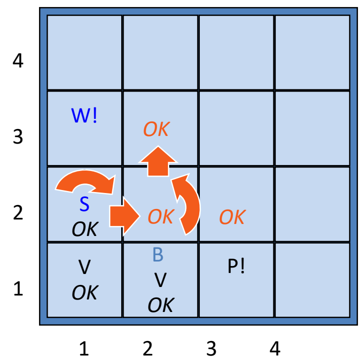


look back into history. We are here only felt breeze no stench.

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

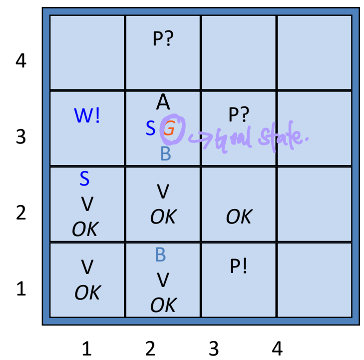
Acting and Reasoning in the Wumpus World

(6) after {F, L, L, F, R, F}
[Stench , nil, nil, nil, nil]



speed up

(10) after {F, L, L, F, R, F ,R, F, L, F}
[Stench, Breeze, Glitter, nil, nil]

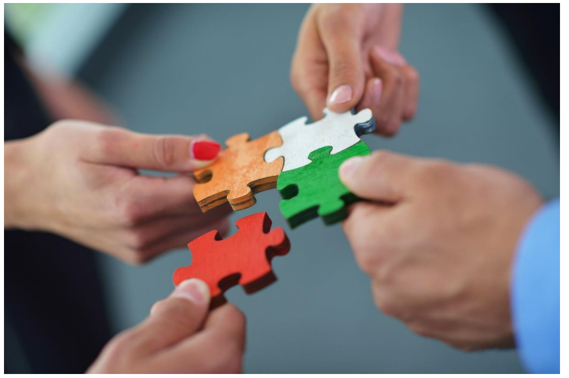


A = Agent G = Glitter, Gold
B = Breeze OK = Safe square
P = Pit V = Visited
S = Stench W = Wumpus

Summary

- **Intelligent agents need …**
  - Knowledge about the world to make good decisions.
- **Knowledge can be …**
  - Defined using a knowledge representation language.
  - Stored in a knowledge base in the form of sentences.
  - Inferred, using an inference mechanism and rules.

Thank you!



NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

CZ3005 Artificial Intelligence

Week 8b – Logical Agent

Yu Han

han.yu@ntu.edu.sg

Nanyang Assistant Professor
School of Computer Science and Engineering
Nanyang Technological University

Recap

- **Intelligent agents need …**
  - Knowledge about the world to make good decisions.
- **Knowledge can be …**
  - Defined using a knowledge representation language.
  - Stored in a knowledge base in the form of sentences.
  - Inferred, using an inference mechanism and rules.

Recap

```
function KB-Agent (percept) returns action
        static  KB,             // a knowledge base
                t               // a time counter, initially 0

        Tell (KB, Make-Percept-Sentence (percept, t))
        action ← Ask (KB, Make-Action-Query (percept, t))
        Tell (KB, Make-Action-Sentence (action, t))
        t ← t + 1
        return action
```

– > 3 steps: interpretation, inference, execution

– > KB: background knowledge (observed)
+ acquired information (deduced)

Knowledge Representations

- **Knowledge representation (KR)**
  - KB: set of sentences –> need to
  - Express knowledge in a (computer-) tractable form
- **Knowledge representation language**
  - Syntax – implementation level
    - Possible configurations that constitute sentences
  - Semantics – knowledge level
    - Facts of the world the sentences refer to
    - e.g. language of arithmetics: x, y numbers
      sentence: “$x \geq y$”, semantics: “greater or equal”

Logic

Reasoning and Logic

- **Logic**
  - Representation + Inference = Logic
    - Where representation = syntax + semantics
- **Reasoning**
  - Construction of new sentences from existing ones
- **Entailment** as logical inference
  - the relationship between sentences whereby one sentence will be true if all the others are also true

Inference

Sentence —*entails*→ Sentence

Representation

World

Semantics *Semantics*

Fact —*follows*→ Fact

try not to use because it generalizes.

Deduction and Induction

- **Mechanical reasoning**
  - Example
    - If a chord sequence is tonal, then it can be generated by a context-sensitive grammar.
    - The twelve-bar blues has a chord sequence that is tonal. |-
    - The twelve-bar blues has a chord sequence that can be generated by a context-sensitive grammar.
- **Deductive inference**
  - KB: Monday $\Rightarrow$ Work, Monday |- Work *sound*

  Rule SAME

  today is monday infer we have to work
- **Inductive inference**
  - KB: Monday $\Rightarrow$ Work, Work |- Monday *unsound!*
  - Generalization e.g., *"all swans are white …"*

today we are working infer that today is monday } not a strong logic

Deduction and Induction

Entailment and Inference

- **Entailment**
  - Generate sentences that are necessarily true, given that the existing sentences are true
  - Notation: KB $|= \alpha$
    - e.g. Wumpus world:

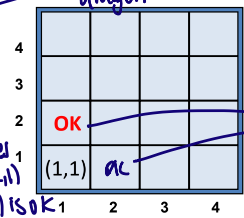
      No smell in (1,1) No Breeze (1,1)

      { "$\neg$S(1,1)", "$\neg$B(1,1)" } $|=$ "OK(2,1)"

      Arithmetics:

      { "$x \geq y$", "$y \geq z$" } $|=$ "$x \geq z$"

no pit & no dragon



concludes that (2,1) & (1,2) is OK

Added to KB as new facts.

- **Inference**
  - The act or process of deriving logical conclusions from premises known or assumed to be true.
  - **Tell**, given KB: (KB $|= \alpha$) !
  - **Ask**, given KB and α: (KB $|= \alpha$) ?

Knowledge Representation Languages

- **Formal (programming) languages**
  - Good at describing algorithms and data structures
    - e.g. the Wumpus world as a 4x4 array, World[2,2] ← Pit
  - Poor at representing incomplete / uncertain information
    - e.g. "there is a pit in [2,2] or [3,1]", or "…a wumpus *somewhere*"
  - > *not <u>expressive</u> enough*
- **Natural languages**
  - Very expressive (too much, thus very complex)
  - More appropriate for communication than representation
  - Suffer from ambiguity
    - e.g. "It's hot!"
    - e.g. "small cats and dogs" compared to "– x + y".

Properties of Representations

- *KR languages should combine the advantages of both programming and natural languages.*
- **Desired properties**
  - Expressive
    - Can represent everything we need to.
  - Concise
  - Unambiguous
    - Sentences have a unique interpretation.
  - Context independent
    - Interpretation of sentences depends on semantics only.
  - Effective
    - An inference procedure allows to create new sentences.

Properties of Semantics

- **Interpretation (meaning)**
  - *Correspondence between sentences and facts*
  - Arbitrary meaning, fixed by the writer of the sentence
    - e.g. Natural languages: meaning fixed by usage (cf. dictionary)
      exceptions: encrypted messages, codes (e.g. Morse)
      chat
  - Systematic relationship: compositional languages
    - *The meaning of a sentence is a function of the meaning of its parts.*
  - <u>Truth value</u>
    - A sentence make a claim about the world –> TRUE or FALSE
    - *Depends on the interpretation and the state of the world*
      - e.g. Wumpus world: S(1,2) true if means "Stench at [1,2]" and the world has a wumpus at either [1,3] or [2,2].

one
un
uno
yi
ichi
…
I
1
—
…

Properties of Inference

- **Definition**
  - *Inference (reasoning) is the process by which conclusions are reached*
  - Logical inference (deduction) is the process that implements entailment between sentences
- **Useful properties**
  - <u>Valid</u> sentence (tautology) → No matter S or ¬S the whole sentence is true
    - iff TRUE under all possible interpretations in all possible worlds.
      - e.g. "S or ¬ S" is valid, "S(2,1) or ¬ S(2,1)", etc. or today is either going to rain or not rain
  - <u>Satisfiable</u> sentence
    - iff there is some interpretation in some world for which it is TRUE
      - e.g. "S and ¬ S" is unsatisfiable

e.g today is going to rain

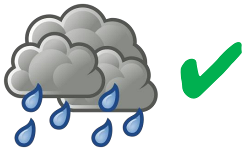
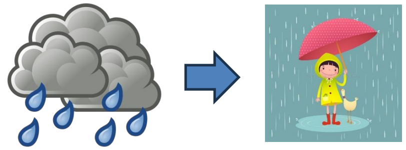
e.g today is going to rain & not rain

either rains or not rain sentence will not be right (unsatisfiable sentence)

Properties of Inference

- **Soundness**
  - A deductive system is considered sound if it never proves a false statement; it only derives conclusions that are true whenever the premises are true.
  - Proof: sequence of operations of a sound inference
    - Record of operations that generate a specific entailed sentence e.g. "Smoke ⇒ Fire" and "Smoke" |= "Fire"
      "Fire ⇒ Call_911" and "Fire" |= "Call_911"
- **Completeness** Satisfiable rules but we don't know conditions are fullfilled and ⇒ Require some facts for conc
  - A deductive system is complete if it can prove all true statements.
  - Any statement that is true within the system can be formally derived or proven using the rules of the system.

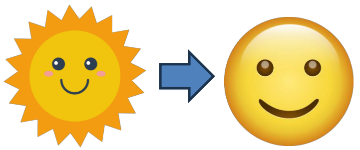
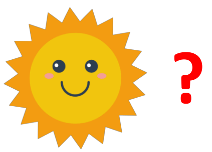
Example: Sound & Complete

Premise 1: It is raining outside. (A)
Premise 2: If it is raining, then the ground is wet. (A implies B)
Using Modus Ponens, we can infer that the ground is wet. (B)

In this case, the argument is both sound and complete.
- It is sound as the conclusion (the ground is wet) logically follows from the premises.
- It is complete as all statements derived from these premises can be proven using the given rule (Modus Ponens).

Example: Sound, but NOT Complete

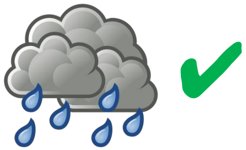



Premise 1: If it is sunny, then people are happy. (A implies B)

Not complete: we don't know right now is it sunny or not. Doesn't tell us if part satisfied or not

- In this case, the argument is sound because the conclusion follows logically from the premise.
- However, this deductive system is not complete because it cannot prove statements about the weather condition taking place.

Example: Complete, but NOT Sound




Premise 1: It is raining outside. (A)
Conclusion: The moon is made of cheese.

↳ dk where this conc came from. doesn't follow any deduction.

- The argument is complete because it tries to make a statement about the moon, which is a valid topic within the system.
- However, it is not sound because the conclusion does not logically follow from the premise;
  - raining outside does not imply anything about the composition of the moon.

–Soundless & Completeness: Refers to properties of entire KB

-Validity & satisfiability : Refers to properties of indv sentences in KB.

Inference and Agent Programs

- **Inference in computers**
  - Does not know the interpretation the agent is using for the sentences in the KB
  - Does not know about the world (actual facts)
  - Knows only what appears in the KB (sentences)
    - e.g. Wumpus world: doesn't know the meaning of "OK", what a
      wumpus or a pit is, etc. – can only see: KB |= "[2,2] is OK"
  - > *Cannot reason informally*
    - does not matter, however, if KB |= "[2,2] is OK" is a valid sentence
- **Formal inference**
  - Can handle arbitrarily complex sentences, KB |= P

Thank you!