# StratLearner: Learning a Strategy for Misinformation Prevention in Social Networks (Author Response)

We thank all the reviewers for their time and constructive comments. This letter will first discuss common issues raised by the reviewers and then respond to individual comments.

**Experiments on real-world graphs (Reviewers 1-4).** We agree that real-world graphs are worth leveraging for experimentation, and we present one result (Table A) on a Facebook graph with $4,039$ nodes from SNAP[1], where StratLearner is trained with $100$ subgraphs from distribution $\phi_{0.1}^{1.0}$ and 270 training examples are used in each learning-based method. Other settings are the same as the experiments in the paper. Overall, similar to Table 1 in the paper, we have the observation that StratLearner outperforms other competitors by an evident margin. We will include this part in our paper to strengthen the experimental studies.

**Table A: Results on Facebook**

| StratL | NB | MLP | GCN |
|---|---|---|---|
| 0.725 (1E-2) | 0.662 (6E-3) | 0.651 (5E-3) | 0.625 (2E-3) |
| **DSPN** | **HD** | **Pro** | **Rand** |
| 0.446 (2E-3) | 0.656 (8E-3) | 0.170 (1E-2) | 0.011 (8E-3) |

**Motivation of the ideas (Reviewers 2 and 4).** The form of our scoring function is motivated by the fact (Theorem 1) that the prevention function (which is the perfect scoring function but unknown) can be factorized as an integration of the distance functions over subgraphs. Because the true distribution is unknown, we attempt to use an affine combination of random subgraphs to approximate the true distribution, where the weights are adjusted using data; the feasibility of doing so is partially justified by Theorem 2 stating that the function approximation can be theoretically bounded. We agree that it would better to highlight the motivations earlier in the manuscript, and we will make revisions accordingly.

**Reviewer 1. a)** It is true that knowing the past optimal solution is not realistic, especially given that the considered problem is NP-hard, and we wish to note that our framework does not require the samples to be optimal to produce good protectors. As long as the sample solutions are of high quality, they are sufficient to guide the model to discriminate between high- and low-quality solutions, which is evidenced by our experiments where the sample solutions are approximations. **b)** The loss function $L(P, S)$ is introduced to measure the difference between $P$ (ground-truth) and $S$ (prediction). The constraints cannot be (easily) reduced to polynomial even if $L(P, S)$ is constant, because the inference problem would still be NP-hard (Theorem 3). **c)** We are not able to answer – in theory – how large $K$ needs to be to ensure the final performance; experimentally, $K = 400$ is sufficient for achieving a decent performance ratio on the considered datasets. In practice, the most cost-effective $K$ might be determined through cross-validation.

**Reviewer 2.** We do not report the running time in the current paper because the entire process is reasonably fast (less than three hours), and we agree that it would better to briefly discuss it. It remains unknown that if StratLearner can scale to large datasets (in terms of time and memory usage), and handling graphs with millions of nodes is nontrivial because combinatorial algorithms are involved, which believe is a worthwhile research task.

**Reviewer 3**. **a)** This paper does not carry out the analysis of the final approximation guarantee on the PM problem, and we would like to thank the reviewer for pointing out the concerns regarding this part, which we believe is an important research direction. We are currently not able to establish the final approximation guarantee or the condition for ensuring that the correct weight or distribution can be learned. The main challenge lies in relating the true error under the structured prediction (with approximate inference) to the approximation ratio of the PM problem; there exist some negative results for similar settings[2], showing that this problem is nontrivial. Nevertheless, the experimental results are very promising, which is motivating us to continue theoretical explorations. **b)** Having observations from the cascades is indeed a reasonable setting, but our problem is more concerned with the case in which we only have data regarding the target problem (PM). We agree that it is interesting to examine the approaches that first learn the model and then perform optimization, but we do not include them in the current experiments because they leverage another type of data. **c)** MLP and GCN are trained using a standard classification loss. We will improve the description to make it clear.

**Reviewer 4**.**a)** The existing methods (e.g., Tong & Du [1]) require that the diffusion model is known to us, while our paper deals with the case where the diffusion model is unknown and the entire problem has to be solved using data. Therefore, the method in Tong & Du [1] is not applicable to our setting, and it is not used as a competitor. Because the method in Tong & Du [1] gives the theoretically optimal solution (under the assumption NP≠P), it is taken as a baseline to evaluate StratLearner and its competitors. **b)** The number of the subgraphs determines the ability of our model to approximate the underling diffusion function; therefore, the model trained with more subgraphs can produce a better generalization performance.

We also thank the reviewers for other valuable comments: **a)** citing the related previous work right after Problem 1; **b)** typo in line 99; **c)** clarifying that one cannot select existing seed for prevention; **d)** clarifying the NP-hardness in line 53; **e)** Line 70 is not correctly phrased. We will address them.

[1] Leskovec, Jure, and Andrej Krevl. "SNAP datasets: stanford large network dataset collection; 2014."

[2] Balkanski, Eric, Aviad Rubinstein, and Yaron Singer. "The limitations of optimization from samples." STOC, 2017.