CS 5/7320 Artificial Intelligence

Making Simple Decisions AIMA Chapter 16

Introduction slides by Michael Hahsler

Decision network slides by Dan Klein and Pieter Abbeel

What is a simple decision?

- A decision that we make frequently + making it once does not affect the future decisions.
- This means we make them in an **episodic environment**.
- Decision theory formalizes making simple decisions.

Decision theory =

Probability theory (evidence & belief)

+

Utility theory (want)

Decision-theoretic Agents (=Utility-based Agent)

Logical agents	Cannot deal with: • Uncertainty • Conflicting goals
Goal-based agents	• Can only assign goal/not goal to states and find goal states.
Utility-based agents	• Assign a utility value to each state. • A rational agent optimizes the expected utility (i.e., is utility-based). • Utility is related to the external performance measure (see PEAS).

Utility

- A utility function $U(s)$ expresses the desirability of being in state s.
- Utility functions are derived from preferences:

$$U(A) > U(B) \Leftrightarrow A \succ B$$

and

$$U(A) = U(B) \Leftrightarrow A \sim B$$

- It is often enough to know a **ordinal utility function** representing a **ranking** of states to make decisions.

Expected Utility of an Action

We need:

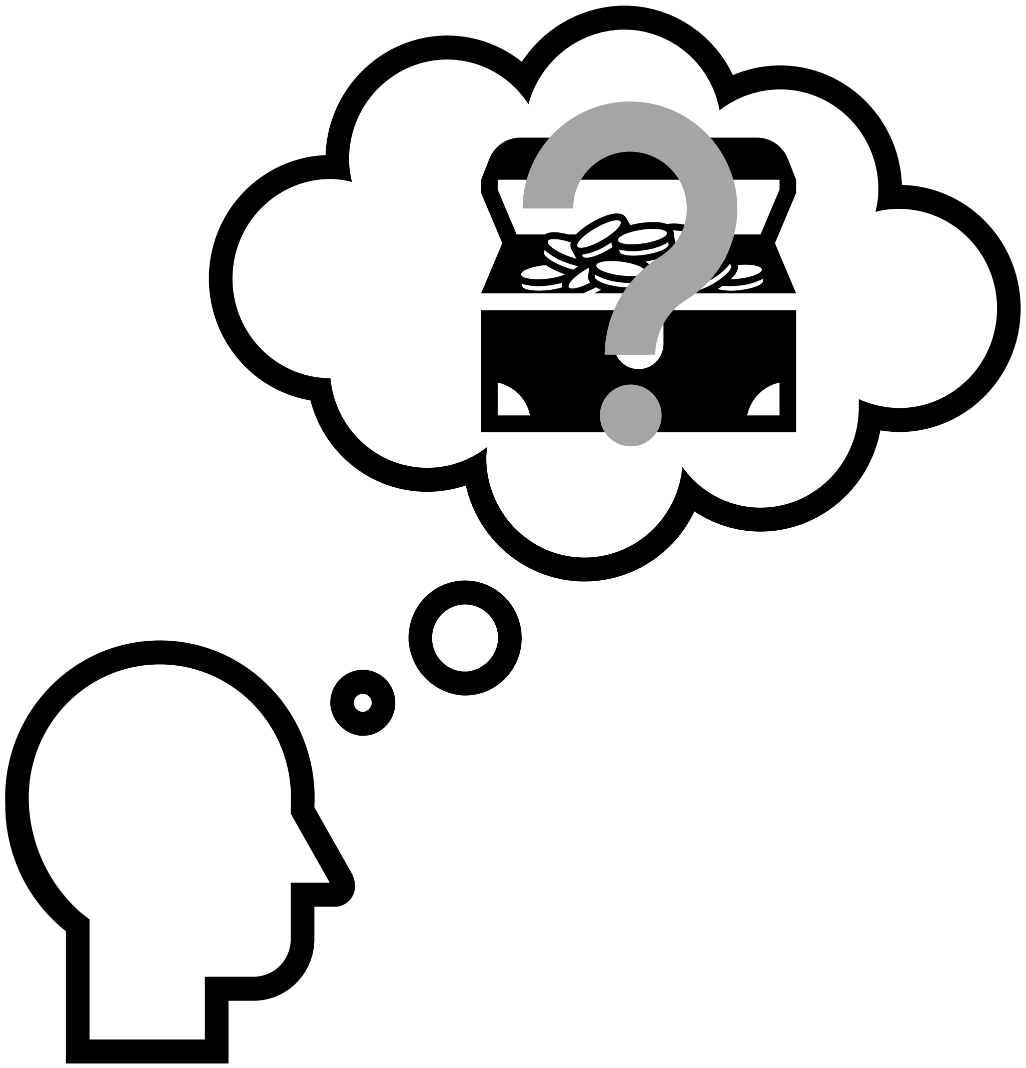
- A **cardinal utility** function $U(s)$ where the number represents levels of absolute satisfaction.
- The probability $P(s)$, that the current state is s.
- Transition probabilities $P(s'|s,a)$.

The probability that action a will get us to state s'

$$P(Result(a) = s') = \sum_{s} P(s)P(s'|s,a)$$

The expected utility of action a over all possible states is

$$EU(a) = \sum_{s'} P(Result(a) = s')U(s')$$



Principle of Maximum Expected Utility (MEU)

Given the expected utility of an action

$$EU(a) = \sum_{s'} P(Result(a) = s')U(s')$$

choose action that maximizes the expected utility:

$$a^* = \text{argmax}_a\, EU(a)$$

Issues:

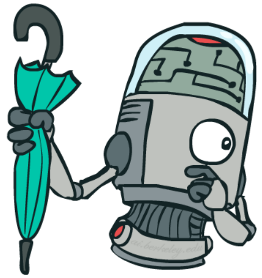
- $P(Result(a) = s')$ needs a casual model.
- $U(s)$ may be hard to estimate. It may depend on what states we can get to from s.
- MEU leads to the “optimizer’s curse” where the estimated expected utility is higher than the actual outcomes with new data.

Decision Networks

Bayes Nets with Actions

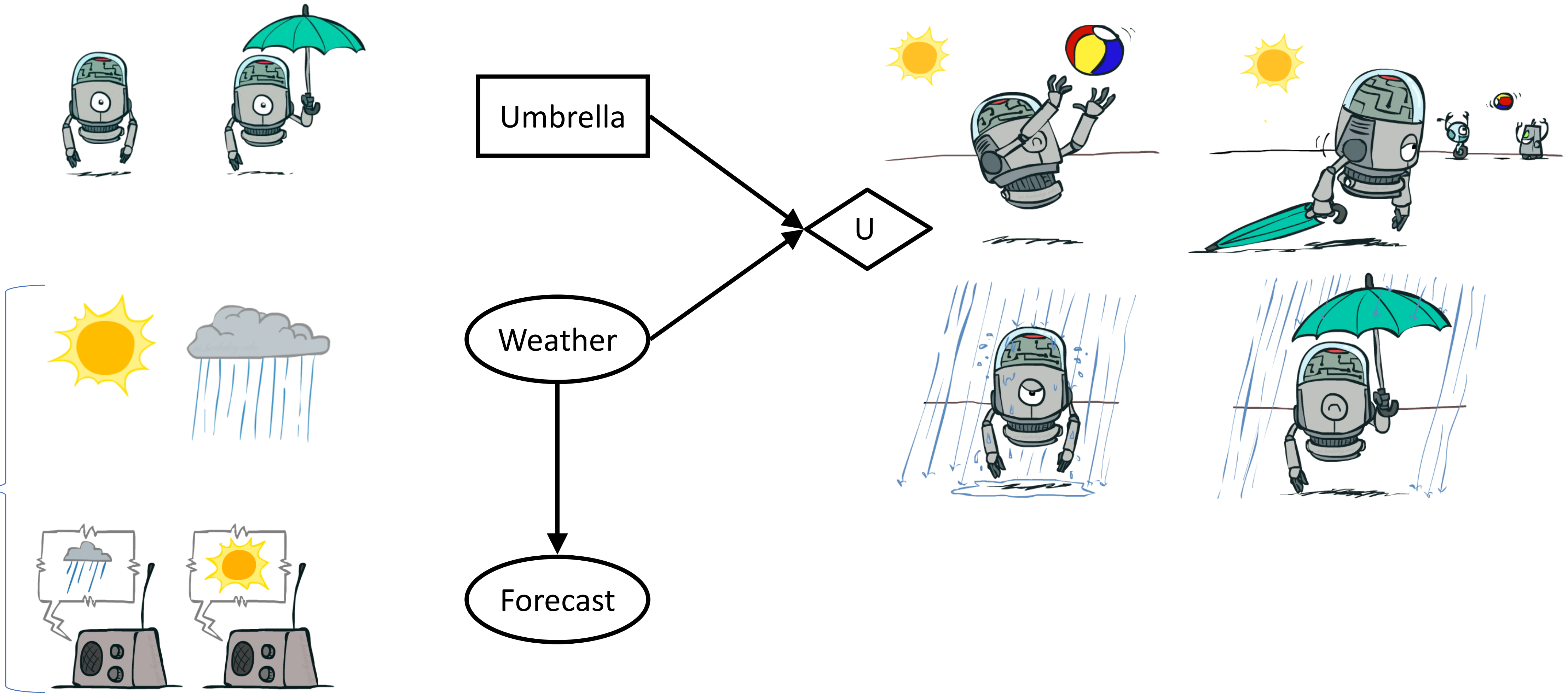
These slides were created by Dan Klein, Pieter Abbeel, Sergey Levine, with some materials from A. Farhadi. All CS188 materials are at http://ai.berkeley.edu

Decision Networks



Decision

Utility



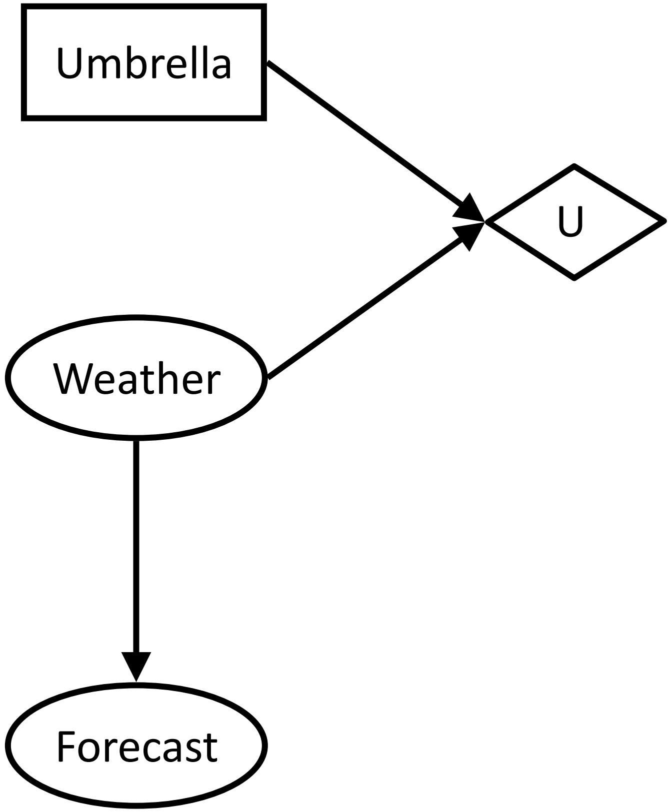
Random Events

Decision Networks

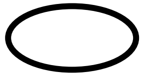
MEU: choose the action which maximizes the expected utility given the evidence.

Decision networks

- Bayes nets with additional nodes for utility and actions.
- Calculate the expected utility for each possible action and choose the best.

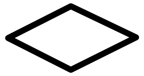


Node types



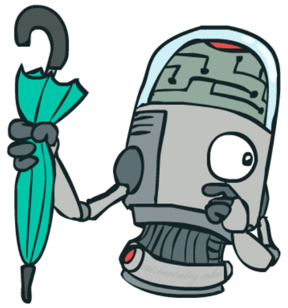
Chance nodes: Random variables in BNs

Action nodes: Cannot have parents, act as observed evidence



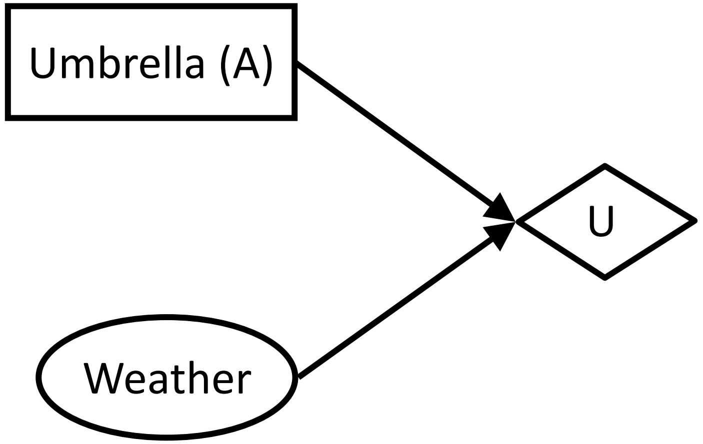
Utility node: Depends on action and chance nodes

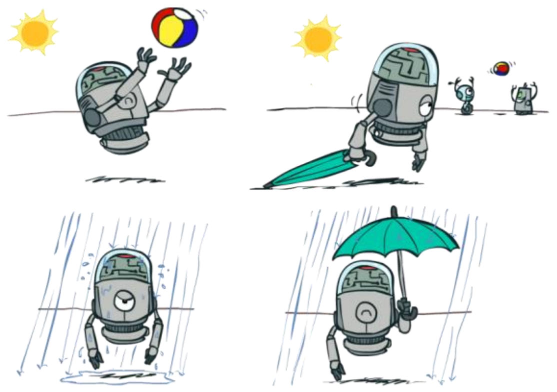
Decision Network without Forecast



Action: Umbrella = leave

$$\mathrm{EU}(\mathrm{leave}) = \sum_w P(w)U(\mathrm{leave}, w)$$
$$= 0.7 \cdot 100 + 0.3 \cdot 0 = 70$$





Action: Umbrella = take

$$\mathrm{EU}(\mathrm{take}) = \sum_w P(w)U(\mathrm{take}, w)$$
$$= 0.7 \cdot 20 + 0.3 \cdot 70 = 35$$

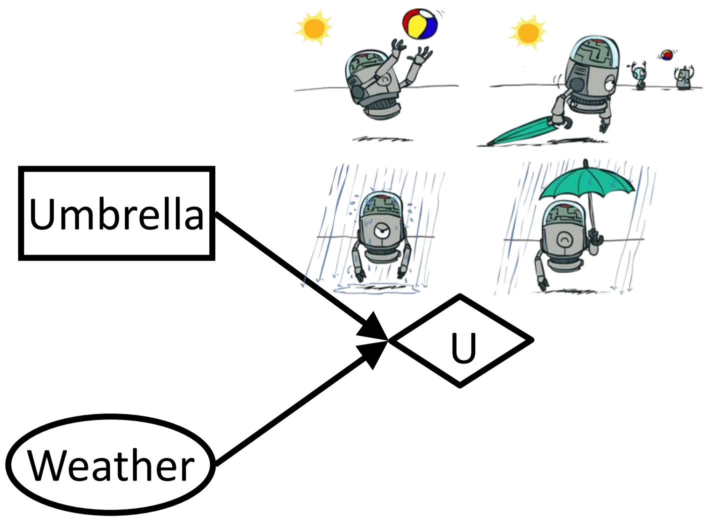
W	$P(W)$
sun	0.7
rain	0.3

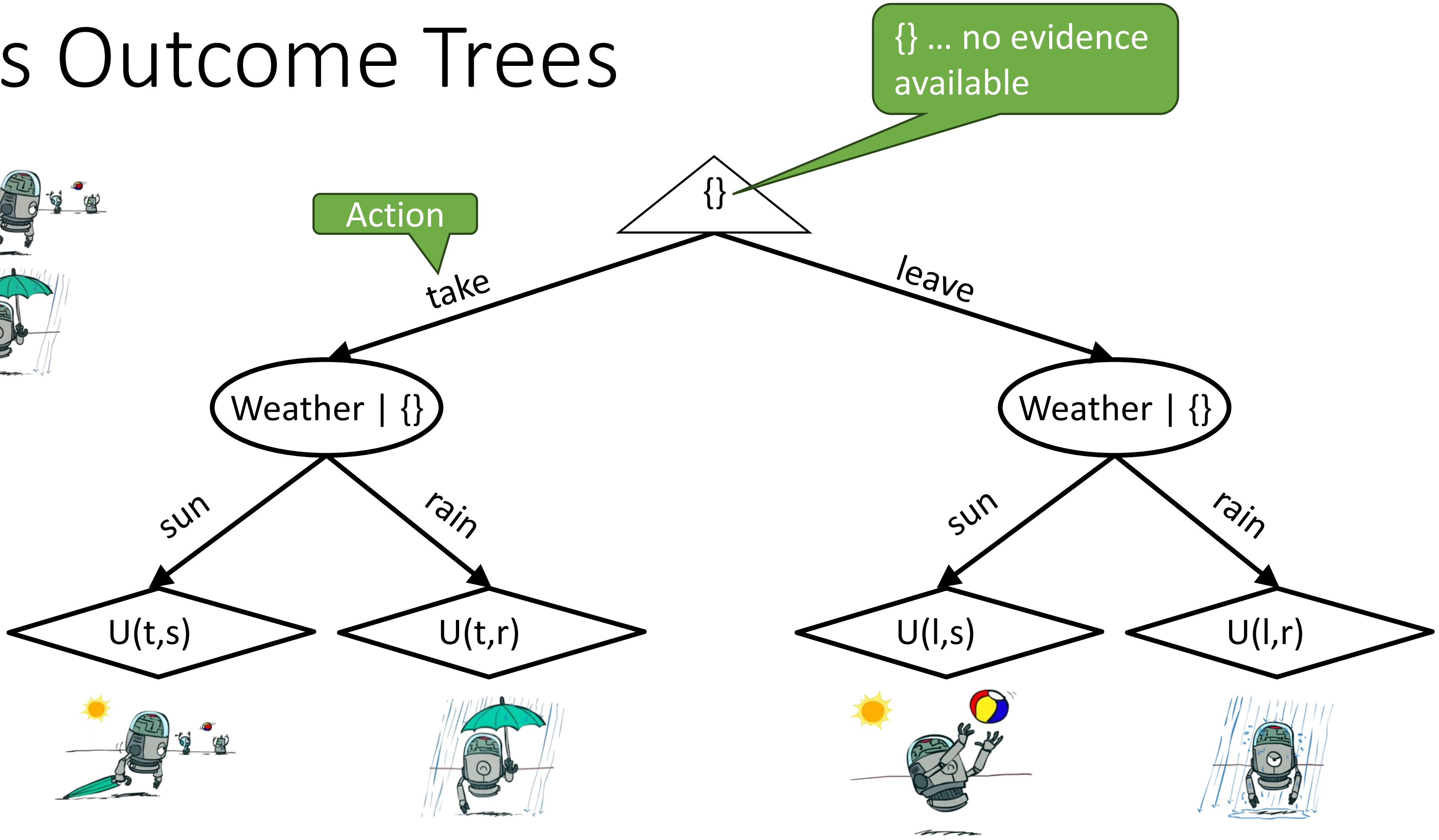
A	W	$U(A,W)$
leave	sun	100
leave	rain	0
take	sun	20
take	rain	70

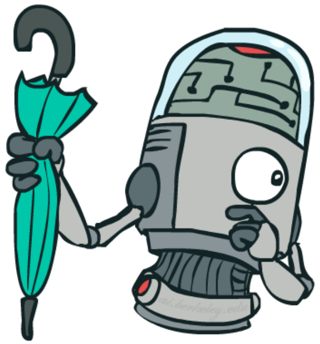
Optimal decision a^* = leave

$$\mathrm{MEU}(\emptyset) = \max_a \mathrm{EU}(a) = 70$$

Decisions as Outcome Trees

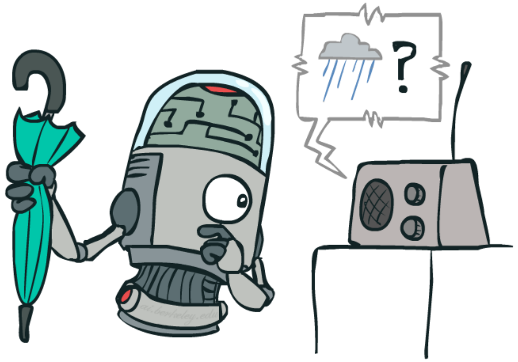






Almost exactly like expectimax tree for stochastic games.

Decision Network with Bad Forecast



Action: Umbrella = leave

$$\mathrm{EU}(\mathrm{leave}|\mathrm{bad}) = \sum_{w} P(w|\mathrm{bad})U(\mathrm{leave}, w)$$

$$= 0.34 \cdot 100 + 0.66 \cdot 0 = 34$$

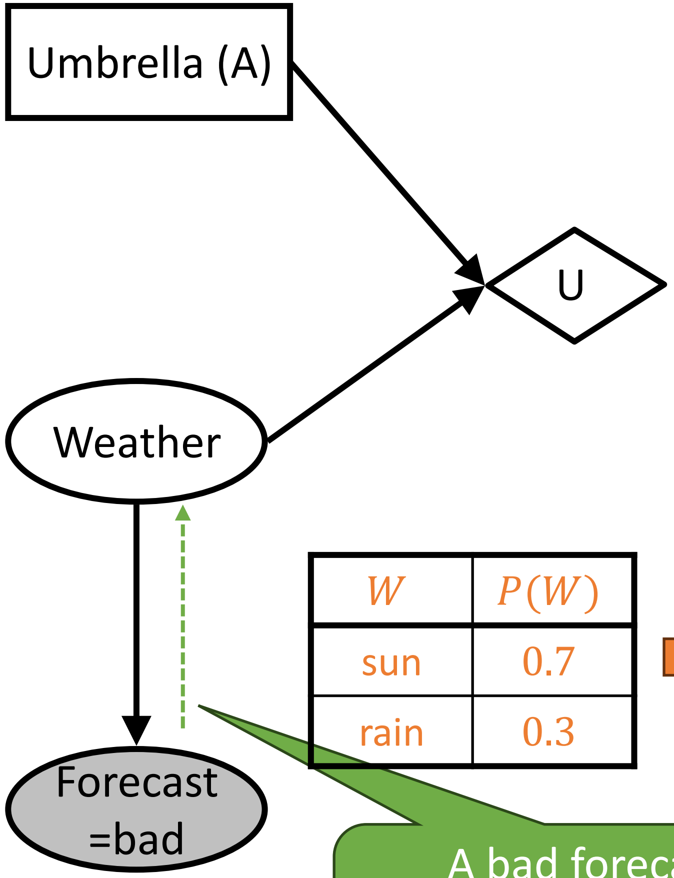
Action: Umbrella = take

$$\mathrm{EU}(\mathrm{take}|\mathrm{bad}) = \sum_{w} P(w|\mathrm{bad})U(\mathrm{take}, w)$$

$$= 0.34 \cdot 20 + 0.66 \cdot 70 = 53$$

Optimal decision = take

$$\mathrm{MEU}(F = \mathrm{bad}) = \max_{a} \mathrm{EU}(a|\mathrm{bad}) = 53$$



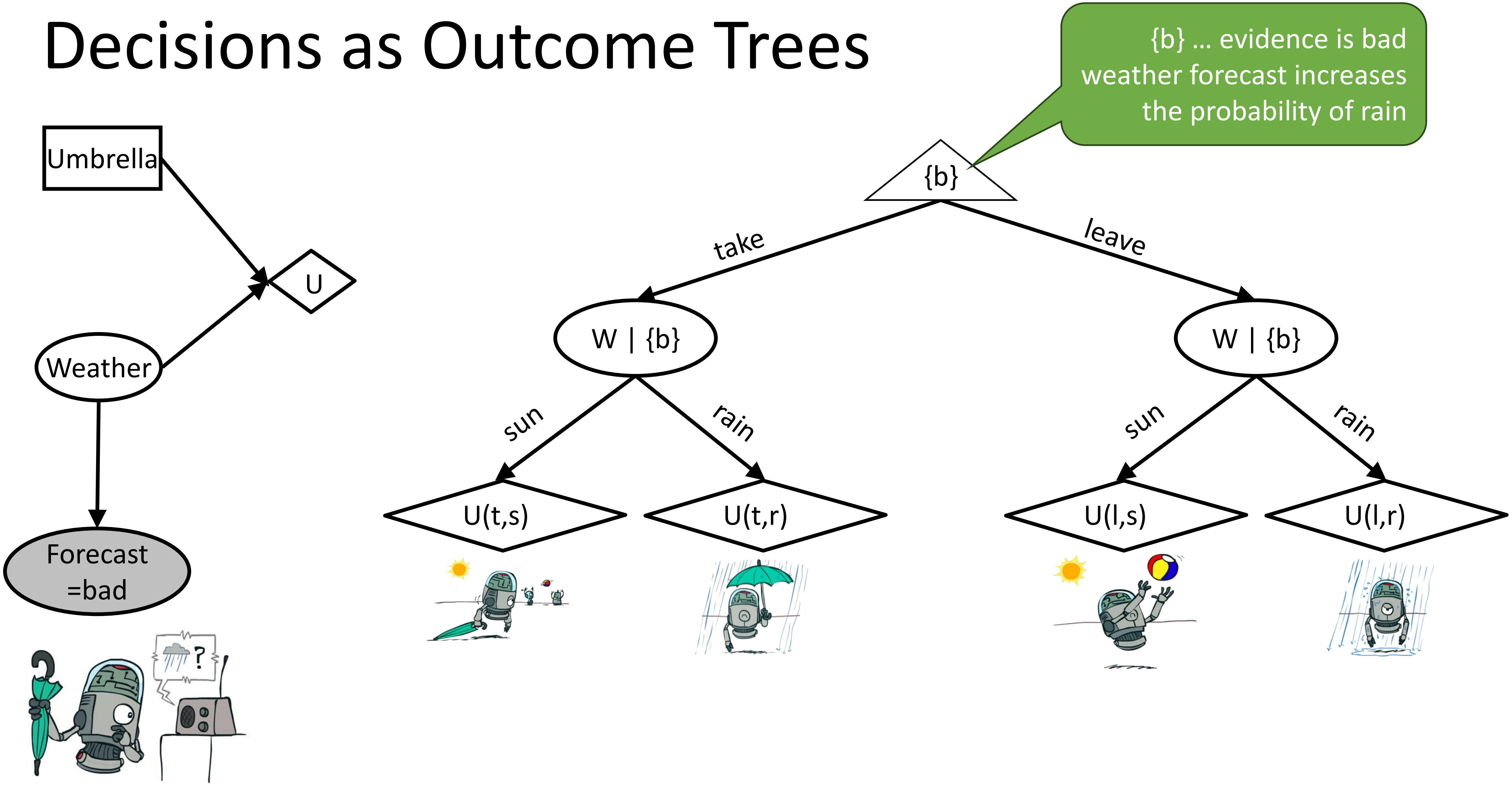
A	W	$U(A,W)$
leave	sun	100
leave	rain	0
take	sun	20
take	rain	70

W	$P(W)$
sun	0.7
rain	0.3

W	$P(W \mid F = bad)$
sun	0.34
rain	0.66

A bad forecast increases the probability of rain!

Decisions as Outcome Trees



Conclusion

Decision networks are an extension of Bayes nets that add actions and utility.

Evidence and independence can be used as for Bayes nets.

Decision networks can be used to make simple decisions (a single, repeating decision, i.e., the environment is episodic).

Sequential decision-making deals with decisions that influence each other and are made over time. This is a more complex decision problem and needs different methods like **Markov Decision Processes.**