Knowledge Tracing Machines: Factorization Machines for Knowledge Tracing

Jill-Jênn Vie
RIKEN Center for Advanced Intelligence Project (AIP)
Tokyo, Japan
vie@jill-jenn.net

Hisashi Kashima
Kyoto University/RIKEN
Kyoto, Japan
kashima@i.kyoto-u.ac.jp

Problem: Knowledge Tracing

We want to predict the performance of students over questions.
$\simeq$ collaborative filtering + | students can attempt a question multiple times
| students can learn between attempts
Fit: Ordered triplets $(i, j, o) \in I \times J \times \{0, 1\}$
$\Rightarrow$ Student i attempted question j and got it correct/incorrect.
Predict: $(i, j, ?)$ for new triplets.

Existing Models

- **Prediction of sequences**: Bayesian or Deep Knowledge Tracing [1]
- **Factor Analysis**: Item Response Theory, Performance Factor Analysis

$$\underbrace{\text{BKT}}_{\text{HMM}} < \text{PFA} \simeq^{[4]} \underbrace{\text{DKT}}_{\text{LSTM}} \leq^{[3]} \text{IRT} \leq^{[\text{this poster}]} \text{KTM}$$

Item Response Theory (IRT)

Students $i \in I$ have unknown level θ_i
Questions $j \in J$ have unknown difficulty d_j

$$\text{logit } p_{ij} = \text{logit Pr(Student } i \text{ answers correctly question } j) = \theta_i - d_j$$

$\Rightarrow$ **really simple, ignores skills & multiple attempts**

Multidimensional item response theory (MIRT): logit $p_{ij} = \langle \boldsymbol{\theta_i}, \boldsymbol{d_j} \rangle + \delta_j$

Performance factor analysis (PFA)

Students $i \in I$ have level θ_i, W_{ik} wins and F_{ik} fails over skill k
Questions $j \in J$ have known requirements KC$(j) \subseteq K$
Skills $k \in K$ have bias β_k and bonus after win γ_k and fail δ_k

$$\text{logit } p_{ij} = \theta_i + \sum_{k \in KC(j)} \beta_k + \gamma_k W_{ik} + \delta_k F_{ik}$$

$\Rightarrow$ **ignores item difficulty**

Additive Factor Model (AFM): only consider attempts (i.e., $\gamma_k = \delta_k$)

Our proposal

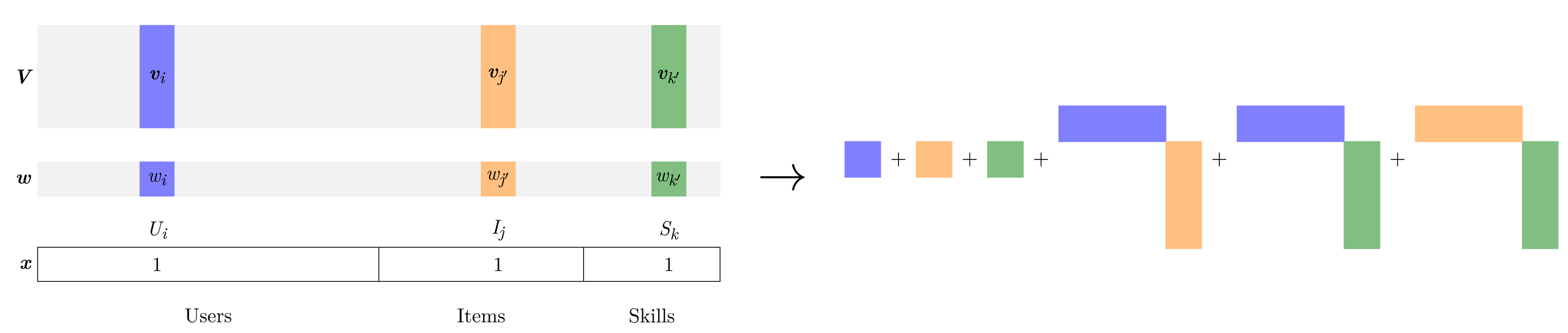
Knowledge Tracing Machines (KTM)

Students $i \in I$, questions $j \in J$ and side information are encoded into x
All entities have a bias w_k and embedding $\boldsymbol{v_k}$ to model pairwise relationships:

$$\text{probit } p(\boldsymbol{x}) = \mu + \underbrace{\sum_{k=1}^{N} w_k x_k}_{\text{logistic regression}} + \underbrace{\sum_{1 \leq k < l \leq N} x_k x_l \langle \boldsymbol{v_k}, \boldsymbol{v_l} \rangle}_{\text{pairwise relationships}}$$

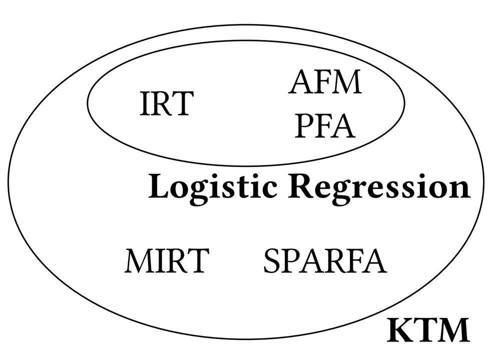
E.g. ψ = probit, $w_k, v_{kf} \sim \mathcal{N}(\mu, 1/\lambda)$, $\mu \sim \mathcal{N}(0, 1)$, $\lambda \sim \Gamma(1, 1)$
Factorization machine trained using **Gibbs sampling** [2] or variational inference

Encoding data into sparse features



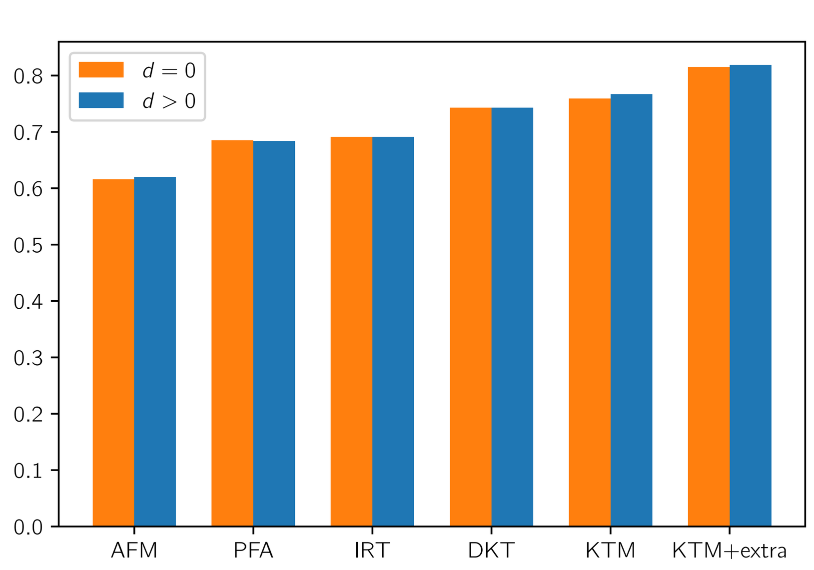
Triplet	Users		Items			Skills			Wins			Fails			Outcome
	1	2	Q_1	Q_2	Q_3	S_1	S_2	S_3	S_1	S_2	S_3	S_1	S_2	S_3	
(2, 2, 1)	0	1	0	1	0	1	1	0	0	0	0	0	0	0	1
(2, 2, 0)	0	1	0	1	0	1	1	0	1	1	0	0	0	0	0
(2, 2, 1)	0	1	0	1	0	1	1	0	1	1	0	1	1	0	1
(2, 3, 0)	0	1	0	0	1	0	1	1	0	2	0	0	1	0	0
(2, 3, 1)	0	1	0	0	1	0	1	1	0	2	0	0	2	1	1
(1, 2, 1)	1	0	0	1	0	1	1	0	0	0	0	0	0	0	1
(1, 1, 0)	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0

KTM take IRT, MIRT, PFA as special cases
Encoding users + items == IRT
Encoding users + skills + attempts == AFM
Encoding users + skills + wins + fails == PFA



Various, large-scale educational datasets

Name	Users	Items	Skills	Skills per item	Entries	Sparsity	Attempts per user
fraction	536	20	8	2.800	10720	0.000	1.000
timss	757	23	13	1.652	17411	0.000	1.000
ecpe	2922	28	3	1.321	81816	0.000	1.000
assistments	4217	26688	123	0.796	346860	0.997	1.014
berkeley	1730	234	29	1.000	562201	0.269	1.901
castor	58939	17	2	1.471	1001963	0.000	1.000

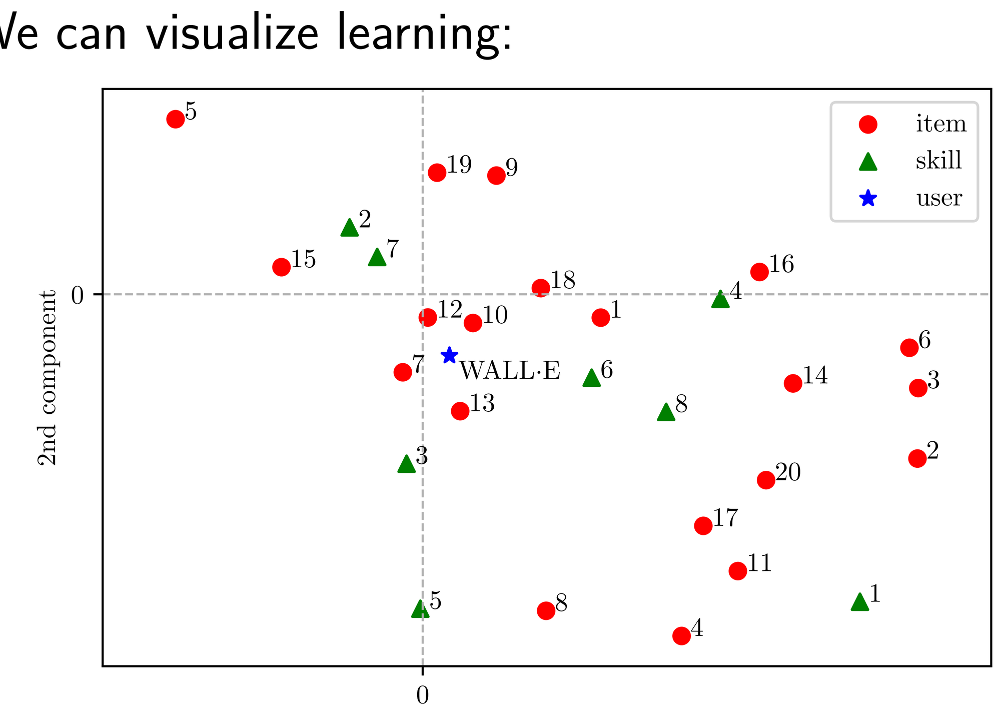


model	dim	AUC	improvement
KTM: items, skills, wins, fails, extra	5	0.819	
KTM: items, skills, wins, fails, extra	0	0.815	+0.05
KTM: items, skills, wins, fails	10	0.767	
KTM: items, skills, wins, fails	0	0.759	+0.02
DKT (Wilson et al., 2016)	100	0.743	+0.05
IRT: users, items	0	0.691	
PFA: skills, wins, fails	0	0.685	+0.07
AFM: skills, attempts	0	0.616	

	AFM	PFA	IRT	MIRTb10	MIRTb20	KTM(iswf0)	KTM(iswf20)	KTM(iswfe5)
assistments	0.6163	0.6849	0.6908	0.6874	0.6907	0.7589	0.7502	**0.8186**
berkeley	0.675	0.6839	0.7532	0.7521	0.7519	0.7753	**0.7780**	–
ecpe	–	–	**0.6811**	0.6807	**0.6810**	–	–	–
fraction	–	–	0.6662	0.6653	**0.6672**	–	–	–
timss	–	–	**0.6946**	0.6939	0.6932	–	–	–
castor	–	–	**0.7603**	**0.7602**	0.7599	–	–	–

Findings

- It is better to learn a item bias
- Side information helps more than latent dimension
- We can handle multiple skills per item
- We can visualize learning:



References

[1] Chris Piech et al. "Deep knowledge tracing". In: *Advances in Neural Information Processing Systems (NIPS)*. 2015, pp. 505–513.

[2] Steffen Rendle. "Factorization Machines with libFM". In: *ACM Transactions on Intelligent Systems and Technology (TIST)* 3.3 (2012), 57:1–57:22. DOI: 10.1145/2168752.2168771.

[3] Kevin H. Wilson et al. "Back to the basics: Bayesian extensions of IRT outperform neural networks for proficiency estimation". In: *Proceedings of the 9th International Conference on Educational Data Mining (EDM)*. 2016, pp. 539–544.

[4] Xiaolu Xiong et al. "Going Deeper with Deep Knowledge Tracing". In: *Proceedings of the 9th International Conference on Educational Data Mining (EDM)*. 2016, pp. 545–550.