

Speech and Language Processing

Dialogue and Conversational Agents
(part 3)

Now that we have a success metric

- Could we use it to help drive learning?
- In recent work we use this metric to help us learn an optimal **policy** or **strategy** for how the conversational agent should behave

New Idea: Modeling a dialogue system as a probabilistic agent

- A conversational agent can be characterized by:
 - The current knowledge of the system
 - A set of **states** S the agent can be in
 - a set of **actions** A the agent can take
 - A **goal** G , which implies
 - A success metric that tells us how well the agent achieved its goal
 - A way of using this metric to create a strategy or **policy** π for what action to take in any particular state.

What do we mean by actions A and policies π ?

- Kinds of decisions a conversational agent needs to make:
 - When should I ground/confirm/reject/ask for clarification on what the user just said?
 - When should I ask a directive prompt, when an open prompt?
 - When should I use user, system, or mixed initiative?

A threshold is a human-designed policy!

- Could we learn what the right action is
 - Rejection
 - Explicit confirmation
 - Implicit confirmation
 - No confirmation
- By learning a policy which,
 - given various information about the current state,
 - dynamically chooses the action which maximizes dialogue success

Another strategy decision

- Open versus directive prompts
- When to do mixed initiative

- How we do this optimization?
- Markov Decision Processes

Review: Open vs. Directive Prompts

- **Open prompt**
 - System gives user very few constraints
 - User can respond how they please:
 - “How may I help you?” “How may I direct your call?”
- **Directive prompt**
 - Explicit instructs user how to respond
 - “Say yes if you accept the call; otherwise, say no”

Review: Restrictive vs. Non-restrictive gramamrs

- **Restrictive grammar**
 - Language model which strongly constrains the ASR system, based on dialogue state
- **Non-restrictive grammar**
 - Open language model which is not restricted to a particular dialogue state

Kinds of Initiative

- How do I decide which of these initiatives to use at each point in the dialogue?

Grammar	Open Prompt	Directive Prompt
Restrictive	<i>Doesn't make sense</i>	System Initiative
Non-restrictive	User Initiative	Mixed Initiative

Modeling a dialogue system as a probabilistic agent

- A conversational agent can be characterized by:
 - The current knowledge of the system
 - A set of **states** S the agent can be in
 - a set of **actions** A the agent can take
 - A **goal** G , which implies
 - A success metric that tells us how well the agent achieved its goal
 - A way of using this metric to create a strategy or **policy** π for what action to take in any particular state.

Goals are not enough

- Goal: user satisfaction
- OK, that's all very well, but
 - Many things influence user satisfaction
 - We don't know user satisfaction til after the dialogue is done
 - How do we know, state by state and action by action, what the agent should do?
- We need a more helpful metric that can apply to each state

Utility

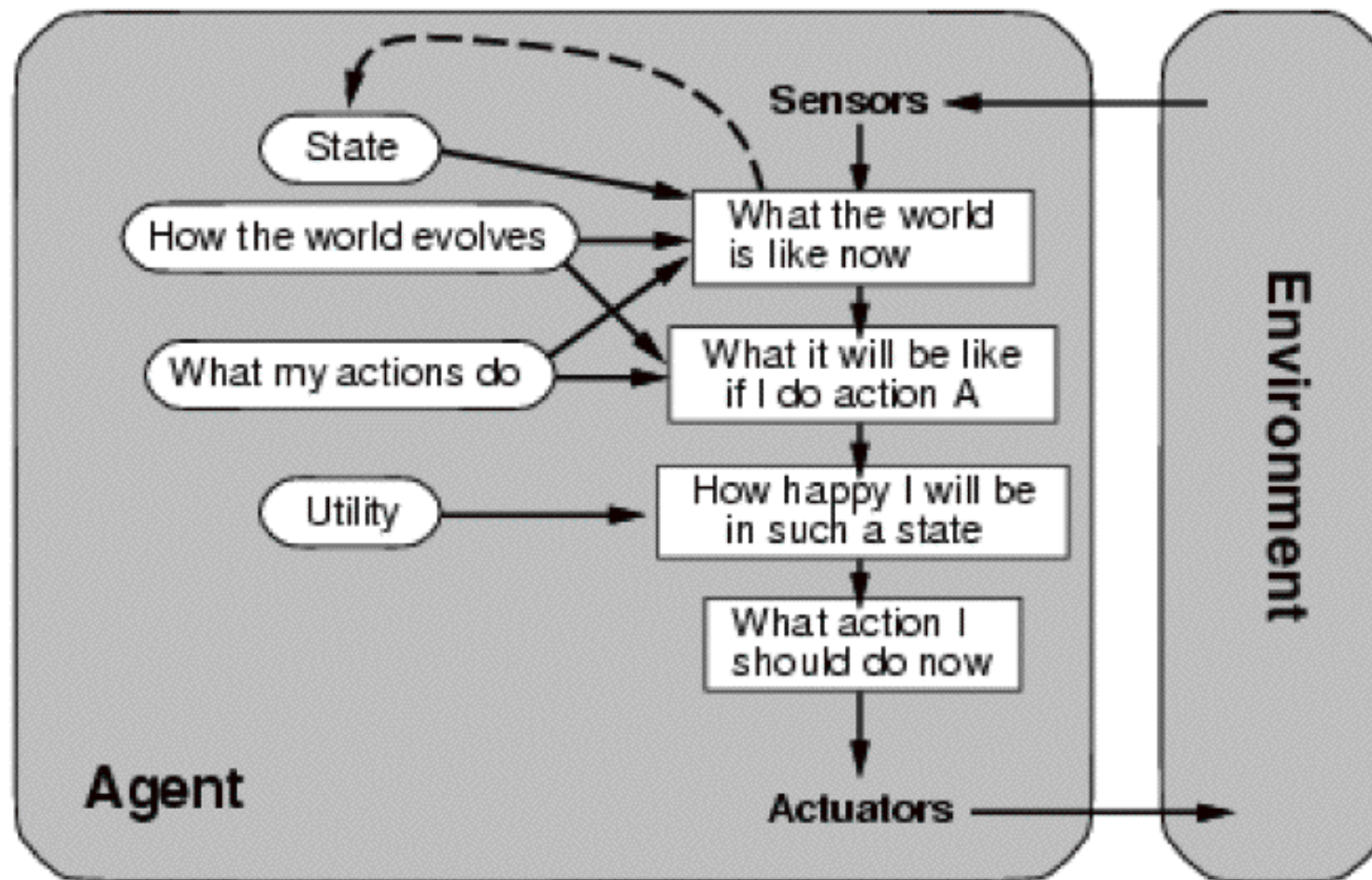
- **A utility function**
 - maps a state or state sequence
 - onto a real number
 - describing the goodness of that state
 - I.e. the resulting “happiness” of the agent
- **Principle of Maximum Expected Utility:**
 - A rational agent should choose an action that maximizes the agent’s expected utility

Maximum Expected Utility

- Principle of Maximum Expected Utility:
 - A rational agent should choose an action that maximizes the agent's expected utility
- Action A has possible outcome states $Result_i(A)$
- E : agent's evidence about current state of world
- Before doing A , agent estimates prob of each outcome
 - $P(Result_i(A) | Do(A), E)$
- Thus can compute expected utility:

$$EU(A | E) = \sum_i P(Result_i(A) | Do(A), E) U(Result_i(A))$$

Utility (Russell and Norvig)



Markov Decision Processes

- Or MDP
- Characterized by:
 - a set of states S an agent can be in
 - a set of actions A the agent can take
 - A reward $r(a,s)$ that the agent receives for taking an action in a state
- (+ Some other things I'll come back to (gamma, state transition probabilities))

MDP

- We can think of a dialogue as a trajectory in state space
- The best policy π^* is the one with the greatest expected reward over all trajectories
- How to compute a reward for a state sequence?

$$S_1 \longrightarrow a_1, r_1 \quad S_2 \longrightarrow a_2, r_2 \quad S_3 \longrightarrow a_3, r_3 \quad \dots$$

Reward for a state sequence

- One common approach: discounted rewards
- Cumulative reward Q of a sequence is discounted sum of utilities of individual states

$$Q([s_0, a_0, s_1, a_1, s_2, a_2 \dots]) = R(s_0, a_0) + \gamma R(s_1, a_1) + \gamma^2 R(s_2, a_2) + \dots$$

- Discount factor γ between 0 and 1
- Makes agent care more about current than future rewards; the more future a reward, the more discounted its value

The Markov assumption

- MDP assumes that state transitions are Markovian

$$P(s_{t+1} \mid s_t, s_{t-1}, \dots, s_0, a_t, a_{t-1}, \dots, a_0) = P_T(s_{t+1} \mid s_t, a_t)$$

Expected reward for an action

- Expected cumulative reward $Q(s,a)$ for taking a particular action from a particular state can be computed by **Bellman equation**:

$$Q(s, a) = R(s, a) + \gamma \sum_{s'} P(s'|s, a) \max_{a'} Q(s', a')$$

- Expected cumulative reward for a given state/action pair is:
 - immediate reward for current state
 - + expected discounted utility of all possible next states s'
 - Weighted by probability of moving to that state s'
 - And assuming once there we take optimal action a'

What we need for Bellman equation

- A model of $p(s' | s, a)$
- Estimate of $R(s, a)$
- How to get these?
- If we had labeled training data
 - $P(s' | s, a) = C(s, s', a) / C(s, a)$
- If we knew the final reward for whole dialogue $R(s_1, a_1, s_2, a_2, \dots, s_n)$
- Given these parameters, can use **value iteration algorithm** to learn Q values (pushing back reward values over state sequences) and hence best policy

Final reward

- What is the final reward for whole dialogue $R(s_1, a_1, s_2, a_2, \dots, s_n)$?
- This is what our automatic evaluation metric PARADISE computes!
- The general goodness of a whole dialogue!!!!

How to estimate $p(s' | s, a)$ without labeled data

- **Have random conversations with real people**
 - Carefully hand-tune small number of states and policies
 - Then can build a dialogue system which explores state space by generating a few hundred random conversations with real humans
 - Set probabilities from this corpus
- **Have random conversations with simulated people**
 - Now you can have millions of conversations with simulated people
 - So you can have a slightly larger state space

State of the art

- Only a few such systems
 - From (former) ATT Laboratories researchers, now dispersed
 - And Cambridge UK lab
- Hot topics:
 - Partially observable MDPs (POMDPs)
 - We don't REALLY know the user's state (we only know what we THOUGHT the user said)
 - So need to take actions based on our BELIEF , I.e. a probability distribution over states rather than the "true state"

Summary

- Utility-based conversational agents
 - Policy/strategy for:
 - Confirmation
 - Rejection
 - Open/directive prompts
 - Initiative
 - +?????
 - MDP
 - POMDP

Summary

- The Linguistics of Conversation
- Basic Conversational Agents
 - ASR
 - NLU
 - Generation
 - Dialogue Manager
- Dialogue Manager Design
 - Finite State
 - Frame-based
 - Initiative: User, System, Mixed
- VoiceXML
- Information-State
 - Dialogue-Act Detection
 - Dialogue-Act Generation
- Evaluation
- Utility-based conversational agents
 - MDP, POMDP