Optimising News Verification in Social Networks

Leonardo Mazzone (s1555214)

***Abstract*— An iterative credibility propagation algorithm for fake news classification is formulated, achieving encouraging accuracy results. The circumstances under which the algorithm is effective are discussed, and a modification is presented in which we account for the popularity of users, in order to identify nodes to target to minimise the impact of fake news diffusion.**

I. Introduction

In order to tackle the emerging issue of the diffusion of misinformation via social media, several approaches based on the automatic identification of fake news and bots have been suggested by literature. However, increasingly social media platforms such as Facebook and YouTube rely on independent organizations to manually verify user-generated content [1], [2]. Because human operators are a scarce resource and slower than trained classifiers, it might be useful to implement an automated ranking system to direct the efforts of fact-checkers. This project turned to this under-studied setting and specifically attempted to address the following:

Problem statement Suppose you have a graph G representing the state of an online micro-blogging platform such as Twitter at one point in time. G contains nodes representing users (with a directed edge between each follower and followee), and nodes for stories that users shared (with directed edges from stories to the users sharing them). Each story has a true label $l \in \{real, fake\}$. For a small number of stories, the label is known. We have a verification budget b, i.e. a total number of stories whose label we can learn. On the other hand, there is no limit on the number of stories that, if fake, can be removed from the graph to produce a new graph G' . Every fake story that is not removed from G has a certain probability of being shared by other users, followers of the original publishers, and as a consequence G' will present new edges between the persisting fake stories and the new sharers. The objective is to assign the budget to b stories such that, after targeting those found to be fake, the number of new edges between fake stories and users in G' is minimized.

II. Related work

Gupta et al. [3] first introduced the idea of an algorithm inspired by PageRank to improve the performance of traditional fake news classifiers that do not take entity relations into account [4]. They model users, tweets, and events as nodes. Works such as [5], [6] build on that by using richer networks representing more granular entities, such as "conflicting viewpoints", and "sub-events". In [7], authors use an iterative, relations-aware algorithm to address the same problem, and model users, Facebook posts, and "likes". However, they focus on building a binary classifier rather than ranking the credibility of stories. All the works mentioned so far deal with fake news recognition, whereas [8] focuses on mitigation: it applies an analogue of the influence maximisation framework [9] to an Independent Cascade Model with the purpose of choosing the best nodes to target when launching a debunking campaign. In their setting they consider only one misinformation campaign at a time, and they have a limited capability to influence it, i.e. they look at it from the perspective of debunkers rather than that of the administration of a social media platform.

III. Methodology

A dataset was constructed using the Twitter API and the labelled tweets made available by FakeNewsNet [10]. FakeNewsNet publishes a repository of IDs of tweets sharing links to news and a classification into real / fake news. A base network G comprising the following type of vertices was constructed:

Story nodes, representing stories (or news items) present in the FakeNewsNet repository

User nodes, representing the authors of tweets, recorded in the FakeNewsNet repository, that spread the stories.

A base network G^D was built by also adding the following:

Domain nodes, representing domain names for the websites originating news items.

Undirected edges connect domain nodes to relevant story nodes and story nodes to relevant user nodes. I chose to model users because there is a high degree

of homophily with respect to the credibility of news in social networks [11]. To my knowledge, no previous work studied the homophily of domains, nor are they taken into account by other classification approaches, but later results shows they are useful in some cases, and modelling their credibility might provide value in itself. After building these base graphs, derivate graphs denoted as G_i and G_i^D have been used for experiments, equal to G and G^D if not for the removal of user nodes with fewer than i neighbours, the removal of the orphaned story nodes, and then of the orphaned domain nodes. User nodes with only one neighbour were never considered, to speed up computation. It turns out that this pruning improves not only the speed, but also the accuracy of credibility propagation. The credibility propagation algorithm procedes as follows:

1) Initialise a small set of story nodes with respect to their true label
2) Until convergence, iteratively update the scores of users, domains, and stories according to equations 1, 2, and 3

$$s(u) = \sum_{n \in N(u)} s(n) / \sum_{u'} s^*(u') \tag{1}$$

$$s(d) = \sum_{n \in N(d)} s(n) / \sum_{d'} s^*(d') \tag{2}$$

$$s(n) = \frac{\sum_{u \in U(n)} s(u) + \sum_{d \in D(n)} s(d)}{\sum_{n'} s^*(n')} \tag{3}$$

In the above equations, s is the score function, s^* is its unnormalised version, and U, N and D are the maps to neighbouring user, story, and domain nodes. A variant of this algorithm, *credibility propagation with audience redistribution* (CPAR) has also been tried out, where at each iteration we compute new story scores $s'(n) = \varepsilon a(n) + (1 - \varepsilon)s(n)$, where a maps stories to their audience size and ε is the redistribution constant.

The performance of the final algorithm was evaluated as the *spared audience ratio* (SAR), or the ratio between the sum of the user audiences for each story that was correctly identified as fake after b attempts, and the sum of user audiences we would spare if we had access to a *credibility oracle* and use it to target the b fake story nodes with largest audiences. As a simplifying assumption, audiences are calculated as the number of followers of each user, which may not reflect their actual importance within the full network, e.g. their capacity to generate cascades.

IV. Results

A. Credibility propagation

As an intermediate step the algorithm was run without factoring in information about user audiences, in order to assess whether it is possible to predict news as fake or real after having initialised a small number of them to their true label. A classifier was built such that it sorted news with respect to their credibility score and predicted them to be fake if the score was found to be above a threshold. The performance of the classifier was measured in terms of average precision, i.e. $\sum_k P(k)\Delta R(k)$ where each k represents a classification threshold, P is the precision function and ΔR is the function of differences of recall values computed with consecutive threshold values.

	G_2	G_2^D	G_3	G_3^D	G_4	G_4^D
Avg.	0.710	0.904	0.945	0.930	0.960	0.940
Std.	0.241	0.112	0.084	0.084	1.4×10^{-5}	0.081
Max.	0.944	0.947	0.960	0.952	0.960	0.960
Min.	0.341	0.334	0.352	0.351	0.960	0.370

TABLE I

AVERAGE PRECISION STATISTICS OVER 50 RANDOM INITIALISATION SETS

At most randomly-chosen 1% of story nodes were initialised (6 of them for all networks) on the basis of their true label: 1 if fake, 0 if true, 0.5 all others. Results are summarised in Table I. On G_2 the performance of the algorithm becomes sensitive to different initialisation sets, with average precision values ranging from 0.34 to 0.94. This result is unintuitive, as it has been verified that the two subgraphs consisting only of fake news and their users, and vice versa, real news and their users, are connected or almost connected (with only few, small, other components) in all cases. Therefore, one might expect that the correct signal should always be able to find a path between story nodes with the same label. However, I conjecture that there are some story nodes which behave like local bridges among different communities, and a different initialisation might cause them to receive more signal from story nodes of the opposite sign, thus polluting their credibility score and blocking many important paths for the propagation of credibility. Additionally, when adding many more users (those with few neighbours) one expects that they will be preferentially attached to the already popular stories (the best candidates to be the aforementioned

local bridges), increasing the chance that those stories will be connected to stories whose label has opposite sign.

The unreliability on G_2 can be partially mitigated using domain nodes to connect events. This allows to increase, across 50 random sets of initialised nodes, the mean while decreasing the standard deviation of average precision. However, even with domains, some particularly unlucky initialisations still give very low performance. Domains seem to have the opposite effect on other networks, presumably because their signal is redundant once we have reliable users, and it increases the chance of the mixed signal problem arising. Hence, for successive exploration, we turn to G_4 [1], on which the best and most consistent results were obtained across runs.

The performance of the classifier based on G_4 was found to be very promising. One must be wary of directly comparing these results with those of other works, because of the different size and composition of the datasets used. However, by choosing a simple split-in-the-middle classification threshold, an accuracy of 0.899 is obtained, slightly better than that reported by [3], equal to about 0.85 with their most sophisticated model. The most likely explanation is that stories, unlike event nodes in [3] are unambiguously described by their URL and it is therefore easy to identify them with absolute precision, making the paths between them and between their users very reliable. Additionally, unlike event nodes, stories are not simply describing an occurrence, but are associated to a particular narrative and stance regarding that occurrence, and have therefore stronger semantic implications. It should also be noted that the truth or falsehood of stories is also unambiguous, as they had been manually classified, and that this project uses a partially-supervised approach. Finally, as already mentioned, domains are an effective carrier of information that allows to resolve the difficulties encountered in some settings, and the removal of weakly-connected users is another effective trick. My model's performance is also seemingly better than that of [5] and [6], who use richer representations but do not model users. My classification accuracy is beaten only by [7], who build the most similar network to mine, but run a significantly different algorithm. This seems to confirm that the key drivers of information are users linked to well-defined stories (or posts in their case), and thus further complicating the network structure might be unnecessary.

[1]I arbitrarily chose to initialise the nodes with highest degree. While this was not an effective heuristic for G_2, it does not matter on G_4, as the performance on it is consistent.

B. Re-ranking with audiences

The naïve approach to factoring in audiences would be to choose a split within the list of news ranked by their credibility score, and sort them again on the audience. This approach has proven problematic due to the fact that the false negatives are the more likely to have large audiences. With a split-in-the-middle classification, the misclassified fake news own almost 84% of the total audience mass. Intuitively, popular fake news is more likely to be shared by users who would not otherwise tend to share fake news. This makes the selection of stories that are both popular, and fake, problematic. To address this problem, CPAR as described in Section III was applied. While promoting fake news with large audiences (see Appendix), it almost never beats choosing nodes purely based on audiences (full redistribution). Presumably, this is due to near-equality of the counts of fake and real news in this dataset. These results might change with a more representative proportion. Only with some specific budgets (e.g. $b = 24$, obtaining a 0.659 SAR), did a 0.1 redistribution improve the spared audience ratio with respect to a full redistribution (while inevitably penalising average precision with respect to a 0 redistribution).

V. Conclusions and future work

Experiments have shown that even a simple network can be very effective in the classification of fake news if users are modelled and aptly pruned, and with a small set of pre-labelled stories. The discordance between credibility and audience-based ranking has been remarked and its solution remains an open problem. Future research should look at whether additional features of popular fake news can be modelled to improve budget distribution. A more rigorous analysis of the graph characteristics that determine the success or failure of credibility propagation is another important direction for further investigation. It remains to be seen whether the credibility scores of users and domains, learned as a by-product, could be used for other applications, such as automatic bot detection. Finally, these results should be reproduced on a larger dataset to increase the confidence in their validity.

REFERENCES

[1] *Fact-checking on Facebook: What publishers should know. https://www.facebook.com/help/publisher/182222309230722. [Online, accessed: November 11th 2019].*

[2] *See fact checks in YouTube search results. https://support.google.com/youtube/answer/9229632?hl=en. [Online, accessed: November 11th 2019].*

[3] *Manish Gupta, Peixiang Zhao, and Jiawei Han. Evaluating event credibility on twitter. In* Proceedings of the 2012 SIAM International Conference on Data Mining*, pages 153–164. SIAM, 2012.*

[4] *Carlos Castillo, Marcelo Mendoza, and Barbara Poblete. Information credibility on twitter. In* Proceedings of the 20th international conference on World wide web*, pages 675–684. ACM, 2011.*

[5] *Zhiwei Jin, Juan Cao, Yu-Gang Jiang, and Yongdong Zhang. News credibility evaluation on microblog with a hierarchical propagation model. In* 2014 IEEE International Conference on Data Mining*, pages 230–239. IEEE, 2014.*

[6] *Zhiwei Jin, Juan Cao, Yongdong Zhang, and Jiebo Luo. News verification by exploiting conflicting social viewpoints in microblogs. In* Thirtieth AAAI Conference on Artificial Intelligence*, 2016.*

[7] *Eugenio Tacchini, Gabriele Ballarin, Marco L Della Vedova, Stefano Moret, and Luca de Alfaro. Some like it hoax: Automated fake news detection in social networks.* arXiv preprint arXiv:1704.07506*, 2017.*

[8] *Ceren Budak, Divyakant Agrawal, and Amr El Abbadi. Limiting the spread of misinformation in social networks. In* Proceedings of the 20th international conference on World wide web*, pages 665–674. ACM, 2011.*

[9] *David Kempe, Jon Kleinberg, and Éva Tardos. Maximizing the spread of influence through a social network. In* Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 137–146. ACM, 2003.*

[10] *Kai Shu, Deepak Mahudeswaran, Suhang Wang, Dongwon Lee, and Huan Liu. Fakenewsnet: A data repository with news content, social context and dynamic information for studying fake news on social media.* arXiv preprint arXiv:1809.01286*, 2018.*

[11] *Michela Del Vicario, Alessandro Bessi, Fabiana Zollo, Fabio Petroni, Antonio Scala, Guido Caldarelli, H Eugene Stanley, and Walter Quattrociocchi. The spreading of misinformation online.* Proceedings of the National Academy of Sciences*, 113(3):554–559, 2016.*

APPENDIX

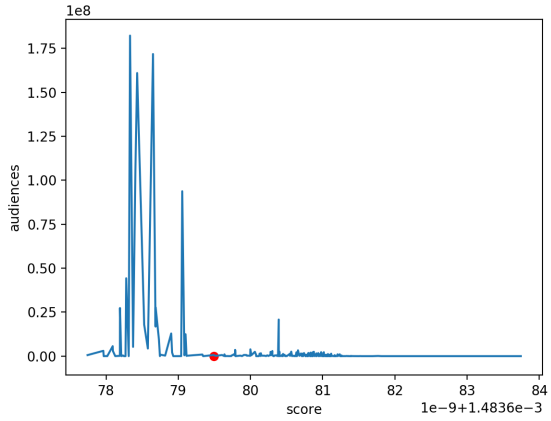

Fig. 1. Credibility propagation score of fake news vs. audience size without audience redistribution. Marked in red is the middle split used for classification

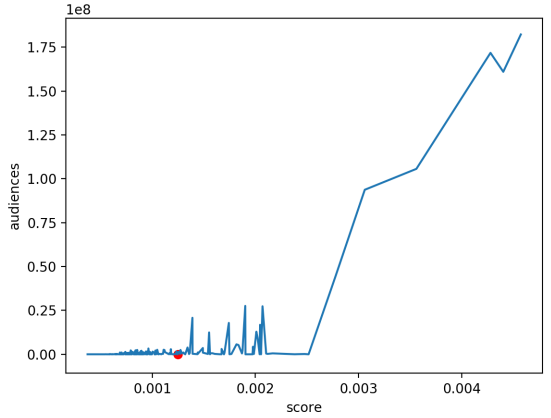

Fig. 2. Credibility propagation score of fake news vs. audience size with a 0.1 redistribution. Marked in red is the middle split used for classification