Algorithmic Social Intervention

Bryan Wilder
Center for Artificial Intelligence in Society
University of Southern California
Social and behavioral intervention

- Social problems: disease, poverty, homelessness...
- Intervention: services, outreach, education
- Limited resources
Motivating question

How can AI be used to improve socially impactful decisions?
Technical focus

- Improving interventions is often a *combinatorial optimization* problem
  \[
  \max_{S \in X} f(S) \quad X \subseteq \{0,1\}^n
  \]

- Select from discrete sets of objects
  - Peer leaders from a social network
  - Assign housing to applicants
  - Patients for extra follow-up
  - ...

- Resources are limited: intervention is subject to constraints
Challenge

• We understand *fully specified* combinatorial optimization problems

\[
\max_{S \in X} f(S, \theta)
\]

• But in most social good domains, we don’t know \( \theta \)!
Research question

How do we solve combinatorial optimization problems which depend on unknown parameters?
Research question

How do we solve combinatorial optimization problems which depend on unknown parameters?

Unique Challenges:
- Unified approach to data/learning and decision making
- Closing the loop from the lab to field evaluation
Robust optimization

Information gathering

Decision-focused learning

Field deployment

Little data

Gather more data

Data-driven decisions

AAAI-17b
AAAMAS-17a
AAAMAS-18a
AAAMAS-18c

AAAI-18b
AAAI-18c
AAAI-18d
AAAMAS-18b
CPAIOR-18

AAAI-18a
AAAMAS-18a
AAAI-19
Field immersion

• Start from real problem
• Work closely with domain experts
• Translate algorithms into practice
Outline

• Deployed application: HIV prevention
  • Have applied similar ideas to obesity prevention, public housing allocation, tuberculosis prevention, etc.

• Robust optimization: handling model uncertainty

• Information gathering: sampling to learn the network

• Field results: deployment with two LA-area drop in centers

• Recent work: decision-focused learning
Example: HIV and homelessness

- 6,000 homeless youth
- 10x HIV prevalence vs general population

Mayor Eric Garcetti: “the moral and humanitarian crisis of our time”
Example: HIV and homelessness

- Shelters conduct educational interventions
- Resource constraints: work with 4-6 youth at a time
- *Peer leaders*: spread message through social network
Example: HIV and homelessness

- Limited budget for total peer leaders trained
- Which nodes lead to greatest influence spread?
- Influence maximization problem
Computational problem

- Limited budget of seed nodes to recruit from a graph $G = (V, E)$
- For $S \subseteq V$, let $f(S, \theta)$ be the expected number of nodes reached when $S$ is recruited as seeds ($\theta = \text{model parameters}$)
- Problem:
  $$\max_{|S| \leq k} f(S, \theta)$$
Independent cascade model

- Most common model in the literature
- Each edge \((u, v)\) has a propagation probability \(p_{u,v}\)
- When \(u\) is influenced, \(v\) is influenced w.p. \(p_{u,v}\)
- \(\theta = (p_{1,1}, p_{1,2}, ... )\)
Background: submodularity

**Diminishing returns:**

\[
f(A \cup \{v\}) - f(A) \leq f(B \cup \{v\}) - f(B) \quad \forall v, \quad B \subseteq A
\]

**Theorem** [Nemhauser, Wolsey, Fisher 1978]: The greedy algorithm obtains a \(1 - \frac{1}{e}\)-approximation for maximizing a monotone submodular function subject to cardinality constraint.
Background: submodularity

- Alternate approach: continuous relaxation $F$
- Fractional decision variable $x$
- $x_i =$ probability include node $i$
- $F(x) = E_{S \sim x}[f(S)]$
- Maximizing $F$ in continuous space + rounding also gives $\left(1 - \frac{1}{e}\right)$-approximation
Influence maximization in the field

Previous work applies these techniques to influence maximization...


But assumes model is known exactly!
Influence maximization in the field

• What happens when we don’t know how influence propagates?
  • Little-to-no data available about homeless youth populations

• Or what the structure of the social network is?
  • Gathering network data requires in-person surveys, week+ of effort
Influence maximization in the field

• Together with social work partners, developed and deployed algorithms addressing these issues

• More than doubled the intervention’s impact compared to status quo
Outline

• Deployed application: HIV prevention
  • Have applied similar ideas to obesity prevention, public housing allocation, tuberculosis prevention, etc.

• Robust optimization: handling model uncertainty
• Information gathering: sampling to learn the network
• Field results: deployment with two LA-area drop in centers
• Recent work: decision-focused learning
Robust optimization

- Given candidate objective functions $f_1 \ldots f_m$ induced by different models, solve

$$\max \min_{|S| \leq k} f_i(S)$$

$\quad f_1 \quad f_2 \quad f_3 \quad \ldots$
Robust optimization

• Max-min is much more difficult than a single submodular function...
  • NP-hard to even approximate, must relax
Robust optimization

- Max-min is much more difficult than a single submodular function...
  - NP-hard to even approximate, must relax
- And for influence maximization, $m$ can be exponentially large!

\[ m = 2^{|E|} \]
Robust optimization

- Max-min is much more difficult than a single submodular function...
  - NP-hard to even approximate, must relax
- And for influence maximization, $m$ can be exponentially large!
- Existing approaches, building on greedy, are either heuristic or take exponential time [Krause et al 2008, He and Kempe 2017]
Robust optimization

- Main contribution: first polynomial-time approximation algorithm for robust submodular optimization with runtime independent of $m$
- Just requires “adversary oracle” (find worst case function)
- Empirically: 10-100x speedup over previous best heuristics
Approach

- Relax to zero-sum game:
  \[
  \max_{p: \text{distribution over sets}} \min_{i=1...m} E_{S \sim p}[f_i(S)]
  \]

- Now, optimizing over *continuous* distribution
Approach

• Relax to zero-sum game:

$$\max_{p: \text{distribution over sets } i=1\ldots m} \min_{S \sim p} E_{S \sim p} [f_i(S)]$$

• Now, optimizing over continuous distribution
• Let $F_1 \ldots F_m$ be the continuous relaxation of each objective
• Strategy: maximize $G(x) = \min_{i=1\ldots m} F_i(x)$
Approach

• Relax to zero-sum game:
  \[
  \max_{p: \text{distribution over sets } i=1\ldots m} \min_{S \sim p} E_S \left[f_i(S)\right]
  \]

• Now, optimizing over continuous distribution

• Let \( F_1 \ldots F_m \) be the continuous relaxation of each objective

• Strategy: maximize \( G(x) = \min_{i=1\ldots m} F_i(x) \)

• Key technical contribution: extending continuous approaches to max-min allows exponential speedup for large \( m \)
EQUATOR Algorithm

• Apply gradient-based method to the function \( G(x) = \min F_i(x) \)
  • Frank-Wolfe algorithm

• Get a (super)gradient of \( G \) just by finding the minimizing \( F_i \)
  • Never need to evaluate all gradients explicitly!

• Technical issues (see paper for details):
  • Controlled random perturbations ensure smoothness of gradients
  • Use correlation gap to design rounding procedure
EQUATOR Algorithm

\[
\begin{bmatrix}
0 & 0 \\
1 & 0 \\
0 & 1
\end{bmatrix}
\]
EQUATOR Algorithm

[0 0] [0 1] [1 0]
EQUATOR Algorithm

\[[0 \ 0]\]
\[[0.1 \ 0.2]\]
\[[0 \ 1]\]
\[[1 \ 0]\]
EQUATOR Algorithm

[0 0]

[0.1 0.2]

[1 0]

[0 1]

\( \nabla F_1 \)

\( \nabla F_2 \)

\( \nabla F_3 \)

\( \nabla F_4 \)
EQUATOR Algorithm

[0 0]

[0.1 0.2]

[0 1]

Adversary oracle: current lowest value is $F_3$
EQUATOR Algorithm

Adversary oracle: current lowest value is $F_3$
Approximation guarantee

**Theorem:** EQUATOR gives a $\left(1 - \frac{1}{e}\right)$-approximation to the optimal solution of the robust problem.

Robust optimization conclusion

- Nuanced models of influence spread are computationally involved
- Scalable algorithms via continuous relaxation
- For a given graph, enables planning under modeling uncertainty
- Next: how do we get that graph?
Outline

• Deployed application: HIV prevention
  • Have applied similar ideas to obesity prevention, public housing allocation, tuberculosis prevention, etc.

• Robust optimization: handling model uncertainty

• Information gathering: sampling to learn the network
  • Field results: deployment with two LA-area drop in centers
  • Recent work: decision-focused learning
Where does the network come from?

Assumed starting point

Real starting point

?
Where does the network come from?

- Data collection is costly and time consuming
  - Digital sources are often inaccurate or missing
  - Week+ for social workers to interview 100 or more people
Where does the network come from?

- Data collection is costly and time consuming
  - Digital sources are often inaccurate or missing
  - Week+ for social workers to interview 100 or more people
- Do we really need to gather the entire network?
Network sampling
Network sampling
Network sampling
Objective

• Query cost: how many nodes were surveyed?
  • Should grow very slowly with \( n \) (# nodes)
• Influence spread: what is the expected number of nodes reached?
• Comparison to \( OPT \), best influence spread by algorithm which sees entire network

\[
\text{approx. ratio} = \frac{E[\text{algorithm's influence spread}]}{OPT}
\]
Hardness

**Theorem:** There is a family of graphs on which any algorithm with strictly sublinear query cost has approximation ratio tending to 0 as $n \to \infty$
**Theorem:** There is a family of graphs on which any algorithm with strictly sublinear query cost has approximation ratio tending to 0 as $n \to \infty$
What now?

- Real networks have useful structure
- Here: two examples
  - Community structure
  - Friendship paradox
Community structure

- Intuition: influence mostly spreads locally, within communities
- We’d like to put one seed in each of the largest $k$ communities
Community structure

- ARISEN algorithm repeatedly:
  - Randomly samples a node
  - Explores that node’s neighborhood via a random walk
  - Estimates the size of that node’s community
- And then seeds nodes that correspond to largest $k$ communities
Community structure

**Theorem:** For community-structured graphs, ARISEN obtains a constant-factor approximation to the optimal influence spread using polylog(n) queries.

Community structure

**Theorem:** For community-structured graphs, ARISEN obtains a constant-factor approximation to the optimal influence spread using polylog(n) queries.

Asymptotically: exponential improvement over exhaustive surveys!

Community structure

- Downside: difficult to implement in some settings
- Homeless youth: can’t find a series of 5-10 youth to simulate a random walk
Friendship paradox

- On average, your friends are more popular than you
Friendship paradox

- Repeatedly
  - Survey a random node
  - Survey one of its neighbors
- First step encourages diversity, second biases towards high-degree/central nodes
Putting it all together

• Combine these ideas into a single system which works in the field
• Needs to minimize need for data, expertise, resources
• Needs to handle domain-specific challenges
  • Homeless youth: peer leaders often don’t attend intervention
Outline

- Deployed application: HIV prevention
  - Have applied similar ideas to obesity prevention, public housing allocation, tuberculosis prevention, etc.
- Robust optimization: handling model uncertainty
- Information gathering: sampling to learn the network
- Field results: deployment with two LA-area drop in centers
  - Recent work: decision-focused learning
Field study

Deployment in collaboration with social work and community partners
• Recruit ~60 youth
• Survey social network
• Train 10-12 peer leaders over 3 interventions
• 1 month follow-up survey
Comparison

• Conducted (so far) 4 studies, each with different algorithm
• Status quo: degree centrality (DC)
• AI-based algorithms: CHANGE, HEALER, DOSIM
  • CHANGE only surveys ~20% of nodes
  • HEALER and DOSIM survey 100%
Results: information spread

- AI-based algorithms dramatically outperform status quo (27% → 70+%)
- CHANGE performs comparable to HEALER/DOSIM, but surveyed only 18% of youth!
Results: behavior change

- Information spread translates into real behavior change!
- CHANGE: comparable/slightly higher conversion rate
Outline

- Deployed application: HIV prevention
  - Have applied similar ideas to obesity prevention, public housing allocation, tuberculosis prevention, etc.
- Robust optimization: handling model uncertainty
- Information gathering: sampling to learn the network
- Field results: deployment with two LA-area drop in centers
- Recent work: decision-focused learning
Results: behavior change

- Information spread translates into real behavior change!
- CHANGE: comparable/slightly higher conversion rate

Percent of informed nodes who started HIV testing

![Graph showing the percent of informed nodes who started HIV testing with categories: CHANGE, HEALER, DOSIM, DC]
Outline

• Deployed application: HIV prevention
  • Have applied similar ideas to obesity prevention, public housing allocation, tuberculosis prevention, etc.

• Robust optimization: handling model uncertainty
• Information gathering: sampling to learn the network
• Field results: deployment with two LA-area drop in centers

• Recent work: decision-focused learning
Decision-focused learning

- Previously: coping with limited data
- Now, switch to settings where data is available
- How can machine learning support decision making?
- Example: could we use administrative data to infer the social network?
  - Program co-attendance, check-in times, etc.
Learning + optimization problem

Observe data

Predictive model

Infer parameter $\theta$

Optimization algorithm

$$\max_{|S| \leq k} f(S, \theta)$$
Typical two-stage approach

Machine learning models
- Neural network
- Gaussian process
- Logistic regression
- Random forest

Goal: maximize accuracy

Optimization algorithms
- Greedy
- Local search
- Mixed-integer program
- LP relaxation

Goal: maximize decision quality
Challenge

• Maximizing accuracy ≠ maximizing decision quality
• “All models are wrong, some are useful”
• Two-stage training doesn’t align with end goal
Two-stage training

Node features

Predictive model

Predicted graph

vs

Real graph

Update model to make predicted graph closer to real
Decision-focused learning

1 0 0 0 0
0 1 0 0 0
0 0 0 0 1
0 0 0 1 0
1 0 0 0 0
0 1 0 0 0
0 1 0 0 0
0 1 0 0 0
0 0 0 0 1
1 0 0 0 0

Node features

Predictive model

Predicted graph

Optimization algorithm

Peer leaders

Update model to improve chosen peer leaders (wrt actual graph)
Approach

- Idea: differentiate optimal solution with respect to $\theta$, train model via gradient descent
  - *Similar approach recently used for convex optimization [Donti et al ’17]*

- Challenge: the optimization problem is discrete!
Approach

• Idea: differentiate optimal solution with respect to $\theta$, train model via gradient descent
  • Similar approach recently used for convex optimization [Donti et al ’17]
• Challenge: the optimization problem is discrete!
• Solution: relax to continuous problem, differentiate, round

Discrete

$$\max_{|S| \leq k} f(S, \theta)$$

$S =$ chosen set

Continuous

$$\max_{x: \sum_i x_i \leq k} F(x, \theta)$$

$x_i =$ probability item $i$ is chosen
Technical challenge

• How to compute $\frac{dx^*}{d\theta}$?
  • Differentiate the output of the optimization algorithm wrt predictions
Technical challenge

- How to compute $\frac{dx^*}{d\theta}$?
  - Differentiate the output of the optimization algorithm wrt predictions
- Idea: (locally) optimal solution must satisfy KKT conditions
- Differentiate those equations at optimum
Technical challenge

• How to compute $\frac{dx^*}{d\theta}$?
  • Differentiate the output of the optimization algorithm wrt predictions

• Idea: (locally) optimal solution must satisfy KKT conditions

• Differentiate those equations at optimum

• Provide appropriate relaxations/prove this works for:
  • Linear programming (bipartite matching, max flow, shortest path...)
  • Submodular maximization (influence maximization, facility location...)

Experiments

- On several domains, using real and synthetic data
- Compare decision-focused learning vs two-stage method
  - Solution quality and accuracy
Experiments

- On several domains, using real and synthetic data
- Compare decision-focused learning vs two-stage method
  - *Solution quality* and *accuracy*
- Decision-focused learning improves optimization by **15-70%**, but makes much less “accurate” predictions

<table>
<thead>
<tr>
<th></th>
<th>Budget allocation</th>
<th>Matching</th>
<th>Diverse recommendation</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>5</td>
<td>10</td>
<td>20</td>
</tr>
<tr>
<td>NN1-Decision</td>
<td>49.18 ± 0.24</td>
<td>72.62 ± 0.33</td>
<td>98.95 ± 0.46</td>
</tr>
<tr>
<td>NN2-Decision</td>
<td>44.35 ± 0.56</td>
<td>67.64 ± 0.62</td>
<td>93.59 ± 0.77</td>
</tr>
<tr>
<td>NN1-2Stage</td>
<td>32.13 ± 2.47</td>
<td>45.63 ± 3.76</td>
<td>61.88 ± 4.10</td>
</tr>
<tr>
<td>NN2-2Stage</td>
<td>9.69 ± 0.05</td>
<td>18.93 ± 0.10</td>
<td>36.16 ± 0.18</td>
</tr>
<tr>
<td>RF-2Stage</td>
<td>48.81 ± 0.32</td>
<td>72.40 ± 0.43</td>
<td>98.82 ± 0.63</td>
</tr>
<tr>
<td>Random</td>
<td>9.69 ± 0.04</td>
<td>18.92 ± 0.09</td>
<td>36.13 ± 0.14</td>
</tr>
</tbody>
</table>

80
Example

- Simulated network intervention task
- Variable to predict: probability each of a set of outreach channels will reach each network node
Example

Ground truth

Two-stage prediction

Decision-focused prediction
Explanation

Decision-focused

$\text{Ground truth}$

$\text{Predicted out-weight}$

$r^2 = 0.94$

Two-stage

$\text{Ground truth}$

$\text{Predicted out-weight}$

$r^2 = 0.64$
Explanation

Our method learns to focus on qualities needed for making good decisions!
Conclusion

• Critical social issues present optimization and learning problems
• Interventions for disease prevention, homelessness, and more can be transformed via computational techniques
Conclusion

• Deployable algorithms present new research challenges
• Must handle data and decisions in unified way
• With all the pieces put together, AI offers powerful means to improve our fellow humans’ lives