Geometric Data Analysis: Fake News Detection on Social Media using Geometric Deep Learning

Roman Plaud*
roman.plaud@eleves.enpc.fr
ENS Paris Saclay

Julie Alberge*
julie.alberge@eleves.enpc.fr
ENS Paris Saclay

ABSTRACT

The objective of this document is to summarize and criticize [6]. In the first part, we describe the different state of the art approaches to detect fake news. We also link this work to a much broader geometric deep learning context. Then, we reimplement the convolutionnal graph neural network model described in [6], and show some of our results on another dataset named UPFD. We obtained a ROC AUC score of 98.91% and an accuracy of 95.71% on the test set on the fake news classification task.

We also discuss the limitations of the article such as its interpretability and we propose to use a random features framework in which we show that the sole propagation structure is relevant to detect fake news. We obtain interesting results with an accuracy of 78.12% accuracy on test set.

KEYWORDS

Geometric Deep Learning, Fake News Detection

1 INTRODUCTION

Detecting fake news via algorithms based on social media data is a challenging issue. Numerous approaches do exist due to the danger of spreading misleading news or disinformation on social media.

The goal of this paper is to summarize [6], and re-implement the algorithm which detects fake news on Twitter. Using geometric deep learning, a type of deep learning method that works on graph-structured data, we demonstrate the effectiveness of their approach on a real-world dataset.

Firstly, we will describe the scientific context of this article by presenting the different existing algorithmic methods to solve our problem and an introduction to geometric deep learning.

Then, we will focus on the method proposed by the authors, a graph attention network. After explaining our implementation, we will display and comment our own results.

Finally, we will highlight the limitations of the article and propose an extension, by randomizing the node features.

2 SCIENTIFIC CONTEXT

2.1 Fake News context

Today, according to the Pew Research Center study [2], 71% of Americans get their information at least sometimes from social media. This leads to some big issues because social media is not fact-checked so it has the potential to spread false or misleading information, known as "fake news." Moreover, identifying a fake news may be hard to an non-expert person and people get often mislead, as during the Brexit campaign or 2016 US election.

First of all, there is no clear consensus on what constitutes "fake news," meaning that some consider only voluntary disinformation as fake news whereas others consider voluntary and misleading information as fake news.

Therefore, detecting fake news via an algorithm is a challenging task that requires understanding the nuanced interpretation of news, knowledge of political or social context, and "common sense", which of course are difficult even for the most advanced natural language processing algorithms.

There have been various approaches to detect fake news, including content-based approaches, social context, and propagation-based approaches. Content-based approaches rely on linguistic features that can capture deceptive cues or writing styles, but they can be defeated by sophisticated fake news and are often language-specific. Social context-based approaches use features such as user demographics, social network structure, and user reactions. Propagation-based approaches study the news proliferation process over time and have shown that fake news spreads differently from true news, forming patterns that could potentially be used for automatic detection. The latter are seen as particularly promising as they are content-agnostic and potentially generalize across different languages.

2.2 Geometric Deep Learning context

The term geometric deep learning is commonly used to describe non-Euclidean deep learning approaches : approaches that do not require an assumption of grid-structured data. This area has received increasing attention in recent years due to the growing importance of graph structured data in a wide range of applications.

Geometric deep learning allows for the incorporation of prior physical knowledge into neural networks architectures and [5] shows that it has been applied in various fields such as healthcare (drug discovery, proteins properties prediction...), social media analysis (community detection, link prediction, fake news detection...) or even astronomy.

Nevertheless, geometric deep learning is a relatively new and rapidly evolving field of study within the broader field of machine learning. While significant progress has been made in the development of neural network architectures and learning algorithms that can operate on non-Euclidean structured data, there is still much research to be done in this area.

Originally, graph-structured data was dealt through spectral graph CNN. Firstly mentioned by [4], convolution operations were performed in the spectral domain. We call spectral domain the spectral decomposition of the Laplacian matrix (normalized adjacency matrix of the graph) associated to a graph. These methods marked a milestone in graph theory. However, it suffered from very high computational complexity as spectral decomposing of an adjacency matrix is a tough task when dimensionnaly is high.

Then, more recently, new graph techniques emerged such as Graph convolutionnal network, that perform a generalized type of convolution on a node and its neighbors (nodes that are linked by an edge). Message massing layers [3] and graph attention layers [7] became key elements to deal with graph-structures data. This article is part of these evolution.

3 PROPOSED METHOD

The architecture proposed in [6] is a four-layer graph convolutional neural network (Graph CNN) with two attention convolutional layers and two fully connected layers. It uses mean-pooling for dimensionality reduction, and employs a hinge loss function and scaled exponential linear unit (SELU) as activation function. The model takes as input a graph representing the news diffusion tree with tweets as nodes and news diffusion paths and social relations as edges. It also uses node features that gather information about user profile, user activities, user network and content of the news.

The authors tested their model on two different settings for fake news detection: URL-wise, in which they attempted to predict the true/fake label of a URL containing a news story from all the Twitter cascades it generated (that is all tweets citing this URL and all retweets of these tweets), and cascade-wise, in which they assumed they were given only one cascade arising from only one tweet citing a URL and attempted to predict the label associated with that URL. They used five randomized training/test/validation splits (5-fold) for URL-wise classification and the same splits for cascade-wise classification. In the URL-wise setting, the model achieved an average ROC AUC (area under the receiver operating characteristic curve) of 92.70 ± 1.80%. In the cascade-wise setting, the model achieved an average ROC AUC of 88.30±2.74%. They also found that the model's performance increased with larger cascades and reached saturation for cascades with at least 6 tweets. They also found that the model's performance was not significantly affected by the time duration of the cascades. In addition, they found that using user descriptions in their model improved performance, but using tweet content did not.

3.1 Dataset

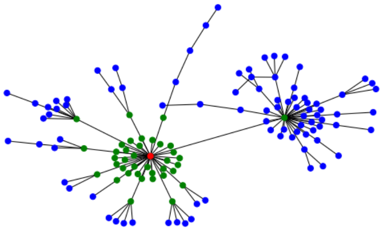

Figure 1: A URL and all its cascade. Red vertex represent the news. Green vertices represent users that tweeted the news and blue vertices represent users that retweeted the tweet mentioning the news

The dataset that the autors used in [6] is not publicly available so we could not exactly fit into the same framework. Thus results will not be comparable.

To overcome this, we used the UPFD (User Preference-aware Fake News Detection) developped by [1]. In this dataset, the proportion of fake news is approximately equal to 50%, which is quite different from the proportion in [6]. This dataset contains diffusion of news through twitter. The construction is slightly different from [6]. Each news diffusion on Twitter is represented by a non-oriented graph in which the root is the news itself (not the tweet in which the news is cited). All the neighbors of the news are the tweets citing the URL of the news and all others edges represent retweets of the initial tweets in which the news URL is cited. We display in Figure 1 an example of a news diffusion in the UPFD dataset.

Data Origin	Number of Graphs	Number of Fake News	Total Number of Nodes	Total Number of edges	Avg. Nodes per Graph
Gossipcop	5464	2732	314,262	308,798	58

Table 1: Statistics about the Gossipcop UPFD dataset

The characteristics of the dataset is summed up in Table 1. Until now, the UPFD is similar to the the one in [6]. However there are two main differences.

- Social connections
  In [6] they use also social connections as edges in graphs. By social connections we mean followers. If a user follows another user it has a directed edge from its node to the other user' node.
- Node features
  For the UPFD dataset we have 4 types of nodes features :
  – Spacy: a 300-dimensional word2vec embedding of the historical content of the user' profile
  – Bert : a 768-dimensional Bert embedding of the content of the user' profile
  – profile : a 10 dimensional features vector gathering all relevant information of the user' profile (name, tweetname, number of followers etc.)
  – content : two concatenated feature vectors of dimension 300 and 10. The first one is the word2vec embedding of the content of the tweet and the second one is the profile vector mentioned above.
  For the dataset of [1] node features
  – User profile (geolocalization and profile settings, language, word embedding of user profile self-description, date of account creation, and whether it has been verified)
  – User activity (number of favorites, lists, and statuses)
  – Network and spreading (number of followers and friends, cascade spreading tree, source device, number of replies, quotes, favorites and retweets for the source tweet)
  – Content (word embedding of the tweet textual content and included hashtags)

3.2 Implementation

Our implementation is available at this link. Using the code provided by [6] and [1] we had no difficulty implementing the model. The hyperameters used are the same as used in [1]. Concerning the model we used a four-layer Graph CNN with two attention messages passing layers (64-dimensional output features map in each) followed by a mean-pooling and two fully connected layers (producing 32- and 2-dimensional output features, respectively) to predict the fake/true class probabilities. We worked on the simplest framework possible: a URL-wise setting (very little preprocessing was needed given the provided dataset). We did not work on cascade size nor duration diffusion.

3.2.1 Ablation study. As in the article, to highlight the performances of the different settings we implemented an ablation study to better understand the importance of each type of features. To perform this we used a 5-fold cross-validation framework. To measure our results, we compute both accuracy on the validation set as well as the ROC-AUC metric. We show in Figure 2 and 3 our results.

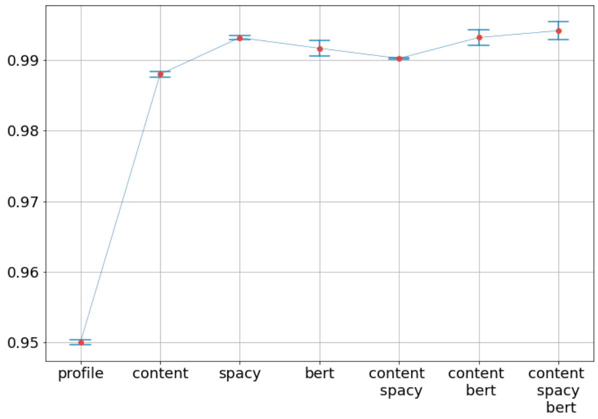

Figure 2: Ablation study result on URL-wise fake news detection. Shown is performance (ROC AUC) for our model trained on subsets of features

We observe that all features achieve more than 95% of ROC AUC metric on average on validation set in our 5-fold framework. The best results are achieved by concatenating all features (content, Spacy and Bert) with a $ROC - AUC$ reaching 99.5%. This metric which is taken as reference in the article yet, it might not be fitted to our dataset. In fact, as shown by 1 is very well balanced with an exact proportion of 50% true news and 50% fake news. We may then focus on the accuracy metric results that are shown in 3. Interestingly, and as mentioned by [6], in our URL-wise setting removing tweet content improve accuracy. In fact, if we use Spacy and tweet content as node features the method achieves 96.50 ± 0.22% whereas removing tweet content we reach 97.00 ± 0.30% which is statistically a significant improvement at a 10% level (t-test was performed). We observe similar behaviour for when we use Bert features instead. Another element which is worth to mention is the relative instability of adding more and more features. For example, in the setting in which we add all features that is content, Spacy and Bert, we observe a significant increase of the length of the 95% confidence interval.

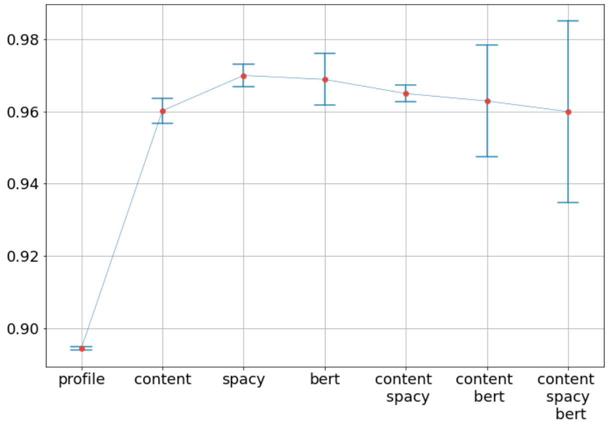

Figure 3: Ablation study result on URL-wise fake news detection. Shown is accuracy for our model trained on subsets of features

3.2.2 Training of the best model. Accordingly to Figure 3 results, and to be able to compare to benchmarks available, we retrained our model only with Spacy features.

Our neural network was trained for 100 epochs, using Adam optimizer with a learning rate of 10^{-3} (with a 10^{-1} decay every 20 epochs) and batch size of 128. Figure 4 depicts the ROC curve of our trained model, that is the trade-off between false positive rate and true positive rate. We obtain a ROC AUC of 98.91% on the provided test set as well as an accuracy of 95.71% which does not match perfectly the benchmark provided by [1] which reach 96.38% of accuracy with our model and our dataset. With super fined hyperparameter tuning we could have reached the benchmark.

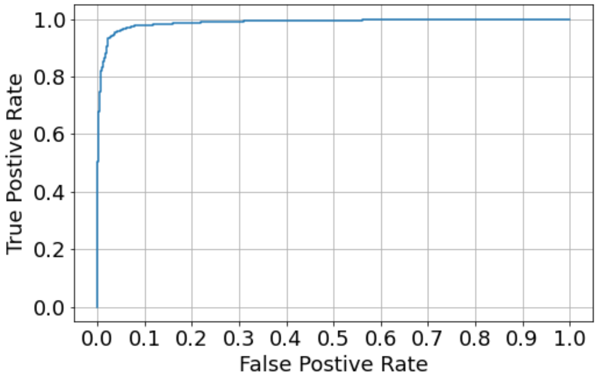

Figure 4: ROC curve of Spacy model on test set

We then display in Figure 5 a 2-dimensional plot of the node embeddings resulting from the output of the last message passing layer of the graph neural network. Each node embedding is colored using their credibility, that is the ratio of the fake news they propagate through their tweets or retweets over the total amount of news they tweet or retweet. As in the article we observe clear clusters of reliable users and not reliable users which shows the abality of the network to separate the different users.

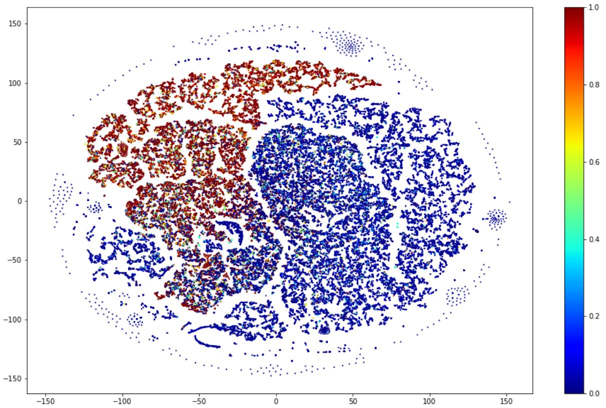

Figure 5: T-SNE node embedding resulting from the output of the last message passing layer our of graph convolutionnal network trained on Spacy node features. Color code is related to user credibility

4 LIMITATIONS AND EXTENSIONS

4.1 Limitations

4.1.1 Dataset. The first limitation that we can highlight is the dataset. Because their dataset isn't available online, we couldn't test other algorithms with their architecture to see which technique may be better compared to the others ones. Moreover, they explain that in their dataset the top-15 most cited URLS represent more than 20% of the total number of the cascades. Given that the content of two tweets citing the same URL may be very similar, we can provide an explanation of why we observe a decrease in the ROC AUC metric. In fact, the model could have overfitted on the content (being very similar) of tweets citing these very represented URLs.

4.1.2 Interpretability and transfer learning. Despite of the embeddings, we can note that the neural network can be hard to interpret. We cannot clearly understand if the neural network learnt some intrinsic features of `Gossipcop` or on the spread of fake news. We do think that, for example, gossip fake news may spread differently than COVID fake news. Moreover, we do not know if the network may have overfitted as it is written above. This may be an argument to explain why our results are different from the ones in the article, which means than our model may be not transferable to another dataset. In this idea, [8] highlights that, by only training their geometrical neural network on `Gossipcop`, the ROC AUC score on the other dataset `Politifact` wasn't satisfying (around 0.6 in ROC AUC).

4.2 Extensions

The article we studied suggests that the structure of the graph contain the information which is necessary to detect fake news. In fact, they explain that removing content from their features enhanced performance and accuracy. In fact, in all their frameworks [6] they always use node features. Then, we thought of a method that allowed us to get rid of all features to be able to concentrate on the structure of the graph only.

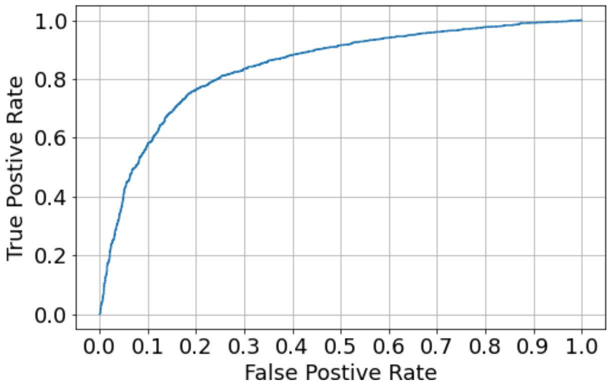

Figure 6: ROC curve of random features model on test set

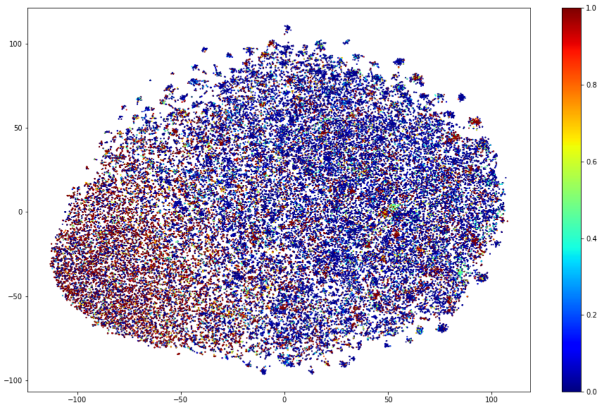

Figure 7: T-SNE node embedding resulting from the output of the last message passing layer our of graph convolutionnal network trained on random node features. Color code is related to user credibility

To do so, we implemented a *random features* framework in which at each iteration, each node feature is initialized with a random 10-dimensional vector features. Thus, absolutely no textual information is given to the network, and it has to learn to classify news from the structure and patterns of tweets and retweets. It appears to be a complex task because intuitively we see no clear explanation of why a fake news would propagate differently. Yet, in this framework we reach 78.12% accuracy and 84.27% ROC AUC on the test set. Figue 6 shows the ROC curve.

That work let us think that there is relevant information on the structure of the propagation of news itself. The network can detect patters of diffusion that are specific to a fake or a true news. We then display, as before, in Figure 7 a 2-dimensional plot of the node embeddings resulting from the output of the last message passing layer of the graph neural network. We see the network struggles much more than before to separate reliable users from unreliable ones. Yet, we can observe a clear tendency of unreliable users to be on the bottom left part of the plot whereas reliable users are everywhere else.

These results indicate that even if we do not achieve extraordinary accuracy with the proposed method, it shows that there is relevant information on the structure of the propagation news itself.

5 CONCLUSION

On the whole, we have seen, with paper written by Monti and al. [6], that with an accessible geometric deep learning architecture, we can obtain satisfying results.

Using the `Gossipcop` graph dataset from the UPFD dataset, on the URL-wise setting, we tested the model applying a 5-fold on our dataset. The ablation study showed that all features were relevant but `Spacy` reach the best accuracy obtaining 95.71% on test set.

We also displayed a low-dimensional users embeddings in function of their credibility and it showed that the network was able to separate reliable users from unreliable users.

Finally, we replaced the node features by randomised vectors and still obtained interesting results (78.12% accuracy on test set). We also noted the lack of interpretability of our network explaining how could it be an issue. Indeed we cannot clearly see what the network learns and how much the model can be transferred to another dataset.

REFERENCES

[1] Yingtong Dou, Kai Shu, Congying Xia, Philip S. Yu, and Lichao Sun. 2021. User Preference-aware Fake News Detection. https://doi.org/10.48550/ARXIV.2104.12259

[2] Amy Mitchell Elisa Sherer. 2021. News Use Across Social Media Platforms in 2020. https://www.pewresearch.org/journalism/2021/01/12/news-use-across-social-media-platforms-in-2020

[3] Justin Gilmer, Samuel S. Schoenholz, Patrick F. Riley, Oriol Vinyals, and George E. Dahl. 2017. Neural Message Passing for Quantum Chemistry. https://doi.org/10.48550/ARXIV.1704.01212

[4] Arthur Szlam Yann LeCun Joan Bruna, Wojciech Zaremba. 2014. Spectral Networks and Deep Locally Connected Networks on Graphs. https://arxiv.org/pdf/1312.6203.pdf

[5] Taco Cohen Petar Veličkovi Michael M. Bronstein, Joan Bruna. 2021. Geometric Deep Learning Grids, Groups, Graphs, Geodesics, and Gauges. https://arxiv.org/pdf/2104.13478.pdf

[6] Federico Monti, Fabrizio Frasca, Davide Eynard, Damon Mannion, and Michael M. Bronstein. 2019. Fake News Detection on Social Media using Geometric Deep Learning. https://doi.org/10.48550/ARXIV.1902.06673

[7] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2017. Graph Attention Networks. https://doi.org/10.48550/ARXIV.1710.10903

[8] Christopher Leckie Yi Han, Shanika Karunasekeran. 2020. Graph Neural Networks with Continual Learning for Fake News Detection from Social Media. https://arxiv.org/pdf/2007.03316.pdf