School of Finance

University of St.Gallen

FAKE NEWS IN SOCIAL NETWORKS

CHRISTOPH AYMANNS
JAKOB FOERSTER
CO-PIERRE GEORG

WORKING PAPERS ON FINANCE NO. 2018/4

SWISS INSTITUTE OF BANKING AND FINANCE (S/BF – HSG)

AUGUST 2017

Fake News in Social Networks

Christoph Aymanns*
University of St. Gallen
SRC, LSE

Jakob Foerster†
University of Oxford

Co-Pierre Georg‡
University of Cape Town
Deutsche Bundesbank

This draft: August 21, 2017

Abstract

We model the spread of news as a social learning game on a network. Agents can either endorse or oppose a claim made in a piece of news, which itself may be either true or false. Agents base their decision on a private signal and their neighbors' past actions. Given these inputs, agents follow strategies derived via multi-agent deep reinforcement learning and receive utility from acting in accordance with the veracity of claims. Our framework yields strategies with agent utility close to a theoretical, Bayes optimal benchmark, while remaining flexible to model re-specification. Optimized strategies allow agents to correctly identify most false claims, when all agents receive unbiased private signals. However, an adversary's attempt to spread fake news by targeting a subset of agents with a biased private signal can be successful. Even more so when the adversary has information about agents' network position or private signal. When agents are aware of the presence of an adversary they re-optimize their strategies in the training stage and the adversary's attack is less effective. Hence, exposing agents to the possibility of fake news can be an effective way to curtail the spread of fake news in social networks. Our results also highlight that information about the users' private beliefs and their social network structure can be extremely valuable to adversaries and should be well protected.

Keywords: social learning, networks, multi-agent deep reinforcement learning

*Corresponding author. E-Mail: christoph.aymanns@gmail.com
†E-Mail: jakob.foerster@cs.ox.ac.uk
‡E-Mail: co-pierre.georg@uct.ac.za

1 Introduction

The spreading of fake news on social networks has raised significant public concern since the United States Office of the Director of National Intelligence concluded that *"Moscow's influence campaign followed a Russian messaging strategy that blends covert intelligence operations–such as cyber activity–with overt efforts by Russian Government agencies, state-funded media, third-party intermediaries, and paid social media users or "trolls""* (Office of the Director of National Intelligence (2017)). James Clapper, former Director of National Intelligence, testified in this context before the Senate Armed Services Committee on foreign cybersecurity threats saying that Russia spread disinformation in the run-up to the 2016 US election using *"fake news"*.

A piece of fake news that appears on a social network typically makes some claim about the world that is factually wrong but not obviously so, such that there is some uncertainty about its veracity. Once it has been posted, users may like, share or comment on this piece of news facilitating its spreading across the social network. The aim of those peddling fake news can be monetary gains, or to influence expectations about the state of the world, as for example in the Pentagon's Military Analysts Program (Barstow, 2008; Jowett and O'Donnell, 2015).

A piece of fake news that is obviously false may be hilarious and might be shared for its comedy value. Instead, a piece of fake news that manages to convince users of a factually wrong claim is clearly more worrying. Whether or not a user, say Alice, is convinced of a particular claim about the state of the world will depend not only on her private information about the state of the world and hence the veracity of the claim, but also on her friends' actions on the social network. If her friend Bob shares the piece of news with a positive comment, Alice interprets his action as supporting the veracity of the claim. Similarly, Bob could voice his opposition.[1] When deciding on the veracity of the claim, Alice will combine the actions of her friends and her private information to reach a judgment. Crucially, in this process Alice's friends are not just passive intermediaries for the piece of news, but instead implicitly pass information about their stance towards the piece to Alice.

In our baseline specification, we assume that users' private information about the state of the world and hence the veracity of a claim is correct on average. That is, on average, relying only on their private information, users should be able to correctly identify fake news as such more often than not. At the same time, users' information is not perfect and on occasion they may fall for a piece of fake news. In this setting, an outside adversary, who wants to sway public opinion in favor of a piece of fake news but simply posts the piece on the social network himself, will likely fail in his efforts. For a successful attack, the adversary has to convince users on the social network to endorse the piece of fake news. Therefore, in our attack specification we allow an adversary to bias the private information of a subset of agents. This could be achieved for example via targeted advertising.

We then study how vulnerable stylized social networks are to this attack specification and

[1] We deliberately ignore political opinions and their spread on social networks in this paper. Instead we are only concerned with the spread of claims that can be unambiguously classified as either true or false.

how knowledge of the presence of adversaries spreading fake news can reduce this vulnerability. We combine a variation of a standard social learning model from the economics literature (see Mossel et al. (2015)) with recent advances in multi-agent reinforcement learning, deep Q-learning (DQN) in particular, to model the agents' decision making process. Specifically, in every period, an agent can either endorse or oppose a claim about the world that is true with a certain probability. Agents receive positive utility from endorsing true and opposing false claims. The agents' information set comprises their neighbors' actions in past periods and a private, noisy signal about the veracity of the claim. We represent each agent as an independent recurrent deep neural network that is trained via deep Q-learning (see eg. Foerster et al. (2016)) to find a strategy that maximizes her utility given her information set.

Our computational approach is distinct from the usual analytical methods for solving social learning models (see for example Golub and Sadler (2016); Mossel et al. (2015)) which can be broadly distinguished along two strands: Bayesian and non-Bayesian learning. The first strand, Bayesian learning, follows the models of Banerjee (1992) and Bikhchandani et al. (1992) where a player sequentially takes an action after observing a private signal and the actions of all other agents she is connected with. The other strand, non-Bayesian learning, follows the linear updating model of DeGroot (1974) where a player takes a decision after aggregating over the decisions of all neighbors in a network (for a discussion of this class of models, also see Molavi et al. (2017)). In our baseline case, when agents receive unbiased signals and follow strategies optimized via reinforcement learning, agent utility comes close to the theoretical, Bayes optimal benchmark. While we cannot claim that the agents are Bayes optimal after having optimized their strategies via reinforcement learning, our results suggest that agents do discover reasonable, close to optimal strategies. The flexibility of our computational framework allows us to study a number of attack specifications which would be difficult to study analytically in the Bayesian learning framework.

As such, the reinforcement learning framework combines the flexibility of computational models with a practical assurance of reasonable, near optimal, agent behavior. We also point out that DQN has successfully been used in a variety of complex problem settings Mnih et al. (2015); Hausknecht and Stone (2015), finding high performing strategies on a broad range of tasks. Since DQN is a model-free method that can efficiently utilize parallel computation, it can be applied in settings which are beyond the realm of analytical solutions or intractable from a Bayesian point of view. A further advantage of our computational approach is that we are able to fully characterize agents' strategies, which in literature of social learning so far have remained largely abstract.

If and how fast a population learns the true state of the economy is an important question and has been extensively studied in the literature on learning in networks (among many others, see Golub and Jackson (2010)). Various results exist that provide conditions under which a population learns the true state of the world absent attempts at manipulation of the agents. More directly related to our paper, Acemoglu et al. (2010) characterize how an adversary can interfere with information aggregation. Theirs is a model of non-Bayesian learning where agents meet according to a Poisson process to exchange information about the underlying state of the

world. There are two types of agents, regular and forceful agents. When regular agents meet both update their beliefs about the state of the world to the average of the two beliefs. When, however, a regular agent meets a forceful agent, there is a $1-\epsilon$ chance that the regular agent adopts the forceful agent's belief. Forceful agents update their beliefs with a small but positive probability and hence interact non-trivially with their neighbors.

While Acemoglu et al. (2010) also study the spread of misinformation, our model allows more sophisticated learning and instead of regular and forceful agents, we have agents and an outside adversary. In particular, as mentioned above agents use strategies that yield close to Bayes optimal performance in the absence of attack. In addition however, agent are able to re-optimize their strategies in the presence of an attacker. Achieving this in an analytical, Bayesian setting would be hard if not intractable.

Our model has several practical implications. First, we show that agents have a different optimal strategy if they are trained knowing about the existence of an adversary. An adversary is far less effective in spreading fake news when agents are aware of the adversary's existence. This highlights the importance of educating users about the possibility that claims made even in seemingly legitimate news outlets could be false. Second, we show that an attack is more likely to succeed if the adversary has knowledge about an agents' position in the social network and her private signals. Hence, a simple way to curtail the spread of fake news is to make it more difficult for an adversary to get access to agents' private information.

2 A model of information diffusion and beliefs in social networks

2.1 Model set up

We start by describing our model of information diffusion in social networks before turning to reinforcement learning as a method to solve the model. Let $\mathcal{G}(N,E)$ be the exogenously given directed graph that defines a social network with a set of agents N and a set of relationships E. We model the agents' binary actions $a^i \in \{0,1\}$ on the social network as the outcome of a game over T time steps in which agents receive a positive payoff for every period they endorse a true claim or oppose a false claim about the world and zero otherwise. For simplicity we assume that the payoff from endorsing a true claim or opposing a false claim is the same.[2] The claim that Hillary Clinton is crooked is an example of the type of claim we are interested in here. Formally, there is a single claim about the world, which is either true or false indicated by the binary variable $\theta \in \{0,1\}$. If the claim is true, we have $\theta = 1$ and $\theta = 0$ otherwise. For simplicity, though without loss of generality, we assume that the claim is either true or false with equal probability. The veracity of the claim θ is revealed at time T. An agent i receives an unobserved stage utility

[2]This set up is similar to the models studied in the literature on social learning, see for example Golub and Sadler (2016); Mossel et al. (2015).

$u_t^i = \mathbb{1}\{a_t^i = \theta\}$ for her action a_t^i at time t. The agent aims to maximize her total discounted utility $U^i = \sum_{t=1}^{T} \gamma^t u_t^i$ with $\gamma \in (0,1]$.

An agent with (out-)neighborhood $B^i = \{j \mid j \to i \in E\}$ conditions her actions on a hidden state $h_t^i \in \mathbb{R}^m$ which is a recursive function of her observation $x_t^i \in \mathbb{R}^{|B^i|+2}$ at time t and the history of states h_{t-1}^i. The agent's observation consists of her private signal $s^i \sim \mathcal{N}(\theta, \sigma^2)$, which is drawn at time $t = 0$, as well as her own and her neighbor's actions in the last time step. That is $x_t^i = (s^i, (a_{t-1}^j)_{j \in \widehat{B}^i})$, where $\widehat{B}^i = B^i \cup i$ is the self-augmented neighborhood of agent i. The agent's hidden state then evolves according to a law of motion

$$h_t^i = H^i(h_{t-1}^i, x_t^i),$$

where H^i is implemented as a recurrent neural network using gated recurrent units (GRU) (Cho et al., 2014), a popular alternative to long short-term memory (Hochreiter and Schmidhuber, 1997) (LSTM). The hidden state allows the agent to keep a memory of past events. Roughly speaking, as the dimensionality m of the hidden state vector increases, the number of internal parameters of H^i increases and we expect H^i to be able encode longer histories. For the technical details of the neural network implementation see Appendix B.

The agent is endowed with a state-action value function $Q^i : \mathbb{R}^m \times \{0,1\} \mapsto \mathbb{R}$ that measures the expected discounted utility conditional on a hidden state h_t^i and an action a_t^i. The agent then follows an ϵ greedy policy to pick her actions such that

$$a_t^i = \begin{cases} \arg\max_a Q^i(h_t^i, a) & \text{with } p = 1 - \epsilon, \\ a \sim U(\{0,1\}) & \text{with } p = \epsilon, \end{cases}$$

where the chosen action a is either the optimal action according to the state-action value function or a random variable drawn from a discrete uniform distribution. If $\epsilon = 0$ the agent will always pick the action that maximizes the state-action value function, whereas for $\epsilon > 0$ the agent will sometimes deviate from this policy and choose an action uniformly at random, i.e. experiment. In Table 1 we summarize the relevant parameters for the model and the set up of the deep Q-learning. Note that currently we are running the model with a relatively limited number of agents due to computational constraints which can be overcome with sufficient resources and a distributed implementation of this setup.

Ex ante, we do not assume a particular parameterization of Q^i and H^i. Instead, they are set up as a deep neural network, see LeCun et al. (2015), that can be trained via reinforcement learning, see Sutton and Barto (1998); Mnih et al. (2015) and appendix B for a review of this method. The trained function Q^i approximates the optimal state-action value function of the game and the trained function H^i computes an informative statistic of the history of the game available to agent i. While these methods are approximate, in the setting we are concerned with in this paper, they produce similar results to optimal benchmarks, as we show in Section 3.

Given that we do not assume a particular parameterization of Q^i and H^i, our analysis consists of two stages. First, in the training stage, we use reinforcement learning to find the

parameters of Q^i and H^i that maximize each agent's total discounted utility U_i. This stage can be thought of as a process through which agents discover a near-optimal policy. In the second, evaluation stage, we take the policies encoded in Q^i and H^i as given and study how effectively agents use private and social information to take actions. This is the process of information diffusion.[3]

We are interested in the effect of an outside adversary, who may bias the private signals of an agent of his choice, on the actions taken by agents. In particular, we ask whether such an adversary is capable of pushing the actions of agents towards endorsing false claims or opposing true claims. We study three scenarios in which the adversary has access to different information when choosing the agent to influence. In the first case, the adversary has no particular information about the agents. In the second case, the adversary knows the agents' position in the network and in the last case, the adversary knows the strength of the agents' private signal.

2.2 Model discussion

Our model makes a number of assumptions that are, while benign, worth discussing. First, we assume that agents receive utility from endorsing real news only, i.e. the utility from endorsing fake news is zero. Agents share news for social reasons, e.g. to get attention, but also to ensure they in turn receive informative news from their neighbors in the future. This is in line with empirical evidence from Lee et al. (2011), who show that users share news because of their informativeness. Second, we assume that agents receive an informative private signal, which is a standard assumption in the literature. Without an informative private signal we should not expect that agents eventually learn the true state of the world. Further note that we take a conservative view with respect to the likelihood that agents act in accordance with the veracity of the claim. That is, given the set up of our utility function, agents are interested in choosing the correct action. In real life, agents might not necessarily be interest primarily in the truth but in claims that confirm their private beliefs.

And third, we assume that reinforcement learning is a good approximation of the behavior of an agent that faces the incentives as set out in our model. Reinforcement learning ensures that the agents' actions are compatible with their incentives and close to optimal in contexts with which the agents are familiar. However, we argue that reinforcement learning can also capture elements of bounded rationality. In particular, when agents are faced with unfamiliar situations, such as the presence of an adversary actively pushing fake news, learned policies can be inappropriate and make agents vulnerable to manipulation. This feature of bounded rationality is what makes reinforcement learning ideally applicable to our situation.

[3] Information diffusion is sometimes referred to as *social learning*. To avoid confusion with reinforcement learning we use the term information diffusion throughout, though.

Parameter	Description	Value
γ	Discount rate	0.95
T	Time steps	20
N	Number of agents	10
m	size hidden state	12
σ^2	Signal variance	1
ϵ	exploration rate	0.05
Learning rate	-	5×10^{-4}
Training episodes	-	50000
RNN unit (hidden state)	-	Gated Recurrent Unit (GRU)
# RNN layers	-	2

Table 1: Model and Deep Q-learning parameters.

3 Measuring information diffusion

For the analysis in the remainder of this paper, the parameters of the neural network are held fixed at their values after training has converged (after 50,000 training episodes). Before studying information diffusion, we must agree on a way to measure it. Intuitively, an agent i should pick the action that corresponds to the value of θ with the highest likelihood, given state h_t^i. This means that an agent's action implicitly reflect her beliefs about the likelihood of the claim being true or false. If the claim is more likely to be true than false, the agent should pick $a^i = 1$, and $a^i = 0$ otherwise. As time goes on and agents observe their neighbors actions they will implicitly learn about their neighbors' signals, their neighbors' neighbors' signals and so on. Therefore, we can easily measure the extent of information diffusion in the social network by computing the average accuracy defined as the sum of the fraction of agents' stage utility

$$A_t = \frac{1}{N}\sum_i u_t^i = \frac{1}{N}\sum_i \mathbb{1}\left\{a_t^i = \theta\right\}.$$

Any value obtained for this measure of accuracy can be contrasted with two benchmarks. First, suppose each agent has to take an action based on her private signal alone. In this case the Bayes optimal policy is $a_t^i = 1$ if $s^i > 0.5$, $a_t^i = 0$ if $s^i < 0.5$, and random (either 0 or 1) otherwise. The accuracy of this policy is $A^1 = \Pr(s^i > 0.5 \mid \theta = 1)$. Now suppose agents were able to share their private signals perfectly so that they can compute the average signal $\widehat{s} = \sum_j s_j/|N|$. The Bayes optimal policy is then simply $a_t^i = 1$ if $\widehat{s} > 0.5$, $a_t^i = 0$ if $\widehat{s} < 0.5$ and random otherwise. The accuracy of this policy is $A^N = \Pr(\widehat{s} > 0.5 \mid \theta = 1)$. For a good policy, we should expect an improvement over time such that $A_0 \approx A^1$ and $A_T \approx A^N$. That is, at first agents act optimally with respect to their private signal alone and then approach the full information benchmark as they learn from their neighbors over time.

Figure 1 shows the accuracy measure A_t for four different network topologies: a star network, a directed ring, a complete network and an instance of a Barabasi-Albert random (pref-

erential attachment) graph with edge attachment parameter 3.[4] Throughout this paper we use the latter as a stereotypical example of a social network graph. Figure 1 shows that agents first make optimal use of their private information and then learn from the actions of their neighbors approaching the full information benchmark. If the network diameter is small, as for the complete network, this process is very quick and within 10% of the Bayes optimal solution within one time step. If the network diameter is large, however, as for the directed ring, the solution comes within the 10% of the Bayes optimal policy only after about 12 time steps. In the long run however, the accuracy of the directed ring appears to plateau at a level very close to the Bayes optimal accuracy and higher relative to the other networks.

In Figure 2 we show the agents action over a single run for the Barabasi-Albert graph. Some agents initially choose the wrong action but learn from their neighbors actions and ultimately converge onto the correct action. The dashed green line shows the average action of an agents' neighbors and they converge to the correct action starting from about $t = 6$.[5] One interesting pattern is that some agents switch their decision more than once. Agent $i = 4$, for example, starts with the wrong action, then switches to the correct action, but probably has a weak private signal so that once she observes that her neighbors chose a different action, she switches again to the wrong action in $t = 3$. Then she realizes that her neighbors have changed their actions on average and she switches to the correct action when actions start converging.

4 Uninformed adversary

Suppose that one agent's private signal is manipulated by an adversary so that it becomes biased away from the true value of θ. An adversary could, for example, show an agent targeted adverts that favor the adversary's point of view. For now, agents are not trained in the presence of an adversary, i.e. they are oblivious to this possibility. We assume that an adversary targets a single agent with an intervention such that that agent's private signal becomes

$$\bar{s}^i = s^i + \beta(1 - 2\theta),$$

where β can be interpreted as the adversary's budget: the larger β, the more the agent's signal will be biased away from the true value of θ. For now let us assume that the adversary has no information about the agents and therefore picks an agent to manipulate at random.

In Fig. 3 we show the learning accuracy in this scenario for a relatively large adversary budget of $\beta = 3$. As expected, learning is severely compromised. In fact, in case of the directed ring, learning fails all together and manipulation is very successful. In Fig. 4 we show the agents action over a single run. Based on their private signal alone 7 out of 10 agents initially pick the

[4] The number edges a new node forms with the existing nodes.

[5] Note that the average neighbor action is quite similar across agents. This shows that the network is well connected and agents are able to sample from a large fraction of the population. As one scales up the size of the network, agents will sample from a smaller fraction of the population and one would expect more heterogeneity in the average action of the agents' neighbors.

correct action. Over time however, the attacker is able to sway the entire population towards the wrong action (i.e. endorsing a claim $a = 1$ that was actually false $\theta = 0$).

5 Informed adversary

In Section 4 we show that manipulating one node in the social network can have large effects on the resulting population accuracy. Next, we investigate how much the efficacy of the attack can be increased if an adversary has access to the private beliefs of the agents and the network structure. This is relevant first, since the technology for targeted advertisement in social networks exists and second, since reports claim that it has been applied to influence both the Brexit referendum and the US election.[6]

To study this, we compute the manipulation efficacy conditional on (i) attacking a particular agent in the network, (ii) attacking an agent with a particular signal strength and (iii) attacking an agent who's neighbors have a particular signal strength. When analyzing this question we can either assume a "naive" population or one that is aware of an adversary that manipulates agents, as outlined in Section 4, by randomly targeting agents in the population with a biased signal. We consider a population aware of the presence of an adversary if it has been trained via reinforcement learning in its presence.

5.1 Attacking based on network position

Let us first consider the case where the adversary has information about the network structure and would like to determine which agent to manipulate given his budget. We define manipulation efficacy as the average difference between the baseline accuracy at time t and the accuracy under attack. Denote by $\bar{A}_t(i)$ the accuracy at time t conditional on agent i being taken over by the adversary. The manipulation efficacy is defined as:

$$\Delta A(i) = \frac{1}{T}\sum_t (A_t - \bar{A}_t(i)).$$

In Fig. 5 and Fig 6 we present our results for naive and manipulation aware populations. In the left panel of both figures agents were trained without an adversary present, while in the right panel, agents were trained with an adverary present. In both cases, we show the manipulation efficacy conditional on the attack on a particular node in the network. The difference between Figures 5 (Barabasi-Albert graph with edge attachment parameter of 3) and Figure 6 (star network) is the network structure.

If the system is trained without the adversary, the attack is very successful and leads to changes in accuracy between 7% and 20% in the Barabasi-Albert graph and 10% to 40% in the

[6] See, for example, The Guardian 7 May 2017, *"The great British Brexit robbery: how our democracy was hijacked"* https://www.theguardian.com/technology/2017/may/07/the-great-british-brexit-robbery-hijacked-democracy accessed on 9 August 2017.

star network. Knowing the network structure can drastically improve the manipulation efficacy. This is particularly obvious in the star network, where targeting the central node is most effective.

These results also indicate that the network structure itself affects the efficacy of manipulation with a very centralized network being more prone to manipulation than a decentralized one. The intuition is simple: if the central node in a star is attacked, all the peripheral nodes who only have one neighbor are easily swayed towards the wrong action. And even if a peripheral node is being attacked, it only needs to convince the central node to make the system as a whole more likely to switch to the non state-matching action.

As before, the effect is substantially smaller if the system is trained in the presence of an adversary. In this case, agents learn to be more careful about the actions of their neighbors since one of them might be attacked and the resulting change in accuracy under attack is much lower (between 0% and 12%). In addition, agents are likely to give less weight to their private signals which might be biased.

5.2 Attacking based on private signals

Next we consider the manipulation efficacy when the adversary has information about the agents' private signals. For this we define an agent's signal strength as the absolute log likelihood ratio of the signal, i.e.:

$$z(s) = |\log f(s \mid \theta = 0, \sigma^2) - \log f(s \mid \theta = 1, \sigma^2)|,$$

where $f(x|\mu, \sigma^2)$ is the normal pdf with mean μ and variance σ^2. Next, we compute the accuracy conditional on the attacker having a signal strength in some interval $z \in [\underline{z}, \overline{z}]$ and a budget β: $A_t(\underline{z}, \overline{z}, \beta)$. Note that the signal strength is computed using the "pre-bias" signal s_i rather than the biased signal $\bar{s}_i$ that the attacked agent observes. The manipulation efficacy is then defined as

$$\Delta A(\underline{z}, \overline{z}, \beta) = \frac{1}{T} \sum_t \left(A_t(\underline{z}, \overline{z}, 0) - A_t(\underline{z}, \overline{z}, \beta) \right).$$

We define the manipulation efficacy as a function of the average signal strength of the attacked agent's neighbors analogously. In Fig. 7 and Fig. 8 we present our results for the naive population only. The results are means and standard deviations over ten independent runs with fixed bin boundaries for $z(s)$ of $\{0.0, 0.5, 1.0, 2.0, 4.0\}$. Figure 7 shows results for varying attacker budgets conditional on the signal strength of the agent that is attacked. Overall, across different values of the attacker budget, the attack effectiveness is decreasing in the signal strength of the attacked agent. For large budgets ($\beta = 3$), the attack effectiveness plateaus for small signal strengths. We attribute the small non-monotonicity in this region to measurement noise. A plateau is to be expected since at some point the attacked agent's biased private signal will always lead him to take the wrong action. No further decline in accuracy is possible beyond this point. Outside the plateau region attacking weak signal agents increases attack effectiveness by

roughly a factor of two relative to random attack.

Results are similar, when the attacker conditions his attack on the average signal strength of the attacked agent's neighbors, see Figure 8. Note that here effectiveness increases at a faster rate as the signal strength falls. This is due to the fact that, since our network is small, low average neighbor signals correspond to states where a large part of the network received weak signals. In such states an attack can be very effective and lead to a large drop in accuracy. This also explains why attack effectiveness goes beyond the plateau in Figure 7; for low signal states there is simply more scope for manipulation. Unlike in Figure 7, the curve does not plateau since for $\beta = 3$ the attacked agent does not always choose the wrong action for all possible signals.

6 Conclusion

The spread of misinformation–fake news–in social networks has the potential to affect major political events like the Brexit referendum or the US presidential election. In this paper we model the spread of fake news through social learning in a network. The main technical innovation of our paper is to optimize the behavior of agents in a social network via multi-agent deep reinforcement learning. This allows for a rich set of agent strategies which, in a limit, perform as well as the Bayes optimal policy if agents share their private signals with all other agents. In another benchmark, our model performs as well as the Bayes optimal policy in which agents cannot share their private signals at all. While currently still in its infancy, we believe that the reinforcement learning approach to modeling the behavior of social and economic agent is promising. Future research should focus on extending and improving this approach to model social systems at scale.

From a societal perspective, the main contribution of our paper is that we provide a starting point for a computational framework to study the vulnerability of social networks to fake news. We use this framework to obtain an intuition for how an attacker might exploit users' private information on social networks to target influencing campaigns effectively. Given the extent to which information about user beliefs and network connectivity can affect the spread of misinformation on the social network in our models, regulators and service providers should consider controlling access to such information more rigorously. Further, we illustrate that once users are aware of the presence of fake news in social networks, they can adapt and become less susceptible to its spread. Thus, awareness of the existence of fake news is crucial in combating this issue.

References

ACEMOGLU, D., A. OZDAGLAR, AND A. PARANDEH-GHEIBI (2010): "Spread of (mis) information in social networks," *Games and Economic Behaviour*, 70, 194–227.

BANERJEE, A. (1992): "A simple model of herd behavior," 107, 797–817.

BARSTOW, D. (2008): "Behind TV Analysts, Pentagon's Hidden Hand," *The New York Times*, 20 April 2008, 1.

BIKHCHANDANI, S., D. HIRSHLEIFER, AND I. WELCH (1992): "A theory of fads, fashion, custom, and cultural change as information cascades," *Journal of Political Economy*, 100, 992–1026.

CHO, K., B. VAN MERRIËNBOER, D. BAHDANAU, AND Y. BENGIO (2014): "On the properties of neural machine translation: Encoder-decoder approaches," *arXiv preprint arXiv:1409.1259*.

DEGROOT, M. H. (1974): "Reaching a consensus," *Journal of American Statistical Association*, 69, 118–121.

FOERSTER, J., Y. M. ASSAEL, N. DE FREITAS, AND S. WHITESON (2016): "Learning to communicate with deep multi-agent reinforcement learning," in *Advances in Neural Information Processing Systems*, 2137–2145.

GOLUB, B. AND M. O. JACKSON (2010): "Naive learning in social networks and the wisdom of crowds," *American Economic Journal: Microeconomics*, 2, 112–149.

GOLUB, B. AND E. SADLER (2016): "Learning in Social Networks," *The Oxford Handbook of the Economics of Networks*.

HAUSKNECHT, M. AND P. STONE (2015): "Deep Recurrent Q-Learning for Partially Observable MDPs," in *Sequential Decision Making for Intelligent Agents – Papers from the AAAI 2015 Fall Symposium*.

HOCHREITER, S. AND J. SCHMIDHUBER (1997): "Long Short-Term Memory," *Neural Computation*, 9, 1735–1780.

JOWETT, G. S. AND V. O'DONNELL (2015): *Propaganda and Persuasion*, SAGE Publishing, sixth edition ed.

LECUN, Y., Y. BENGIO, AND G. HINTON (2015): "Deep learning," *Nature*, 521, 436–444.

LEE, C. S., L. MA, AND D. H.-L. GOH (2011): "Why do People Share News in Social Media?" in *International Conference on Active Media Technology*, 129–140.

MNIH, V., K. KAVUKCUOGLU, D. SILVER, A. A. RUSU, J. VENESS, M. G. BELLEMARE, A. GRAVES, M. RIEDMILLER, A. K. FIDJELAND, G. OSTROVSKI, S. PETERSEN, C. BEATTIE, A. SADIK, I. ANTONOGLOU, H. KING, D. KUMARAN, D. WIERSTRA, S. LEGG, AND D. HASSABIS (2015): "Human-level control through deep reinforcement learning," *Nature*, 518, 529–533.

MOLAVI, P., A. TAHBAZ-SALEHI, AND A. JADBABAIE (2017): "Foundations of non-Bayesian Social Learning," mimeo, Columbia University.

MOSSEL, E., A. SLY, AND O. TAMUZ (2015): "Strategic learning and the topology of social networks," *Econometrica*, 83, 1755–1794.

OFFICE OF THE DIRECTOR OF NATIONAL INTELLIGENCE (2017): "Background to 'Assessing Russian Activities and Intentions in Recent US Elections': The Analytic Process and Cyber Incident Attribution," Report.

SUTTON, R. S. AND A. G. BARTO (1998): *Reinforcement Learning: An Introduction*, MIT Press.

A Figures

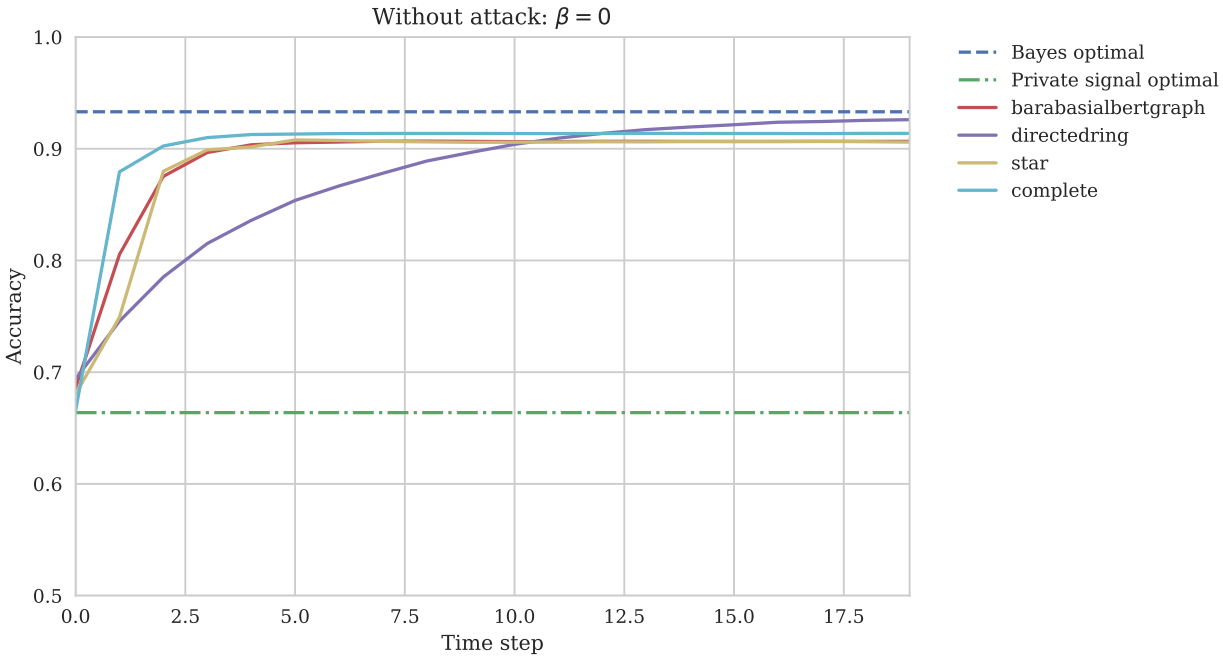

Figure 1: Information diffusion on different social networks measured at time step t as accuracy A_t. Results are shown for different topologies of the social network and two benchmark cases. The Bayes optimal benchmark is obtained in the limit where each agent observes the average private signal $\hat{s}$. The privately optimal benchmark is obtained when each agent observes only her private signal s. All four networks (complete network, star, directed ring, and Barabasi-Albert graph with $m = 3$) have $|N| = 10$ agents and $T = 20$ time steps and $\sigma^2 = 1$.

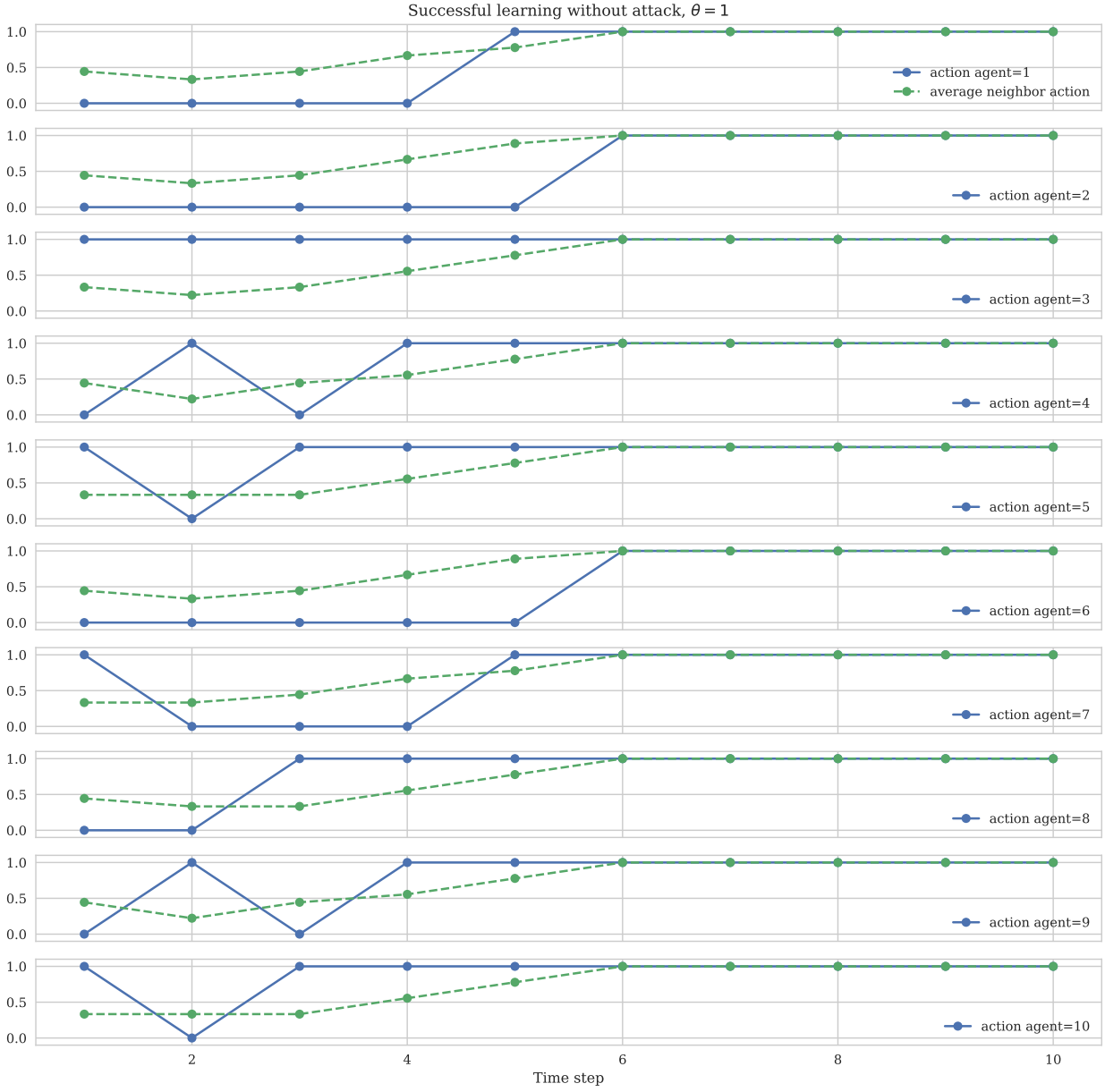

Figure 2: Information diffusion on a Barabasi-Albert graph with $m = 3$. Average neighbor actions (dashed green line) and agent i's actions (solid blue line) for all $|N| = 10$ agents. For $\theta = 1$ the claim is true, while $\theta = 0$ it is false, here the the claim is true. We use $\sigma^2 = 1$ and $T = 20$ time steps.)

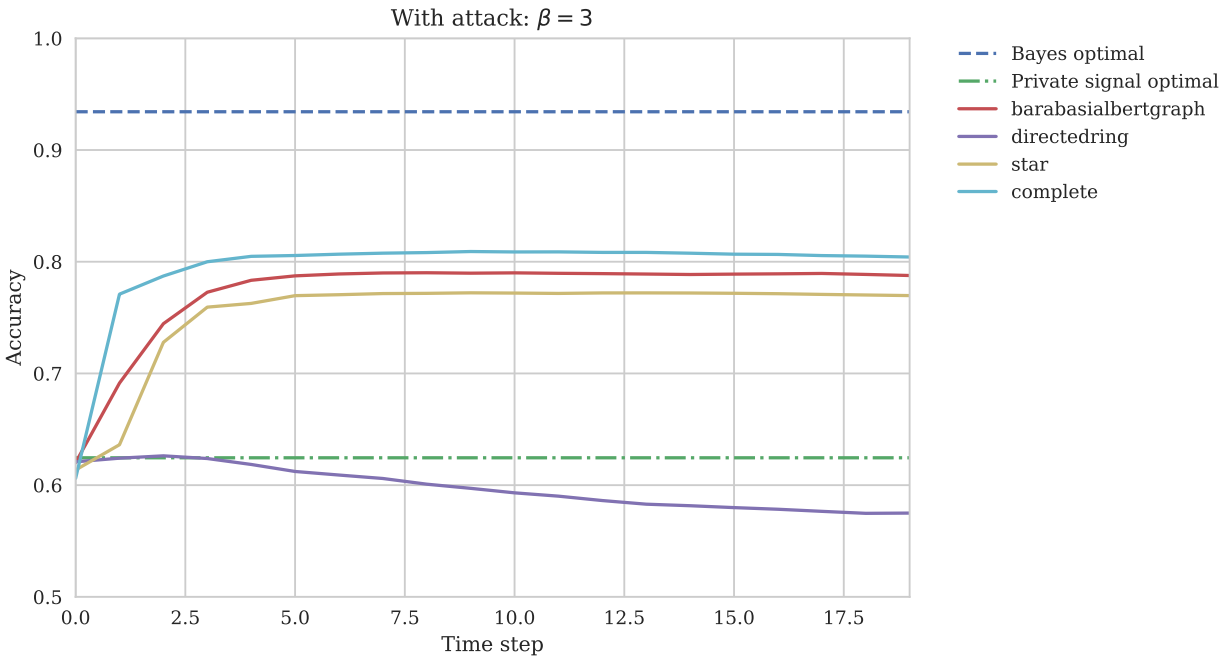

Figure 3: Information diffusion on different social networks measured at time step t as accuracy A_t. Results are shown for a relatively large adversary budget of $\beta = 3$, different topologies of the social network and two benchmark cases. The Bayes optimal benchmark is obtained in the limit where each agent observes the average private signal $\hat{s}$. The privately optimal benchmark is obtained when each agent observes only her private signal s. All four networks (complete network, star, directed ring, and Barabasi-Albert graph with $m = 3$) have $|N| = 10$ agents and $T = 20$ time steps and $\sigma^2 = 1$.

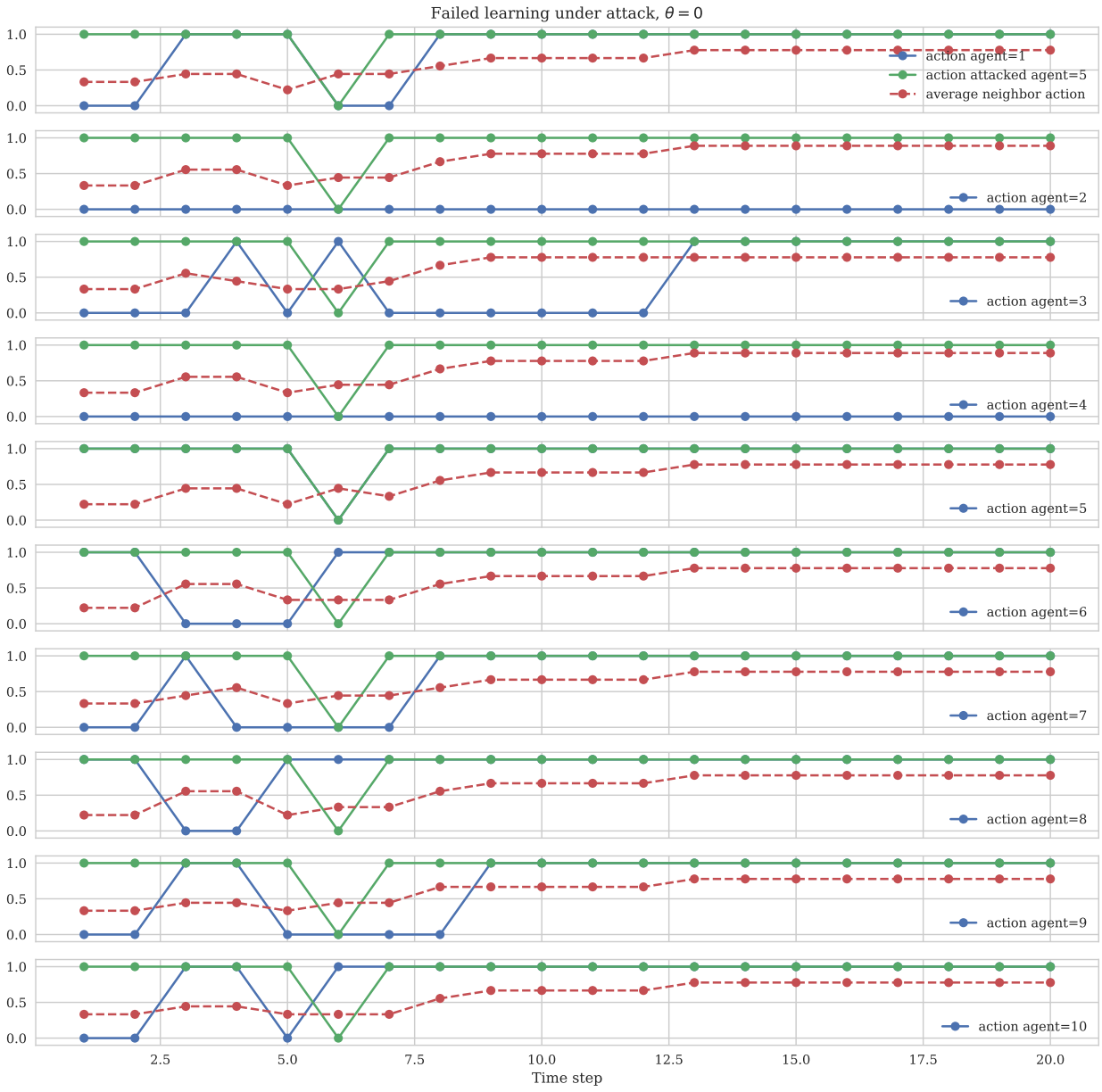

Figure 4: Failed information diffusion on a Barabasi-Albert graph with $m = 3$. Average neighbor actions (dashed red line) and agent i's actions (solid blue line) for all $|N| = 10$ agents. For $\theta = 1$ the claim is true, while $\theta = 0$ it is false, here the the claim is false. We use $\sigma^2 = 1$ and $T = 1, \ldots, 20$ time steps. Results are obtained for $RNN = 12$.)

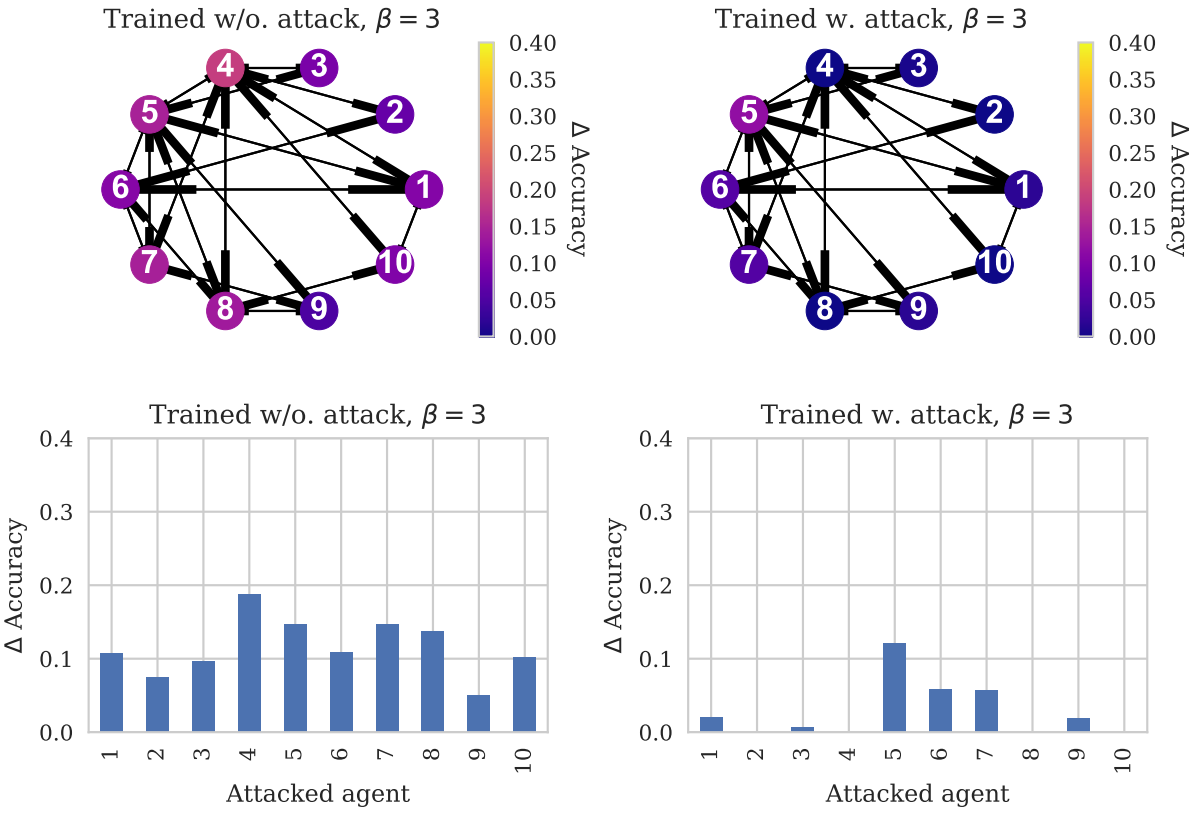

Figure 5: The effect of an attack on agent j on information diffusion. Left column: Agents are trained in the absence of an attacker but then tested in the presence of an attacker. Right column: Agents are trained and tested in the presence of an attacker. Top: Structure of the network. The color of each node corresponds to the change in accuracy an attack on this node would cause. Bottom: Change in accuracy due to attack on agent $j = 1, \ldots, 10$.

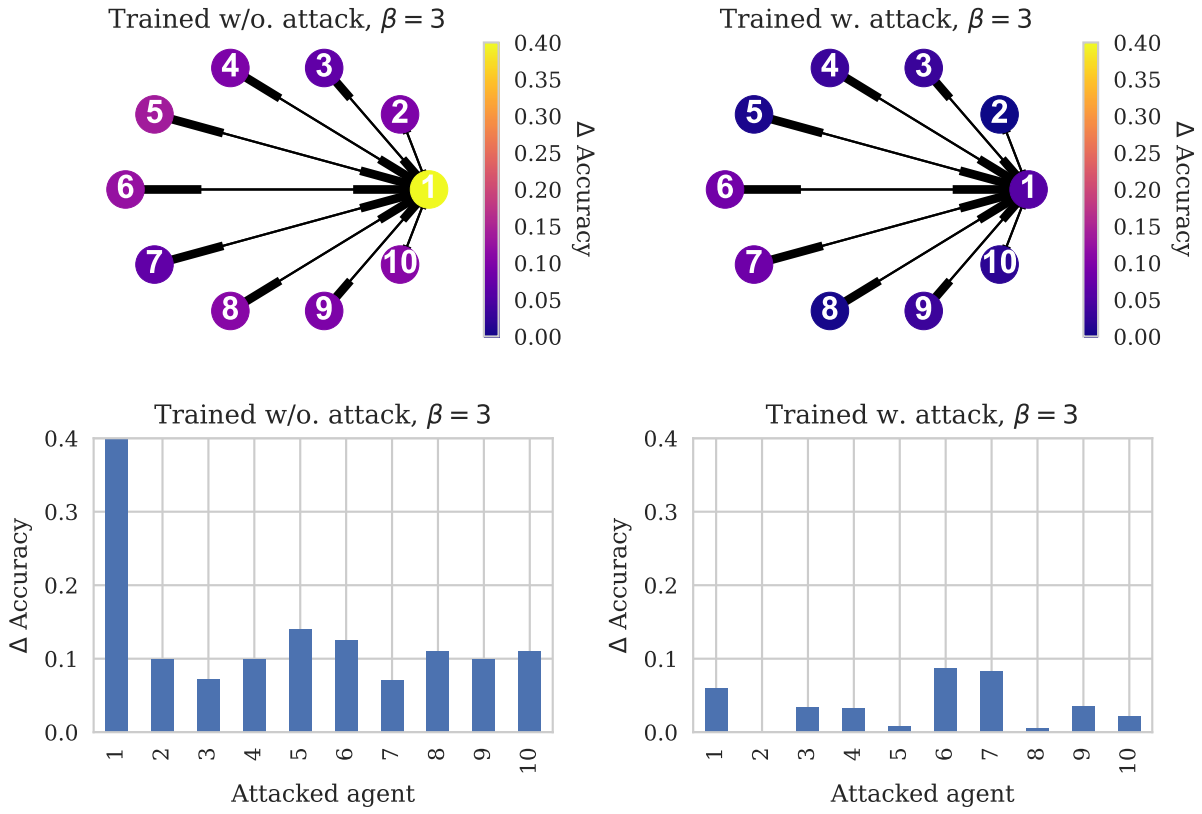

Figure 6: The effect of an attack on agent j on information diffusion. Left column: Agents are trained in the absence of an attacker but then tested in the presence of an attacker. Right column: Agents are trained and tested in the presence of an attacker. Top: Structure of the network. The color of each node corresponds to the change in accuracy an attack on this node would cause. Bottom: Change in accuracy due to attack on agent $j = 1, \ldots, 10$.

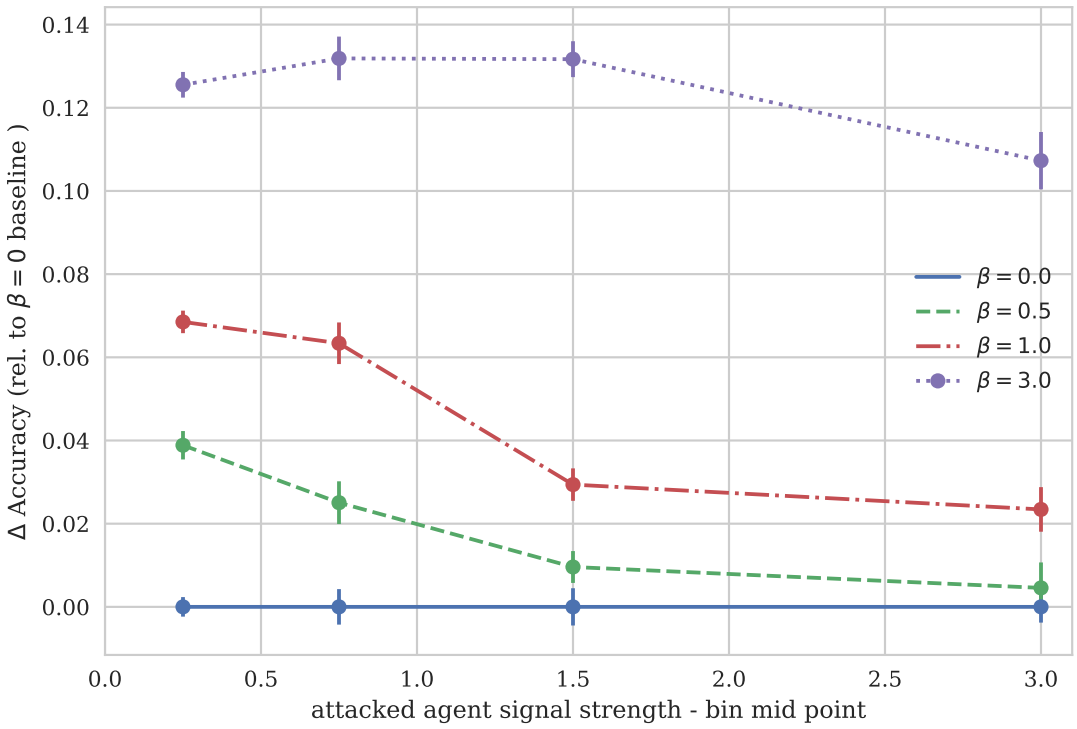

Figure 7: Manipulation efficacy as a function of the attacker's signal strength on a Barabasi-Albert graph with $m = 3$.

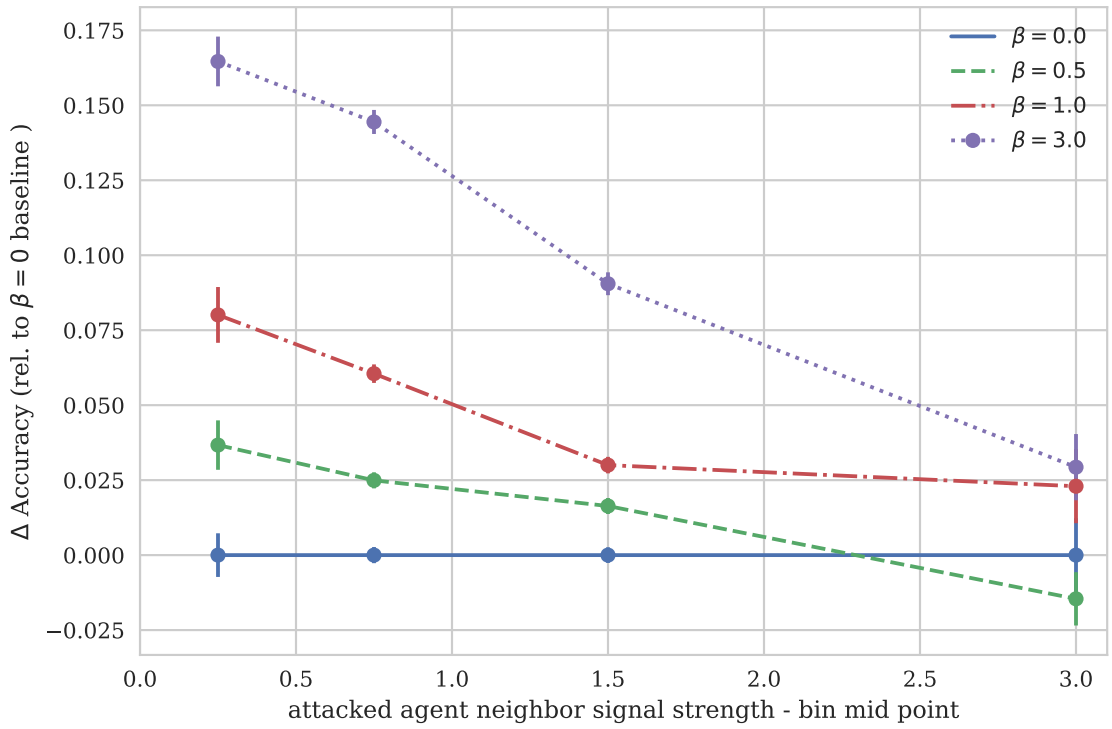

Figure 8: Manipulation efficacy as a function of the attacker's signal strength on a Barabasi-Albert graph with $m = 3$.

B Q Learning

B.1 Single-Agent Reinforcement Learning

In single agent RL the task of the agent is to maximize the average discounted utility $U_t = \sum_{t=0}^{T} \gamma^t u_t$. Here u_t is the agent's utility at time t, which gets discounted with $\gamma \in [0,1)$ (Sutton and Barto, 1998). For each state $x_t \in X$ the agent choses actions $a_t \in A$ with probability $\pi(a|x)$, where $\pi(a|x)$ is the agent's policy.

Q-learning relies on an action-value function Q, which estimates the average discounted utility for any given state action pair: $Q^\pi(x,a) = \mathbb{E}[U_t|x_t = x, a_t = a]$. For any sample the current estimate can be compared to a greedy one-step lookahead using the Bellman optimality operator, $\mathcal{T}Q(x,a) = \mathbb{E}_{x'}\left[u + \gamma \max_{a'} Q(x',a')\right]$. This iterative process results in the optimal Q-function

$$Q^*(x,a) = \max_\pi Q^\pi(x,a),$$

which trivially defines the optimal policy $\pi^*(x,a) = \delta(\arg\max_{a'} Q^*(x,a') - a)$, where $\delta(\cdot)$ is the Dirac-delta function.

DQN (Mnih et al., 2015) uses a neural network parametrized by ϕ to represent the Q-function. In order to insure sufficient exploration, agents chose actions from an ϵ-greedy policy during training. After collecting a batch of experiences from the interaction with the environment the parameters are updated to minimize the DQN-loss function:

$$\mathcal{L}(\phi) = \sum_{j=1}^{b} [(y_j^{DQN} - Q(x_j, a_j; \phi))^2], \tag{1}$$

Here $y_j^{DQN} = u_j + \gamma \max_{a'_j} Q(x'_j, a'_j; \phi^-)$, is the target function and ϕ^- is the target network, which contains a stale copy of the parameters. This target network helps to stabilize the training. So far we have assumed that the agent has access to the Markov state, x, of the system. In partially observable settings this state needs to be estimated based on the action-observation history of the agent, h. In recurrent deep RL (Hausknecht and Stone, 2015), this can be achieved using recurrent neural networks, such as LSTM (Hochreiter and Schmidhuber, 1997) or GRU which we use here.

B.2 Multi-Agent Reinforcement Learning and Independent Q-Learning

In Multi-Agent Reinforcement Learning, each agent $i \in N$ receives a private observation $O(x,i)$, where i is the agent index and O is the observation function. The agents also receive individual utility u_t^i and take actions a_t^i. Furthermore the state-transition conditions on the joined action $\mathbf{a} \in \mathbf{A} \equiv A^n$.

In Independent Q-Learning (IQL) each agent further estimates a Q-function $Q^i(h^i, a^i)$, treating the other agents and their policies as part of a non-stationary environment. IQL com-

monly uses parameter-sharing across agents combined with an agents specific index in the observation function in order to accelerate learning while still allowing for specialization of policies.