Can ML Solve Fake Reviews?

Roland Maio Presenting Joint Work with Ryan Amos, Joe Calandrino, and Augustin Chaintreau

ESP Research Symposium. November 20, 2020.

Online Reviews

- Important source of information.
- Used by web platforms to make recommendations.
- Relied on by consumers.
- Drive business.

Fake Reviews

- Distort information.
- Illegitimately manipulate outcomes.
- Harm consumers, web platforms, and legitimate businesses.
- Okay, but… this is old news.

Fake Reviews

- Distort information.
- Illegitimately manipulate outcomes.
- Harm consumers, web platforms, and legitimate businesses.
- Okay, but… this is old news.
- Yes, and… it's still a significant problem.

Filtering

- First proposed in 2006.
- Predominant approach to addressing fake reviews ever since.
- Detect fake reviews and remove.
- ML is the engine.

Does Filtering Work?

Maybe:

- Seems to be successful for some web platforms, in particular Yelp.
- Approach may be sound, but practically difficult to implement; collecting good data is extremely hard.

Does Filtering Work?

Maybe not:

- Fake reviews are still a major problem.
- Seems ineffective for many web platforms, in particular Amazon.
- Some attacks clearly require substantial investment in money, time, and technical expertise: Clearly, fake reviews can be extremely effective and economic.

Does Filtering Work?

And if it works now, will it work forever?

- Arms race aspect to fake reviews.
- Fake-review generation technologies advancing impressively.

Research Question: Can we say something about this?

Does Filtering Work?

And if it works now, will it work forever?

- Arms race aspect to fake reviews.
- Fake-review generation technologies advancing impressively.

Research Question: Can we say something about this?

How can we say something about this?

Research Approach

- Formulate a sequential game to model the arms race between a web platform and fake reviewer.
- Assume the fake reviewer can create any reviews.
- Assume the web platform can always deploy an optimal filter, that is the Bayes-optimal filter.
- See what happens!

The Fake Reviews Game

- 2 players: A fake reviewer F and a web platform P.
- Review feature vector space X
- Non-fake reviews are modeled by a probability distribution p over X.
- The players interact over T sequential rounds.

The Fake Reviews Game

- On round t, F chooses a probability distribution q_t that models the fake reviews it creates, and a quantity parameter a_t of fake reviews to create.
- The fake reviews mix in with the non-fake reviews creating a mixture distribution $r_t = (1-a_t)p + a_t q_t$.
- On round t, P chooses a classifier C_t that predicts for each review feature vector x whether it is fake or not.

The Fake Reviews Game

- The players receive a payoff for each round. Their objectives are to optimize their average round payoffs.
- On each round t, P receives a payoff that is the accuracy of its classifier C_t.
- On each round t, F receives a payoff that is… well, what should its payoff be?

Fake Reviewer Payoff: A Misstart

- One thought is that *F* is mounting a classic *evasion attack*; *F* is trying to successfully post as many fake reviews on the platform as possible.
- Modeling email spam as an evasion attack has been successful in the past.
- In this case, it's payoff should be tied to how many fake reviews are misclassified by the web platform's filter.

More Than An Evasion Attack

- Except, if *F* is launching an evasion attack, then it is always "optimal" to copy *p*.
- But if *F* just copies *p*, notice that this does not change the outcomes of the web platform's recommendations or consumers' purchasing decisions.
- So fake reviews are *not* an evasion attack.

Downstream Attacks

- We therefore define the class of *downstream attacks*, in which the fake reviewer's goal is to manipulate the outcome of algorithms that take as input, at least in part, some of the reviews *which are decided* by the web platform.

The Fake Reviewer's Payoff

- The web platform constructs a review input distribution from all the reviews it receives.
- The fake reviewer would prefer the web platform's algorithms to produce some outputs over others.
- Certain input distributions are more conducive to producing those outcomes than others.
- The fake reviewer's wants to induce a target input q^*.
- The fake reviewer's payoff quantifies how well it induces q^*.

Downstream Input Construction

- Downstream attacks surface the importance of how the web platform chooses which reviews to pass as input to further algorithms, the downstream input construction.
- Online construction: Maintain a collection of input reviews. On round *t*, add all the reviews predicted to be non-fake to the collection of input reviews.
- Batch construction: Store all the reviews ever received, on each round, reclassify every single review.

Online Construction is Bad

Result: If the web platform uses online construction, even if it is able to deploy the Bayes-optimal classifier on each round, then the fake reviewer can induce any target input distribution q^* as long as there are enough rounds.

Key idea: The fake reviewer can use its powers to control which reviews are classified as fake or non-fake on each round.

Batch Construction is Mixed

Result: If the web platform uses batch construction in combination with the Bayes-optimal classifier, then (1) the fake reviewer controls which reviews are classified as fake or non-fake, but (2) the input distribution cannot differ too much from the non-fake review distribution.

Conclusion

- Fake reviews are *not* classic evasion attacks, in particular, a fake reviewer can benefit from both fake reviews that evade the filter *and* fake reviews that do not.
- ML alone cannot solve fake reviews, ML can be a critical component of a comprehensive strategy that incorporates non-ML components.
- We explored one algorithmic non-ML component, we are excited for the potential of incentives-based components!

Questions?