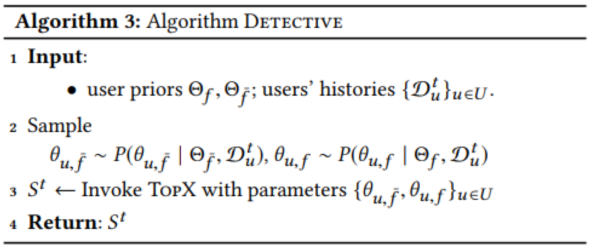
Fake News Detection in Social Networks via Crowd Signals

Shashank Srikanth

1 Introduction

1. They leverage crowd signals for detecting fake news. They propose a method to learn the user's flagging accuracy in order to use the user's flag effectively. Their goal is to minimize the spread of misinformation (fake news) and they propose a algorithm called DETECTIVE for the same.

2. The proposed DETECTIVE algorithm is Bayesian and is able to trade off between exploitation (news that maximize objective) and exploration (news that help in learning user activity).



```
Algorithm 3: Algorithm DETECTIVE
Input:
    • user priors $\Theta_f, \Theta_{\bar{f}}$; users' histories $\{\mathcal{D}_u^t\}_{u \in U}$.
Sample
    $\theta_{u,\bar{f}} \sim P(\theta_{u,\bar{f}} \mid \Theta_{\bar{f}}, \mathcal{D}_u^t), \theta_{u,f} \sim P(\theta_{u,f} \mid \Theta_f, \mathcal{D}_u^t)$
$S^t \leftarrow$ Invoke TopX with parameters $\{\theta_{u,\bar{f}}, \theta_{u,f}\}_{u \in U}$
Return: $S^t$
```

Figure 1: Proposed algorithm: DETECTIVE

2 Methodology

1. The method proposed is able to learn the flagging accuracy of users using their historical data and unlike other work by Kim, this method uses discrete epochs with fixed budget in each epoch.

2. The main objective of the algorithm is to minimize the spread of misinformation and they thus, define the utility of blocking news as the number of users saved from being exposed to fake news.

3. During each epoch, a set of news X^t and set of active users A^t are initialized. These users are further categorized as exposed and non-exposed users respectively. During the epoch, the algorithm finds the set of users to whom the news propagated and the set of users who flagged a news as fake (Not all users may flag a news as fake). At the end of each epoch, it selects a subset of news from the list of active news that is sent to experts for validation and then they are subsequently blocked.

4. Each user is represented using two values α_u and β_u. The users can be categorized as "News hater", "News lover", "Expert" & "Spammer" based on these values. γ_u represents the probability with which a user reviews a comment. Given a user's data history and a prior distribution over all the user parameters, a data matrix D_u^t is computed using the expert's labels.
$$D_u^t = \begin{pmatrix} d_{u,\bar{f}|\bar{f}}^t & d_{u,\bar{f}|f}^t \\ d_{u,f|\bar{f}}^t & d_{u,f|f}^t \end{pmatrix}$$

5. e.g., $d_{u,\bar{f}|\bar{f}}^t$ represents the count of news a user has labelled as not fake and the acquired label (expert) was not fake. Using this data matrix, the posterior distribution of the users parameters can be computed using Bayes rule.

6. Given, a user's parameters, the labels for a given news can be computed using Bayes rule and assuming that user's label are generated independently. At the end of every epoch, the algorithm greedily selects the K news that maximize the total expected utility.

7. The DETECTIVE algorithm, shown in figure 1, actively trades off between exploration and exploitation using posterior sampling mechanism. Each time the algorithm is called, it samples the user parameters as mentioned above and uses their posterior distributions to get the top K news that are sent to the experts.

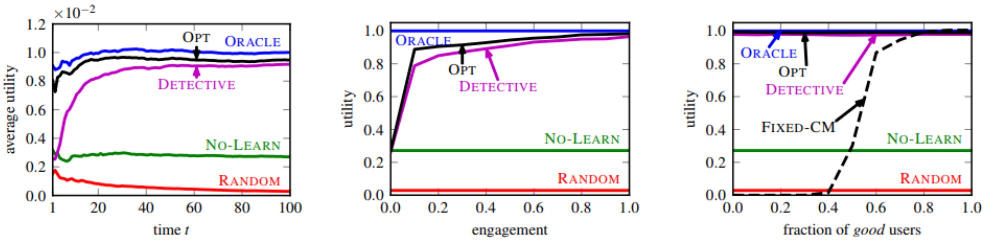


Figure 2: Experimental results of proposed algorithm

3 Analysis

1. The dataset used for the analysis was the social circles Facebook graph and the news spread was modelled using an independent cascade. The prior for seeding a fake news was set to about 20%. They consider three different types of users: Good users ($\alpha_u = \beta_u = 0.9$), spammers ($\alpha_u = \beta_u = 0.1$) and indifferent users ($\alpha_u = \beta_u = 0.5$).

2. They compare their algorithms with few other variants of the algorithm:

   (a) OPT (Knows the user's true parameters)

   (b) ORACLE (Knows the true labels of news).

   (c) FIXED-CM (Does not distinguish between users)

   (d) NO-LEARN (Selects news just based on highest utility)

   (e) RANDOM

3. As can be seen in figure 2, the DETECTIVE algorithm is able to achieve an average utility that is comparable to that of ORACLE. It is also able to perform well even in settings where the user participation is very low. Finally, the third figure shows that the algorithm is robust to spammers (adversarial users).