Reinforcement Learning Fundamentals

Lecture 8: Contextual RL and Full RL

Dr Sandeep Manjanna
Assistant Professor, Plaksha University
sandeep.manjanna@plaksha.edu.in

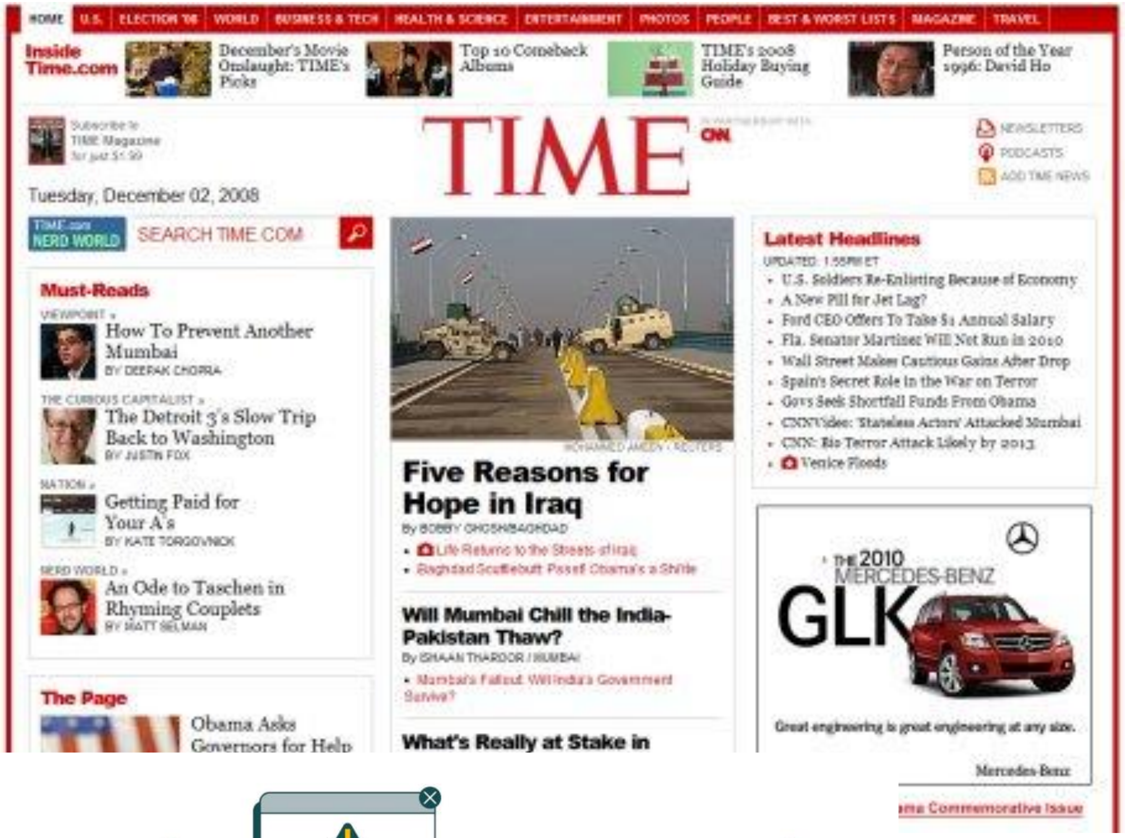
Any implementation of UCB algorithm for Bandit problem?

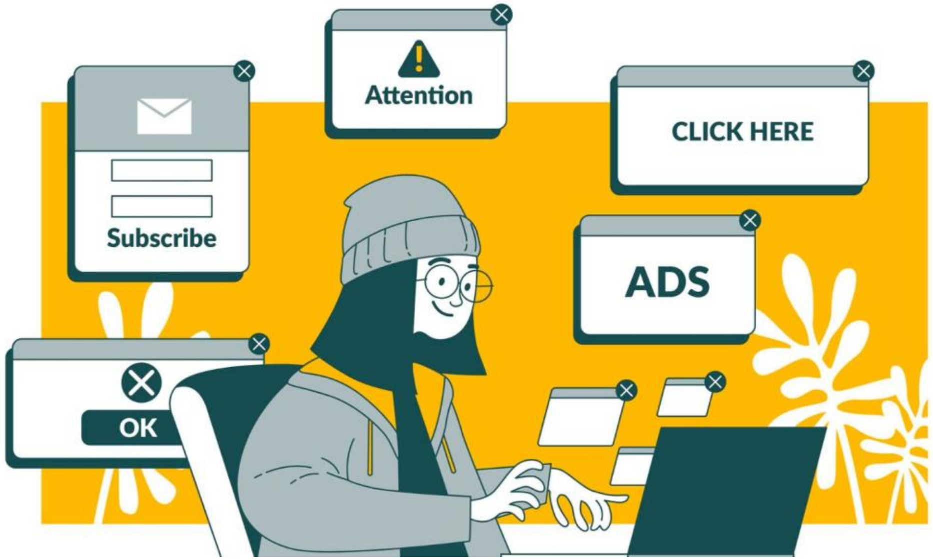
In today's class…

- Contextual Bandit
- Temporal difference
- Full RL Problem

Contextual Bandits

- Customization
  - Different news / ads for different users.
- Different recommendation for different users
  - One UCB for each user?
- **Not a good solution. Why??**
  - Hard to Train
    - Lots and lots of users
    - Less experience with each user
    - Preferences / relevance change over time





Contextual Bandits

- Can we group users into categories? And then run a UCB for each user group.

- How to categorize / what are the possible parameters we can use to categorize?
    - Age, Gender, Browsing behaviour, Location
    - Demographic or behaviour or engagement features, etc.

- We don't need to know the person X, all we need to know is the attributes of X.

Contextual Bandits

- Now, assume that the parameters of the reward distributions are determined by a set of hyperparameters (features / attributes of the users).
    - $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}$ of the reward distribution are a function of the features / attributes of the user.

- The statistic used to choose an arm is now dependent on these features or attributes.

- Instead of learning ***Q(a)***, we will learn ***Q(s,a)***. Now we will track ***Q(s,a)*** and $n_{s,a}$.
    - ***Q(s,a)*** represents the value of taking action ***a*** in the context ***s***.
    - $n_{s,a}$ represents the number of times action ***a*** is chosen w.r.t context ***s***.

- Can action ***a*** also be represented with a set of features? What's the use?
    - Change of stories / ads will not need to start a new bandit.

Contextual Bandits

- LinUCB by Li et al., in 2010
- One of the more popular contextual bandit algorithms
- *Predicted expected reward* assumed to be a linear function of the features
  – Use ridge regression to fit parameters
  – Can derive upper confidence bounds for the regression fit
  – Use UCB like action selection
  – Gives better performance with lesser "training" data

- Contextual bandit is a powerful extension of Bandit setting.

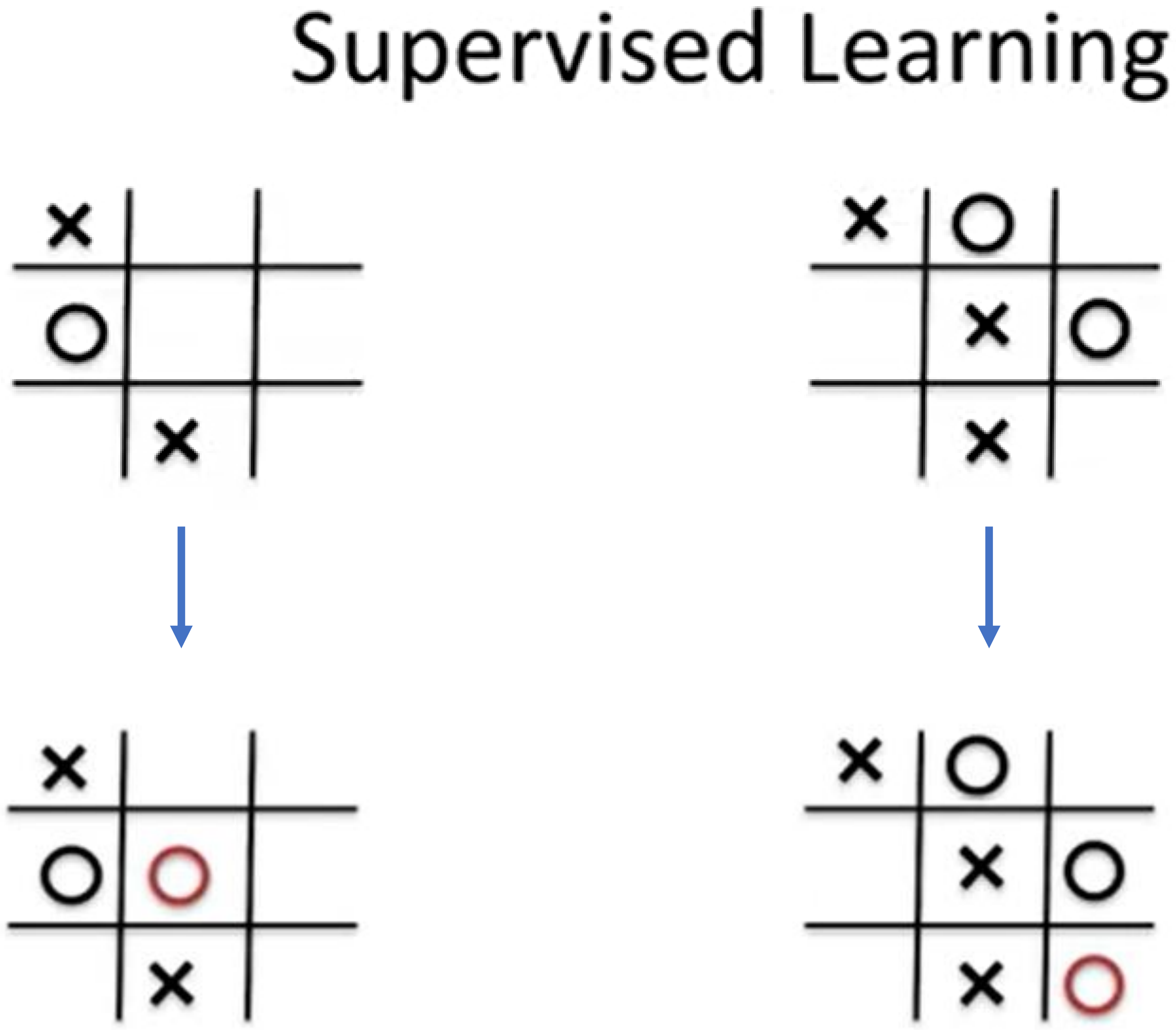
Bandit / Immediate
Only actions

Contextual Bandit
Actions, and Context

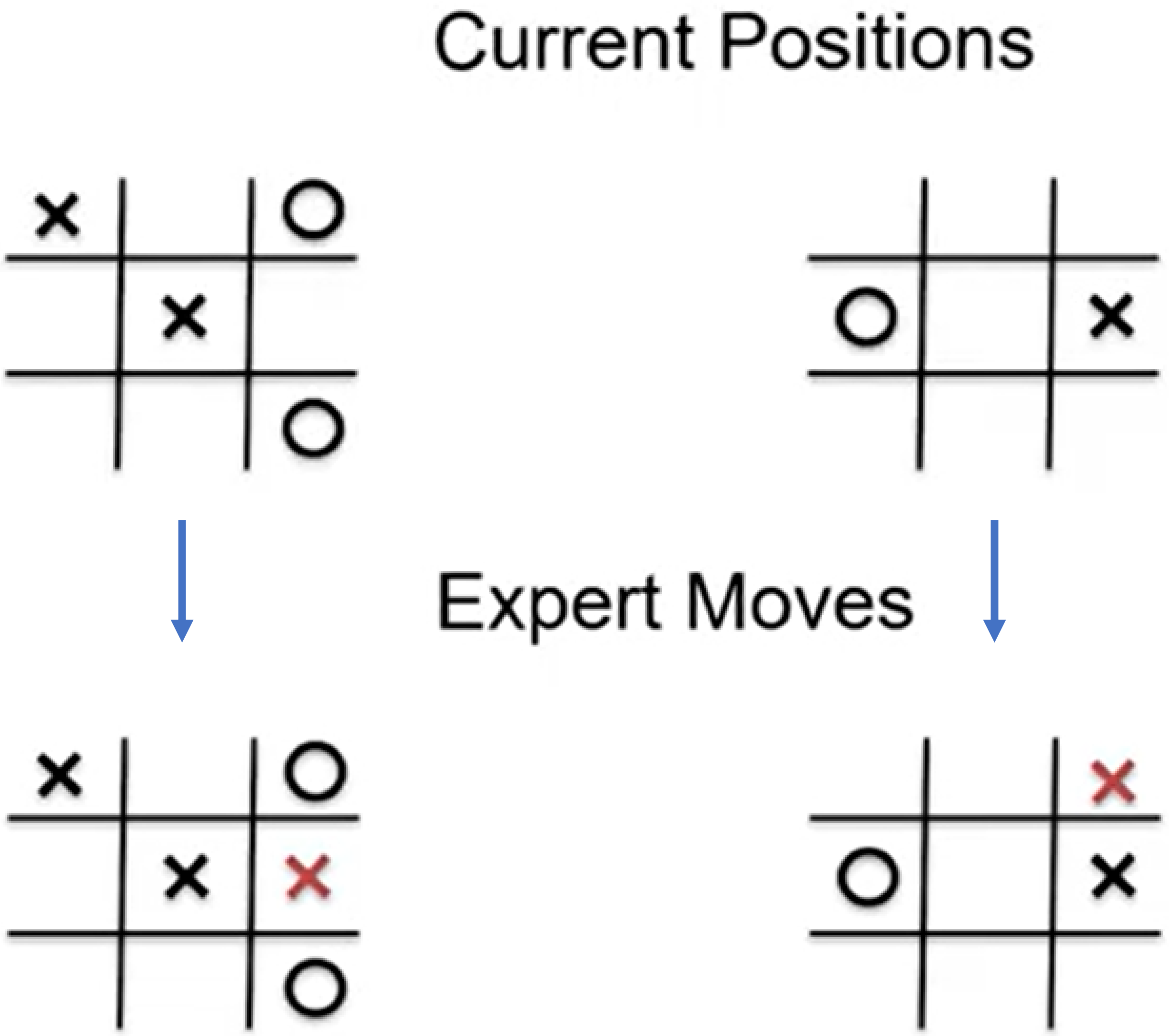
Full RL Problem
Actions, Context, and Sequence

Slide from Prof. Ravindran's course: "Reinforcement Learning."

Example game: Tic-Tac-Toe



Example game: Tic-Tac-Toe



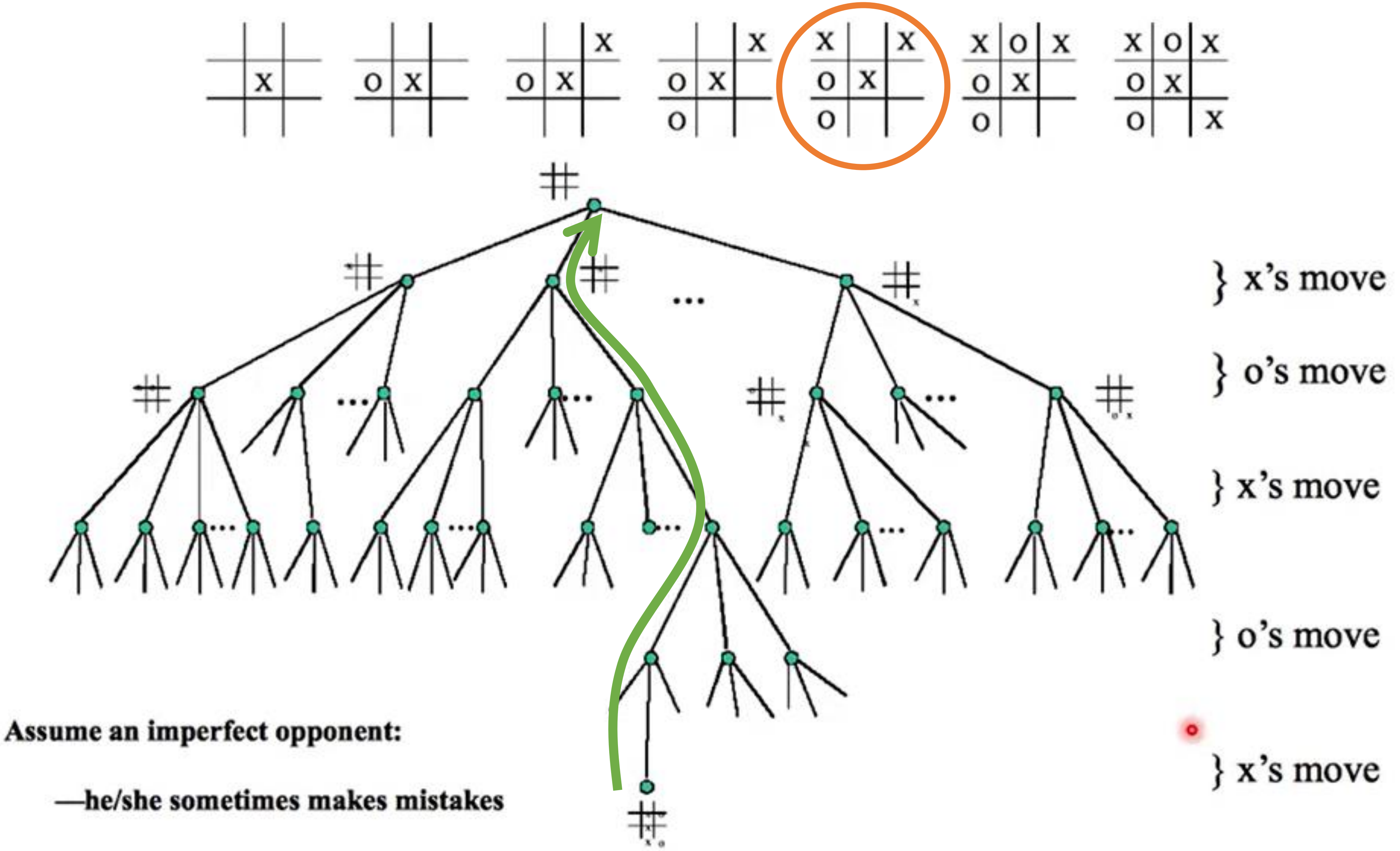
Example game: Tic-Tac-Toe

How to do this with Reinforcement Learning?

- Don’t have to tell how to play. Only inform about the legal moves.
- Learn from evaluation
    - Win gives 1 point
    - Loss gives -1 point
    - Draw gives 0 points
- Learn by playing repeatedly

MENACE (Michie and Chambers in 1960)

Example game: Tic-Tac-Toe



Temporal Difference

Barto, Sutton, Anderson in 1983

Intuition: Prediction of outcomes made at time ***t+1*** is better than the prediction of outcomes made at time ***t***.

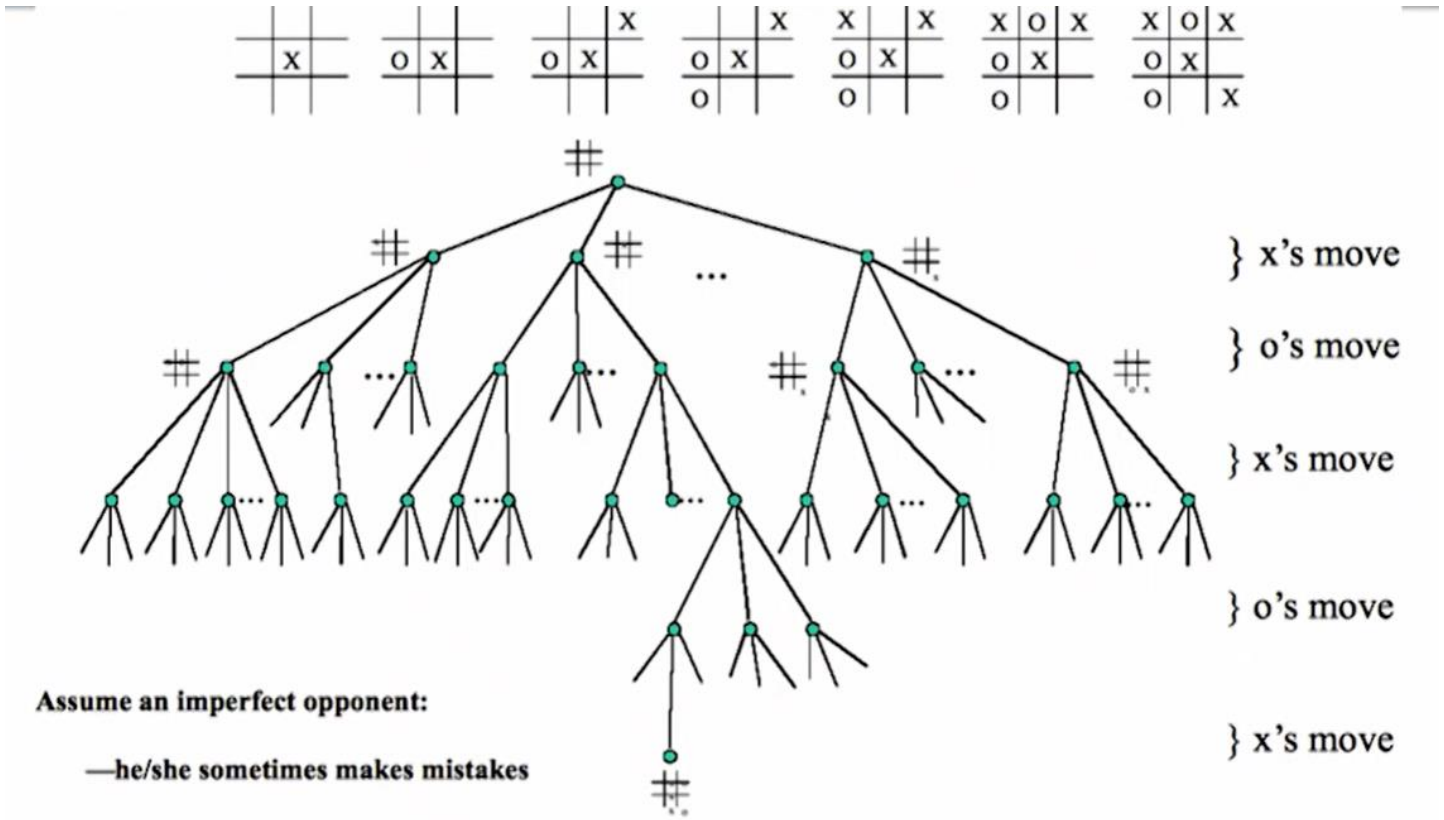
- Hence, the predictions made at later timestep can be used to update the predictions made at earlier timestep.

Created significant impact in behavioral psychology and neuroscience.

TD in Brain (Monkey Experiment).

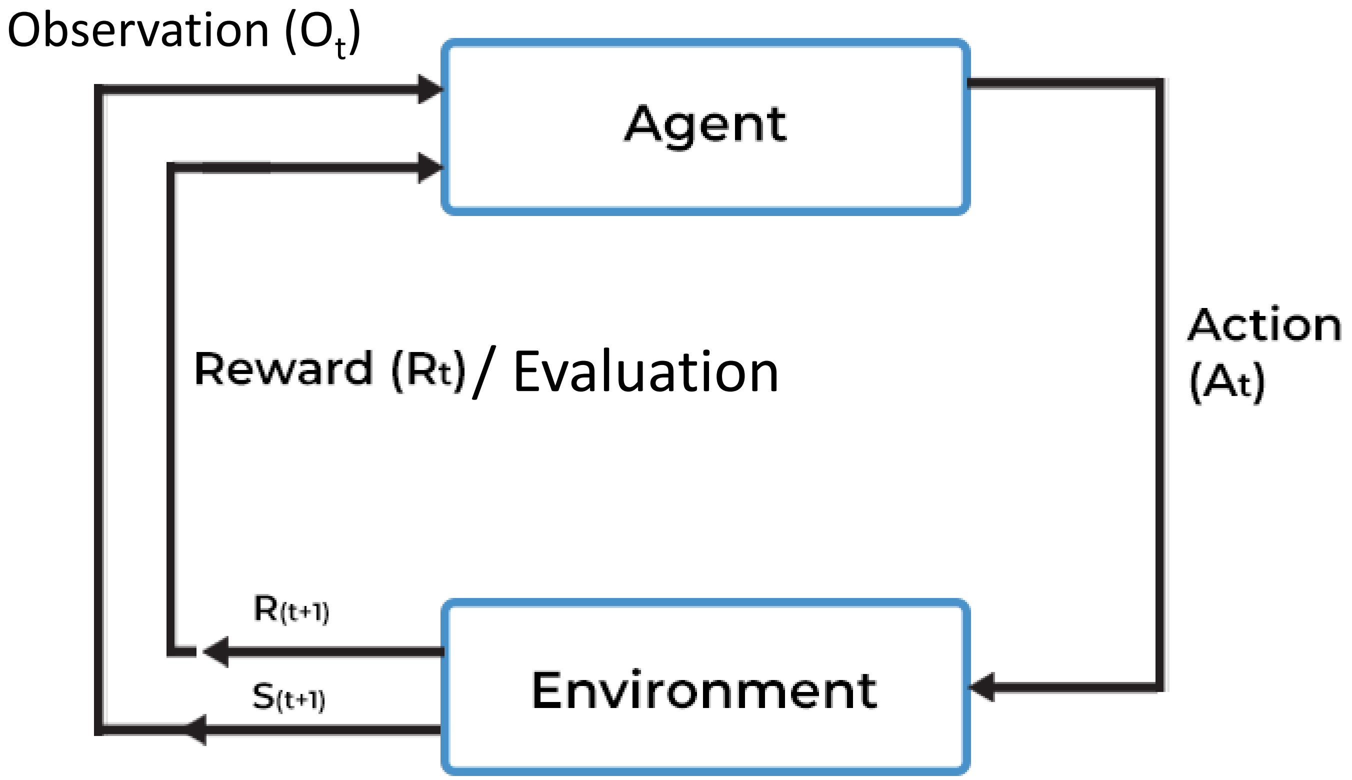
Full RL Problem

Tic-Tac-Toe example:



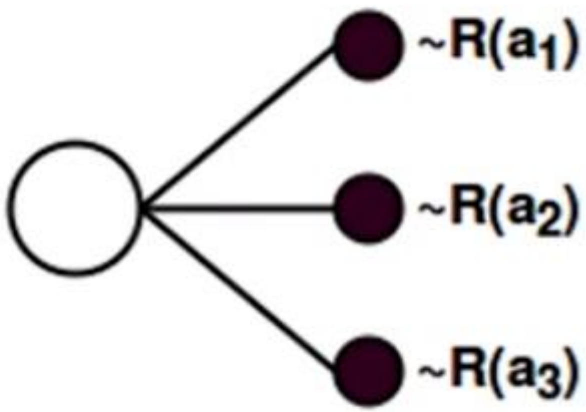
1. Sequence of decisions.
2. Reward is delayed.
3. The second problem in the sequence depends on what action you chose in the first problem.

Full RL Problem

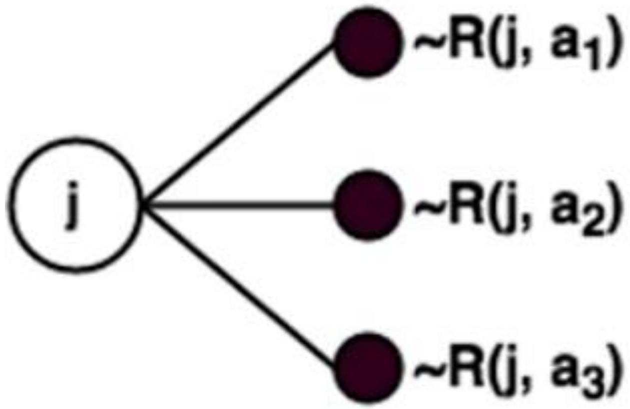


- States
- Environment
- Rewards
- Policy
- Value function
- Model

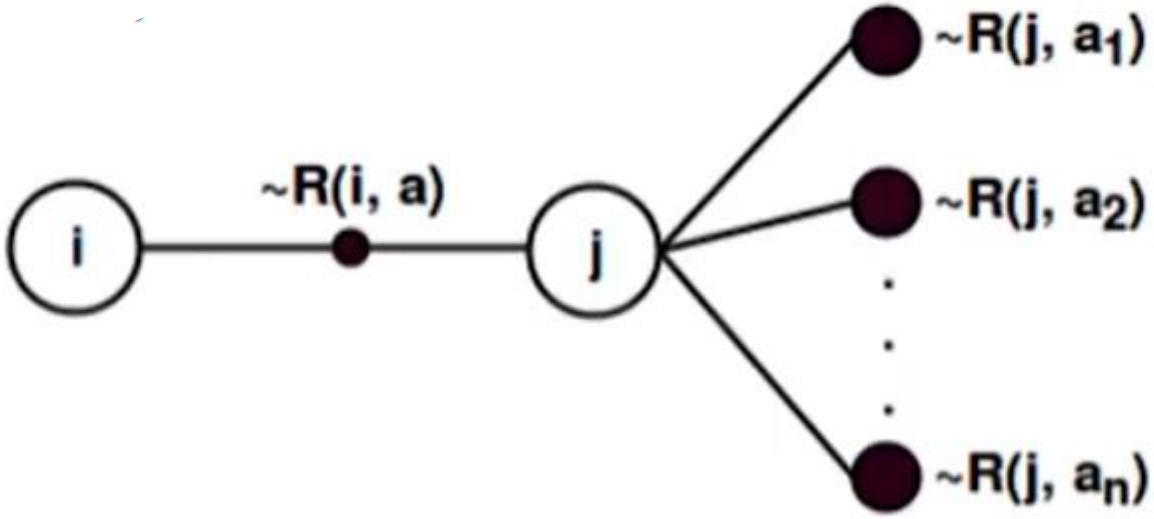
Full RL Problem



Bandit Problem



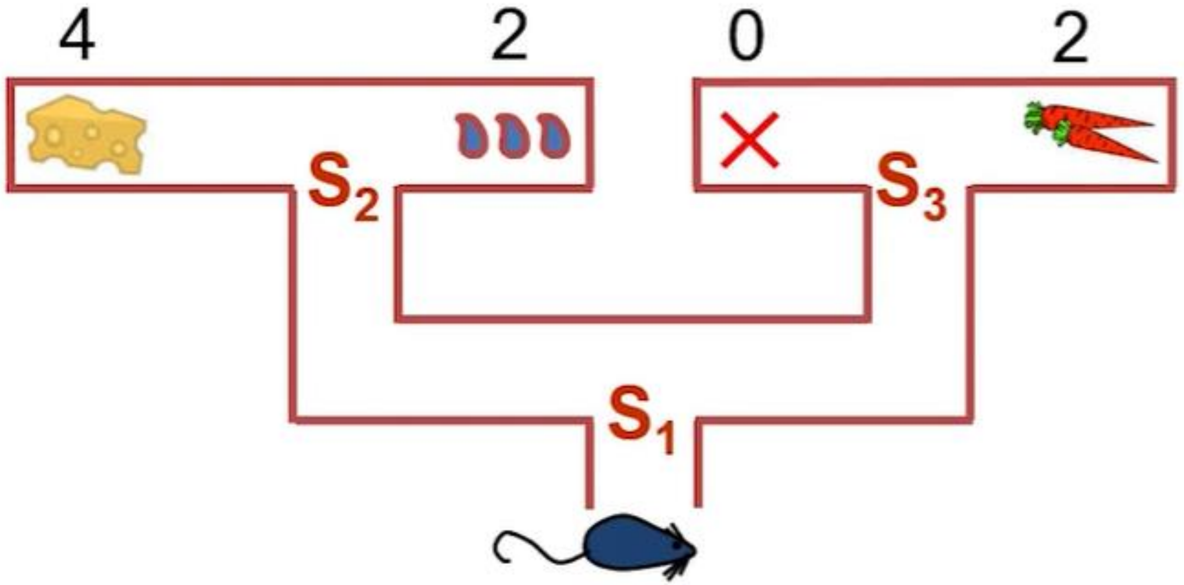
Contextual Bandit



Full RL Problem

Slide from Prof. Ravindran's course: "Reinforcement Learning."

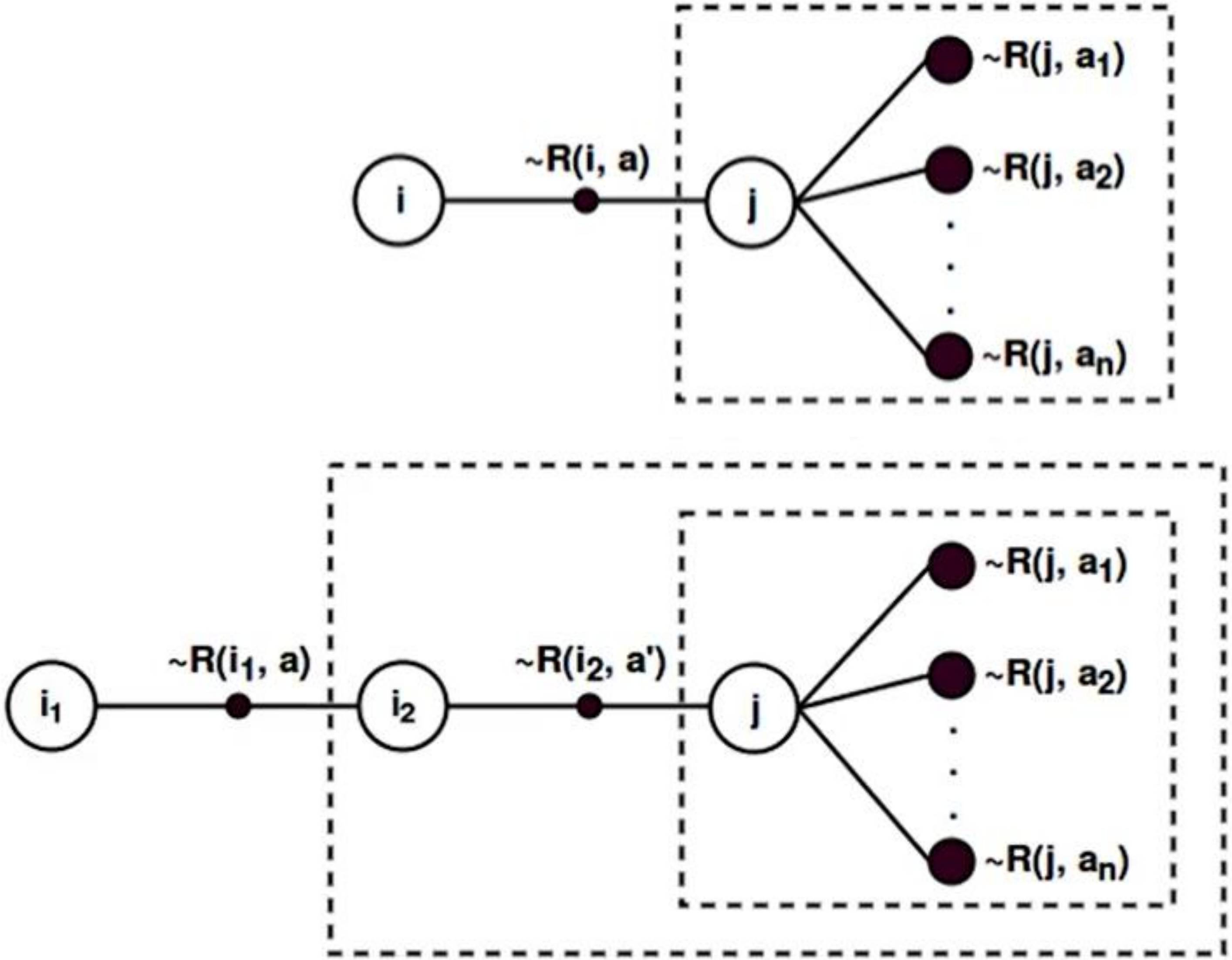
Action at a Temporal Distance



- learning an appropriate action at S_1:
  - –depends on the actions at S_2 and S_3
  - –gains no immediate feedback
- Idea: use prediction as surrogate feedback

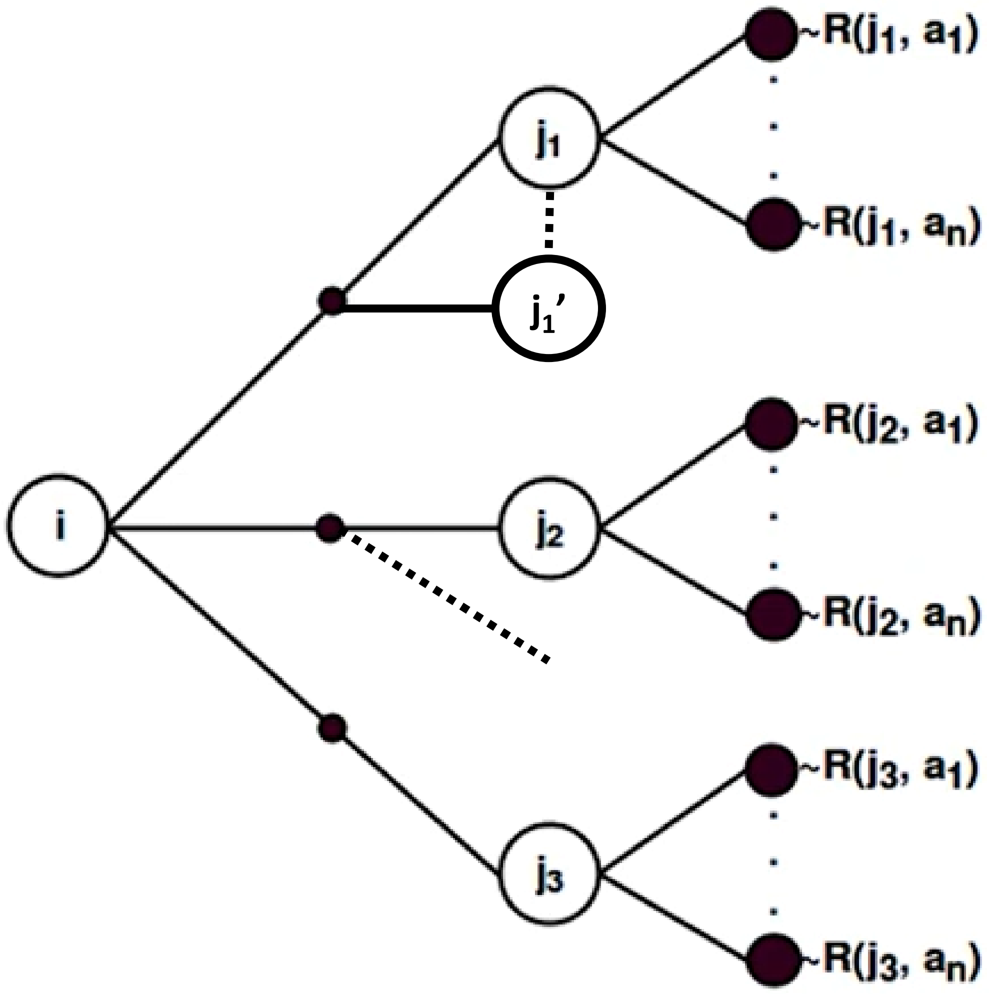
Slide from Prof. Ravindran's course: "Reinforcement Learning."

Full RL Problem



Slide from Prof. Ravindran's course: "Reinforcement Learning."

Full RL Problem



Slide from Prof. Ravindran's course: "Reinforcement Learning."