Fake News Detection Report

Clément Rouvroy*
clement.rouvroy@ens.psl.eu
Ecole normale supérieure
Paris, Île-de-France, FR

Grégoire Le Corre*
gregoire.le.corre@ens.psl.eu
Ecole normale supérieure
Paris, Île-de-France, FR

Niloo S?*
ns2434@caa.columbia.edu
?
?, ?, ?

ABSTRACT

Na,g

1 DATASET

1.1 Original dataset

Given a URL u, they build a graph G_u on tweets like this:

- T_u is the set of tweets related to the URL. A tweet is related to the URL if it mention it.
- There is an edge between two nodes (tweet) i, j if i follows j, or j follows i, or the news spread from i to j, or the news spread from j to i. By *news spreading* they mean that i is retweeting j.

On this graph, they define *spreading trees* by sorting the nodes by timestamps. The first node (the earlier one) is the root and then, given a node (tweet) t_u^i, if he follows an author of a tweet in $\{t_u^0, \ldots, t_u^{i-1}\}$ then there is an edge between the very last tweet in $\{t_u^0, \ldots, t_u^{i-1}\}$ that has been written by an author the author of t_u^i is following. If he follows no one then the edge is between the tweet from the most influent user (the most recent from it) and t_u^i.

Hence here they have access to two distinct objects: the graph G_u and the spreading trees. Likely for us (otherwise this would be impossible to replicate), there model is learning on the spreading trees, called *cascades*. Moreover, the URL information is only used to classify the cascades between the fakes and the reals. They have around 10^5 cascades in total.

Each node is composed of a feature vector that encodes user profile information, content of the tweet and network and spreading information (social connections between the users, number of followers and friends, ... *i don't understand how they manage to do this, if someone know I would like, are they using transformer ?*)

1.2 Our dataset

Our dataset is GNN-FakeNews and it was extracted using the FakeNetTools, that requires an X's api key to generate dataset (which is limited to a hundred pull per month so not really usable for us). This dataset consists of roughly 6000 cascades so is less important, but is built the same way. The only difference is that a node has a feature that is the concatenation of a vector representing its profile informations and a vector that has been created using BERT on the content of the tweet.

2 MODEL

This is the same as the paper Niloo, see page 6, figure 5, we even took the Hinge Loss and the SELU activation function.

*Both authors contributed equally to this research.