Scaling Recommender Systems Research

Michael Ekstrand, Texas State University

@mdekstrand

http://md.ekstrandom.net

RecSys ’15 Workshop on Large-Scale Recommender Systems

Who am I?

- Ph.D 2014 from Minnesota (GroupLens)
- Now teaching and researching at Texas State
- Lead developer of LensKit toolkit
- Research focus: human-algorithm interaction
  - Implication: what I care about is how the algorithm can improve its users' lives.

Scale?

- MovieLens:
  - 200K users
  - 22M ratings
- LensKit:
  - Single (sometimes large!) machine

Large scale?

Large Scale

I want to talk about *large-scale evaluation* of recommender systems.

- Many questions
- Many conditions
- Underlying data may be small

Goals of Today

- What are the dimensions of scaling research?
- What are some challenges?
- What are some (partial) solutions?
- What can be done to help?

Research Methods (Context)

- Offline evaluation and measurement
  - Accuracy metrics
  - Traditional non-accuracy metrics
  - Bespoke measurements
- Online user logging
- User studies and surveys
- Usually interested in user satisfaction, meeting user needs

Scaling Research

How do we scale up our *research capacity*?

How do we do this in academic settings?

Many Algorithms

- Easy to have hundreds, thousands of algorithms
  - Especially when tuning
- How to efficiently test?
  - Cannot throw at users!
  - Expensive to test even offline
- How to efficiently tune?
  - Many combinations
  - Many metrics

3 approaches

Goal: reduce cost of testing

- Reduce # of things to test
- Reduce cost of running a test
- Parallelize testing (reduces time cost, but not CPU hours/power)

Tuning: reducing # of offline tests

Grid search… is super-expensive.

Burke's talk Thursday: need tuning strategies! We have been working on this.

Random search works remarkably well [Bergstra & Bengio, JMLR 13(Feb) 2012]

Random search

- If 5% of response surface is ‘good enough’…
- …then random search of 60 points will hit with 95% success…
- … and is trivially parallelizable

Reducing cost of offline tests

LensKit improves throughput by reusing components across algorithms.

- E.g. similarity matrix unchanged by prediction aggregation strategy
- Identify identical components, build them once
- Enabled by dependency-injected architecture

Opportunity for cost reduction

Open question: can we tune/test on subsets?

What results from subsets translate to full data set?

Reducing user testing cost

Basic pruning strategy – may be familiar.

1. Generate ideas
2. Test many ideas offline
3. Pick best (and most different) for user testing

Problem: measurement

- What can we measure?
- Common refrain: offline metrics weakly correlate with online metrics
- Consistent with my results [RecSys 2014]
  - Accuracy weakly correlates with satisfaction
  - ILS weakly correlates with perceived diversity
- EPFL paper on Friday was very promising
- Weak is better than nothing, but we need better metrics

Aside: research success/process

Repeated question/topic: industry, give us tasks!

I don’t quite agree. But would like

- Reviewers, stop emphasizing sketchy metrics.
- Industry, provide data/access/collaboration?

Model: Plista challenges, NewsREEL data.

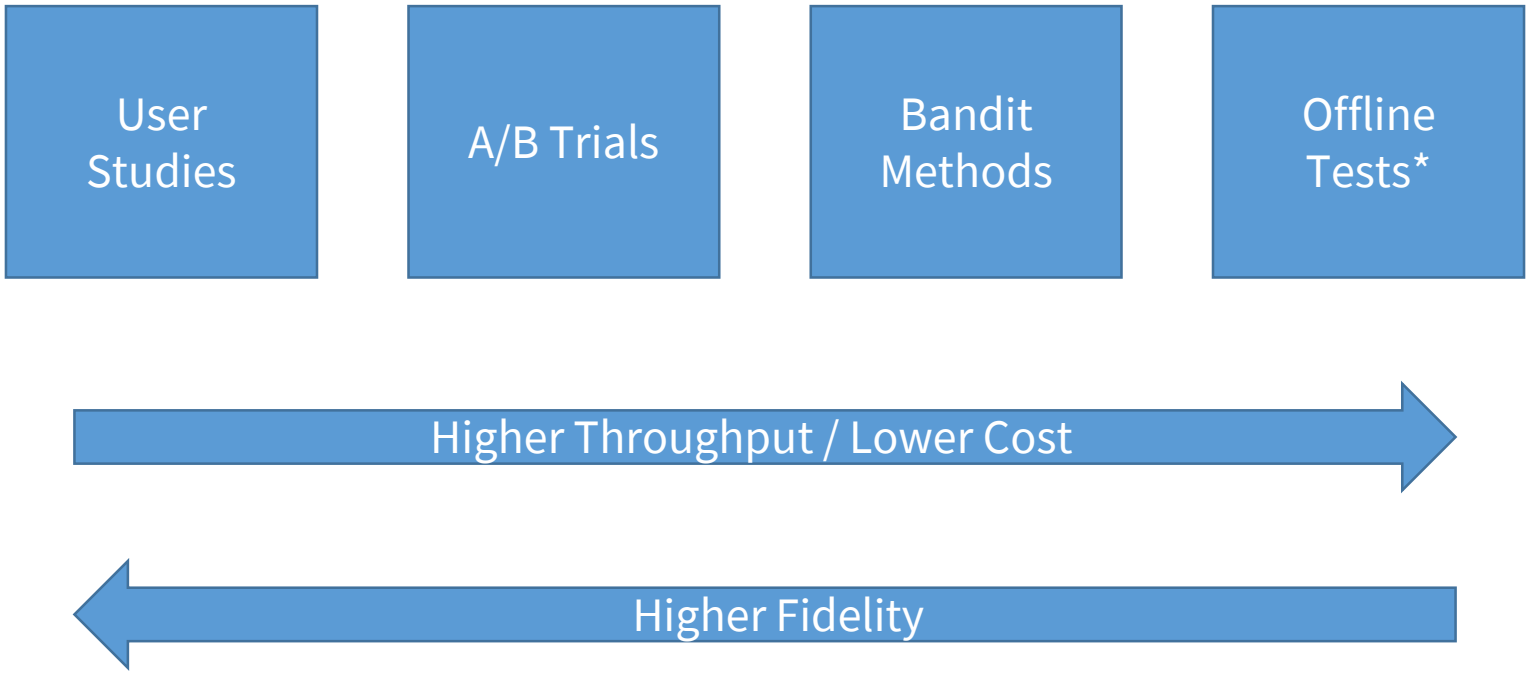
Complication: human subjects ethics standards

Ethics

Want a huge challenge?

Scale informed consent.

User-Centric Evaluations



On Bandits

- A/B tests based on time-tested, familiar scientific experimental methods
- Still very difficult to do well
- Bandits: often the scientific robustness seen as being in the way
- Result:
  - Very good for efficiently finding the best
  - Much harder to understand *why* it's the best

Why not?

- Easy to optimize key metric
- Hard to compare diagnostic and supplementary metrics in a scientifically valid fashion
- Producing generalizable knowledge is hard
- However: bandits are seeing use in clinical trials
  - So ‘not yet bandits’, not ‘not bandits’

What challenges?

- Increasing throughput of user studies
  - Maintain statistical validity
- Increasing fidelity of A/B tests and bandits
  - Instrumentation
  - Statistical robustness
  - What does a bandit outcome tell us?

What Do Users Want?

Or, moving past behaviorism.

Observing what users do is easy!

Understanding what they want is hard. *Really* want is harder yet.

Question: are users satisfied with their actions?

Opportunities

- The *intention-behavior gap*
- Can recommenders help bridge this gap?
- At scale?

Don’t recommend booze to an alcoholic?

What are failures?

- Who is lost/hurt/offended?
  - Small incidents
  - Possibly unique
  - Have disproportionate impact (ragequitting with angry tweets, #UnitedBreaksGuitars)
  - Business rules can help (don’t recommend tweeting health products)
  - Empathic design by diverse team is important
- How can we scale ‘do no harm’?

To sum up

- Understanding what our systems are doing is hard
- Understanding deep behavior impact is very hard
- Testing many things is hard but doable
- Need ongoing industry-academic conversation
  - This conference does that – love it