# Empirical Evaluation of a Reinforcement Learning Spoken Dialogue System

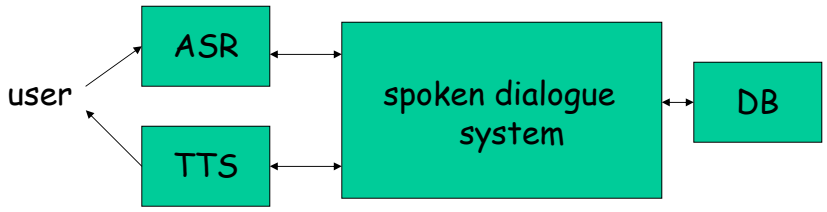
S. Singh, M. Kearns, D. Litman, M. Walker
AT&T Labs
AAAI 2000

## Motivation

- Builders of spoken dialogue systems face fundamental design choices that strongly influence system performance
- Can performance be improved by adapting a system's dialogue strategy via reinforcement learning?

## Spoken Dialogue Systems

- Provide automated telephone access to DB
- Front end: ASR + TTS
- Back end: DB
- Middle: dialogue strategy is key component




## Typical System Design: Sequential Search

- Choose and implement a particular, "reasonable" dialogue strategy
- Field system, gather dialogue data
- Do simple statistical analyses
- Re-field improved dialogue strategy
- Can only examine a handful of strategies

## Why Reinforcement Learning?

- ASR output is noisy; user population leads to stochastic behavior
- Design choices have long-term impact; temporal credit assignment problem
- Many design choices can be fixed, but
  - Initiative strategy
  - Confirmation strategy
- Many different performance criteria

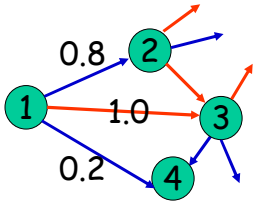
## Example: Initiative Strategy

- System initiative vs. user initiative:
  - "Please state your departure city."
  - "Please state your desired itinerary."
  - "How can I help you?"
- Influences user expectations
- ASR grammar must be chosen accordingly
- Best choice may differ from state to state!
- May depend on user population & task

Suited to MDPs and Reinforcement Learning!

# Markov Decision Processes

- System state s (in S)
- System action a in (in A)
- Transition probabilities P(s'|s,a)
- Reward function R(s,a) (stochastic)
- Fast algorithms for optimal policy
- Our application: P(s'|s,a) models the population of users




# SDSs as MDPs

Initial system utterance

Initial user utterance

$s_1 \rightarrow u_1 \rightarrow s_2 \rightarrow u_2 \rightarrow s_3 \rightarrow u_3 \rightarrow \ldots$

+ system logs

Actions have prob. outcomes

$a_1 \rightarrow e_1 \rightarrow a_2 \rightarrow e_2 \rightarrow a_3 \rightarrow e_3 \rightarrow \ldots$

estimate transition probabilities...
P(next state | current state & action)
...and rewards...
R(current state, action)
...from set of exploratory dialogues

Violations of Markov property! Will this work?

## The RL Approach

(Levin, Pieraccini, Eckert; Singh, Kearns, Litman, Walker)

- Build initial system that is deliberately exploratory wrt state and action space
- Use dialogue data from initial system to build a Markov decision process (MDP)
- Use methods of reinforcement learning to compute optimal strategy of the MDP
- Re-field (improved?) system given by the optimal policy

## The Application

- Dialogue system providing telephone access to a DB of activities in NJ
- Want to obtain 3 attributes:
  - activity type (e.g., wine tasting)
  - location (e.g., Lambertville)
  - time (e.g., morning)
- Failure to bind an attribute: query DB with don't-care


# The State Space

| *Feature* | *Values* | *Explanation* |
|---|---|---|
| Attribute (A) | 1,2,3 | Which attribute is being worked on |
| Confidence/ Confirmed (C) | 0,1,2<br>3,4 | 0,1,2 for low, medium and high ASR confidence<br>3.4 for explicitly confirmed, disconfirmed |
| Value (V) | 0,1 | Whether value has been obtained for current attribute |
| Tries (T) | 0,1,2 | How many times current attr has been asked |
| Grammar (G) | 0,1 | Whether open or closed grammar was used |
| History (H) | 0,1 | Whether trouble on any previous attribute |

N.B. Non-state variables record attribute values; state does not condition on previous attributes!

Will this work!

# Sample Actions

- Initiative (when T = 0):
  - open or constrained prompt?
  - open or constrained grammar?
  - N.B. might depend on H, A,…
- Confirmation (when V = 1)
  - confirm or move on or re-ask?
  - N.B. might depend on C, H, A,…
- Only allowed "reasonable" actions
- Results in 42 states with (binary) choices
- Small state space, large policy space

## The Experiment

- Designed 6 specific tasks, each with web survey
- Gathered 75 internal subjects
- Split into training and test, controlling for M/F, native/non-native, experienced/inexperienced
- 54 training subjects generated 311 dialogues
- Exploratory training dialogues used to build MDP
- Optimal strategy for objective TASK COMPLETION computed and implemented
- 21 test subjects performed tasks and web surveys for modified system generated 124 dialogues
- Did statistical analyses of performance changes

## Reward Function

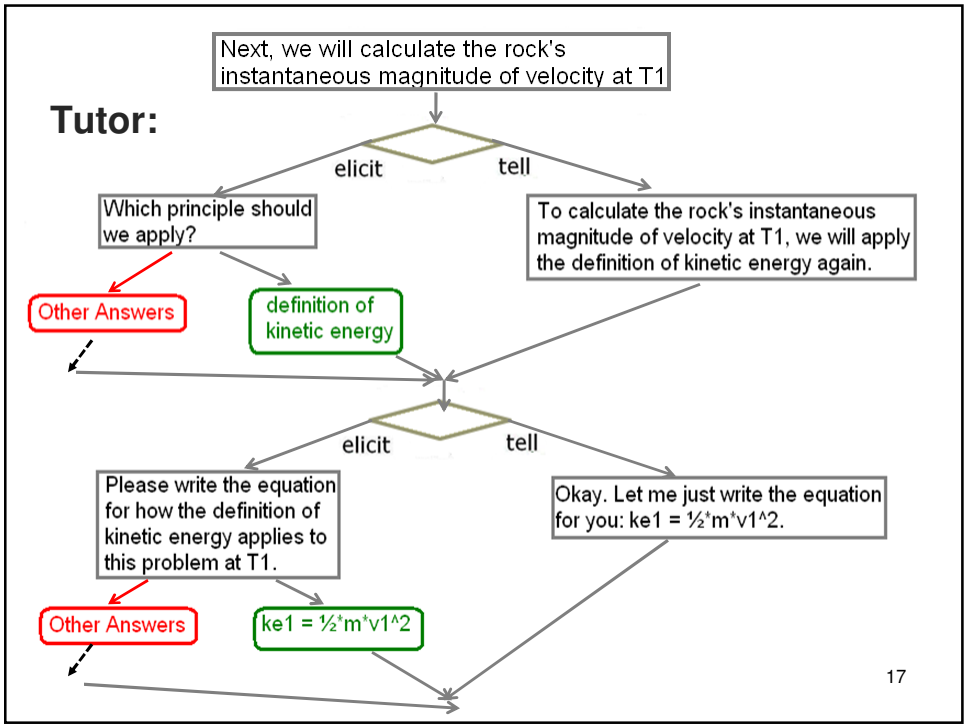
- Objective task completion:
  - -1 for an incorrect attribute binding
  - 0,1,2,3 correct attribute bindings

## Main Result

- Objective task completion:
  - train mean ~ 1.722, test mean ~ 2.176
  - two-sample t-test p-value ~ 0.0289

## Caveats

- Must still choose states and actions
- Must be exploratory with taste
- Data sparsity
- Violations of the Markov property
- A formal framework and methodology, hopefully automating one important step in system design




## Do Micro-Level Tutorial Decisions Matter: Applying Reinforcement Learning To Induce Pedagogical Tutorial Tactics: *Min Chi dissertation*

What is the best action for the **tutor (agent)** to take at any **tutorial context (state)** in order to maximize **students' learning (delayed reward)** at the end?

18

## Representational Choices Makes a Huge Difference

| | Study 2 (didn't work) | Study 3 (worked) |
|---|---|---|
| **Training Corpus** | Exploratory | Exploratory, Suboptimal, Combined |
| | | |
| **Reward** | NLG, median split | NLG or (1-NLG) |
| | | |
| **State Representation** | 18 features<br>Median Split Discretization<br>Maximum: 4<br>Greedy feature selection | 50 features<br>K-means Discretization<br>Maximum: 6<br>11 feature selection methods |

19

## Conclusions

- MDPs and RL a natural and promising framework for (automated) dialogue strategy design
- Have algorithms for **learning** dialogue strategy from data
- Broadly applicable: varying sensors (ASR, NL) and actions (initiative, confirmation, sales); web-based dialogue systems
- Our application: first empirical test of formalism
- Resulted in measurable and significant system improvements
- Care in application: choice of states and actions; gathering exploratory data; choice of reward to optimize