

Adversarial Statistical Relational AI

Daniel Lowd
University of Oregon

Outline

- Why do we need adversarial modeling?
 - Because of the dream of AI
 - Because of current reality
 - Because of possible dangers
- Our initial approach and results
 - Background: adversarial learning + collective classification
 - Robustness through adversarial simulation
[Torkamani & Lowd, ICML'13]
 - Robustness through regularization
[Torkamani & Lowd, ICML'14]

What is StarAI?

“Theoretically, combining logic and probability in a unified representation and building general-purpose reasoning tools for it has been the dream of AI, dating back to the late 1980s. Practically, successful StarAI tools will enable new applications in several large, complex real-world domains including those involving big data, social networks, natural language processing, bioinformatics, the web, robotics and computer vision. Such domains are often characterized by rich relational structure and large amounts of uncertainty. Logic helps to effectively handle the former while probability helps her effectively manage the latter. We seek to invite researchers in all subfields of AI to attend the workshop and to explore together the dream of AI: Unifying logic and probability!

Who is StarAI?

“Specifically, the workshop will encourage active participation from researchers in the following communities:

- satisfiability (SAT)
- knowledge representation (KR)
- constraint satisfaction and programming (CP)
- (inductive) logic programming (LP and ILP)
- graphical models and probabilistic reasoning (UAI)
- statistical learning (NIPS, ICML, and AISTATS)
- graph mining (KDD and ECML PKDD)
- probabilistic databases (VLDB and SIGMOD).”

[www.starai.org]

Who is StarAI?

“It will also actively involve researchers from more applied communities, such as:

- natural language processing (ACL and EMNLP)
- information retrieval (SIGIR, WWW and WSDM)
- vision (CVPR and ICCV)
- semantic web (ISWC and ESWC)
- robotics (RSS and ICRA).”

[www.starai.org]

Almost everyone doing
AI research!

Statistical Relational AI

- The real world is complex and uncertain
- Logic handles complexity
- Probability handles uncertainty

Adversarial Statistical Relational AI

- The real world is complex, uncertain, and adversarial
- Logic handles complexity
- Probability handles uncertainty
- Game theory handles adversarial interaction
- Include researchers in multi-agent systems (AAMAS) and security (CCS)

If you want to unify AI, why stop with logic and probability?

Outline

- Why do we need adversarial modeling?
 - Because of the dream of AI
 - **Because of current reality**
 - Because of possible dangers
- Our initial approach and results
 - Background: adversarial learning + collective classification
 - Robustness through adversarial simulation
[Torkamani & Lowd, ICML'13]
 - Robustness through regularization
[Torkamani & Lowd, ICML'14]

Example: Social Network Spam

Which users are spammers?

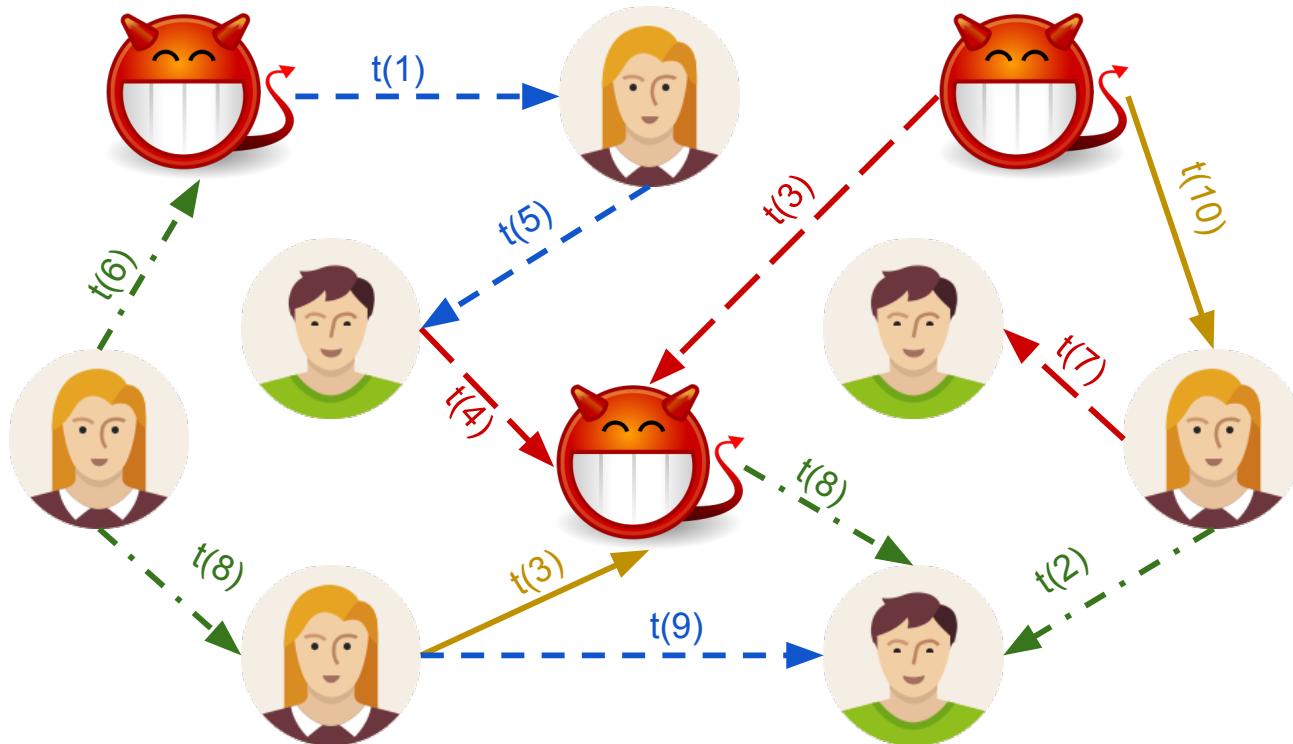


Image credit: [Fakhraei et al., 2015]

Example: Fraud Detection in Online Auctions

Which people are fraudsters or accomplices?

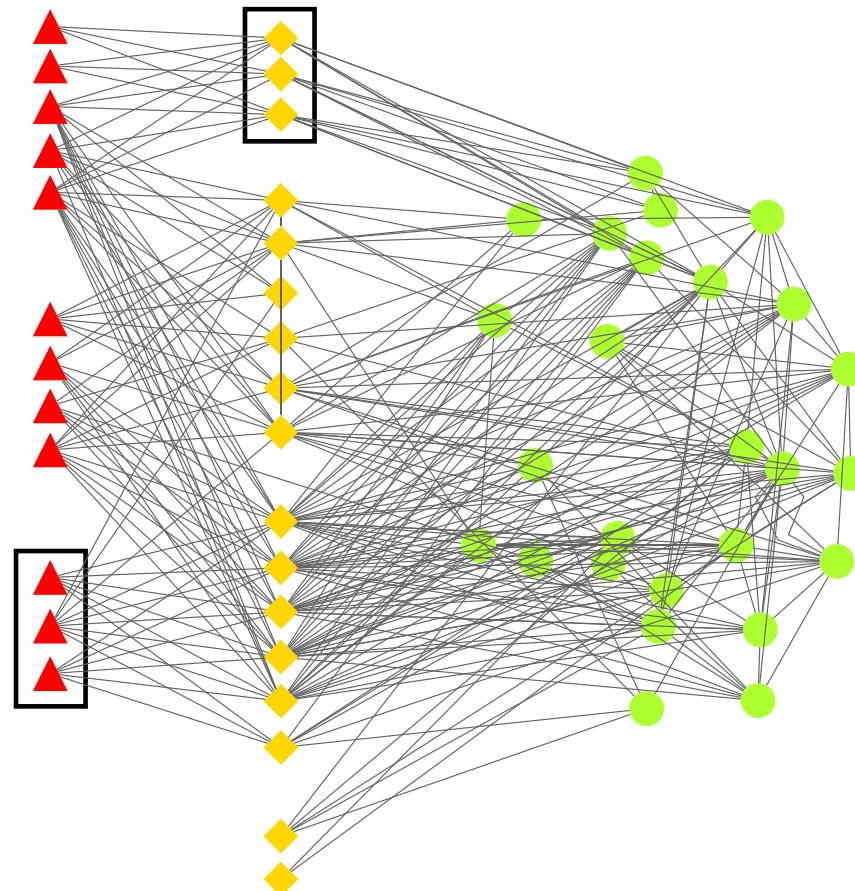


Image credit: [Chau&al06]

Example: Securities Dealers

Which brokers are likely to receive complaints in the future?

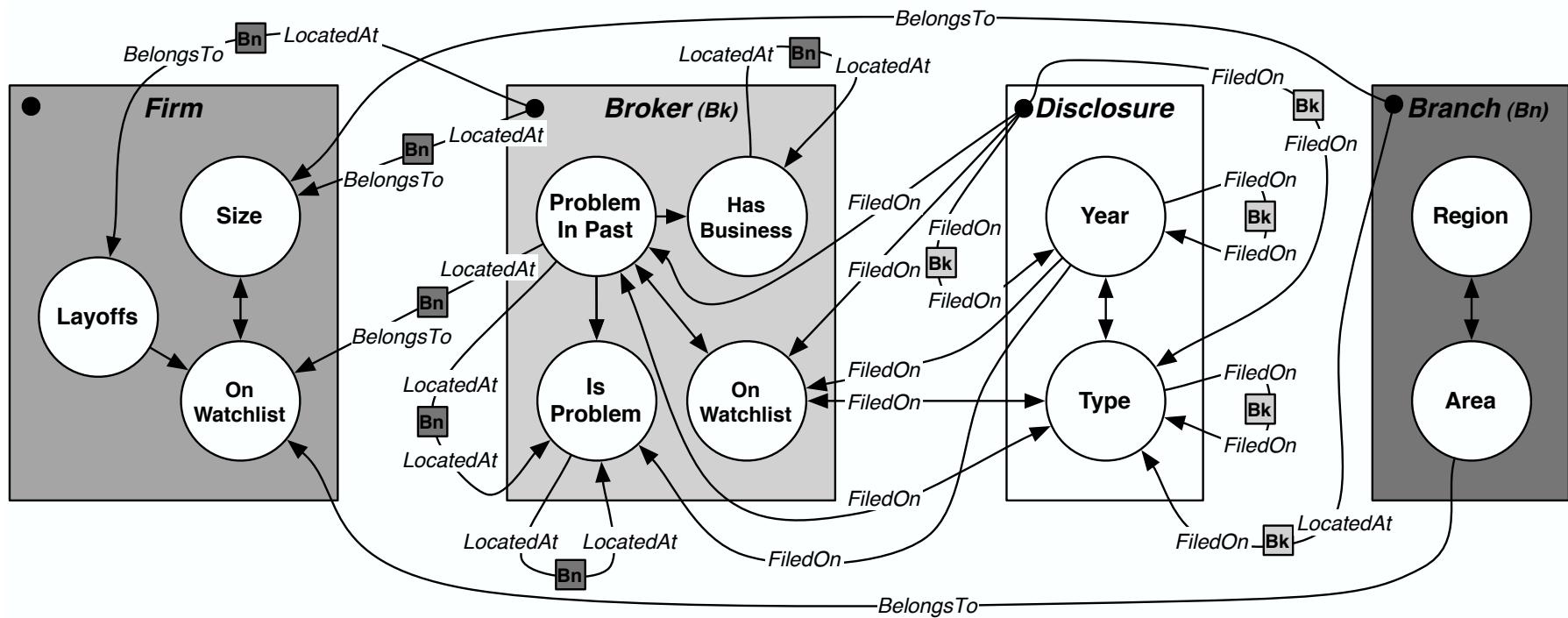


Figure 13: RDN for the NASD data set (1999).

Image credit: [Neville&Jensen07]

More Examples

- Web spam
- Worm detection
- Fake reviews
- Counterterrorism

Common themes:

1. Adversaries can be detected by their relationships as well as their attributes.
2. Adversaries may change their behavior to avoid detection.

Outline

- Why do we need adversarial modeling?
 - Because of the dream of AI
 - Because of current reality
 - **Because of possible dangers**
- Our initial approach and results
 - Background: adversarial learning + collective classification
 - Robustness through adversarial simulation
[Torkamani & Lowd, ICML'13]
 - Robustness through regularization
[Torkamani & Lowd, ICML'14]

Robustness and Safety in AI

- Many AI systems interact with people – this is a vulnerability and a liability.
- How can we know that an AI system is correct, safe, or robust?
- Adversarial reasoning and modeling can help build more robust systems by optimizing pessimistically.

Related Work on Multi-Agent StarAI

- Poole, 1997: Independent Choice Logic
- Rettinger et al., 2008: A Statistical Relational Model for Trust Learning
- Lippi, 2015: Statistical Relational Learning for Game Theory

Outline

- Why do we need adversarial modeling?
 - Because of the dream of AI
 - Because of current reality
 - Because of possible dangers
- Our initial approach and results
 - Background: adversarial learning + collective classification
 - Robustness through adversarial simulation
[Torkamani & Lowd, ICML'13]
 - Robustness through regularization
[Torkamani & Lowd, ICML'14]

Special Case: Static Prediction Games

- In general, we may have arbitrary agents, utility functions, and game structures. **Hard.**
- Prediction games
 - First player chooses the model (e.g., spam filter)
 - Second player chooses the test data (e.g., spam)
- Domains: Social network spam, online auction fraud, bad securities dealers, web spam, fake reviews, and more!

Example: Spam Filtering

1.

From: spammer@example.com

Cheap mortgage now!!!

A blue arrow points downwards from the email content box to a horizontal blue bar. This bar has a vertical tick mark on its left side, corresponding to the word 'cheap' in the list below.

cheap = 1.0
mortgage = 1.5
2.

Feature Weights

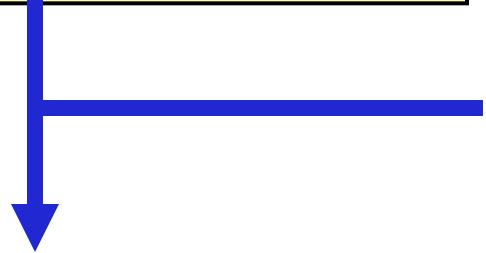
3. Total score = 2.5 > 1.0 (threshold)



Example: Spammers Adapt

1. 

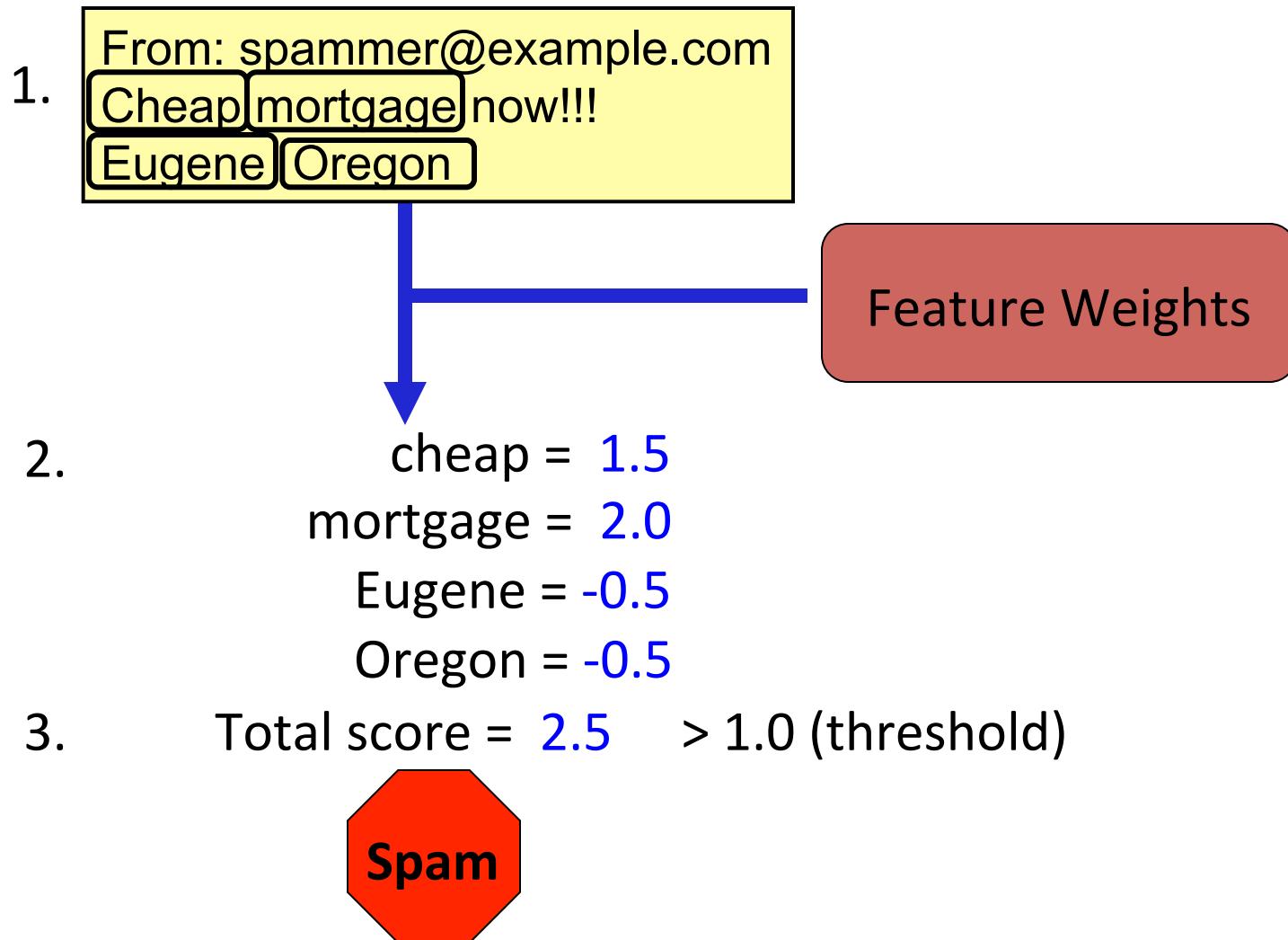
From: spammer@example.com
Cheap mortgage now!!!
Eugene Oregon


2. 

cheap = 1.0
mortgage = 1.5
Eugene = -1.0
Oregon = -1.0
3. Total score = 0.5 < 1.0 (threshold)



Example: Classifier Adapts



Adversarial Classification as a Game

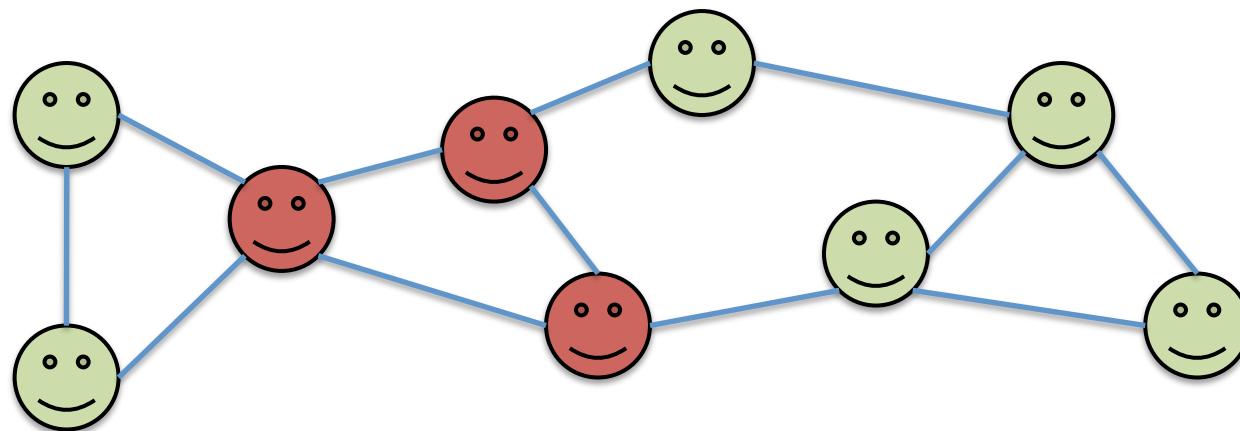
- Learner selects a classifier c .
- Adversary selects modified evidence x .
- Each receives a reward based on how correct the classifier was and how corrupt the evidence was.

Previous work: Assumes instances are independent!

(e.g., Dalvi&al04; Globerson&Roweis06; Teo&al08;
Dekel&Shamir08; Xu&al09; Brückner&Scheffer09;
Brückner&Scheffer11)

Collective Classification

Label a set of objects using the relationships among them as well as their attributes.



Markov Logic Networks

A **Markov Logic Network (MLN)** is a log-linear model where the features are counts of satisfied formulas. Given a finite set of constants, this defines a probability distribution over possible worlds:

$$\log P(x) = \sum_i w_i n_i(x) - A = w^T \phi(x) - A$$

Weight of formula i

No. of true groundings of formula i in x

Conditional distribution of query atoms (y) given evidence (x):

$$\begin{aligned} \log P(y | x) &= w^T \phi(x, y) - A(x) \\ &= score(w, x, y) - A(x) \end{aligned}$$

Markov Logic Networks for Collective Classification

1. The label for an object o depends on its attributes:

$$\text{HasAttribute}(o, +a) \Rightarrow \text{Label}(o, +c)$$

2. Related objects are more likely to have similar labels:

$$\text{Related}(o, o') \wedge \text{Label}(o, +c) \Rightarrow \text{Label}(o', +c)$$

Create copies of these rules for each class and attribute, and then learn a weight for each rule.

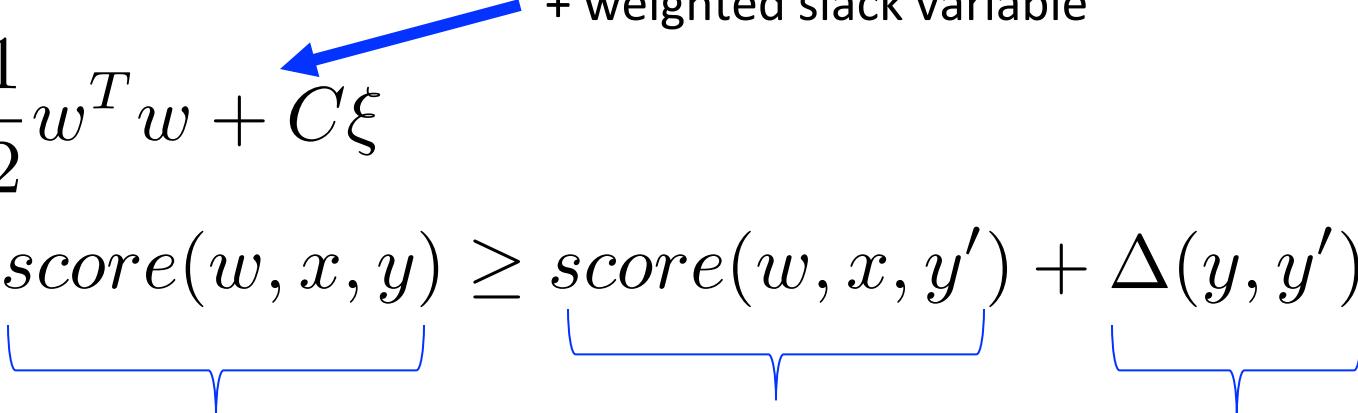
Max-Margin Weight Learning

Goal: Select w to maximize the margin between true labeling y and any alternate labeling y' .

$$\min_w \frac{1}{2} w^T w + C\xi$$

Maximize the margin
+ weighted slack variable

$$\text{s.t. } score(w, x, y) \geq score(w, x, y') + \Delta(y, y') - \xi \quad \forall y'$$



Score of the true labeling Score of an alternate labeling Differences between the labelings

Max-Margin Weight Learning

Goal: Select w to maximize the margin between true labeling y and any alternate labeling y' .

Learner's loss from the best
alternate labeling
(biggest margin violation)

$$\min_w \frac{1}{2} w^T w + C \left(\max_{y'} [score(w, x, y') - score(w, x, y) + \Delta(y, y')] \right)$$



L_2 regularizer
on the weights.

Special Case: Associative Markov Networks

- If the weights of the second formula are positive, then linked nodes are more likely to have the same label:



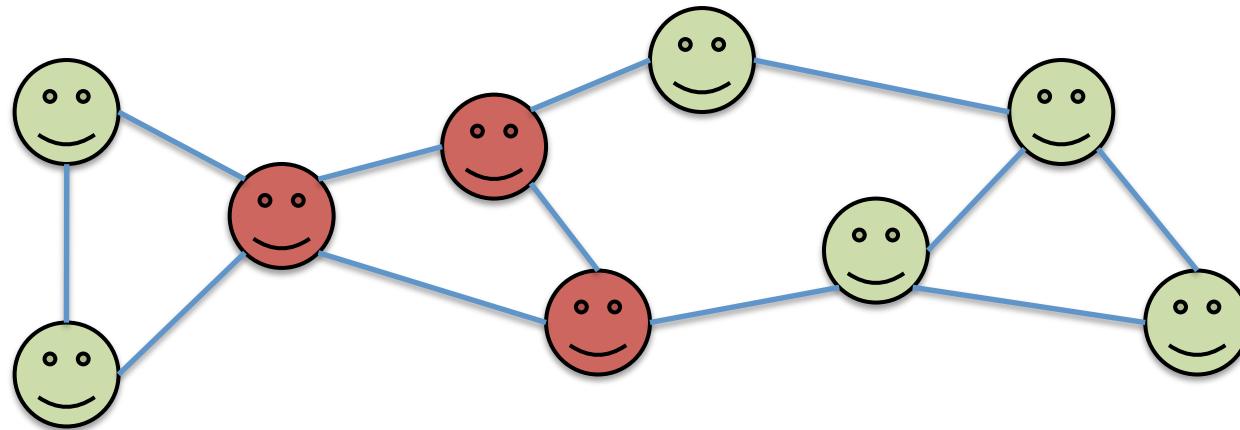
- Inference can be done in polynomial time with graph cuts or as a linear program.
[Kolmogorov&Zabin04]
- Learning can be done in polynomial time with a convex quadratic program. [Taskar&al04]

Outline

- Why do we need adversarial modeling?
 - Because of the dream of AI
 - Because of current reality
 - Because of possible dangers
- Our initial approach and results
 - Background: adversarial learning + collective classification
 - Robustness through adversarial simulation
[Torkamani & Lowd, ICML'13]
 - Robustness through regularization
[Torkamani & Lowd, ICML'14]

Adversarial Collective Classification

We want to robustly label related entities who are actively working to avoid detection.



Assume: Adversary can modify up to D attributes.
(e.g., add/remove words from spam web pages)

Convex Adversarial Collective Classification

Modify our associative Markov network by assuming a worst-case adversary:

$$\min_w \frac{1}{2} w^T w + C \left(\max_{x', y'} [score(w, x', y') - score(w, x', y) + \Delta(y, y')] \right)$$

$$\text{s.t. } \Delta(x, x') \leq D$$

Enforce a margin between true labeling and alternate labeling given worst-case adversarially modified data.

Convex Adversarial Collective Classification

Reformulating as a quadratic program:

1. Remove bilinearities in the score function by introducing auxiliary variables.
2. Replace the inner maximization with its dual minimization problem.

Theorem: For binary-valued labels and features, the adversary's maximization has an integral solution. Thus, the relaxed learning problem is exact.

(Can be extended with multiple types of relations, as long as all are associative.)

Datasets

- Political blogs
 - 2004 blog data collected by Adamic (2005)
 - We recrawled in February 2012 and May 2012 to add words, remove dead blogs.
 - Selected 100 words with mutual information
- Reuters
 - 4 classes from the ModApte split: crude, grain, trade, money-fx
 - Split into 7 time periods, each with 300-400 articles
 - Added links to 2 most similar articles (TF-IDF)
 - Selected 200 words with mutual information

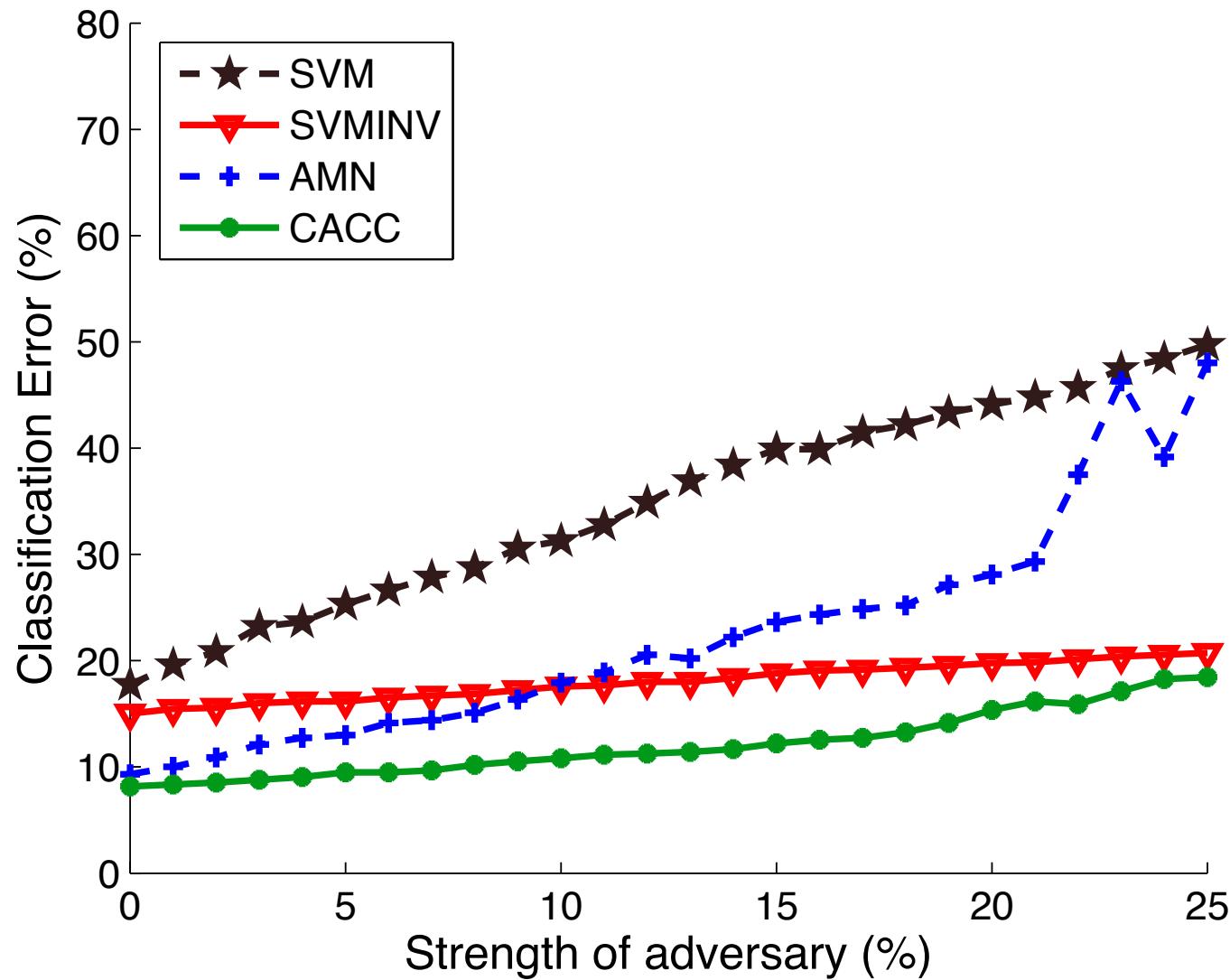
Experimental Methods

Method	Relational?	Adversarial?
Linear SVM	No	No
SVM-Invar [Teo&al08]	No	Yes
AMN [Taskar&al04]	Yes	No
CACC [Torkamani&Lowd13]	Yes	Yes

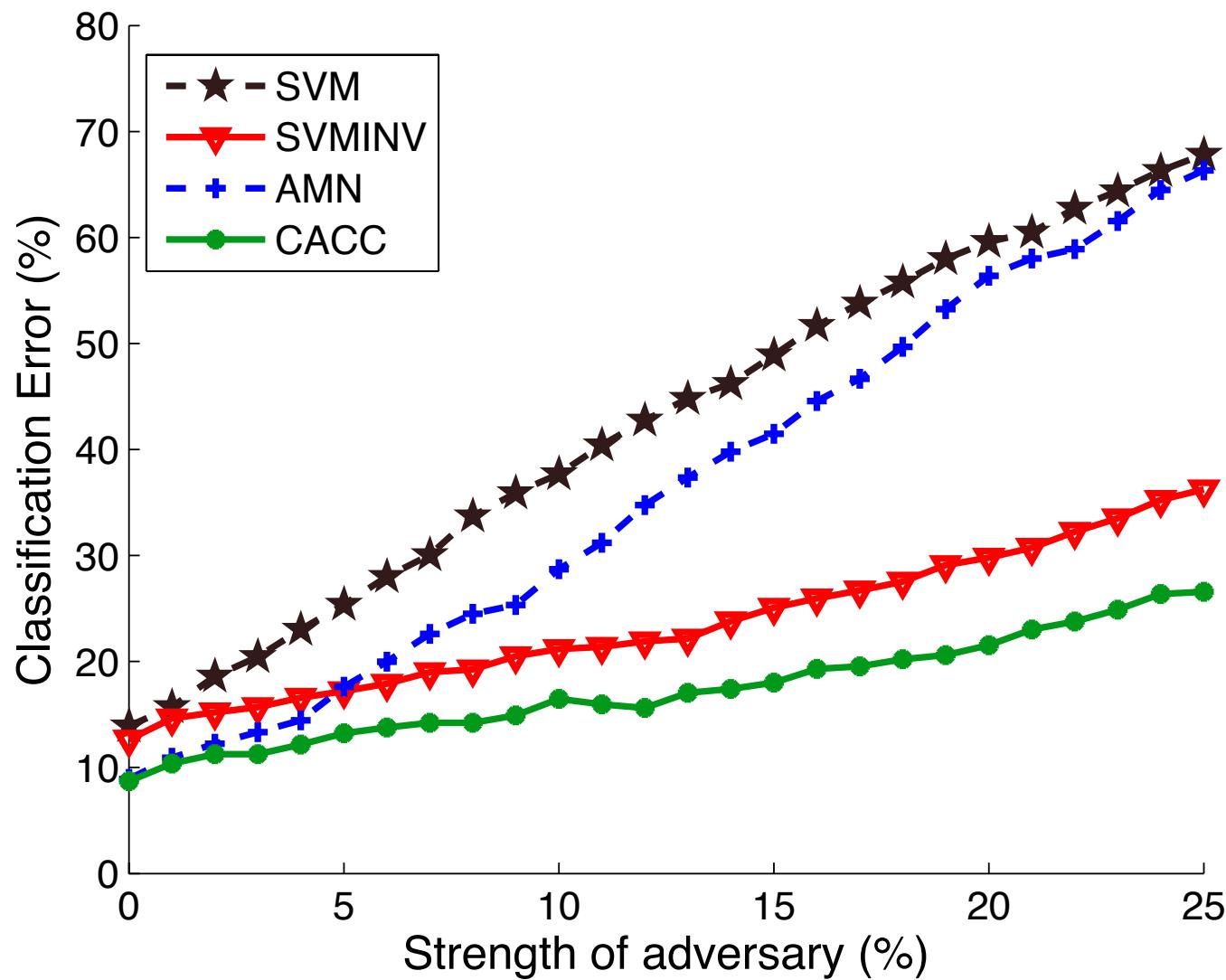
Tuning: Select parameters to maximize performance on validation data against adversary who could modify 10% of the attributes.

Evaluation: Measured accuracy on test data against simulated adversaries with budgets from 0% to 25%.

Results: Political Blogs, Tuned for 10% adversary

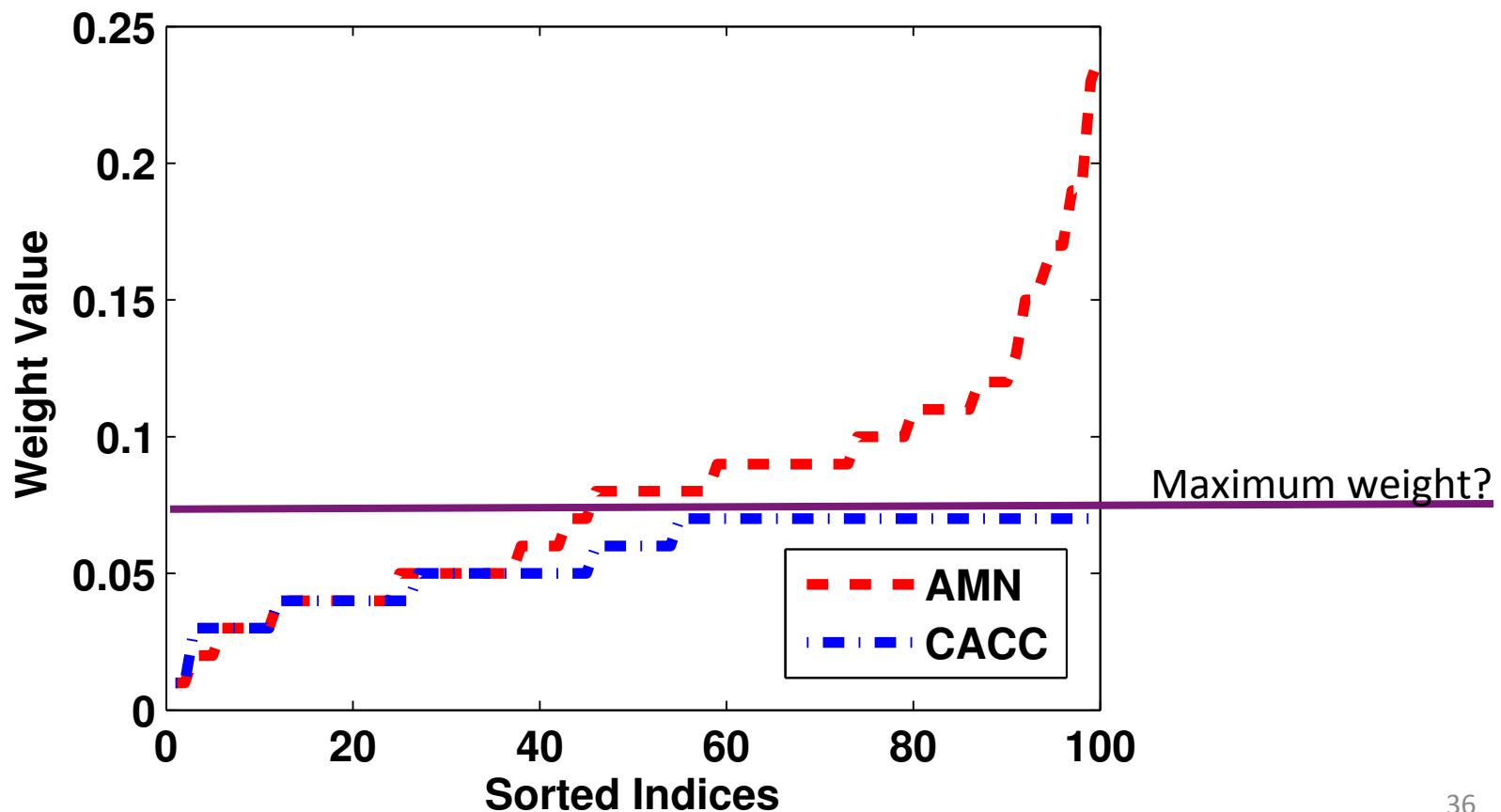


Results: Reuters, Tuned for 10% adversary



Weight Distribution

Intuitively, if an adversary can change some of the attributes
then we want to avoid placing high weights on any attributes.
CACC does this automatically:



Adversarial Regularization

- Empirically, optimizing performance against a simulated adversary can lead to bounded weights.
- What if we avoid simulating the adversary and instead just bound the weights?
- We can show that the two are equivalent!
(Under a slightly different adversarial model than we used before.)
- More generally, we can achieve adversarial robustness on any structured prediction problem by adding a regularizer.

Adversarial Model

- Previously, we assumed the adversary could modify the **evidence**, x , by a small number of changes.
- Now we assume that the adversary can modify the **feature vector**, $\phi(x,y)$, by a small vector $\delta/2$.
 - Thus, they can modify the **difference between two feature vectors**, $\phi(x,y') - \phi(x,y)$, by δ .
 - Thus, they can modify the **difference between two scores**, $\text{score}(w,x,y') - \text{score}(w,x,y)$, by $w^T\delta$.

Optimization Problem

$$\min_w \frac{1}{2} w^T w + C \max_{\delta \in S, y'} [score(w, x, y') - score(w, x, y) + w^T \delta + \Delta(y, y')]$$

Which is equivalent to:

$$\min_w \frac{1}{2} w^T w + \max_{\delta \in S} w^T \delta + C \max_{y'} [score(w, x, y') - score(w, x, y) + \Delta(y, y')]$$

Ellipsoidal Uncertainty

(c.f. [Xu et al., 2009] for robustness of regular SVMs.)

Suppose the adversary is constrained by a norm:

$$S = \{\delta \mid \|M\delta\| \leq 1\}$$

Theorem: Robustness over S is equivalent to adding the dual norm as a regularizer:

$$\begin{aligned} \min_w \frac{1}{2} w^T w + & \boxed{\|M^{-1}w\|} \\ & + C \max_{y'} [score(w, x, y') - score(w, x, y) + \Delta(y, y')] \end{aligned}$$

Special case: For L_1 ball, the dual norm is L_∞ (max).

Polyhedral Uncertainty

Suppose the adversary is constrained to a polyhedron:

$$S = \{\delta | A\delta \leq b\}$$

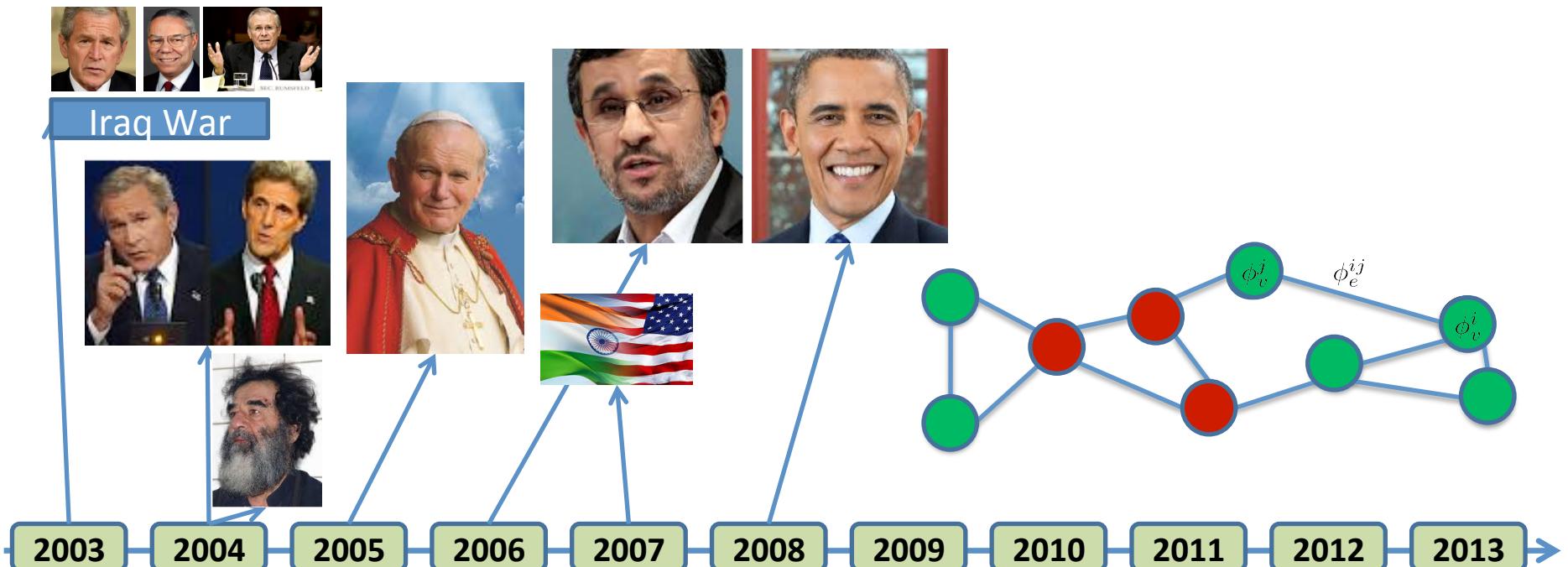
Theorem: Robustness over S is equivalent to adding a linear regularizer in a transformed weight space:

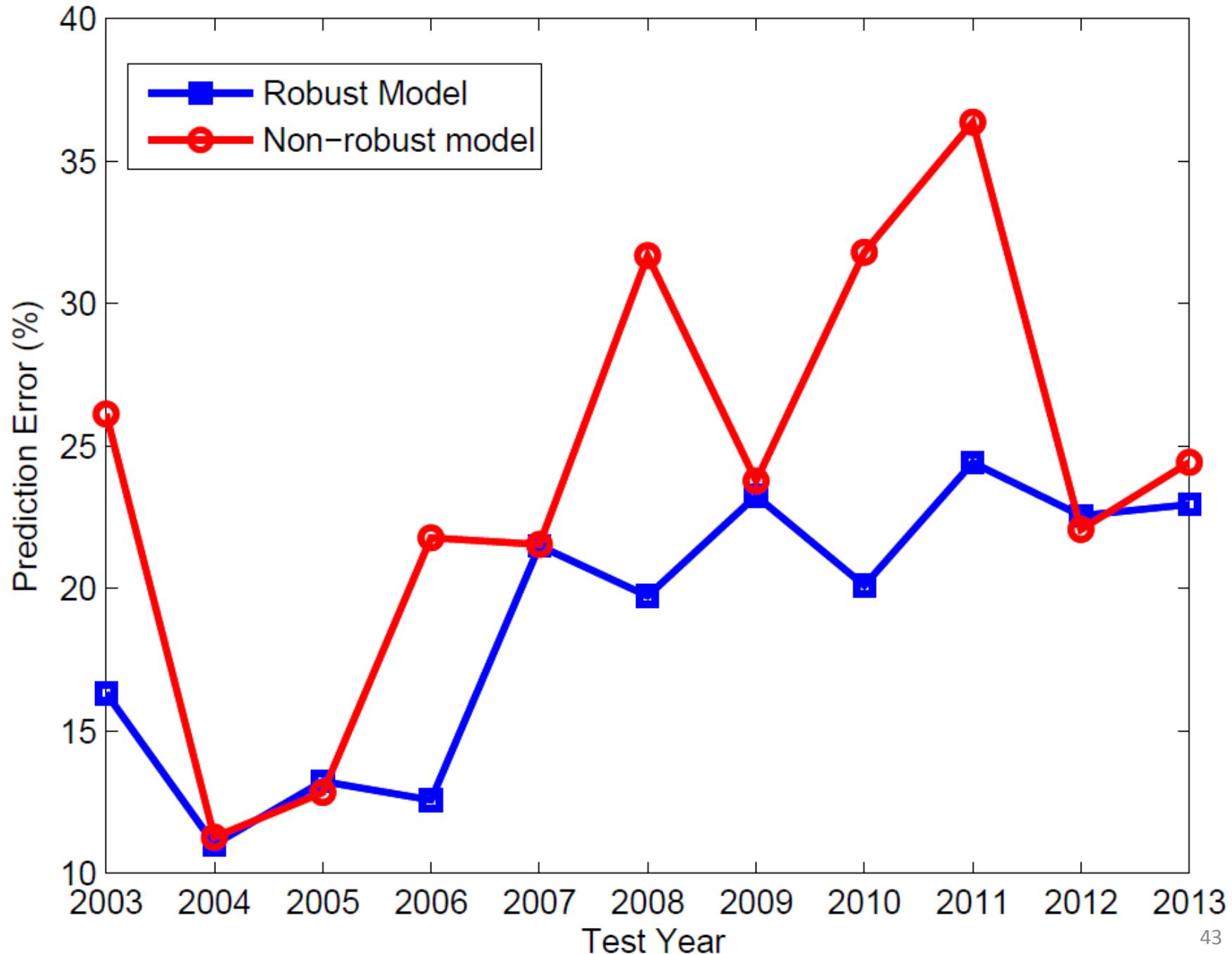
$$\begin{aligned} \min_{\lambda} \frac{1}{2} \lambda^T (AA^T) \lambda + & \boxed{b^T \lambda} \\ & + C \max_{y'} [\lambda^T (\boxed{A^T} \phi(x, y) - \boxed{A^T} \phi(x, y')) + \Delta(y, y')] \end{aligned}$$

We can also let S be the intersection of a polyhedron and an ellipsoid and obtain a generalization of both results.

Robustly Classifying 11 years of Political Blogs

- Goal: Label each blog as liberal or conservative
- Political blogs dataset (Adamic and Glance, 2005)
+ bag-of-words features from each year
- Train/tune on 2004 and test on every year.
- Robust model: Assume adversary can modify up to k words and k links.





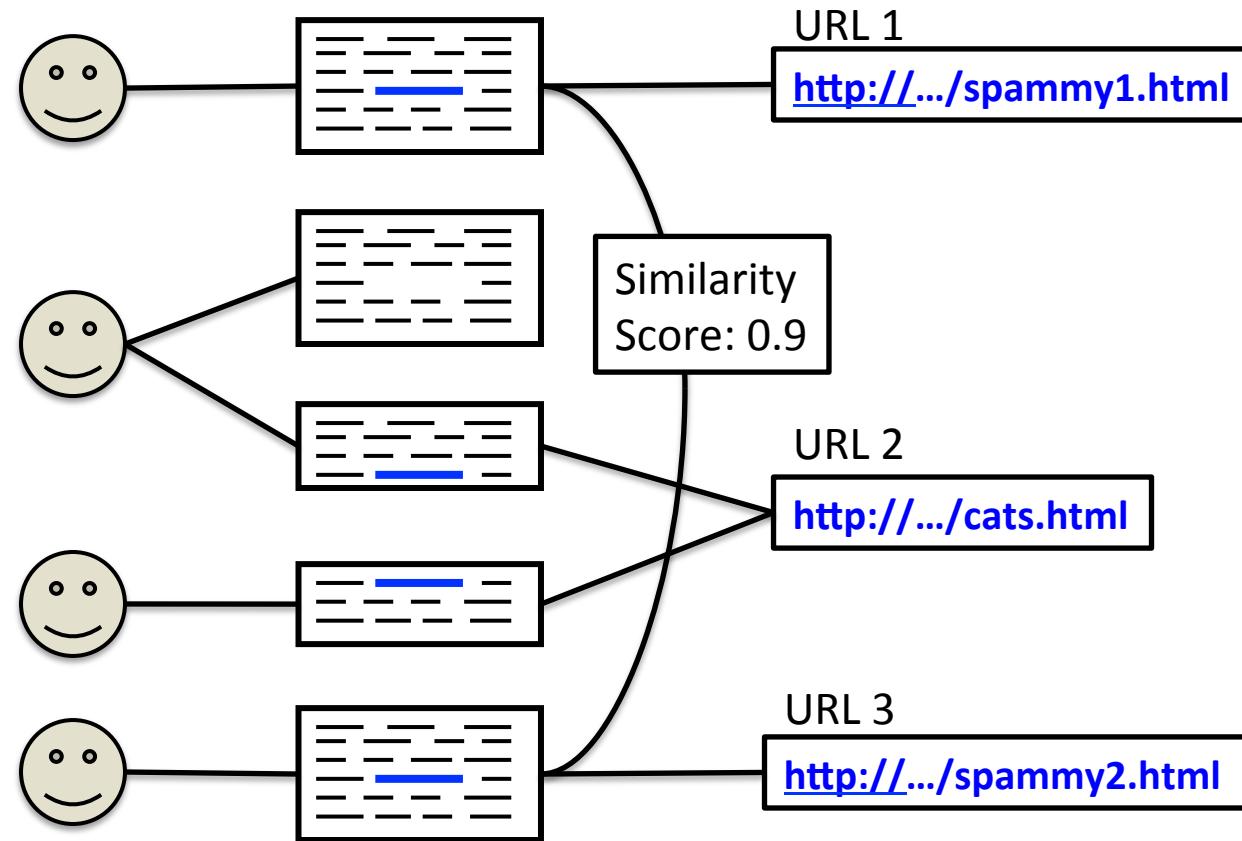
Ongoing Work: Large-Scale Applications

- Comment spam on YouTube
- Abuse and spam on SoundCloud
- Social network spammers on Tagged.com
- Fraudulent images (DARPA MediFor program)

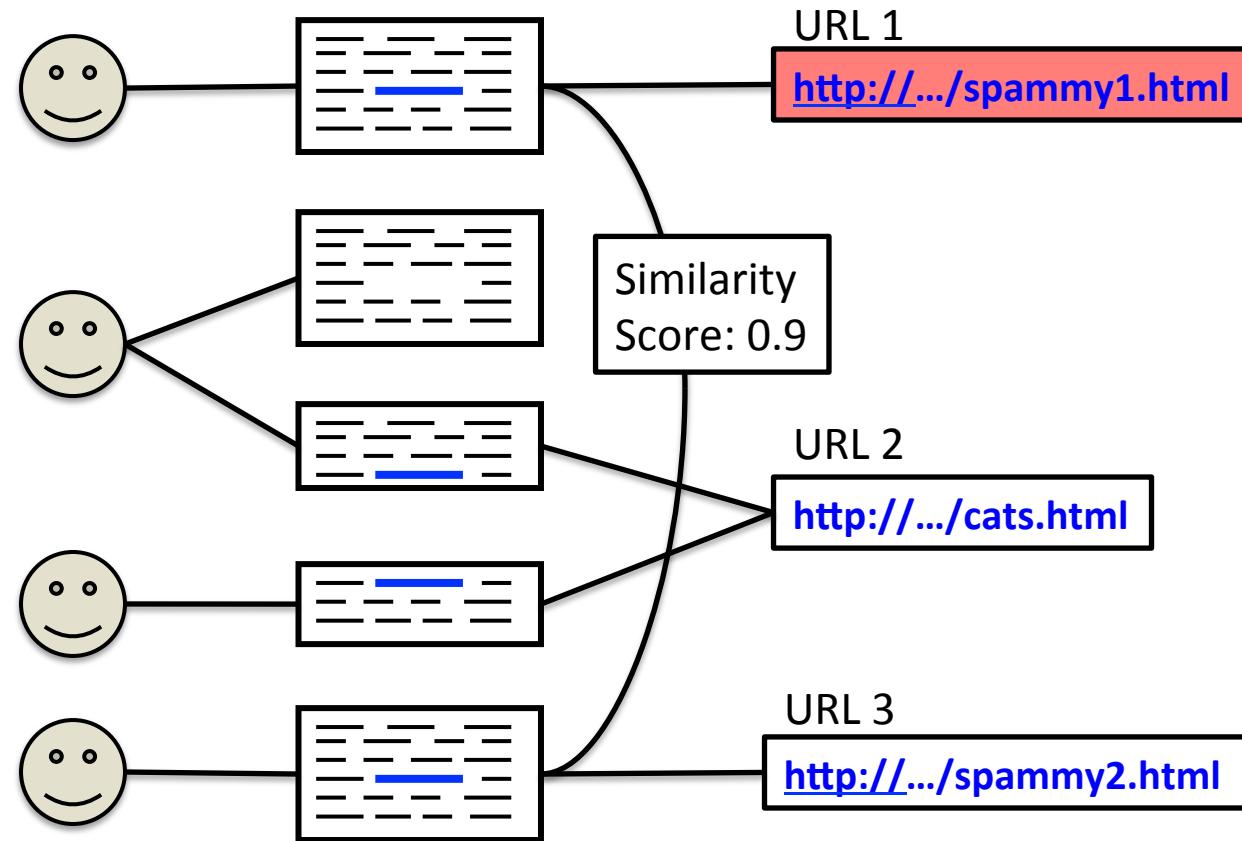
Challenges:

- Multiple types of relations
- Complex adversaries
- Millions of objects to label

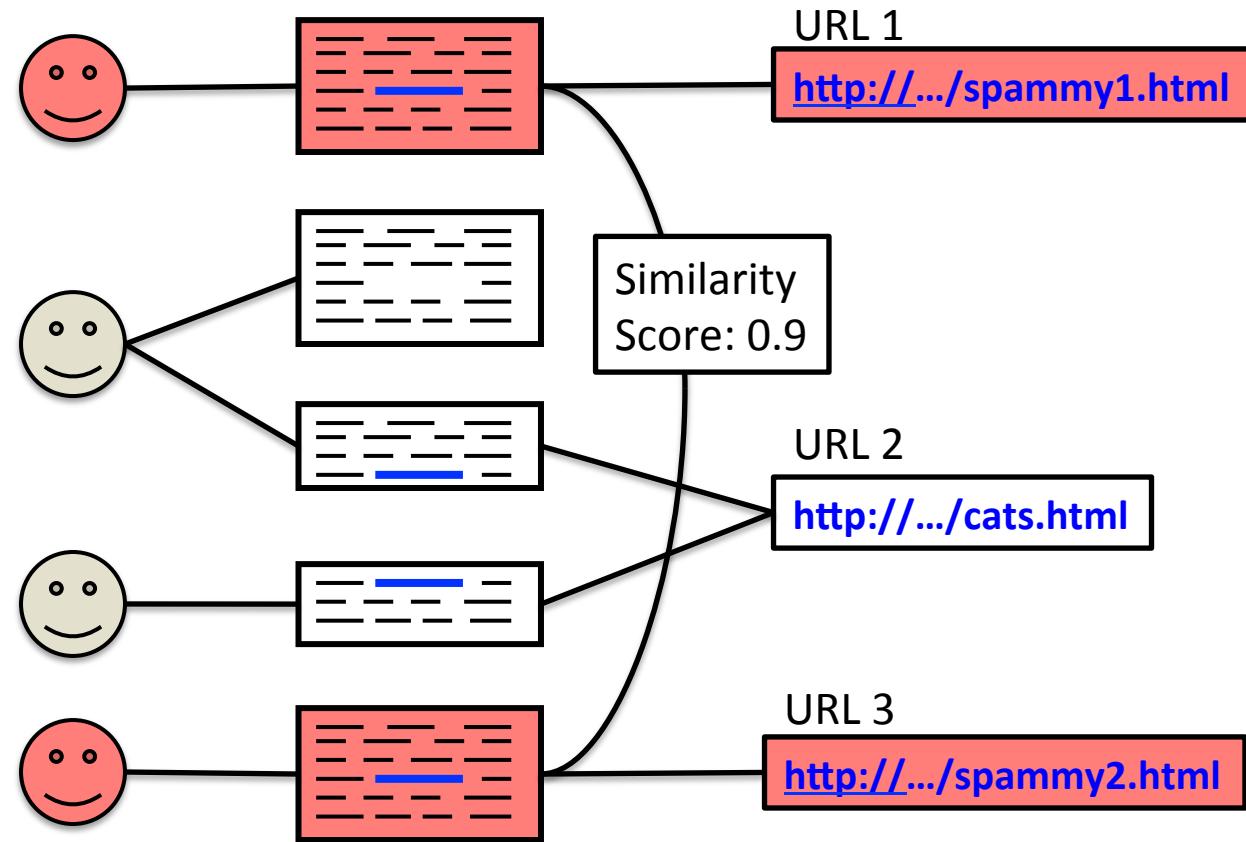
Detecting Spamy Comments



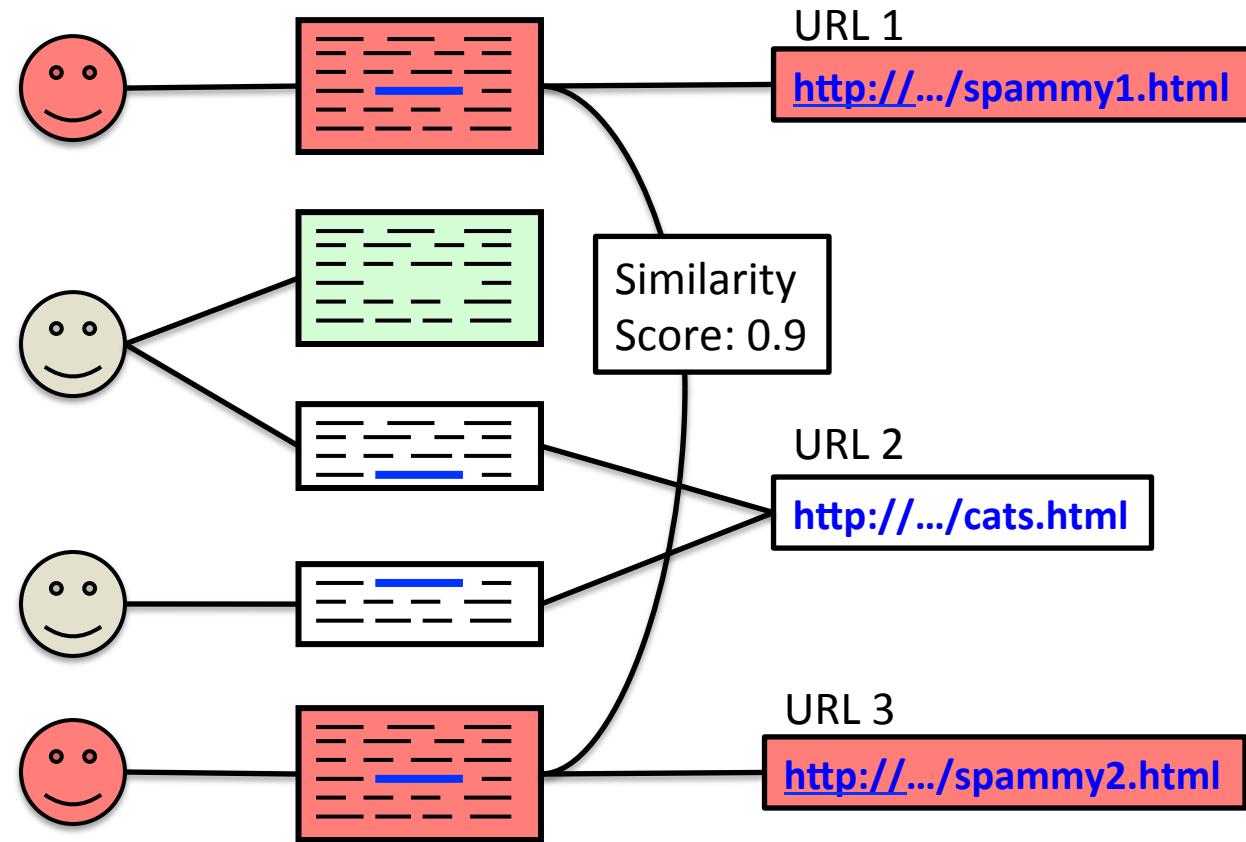
Detecting Spamy Comments



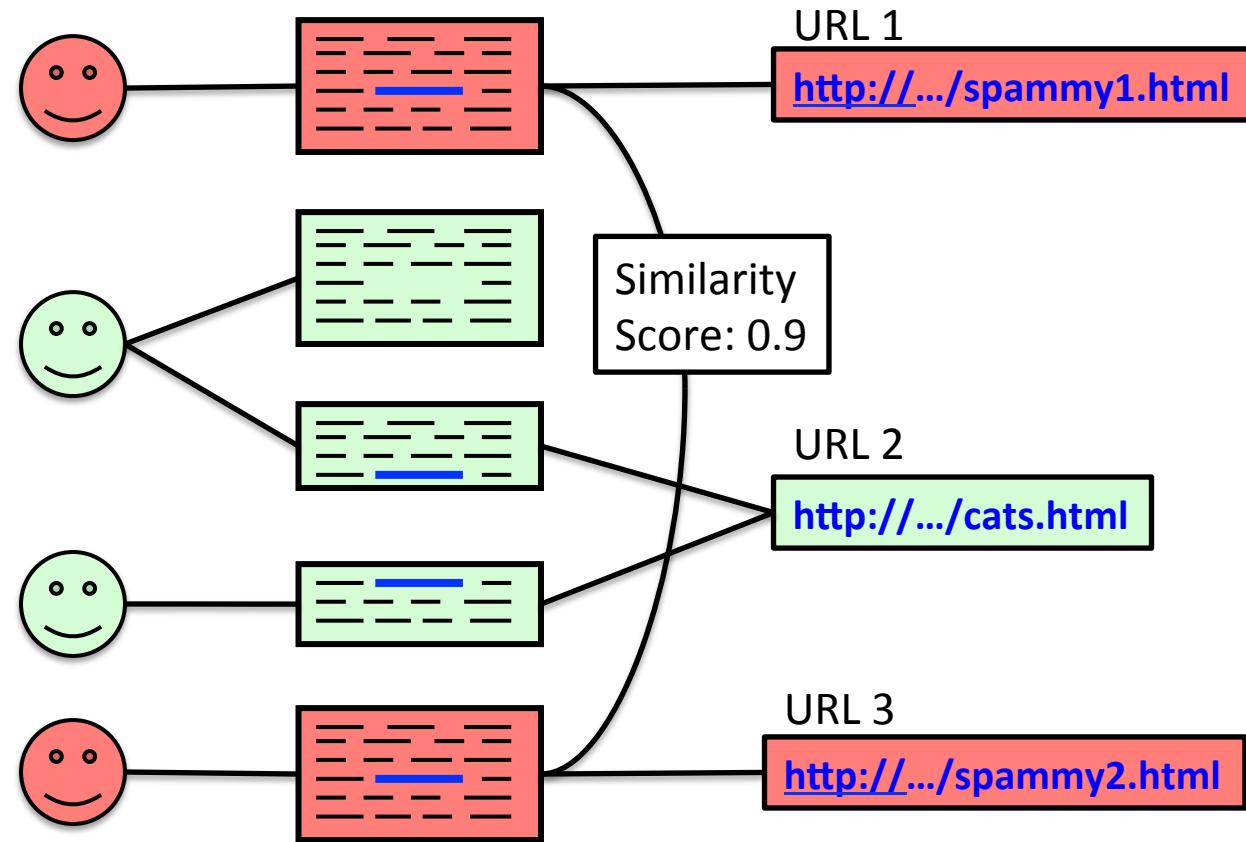
Detecting Spamy Comments



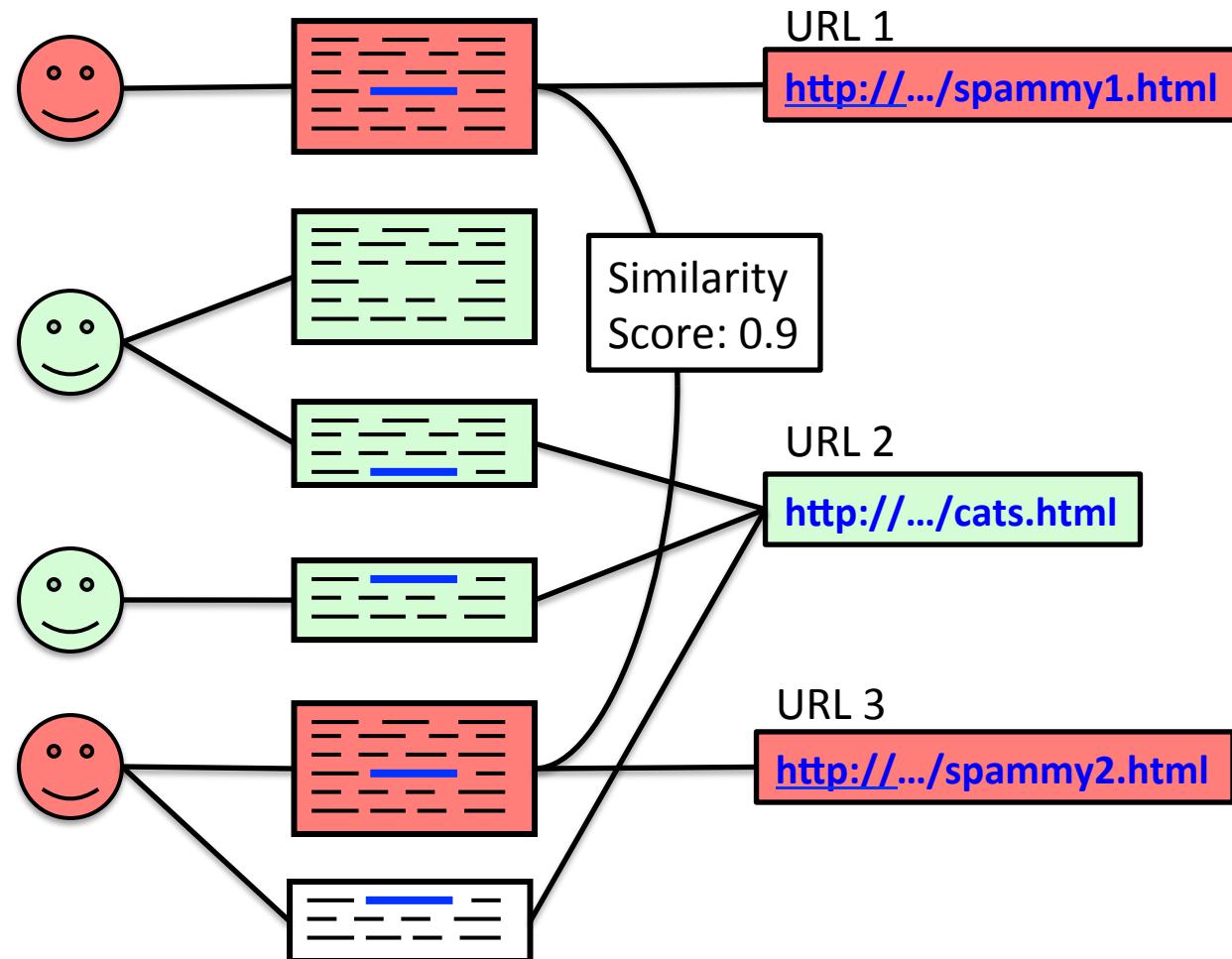
Detecting Spamy Comments



Detecting Spamy Comments



Adversary Response



Open Questions

- Non-zero-sum games
- Representing strategies:
Weights, decision nodes, distributions?
- Integrate with planning, reinforcement learning
- When is adversarial modeling unnecessary?
- Best methods for validating adversarial models (outside of industry)

Conclusion

- StarAI needs adversarial modeling
 - To fulfill long-term AI vision
 - To solve current applications
 - To improve robustness/safety
- Two ways to learn robust relational classifiers:
 - Embed the adversary inside the optimization problem
 - Construct an equivalent regularizer
(Special case: set a maximum weight!)
 - Empirically, these models are robust to malicious adversaries and non-malicious concept drift.
- Many open questions and challenges!