COMP 4431: Artificial Intelligence

Zhiyuan Wen (Research Assistant Professor)

zyuanwen@polyu.edu.hk

Department of Computing

The Hong Kong Polytechnic University

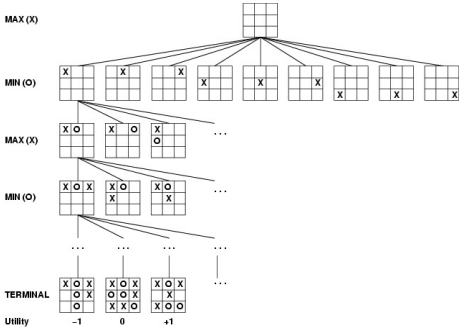
Learning Outcomes of Lecture 04

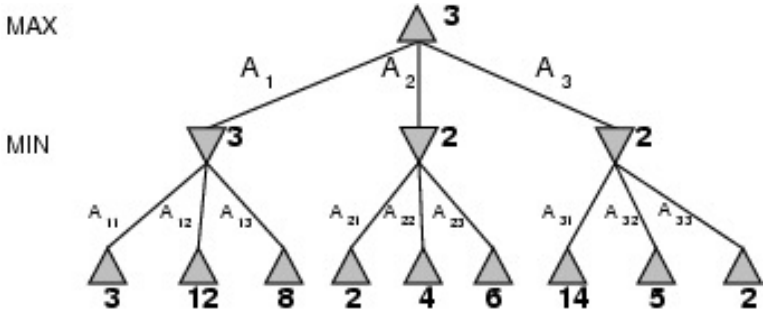


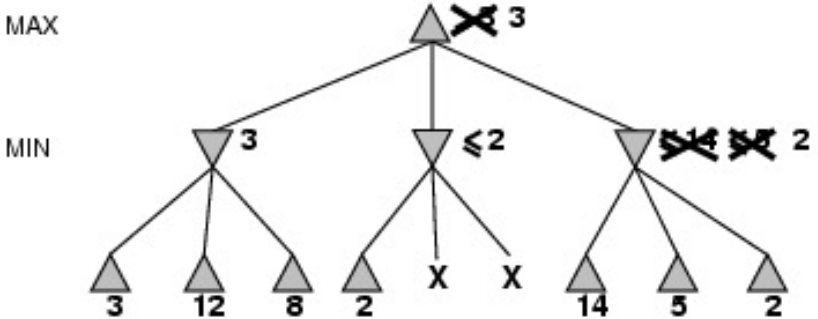
- Understand the Concept of Constraint Satisfaction Problem (CSP)
- Know How to Use Heuristic Rules to Solve CSP Step-by-Step
- Know the Definition Games in the Context of AI
- Have a Basic Understanding of LLM-based Multi-agent System

Recap: α-β Pruning

- Gaming with opponents (rational agents)
- One always want to maximize the utility, the other want to minimize
- **Why Pruning? Reduce the size of search space**
- α：Best already explored option along path to the root for maximizer
- β：Best already explored option along path to the root for minimizer







Agenda

- **Decision Tree**
  - **Formulation**
  - **ID3**
  - **Pros & Cons**
  - **Development**

Template of the Group Project proposal will be released on the Blackboard this week (Notice the announcement)

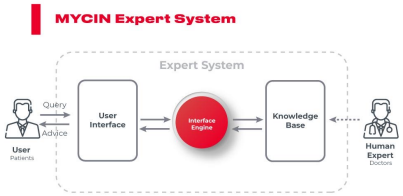
DDL: 10.17 23:59

Decision Tree

- **The second rise of AI (1980-1987)**
  - **The knowledge revolution**
  - **Expert system has been widely used till now**
    - Answers questions or solves problems about a specific domain of knowledge
    - Reasoning by logical rules that are derived from the knowledge of experts

Pre-defined rules cannot cover all situations in real scenarios

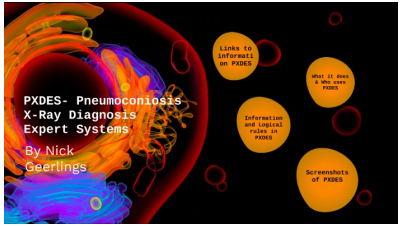
The Second Winter



MYCIN: Recognizing Bacteria



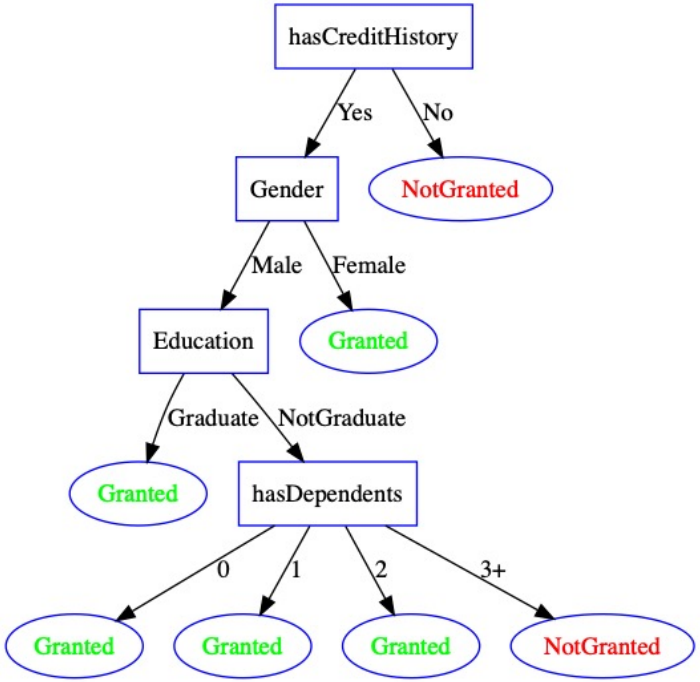
DENDRAL: Molecular structure prediction



PXDES: Lung cancer prediction

Decision Tree

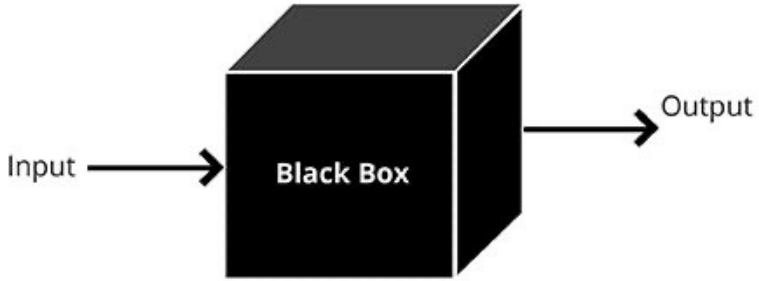
- **Interpretable Model: (Why we still are using them)**
  - **Operation research, business intelligence**



The Third Rise

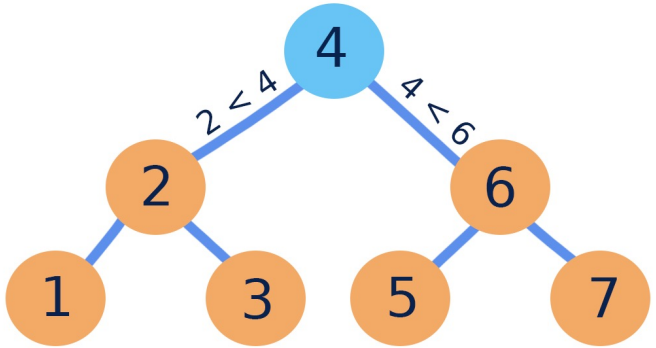
Machine Learning Model that simulates Expert Systems

Generative Models:
NN, LLM, Diffusion model…

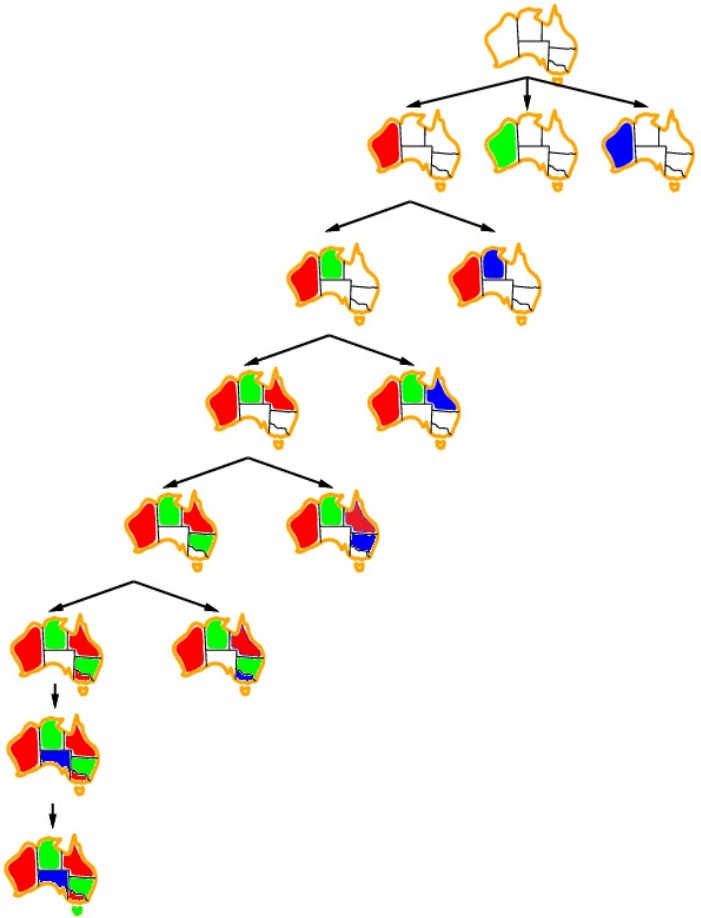


Decision Tree

- **Trees are widely used in Computer Science**
  - **Binary Search/Sorting**
  - **Search Tree**
  - **…**



In Order Traversal: 1 2 3 4 5 6 7

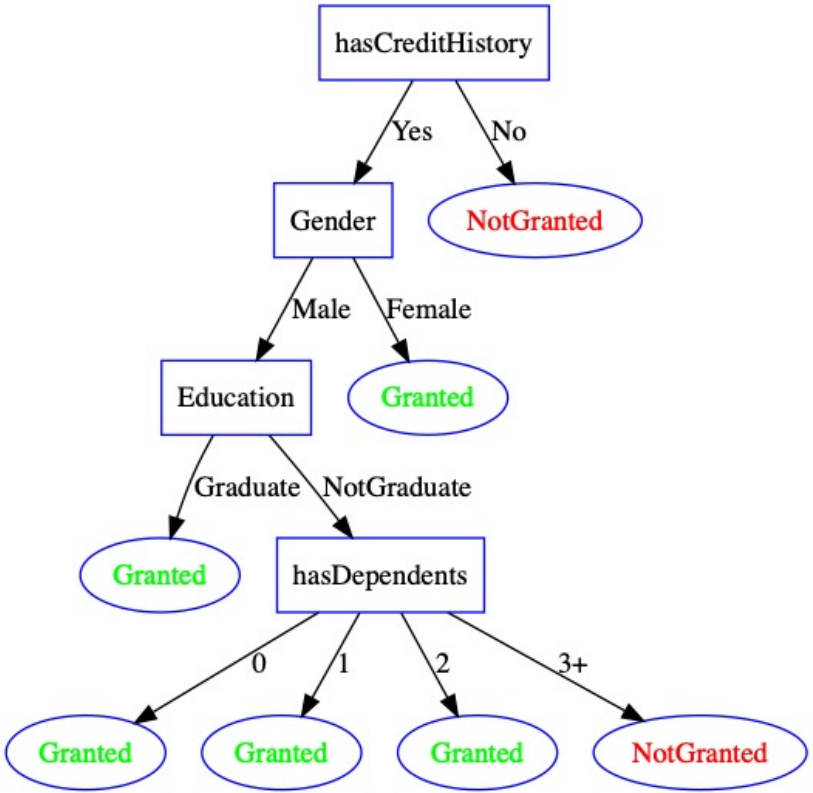


Formulation

- Base Hypothesis: $g_t(x)$: leaf at path t
- Condition $q_t(x)$: [is **x** on the path t]

$$G(x) = \sum_{t=1}^{T} q_t(x) * g_t(x)$$

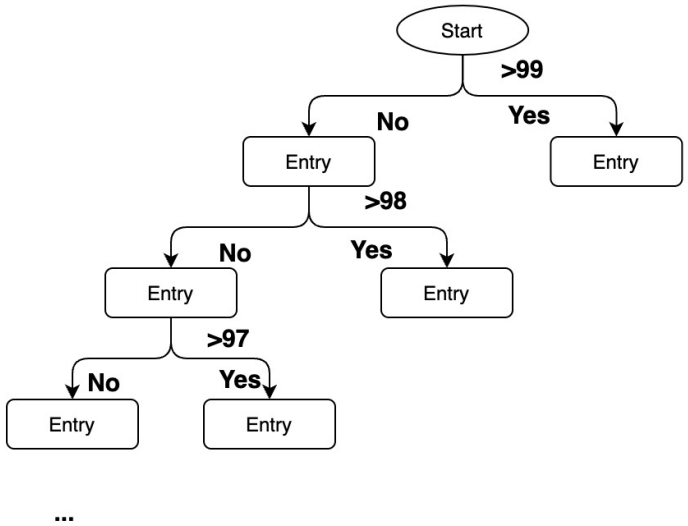
Mimicking human reasoning process

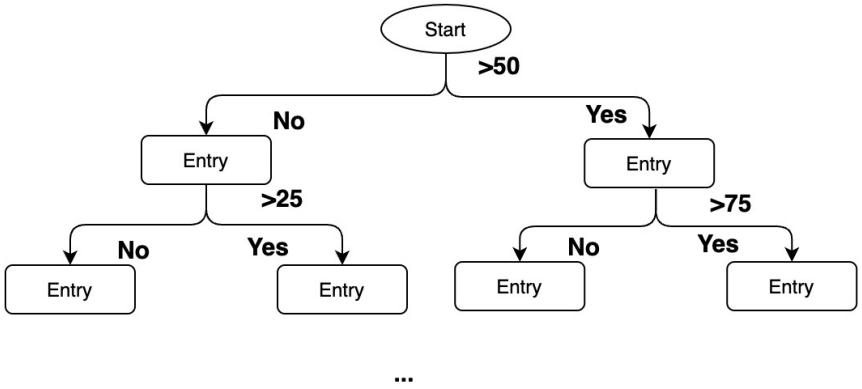


What is a good decision tree?

- **Depth?**
- **Number of Branches/decisions?**
- **Balance?**

$$G(x) = \sum_{t=1}^{T} q_t(x) * g_t(x)$$

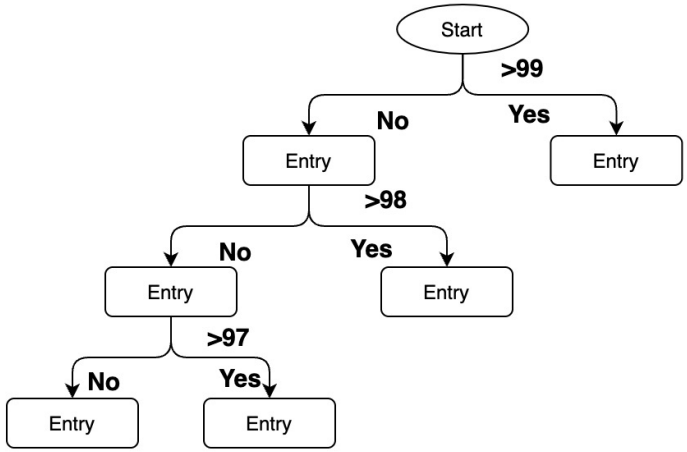




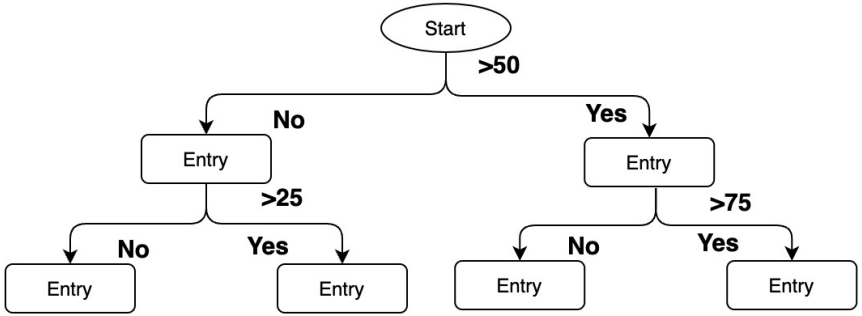
What is a good decision tree?

- **Depth?**
- **Number of Branches/decisions?**
- **Balance?**

$$G(x) = \sum_{t=1}^{T} q_t(x) * g_t(x)$$



...



...

Make less decisions, make good decisions

How to get a good decision tree?

- **Iterative Dichotomiser 3 (ID3)**
  - dichotomisation means dividing into two completely opposite things.
  - iteratively divides attributes into groups which are the most dominant attribute and others to construct a tree.
  - the most dominant attribute -> decisions

How do we know the most dominant attribute?

Day	Outlook	Temp.	Humidity	Wind	Decision
1	Sunny	Hot	High	Weak	No
2	Sunny	Hot	High	Strong	No
3	Overcast	Hot	High	Weak	Yes
4	Rain	Mild	High	Weak	Yes
5	Rain	Cool	Normal	Weak	Yes
6	Rain	Cool	Normal	Strong	No
7	Overcast	Cool	Normal	Strong	Yes
8	Sunny	Mild	High	Weak	No
9	Sunny	Cool	Normal	Weak	Yes
10	Rain	Mild	Normal	Weak	Yes
11	Sunny	Mild	Normal	Strong	Yes
12	Overcast	Mild	High	Strong	Yes
13	Overcast	Hot	Normal	Weak	Yes
14	Rain	Mild	High	Strong	No

https://sefiks.com/2017/11/20/a-step-by-step-id3-decision-tree-example/

Iterative Dichotomiser 3 (ID3)

- **Entropy**
  - the amount of randomness that exists in a particular set of data.

$$Entropy(S) = \sum -p_i * log_2(p_i)$$

- **Information Gain**
  - Information gain measures the extent to which the uncertainty is reduced due to the introduction of features/conditions.

$$IG(S,A) = Entropy(S) - \sum(\frac{|S_v|}{|S|} * Entropy(S_v))$$

Day	Outlook	Temp.	Humidity	Wind	Decision
1	Sunny	Hot	High	Weak	No
2	Sunny	Hot	High	Strong	No
3	Overcast	Hot	High	Weak	Yes
4	Rain	Mild	High	Weak	Yes
5	Rain	Cool	Normal	Weak	Yes
6	Rain	Cool	Normal	Strong	No
7	Overcast	Cool	Normal	Strong	Yes
8	Sunny	Mild	High	Weak	No
9	Sunny	Cool	Normal	Weak	Yes
10	Rain	Mild	Normal	Weak	Yes
11	Sunny	Mild	Normal	Strong	Yes
12	Overcast	Mild	High	Strong	Yes
13	Overcast	Hot	Normal	Weak	Yes
14	Rain	Mild	High	Strong	No

Iterative Dichotomiser 3 (ID3)

The overall data Entropy:

Entropy(Decision)

$= - p(Yes) * \log_2 p(Yes) - p(No) * \log_2 p(No)$

$= - (9/14) * \log_2(9/14) - (5/14) * \log_2(5/14) =$ **0.940**

Is wind the dominant attribute?

Gain(Decision, Wind)

$= Entropy(Decision) - \sum [p(Decision|Wind) * Entropy(Decision|Wind)]$

$= Entropy(Decision) - [p(Decision|Wind=Weak) * Entropy(Decision|Wind=Weak)] - [p(Decision|Wind=Strong) * Entropy(Decision|Wind=Strong)]$

Day	Outlook	Temp.	Humidity	Wind	Decision
1	Sunny	Hot	High	Weak	No
2	Sunny	Hot	High	Strong	No
3	Overcast	Hot	High	Weak	Yes
4	Rain	Mild	High	Weak	Yes
5	Rain	Cool	Normal	Weak	Yes
6	Rain	Cool	Normal	Strong	No
7	Overcast	Cool	Normal	Strong	Yes
8	Sunny	Mild	High	Weak	No
9	Sunny	Cool	Normal	Weak	Yes
10	Rain	Mild	Normal	Weak	Yes
11	Sunny	Mild	Normal	Strong	Yes
12	Overcast	Mild	High	Strong	Yes
13	Overcast	Hot	Normal	Weak	Yes
14	Rain	Mild	High	Strong	No

Iterative Dichotomiser 3 (ID3)



Is wind the dominant attribute?

Gain(Decision, Wind)

= Entropy(Decision) – ∑ [p(DecisionlWind) *Entropy(DecisionlWind)]

= Entropy(Decision) – [p(DecisionlWind=Weak) * Entropy(DecisionlWind=Weak)] – [p(DecisionlWind=Strong) * Entropy(DecisionlWind=Strong)]

Entropy(DecisionlWind=Weak) = – p(No) * log2p(No) – p(Yes) * log2p(Yes) = – (2/8) * log2(2/8) – (6/8) * log2(6/8) = 0.811

Entropy(DecisionlWind=Strong) = – p(No) * log2p(No) – p(Yes) * log2p(Yes) = – (3/6) * log2(3/6) – (3/6) * log2(3/6) = 1

Gain(Decision, Wind) = 0.940 – [(8/14) * 0.811] – [(6/14)* 1] = **0.048**

Day	Outlook	Temp.	Humidity	Wind	Decision
1	Sunny	Hot	High	Weak	No
3	Overcast	Hot	High	Weak	Yes
4	Rain	Mild	High	Weak	Yes
5	Rain	Cool	Normal	Weak	Yes
8	Sunny	Mild	High	Weak	No
9	Sunny	Cool	Normal	Weak	Yes
10	Rain	Mild	Normal	Weak	Yes
13	Overcast	Hot	Normal	Weak	Yes

Day	Outlook	Temp.	Humidity	Wind	Decision
2	Sunny	Hot	High	Strong	No
6	Rain	Cool	Normal	Strong	No
7	Overcast	Cool	Normal	Strong	Yes
11	Sunny	Mild	Normal	Strong	Yes
12	Overcast	Mild	High	Strong	Yes
14	Rain	Mild	High	Strong	No

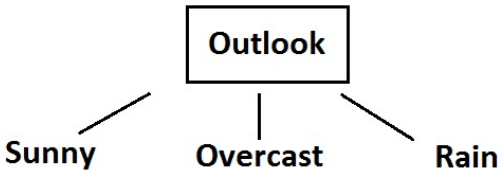
Iterative Dichotomiser 3 (ID3)

Gain(Decision, Wind) = 0.940 – [(8/14) * 0.811] – [(6/14)* 1] = **0.048**

Gain(Decision, Outlook) = **0.246**

Gain(Decision, Temperature) = **0.029**

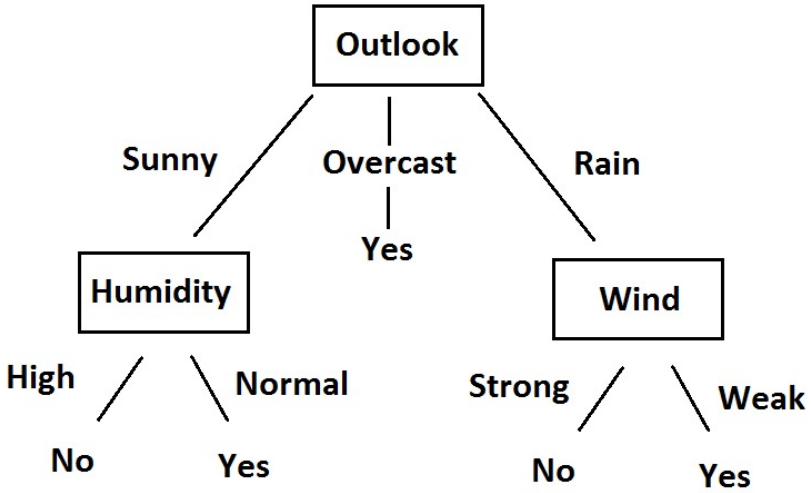
Gain(Decision, Humidity) = **0.151**



Day	Outlook	Temp.	Humidity	Wind	Decision
1	Sunny	Hot	High	Weak	No
2	Sunny	Hot	High	Strong	No
3	Overcast	Hot	High	Weak	Yes
4	Rain	Mild	High	Weak	Yes
5	Rain	Cool	Normal	Weak	Yes
6	Rain	Cool	Normal	Strong	No
7	Overcast	Cool	Normal	Strong	Yes
8	Sunny	Mild	High	Weak	No
9	Sunny	Cool	Normal	Weak	Yes
10	Rain	Mild	Normal	Weak	Yes
11	Sunny	Mild	Normal	Strong	Yes
12	Overcast	Mild	High	Strong	Yes
13	Overcast	Hot	Normal	Weak	Yes
14	Rain	Mild	High	Strong	No

Iterative Dichotomiser 3 (ID3)

Iteratively construct the nodes in the decision tree:

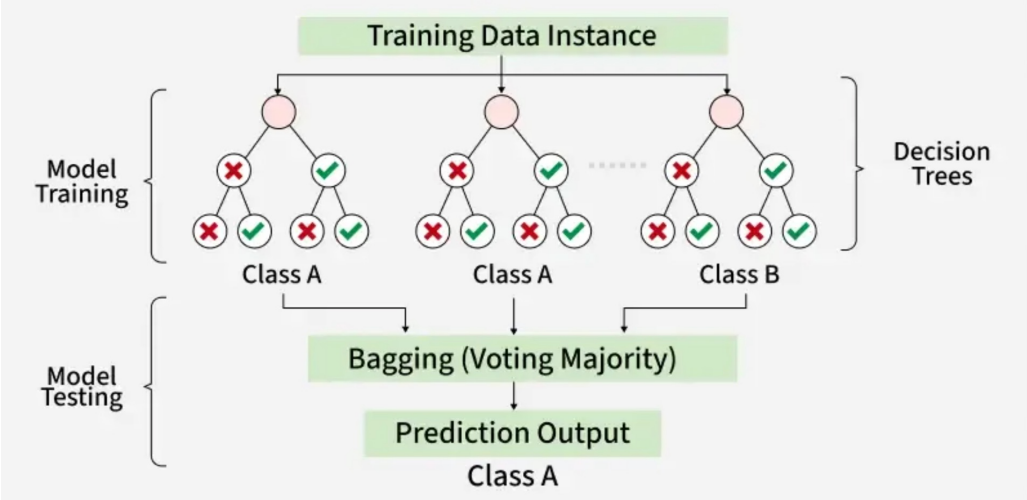


Divide and Conquer

Day	Outlook	Temp.	Humidity	Wind	Decision
1	Sunny	Hot	High	Weak	No
2	Sunny	Hot	High	Strong	No
3	Overcast	Hot	High	Weak	Yes
4	Rain	Mild	High	Weak	Yes
5	Rain	Cool	Normal	Weak	Yes
6	Rain	Cool	Normal	Strong	No
7	Overcast	Cool	Normal	Strong	Yes
8	Sunny	Mild	High	Weak	No
9	Sunny	Cool	Normal	Weak	Yes
10	Rain	Mild	Normal	Weak	Yes
11	Sunny	Mild	Normal	Strong	Yes
12	Overcast	Mild	High	Strong	Yes
13	Overcast	Hot	Normal	Weak	Yes
14	Rain	Mild	High	Strong	No

Iterative Dichotomiser 3 (ID3)

- **Pros:**
  - Intuitive, easy to implement
- **Cons:**
  - Continuous Features?
  - Calculation of Information Gain?
  - Overfitting? Unseen features?
- **Improvements:**
  - C4.5 discretizes continuous features, adds regulazation…
  - CART supports both classification and regression…
  - Forest, Bagging, Boosting…



Random Forest

https://www.geeksforgeeks.org/machine-learning/random-forest-algorithm-in-machine-learning/

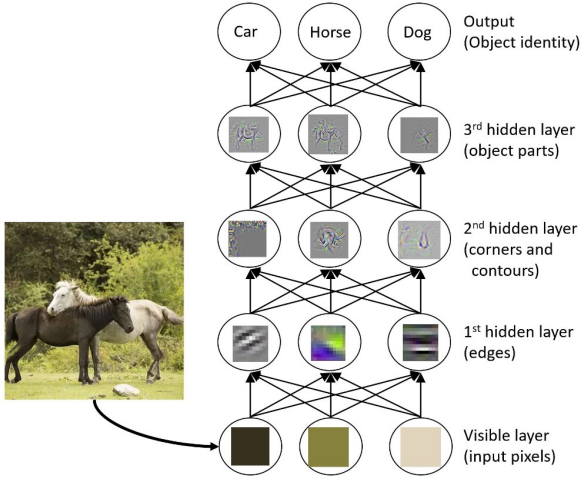
Decision Tree

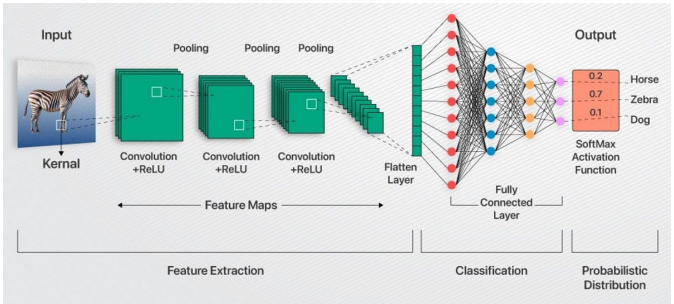
- **Practice:**
  - **Kaggle Competitions (https://www.kaggle.com/competitions/titanic/overview)**
  - **Python sklearn (https://scikit-learn.org/stable/modules/tree.html)**

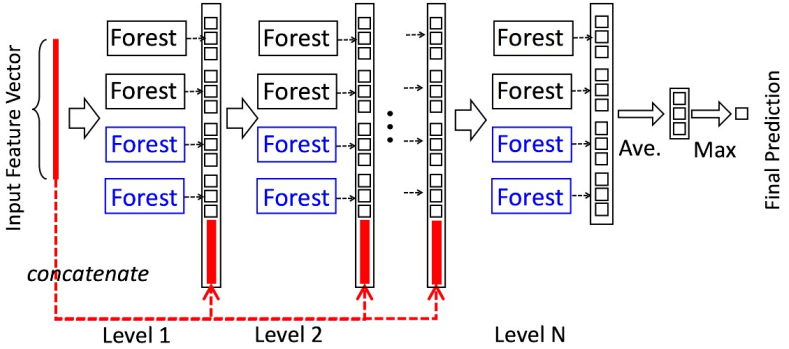
Quiz Time!

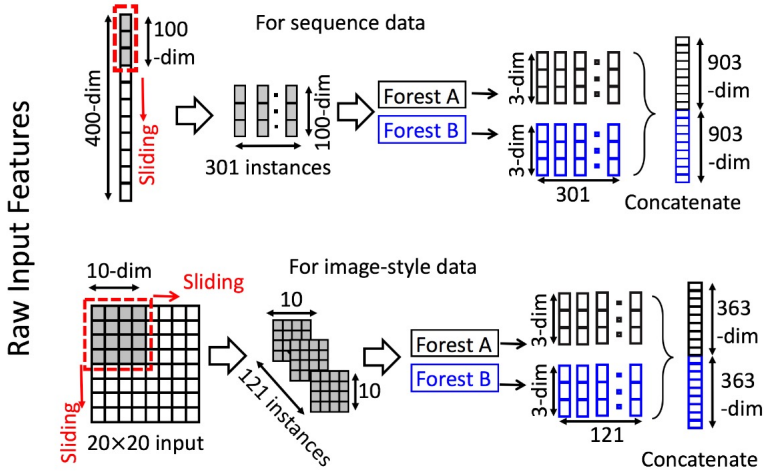
(15 mins)

Development: Deep Forest







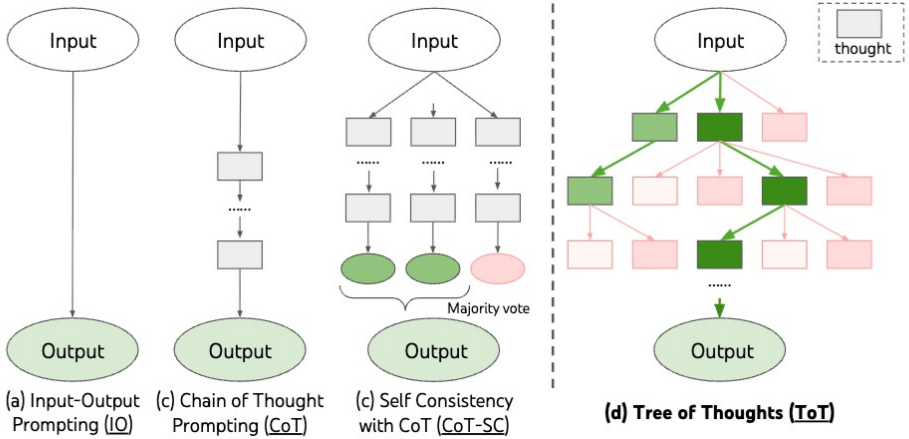


Zhou Z H, Feng J. Deep forest[J]. National science review, 2019, 6(1): 74-86.

Development: Tree of Thought

- ToT abstracts intermediate reasoning units into “thoughts”. While generating multiple next-step ideas, it uses model self-evaluation to select, backtrack, and prune, essentially transforming problem-solving into a heuristic search on a thought tree.

- Compared to traditional chained thought (CoT), ToT performs local branch exploration at each intermediate state and uses a global BFS/DFS strategy for lookahead and backtracking, thus avoiding the local greedy flaw of left-to-right one-shot decoding.



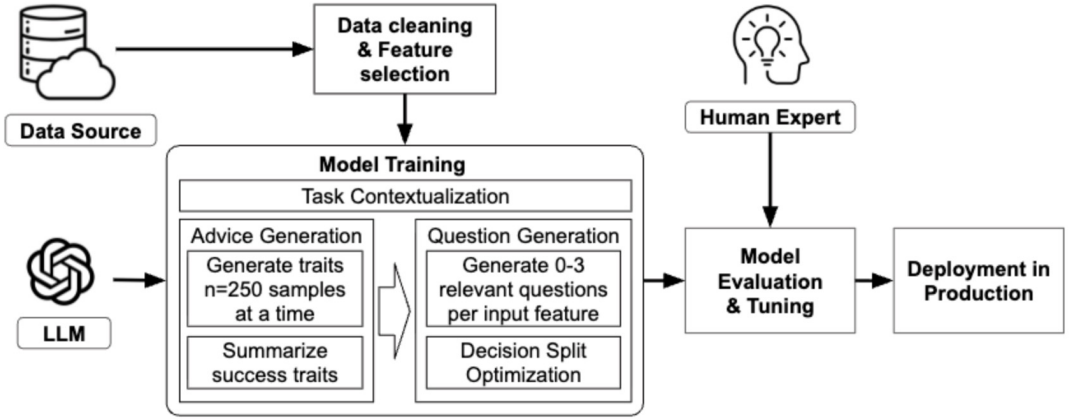
Tree for LLM

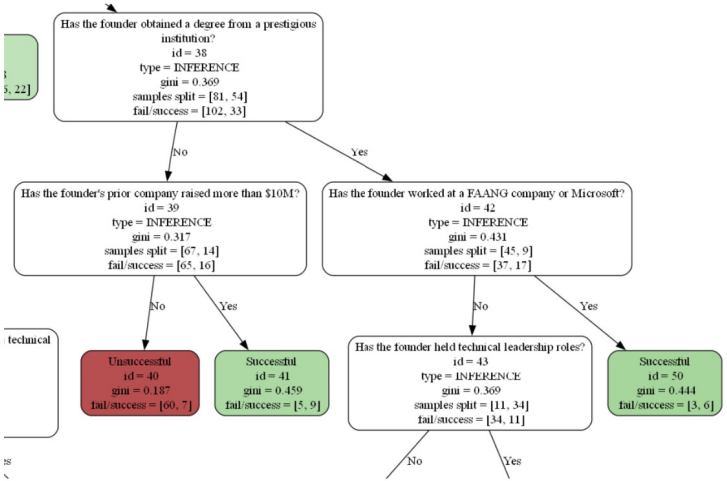
Yao S, Yu D, Zhao J, et al. Tree of thoughts: Deliberate problem solving with large language models[J]. Advances in neural information processing systems, 2023, 36: 11809-11822.

Development: LLM-powered Decision Trees

- LLM-generated questions to select attributes

LLM for Tree

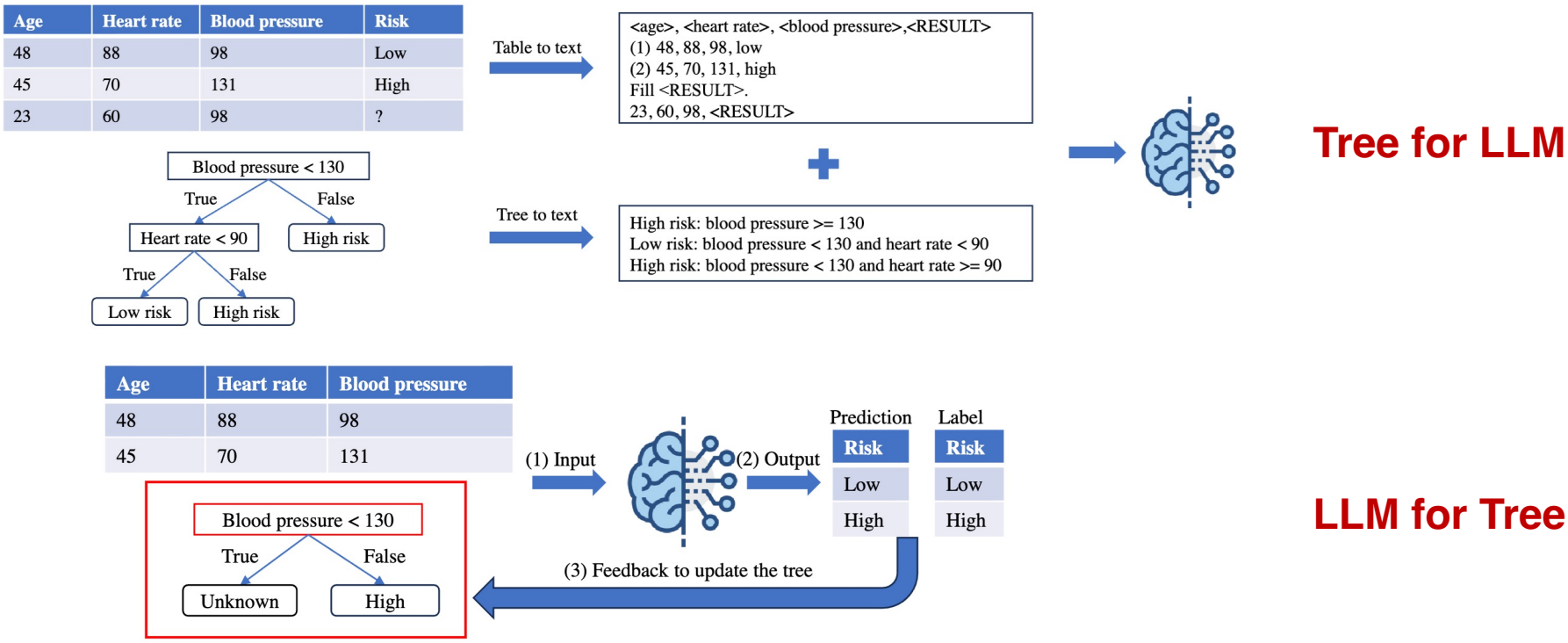




Xiong S, Ihlamur Y, Alican F, et al. Gptree: Towards explainable decision-making via llm-powered decision trees[J]. arXiv preprint arXiv:2411.08257, 2024.

Development: Tree as a Prompt

- The interpretable decision tree "transcribed into text rules" is attached to the original query as an additional hint to feed the black box LLM, and the prediction feedback of the LLM is further used to update the tree.



Li Q, Liang Y, Diao Y, et al. Tree-as-a-Prompt: Boosting Black-Box Large Language Models on Few-Shot Classification of Tabular Data[J]. 2023.

Development: LLM+RL for Decision Tree Generation

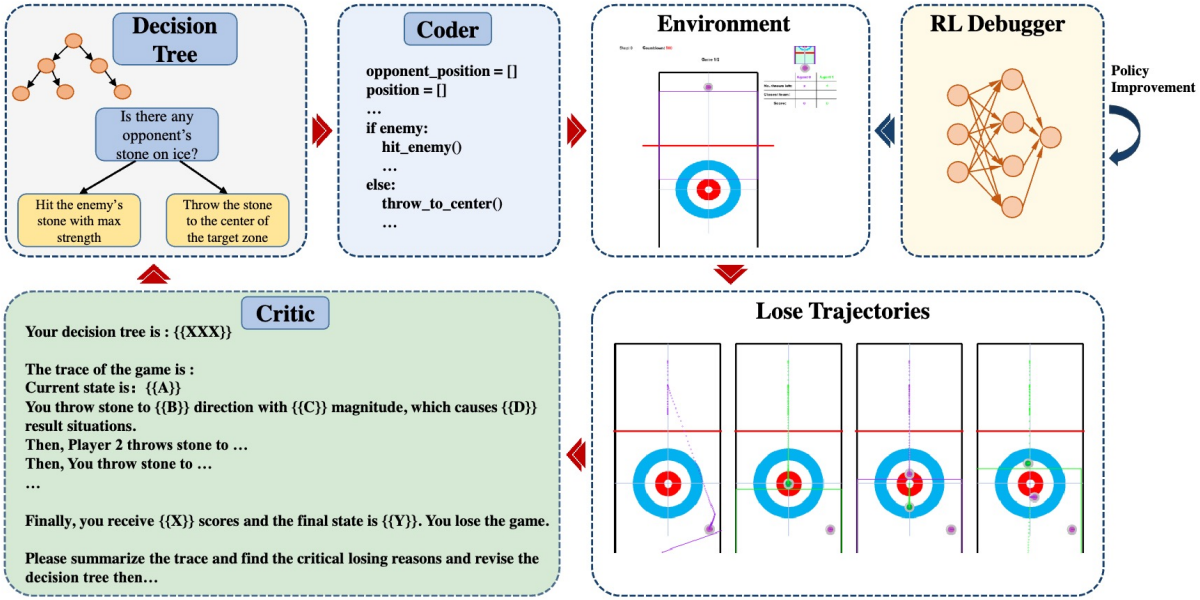
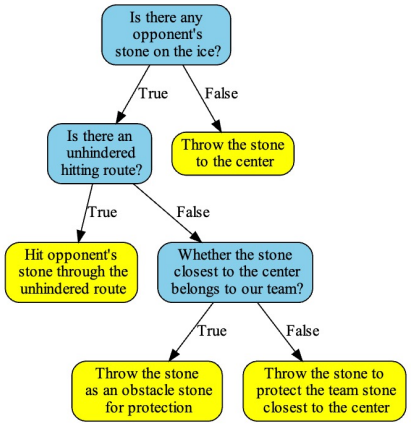
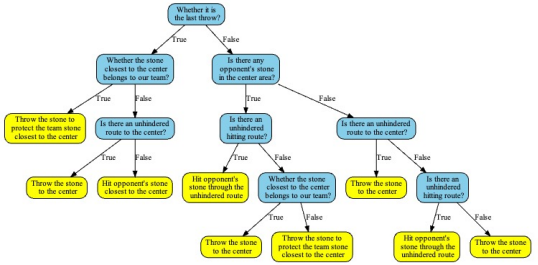


Fig. 2: An illustration of the overall framework.



(a) Final Decision Tree Generated by human.



(b) Final Decision Tree Generated by LLM.

LLM for Tree

Lin J, Zhao J, Liu L, et al. RL-LLM-DT: An Automatic Decision Tree Generation Method Based on RL Evaluation and LLM Enhancement[J]. arXiv preprint arXiv:2412.11417, 2024.


thank
you