Artificial Intelligence *in* Education

Dovan Rai

AI in Education: *80 minutes Tour*

- **AI in Education:** *City Tour (15 min)*
- **Intelligent Tutoring System (ITS):** *Restaurant Stop (30 mins)*
  - **Bayesian Modeling vs. Deep Learning:** *Duel Match (15 mins)*
  - **Causal Modeling:** *Detour (5 mins)*

- **AI in Education- Future Frontiers:** *Mountain View (10 mins)*
- **Holy Grail:** *Discussions (5 mins)*

Scope:
Mostly focus on 'AI in Education Research' not covering commercial/ industry products

Teaching Robots

ozobot

AI in Education: *80 minutes Tour*

- **AI in Education:** *City Tour (15 min)*
- **Intelligent Tutoring System (ITS):** *Restaurant Stop (30 mins)*
  - **Bayesian Modeling vs. Deep Learning:** *Duel Match (15 mins)*
  - **Causal Modeling:** *Detour (5 mins)*

- **AI in Education- Future Frontiers:** *Mountain View (10 mins)*
- **Holy Grail:** *Discussions (5 mins)*

AI in Education: City Tour (Hop-on/ Hop-off)


Educational Psychology
AI in Education
Artificial Intelligence

Create Intelligent Learning Technologies

Educational Psychology

AI in Education

Artificial Intelligence

Based on Learning Theories & Principles

A little History

- *First International Conference in AI in Ed- 1987*
- *First International Conference on Intelligent Tutoring Systems- 1988*
- *First Edition of International Journal of Artificial Intelligence in Education - 1989*

AI in Education: City Tour

- Intellectual Neighborhoods
- Research Clusters
- Conceptual Rivers
- Showcases and Landmarks
- Festivals
- Cliffs and Dead-ends
- New Frontiers
- Holy Grails

Intellectual Neighborhoods

- Cognitive Science
- Learning Science
- Education Psychology
- Human-Computer Interaction
- Inferential Statistics
- Natural Language Processing
- User Modeling
- Intelligent Virtual Agents
- Affective Computing

Research Clusters

- Intelligent Tutoring Systems (ITS)
- Intelligent Pedagogical Agents
- Intelligent Educational Games
- Teachable Agents
- Learning Companions
- Educational Datamining
- Learning Analytics
- Authoring Tools
- Affect Detectors
- Collaborative Conversational Agents
- Automatic Hint Generation
- Open Learner Models
- Metacognition & Self-Regulation
- Learning at Scale

Conceptual Rivers

flowing from **Educational Psychology**

- Constructivist and Constructionist Learning
- Problem based learning
- Situated Learning
- Cognitive Theory of Multimedia Learning
- Working Memory Overload
- Spaced Learning
- Self-Efficacy
- Mastery vs. Performance Goals

Conceptual Rivers

flowing from **Artificial Intelligence**

- Expert Systems
- Rules, Ontologies, Semantics, Knowledge-Base
- Probabilistic Graphical Models
- Prediction, Inference
- Reinforcement Learning
- Deep Learning
- Deep Reinforcement Learning

AIED’s Intelligent Showcases

Cognitive Tutors: Systems that can provide support on every step in a student’s thinking process

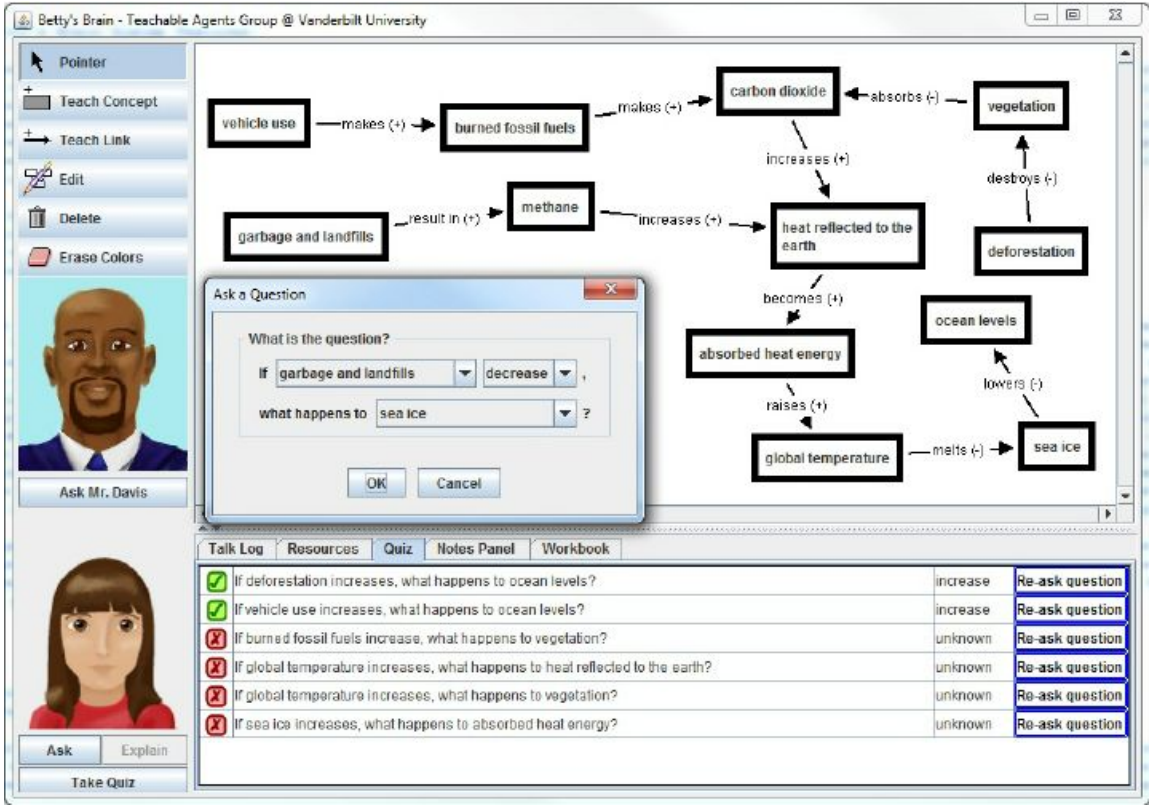
AutoTutor: Systems that can talk with students in natural language

Reasoning mind Genie 2: Systems that model complex teacher and tutor pedagogical strategies

Mathspring: Systems that recognize and respond to differences in student emotion

Betty’s Brain: Simulated students that enable human students to learn by teaching

Betty's Brain: Teachable Agent



Intelligent Pedagogical Agents

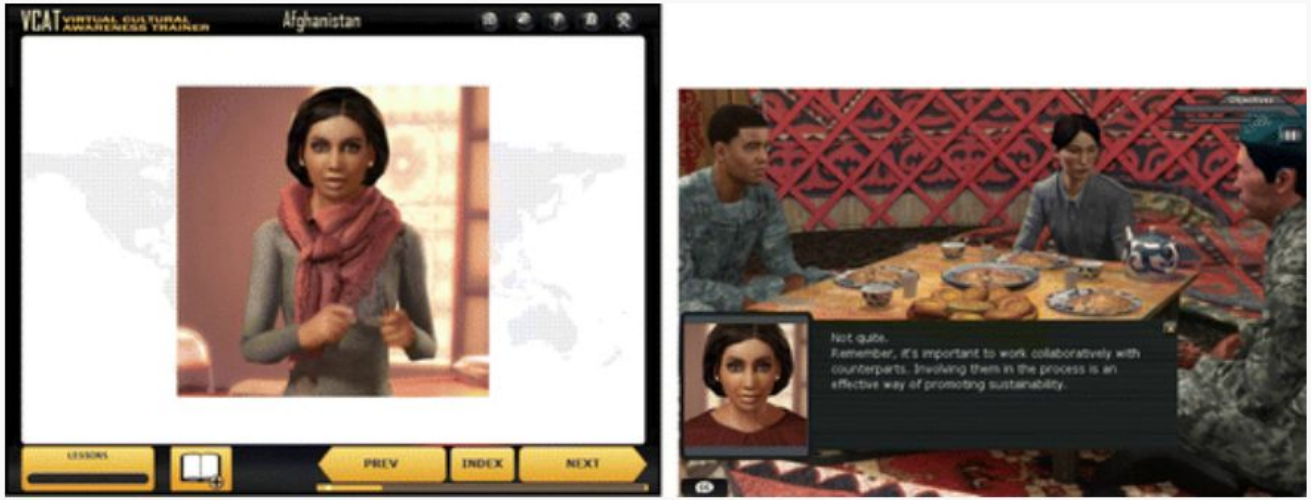


Fig. 1

VCAT Virtual Coach (left) and virtual role-play (right)

Crystal Island: Intelligent Educational Game

Artificial Intelligence in Education

Research Samples

Go to Research Samples Slides

AI in Education: *80 minutes Tour*

- **AI in Education:** *City Tour (15 min)*
- **Intelligent Tutoring System (ITS):** *Restaurant Stop (30 mins)*
  - **Bayesian Modeling vs. Deep Learning:** *Duel Match (15 mins)*
  - **Causal Modeling:** *Detour (5 mins)*

- **AI in Education- Future Frontiers:** *Mountain View (10 mins)*
- **Holy Grail:** *Discussions (5 mins)*

Intelligent Tutoring Systems (ITS)

Adaptive Learning

Learners are different.

They change as they learn.

Learners differ in many ways:

- Knowledge state
- Interest
- Goals
- Affective state
- Strategic behaviors
- Learning styles

Three Loops of Adaptivity

Design-loop adaptivity: data-driven decisions made before and between iterations of system design, in which a course or system is updated based on data about student learning-specifically, data collected with the same system or course.

Task-loop adaptivity: data-driven decisions the system makes to select instructional tasks for the learner.

Step-loop adaptivity: data-driven decisions the system makes in response to individual actions a student takes within an instructional task.

Modeling

How does ITS model the teaching-learning world ?

Three Models

- Domain-Model
- Student-Model
- Tutor/Pedagogical- Model

Quick look into **Domain Modeling**

Production-rule models (Cognitive Tutors)

Constraint-based models (SQL-Tutors)

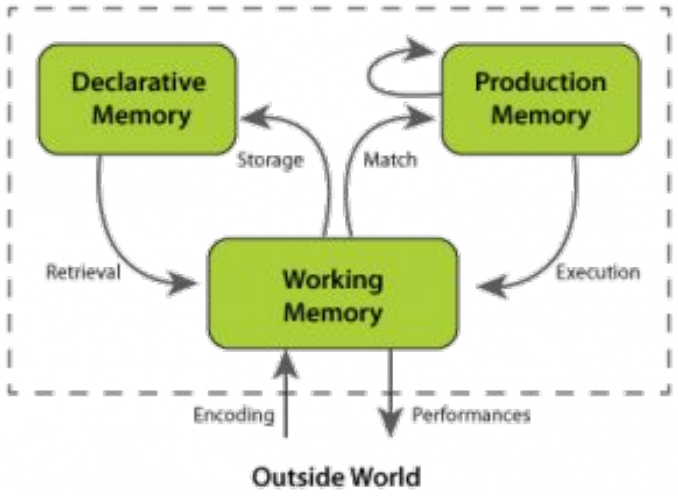
A Comparative Analysis of Cognitive Tutoring and Constraint-Based Modeling

Antonija Mitrovic[1], Kenneth R. Koedinger[2] and Brent Martin[1]

[1]Intelligent Computer Tutoring Group,
University of Canterbury, Christchurch, New Zealand
{tanja,brent}@cosc.canterbury.ac.nz

[2]Human-Computer Interaction Institute, Carnegie Mellon University
koedinger@cmu.edu

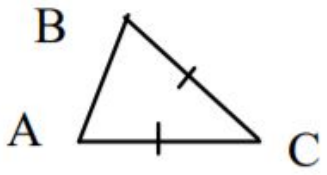
Cognitive Tutors vs. Constraint-based Tutors



ACT-R Theory of Cognition

Represent declarative knowledge as constraints rather than chunks, propositions, or schemas.

No Model for buggy errors



Angle A is 65.
What is angle C?

Two correct production rules:

IF goal is to find an angle in an isosceles triangle ABC and AC = AB and angle A is known
THEN set the value of angle B to A.

IF goal is to find an angle in a triangle ABC and angles A and B are known,
THEN set the value of C to 180-A-B

Buggy production rule:

IF goal is to find an angle in an isosceles triangle ABC
and angle A and C are at the bottom of the triangle and angle A is known
THEN set the value of angle C to A.

Fig. 1. Three production rules for computing the size of an angle

"If <relevance condition> is true, then <satisfaction condition> had better also be true, otherwise something has gone wrong."

C_{r1}: A base angle of an isosceles triangle is known (θ_1),
And the student has calculated the size of the other base angle θ_2
C_{s1}: The size of θ_2 is θ_1

C_{r2}: A base angle of an isosceles triangle is known (θ_1),
And the student has calculated that the size of another angle θ_2 that equals θ_1,
C_{s2}: θ_2 is a base angle

C_{r3}: Two angles of a triangle are known (θ_1 and θ_2),
And the student has calculated the size of the third angle θ_3
C_{s3}: The size of θ_3 is $(180-\theta_1-\theta_2)$

Fig. 2. Three constraints that check whether the size of an angle is correct

Table 2. Comparative analysis of CBM and MT

Property	Model Tracing	Constraint-Based Modeling
Knowledge representation	Production rules (procedural)	Constraints (declarative)
Cognitive fidelity	Tends to be higher	Tends to be lower
What is evaluated	Action	Problem state
Problem solving strategy	Implemented ones	Flexible to any strategy
Solutions	Tend to be computed, but can be stored	One correct solution stored, but can be computed
Feedback	Tends to be immediate, but can be delayed	Tends to be delayed, but can be immediate
Problem-solving hints	Yes	Only on missing elements, but not strategy
Problem solved	'Done' productions	No violated constraints
Diagnosis if no match	Solution is incorrect	Solution is correct
Bugs represented	Yes	No
Implementation effort	Tends to be harder, but can be made easier with loss of other advantages	Tends to be easier, but can be made harder to gain other advantages

Constraint-based modeling :

requires less time and effort to build; less comprehensive feedbacks.

Model-tracing tutors:

require more time and effort to build, more specific feedbacks

Student Modeling

Student Model : Purpose

Estimate

To estimate current level of knowledge of a skill
To estimate learning of an individual

Detect

Detect Boredom/Frustration/Confusion

Predict

Predict student success or failure
Predict Engagement
Predict Student Learning

Student Modeling : Algorithms

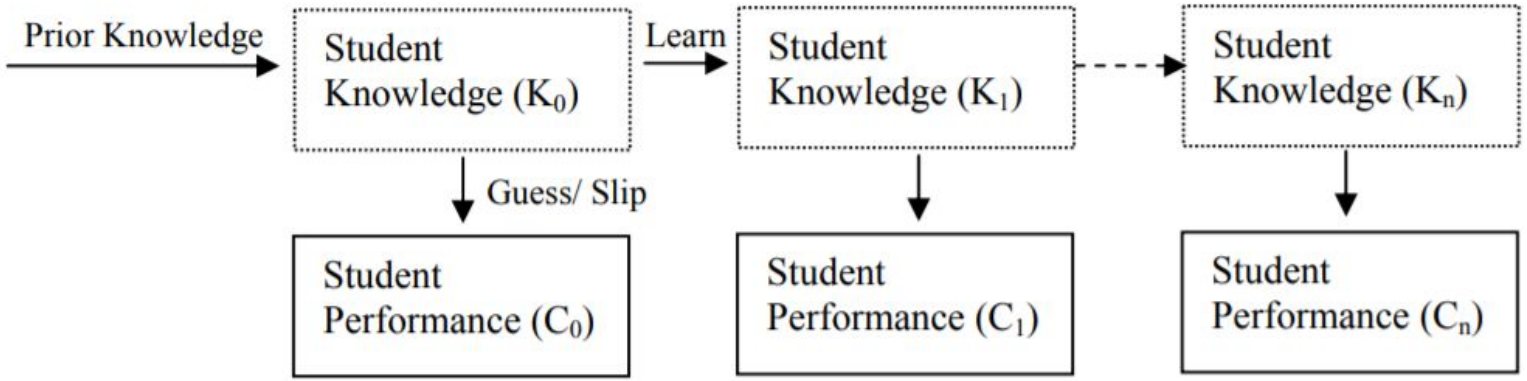
- **Bayesian Knowledge Tracing (BKT)**
- **Performance Factors Analysis (PFA)**
- **Learning Factors Analysis (LFA)**
- **Partially Observable Markov Decision Processes (POMDPs)**
- **Item Response Theory (IRT)**
- **Knowledge Space Theory (KST)**

Student Modeling

- **Bayesian Knowledge Tracing (BKT)**
- **Performance Factors Analysis (PFA)**
- **Learning Factors Analysis (LFA)**
- **Partially Observable Markov Decision Processes (POMDPs)**
- **Item Response Theory (IRT)**
- **Knowledge Space Theory (KST)**

Bayesian Knowledge Tracing (BKT)

Dynamic Bayesian Networks (DBN)

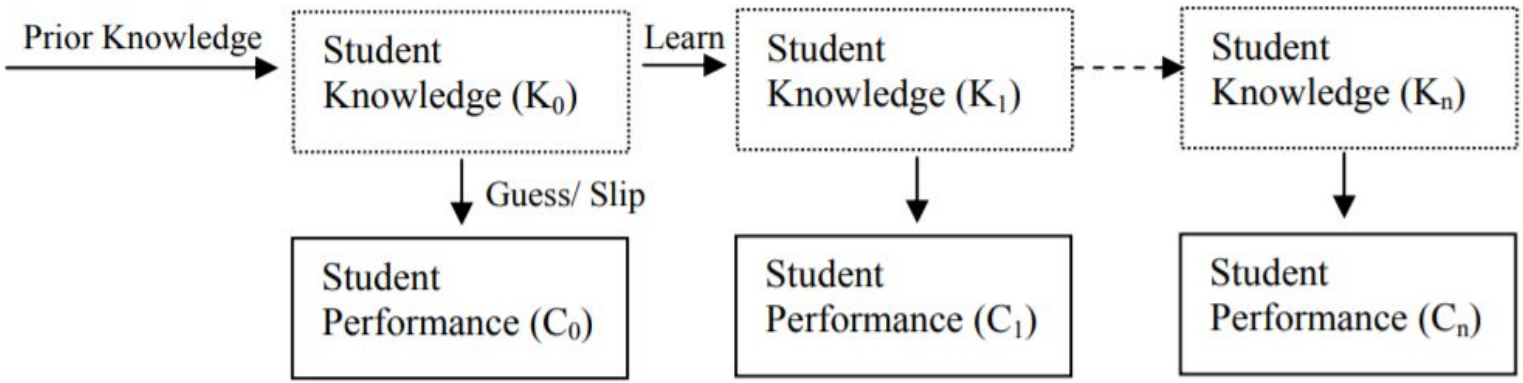


$$Prior\ Knowledge = Pr\ (K_0 = True)$$

$$Guess = Pr\ (C_n = True \mid K_n = False)$$

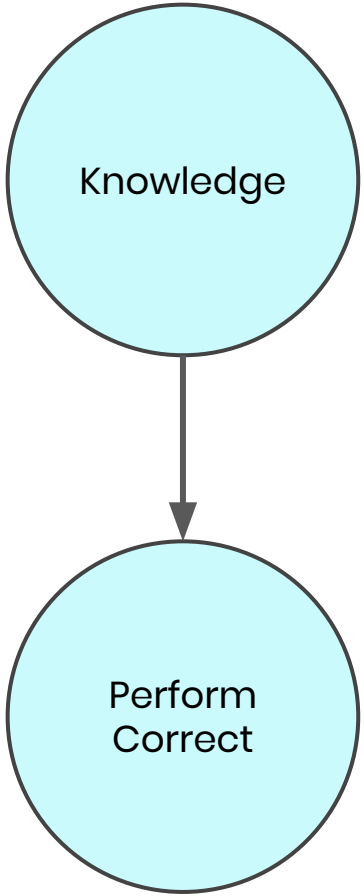
$$Slip = Pr\ (C_n = False \mid K_n = True)$$

$$Learning\ rate = Pr\ (K_n = True \mid K_{n-1} = False\ )$$



a qualitative part (expressing the relationships between the variables with semantics based on the concept of conditional independence) and

a quantitative part (the numerical values of a set of conditional probability distributions).



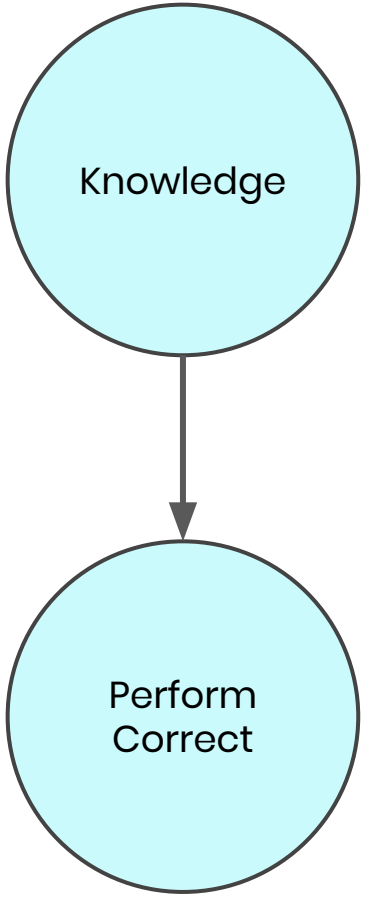
Knowledge

True	False
a	b

Perform Correct

Knowledge	True	False
Knowledge =True	c	d
Knowledge =False	e	f

Graph



Conditional Probability Tables (CPT)

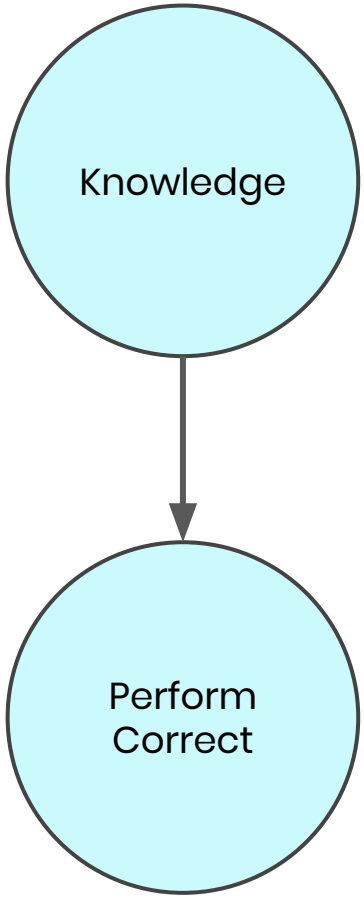
Knowledge

True	False
a	b

Perform Correct

Knowledge	True	False
Knowledge =True	c	d
Knowledge =False	e	f

Ideal World



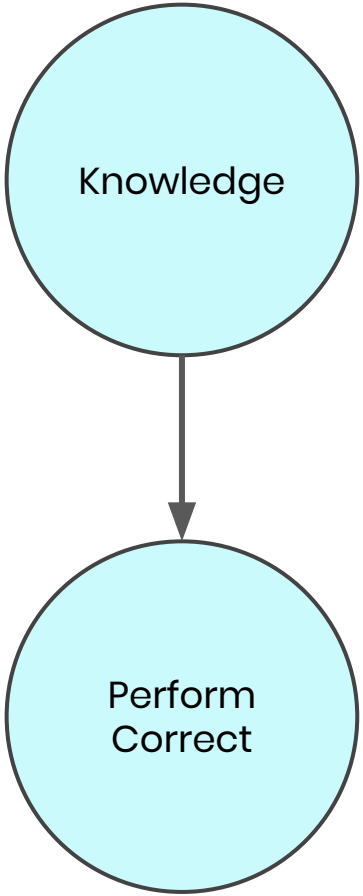
Knowledge

True	False
0.5	0.5

Perform Correct

Knowledge	True	False
Knowledge =True	1	0
Knowledge =False	0	1

Real World: Guessing and Slipping



Knowledge

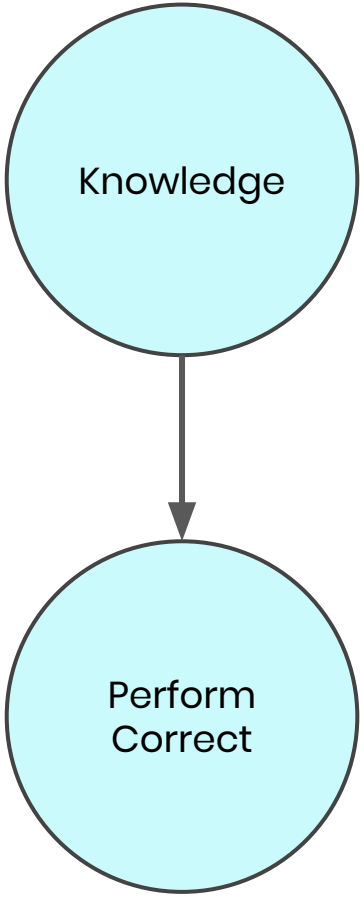
True	False
0.5	0.5

Perform Correct

Knowledge	True	False
Knowledge =True	0.8	0.2
Knowledge =False	0.3	0.7

Model Student A:

High knowledge
but
Careless



Knowledge

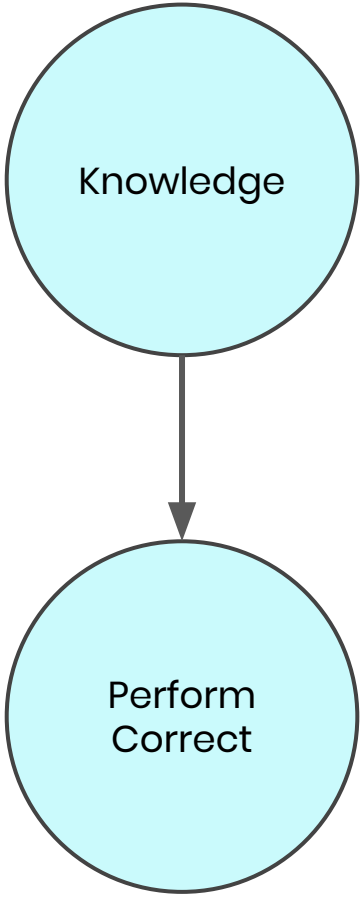
True	False
0.9	0.1

Perform Correct

Knowledge	True	False
Knowledge =True	0.6	0.4
Knowledge =False	0.3	0.7

Model Student B:

Low knowledge;
Clever at guessing



Knowledge

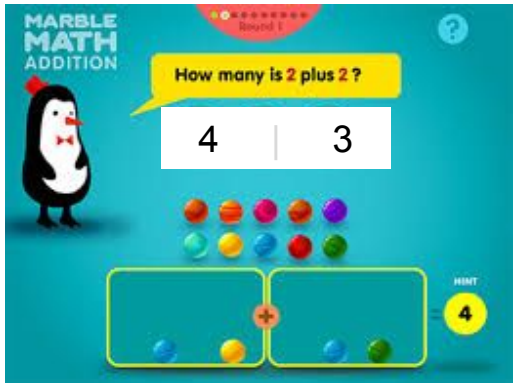
True	False
0.3	0.7

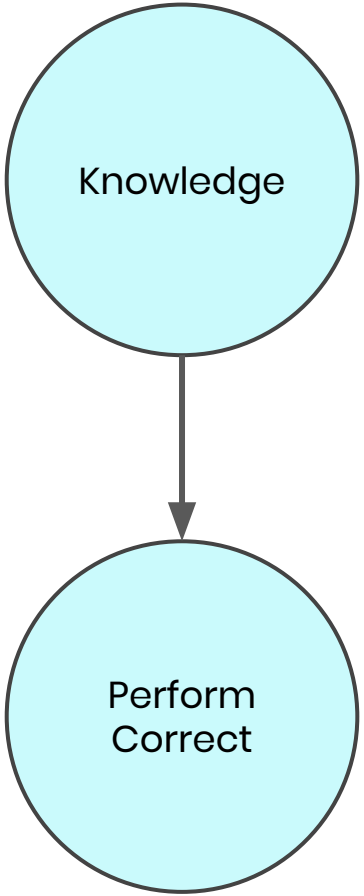
Perform Correct

Knowledge	True	False
Knowledge =True	0.8	0.2
Knowledge =False	0.6	0.3

Tutor A:

Only two options as multiple choice





Knowledge

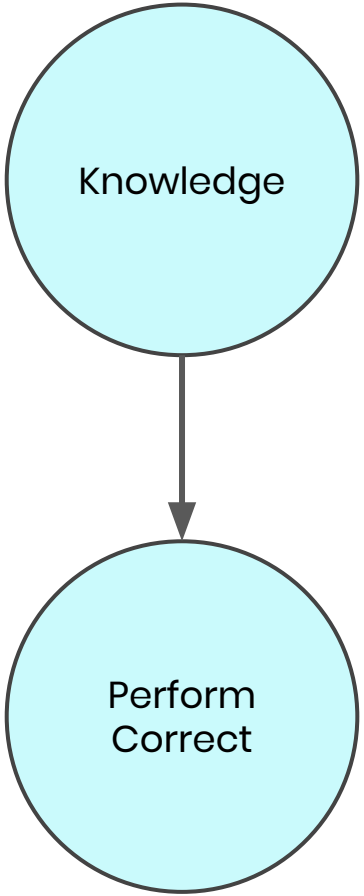
True	False
0.5	0.5

Perform Correct

Knowledge	True	False
Knowledge =True	0.8	0.2
Knowledge =False	0.5	0.5

Tutor B:

Confusing Interface



Knowledge

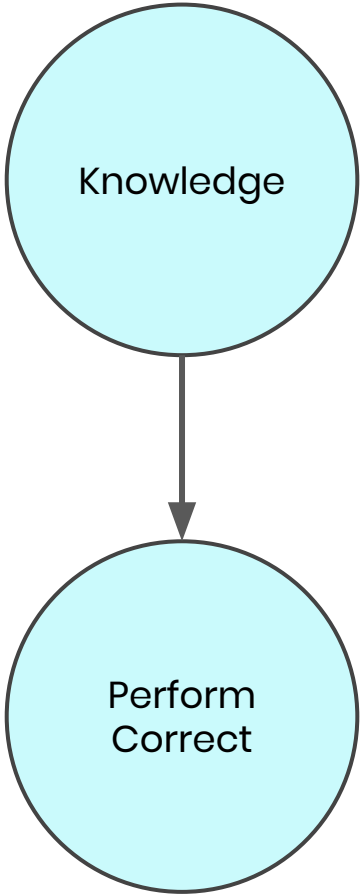
True	False
0.5	0.5

Perform Correct

Knowledge	True	False
Knowledge =True	0.4	0.6
Knowledge =False	0.3	0.7

Classroom A:

High Knowledge Students



Knowledge

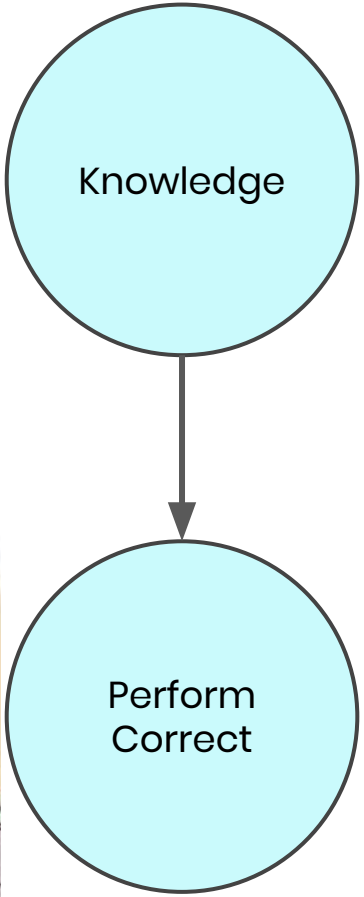
True	False
0.9	0.1

Perform Correct

Knowledge	True	False
Knowledge =True	0.8	0.2
Knowledge =False	0.3	0.7

Classroom B:

Cheaters



Knowledge

True	False
0.5	0.5

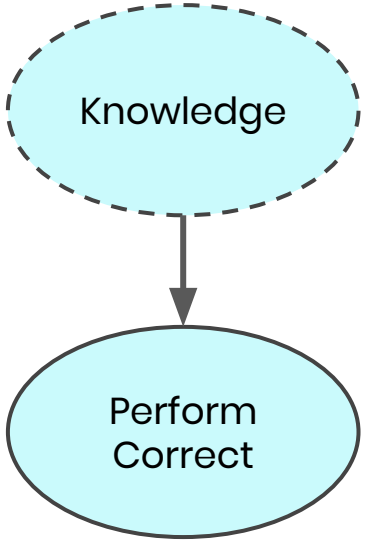
Perform Correct

Knowledge	True	False
Knowledge =True	0.8	0.2
Knowledge =False	0.7	0.3

Inference in Bayesian Network

Bayesian Inference

Bayes comes into Bayesian Network.



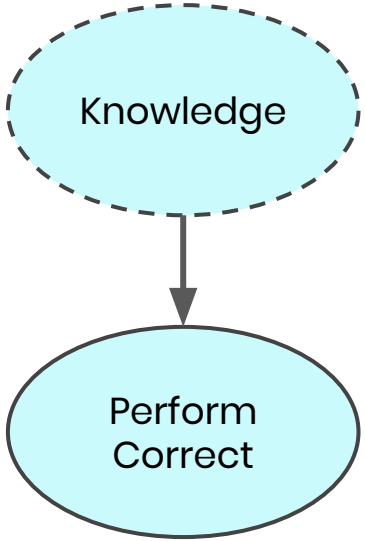
Knowledge

True	False
0.5	0.5

Perform Correct

Knowledge	True	False
True	0.8	0.2
False	0.3	0.7

This is a probabilistic graphical representation of knowledge-performance model.



Knowledge

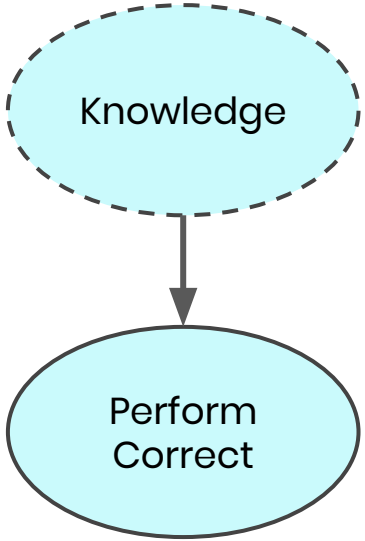
True	False
0.5	0.5

Perform Correct

Knowledge	True	False
True	0.8	0.2
False	0.3	0.7

This is a probabilistic graphical representation of knowledge-performance model.

It tells us how the performance would be like given knowledge.



Knowledge

True	False
0.5	0.5

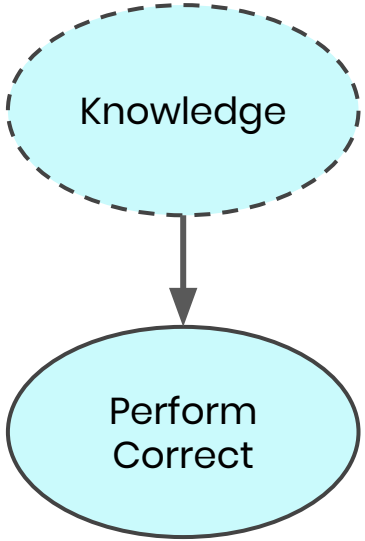
Perform Correct

Knowledge	True	False
True	0.8	0.2
False	0.3	0.7

This is a probabilistic graphical representation of knowledge-performance model.

It tells us how the performance would be like given knowledge.

But in real-world, we can see performance but cannot see knowledge.



Knowledge

True	False
0.5	0.5

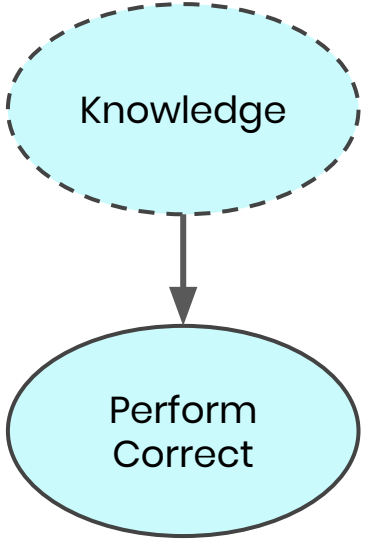
Perform Correct

Knowledge	True	False
True	0.8	0.2
False	0.3	0.7

This is a probabilistic graphical representation of knowledge-performance model.

It tells us how the performance would be like given knowledge.

But in real-world, we can see performance but cannot see knowledge. Knowledge is hidden (latent)



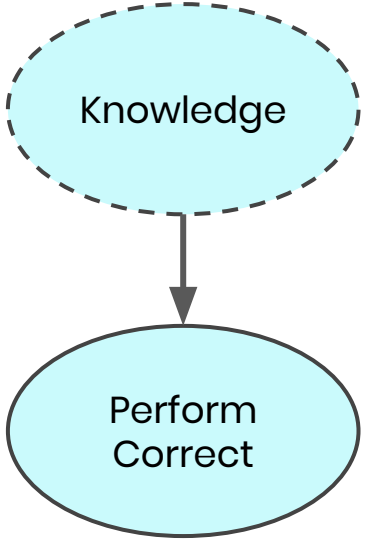
Knowledge

True	False
0.5	0.5

Perform Correct

Knowledge	True	False
True	0.8	0.2
False	0.3	0.7

This is where Bayesian Inference comes.
Using the Bayes Theorem, we can deduce knowledge from performance.



Knowledge

True	False
0.5	0.5

Perform Correct

Knowledge	True	False
True	0.8	0.2
False	0.3	0.7

P(Know) → CPT
P(Correct) → Observable

P(Correct | Know) → CPT
P(Know | Correct) = ?

$$P(\text{Know} \mid \text{Correct}) = \frac{P(\text{Correct} \mid \text{Know}) * P(\text{Know})}{P(\text{Correct})}$$

Learning in Bayesian Networks

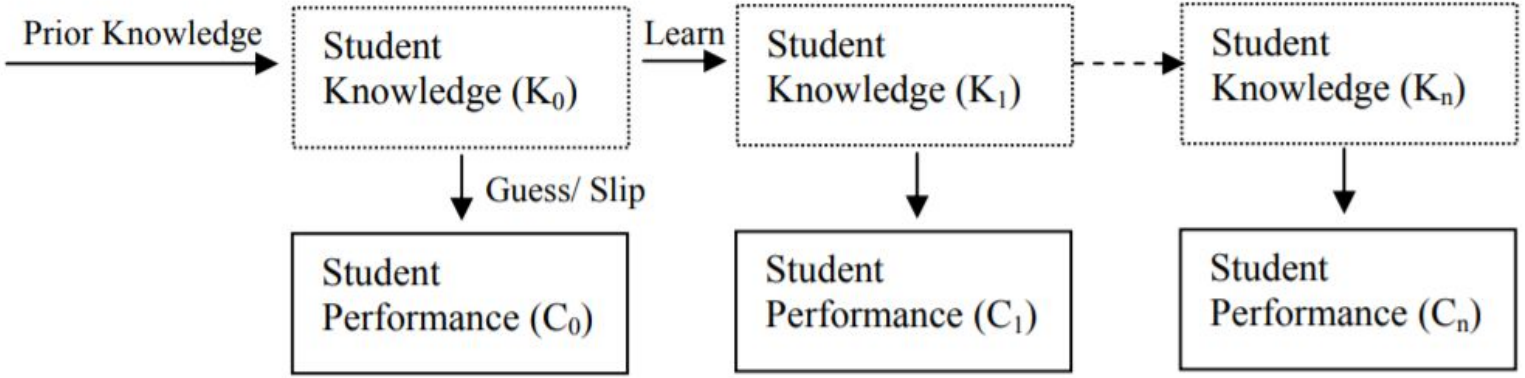
Maximum Likelihood Estimation (ML)

Expectation-Maximization (EM)

Gradient Descent

Markov Chain Monte Carlo (MCMC)

Bayesian Knowledge Tracing (BKT)



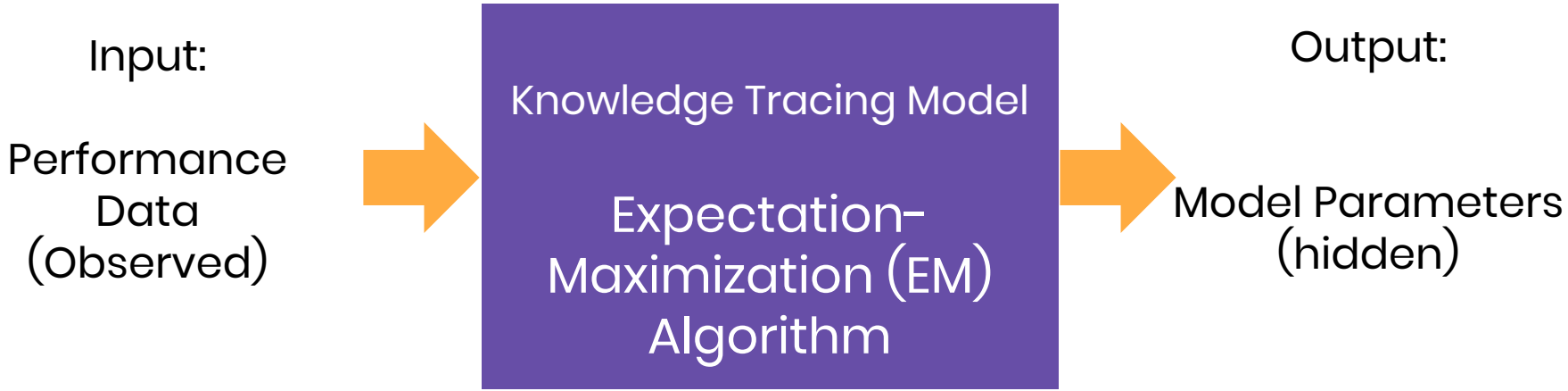
$$Prior\ Knowledge = Pr\ (K_0 = True)$$

$$Guess = Pr\ (C_n = True \mid K_n = False)$$

$$Slip = Pr\ (C_n = False \mid K_n = True)$$

$$Learning\ rate = Pr\ (K_n = True \mid K_{n-1} = False\ )$$

Bayesian Knowledge Tracing (BKT)



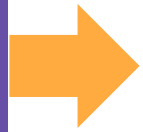
Bayesian Knowledge Tracing (BKT)

Input:
Performance Data
(Observed)

Student	Time	Performance
A	a	0
A	b	1
B	a	1
B	b	1
C	a	0

Knowledge Tracing Model

Expectation-Maximization (EM) Algorithm



Output:
Model Parameters
(hidden)

Parameters	Value
Prior Knowledge	0.52
Learning	0.09
Guess	0.39
Slip	0.16

What do we do with these parameters?

Helps to predict next performance.

How is the student going to perform in next question?

Shall we give more difficult problem?

Shall we offer help?

Understand student learning behavior ??

This is where it gets tricky !

Limitations of Learning in Bayesian Networks.

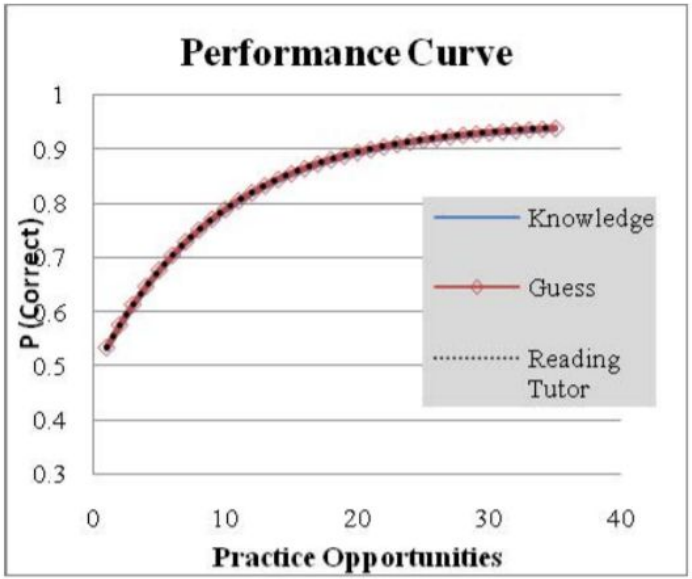
Multiple global maxima.

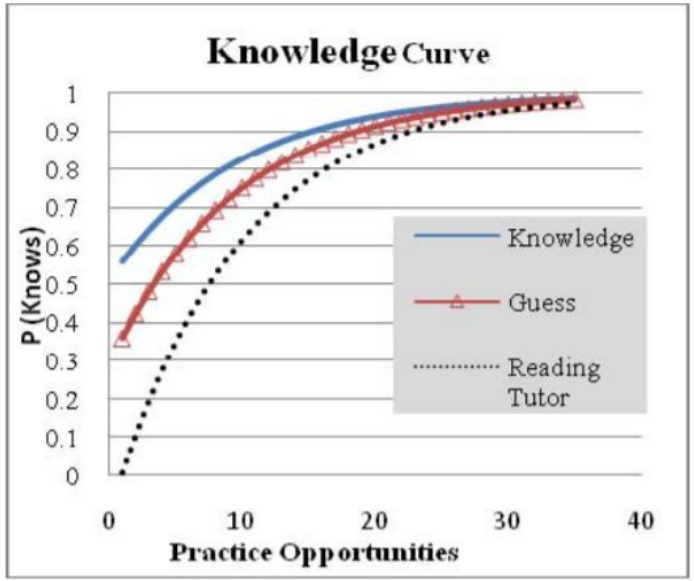
In EM, we need to start from some priors.

Final parameter estimations are sensitive to those initial values.

Table 1. Parameters for three hypothetical knowledge tracing models

Parameter	Model		
	Knowledge	*Guess*	*Reading Tutor*
Prior Knowledge	0.56	0.36	0.01
Learning	0.1	0.1	0.1
Guess	0.00	0.3	0.53
Slip	0.05	0.05	0.05





Prediction vs. Inference

Predictive Accuracy

vs.

Parameter plausibility/ Interpretability

Using Dirichlet priors to improve model parameter plausibility

Dovan Rai, Yue Gong, and Joseph E. Beck
{dovan, ygong, josephbeck}@wpi.edu
Computer Science Department, Worcester Polytechnic Institute

In Expectation Maximization (EM) algorithm, we have to start with some initial value of the parameter,

and final parameter estimations are sensitive to those initial values.

Proposed solution: Dirichlet Priors

Dirichlet prior is an approach used to initialize conditional probability tables when training a Dynamic Bayesian network. Dirichlet distributions are specified by a pair of numbers (α,β).

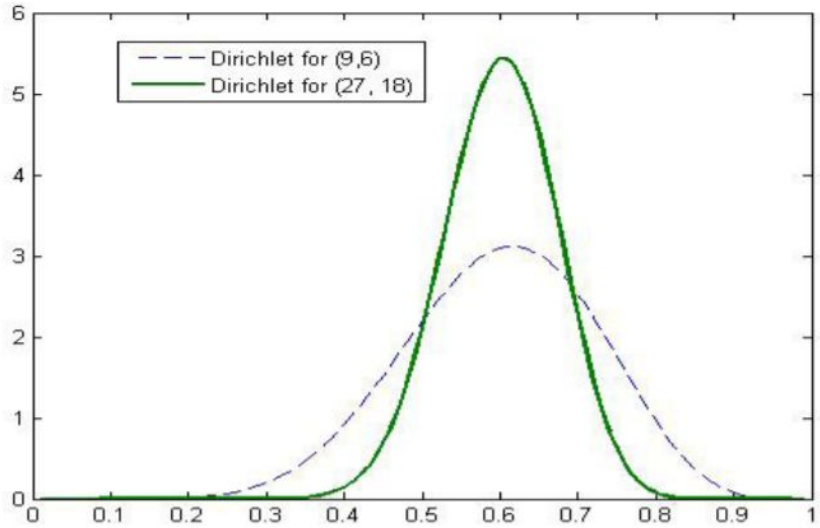


Figure 3. Sample Dirichlet Distributions demonstrating decreasing variance

Dirichlets enable researchers to not only specify the most likely value for a parameter but the confidence in the estimate.

Researchers can use Dirichlets to set confidence on priors.

If the variance is less, we are surer about the priors, whereas if the variance is high, we are less sure about the priors.

Each of the four parameters will not only have different mean values, but different degrees of certainty.

A group of students start with similar incoming knowledge but have variable learning.

Then Dirichlet prior will set higher confidence in students' prior knowledge (e.g.: $\alpha, \beta = 20, 34$)

but lower confidence in students' learning (e.g.: $\alpha, \beta = 1, 4$).

As a result, prior knowledge parameter estimation will be more biased towards prior or distribution's mean whereas learning will have more tendency to move away from prior value.

Where do you get those (α, β) priors??

An automatic approach for selecting priors

1. Initialize EM with fixed priors from our rough estimates of the domain. Then use EM to estimate the model parameters for each skill in the domain

2. For all four parameters (guess, slip, K_0, learning)

- Compute the mean (μ) and variance (σ^2) of the parameter estimates
- Weight the mean and variance by the number of cases (n) of each skill.
  Specifically, for each parameter P of skill i,
  - $weight_i = \sqrt{n_i}$
  - $\mu' = \sum Pi * weight_i / \sum weight$
  - $\sigma^{2\prime} = \sum weight_i * (P_i - \mu_p)^2 / \sum weight$
- Select α and β to generate a Dirichlet with the same mean and variance as the estimates
  Specifically, solve for α and β such that:
  - $\alpha = (\mu'^2 / \sigma^{2\prime}) * (1- \mu') - \mu'$
  - $\beta = \alpha *((1/ \mu')-1)$

3. We now have one Dirichlet distribution described by (α, β) for each of the four parameters

4. Reestimate two kinds of knowledge tracing models: a fixed prior model with initial value of μ' and Dirichlet prior model using the (α, β) pairs.

AI to enhance human learning

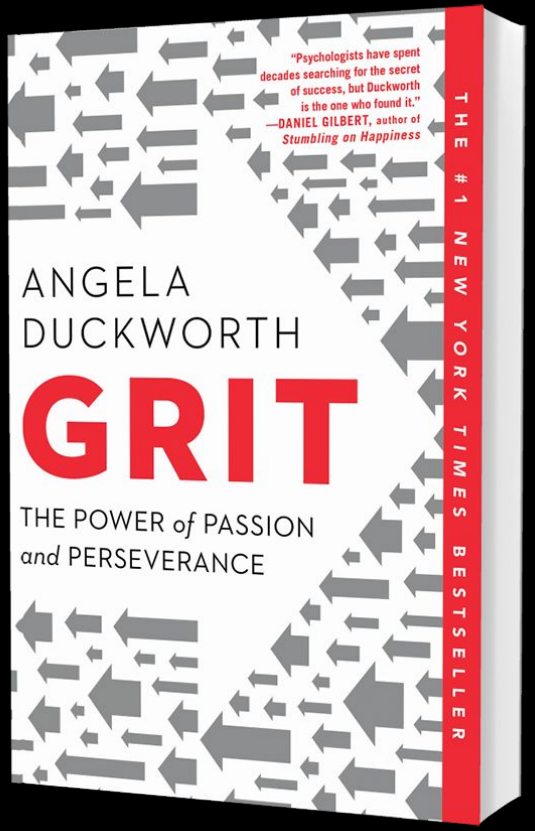
Educational Datamining

AI to understand human learning

Self-Discipline: Impact on students' knowledge, performance and behavior

Dovan Rai, Joseph E. Beck and Yue Gong
Computer Science Department, Worcester Polytechnic Institute


TED Talks
EDUCATION
TED


"Psychologists have spent decades searching for the secret of success, but Duckworth is the one who found it."
—DANIEL GILBERT, author of Stumbling on Happiness
ANGELA DUCKWORTH
GRIT
THE POWER of PASSION and PERSEVERANCE
THE #1 NEW YORK TIMES BESTSELLER

1. Does self-discipline have a significant impact when it comes to knowledge acquisition and performance within ITS?
2. Does the ITS community need to consider self-discipline while designing ITS?

Survey of students

Table 1 Self-discipline groups

Self-discipline groups	Mean Self-discipline	Mean Inconsistency
High self-discipline (N=45)	16.3	0.4
Medium self-discipline (N=45)	7.9	0.5
Low self-discipline (N=44)	-4.4	0.6
Inconsistent (N=37)	0.6	1.7

Self-disciplined students perform better.

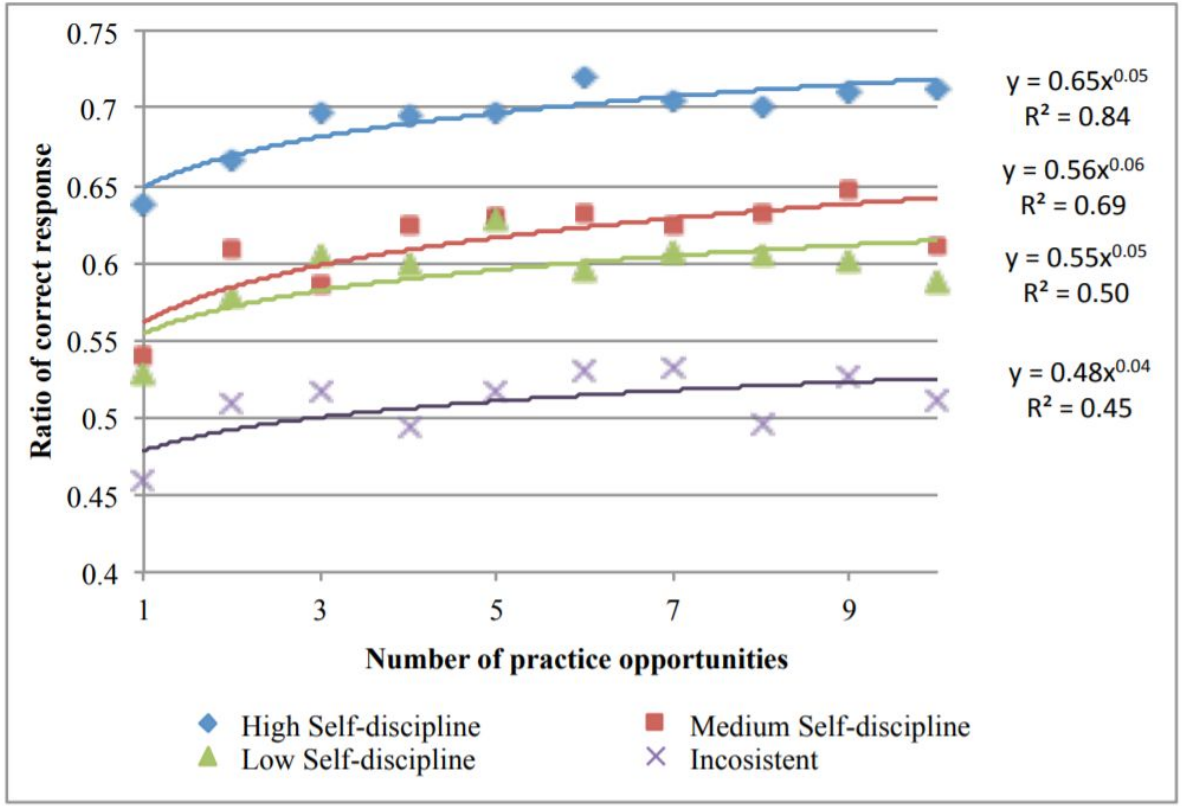


Figure 1 Performance plot across practice opportunities

Are they also more consistent?

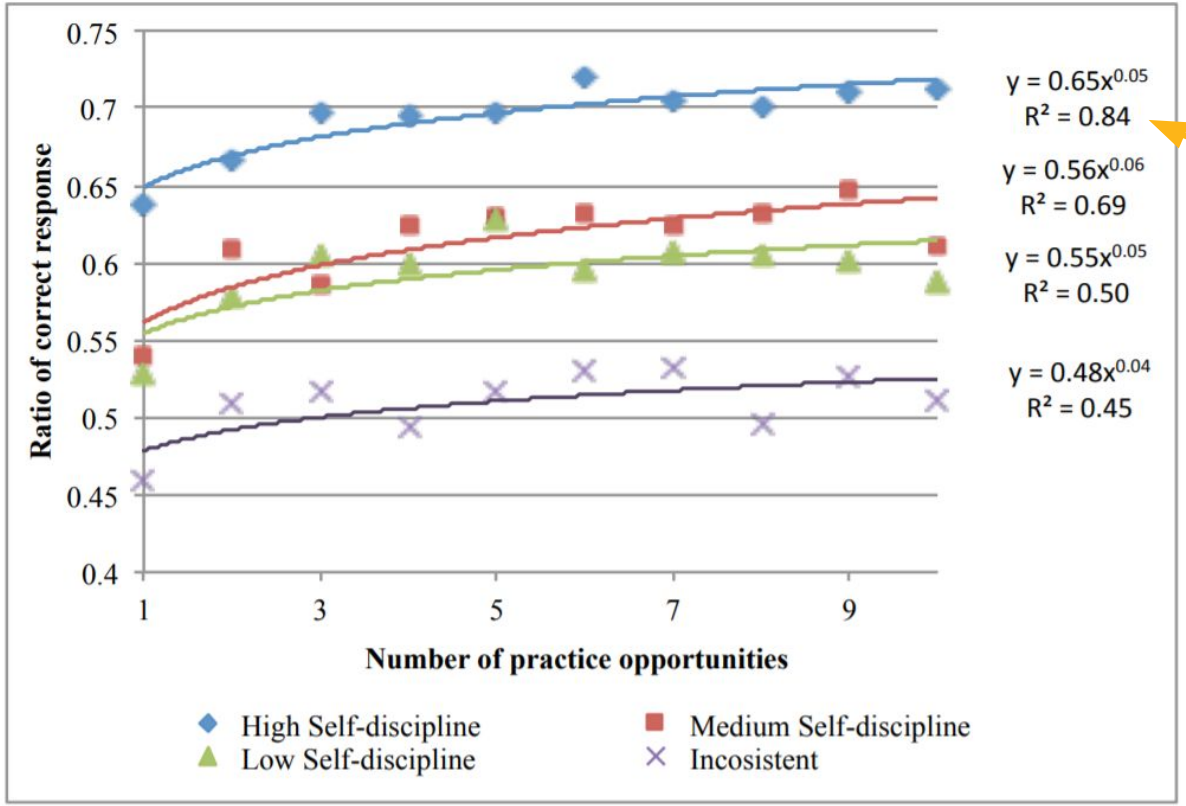


Figure 1 Performance plot across practice opportunities

Are they also more consistent?

- High self-discipline: 18,850
- Medium self-discipline: 13,465
- Low self-discipline: 14,430
- Inconsistent: 11,040

Curves generated from larger numbers of data points will tend to be smoother and exhibit more lawful learning.

Are they also more consistent?

- High self-discipline: 18,850
- Medium self-discipline: 13,465
- Low self-discipline: 14,430
- Inconsistent: 11,040

Table 2 Mean Bootstrapped R^2 and MSE values

	R^2 value (95% CI) [higher is better]	MSE(normalized) [lower is better]
High	0.60 ± 0.009	1
Medium	0.53± 0.01	3.9
Low	0.38 ± 0.01	884
Inconsistent	0.30 ± 0.01	2.1

We are interested in which group of students improves more quickly, but there are several confounds to consider:

- Identifiability of knowledge tracing parameters: perhaps viewing theoretic performance after taking into account all four model parameters will paint a clearer picture?
- Ceiling effect: as performance gets higher, additional improvement becomes more difficult.
- Contamination in the groups: perhaps some students systematically over- or under-reported their self-discipline.

Testing Identifiability

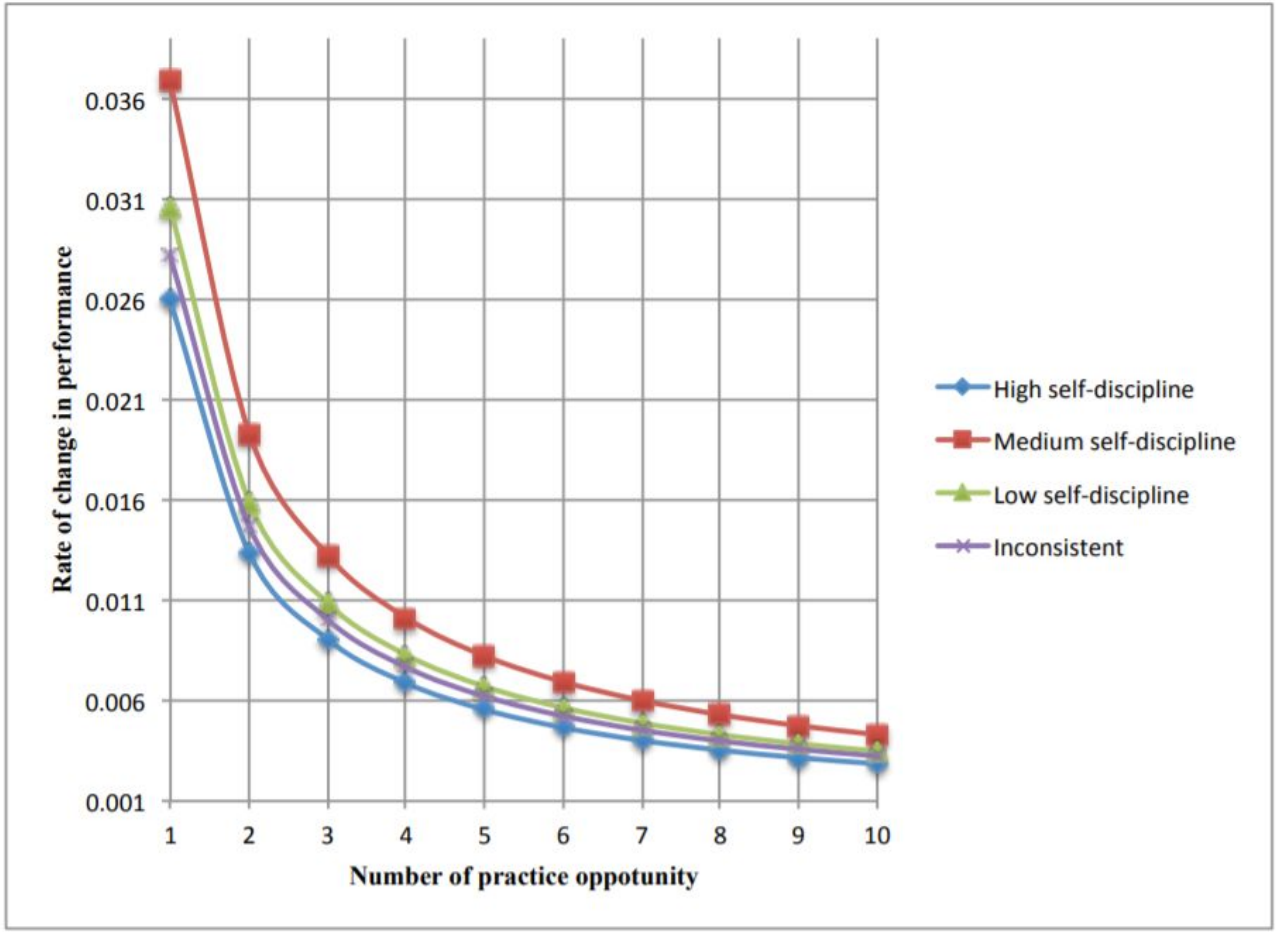


Figure 4 Rate of change in performance based on real performance plot

Checking for a ceiling effect

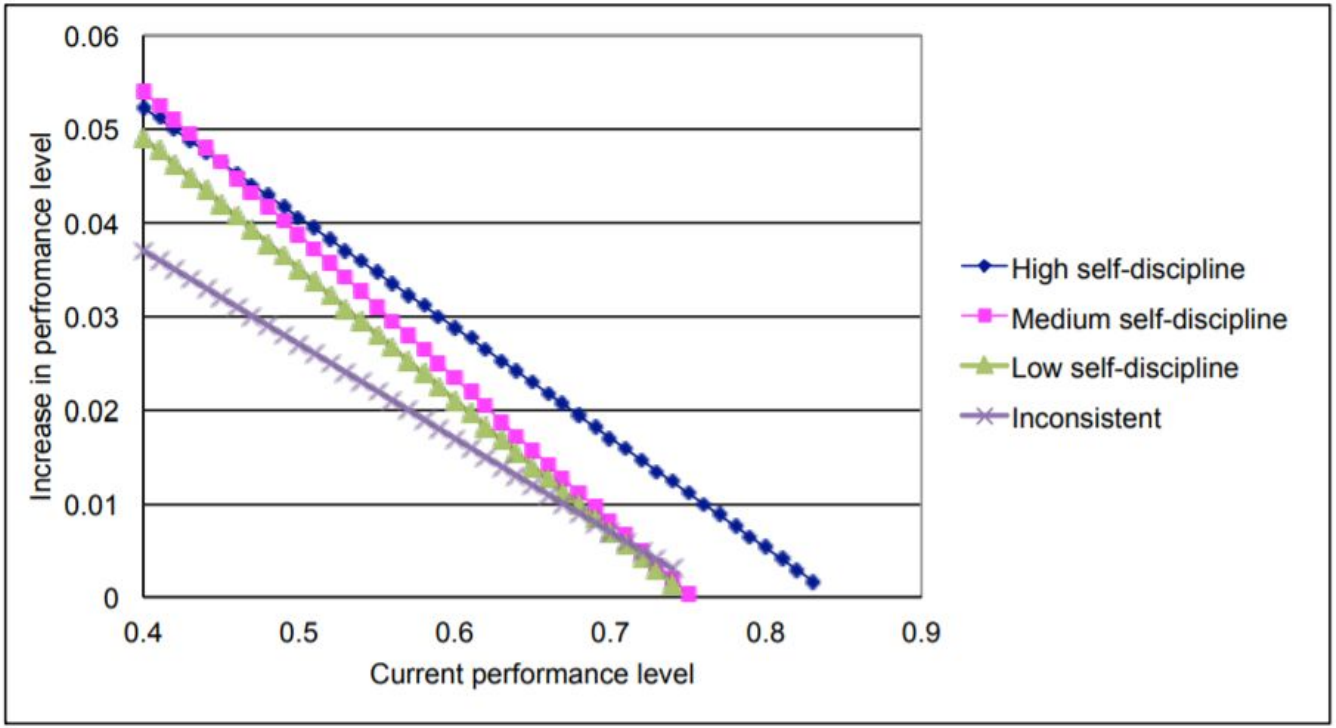


Figure 5 Rate of change in performance across base performance level

Does self-discipline have effect on student behavior?

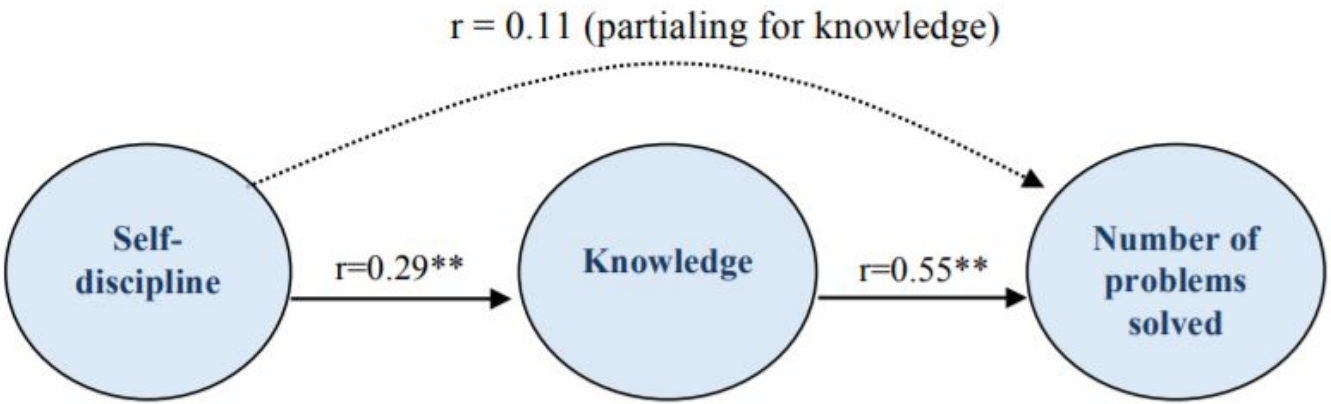


Figure 6 Self-discipline's impact on number of problems solved

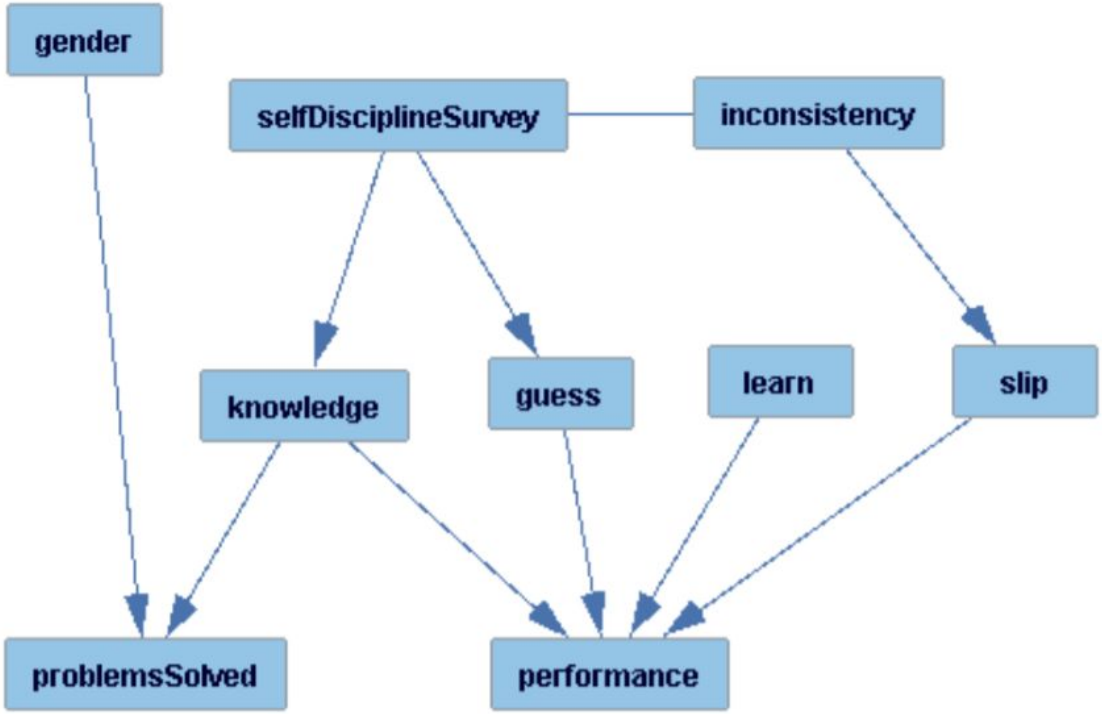


Figure 7 Causal model

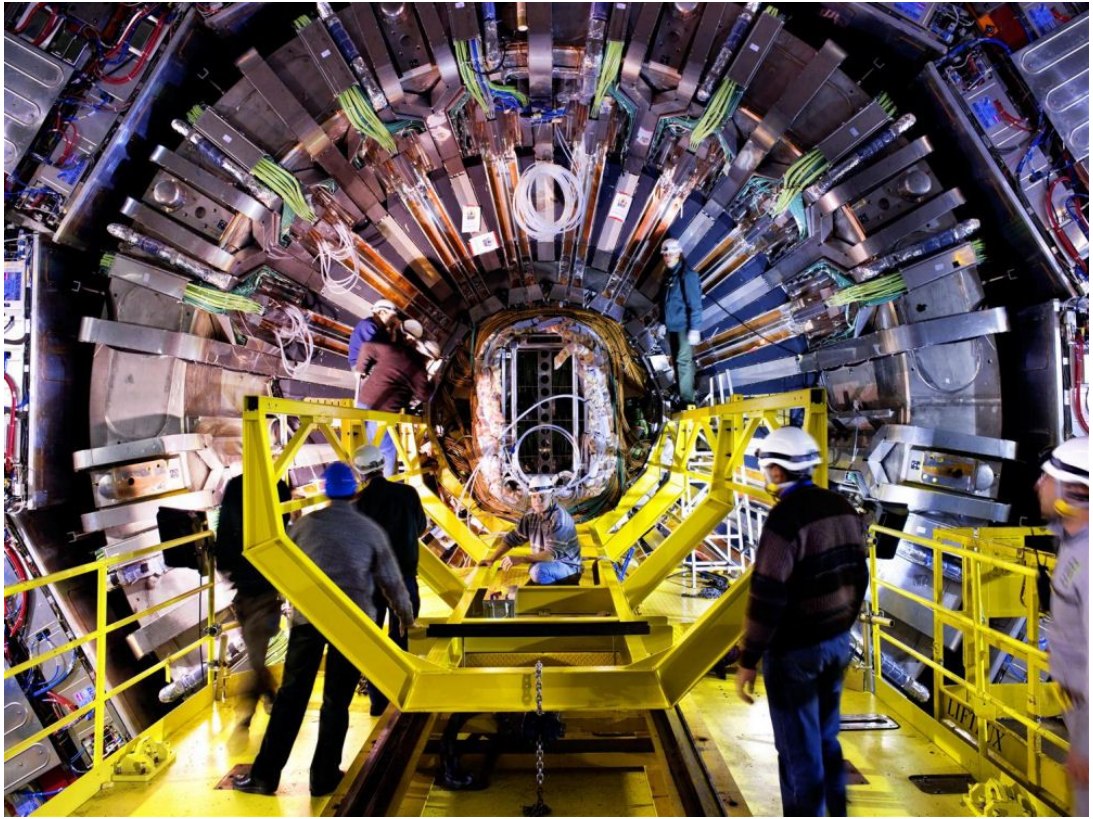
AI in Education: *80 minutes Tour*

- **AI in Education:** *City Tour (15 min)*
- **Intelligent Tutoring System (ITS):** *Restaurant Stop (30 mins)*
  - **Bayesian Modeling vs. Deep Learning:** *Duel Match (15 mins)*
  - **Causal Modeling:** *Detour (5 mins)*

- **AI in Education- Future Frontiers:** *Mountain View (10 mins)*
- **Holy Grail:** *Discussions (5 mins)*

AIED: Land of Bayesians

Bayesian Knowledge Tracing (BKT)



AIED: Land of Bayesians

Knight of Deep Learning

Knight of Deep Learning

Intimidating Reputation:

In one domain after the next, deep learning has achieved gains over traditional approaches.

Deep learning discards hand-crafted features in favor of representation learning, and deep learning often ignores domain knowledge and structure in favor of massive data sets and general architectural constraints on models

Deep Knowledge Tracing

Chris Piech*, **Jonathan Bassen***, **Jonathan Huang***‡, **Surya Ganguli***,
Mehran Sahami*, **Leonidas Guibas***, **Jascha Sohl-Dickstein***†
*Stanford University, †Khan Academy, ‡Google
{piech,jbassen}@cs.stanford.edu, jascha@stanford.edu,

Advances in Neural Information Processing Systems 28 (NIPS 2015)

Cited by 253

Bayesian Knowledge Tracing (BKT)

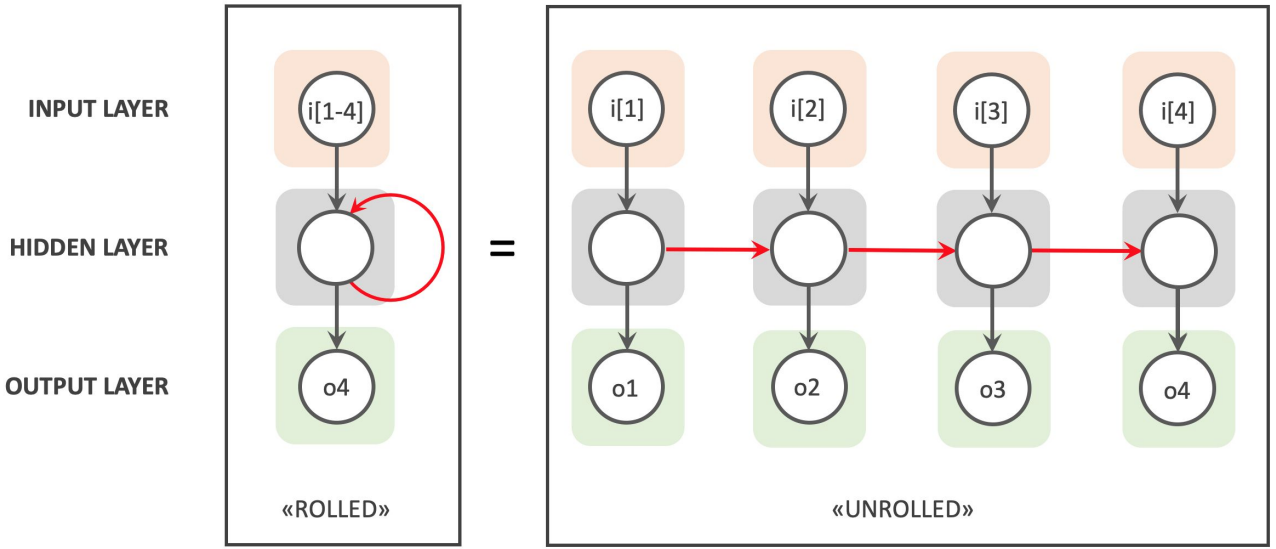
Deep Knowledge Tracing (DKT)

Duel

6 Results

On all three datasets Deep Knowledge Tracing substantially outperformed previous methods. On the Khan dataset using an LSTM neural network model led to an AUC of 0.85 which was a notable improvement over the performance of a standard BKT (AUC = 0.68), especially when compared to the small improvement BKT provided over the marginal baseline (AUC = 0.63). See Table 1 and Figure 3(b). On the Assistments dataset DKT produced a 25% gain over the previous best reported result (AUC = 0.86 and 0.69 respectively) [23]. The gain we report in AUC compared to the marginal baseline (0.24) is more than triple the largest gain achieved on the dataset to date (0.07).

Recurrent Neural Network (RNNs)



RNNs support processing of sequential data by the addition of a loop. This loop allows the network to step through sequential input data whilst persisting the state of nodes in the Hidden Layer between steps - a sort of *working memory*.

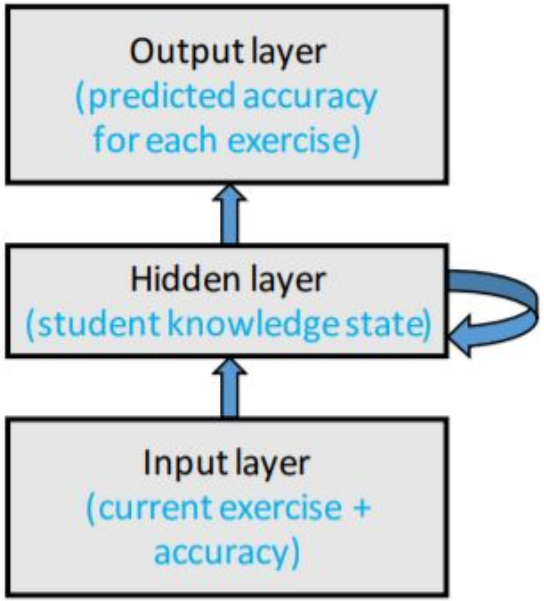


Figure 1: Deep knowledge tracing (DKT) architecture. Each rectangle depicts a set of processing units; each arrow depicts complete connectivity between each unit in the source layer and each unit in the destination layer.

How is RNN Different?

Rather than constructing a separate model for each skill, DKT models all skills jointly. The input to the model is the complete sequence of exercise-performance pairs, {(Xs1, Ys1) ...(Xst, Yst)...(XsT , YsT)}, presented one trial at a time.

In addition to the input and output layers representing the current trial and the next trial, respectively, the network has a hidden layer with fully recurrent connections (i.e., each hidden unit connects back to all other hidden units). **The hidden layer thus serves to retain relevant aspects of the input history as they are used.**

In contrast to Dynamic Bayesian Networks as they appear in education, which are also dynamic, RNNs have a high dimensional, continuous, representation of **latent state**.

RNNs does not need expert annotations (it can learn concept patterns on its own) and (2) it can operate on any student input that can be vectorized.

One disadvantage of RNNs over simple hidden Markov methods is that they require large amounts of training data, and so are well suited to an online education environment, but not a small classroom environment.

Some time to recover...

How Deep is Knowledge Tracing?

Mohammad Khajah
Dept. of Computer Science
University of Colorado
Boulder, Colorado 80309
mohammad.khajah@colorado.edu

Robert V. Lindsey
Dept. of Computer Science
University of Colorado
Boulder, Colorado 80309
robert.lindsey@colorado.edu

Michael C. Mozer
Dept. of Computer Science
University of Colorado
Boulder, Colorado 80309
mozer@colorado.edu

EDM, 2016

Deep learning models are fundamentally nonparametric, in the sense that **interpreting** individual weights and individual unit activations in a network is pretty much **impossible**.

DKT is a very general architecture.

DKT clearly has the capacity to encode learning dynamics that are outside the scope of BKT.

This capacity is what allows DKT to discover structure in the data that BKT misses.

Where Does BKT fall short?

Recency Effects

Recurrent neural networks tend to be more strongly influenced by recent events in a sequence than more distal events . Consequently, DKT is well suited to exploiting recent performance in making predictions. In contrast, the generative model underlying BKT supposes that once a skill is learned, performance will remain strong, and that a slip at time t is independent of a slip at t + 1.

Contextualized Trial Sequence

Because DKT is fed the entire sequence of exercises a student receives in the order the student receives them, it can potentially infer the effect of exercise order on learning. In contrast, because classic BKT separates exercises by skill, preserving only the relative order of exercises within a skill, the training sequence for BKT is the same regardless of whether the trial order is blocked or interleaved.

Inter-Skill Similarity

DKT has the capacity to encode inter-skill similarity.

In contrast, classic BKT treats each skill as an independent modeling problem and thus can not discover or leverage inter-skill similarity

Individual Variation in Ability

DKT is presented with a student's complete trial sequence.

Because BKT models each skill separately from the others, it does not have the contextual information needed to estimate a student's average accuracy or overall ability.

Extending BKT

Forgetting

To better capture recency effects, BKT can be augmented to allow for forgetting of skills. Forgetting corresponds to fitting a BKT parameter $F \equiv P(K_{s,i+1} = 0 \mid K_{si} = 1)$, the probability of transitioning from a state of knowing to not knowing a skill. In standard BKT, $F = 0$.

Skill Discovery

Incorporating Latent Student-Abilities

In summary, enhanced BKT appears to perform as well on average as DKT across the four data sets

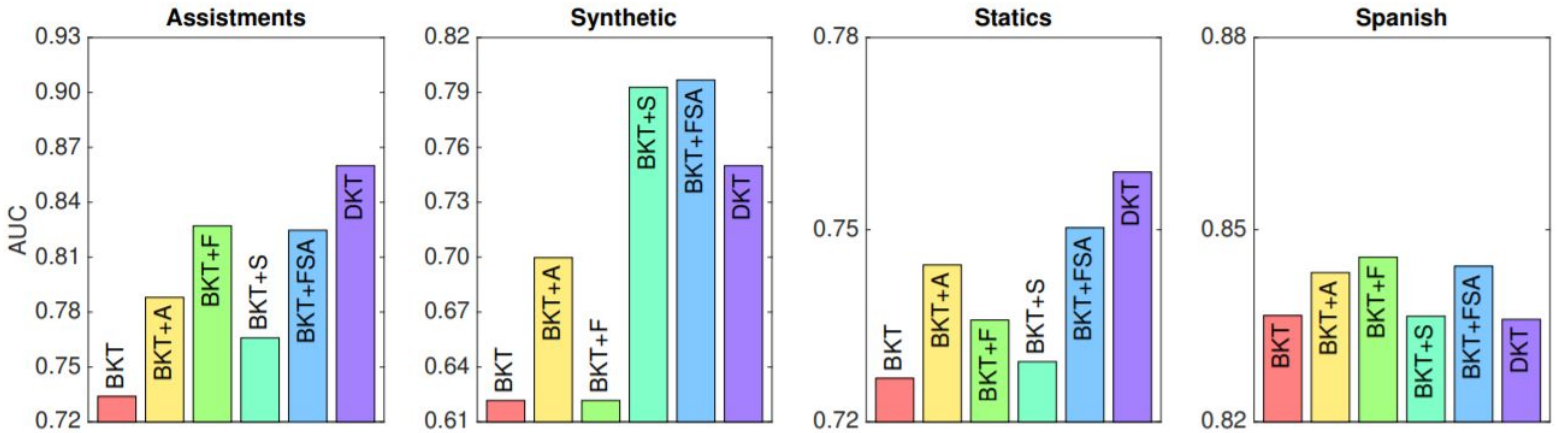


Figure 2: A comparison of six models on four data sets. Model performance on the test set is quantified by AUC, a measure of how well the model discriminates (predicts) correct and incorrect student responses. The models are trained on one set of students and tested on another set. Note that the AUC scale is different for each graph, but tic marks are always spaced by .03 units in AUC. On ASSISTMENTS and SYNTHETIC, DKT results are from Piech et al. [22]; on STATICS and SPANISH DKT results are from our own implementation. BKT= classic Bayesian knowledge tracing; BKT+A= BKT with inference of latent student abilities; BKT+F= BKT with forgetting; BKT+S= BKT with skill discovery; BKT+FSA= BKT with all three extensions; DKT= deep knowledge tracing

DKT does not require its creators to analyze the domain and determine sources of structure in the data.

This flexibility makes DKT robust on a variety of datasets with little prior analysis of the domains.

Although training recurrent networks is computationally intensive, tools exist to exploit the parallel processing power in graphics processing units (GPUs), which means that DKT can scale to large datasets

DKT's advantages come at a price: interpretability. DKT is massive neural network model with tens of thousands of parameters which are near-impossible to interpret.

Although the creators of DKT did not have to invest much up-front time analyzing their domain, they did have to invest substantive effort to understand what the model had actually learned.

Our proposed BKT extensions achieve predictive performance similar to DKT whilst remaining interpretable: the model parameters (forgetting rate, student ability, etc.) are **psychologically meaningful**.

How deep is knowledge tracing?

Deep learning refers to the discovery of representations. Our results suggest that representation discovery is not at the core of DKT's success. We base this argument on the fact that our enhancements to BKT bring it to the performance level of DKT without requiring any sort of subsymbolic representation discovery.4 Representation discovery is clearly critical in perceptual domains such as image or speech classification. But the domain of education and student learning is high level and abstract. The input and output elements of models are psychologically meaningful. The relevant internal states of the learner have some psychological basis. The characterization of exercises and skills can—to at least a partial extent—be expressed symbolically.

Conclusion:

As long as there are sufficient data to constrain the model, DKT is more powerful than classic BKT.

BKT arose in a simpler era, an era in which data and computation resources were precious.

DKT reveals the value of relaxing these constraints in the big data era.

But despite the wild popularity of deep learning, there are many ways to relax the constraints and build more powerful models other than creating a black box predictive device with a vast interconnected tangle of connections and parameters that are nearly impossible to interpret.

Knowledge tracing may be a domain that does not require ‘depth’;

shallow models like BKT can perform just as well and offer us greater interpretability and explanatory power.

Bayesian Learning vs. Deep Learning: Who is the winner???

Do you think interpretability is important?

Is it non-negotiable?

Improving Sensor-Free Affect Detection Using Deep Learning

Anthony F. Botelho[1], Ryan S. Baker[2], and Neil T. Heffernan[1]

[1] Worcester Polytechnic Institute, Worcester, MA
{abotelho,nth}@wpi.edu,
[2] Teachers College, Columbia University, New York, NY
ryanshaunbaker@gmail.com

2017

Expert Feature-Engineering vs. Deep Neural Networks: Which is Better for Sensor-Free Affect Detection?

Yang Jiang[1], Nigel Bosch[2], Ryan S. Baker[3], Luc Paquette[2], Jaclyn Ocumpaugh[3], Juliana Ma. Alexandra L. Andres[3], Allison L. Moore[4], Gautam Biswas[4]

[1] Teachers College, Columbia University, New York, NY, United States
yj2211@tc.columbia.edu
[2] University of Illinois at Urbana-Champaign, Champaign, IL, United States
{pnb, lpaq}@illinois.edu
[3] University of Pennsylvania, Philadelphia, PA, United States
{rybaker, ojaclyn}@upenn.edu
aandres@gse.upenn.edu
[4] Vanderbilt University, Nashville, TN, United States
{allison.l.moore, gautam.biswas}@vanderbilt.edu

2018

KAPAA vs. AUC

Table 1. Cross-validated performance of affect and behavior detector using feature engineering and deep learning.

Affect/Behavior	Feature Engineering				Deep Learning		
	Feature Set	Classifier	Kappa	A′	Model	Kappa	A′
Boredom	Basic	Logistic regression	0.278	0.682	GRU	0.103	0.672
Confusion	All	Logistic regression	0.091	0.568	GRU	0.091	0.566
Delight	Basic	Step regression	0.070	0.570	GRU	0.035	0.649
Engaged Concentration	All	Logistic regression	0.142	0.624	GRU	0.138	0.619
Frustration	All	Logistic regression	0.056	0.634	GRU	0.041	0.572
Off-Task Behavior	Basic + Sequence	Logistic regression	0.369	0.725	LSTM	0.268	0.761
Average			0.168	0.634		0.112	0.640

Featured engineering can differentiate presence of affective state better.

Deep learning can differentiate affective states better.

Conclusion

Deep learning model would be preferable if we want to integrate a detector with a tunable threshold or multiple thresholds, but would be less useful for a single threshold making a single distinction at 50% confidence.

In comparing the two modeling approaches to each other, a key advantage of deep neural networks is their capability to automatically derive meaningful features from raw interaction data. Indeed, this is the core capability that has driven advances in deep learning models, and what distinguishes them from shallow neural networks. However, it is also arguable that time saved by this advantage is lost due to time spent refining the structure of neural networks, which is also an open-ended, time-consuming task for any new domain. As such, further research is needed to quantify this tradeoff.

On the other hand, the traditional feature engineering approach has its own strengths. The resulting models using feature engineering, particularly simple models such as logistic regression, are more interpretable from a psychological and educational perspective because they provide meaningful information on which features are more strongly associated with each affective and behavioral construct of student engagement. Conversely, deep learning models are typically more complex and the model parameters are difficult to analyze and interpret.

This flexibility, however, also exhibits a drawback in terms of lacking interpretability; the large number of parameters and complexity of each model used in this work make it infeasible to study and understand how the model makes its predictions from the features it has available, particularly as it learns from previous time steps. At best, we can understand that the model is relatively better at predicting the more common categories (boredom and concentration) than the more scarce classes (frustration and confusion).

AI in Education: *80 minutes Tour*

- **AI in Education:** *City Tour (15 min)*
- **Intelligent Tutoring System (ITS):** *Restaurant Stop (30 mins)*
  - **Bayesian Modeling vs. Deep Learning:** *Duel Match (15 mins)*
  - **Causal Modeling:** *Detour (5 mins)*

- **AI in Education- Future Frontiers:** *Mountain View (10 mins)*
- **Holy Grail:** *Discussions (5 mins)*

Causal Modeling

Go to Causal Modeling Slides

AI in Education: *80 minutes Tour*

- **AI in Education:** *City Tour (15 min)*
- **Intelligent Tutoring System (ITS):** *Restaurant Stop (30 mins)*
    - **Bayesian Modeling vs. Deep Learning:** *Duel Match (15 mins)*
    - **Causal Modeling:** *Detour (5 mins)*

- **AI in Education- Future Frontiers:** *Mountain View (10 mins)*
- **Holy Grail:** *Discussions (5 mins)*

International Journal of Artificial Intelligence in Education
June 2016, Volume 26, Issue 2, pp 600–614 | Cite as

Stupid Tutoring Systems, Intelligent Humans

Authors Authors and affiliations

Ryan S. Baker

We are left with a bit of a puzzle. ITS research has been successful at producing impressive technologies (and there are many beyond the small sample discussed here), and ITS systems are now being used by tens or hundreds of thousands of learners, but the systems being used at scale are generally not representative of the full richness that research systems demonstrate.

International Journal of Artificial Intelligence in Education
March 2016, Volume 26, Issue 1, pp 25–36 | Cite as

Face-to-Face Interaction with Pedagogical Agents, Twenty Years Later

Authors | Authors and affiliations

W. Lewis Johnson, James C. Lester

Future with Pedagogical Agents

One can imagine a future in which every learner has her own pedagogical agent—or perhaps a cast of pedagogical agents—that accompanies her from the time she is young through adulthood and on into senescence.

And rather than being limited to a particular subject matter, they could support all subject matters, and expand to metacognitive and self-regulatory skills and beyond. They could also support complex collaborative problem solving in which teams of learners and their agents coordinate their learning activities.

Will learners come to trust and find common ground with agents as they do with their real teachers and mentors? As the boundary between interactive agents and portrayals of real people dissolves, will learners treat agents as fictional of real? Will this still be a meaningful distinction?

International Journal of Artificial Intelligence in Education
June 2016, Volume 26, Issue 2, pp 796–808 | Cite as

Optimists' Creed: Brave New Cyberlearning, Evolving Utopias (Circa 2041)

Authors Authors and affiliations

Winslow Burleson, Armanda Lewis

This essay imagines the role that artificial intelligence innovations play in the integrated living, learning and research environments of 2041.

Artificial Intelligence in Education (AIED) has transitioned from what was primarily a research endeavour, with educational impact involving millions of user/learners, to serving, now, as a core contributor to democratizing learning (Dewey 2004) and active citizenship for all (billions of learners throughout their lives).

The way I see it...

Two paths:

MOOCs + Massive Data + Deep Learning (take care of latents)

Smaller data + Probabilistic Modeling + Causal Modeling (reveals latents)

Takes care of Latents

Reveals Latents

An AI Pioneer Wants His Algorithms to Understand the 'Why'

Deep learning is good at finding patterns in reams of data, but can't explain how they're connected. Turing Award winner Yoshua Bengio wants to change that.

“It’s a big thing to integrate [causality] into AI,” says University of Montreal researcher Yoshua Bengio.

International Journal of Artificial Intelligence in Education
June 2016, Volume 26, Issue 2, pp 544–560 | Cite as

The Evolution of Research on Digital Education

Authors | Authors and affiliations

Pierre Dillenbourg

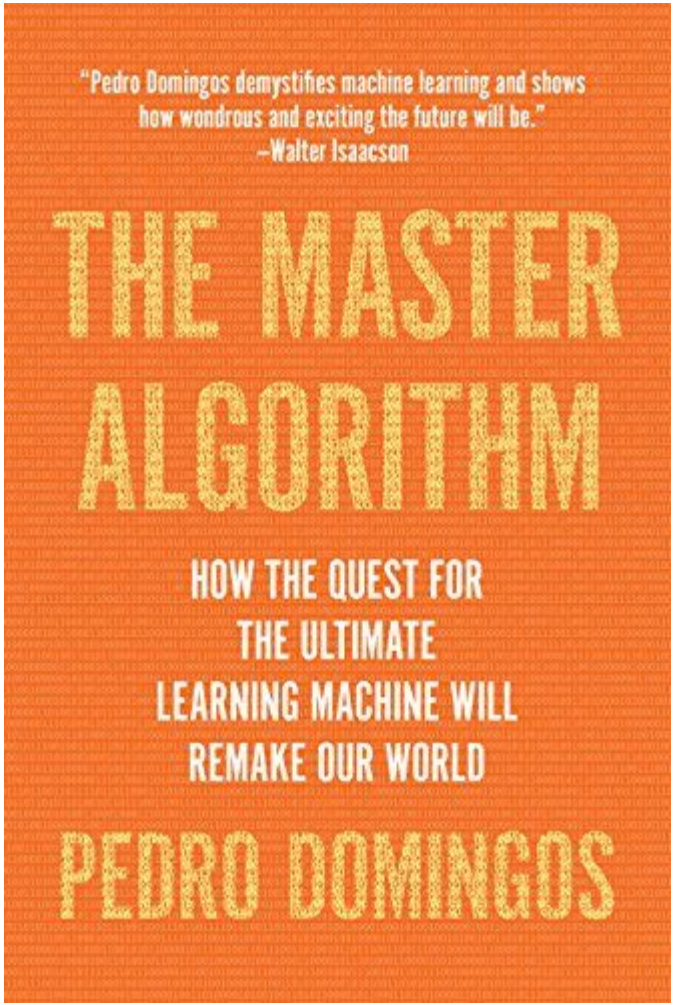
From Semantics to Probability

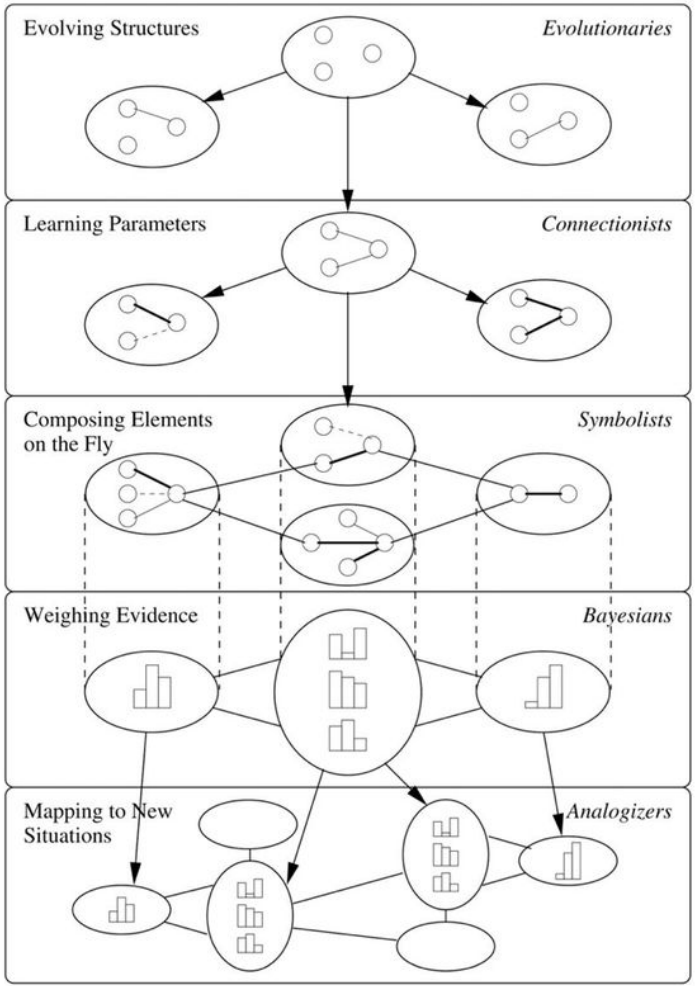
The use of Bayesian networks for learner modelling appeared in the early nineties (1995)

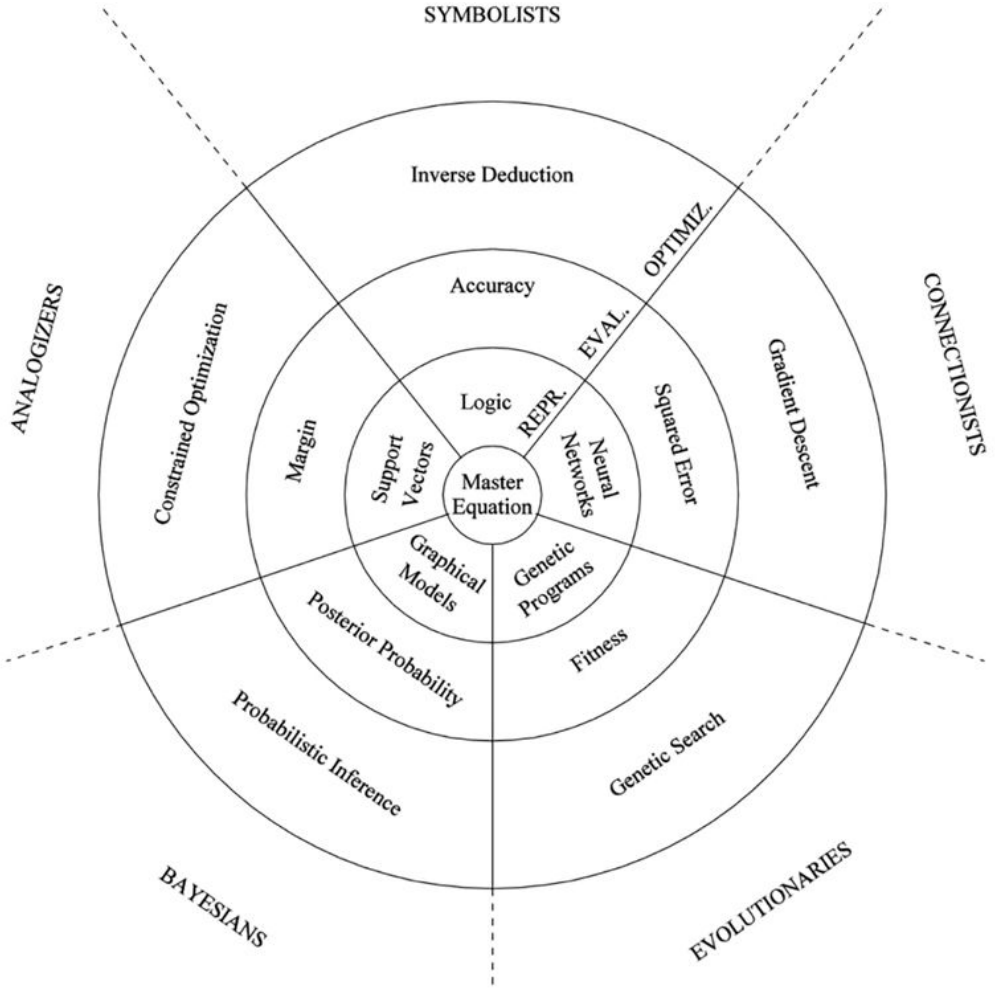
Hence, a challenge for the future of AIED is the integration of multiple levels of abstraction, from raw signals to domain ontologies, into multi-layered algorithms.

Different approaches are first perceived as mutually exclusive but then integrated for their complementarity.

Wisdom comes with time. Such a two-phase cycle takes between 5 and 10 years, so we should witness of few more of them in the coming decade.


"Pedro Domingos demystifies machine learning and shows how wondrous and exciting the future will be."
–Walter Isaacson
THE MASTER ALGORITHM
HOW THE QUEST FOR THE ULTIMATE LEARNING MACHINE WILL REMAKE OUR WORLD
PEDRO DOMINGOS


Evolving Structures
Evolutionaries
Learning Parameters
Connectionists
Composing Elements on the Fly
Symbolists
Weighing Evidence
Bayesians
Mapping to New Situations
Analogizers



If we find a Master Algorithm, let's apply to Education.

Discussions

AI in Education: Holy Grail

2016-04-28

Modes and Mechanisms of Game-like Interventions in Intelligent Tutoring Systems

Dovan Rai
Worcester Polytechnic Institute

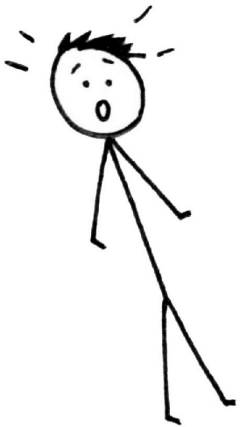
AI in Education: 80 mins Tour

This is not the only face of Rigor.

AI is Diverse

Python for Machine Learning

Dovan Rai



Seaborn

pandas

$y_i t = \beta' x_{it} + \mu_i + \epsilon_{it}$

ANACONDA

CONDA

Keras

scikit learn

NumPy

TensorFlow

jupyter

python™

matplotlib

Practice and Transcend

ANACONDA

CONDA

K Keras

scikit learn

pandas

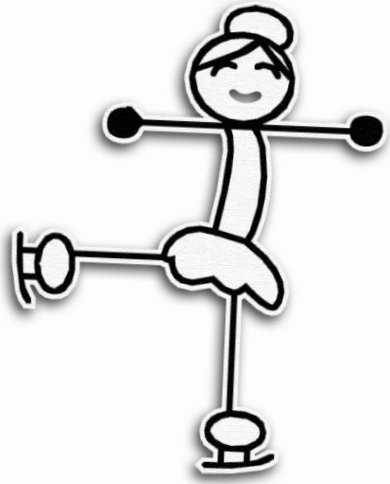
NumPy

TensorFlow

Seaborn

matplotlib

Create and have fun...



ANACONDA

CONDA

Keras

scikit learn

pandas

NumPy

TensorFlow

Seaborn

matplotlib

Thank You !!