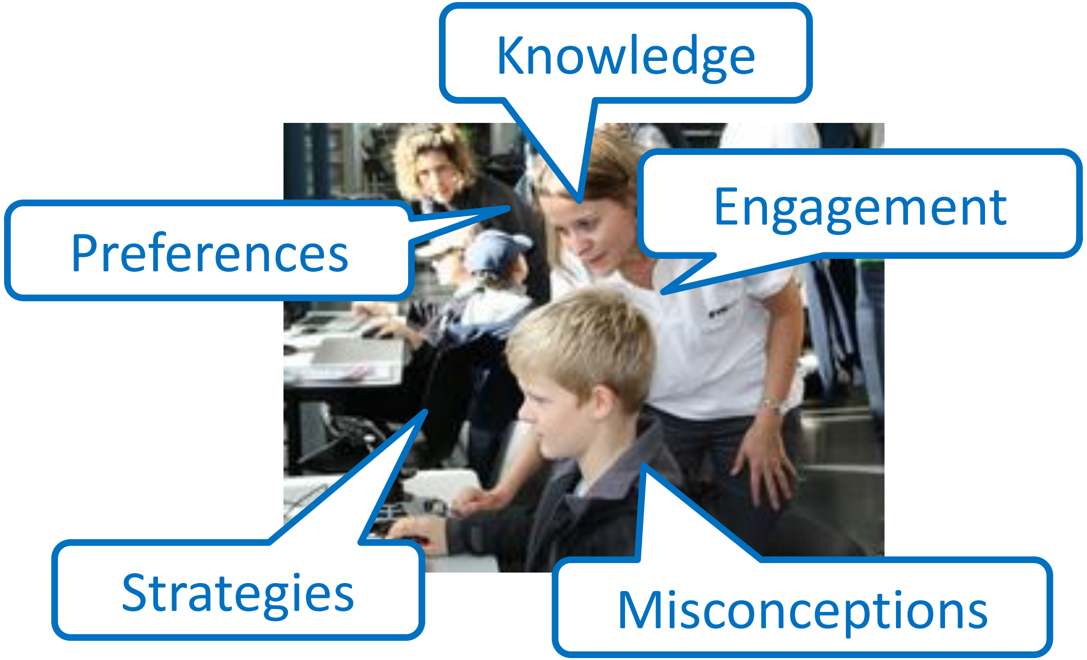
Modeling and Individualizing Learning in Computer-Based Environments

Tanja Käser

January 2020

EPFL SDSC

Human teachers individualize learning



Student models enable individualization

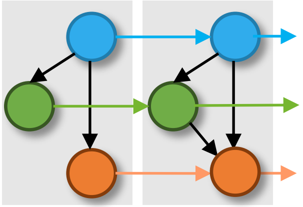
Interaction

- Key stroke
- Mouse Click
- Speech
- Video

Student models enable individualization

Interaction

- Key stroke
- Mouse Click
- Speech
- Video



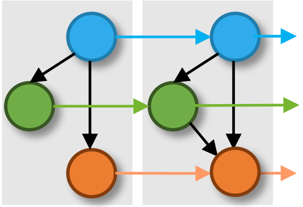
Model

- Detection
- Representation
- Prediction

Student models enable individualization

Interaction

- Key stroke
- Mouse Click
- Speech
- Video



Model

- Detection
- Representation
- Prediction

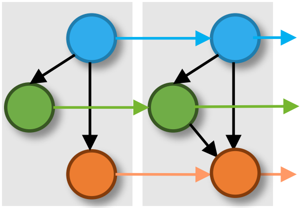
Individualization

- Select new task
- Give feedback
- Provide hint

Student models enable individualization

Interaction

- Key stroke
- Mouse Click
- Speech
- Video



Model

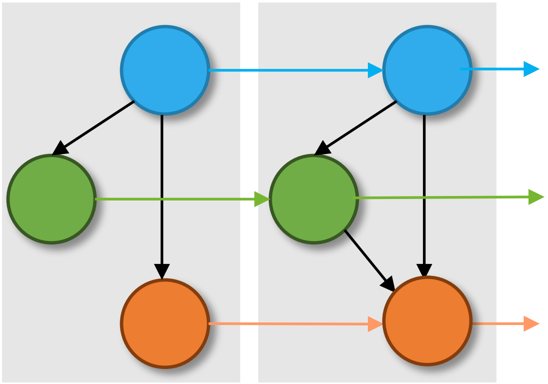
- Detection
- Representation
- Prediction

Individualization

- Select new task
- Give feedback
- Provide hint

Modeling and Individualizing Learning in Computer-Based Environments

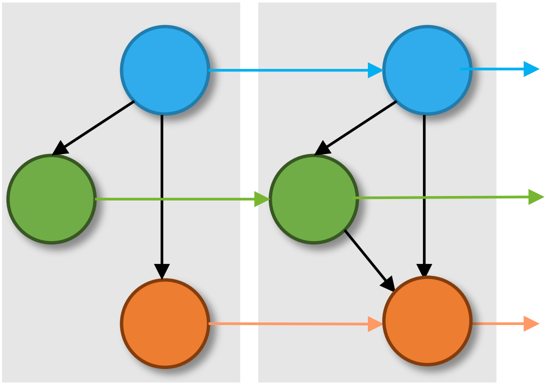
Modeling and predicting knowledge



What?

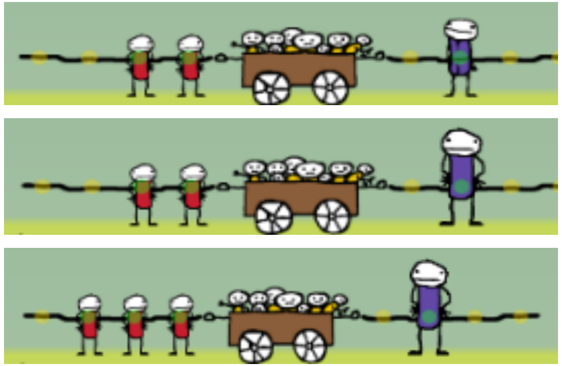
Modeling and Individualizing Learning in Computer-Based Environments

Modeling and predicting knowledge



What?

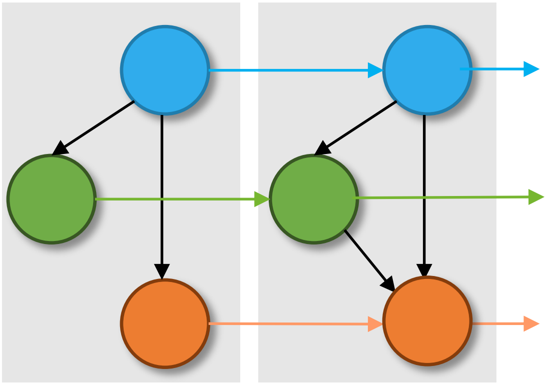
Detecting learner choices and strategies



How?

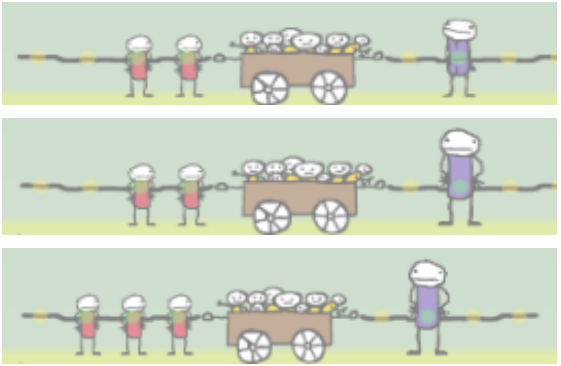
Modeling and Individualizing Learning in Computer-Based Environments

Modeling and predicting knowledge



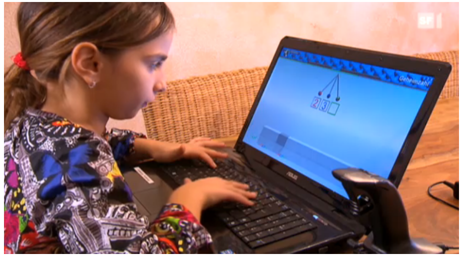
What?

Detecting learner choices and strategies



How?

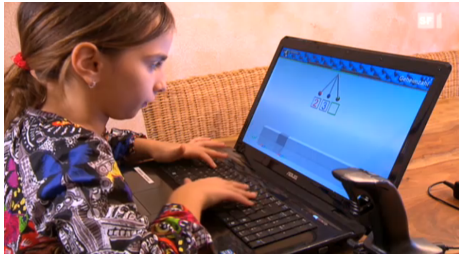
Inferring knowledge based on student answers



Subtraction 0-10

1	2		…		n
0	0	1	0	1	1

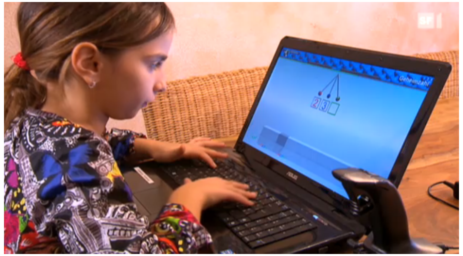
Inferring knowledge based on student answers



Subtraction 0-10

1	2		...		n	n+1
0	0	1	0	1	1	?

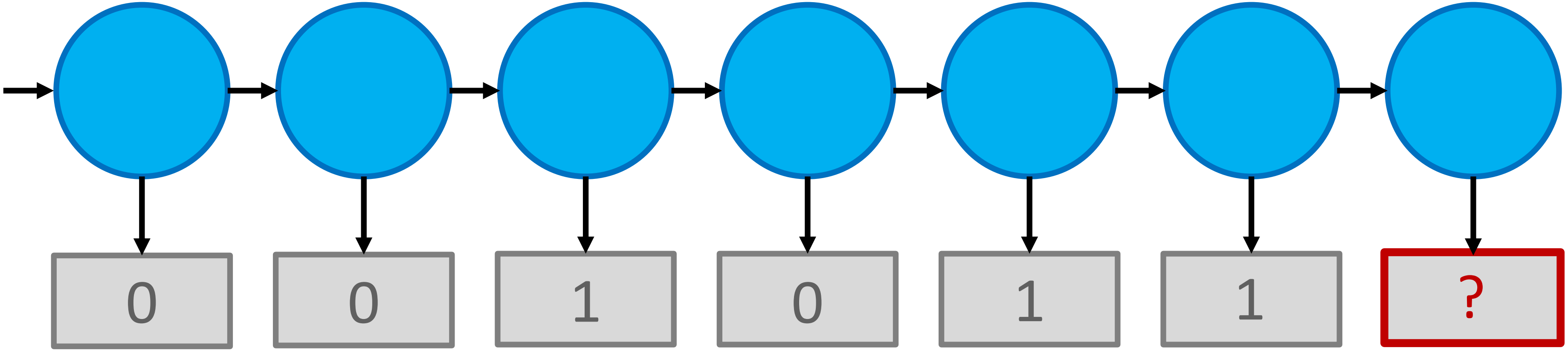
Bayesian Knowledge Tracing (BKT)



Subtraction 0-10

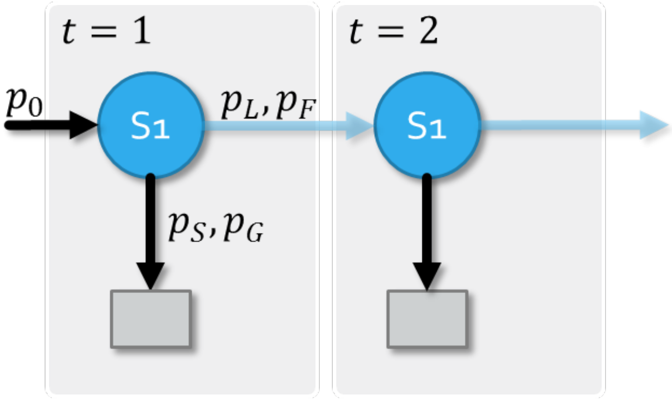
Latent variable

Observed variable

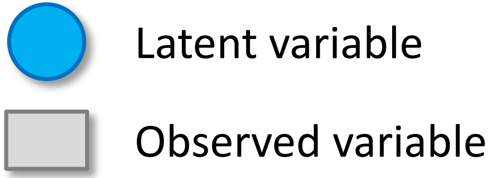


BKT models are simple, efficient, and interpretable

Bayesian Knowledge Tracing (BKT)

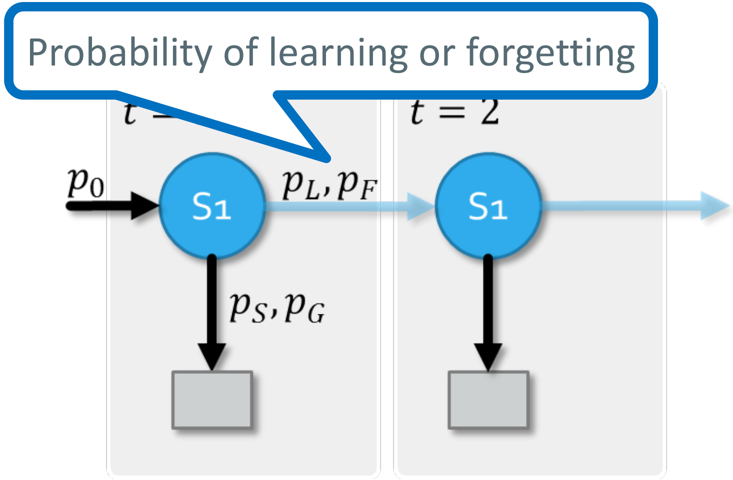


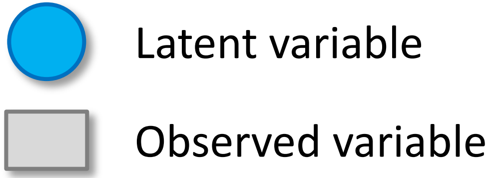
$$\theta = \{p_0, p_L, p_F, p_S, p_G\}$$



BKT models are simple, efficient, and interpretable

Bayesian Knowledge Tracing (BKT)

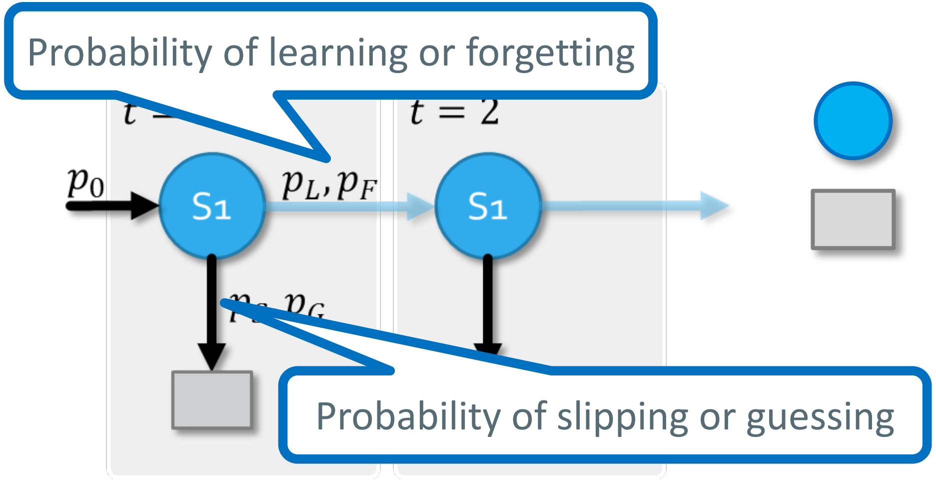




$$\theta = \{p_0, p_L, p_F, p_S, p_G\}$$

BKT models are simple, efficient, and interpretable

Bayesian Knowledge Tracing (BKT)



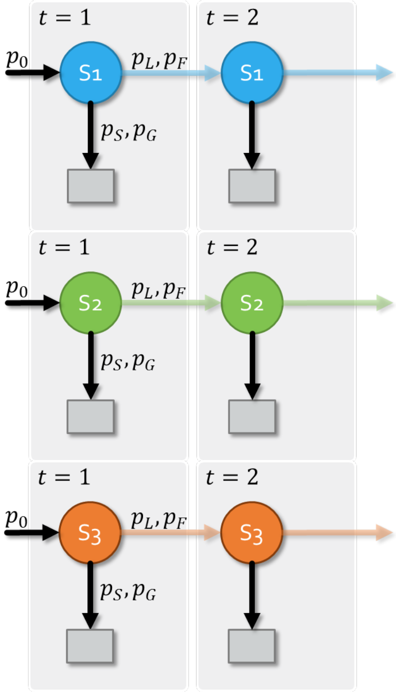
Latent variable

Observed variable

$$\theta = \{p_0, p_L, p_F, p_S, p_G\}$$

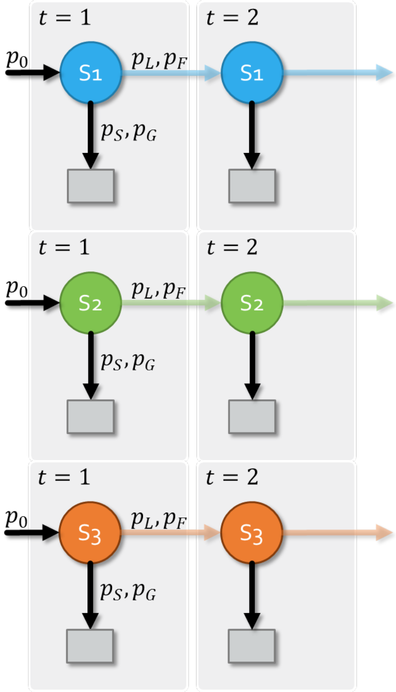
… but they have limited representational power

Bayesian Knowledge Tracing (BKT)

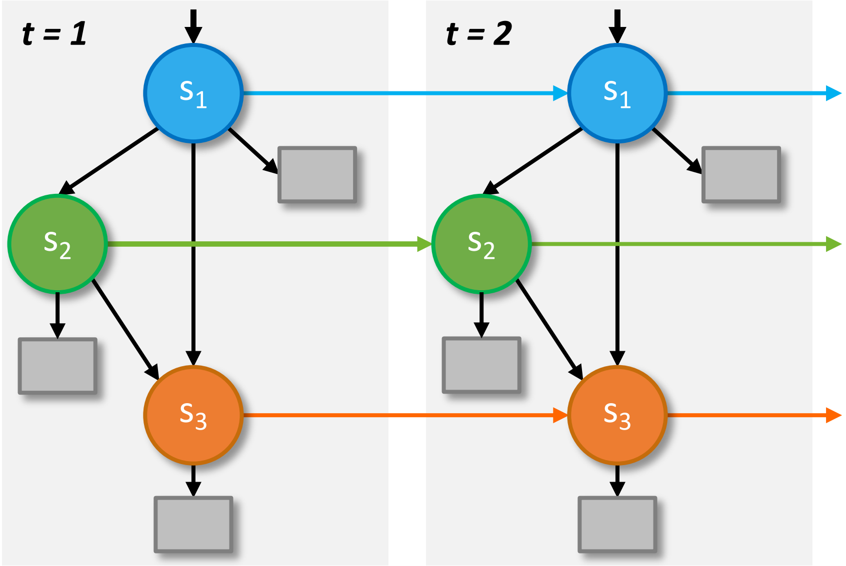


DBNs can model interactions between variables

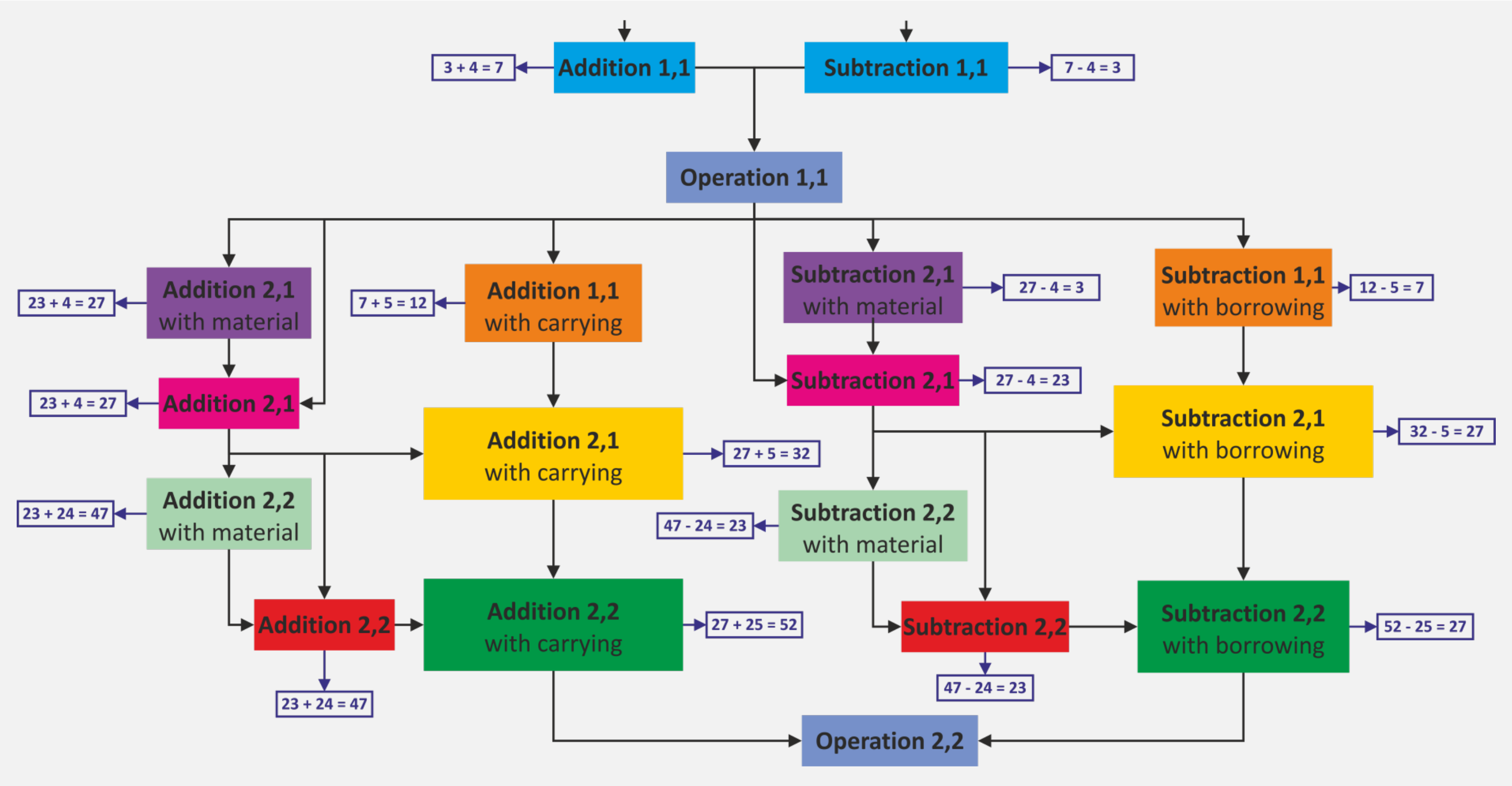
Bayesian Knowledge Tracing (BKT)



Dynamic Bayesian Networks (DBN)



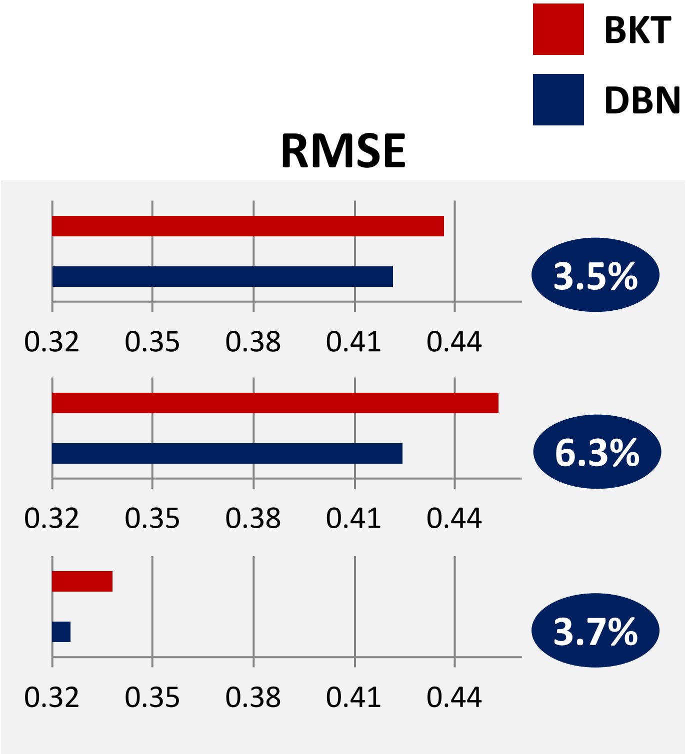
Example: DBN representing mathematical skills



[Käser et al., Frontiers 2013; Käser et al., AISTATS 2014]

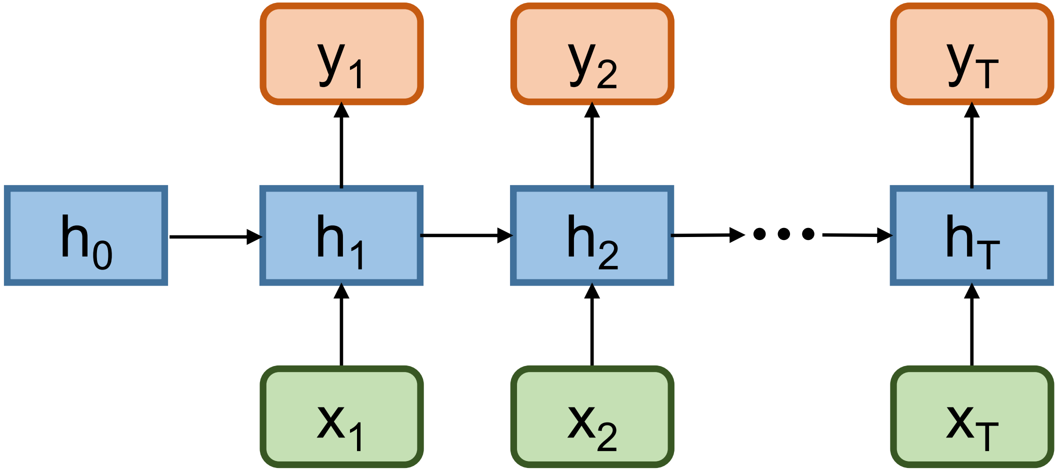
DBNs outperform BKT in different learning domains

Learning Domain	Students	Observations
Subtraction	1581	158’100
Physics	77	38’500
Algebra	6043	3’021’500



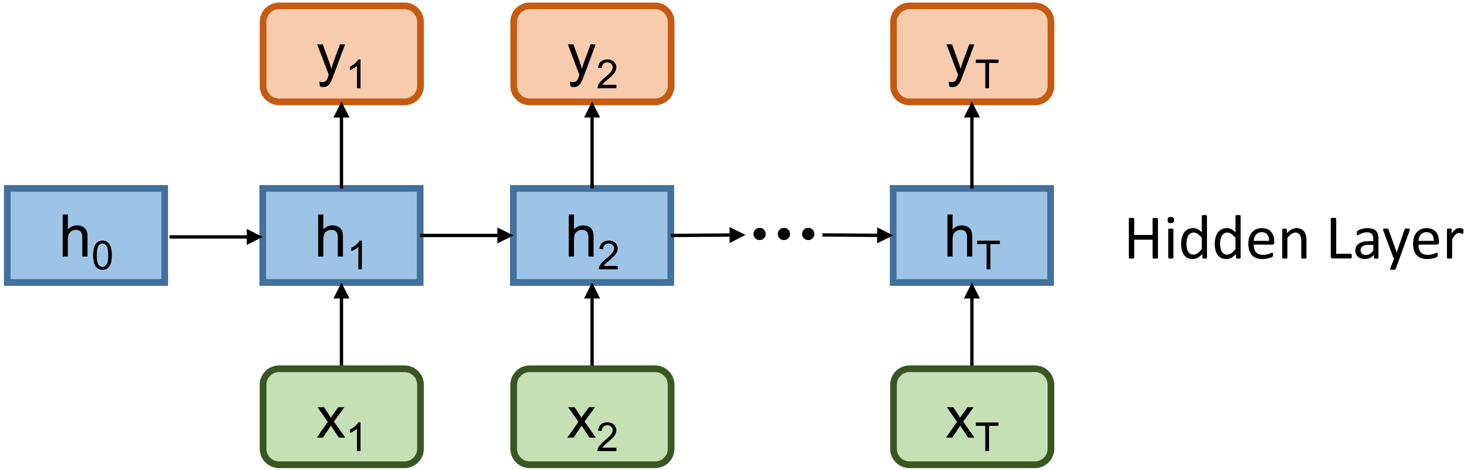
[Käser et al., ITS 2014; Käser et al., IEEE TLT 2017]

Deep Knowledge Tracing



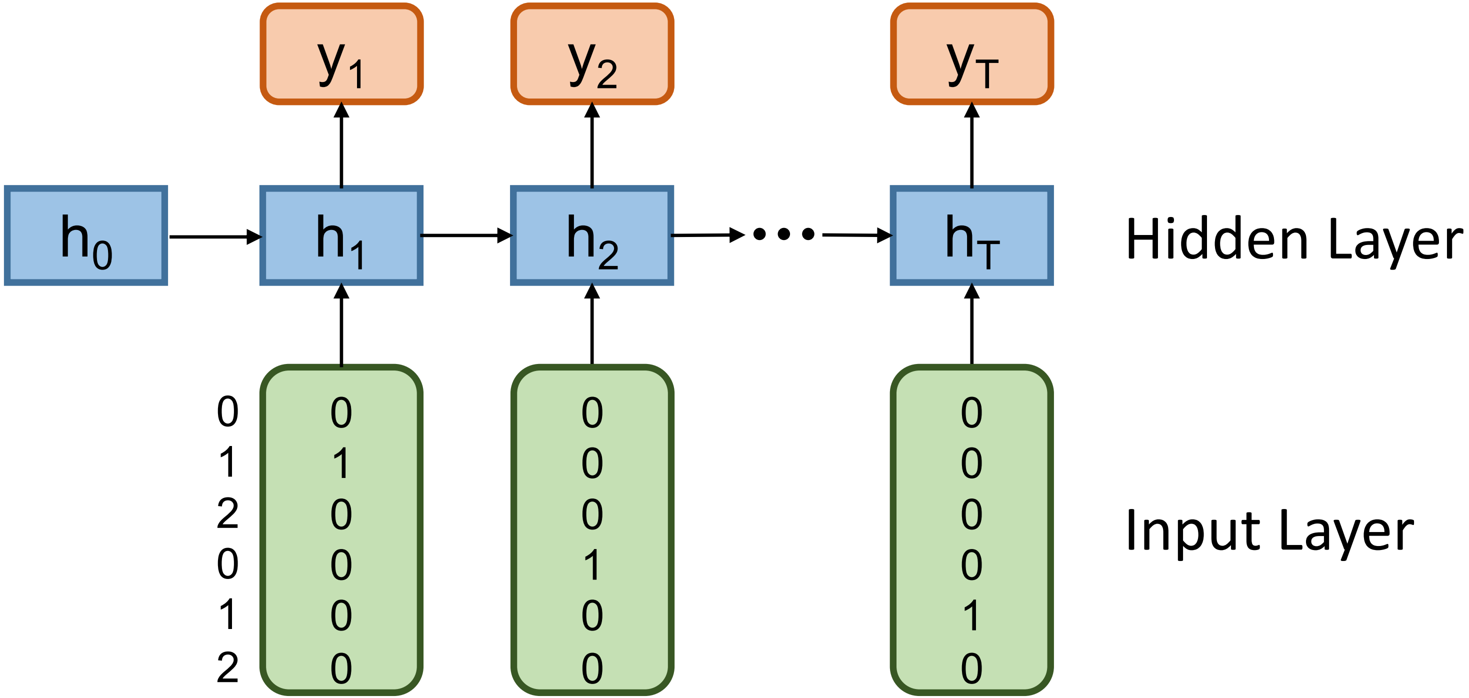
[Piech et al., NIPS 2015]

Hidden layer captures relevant information



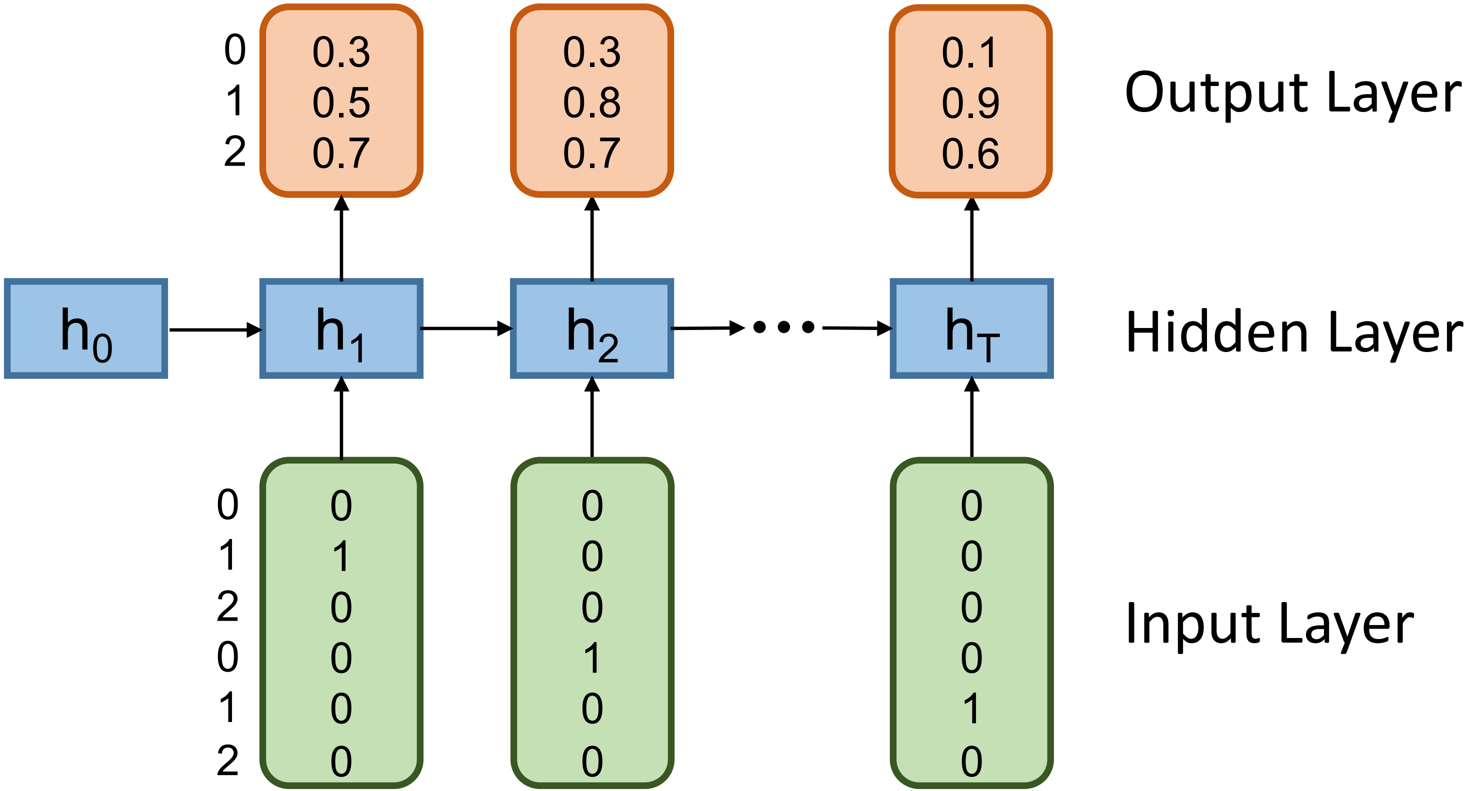
[Piech et al., NIPS 2015]

Input layer represents observations



[Piech et al., NIPS 2015]

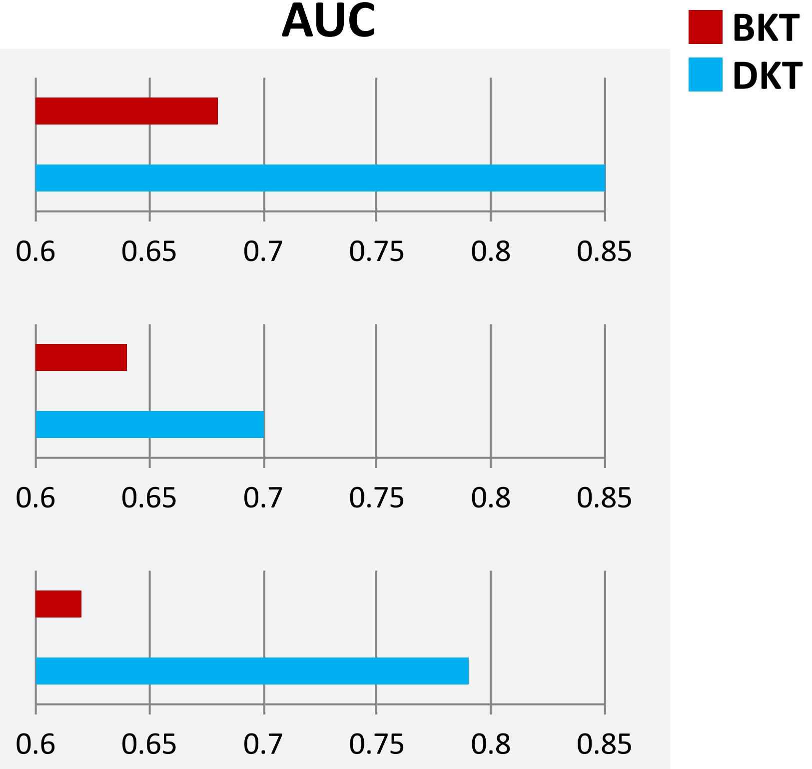
Output layer consists of predicted probabilities



Deep Knowledge Tracing outperforms BKT

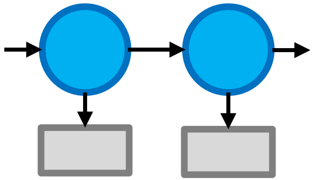
Data Set	Students	Observations
Khan Academy (Math)	47’500	1’435’000
Assistments (Math)	19’457	707’944
KDD Cup 2010 (Algebra)	574	607’026

AUC



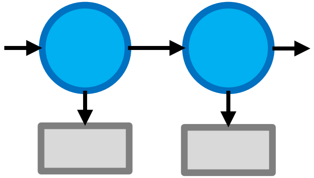
[Piech et al., NIPS 2015; Xiong et al., EDM 2016]

Modeling and Predicting Student Knowledge

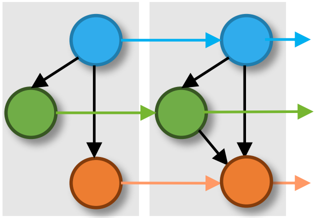


Bayesian Knowledge Tracing is simple, efficient, and interpretable

Modeling and Predicting Student Knowledge

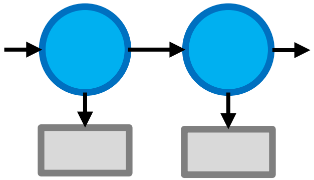


Bayesian Knowledge Tracing is simple, efficient, and interpretable

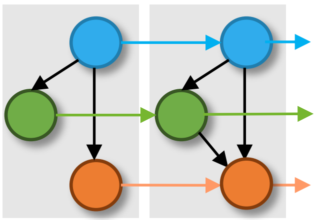


Dynamic Bayesian Networks can represent the hierachical relations between the different skills

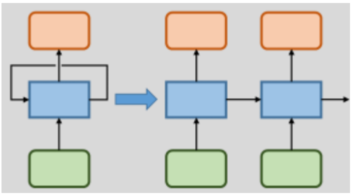
Modeling and Predicting Student Knowledge



Bayesian Knowledge Tracing is simple, efficient, and interpretable



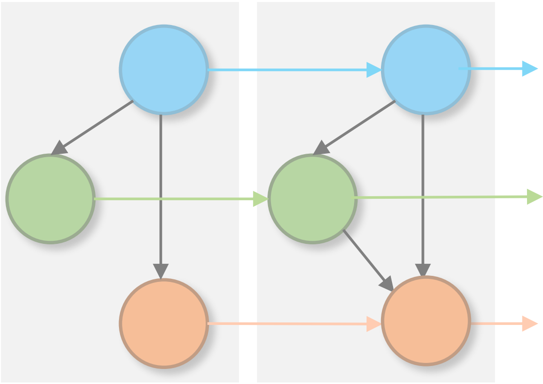
Dynamic Bayesian Networks can represent the hierachical relations between the different skills



Deep Knowledge Tracing can learn non-linear relationships and implicitly captures the relations between the skills

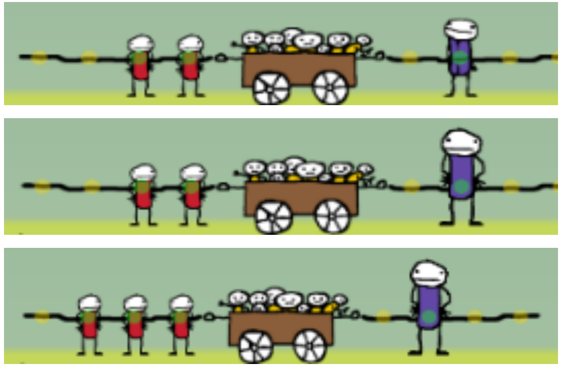
Modeling and Individualizing Learning in Computer-Based Environments

Modeling and predicting knowledge



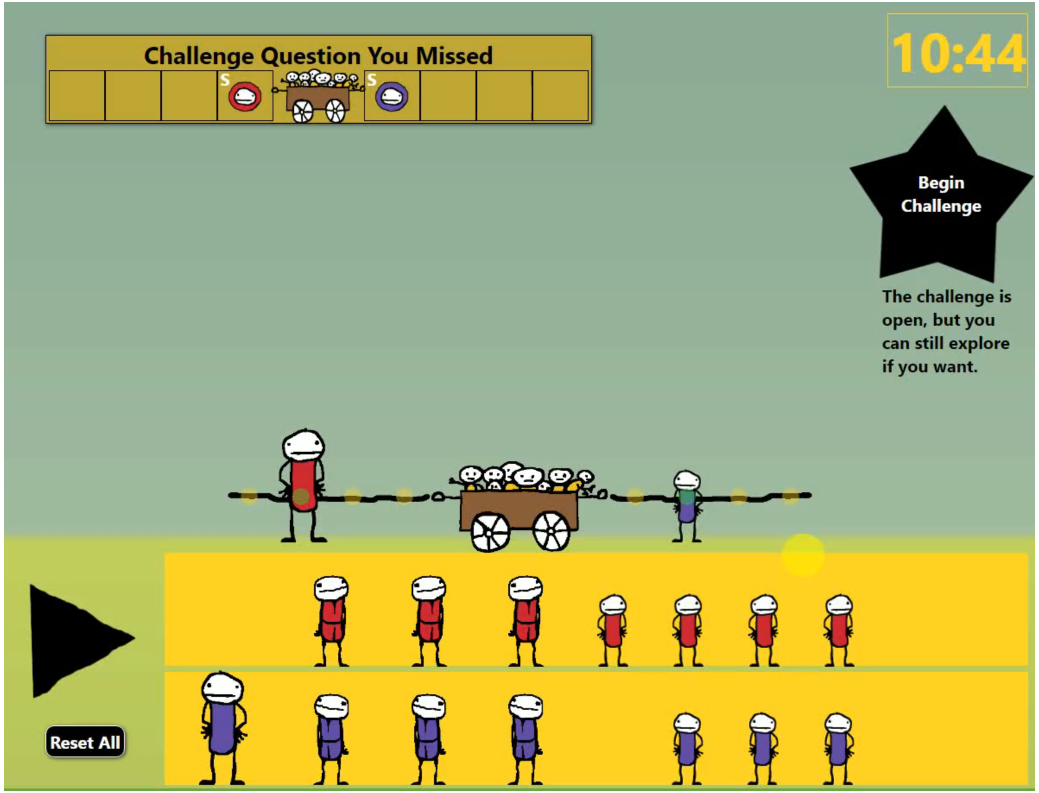
What?

Detecting learner choices and strategies



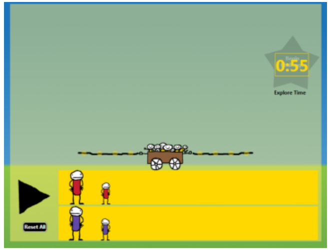
How?

Which team wins the tug-of-war?



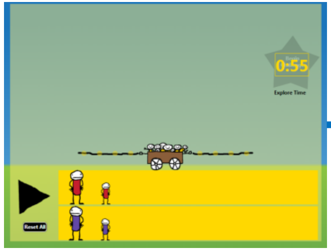
Students can freely choose between two modes

Intro



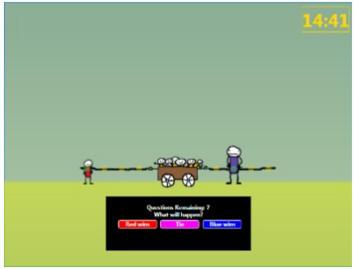
Students can freely choose between two modes

Intro

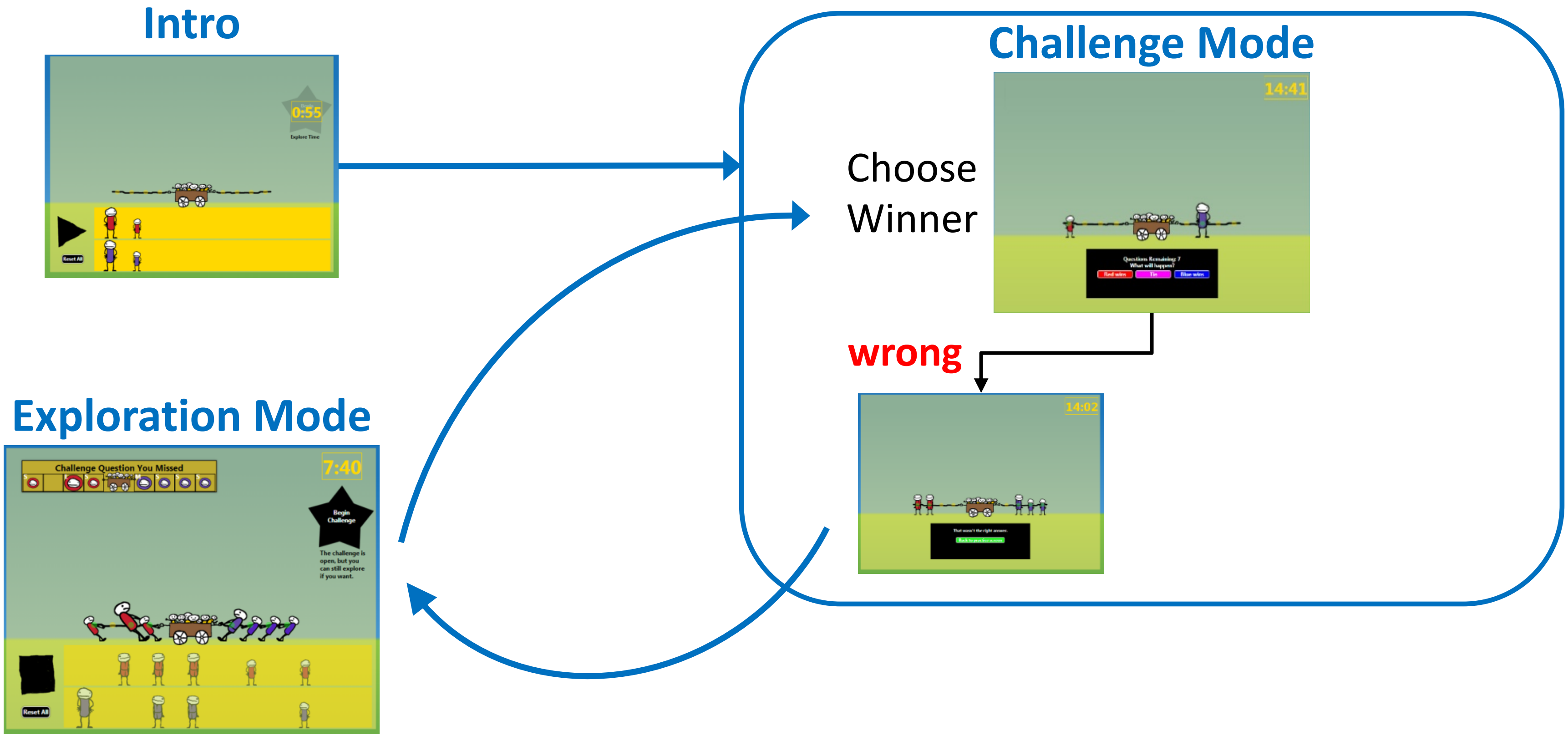


Challenge Mode

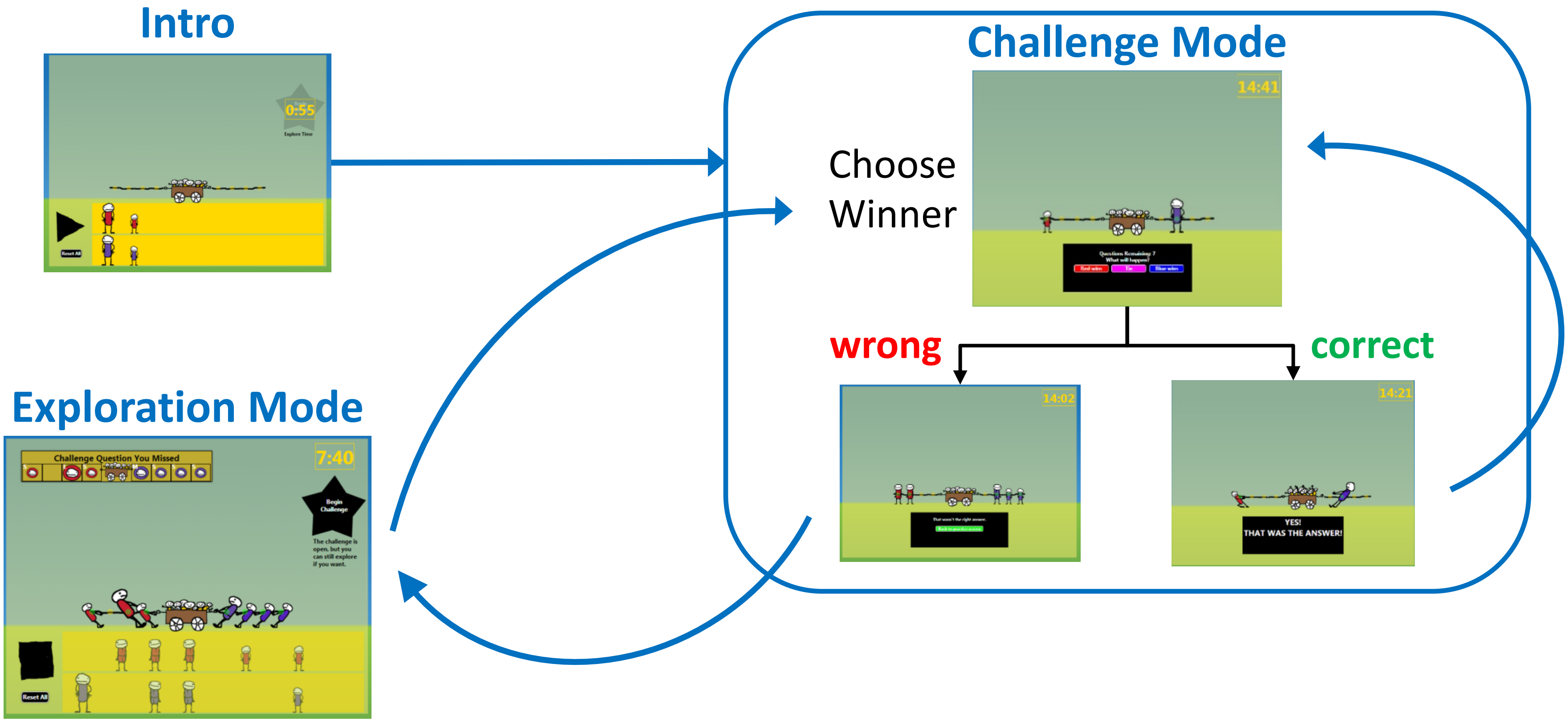
Choose Winner



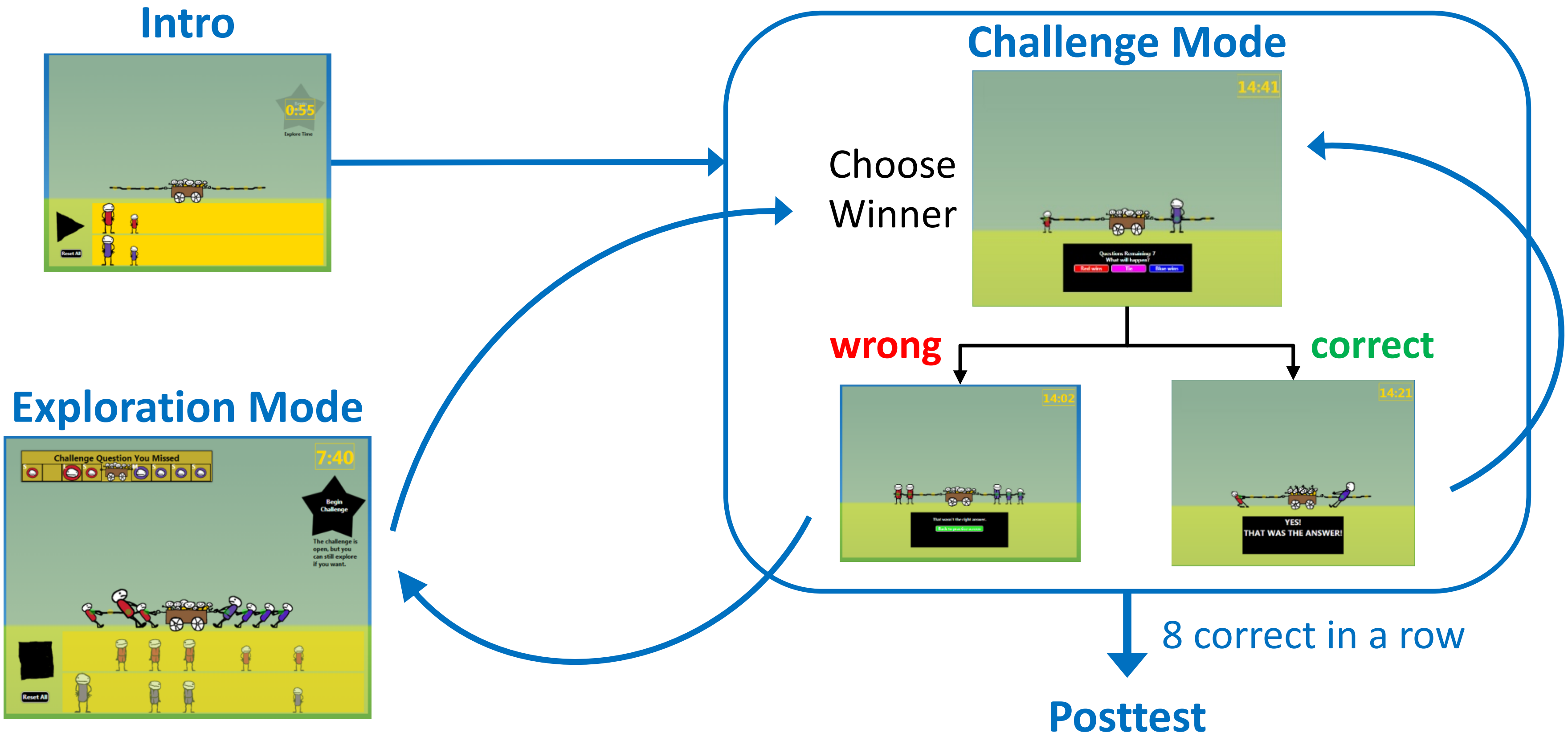
Students can freely choose between two modes



Students can freely choose between two modes

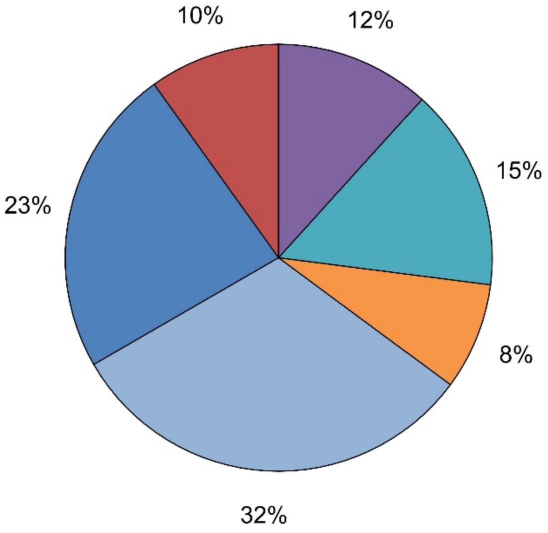


Students can freely choose between two modes

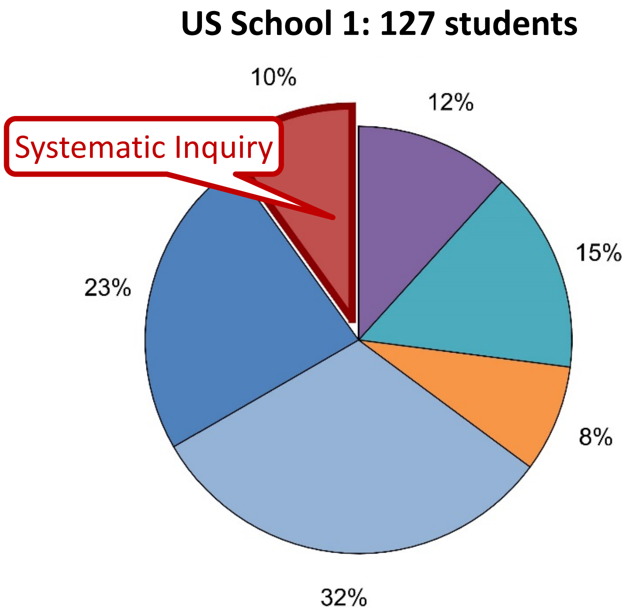


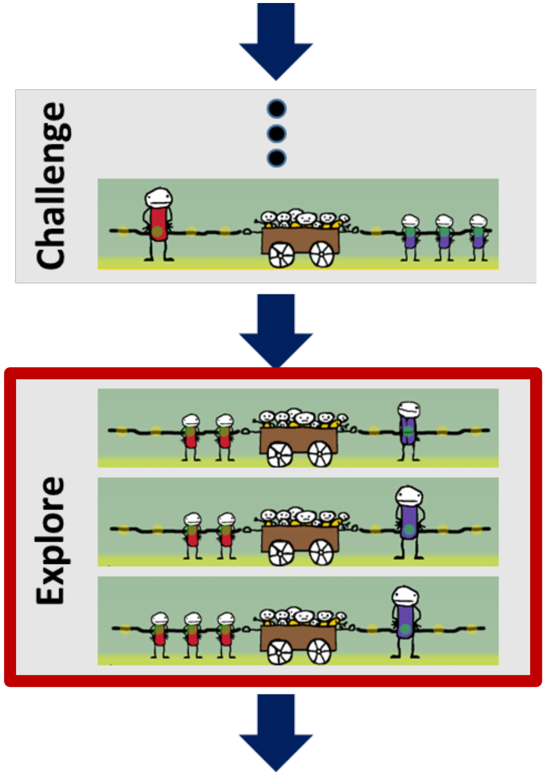
Students can be divided into six different clusters

US School 1: 127 students

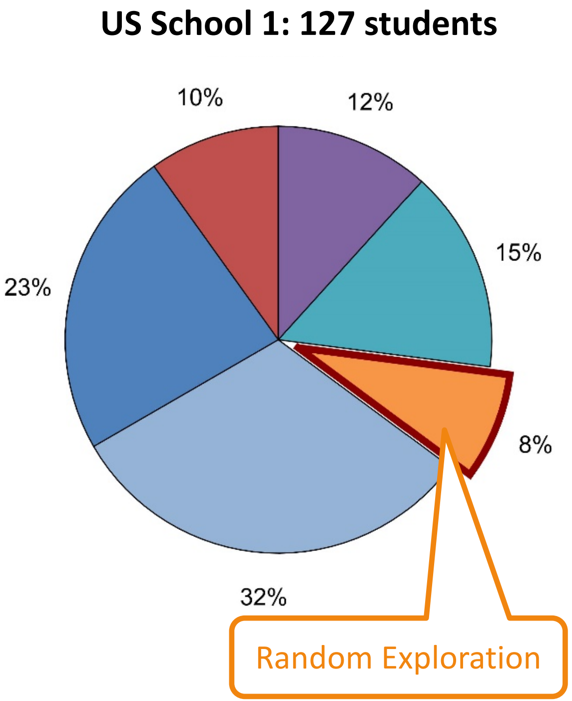


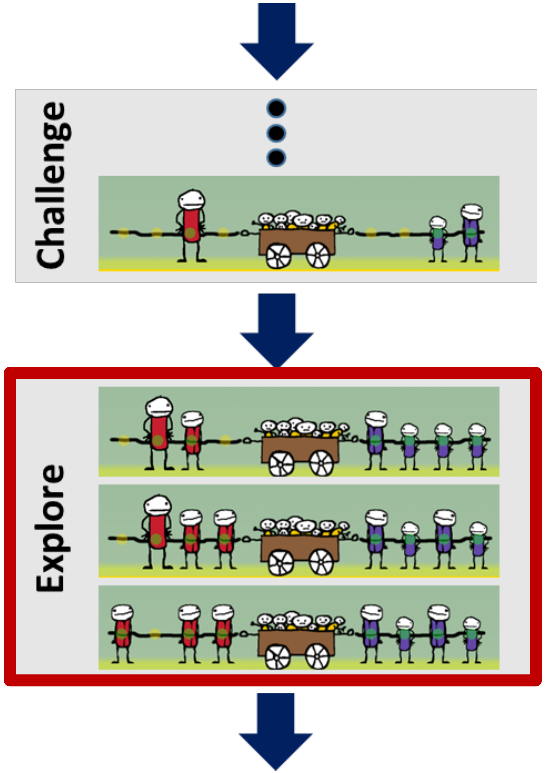
The best students explore systematically



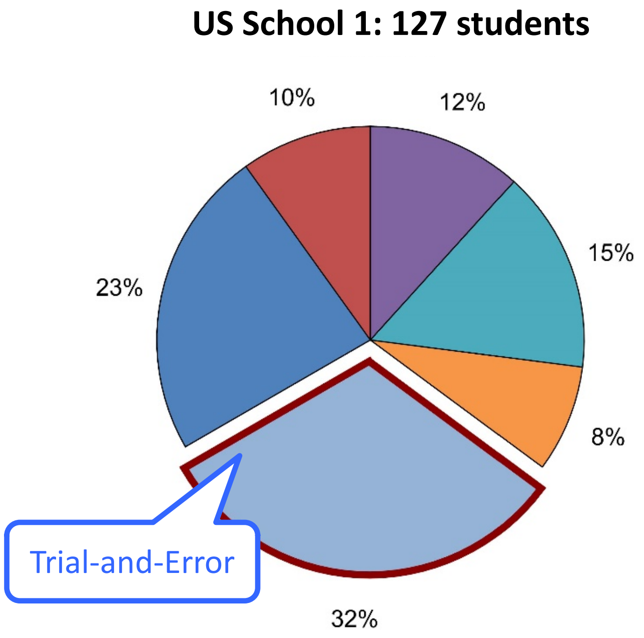


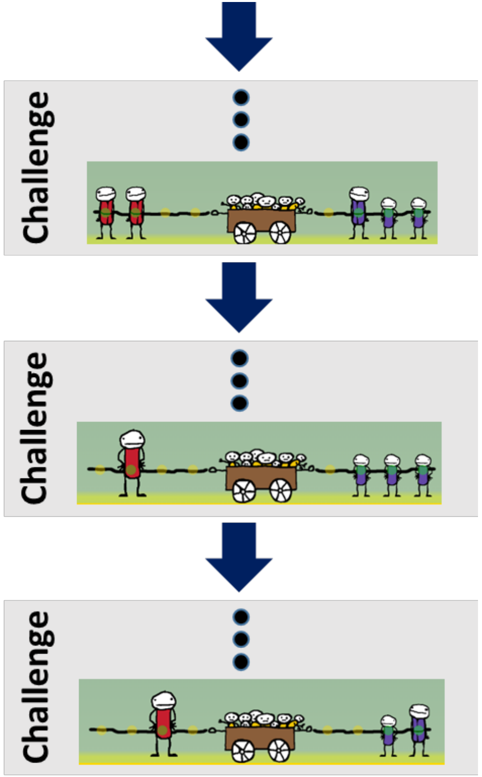
Persistent inquiry alone is not enough



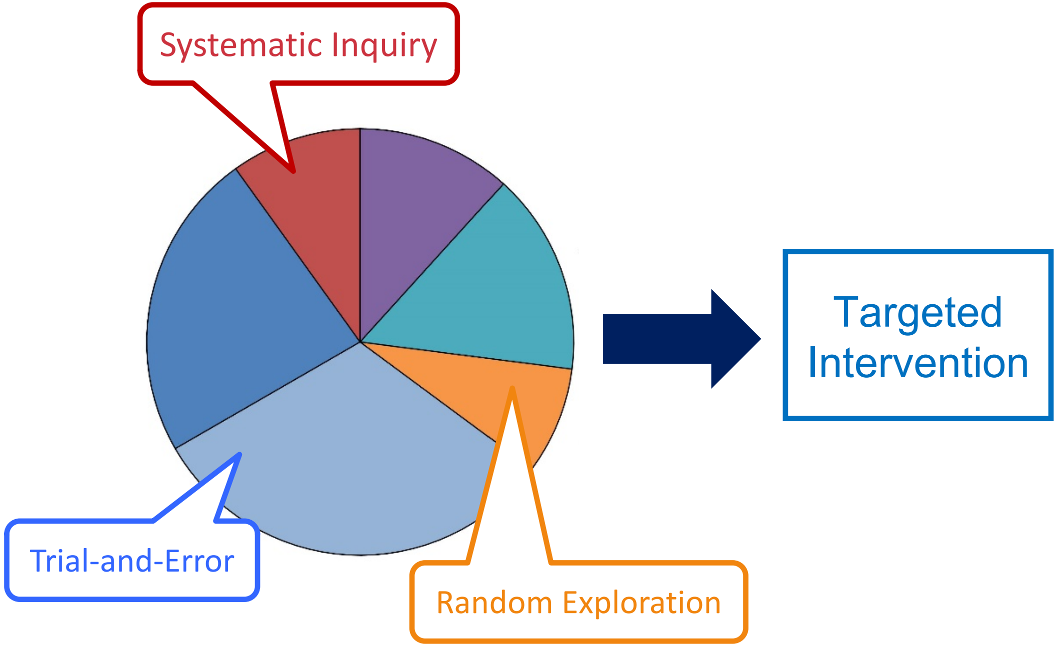


Many students just try to beat the game

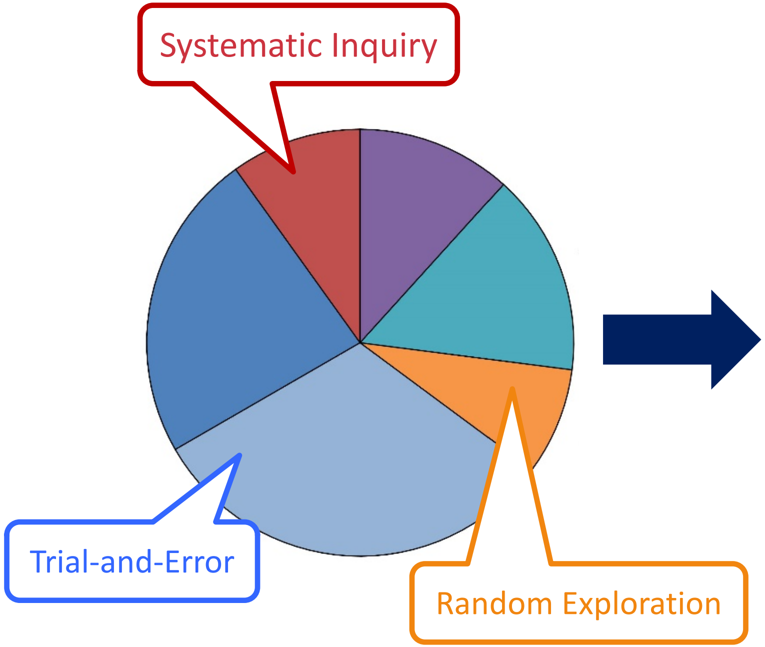




Adaptation based on students' learning behavior

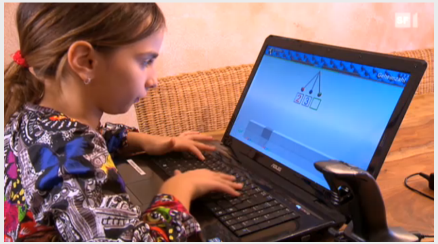


Exploring the use of recurrent neural networks



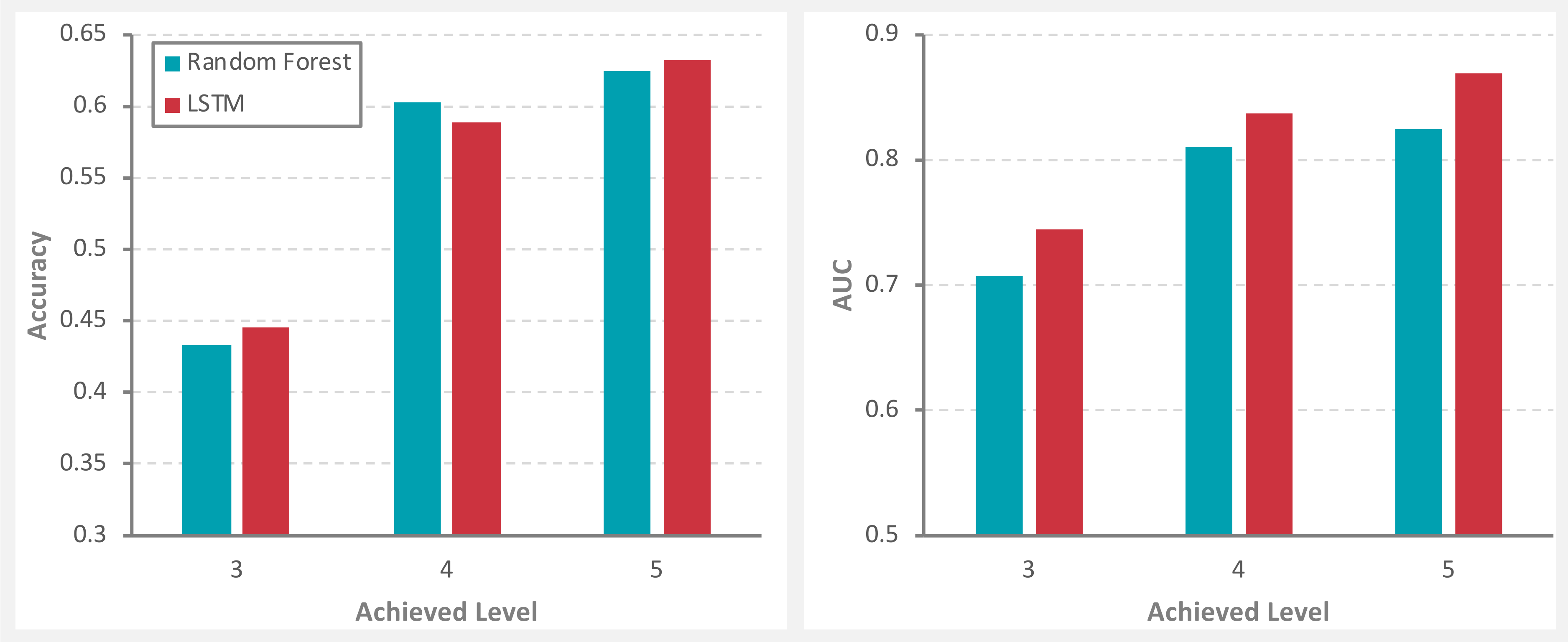
Targeted Intervention

Online Classification



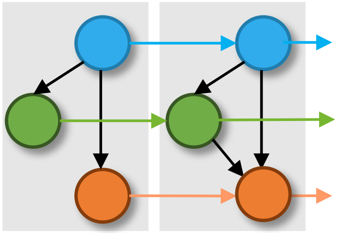
Recurrent Neural Networks

LSTMs are similar or better at important levels



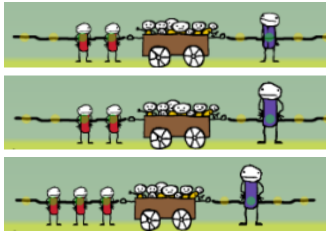
[Käser & Schwartz, EDM 2019]

Modeling and Individualizing Learning in Computer-Based Environments



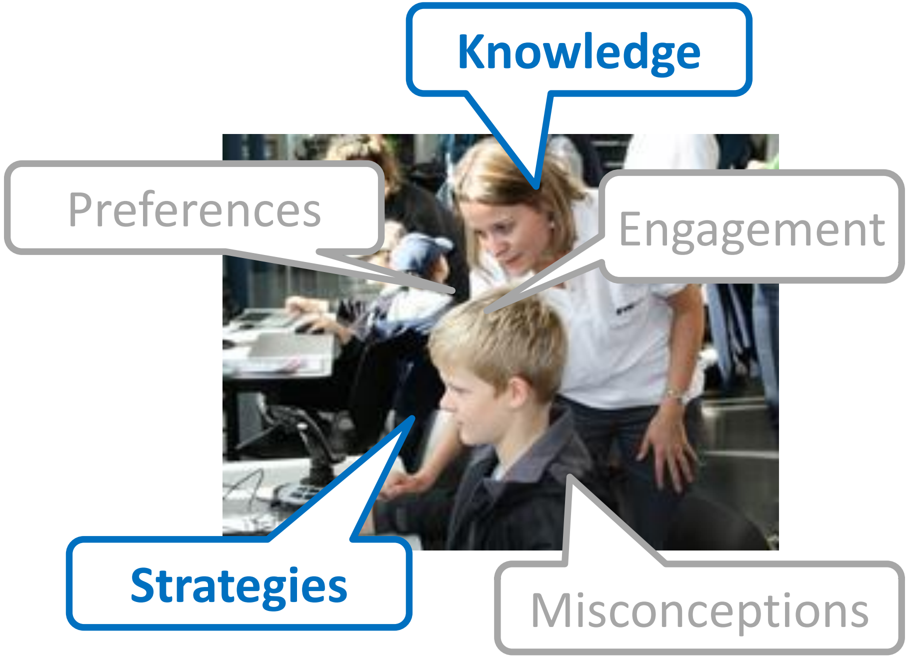
What?

Improving predictions of knowledge

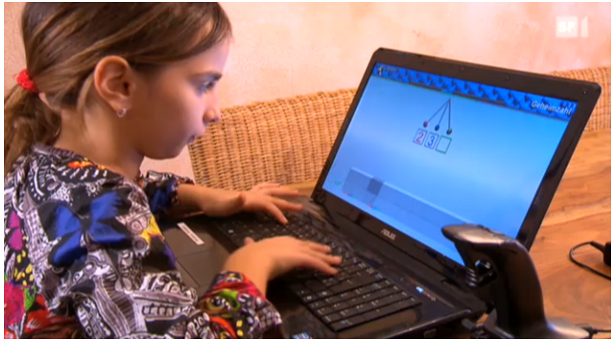


How?

Detecting & classifying learner choices and strategies



Questions?



tanja.kaeser@sdsc.ethz.ch

References

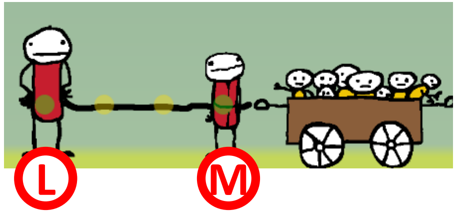
1) Corbett, A. T., and Anderson, J. R. (1995*). Knowledge tracing: Modeling the acquisition of procedural knowledge*. User Modeling and User-Adapted Interaction
2) Yudelson, M.V., Koedinger, K.R., and Gordon, G.J. (2013). *Individualized bayesian knowledge tracing models*. Proceedings of AIED
3) Käser, T., Baschera, G., Kohn, J., Kucian, K. , Richtmann, V., Grond, U., Gross, M., and von Aster, M. (2013). *Design and evaluation of the computer-based training program Calcularis for enhancing numerical cognition.* Frontiers in Psychology
4) Käser, T., Klingler, S., Schwing, A., and Gross, M. (2014). *Computational Education using Latent Structured Prediction.* Proceedings of AISTATS
5) Käser, T., Klingler, S., Schwing, A., and Gross, M. (2014). Beyond KnowledgeTracing: Modeling Skill Topologies with Bayesian Networks. Proceedings of ITS
6) Piech, C., Bassen, J., Huang, J., Ganguli, S., Sahami, M., Guibas, L., and Sohl-Dickstein, J. (2015). *Deep Knowledge Tracing*. Proceedings of NIPS
7) Xiong, X., Zhao, S., Van Inwegen, E. G., Beck, J. E. (2016*). Going Deeper with Deep Knowledge Tracing*. Proceedings of EDM
8) Käser, T., Klingler, S., Schwing, A., and Gross, M. (2017). Dynamic Bayesian Networks for Student Modeling. *IEEE Transactions on Learning Technologies*
9) Käser, T., and Schwartz, D. L. (2019). Exploring Neural Network Models for the Classification of Students in Highly Interactive Environments. *Proceedings of EDM*

BACKUP

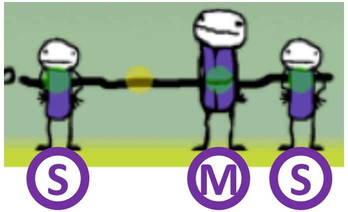
Description of US data sets

	US School 1	US School 2
Number of students	127	165
Age	8^{th} grade	8^{th} grade
Time in exploration mode	42%	23%
Students passing the game	87%	97%
Students with perfect post-test	24%	34%
Average post-test score	2.1	2.6

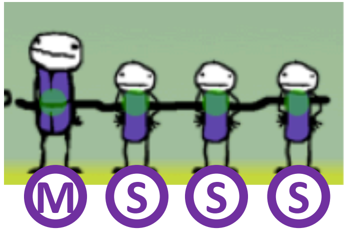
Posttest



1) ☐

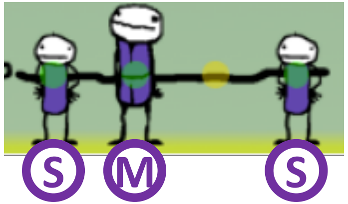


2) ☒



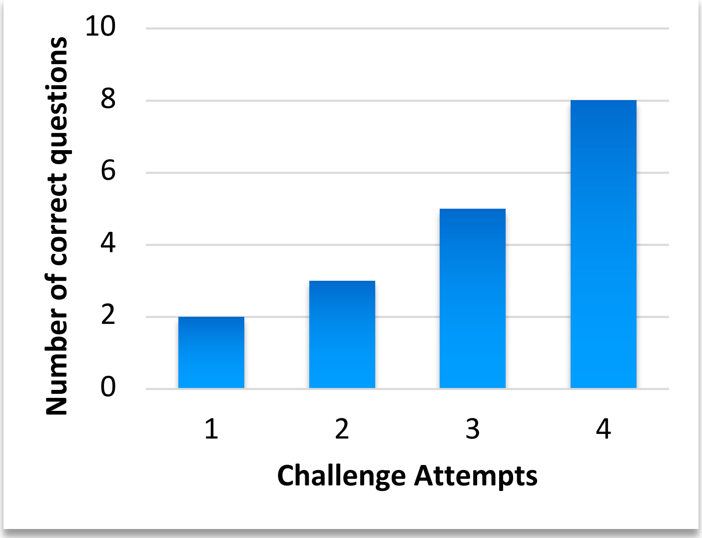
⋮

10) ☐



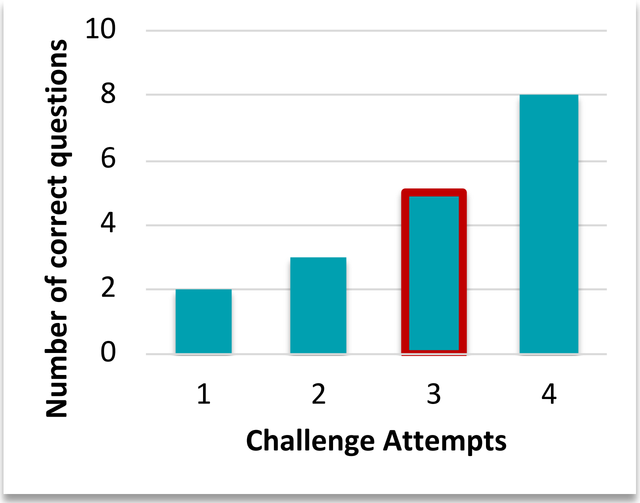
Clustering students based on features describing their exploration behavior

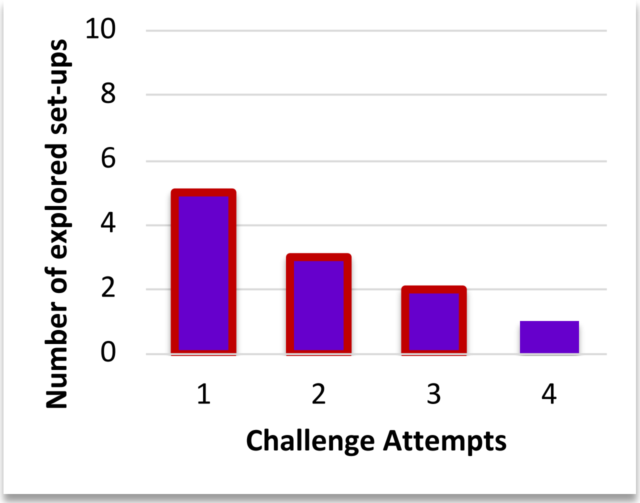
➡ **Number of challenge questions answered until passing a level (NC)**



Clustering students based on features describing their exploration behavior

- Number of challenge questions answered until passing a level (NC)
- **Number of explored set-ups until passing a level (NS)**





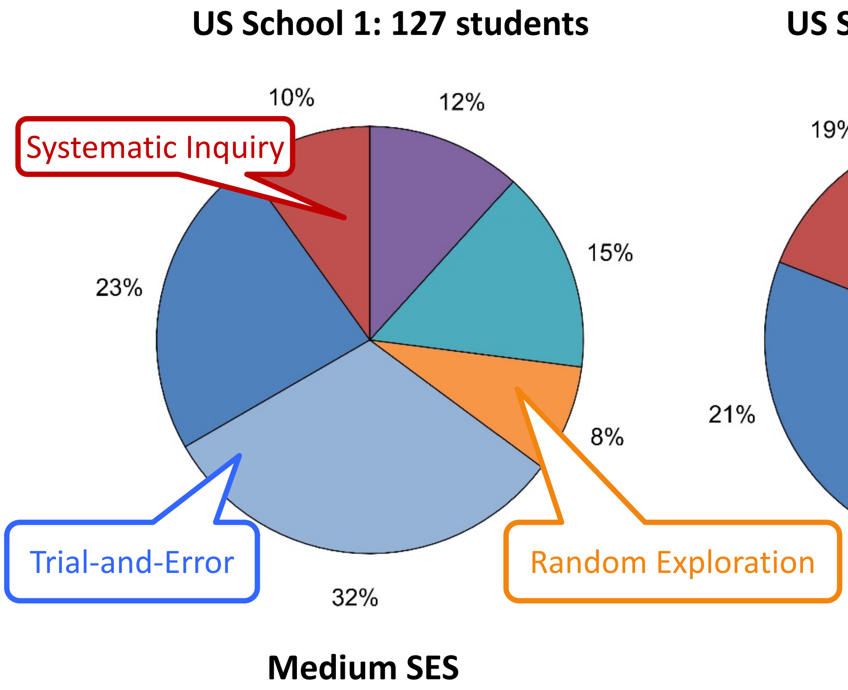
Clustering students based on features describing their exploration behavior

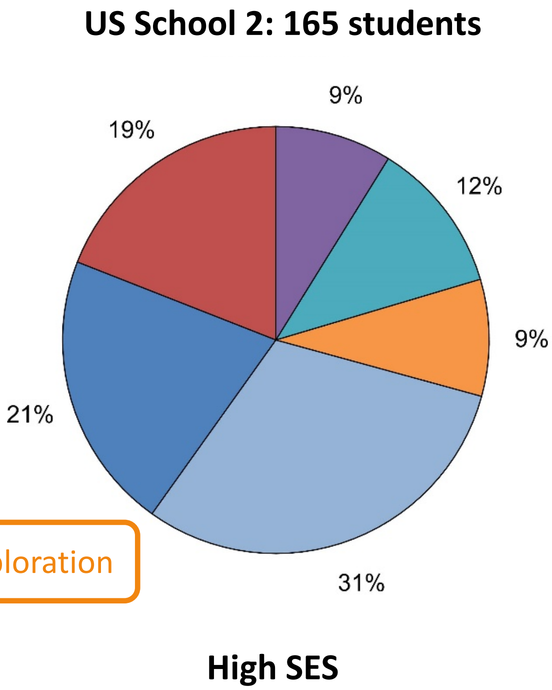
- Number of challenge questions answered until passing a level (NC)
- Number of explored set-ups until passing a level (NS)
- **Number of explored set-ups rated as strong until passing a level (NSS)**

Strong

Large = 3*Small

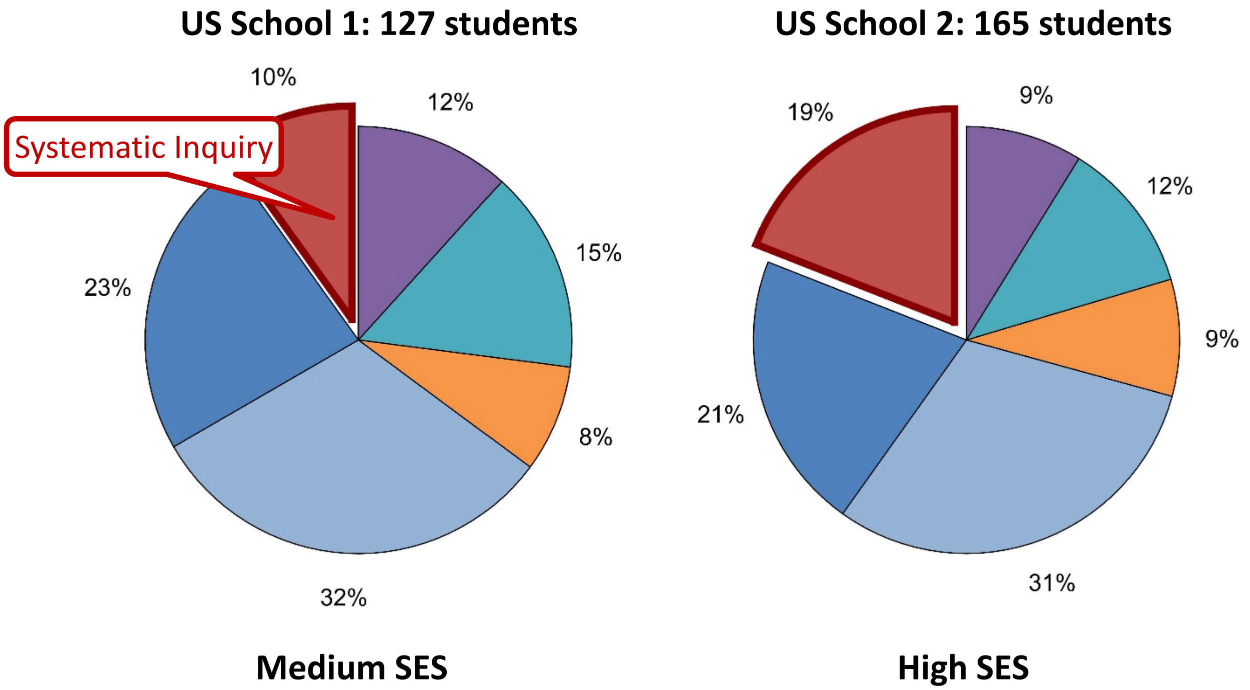
The cluster solution was replicated on a second independent data set





[Käser & Schwartz, IJAIED (under review)]

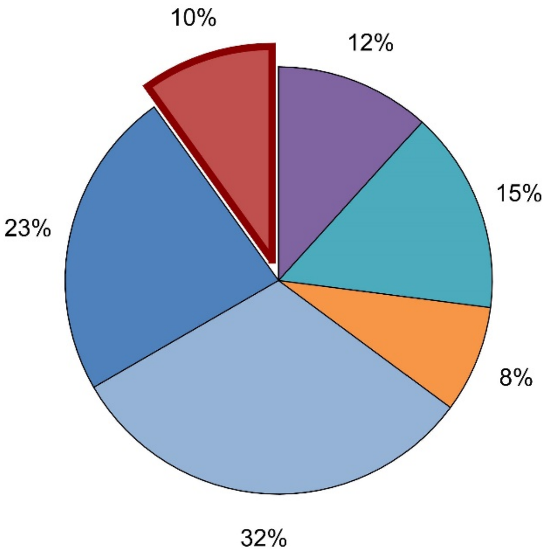
More students explore systematically



[Käser & Schwartz, IJAIED (under review)]

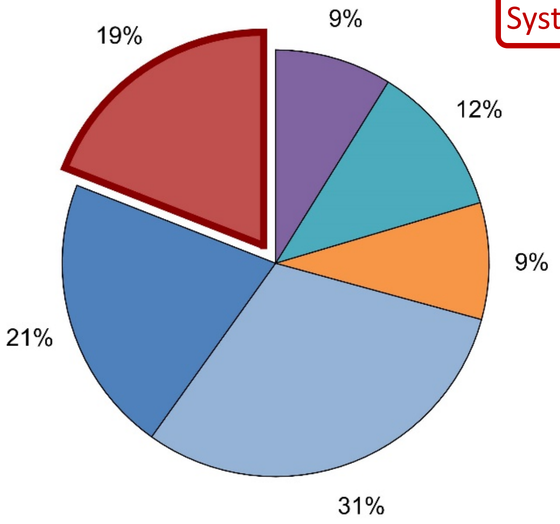
Exploring students' inquiry strategies across cultural context

US School 1: 127 students



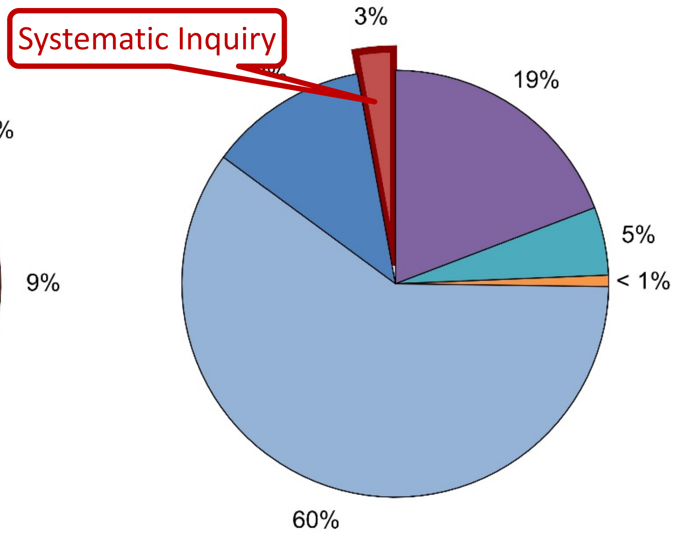
Medium SES

US School 2: 165 students



High SES

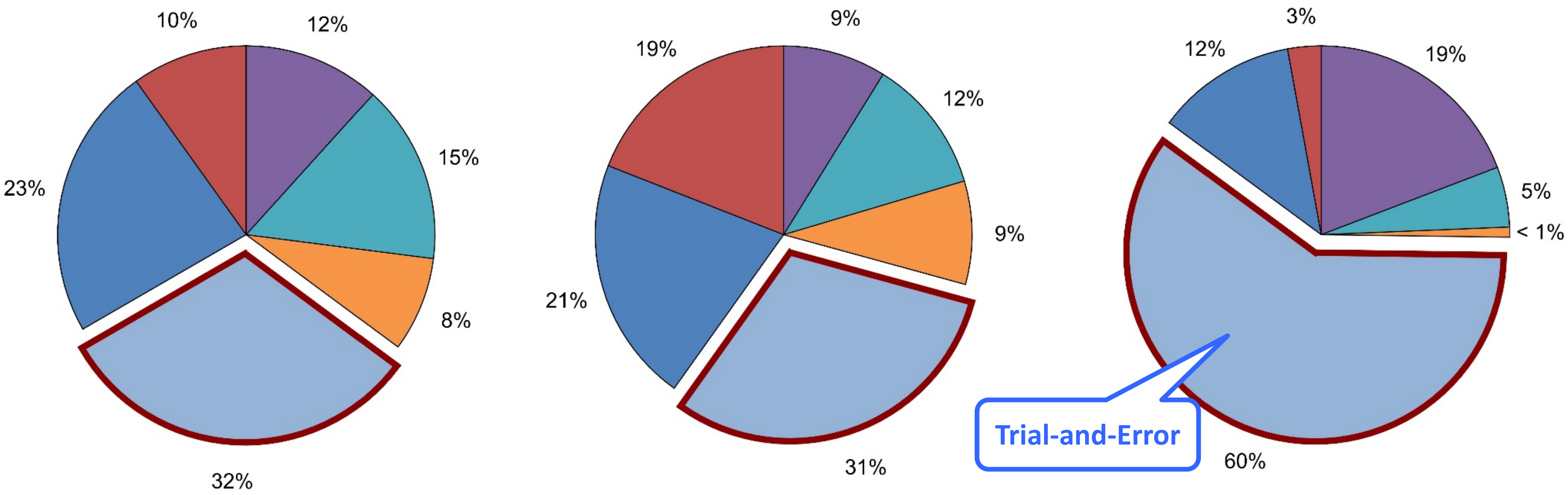
Colombian Schools: 349 students



Low-Medium SES

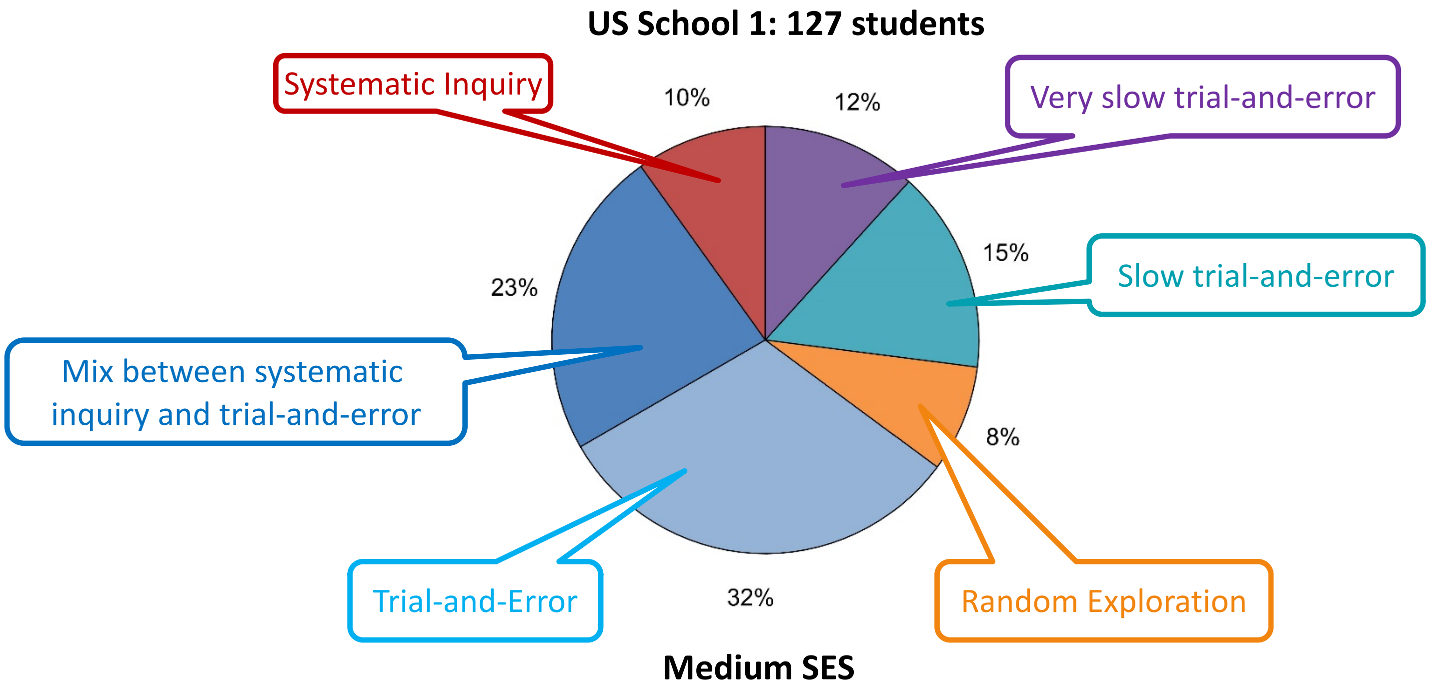
Exploring students' inquiry strategies across cultural context

US School 1: 127 students | **US School 2: 165 students** | **Colombian Schools: 349 students**



Medium SES | **High SES** | **Low-Medium SES**

Clusters can be semantically interpreted



Pairwise Clustering

Constant shift embedding transformation

↓

similarities = distances in higher-dimensional Euclidean space

↓

k-Means Clustering

Computation of BIC

$$\boxed{BIC = -2 \cdot \log(L) + k \cdot \log(n) + (k-1) + 1}$$

- L = likelihood of data
  - Fit Gaussian distribution per cluster
  - Estimate variance by distance to cluster centroid
  - Estimate mean by cluster centroid
  - Sum up gaussians over all clusters, taking into account the cluster probability
- k = number of clusters
- n = number of effective dimensions of transformation matrix

Likelihood Computation

- Variance $\boldsymbol{\sigma}^2$: $\frac{\mathbf{1}}{\boldsymbol{R-k}} \cdot \sum_i (\boldsymbol{x_i} - \boldsymbol{cc})^2$
  - R: Sample size
  - k: Number of clusters
  - cc: Centroid of according cluster
- $L_c = \frac{\mathbf{1}}{\boldsymbol{p_c}} \cdot \sum \frac{\mathbf{1}}{\sqrt{\mathbf{2}\boldsymbol{\pi\sigma}^2}} \cdot \boldsymbol{e}^{(\frac{\boldsymbol{x_i - cc}}{\boldsymbol{\sigma}})^2}$
  - p_c: Prior probability for cluster

Cluster Stability

- US School 1: Original data set
- US School 2: New data set
- Cluster US School 1 -> Original clustering solution (*OC*)
- k-Nearest Neighbor assigns each sample from school 2 to a cluster *c* of *OC* -> vector of predicted labels $\mathbf{l}_{\mathbf{p}}$
- Cluster US School 2 -> New clustering solution with labels $\mathbf{l}_{\mathbf{NC}}$
- Cluster stability = Hamming distance between $\mathbf{l}_{\mathbf{p}}$ and $\mathbf{l}_{\mathbf{NC}}$

[Lange et al., Neural Comput. 2004]

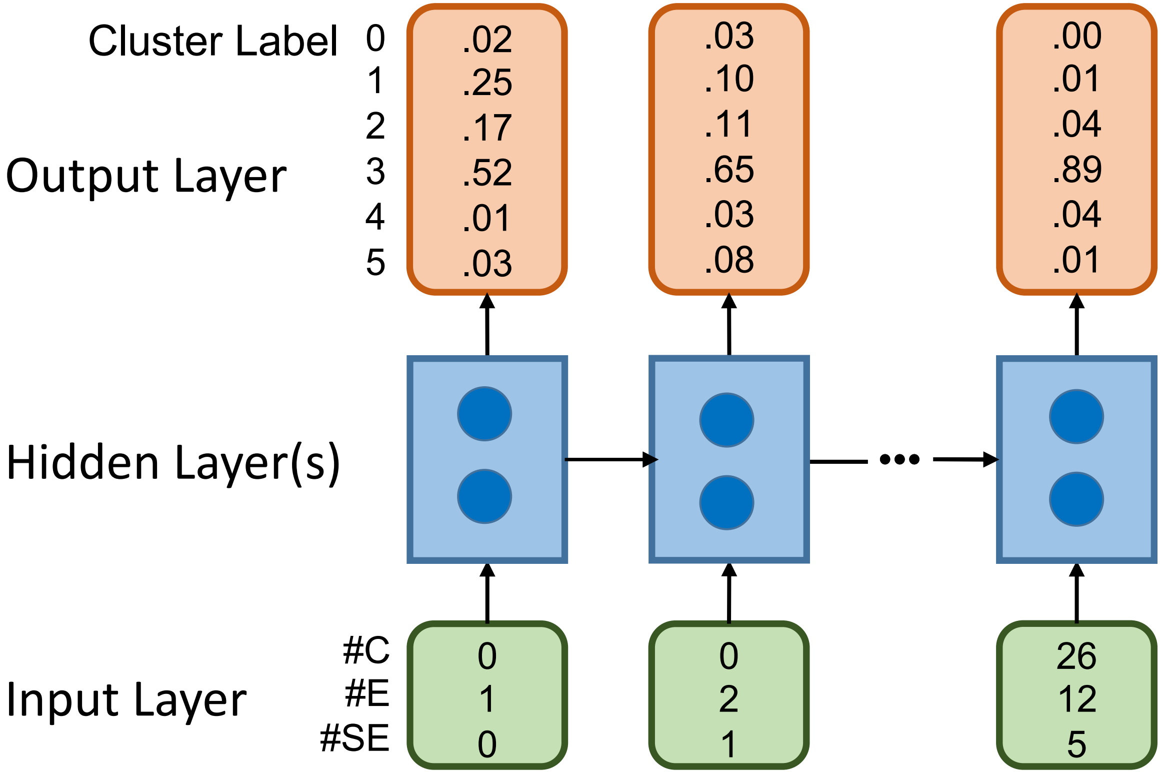
Exploring the use of recurrent neural networks

		1 x 4	1 x 8	1 x 16	1 x 32	2 x 2	2 x 4	2 x 8	2 x 16
Predicting Sequence	LSTM								
	GRU								
Optimized for point in time	LSTM								
	GRU								

Exploring the use of recurrent neural networks

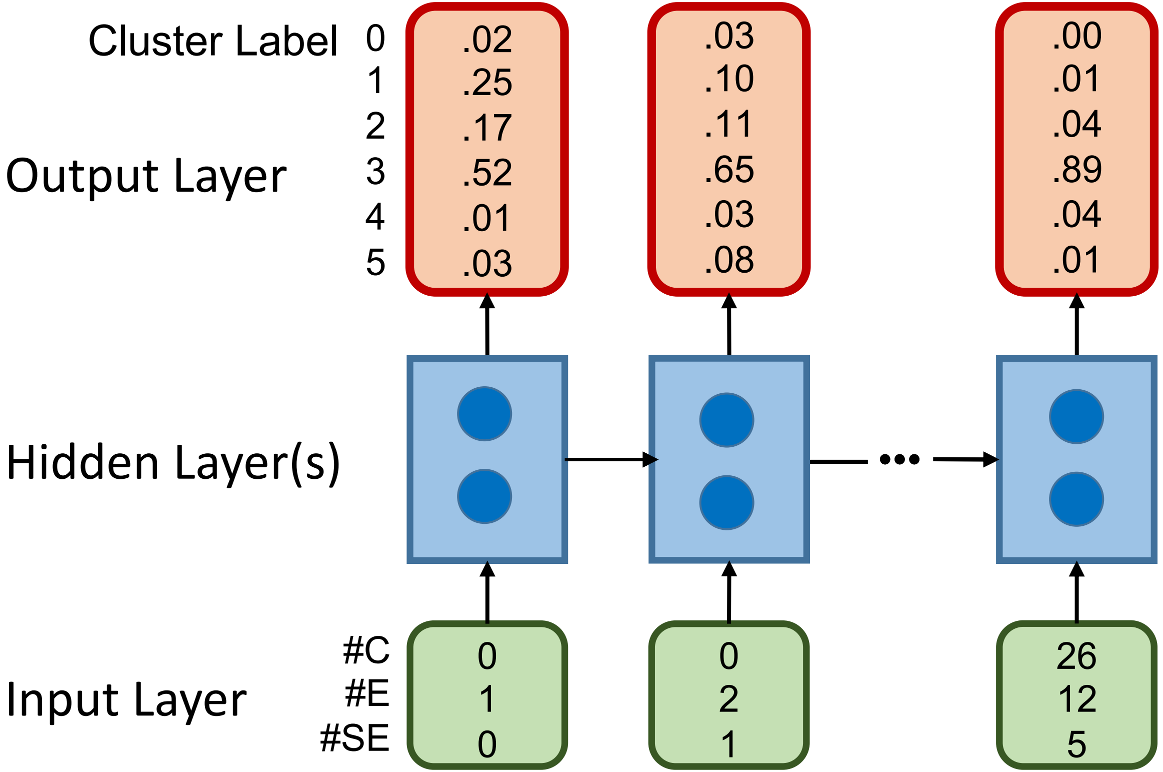
		1 x 4	1 x 8	1 x 16	1 x 32	2 x 2	2 x 4	2 x 8	2 x 16
LSTM	Predicting Sequence								
GRU	Predicting Sequence								
LSTM	Optimized for point in time								
GRU	Optimized for point in time								

Output layer consist of predicted probabilities



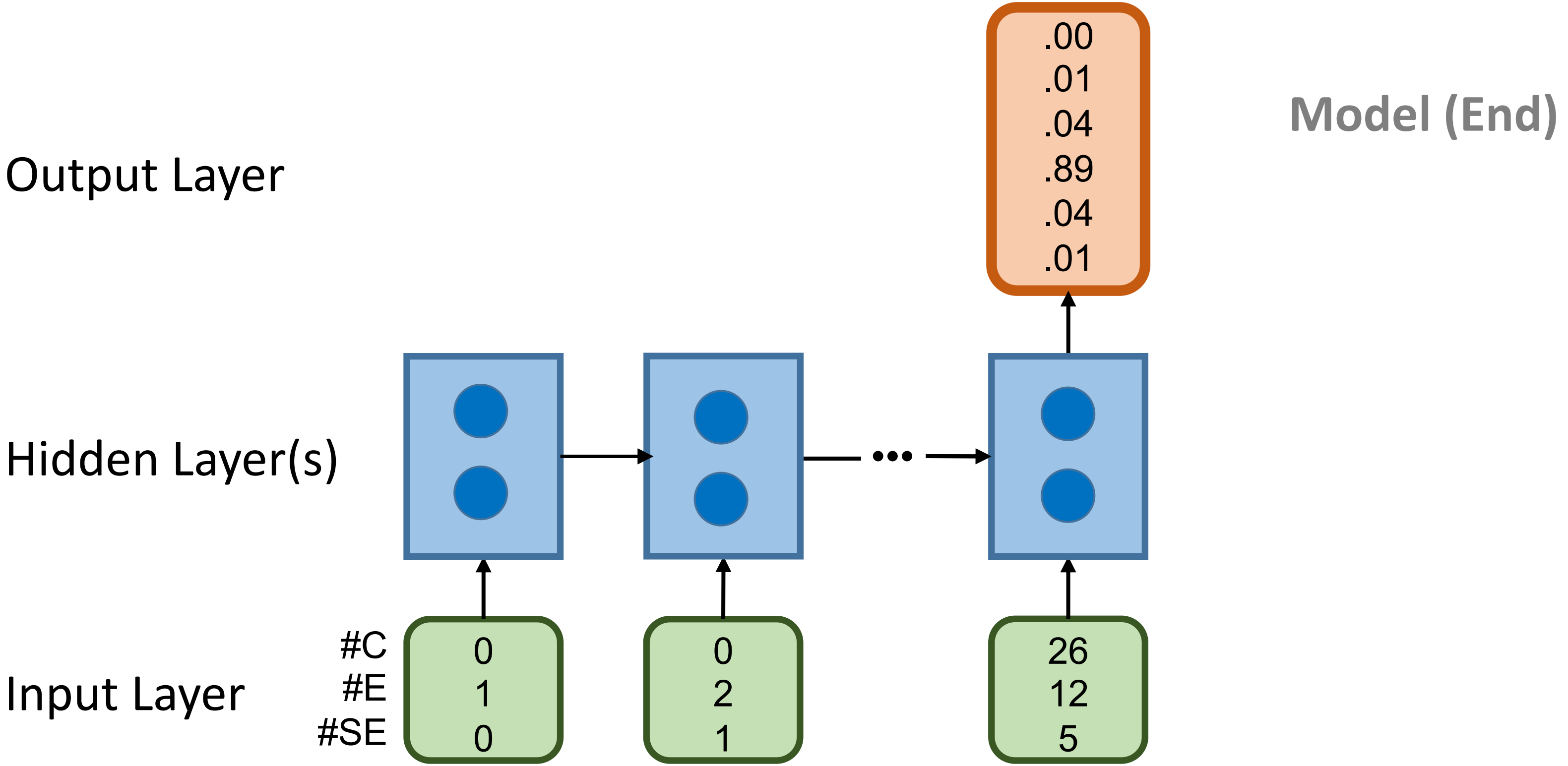
$$y_t = \mathrm{softmax}(W_{yh} h_t + b_y)$$

Model outputs a probability at each time step

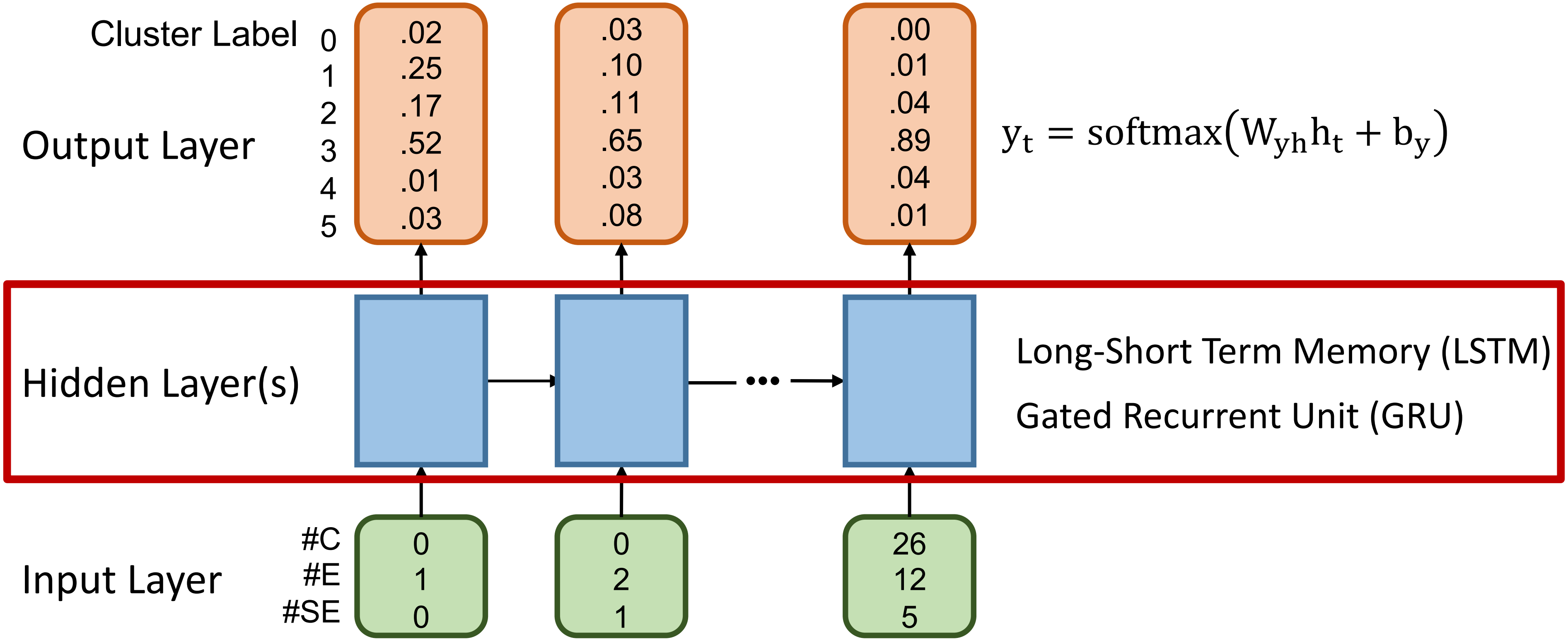


Model (Seq)

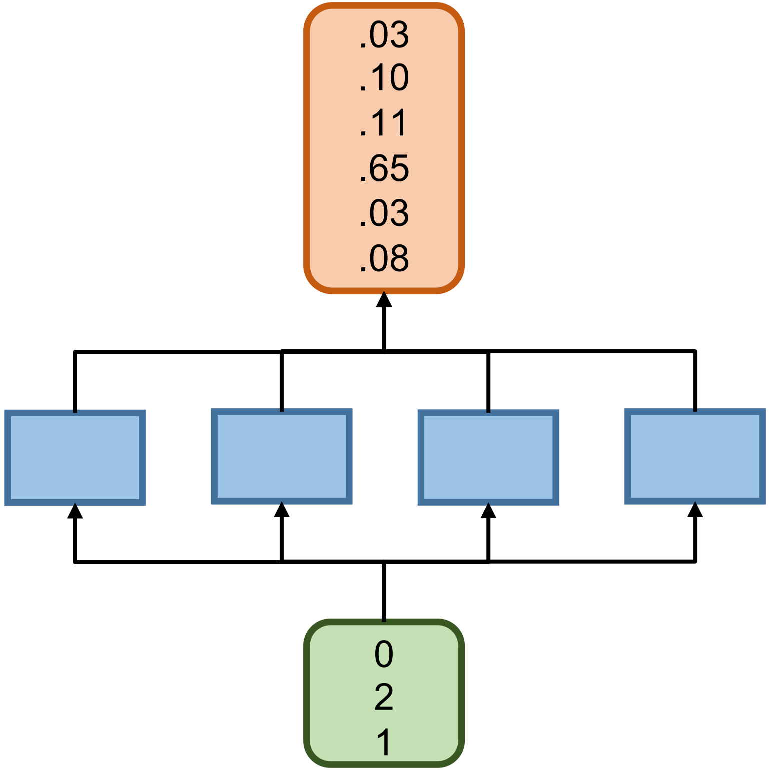
Model outputs a probability at the end



Hidden layer captures relevant information

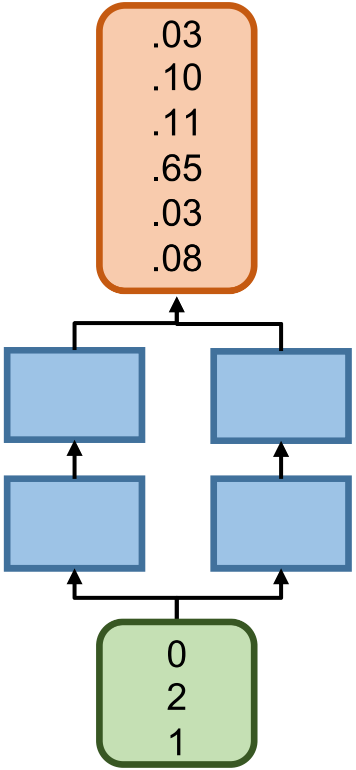


Number of hidden layers and cells per layer vary



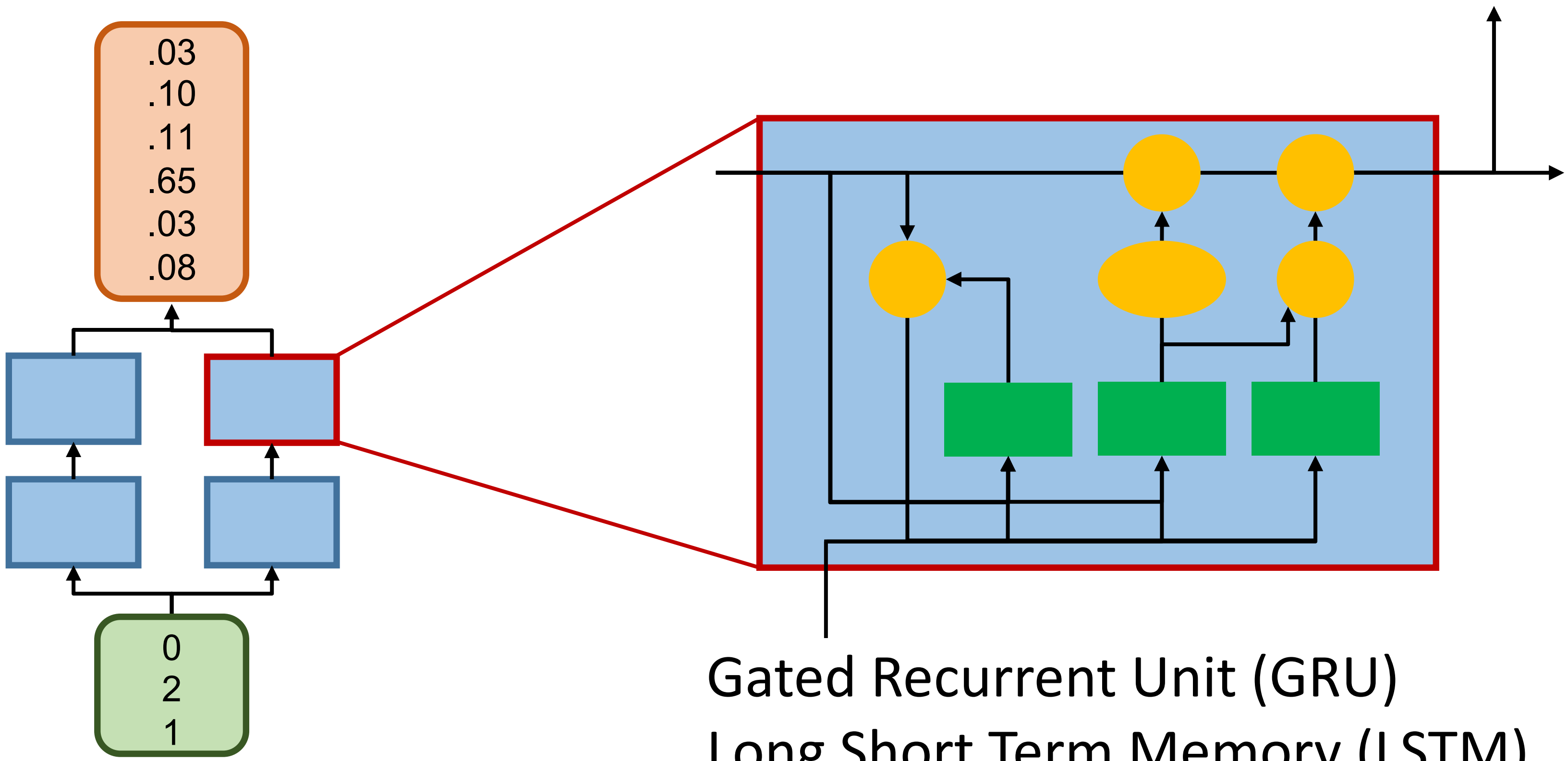
1x4

Number of hidden layers and cells per layer vary



2x2

Architecture of cells varies



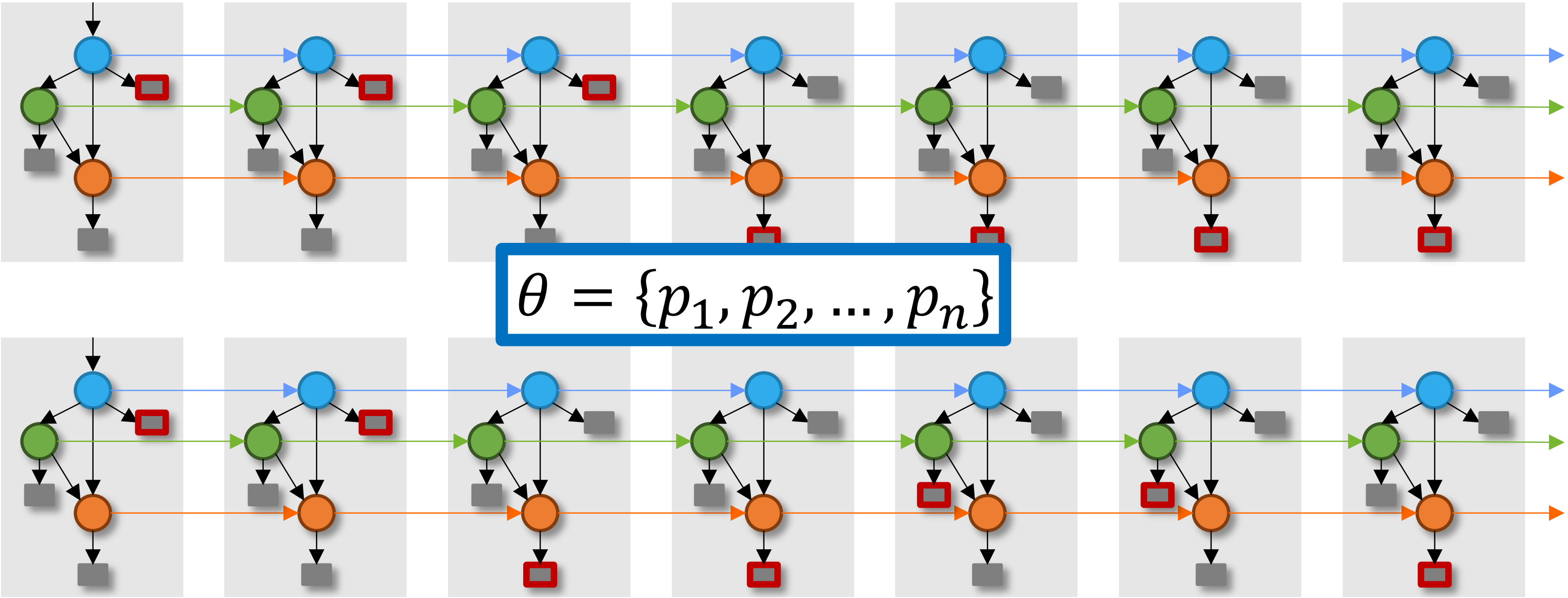
Gated Recurrent Unit (GRU)
Long Short Term Memory (LSTM)

Parameter learning is computationally intractable

Student
$m = 1$

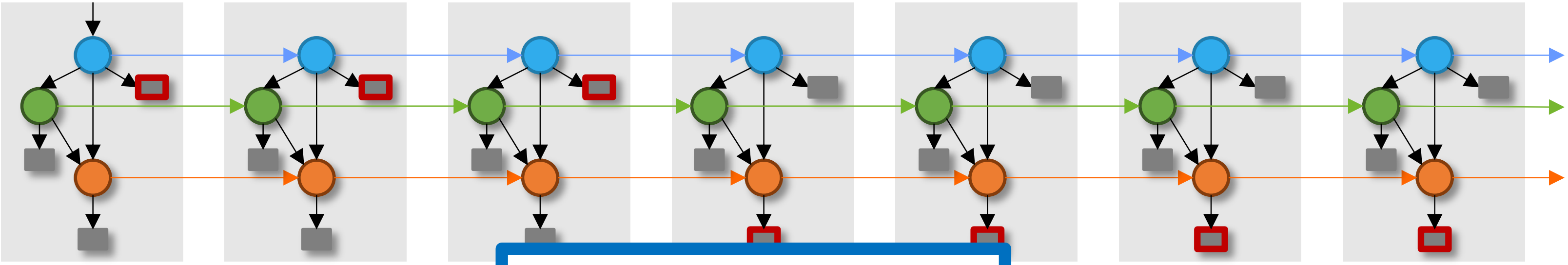
⋮

Student
$m = M$



Parameter learning is computationally intractable

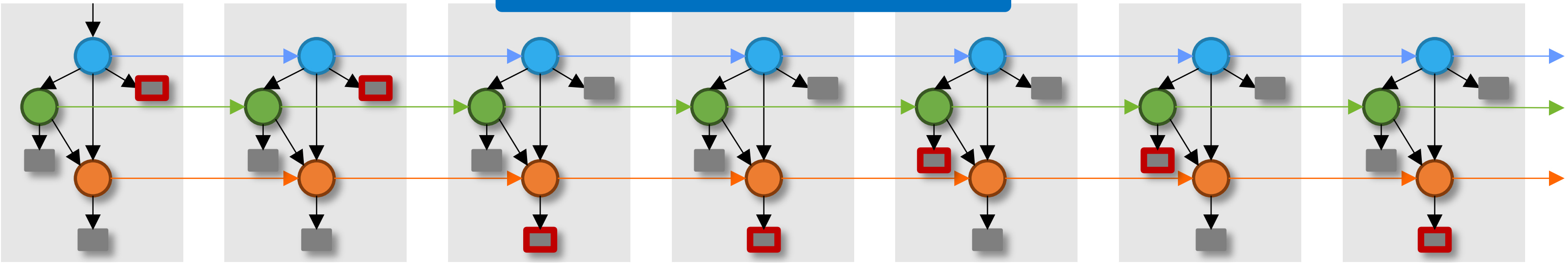
Student $m = 1$



⋮

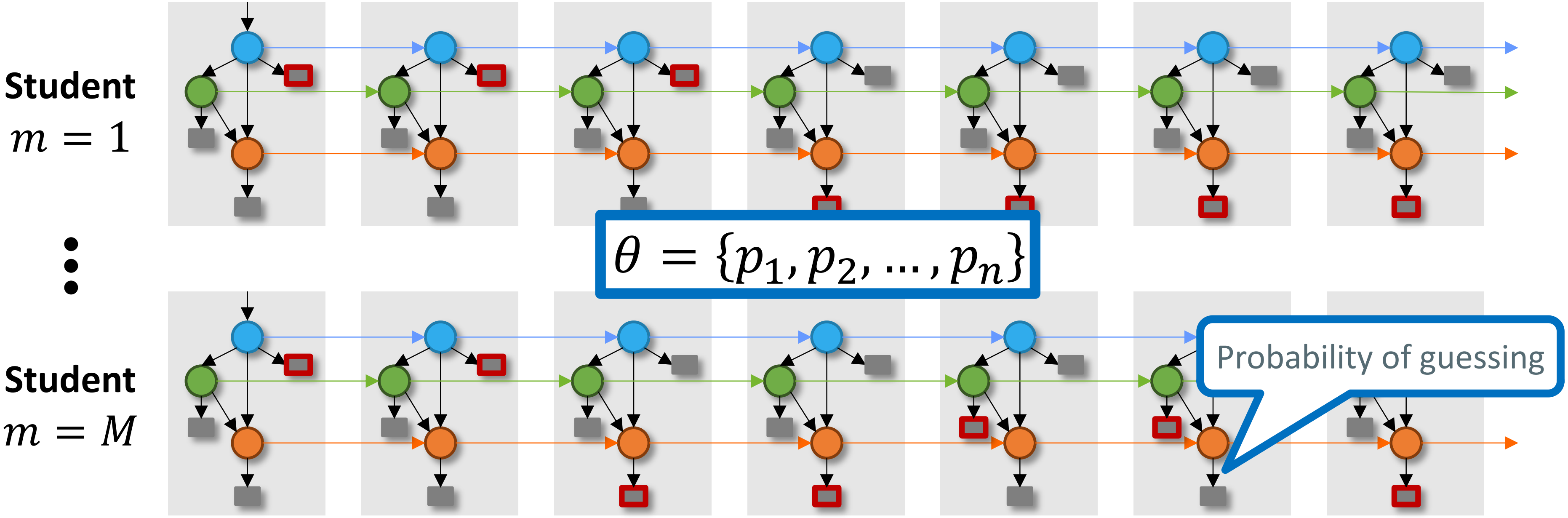
$$\theta = \{p_1, p_2, \dots, p_n\}$$

Student $m = M$



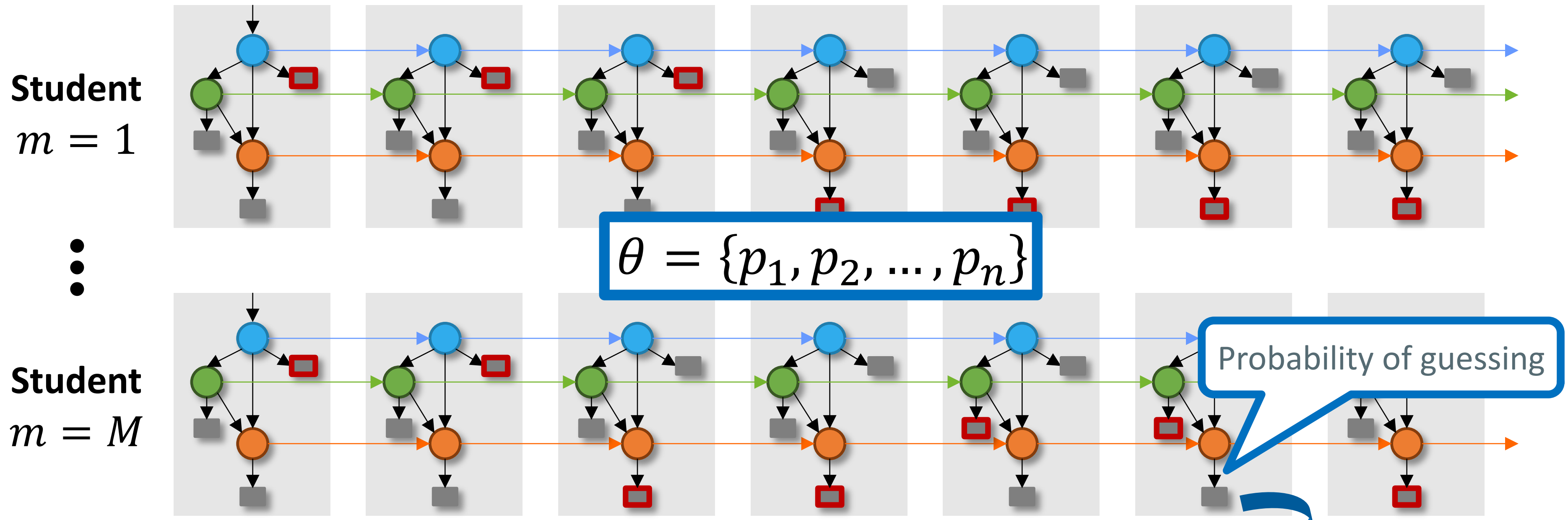
$$\min_{\theta} -\sum_{m} \ln\left(\sum_{h_m} p(y_m, h_m|\theta)\right)$$

Parameter constraints guarantee interpretability



$$\min_{\theta} -\sum_{m} \ln\left(\sum_{h_m} p(y_m, h_m|\theta)\right)$$

Parameter constraints guarantee interpretability



$$\min_{\theta} -\sum_{m} \ln\left(\sum_{h_m} p(y_m, h_m | \theta)\right) \quad \boldsymbol{s.t.\,\theta \in C}$$

$p_G < 0.3$

[Käser et al., AISTATS 2014]

From probabilistic notation to log-linear formulation

$$L(\theta) = \sum_m \ln\left(\sum_{h_m} p(y_m, h_m|\theta)\right)$$

$$L(w) = \sum_m ln\left(\sum_{h_m} exp(\mathbf{w}^T\phi(y_m, h_{,m}) - \ln(Z))\right)$$

[Käser et al., AISTATS 2014]

From probabilistic notation to log-linear formulation

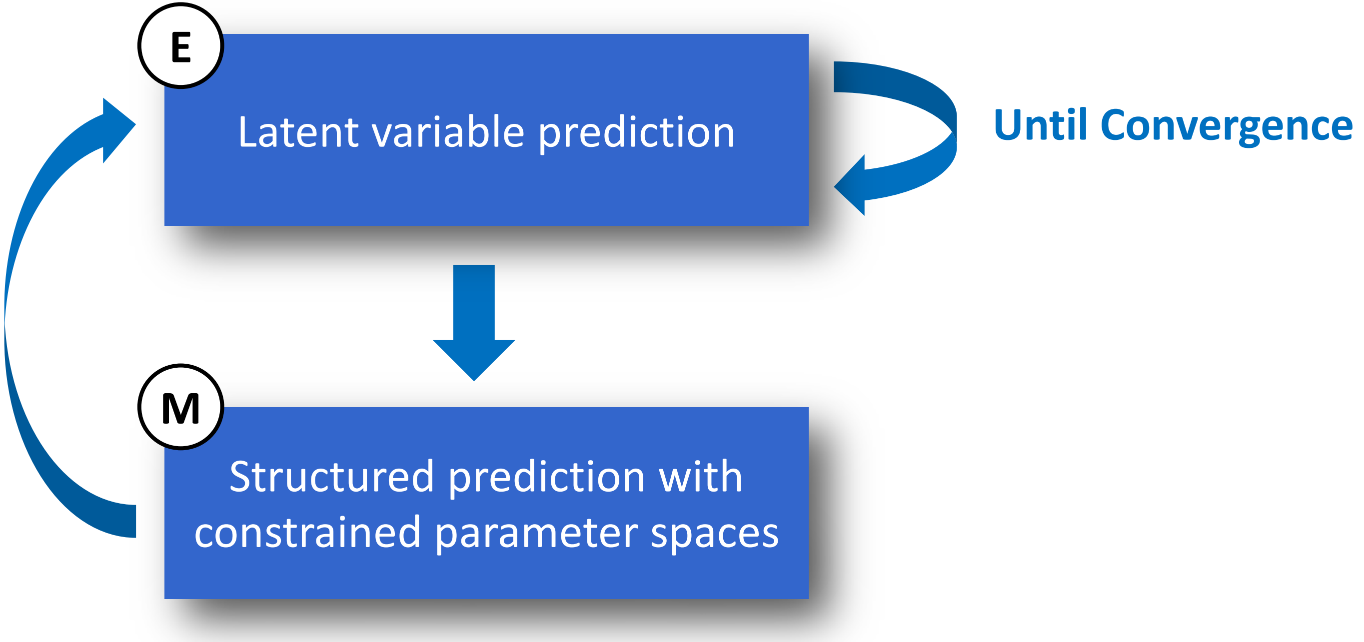
$$L(\theta) = \sum_m \ln\left(\sum_{h_m} p(y_m, h_m|\theta)\right)$$

$$\phi = 1 - 2v, V \in Y \cup H$$

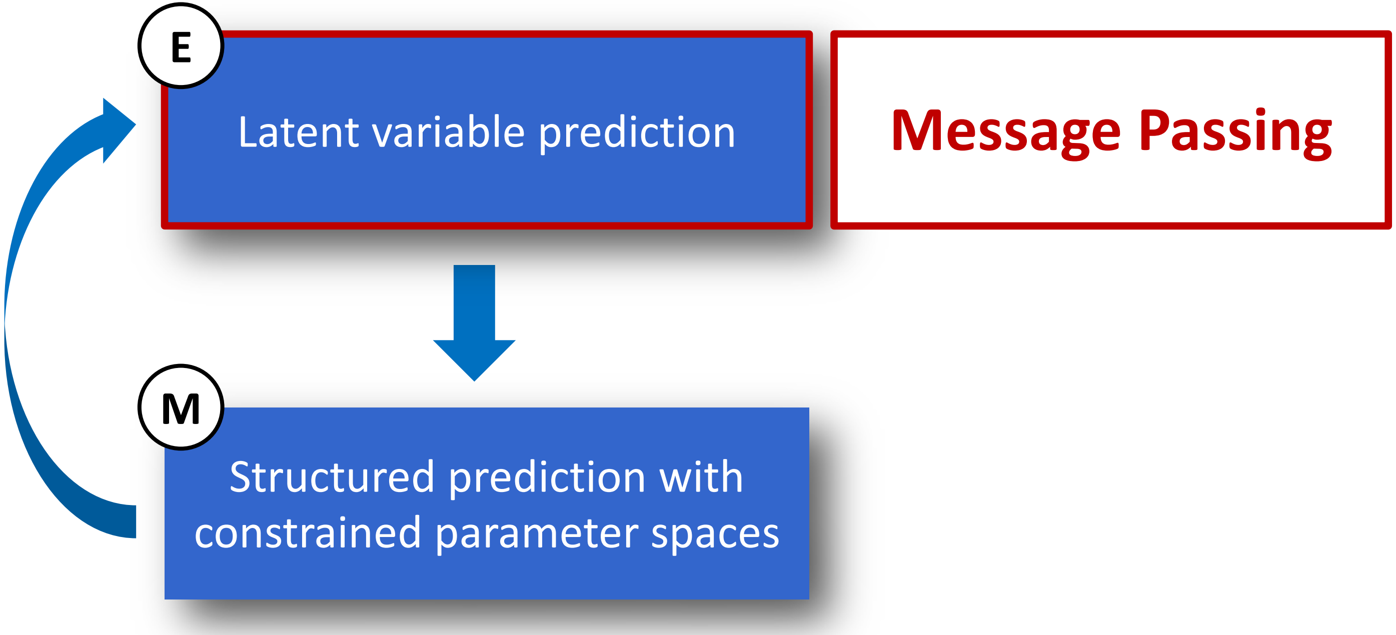
$$L(w) = \sum_m ln\left(\sum_{h_m} exp(\mathbf{w}^T \phi(y_m, h_{,m}) - \ln(Z))\right)$$

[Käser et al., AISTATS 2014]

Constrained structured prediction with latent variables

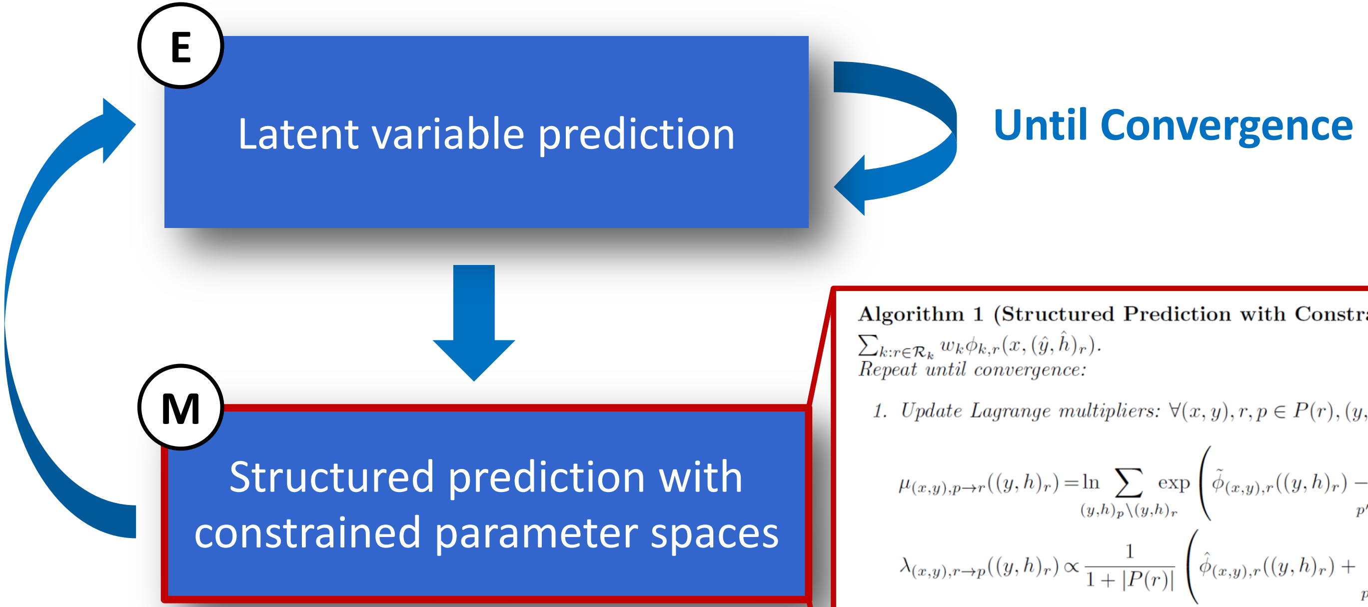


Constrained structured prediction with latent variables



[Schwing et al., ICML 2012]

Constrained structured prediction with latent variables



Algorithm 1 (Structured Prediction with Constrained Parameter Spaces) *Let* $\tilde{\phi}_{(x,y),r}((\hat{y},\hat{h})_r) = \sum_{k:r\in\mathcal{R}_k} w_k \phi_{k,r}(x,(\hat{y},\hat{h})_r)$.
Repeat until convergence:

1. *Update Lagrange multipliers:* $\forall (x,y), r, p \in P(r), (y,h)_r$

$$\mu_{(x,y),p\to r}((y,h)_r) = \ln \sum_{(y,h)_p \setminus (y,h)_r} \exp\left(\tilde{\phi}_{(x,y),r}((y,h)_r) - \sum_{p'\in P(p)} \lambda_{(x,y),p\to p'}((y,h)_{p'}) + \sum_{r'\in C(p)\setminus r} \lambda_{(x,y),r'\to p}((y,h)_{r'})\right)$$

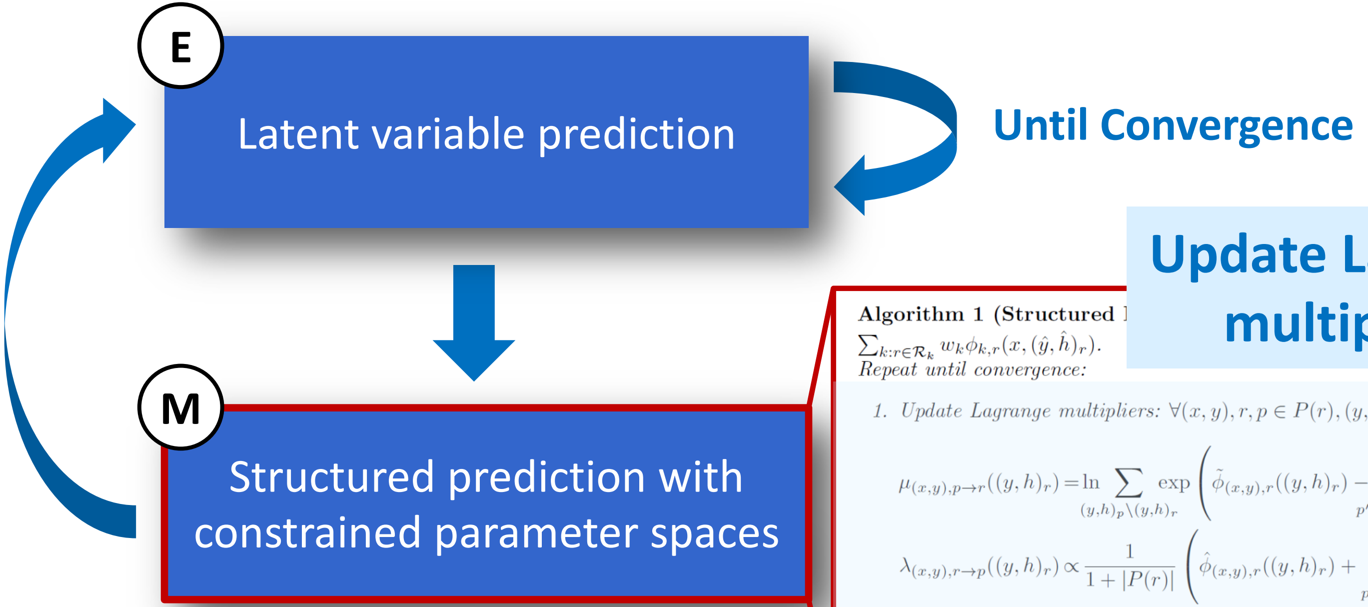
$$\lambda_{(x,y),r\to p}((y,h)_r) \propto \frac{1}{1+|P(r)|}\left(\hat{\phi}_{(x,y),r}((y,h)_r) + \sum_{p'\in P(r)} \mu_{(x,y),p'\to r}((y,h)_r)\right) - \mu_{(x,y),p\to r}((y,h)_r)$$

2. *Perform a gradient step and project the result onto the constraint set* $\mathcal{C}$*:*

$$w \leftarrow P_{\mathcal{C}}[w - \gamma \nabla_w f(\lambda, w)]$$

[Käser et al., AISTATS 2014]

Constrained structured prediction with latent variables



Update Lagrange multipliers

Algorithm 1 (Structured ... **ces)** *Let* $\tilde{\phi}_{(x,y),r}((\hat{y},\hat{h})_r) = \sum_{k:r\in\mathcal{R}_k} w_k \phi_{k,r}(x,(\hat{y},\hat{h})_r)$.
Repeat until convergence:

1. *Update Lagrange multipliers:* $\forall (x,y), r, p \in P(r), (y,h)_r$

$$\mu_{(x,y),p\to r}((y,h)_r) = \ln \sum_{(y,h)_p \setminus (y,h)_r} \exp \left(\tilde{\phi}_{(x,y),r}((y,h)_r) - \sum_{p' \in P(p)} \lambda_{(x,y),p\to p'}((y,h)_{p'}) + \sum_{r' \in C(p)\setminus r} \lambda_{(x,y),r'\to p}((y,h)_{r'}) \right)$$

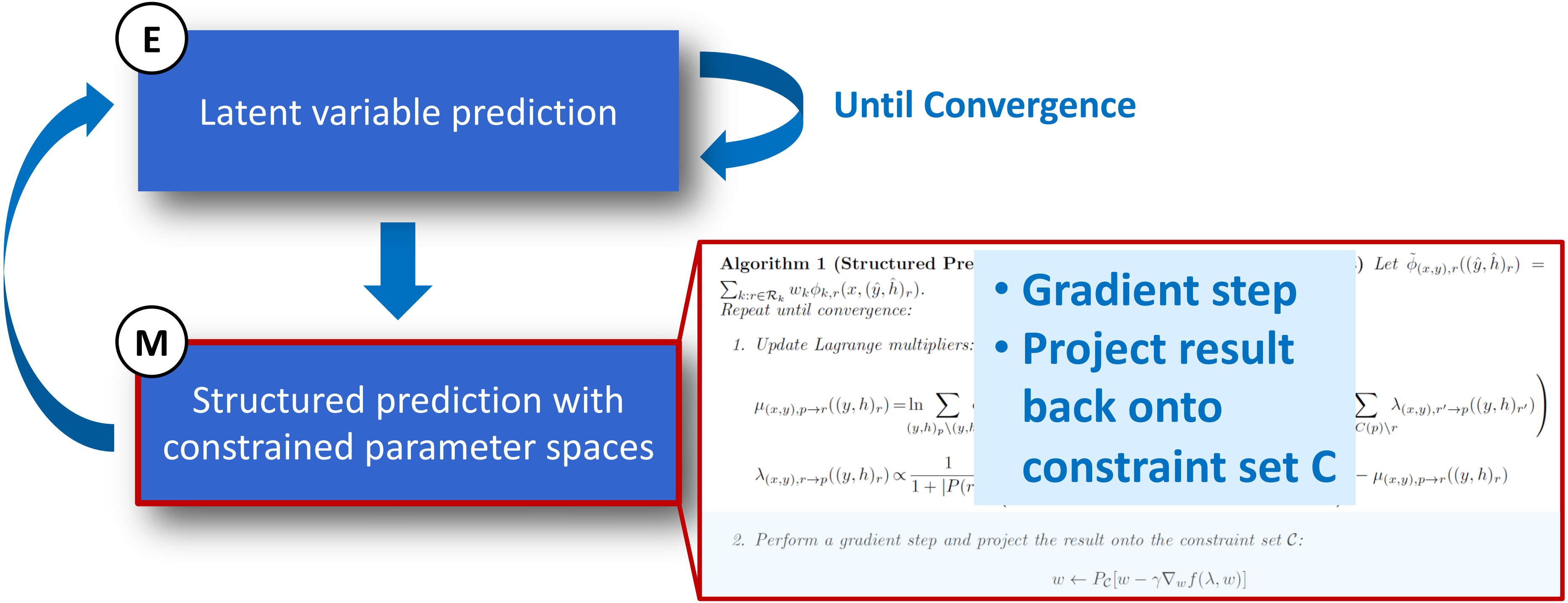
$$\lambda_{(x,y),r\to p}((y,h)_r) \propto \frac{1}{1+|P(r)|} \left(\hat{\phi}_{(x,y),r}((y,h)_r) + \sum_{p' \in P(r)} \mu_{(x,y),p'\to r}((y,h)_r) \right) - \mu_{(x,y),p\to r}((y,h)_r)$$

2. *Perform a gradient step and project the result onto the constraint set* $\mathcal{C}$*:*

$$w \leftarrow P_{\mathcal{C}}[w - \gamma \nabla_w f(\lambda, w)]$$

[Käser et al., AISTATS 2014]

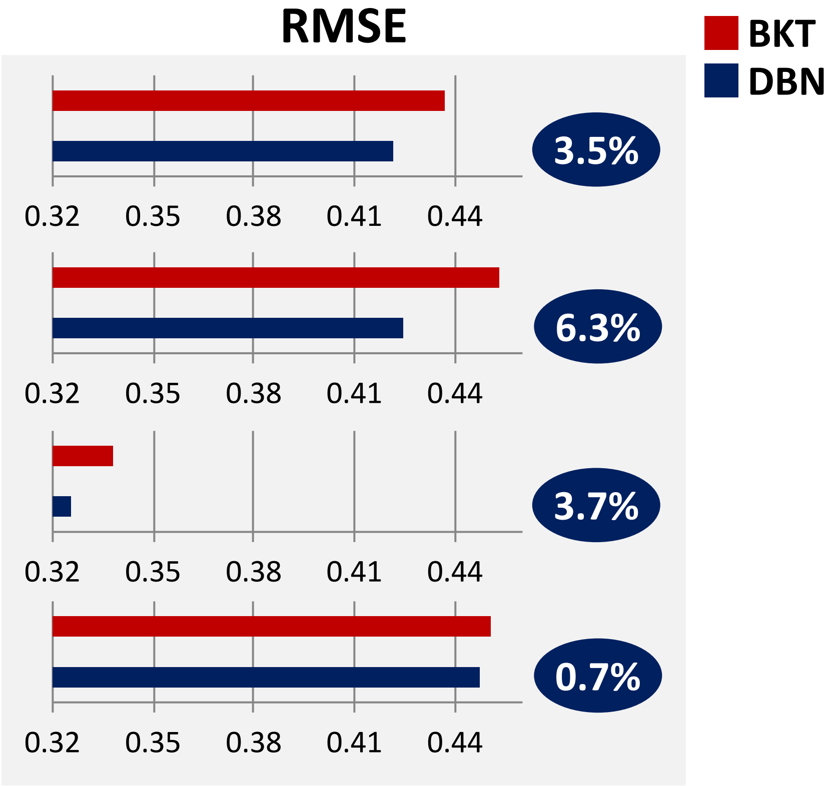
Constrained structured prediction with latent variables



[Käser et al., AISTATS 2014]

DBNs outperform BKT in different learning domains

Learning Domain	Students	Observations
Subtraction	1581	158’100
Physics	77	38’500
Algebra	6043	3’021’500
Spelling	7265	1’453’000



[Käser et al., ITS 2014; Käser et al., IEEE TLT 2017]