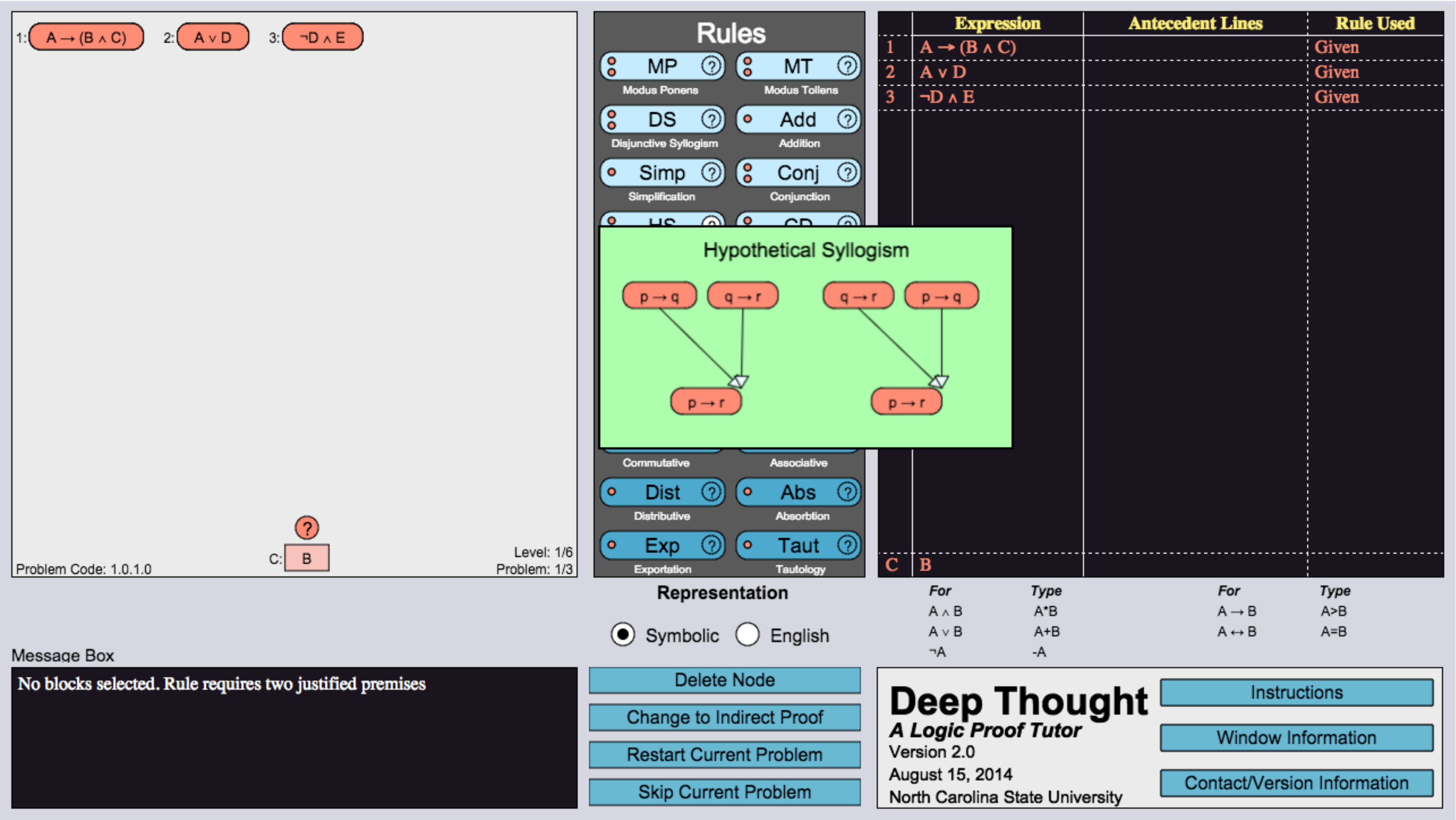
Assigned Project

Dr. Min Chi
Special Thanks to: Shitian Shen
Department of Computer Science
North Carolina State University

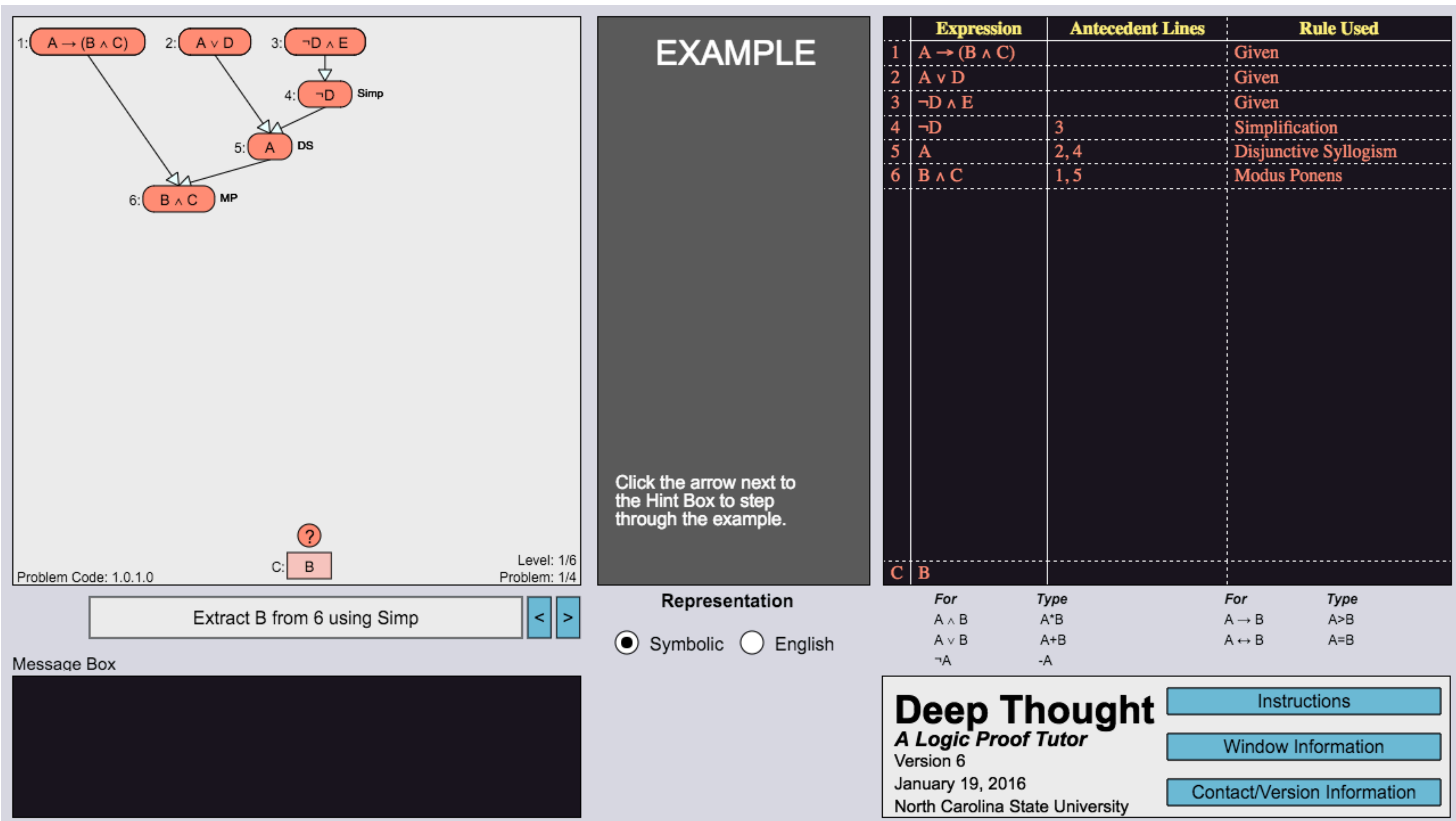
Introduction

- Intelligent Tutoring System contains a set of actions
- Deep Thought (Dr. Barnes, 2015) can take two actions:
  - *Problem Solving* (PS)
  - *Work Example* (WE)

Problem Solving



Work Example



Question

When to assign PS or WE to students ?

Pedagogical strategy is defined as policies to decide what the system action to take next in the face of alternatives.

Induce Pedagogical Strategy

- Inducing pedagogical strategy is challenging
  - Hard code
  - Data driven

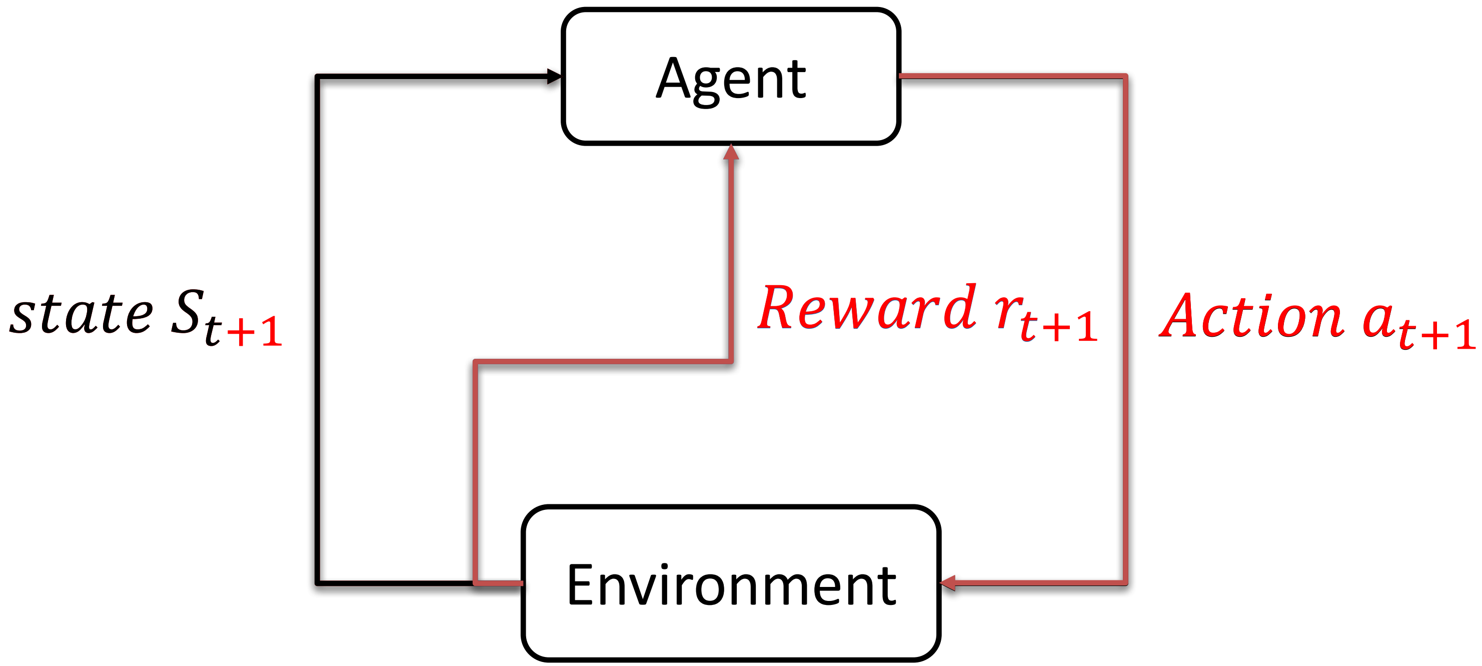
Reinforcement Learning vs. **Inducing Pedagogical strategy**

What is the best action for the agent *(tutor)*
to take in any state *(learning context)*
in order to maximize reward *(student learning)*

Reinforcement Learning:

- Model-based vs Model-free Reinforcement Learning
  - Model-based
    - Generating data is expensive (ITS)
    - Learn from the model instead of data sets
  - Model-free
    - Collecting data is trivial (playing chess)
    - Learn from data sets directly

Agent Environment Interaction



Markov Decision Process: Definition

- A Mathematical framework for representing a reinforcement learning task
- A tuple $< S, A, T, R, \pi >$

State Set	
Action Set	
Transition Probability	
Rewards	
Policy	

Value Iteration: Algorithm

1. $V_0(s) = 0,\ for\ s\ \in$ S

Initialization

2. For k

$\Delta \leftarrow 0$

For each $s\ \in$ S

$v \leftarrow V_{k-1}(s)$

$V_k(s) \leftarrow \max_a \sum_{s'} T^a_{ss'}[R^a_{ss'} + \gamma V_{k-1}(s')]$

$\Delta \leftarrow \max(\Delta, |v - V_k(s)|)$

Until $\Delta \leftarrow \theta$ (a small positive number)

Maximizing Value Function

3. $\pi(s) = \ arg\max_a \sum_{s'} T^a_{ss'}[R^a_{ss'} + \gamma V^\pi(s')]$

Policy Generation

Value Iteration: Example

- Transfer Data into trajectories
  - State set : $\{S_1, S_2\}$
  - Action set: $\{PS,\ WE\}$

$$S_1 \xrightarrow{PS,0} S_2 \xrightarrow{PS,0} S_2 \xrightarrow{WE,50} S_1 \xrightarrow{PS,0} \ldots \xrightarrow{WE,0} S_2 \xrightarrow{PS,100} S_1 \xrightarrow{WE,0}$$
$$S_2 \xrightarrow{PS,0} S_2 \xrightarrow{WE,50} S_1 \xrightarrow{PS,0} \ldots \xrightarrow{WE,0} S_2 \xrightarrow{PS,0} S_1 \xrightarrow{WE,-50} S_1 \xrightarrow{WE,0}$$
$$S_2 \xrightarrow{PS,100} S_1 \xrightarrow{WE,0} S_2 \xrightarrow{PS,100} \ldots \xrightarrow{WE,0} S_1 \xrightarrow{PS,0} S_2 \xrightarrow{WE,0} S_2 \xrightarrow{WE,0} T$$

Value Iteration: Example

- Transition Probability

$$P(S_1|S_2, PS) = \frac{\#(S_2 \xrightarrow{PS} S_1)}{\#(S_2 \xrightarrow{PS} S_1) + \#(S_2 \xrightarrow{PS} S_2)} = \frac{1}{4}$$

- Expected Rewards

$$R(S_1|S_2, PS) = \frac{\sum r(S_2 \xrightarrow{PS} S_1)}{\#(S_2 \xrightarrow{PS} S_1)} = 20$$

Value Iteration: Example

- Transition probability $T_{ss'}^{a}$

PS

1/4	3/4
1/2	1/2

WE

1/2	1/2
2/3	1/3

- Reward function $R_{ss'}^{a}$

PS

10	40
20	30

WE

20	30
45	5

Value Iteration: Example

K				
0	0		0	
1	32.50	PS	31.67	WE
2	61.18	PS	60.67	WE
3	87.22	PS	86.58	WE
4	110.56	PS	109.97	WE
121	320.90	PS	320.30	WE
122	320.90	PS	320.30	WE

$$V_1(S_1) = max \begin{cases} \frac{1}{4}(10 + 0.9 * 0) + \frac{3}{4}(40 + 0.9 * 0) = 32.50 \quad PS \\ \ldots \end{cases}$$

$$V_1(S_2) = max \begin{cases} \frac{1}{2}(20 + 0.9 * 0) + \frac{1}{2}(30 + 0.9 * 0) = 25 \quad PS \\ \frac{2}{3}(45 + 0.9 * 0) + \frac{1}{3}(5 + 0.9 * 0) = 31.67 \quad WE \end{cases}$$

$$V_2(S_1) = max \begin{cases} \frac{1}{4}(10 + 0.9 * 32.5) + \frac{3}{4}(40 + 0.9 * 31.67) = 61.18 \quad PS \\ \ldots \end{cases}$$

$$V_2(S_2) = max \begin{cases} \frac{1}{2}(20 + 0.9 * 32.5) + \frac{1}{2}(30 + 0.9 * 31.67) = 53.87 \quad PS \\ \frac{2}{3}(45 + 0.9 * 32.5) + \frac{1}{3}(5 + 0.9 * 31.67) = 60.67 \quad WE \end{cases}$$

Optimal policy π^*:

$$S_1 \rightarrow PS$$
$$S_2 \rightarrow WE$$

Policy Evaluation

- Expected Cumulative Reward (Tetreault, 2006)

$$ECR = \sum_{i=1}^{m} \frac{N_i}{N_1 + N_2 + \cdots + N_m} \times V^{\pi}(S_i)$$

Where S_i is the starting state, N_i is the times that S_i exists as starting state

- The higher ECR of the policy means the better policy

The Challenge is:

What is the best action for the **agent** ***(tutor)***
to take in any **state** ***(learning context)***
in order to maximize **reward** ***(student learning)***

Challenge: State Representation

How to design states representing environment ?

State Representation: Feature Selection for RL

- Three types of feature selection methods
  - Filtered approach
    - Feature Selection process is independent to model construction
    - Evaluating the independence between reward with feature (Hirotaka, Masashi 2010)
  - Wrapper approach
    - Feature subsets are evaluated by predefined score function
    - Monte Carlo tree search algorithm (Gaudel 2010)
  - Embedded approach
    - Feature selection and model construction are executed simultaneously
    - Least Square Temporal Difference with lasso regularized item (Kolter 2009)

Previous research: Correlation-based Methods: High vs Low

- When selecting features, should we select the feature that is most correlated (High) or uncorrelated (Low) to current optimal feature set ?
- In Supervised Learning, features with high correlation with labels are selected (C. Lee, 2010; L Yu & H Liu, 2003)
- In RL, the answer is not straightforward

Research Question: Low vs. High

- Choosing most correlated features (High)
  - Most likely to be related to decision making
  - May not make more contribute than current optimal feature set
- Choosing most uncorrelated features (Low)
  - Raise the diversity of feature set
  - Take the risk of involving irrelevant or noisy features

Correlation Metrics

Given labeled data, we can compute some simple score S(i) that measures how informative each feature X is about the class labels Y.

- Chi-square (CHI) (Zibran, 2007)

$$\chi^2 = \sum_i \frac{(X_i - Y_i)^2}{Y_i}$$

- Information Gain (IG) (C. Lee, 2010)

$$IG(X,Y) = H(Y) - H(X|Y)$$

Correlation Metrics

- Information Gain Ratio (IGR) (J. T. Kent, 1983)

$$IGR(X,Y) = \frac{H(X) - H(X|Y)}{H(Y)}$$

- Symmetric Uncertainty (SU) (L. Yu, H. Liu, 2003)

$$SU(X,Y) = \frac{H(X) - H(X|Y)}{H(X) + H(Y)}$$

- Weighted Information Gain (WIG) (We proposed)

$$WIG(X,Y) = \frac{H(X) - H(X|Y)}{(H(X) + H(Y))H(Y)}$$

Correlation-based Feature Selection Methods

- Feature Selection for model-based RL
- Apply correlation between current optimal feature set and potential feature as the feature selection criteria
- Forward feature selection strategy

10 Correlation-based Methods

- Explore both high and low correlation
- Obtain 10 correlation-based feature selection methods (5 correlation metrics $\times$ 2 correlation types)

	High	**Low**
CHI	CHI-High	CHI-Low
IG	IG-High	IG-Low
IGR	IGR-High	IGR-Low
SU	SU-High	SU-Low
WIG	WIG-High	WIG-Low

Other Implemented Methods

- Ensemble Methods
  - 10 correlation-based methods
  - 4 RL based methods
- RLPreviousFS
  - 4 RL based methods
  - 2 PCA based methods
  - 4 PCA & RL based methods

Intelligent Tutoring System

- Deep Thought (Dr. Barnes, 2015)
  - A rule-based tutoring system for teaching logic proof problems
  - Student solves 1-3 problems per level (Total 6 levels)
  - Level score ($LevelScore_i, i \in [1,6]$) is given for each student based on his/her performance on the last problem in the level i

Deep Thought : Reward Function

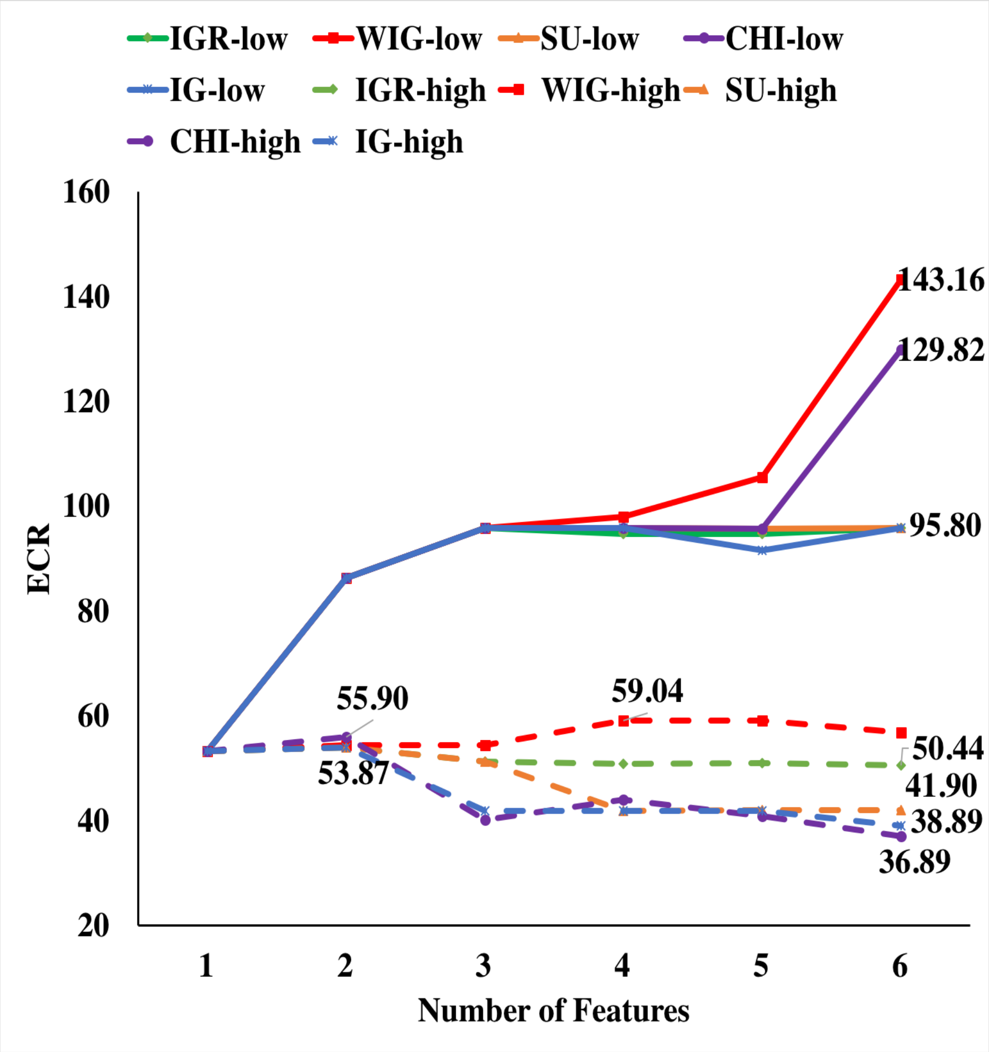
- Immediate Reward
  - $R_1 = LevelScore_1$
  - $R_i = LevelScore_i - LevelScore_{i-1},\ i \in [2,6]$
- Delayed Reward

$$R_{delay} = LevelScore_6 - LevelScore_1$$

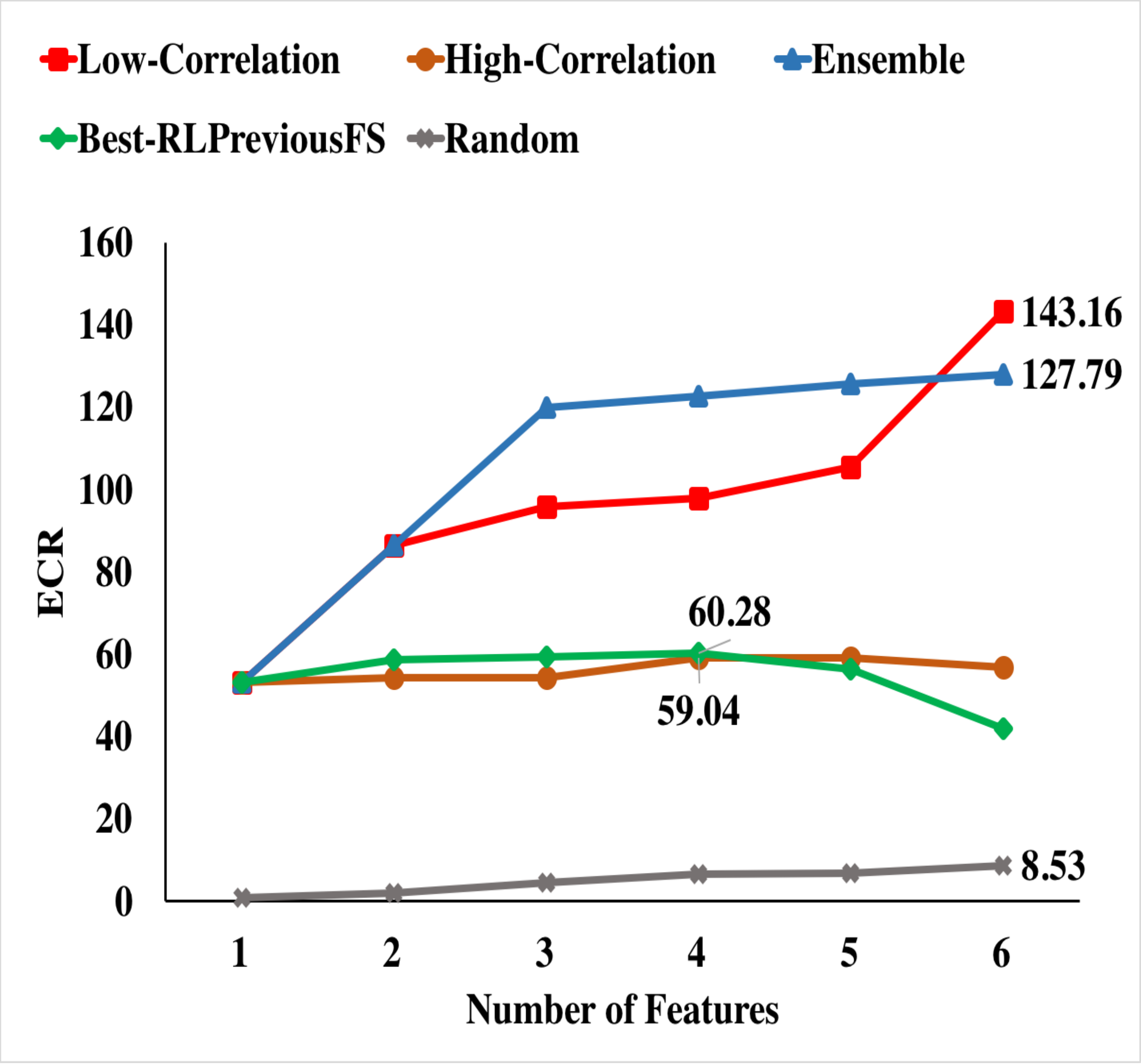
Deep Thought Data Sets

- Total 303 students in Fall 2014 and Spring 2015
- Average time spend in tutor is 416.60 minutes
- Total 135 features
- Action set
  - should it ask student to solve the next problem (PS)
  - should it provide an example to show the student how to solve the next problem (WE)

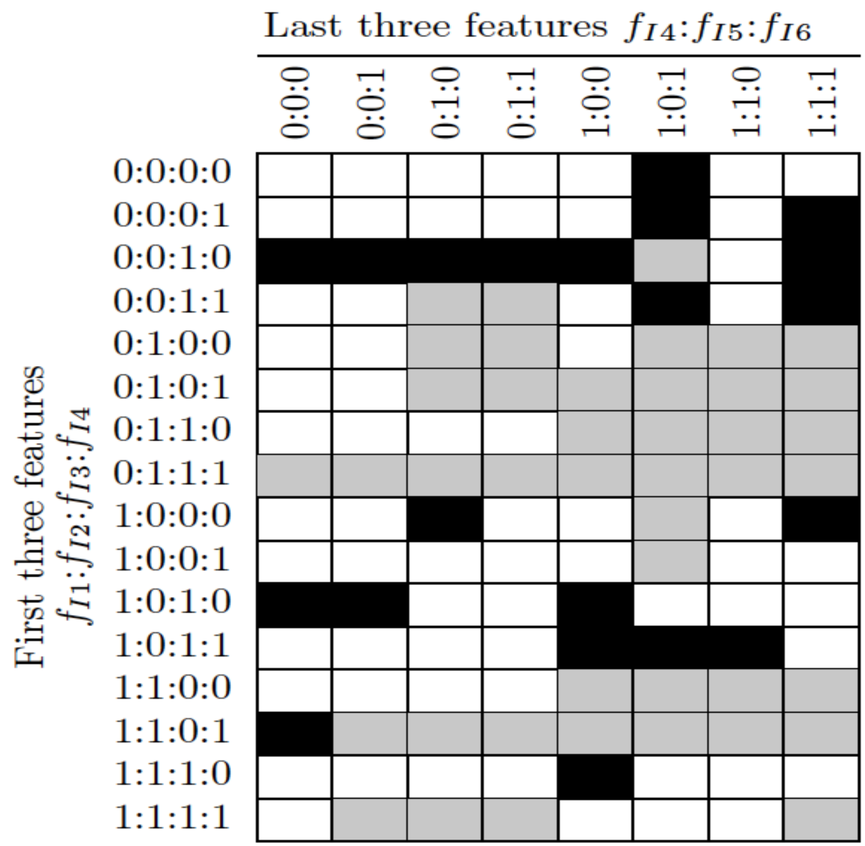
Result: High vs Low correlation



Results: Overall Evaluation



Induced Pedagogical Strategy

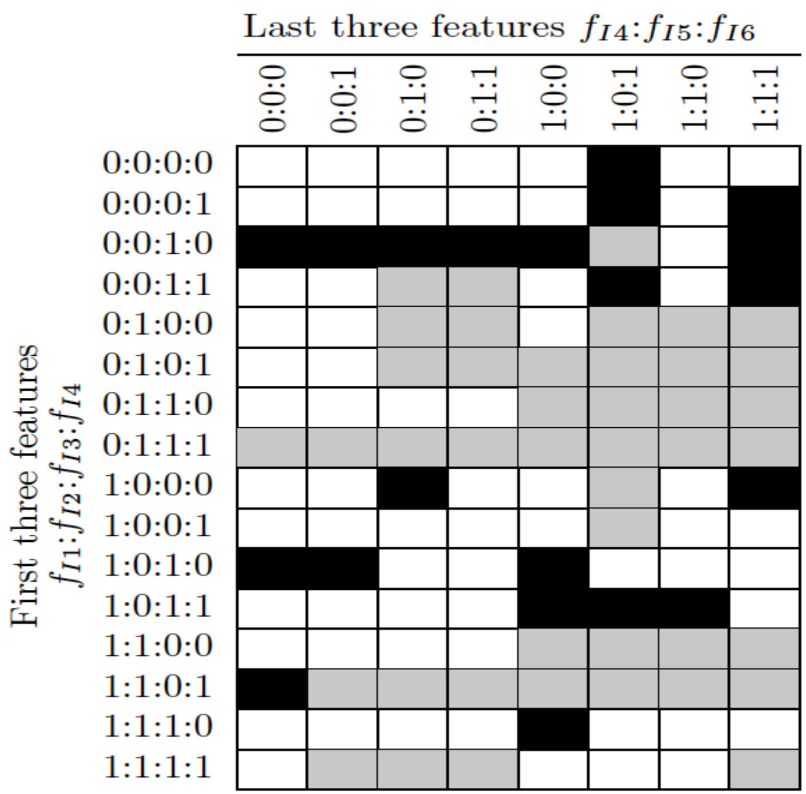


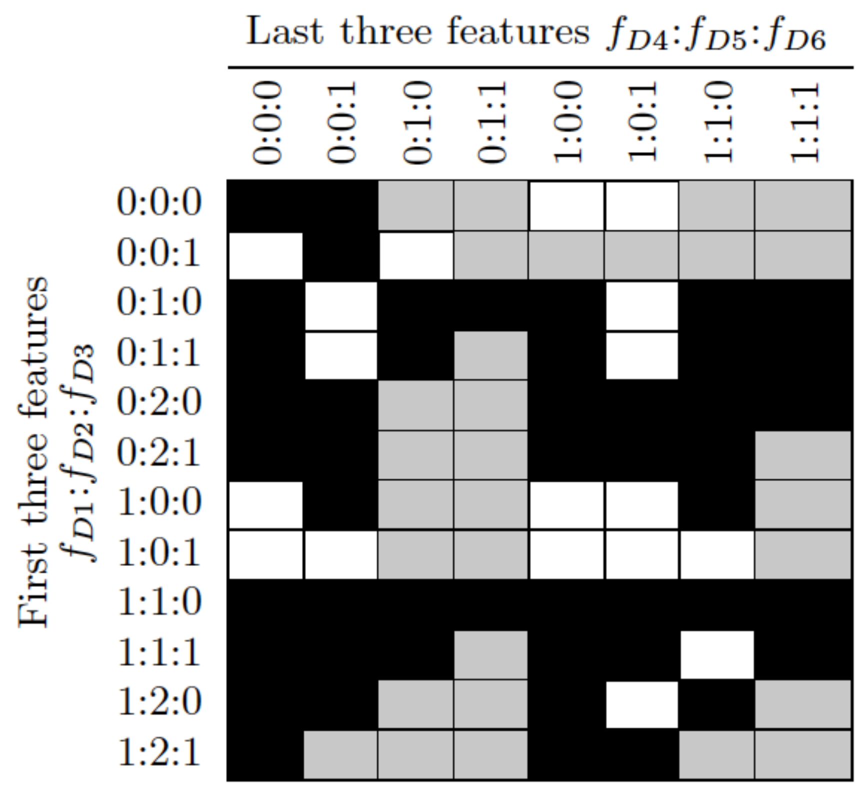
64 rules associated with WE (White)

21 rules associated with PS (Black)

43 no rules (Gray)

Induced Pedagogical Strategy





The best Policy

64 rules associated with WE (White)

21 rules associated with PS (Black)

43 no rules (Gray)

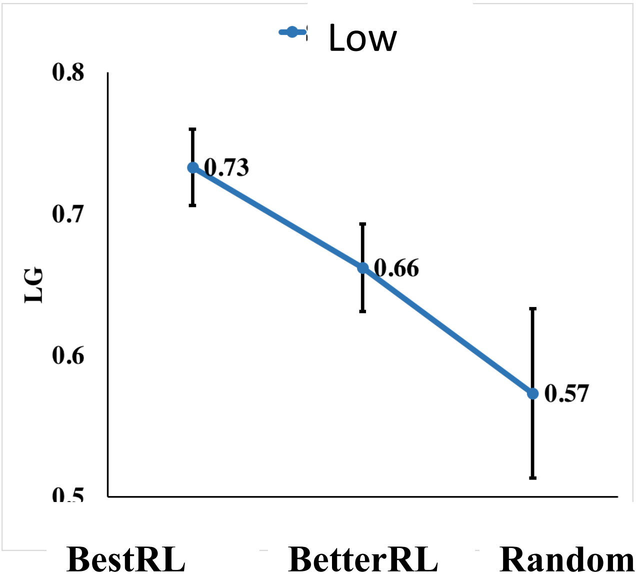
Another Policy

18 rules associated with WE

48 rules associated with PS

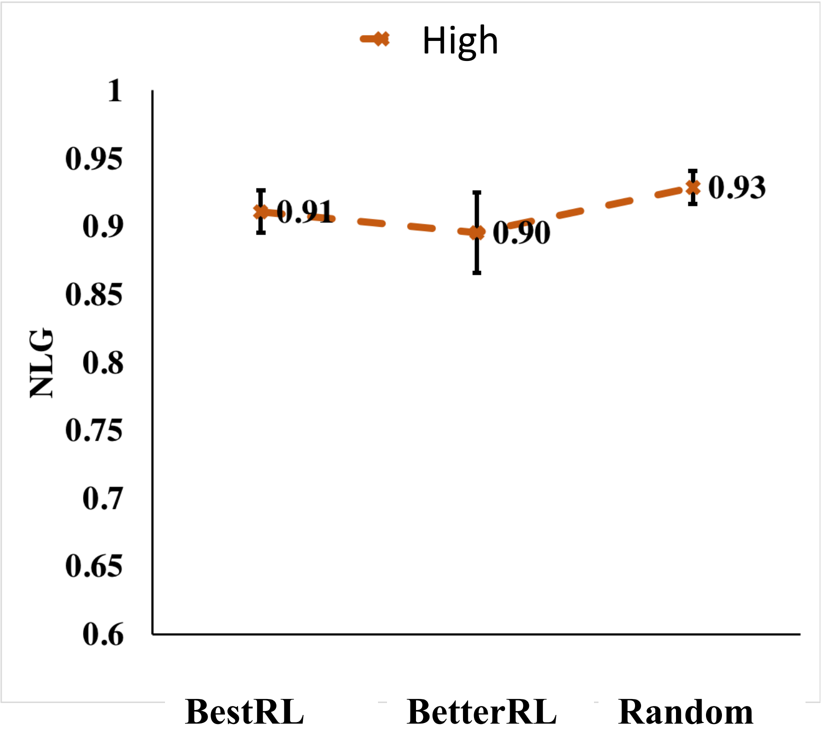
30 no rules

Learning Performance Result



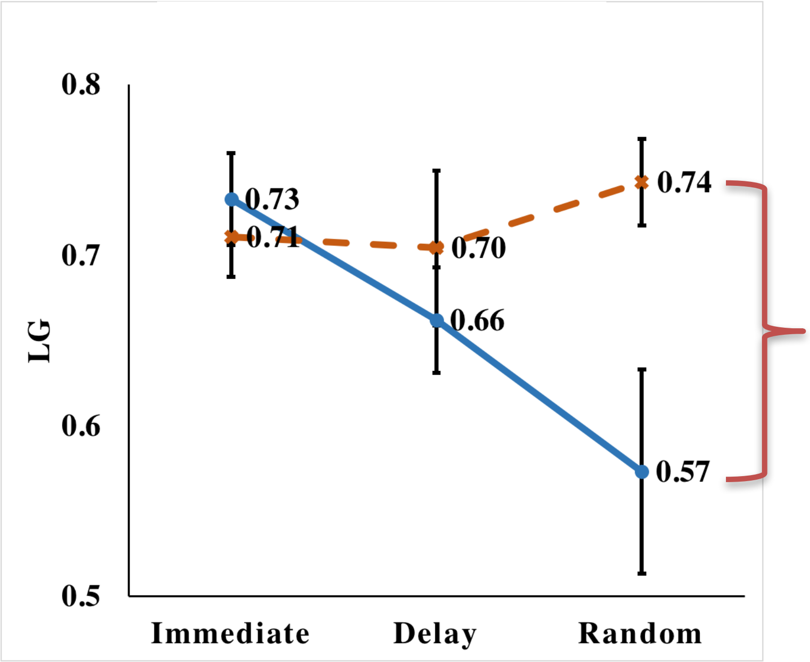
- Significant difference among three Low groups
  - $F(2,46) = 3.99,\ p = 0.025$
- BestRLPolicy-Low group significant outperforms BetterRLPolicy-Low
  - $t(27) = 2.69,\ p = 0.012$
- BestRLPolicy-Low group marginally outperforms BetterRLPolicy-Low
  - $t(35) = 1.67,\ p = 0.098$

Learning Performance Result



- No significant difference between three High groups

Learning Performance Result



Your Task: State represenation

- Discretization the features
- Feature Extraction and/or Feature selection **(explore new methods)**
- No more than 8 features (new or selected features)

- Evaluation: ECR

- Rank all the project: [80-100] * 0.1 points.
- Presentation: 5 points
- Submit your code and we will run it.

Publications

- Shitian Shen, M Chi, "*Reinforcement Learning: the Sooner the Better or the Later the Better ?*", The 24th ACM User Modeling, Adaptation and Personalization (ACM UMAP), 2016 (Full paper)
- Shitian Shen, M Chi, "*Aim Low: Correlation-based Feature Selection for Model-based Reinforcement Learning*", 9th International Conference on Educational Data Mining (EDM), 2016 (Short paper)
- Shitian Shen, M Chi, "*An Analysis of Feature Selection and Reward Function for Model-Based Reinforcement Learning*", 13th International Conference on Intelligent Tutoring System (ITS), 2016 (Poster)