DeepMind RecSys Research

Ray Jiang (rayjiang@google.com)
October 5, 2018

Outline

- Research on robust recommender systems
  - Slate Optimization
  - Verification
  - Long-term Value Prediction
  - Learning from Delayed Signals
- Future Directions
- Conclusion

Slate Optimization

Ray Jiang, Sven Gowal, Yuqiu Qian,
Timothy A. Mann, Danilo J. Rezende

https://arxiv.org/abs/1803.01682

The Problem

Maximize user engagement on a whole page (slate) considering layout biases

Formulation

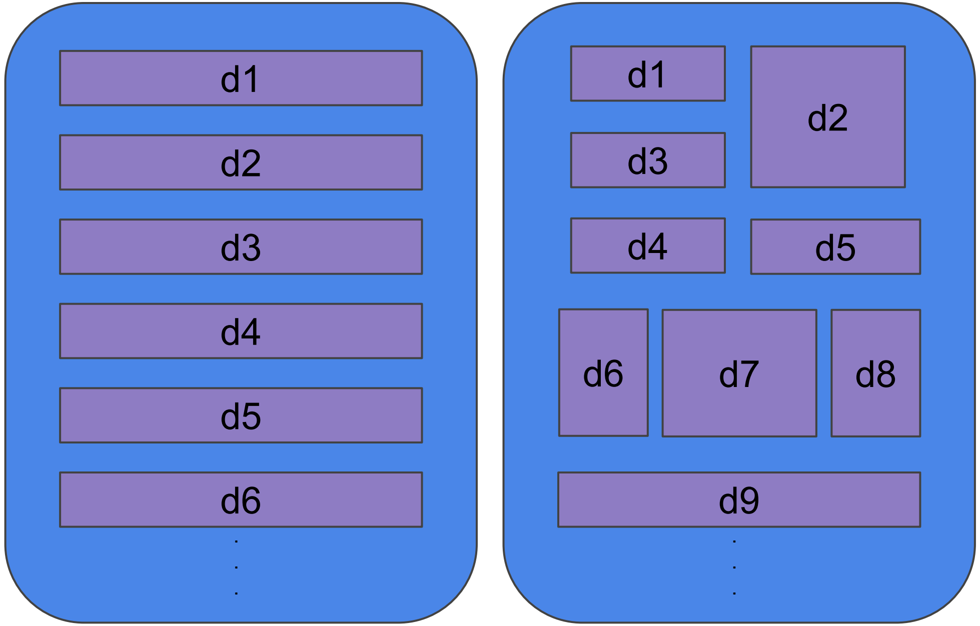
$\mathcal{D}$: a corpus of documents

k : size of the slate

Slate : $\mathbf{s} = (d_1, d_2, \ldots, d_k), d_i \in \mathcal{D}$.

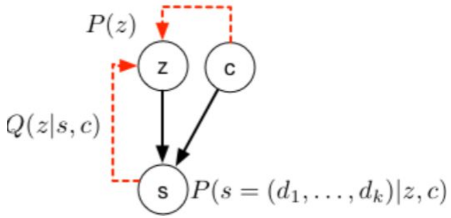
Response : $\mathbf{r} = (r_1, r_2, \ldots, r_k), r_i \in \mathcal{R}$.

The problem is to generate slates that optimizes $\mathbb{E}[\sum_{i=1}^{k} r_i]$.

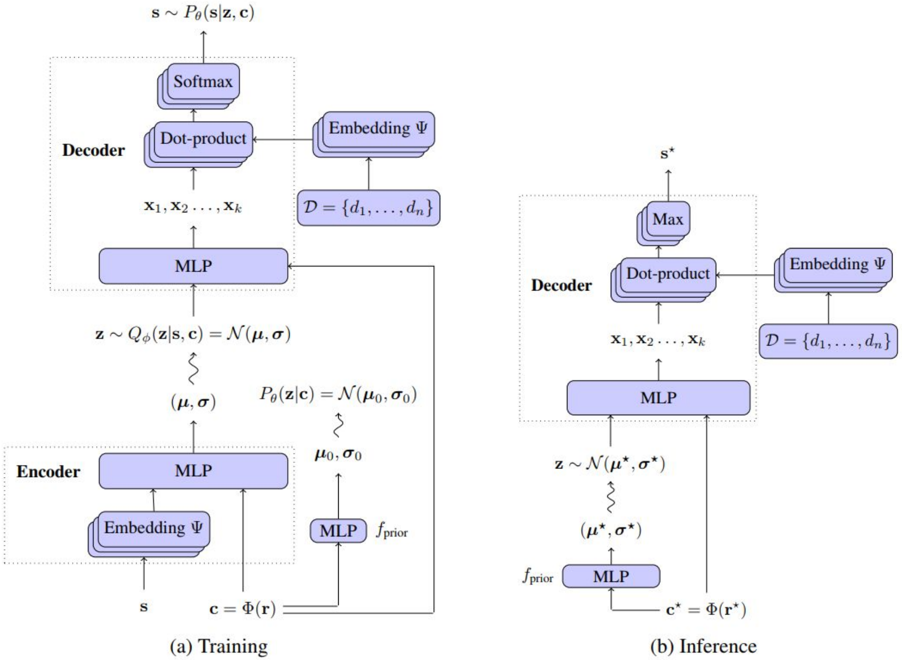


Potential Solution

Conditional Variational Autoencoder (CVAE) structure:



List-CVAE:

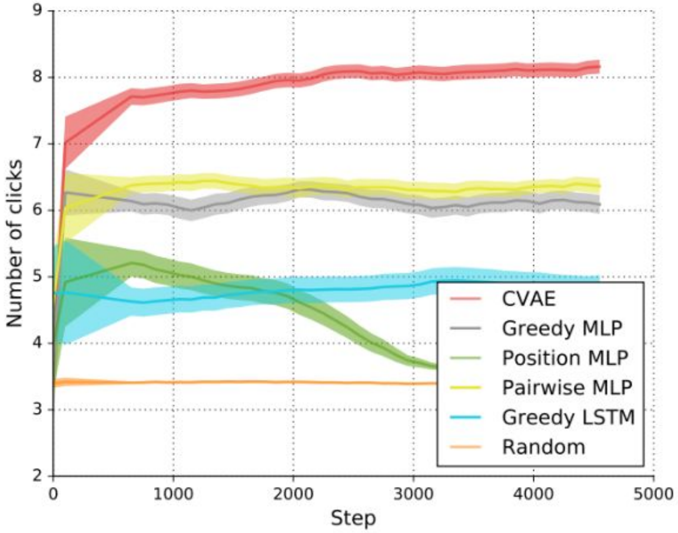


Loss function:

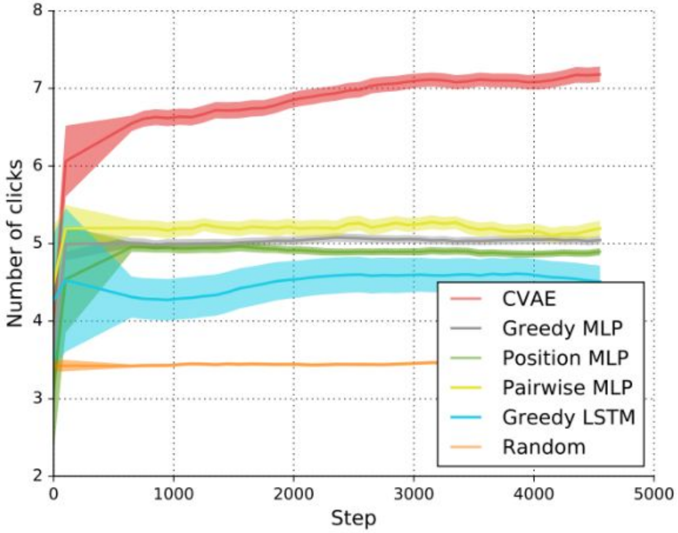
$$\mathcal{L} = \beta \mathrm{KL}\left[Q_\phi(\mathbf{z} \mid \mathbf{s}, \mathbf{c}) \| P_\theta(\mathbf{z})\right] - \mathbb{E}\left[\log P_\theta(\mathbf{s} \mid \mathbf{z}, \mathbf{c})\right]$$

Experimental Results on user simulation data

$$r_i \sim \mathcal{B}\left[\left(A_i \prod_{j=1}^{i} W_{i,d_i,j,d_j}\right)\Big|_{[0,1]}\right]$$



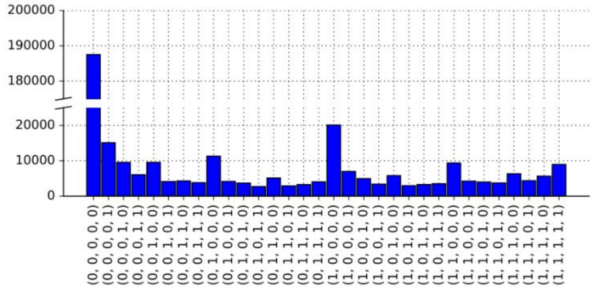
(a) $n = 100, k = 10$



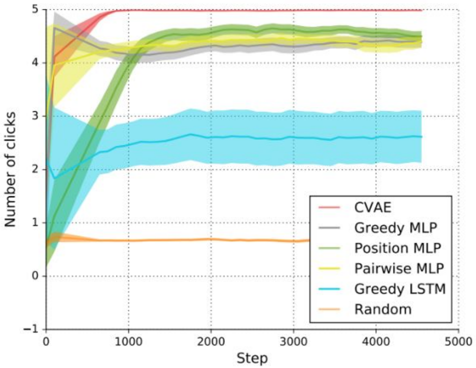
(b) $n = 1000, k = 10$

Superior performance over MLP on medium corpora of documents (100 or 1000)

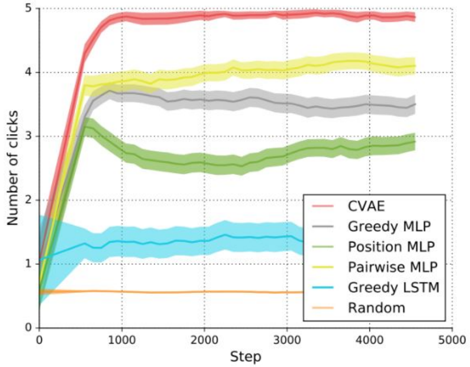
Experiment results on RecSys 2015 Challenge dataset



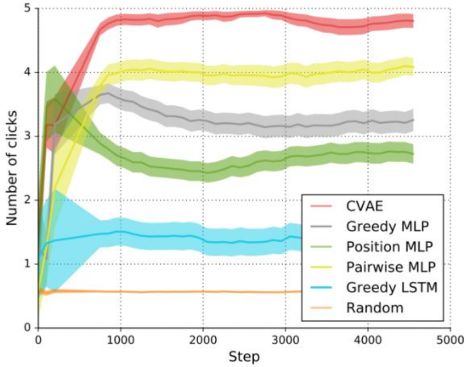
(a) Data distribution by user responses



(b) $n = 10,000$ documents

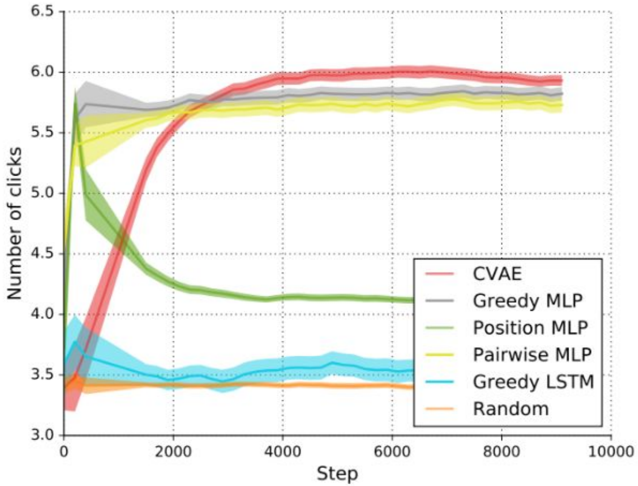


(c) $n = 1$ million documents



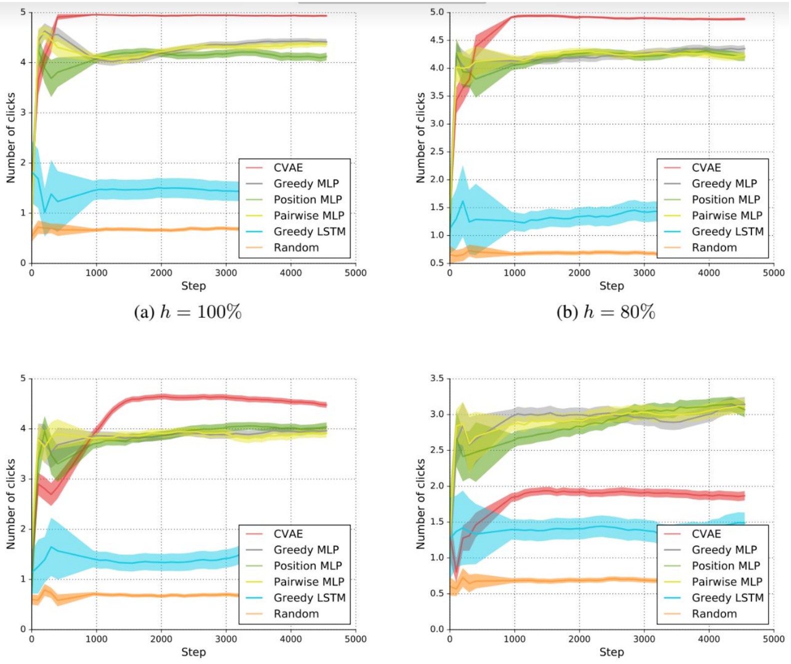
(d) $n = 2$ million documents

Personalization & Generalization Test



(c) Personalization Test: $|\mathcal{U}| = 50, n = 100, k = 10$

$$r_i^u \sim \mathcal{B}\left[\left(A_{\pi_u(i)} \prod_{j=1}^{i} W_{i,d_{\pi_u(i)},j,d_{\pi_u(j)}}\right)\Big|_{[0,1]}\right]$$



(a) $h = 100\%$ (b) $h = 80\%$ (c) $h = 60\%$ (d) $h = 40\%$

Using only the worst h% slates from the training set.

Summary

- List-CVAE works better than MLP score ranked slates when there are interdependencies between documents in a slate and positional bias.
- List-CVAE can generalize over user responses conditioning to some extent. Thus if the training set doesn't contain any optimal slates, the model may still be able to infer what optimal slates are for a user.
- Future directions :
  - Test different model structures, such as GQN, Draw, GraphNet.
  - Test the approach on large action space problems in RL environments for total reward optimization over a sequence of slates.
  - Model the distribution of conditions.

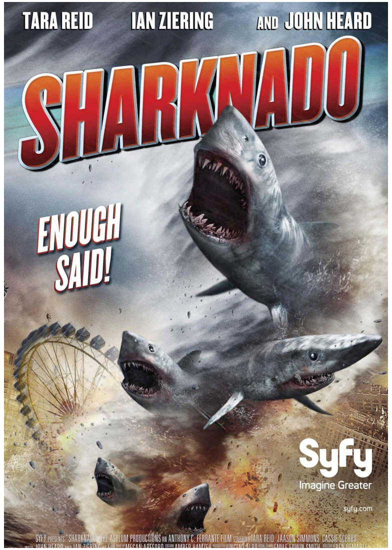
Verification

Krishnamurthy (Dj) Dvijotham, Robert Stanforth,
Sven Gowal, Timothy Mann, Pushmeet Kohli

https://arxiv.org/abs/1803.06567

The Problem

Supervised learning systems are fragile, susceptible to making mistakes in novel situations

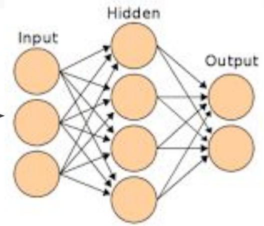


???

Potential Solution

Test

input

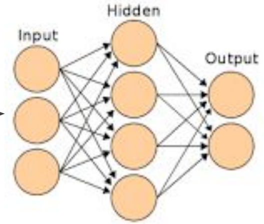


Output
label
correct?

Statistical test: If model does well on “large enough” random test set, accept

Test

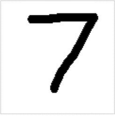
input

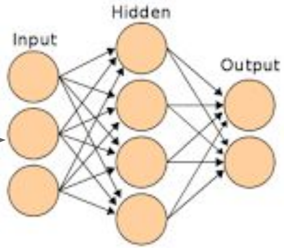


Output
label
correct?

Adversarial test: If model does well on “adversarially perturbed” test set, accept

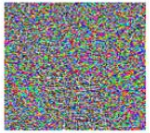
Module level verification



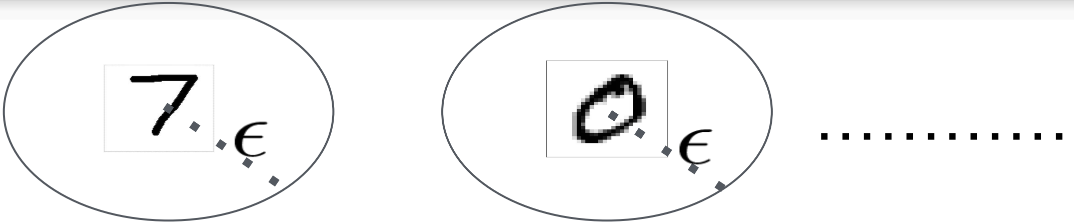


Seven

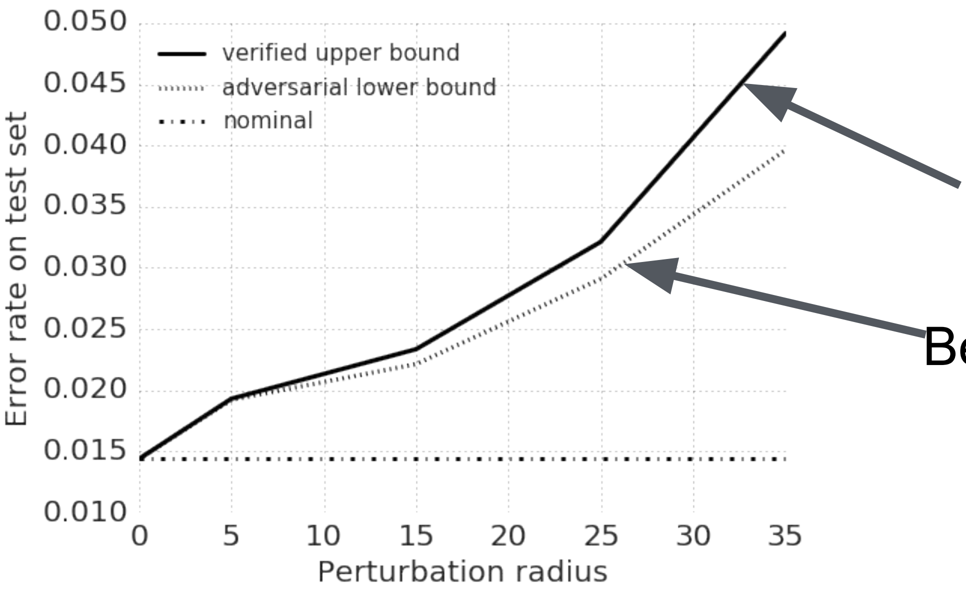
Specification: Output remains “Seven” for **ALL IMAGES** of the form

 $\pm\epsilon$ 

Experimental Results



How many test examples have no adversarial examples within a distance ϵ?



Verified upper bound on any attack

Best attack found

Summary

- Fragility of ML systems can be mitigated by providing guarantees of specifications (rules based on domain knowledge that we believe the ML system ought to satisfy)
- Novel verification methods - achieves SOTA results
- Compatible with learning - can learn verifiable models that are guaranteed to satisfy a given specification.
    - Training verified learners with learned verifiers (https://arxiv.org/abs/1805.10265)
- Future directions:
    - Scalability
    - Richer specifications
    - Extensions to RNNs, RL etc.

Long-term Value Prediction

Timothy A. Mann, Hugo Penedones, Shie Mannor, Todd Hester

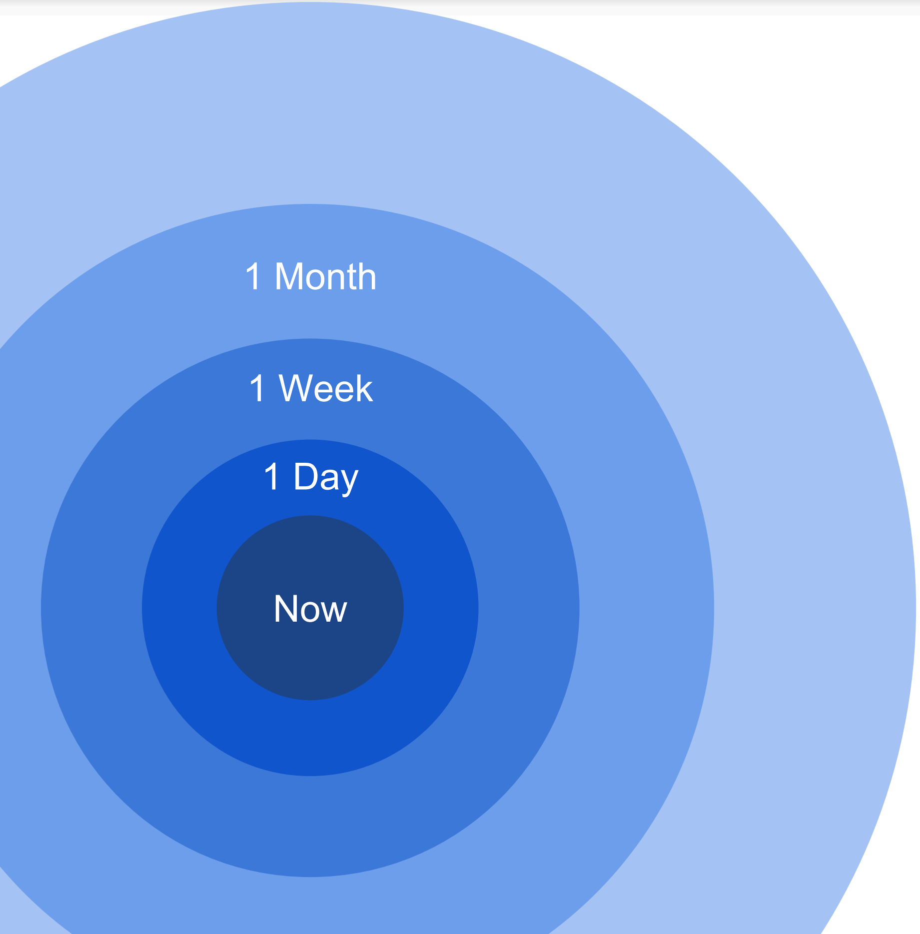
https://arxiv.org/abs/1612.09465

The Problem

How will recommendations influence a customer over time?

Customer Today | Recommendations... | 10 Years Later

The Problem (continued)

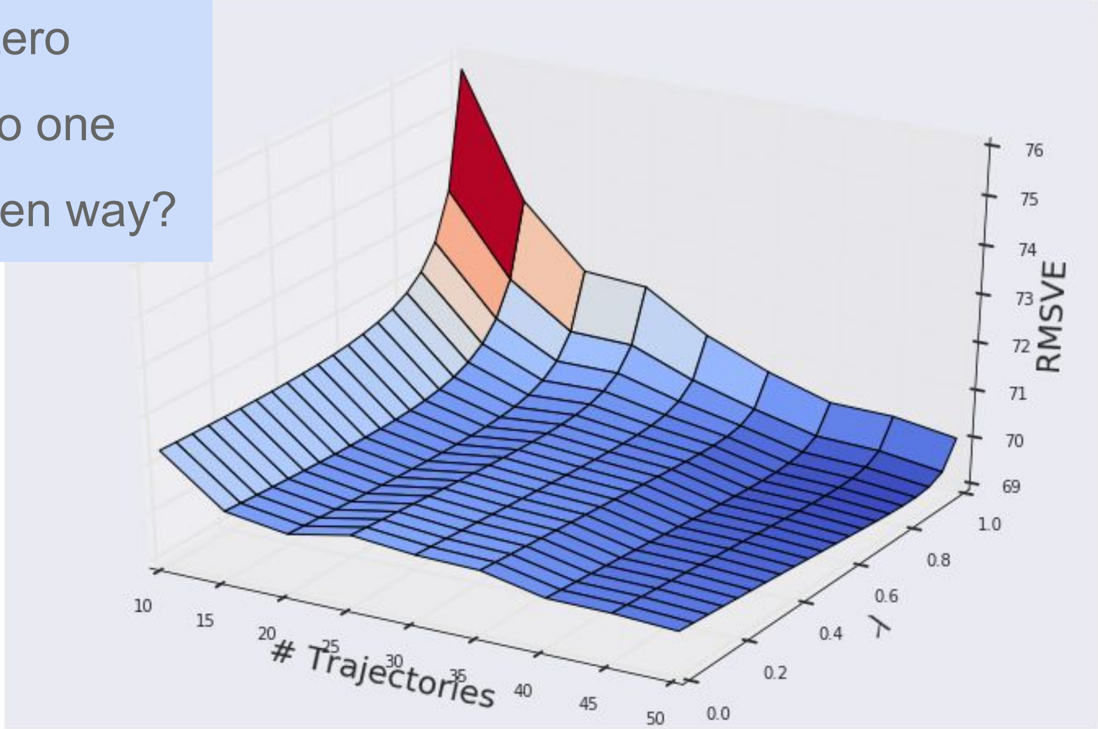


- Predicting over long horizons:
  - Introduces uncertainty
  - Variance is high
  - Little data

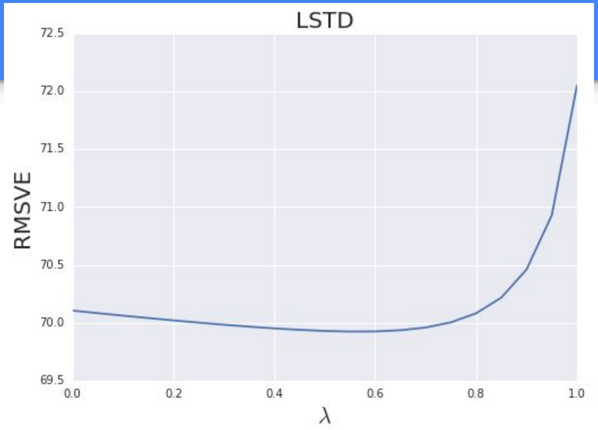
- Temporal Difference learning:
  - Can trade bias for variance
  - Controlled by a parameter λ

Temporal Difference learning

- With little data set λ close to zero
- With a lot of data set λ close to one
- Can we select λ in a data-driven way?

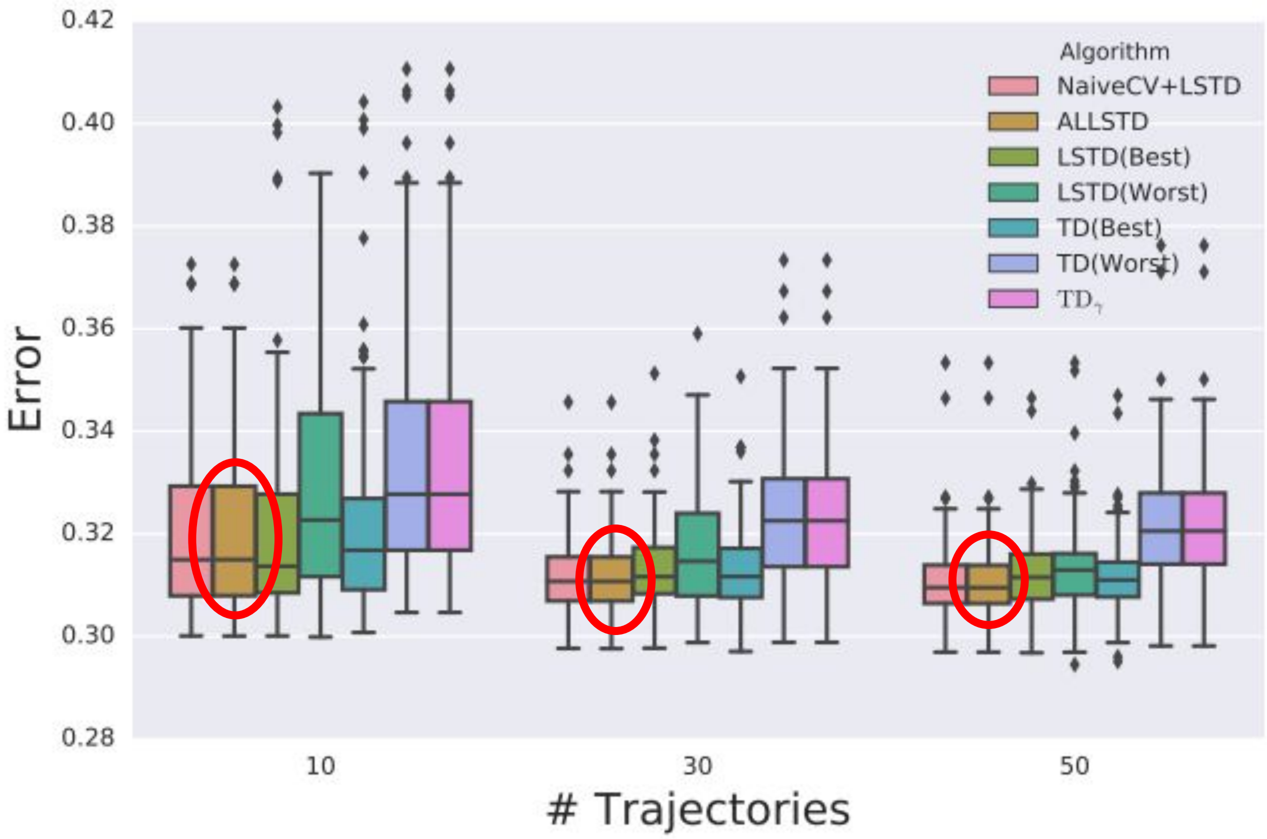


Potential Solution



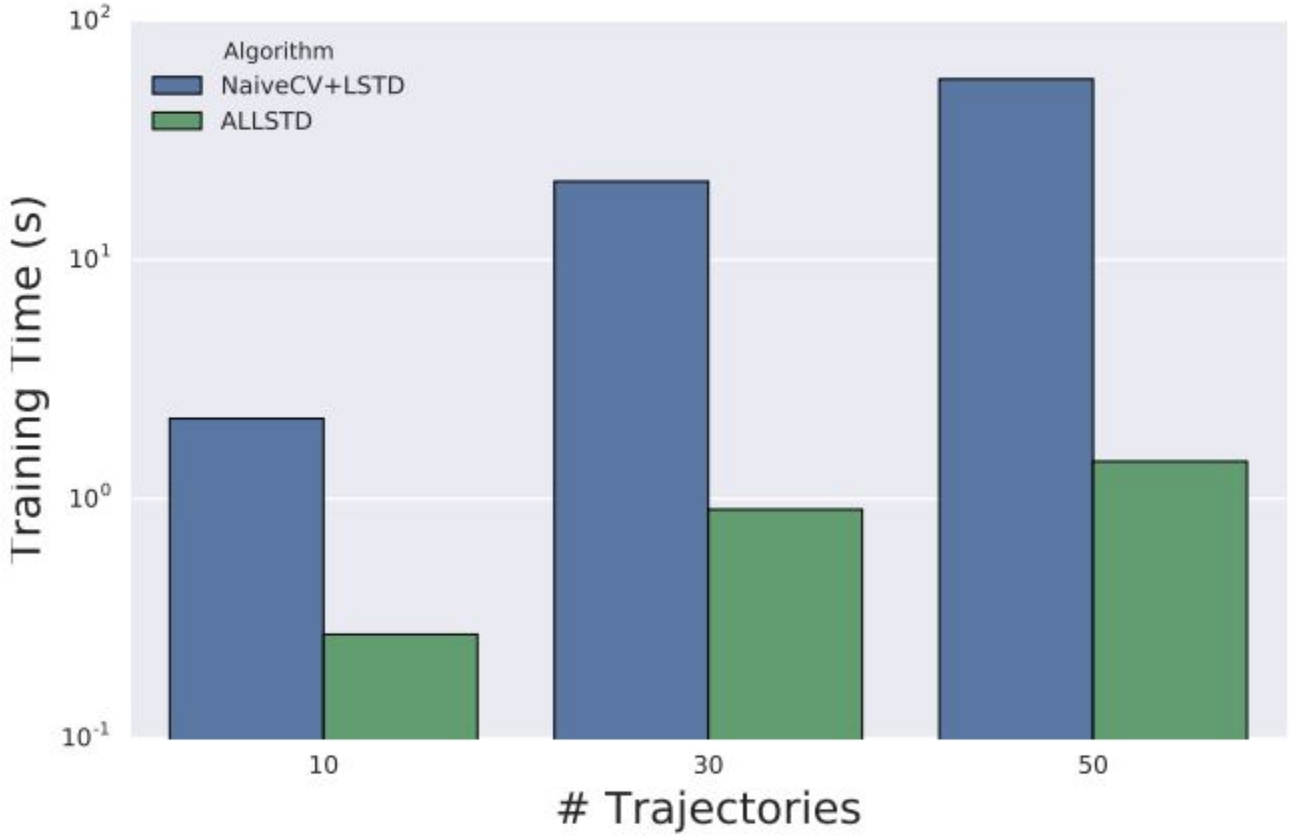
- We derived a very efficient form of cross validation
- Use cross validation to select λ
- For *k* parameters, *n* trajectories, and *d* dimensional observations
    - Proposed Approach: $\mathbf{O}(d^3+kd^2n)$ ← *Less than running LSTD k times*
    - Naive Approach: $\mathbf{O}(kn[d^3+d^2n])$ ~ *Running LSTD kn times*

Experimental Results



Proposed Algorithm achieves **smallest possible error** with different amounts of data.

Experimental Results (continued)



Proposed algorithm is significantly faster than a naive implementation (notice the log-scale)

Summary

- Developed an efficient algorithm for predicting long-term value
- This algorithm automatically trades off bias/variance in a data-driven way
- Runs much faster than a naive implementation

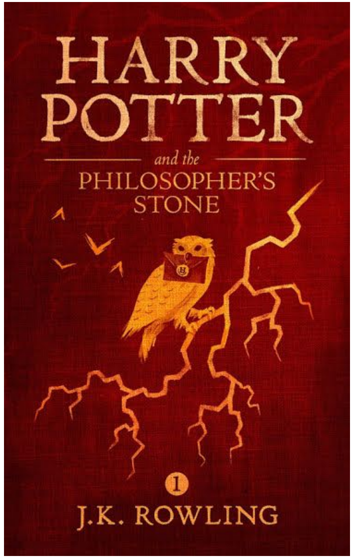
Learning from Delayed Signals

Timothy A. Mann, Sven Gowal, Ray Jiang,
Huiyi Hu, Balaji Lakshminarayanan, Andras Gyorgy

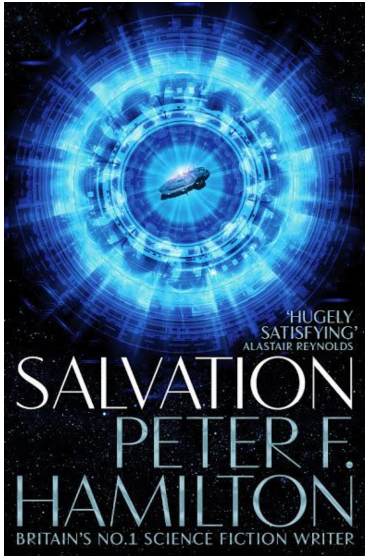
https://arxiv.org/abs/1807.09387

The Problem

Meaningful signals are often delayed



(A)



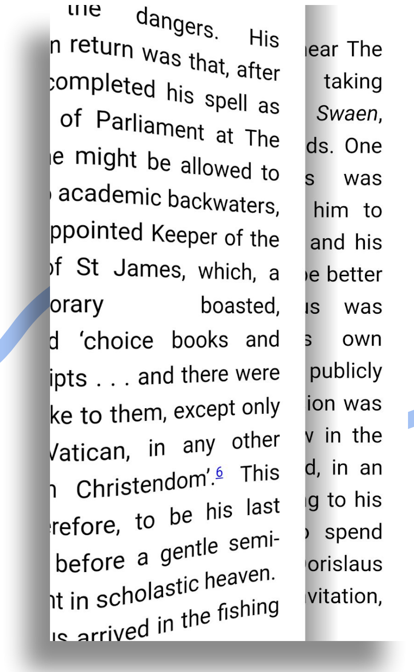
(B)

- People buy more books if they read the books they've acquired
- Want to optimize for completion rate
- Consider two books:
  - (A) has a high completion rate
  - (B) came out a few days ago
- We won't know (B)'s completion rate for at least a month
- But we could be missing out on sales!

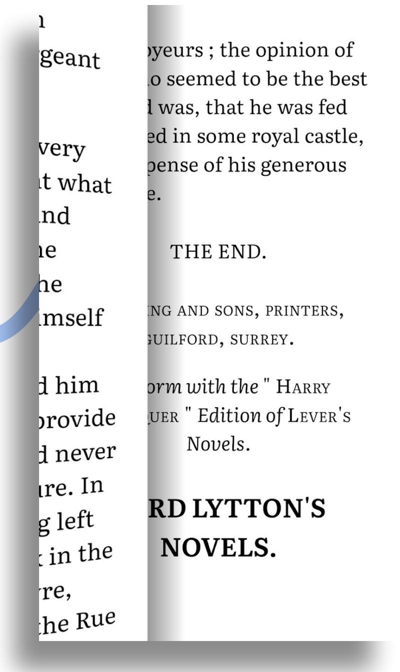
Potential Solution

We don’t have to wait until a book is finished to learn about completion rate

Time of Acquisition

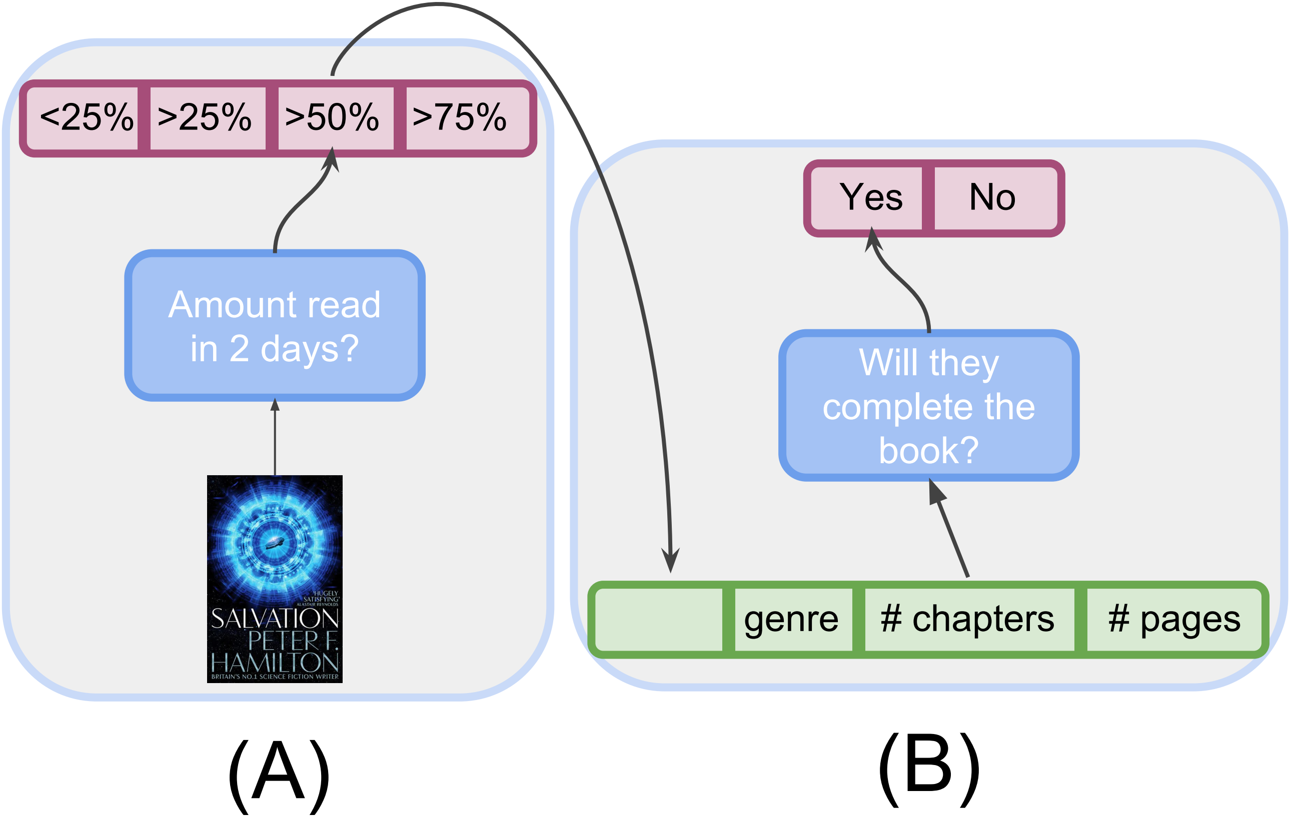


Read 25% after 2 days



Read 100% after 2 weeks

Proposed Solution (continued)



Predict with two models:

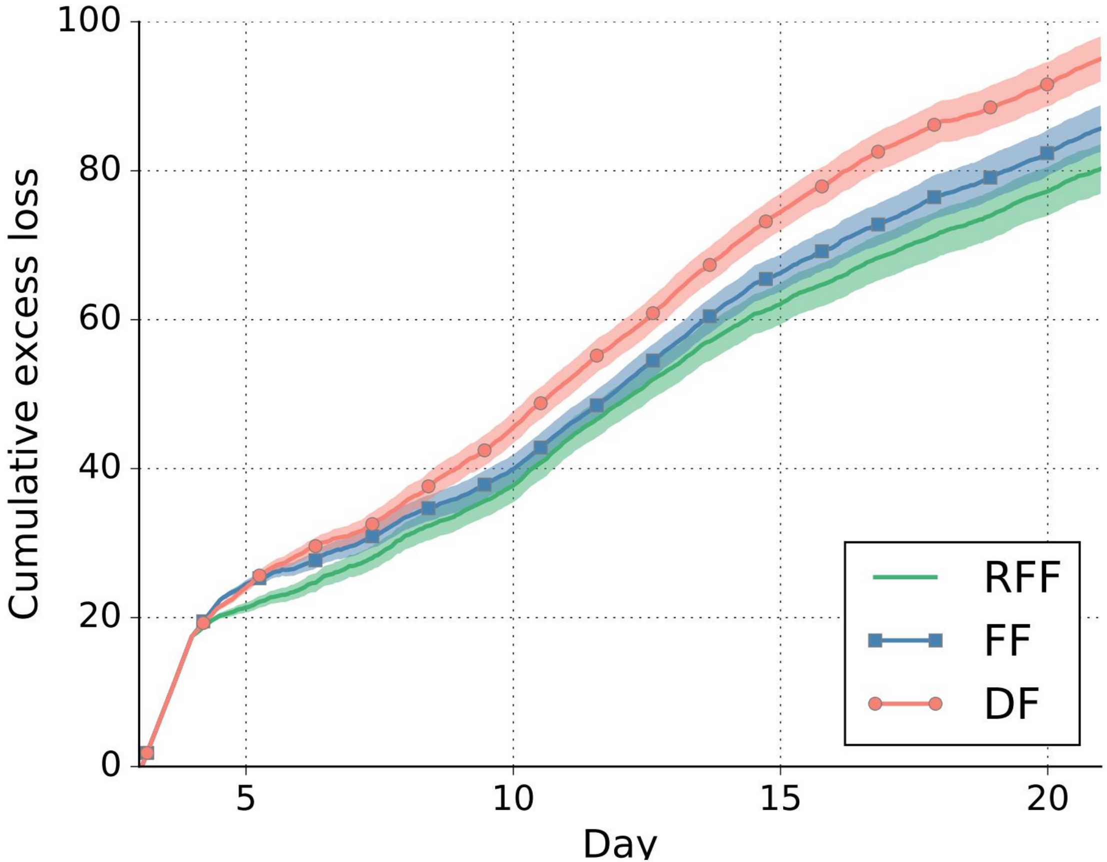
- Model (A)
  - Depends on item
  - Updated quickly
- Model (B)
  - Generalizes over items
  - Updates slowly

Experimental Results

Assumption: $P(y|x) = P(y|z)\ P(z|x)$

Two proposed approaches:

- **FF:** As in previous slide
- **RFF:** Introduces residual to correct for factorization assumption
- Lower loss when new items are constantly being added



Summary

- Factorizing predictions can mitigate the impact of delay:
    - Create two models.
    - Exploit information before the label arrives.
    - One model updates quickly but depends on the specific item.
    - Second model updates slowly but generalizes.

- Residual correction can give better results when factorization is only approximately correct.

Future Directions

Future Directions

- ML fairness for recommendation systems
- Simulating recommender systems
- List-CVAE as a policy network in RL environments

Questions?