Misinformation Detection On Reddit Using Post-to-Post Networks

Shaunak Divine
Cockrell School of Engineering, University of Texas
Austin, Texas, United States
shaunak.divine@gmail.com

Stephen Young
Cockrell School of Engineering, University of Texas
Austin, Texas, United States
stephensouthworthyoung@gmail.com

Abstract

This paper explores the application of a Graph Convolutional Network (GCN) to predict the credibility of URL shares in the r/Conservative subreddit. By categorizing URLs shared into two distinct categories – credible and untrustworthy – we aim to understand how users interact with misinformation, and predict if a post sharing a URL on Reddit is credible. Our approach utilizes a GCN that is trained on a post-to-post network using user reaction scores as edge weights and text embeddings from the original post. With this we achieve ~70% classification accuracy.

I. Introduction

With the spread of misinformation more prevalent than ever, it is more difficult for social media users to separate the real events from the fake. As people everywhere expand their online presence, more news is being absorbed through social media. Yet, as accessibility to news becomes easier, the risk of spreading a deliberately fake article increases. Moreover, catching fake articles is a time-consuming process that often is unable to work quickly enough to identify misinformation before it spreads across users. Many social media platforms have slow or inadequate methods to flag untrustworthy URLs, allowing misinformation to transfer uncontained. The main challenge behind this lies in accurately classifying these shared URLs as credible news or unreliable news, as well as predicting the amount of community interaction this information will generate. To solve this, we aim to use a novel approach of combining the use of a post-to-post network with user sentiment weighted edges and text embeddings to train a GCN to accurately classify shared URLs as credible or untrustworthy. Our contributions thus far are as follows: Stephen has concentrated on the data cleaning and embeddings while Shaunak on building the graph and integrating graph structure in the GCN module. Stephen and Shaunak both worked on the creation and training of our GCN model.

II. Previous Works

When investigating Reddit as a platform, we found that it is very conducive to discussion of current and political events due to its open forum based interaction system. While this makes it more accessible, the issue is Reddit does not have a full-proof fact checking system, as it relies on community members called moderators or admins to identify misinformation, leaving users to identify misinformation for themselves [11]. Even with these admins, studies have found that less than 5% of Reddit posts are flagged as misinformation, leading us to believe that misinformation may be spreading unchecked [1]. In many past examples of misinformation identification, the primary method used is natural language processing (NLP) [7]. In one such example, de Oliveira et al. use NLP to analyze text and determine if it was likely written by an AI based on the word choice and ordering [5]. Particular to Reddit, NLP has been used to classify health related misinformation with high accuracy in specific contexts, showing its potential for automated detection [13]. While this approach is effective, it takes careful analysis of entire articles and could be more efficient. Building on this, another method to investigate fake news is sentiment analysis to identify suggestive or eye-catching writing style [2]. This works well with NLP and finding AI generated text but also requires full text analysis. With over 138,000 active subreddits and over 10 million monthly users, this makes application of NLP to every post tedious and difficult to scale [3]. One example of a particularly powerful NLP model is BERT (Bidirectional Encoder Representations from Transformers) [6]. BERT's distinctive feature lies in its method of conditioning on both left and right contexts across all layers, a significant difference from traditional NLP models. This allows it to be fundamentally simple while extremely powerful for NLP. In our study, while we will utilize BERT, our focus will not be on NLP. Instead, we will use BERT in combination with graph structure. Outside of text-based detection, there have been elaborate studies to detect misinformation based on graph structure. In one study by Michail et. al, a network of social media users, their connections, and misinformation transmittance data are analyzed to identify fake news campaigns [10]. In this study, a GCN is used and learns solely on the graph structure, as the text is completely omitted. We aim to take this approach a step further by combining text embeddings with graph structure to optimize our GCN's capabilities. There have also been studies examining the propagation of false news through a social network. While it is clear that misinformation traverses a network differently [9], Reddit presents itself as an outlier. Twitter and other similar social networks find that misinformation propagates faster and more widespread, but on Reddit, studies have shown that fact-checked news proven as true leads to more shares and creates more discussion [4]. With this information, there have been positive results for identifying misinformation on Reddit. Methods such as CountVectorizer and MultinominalNB have been used with approximately 90% success rate in classifying shared news as real or fake [12]. We aim to build on these results using a novel post-to-post network to classify misinformation through connections, sentiment, and text analysis.

III. Approach

Our approach centers around a post-to-post network to classify misinformation on Reddit posts. To build this network, we use a dataset of posts and comments from r/Conservative, the largest conservative subreddit. This dataset is made up of posts p, where p has author a_p, hyperlink (news URL provided) l_p, and a score s_p associated with it. The score is calculated based on the amount of interaction a post receives in r/Conservative. In this dataset, each post p has a set of comments from users on p denoted as c_p, and A_p is the set of all users who commented on post p. In a post-to-post network, we define the graph as G, the set of nodes as V, and the set of edges as E. Graph G is represented by the following equation: $G = <V,E>$. Each post p is a node $p \varepsilon V$ in an undirected graph. An edge $e \varepsilon E$ connects two posts $p,q \varepsilon V$. An edge e is formed between two posts p and q if $A_p \cap A_q \neq \varnothing$ where $A_p \cap A_q$ represents the set of users who commented both on post p and post q. That is, post p and q are only connected by e if there is a user that comments on both posts. Our post-to-post network implementation is illustrated in Figure 1.

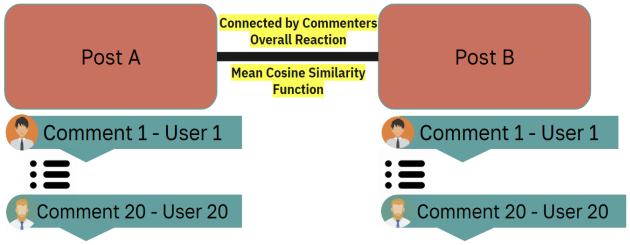


Figure 1: Model of our post-to-post network: posts are nodes and shared commenters are undirected edges. Edge weight is calculated using cosine similarity between comments.

Each edge e also has a weight ω that is calculated based on a comment sentiment function. To do this, we attempted two different sentiment implementations. First, we began by implementing TextBlob's subjectivity and polarity functions. These would provide each comment with a subjectivity – how opinionated or factual the text is – and a polarity – how positive or negative a text is. Once these were calculated for each comment pair, we averaged these values for each post in the network, and attached these as edges between two posts. However, we found that this approach limited the value we could pull from the number of comments a post had accrued. This also seemed to cause our model to overfit more, so we pivoted to a second approach. We moved from TextBlob's subjectivity and polarity to a holistic view of comment reaction. The goal of this was to give each edge an overall reaction score that would provide more insight on the connection between the posts and user comments. To calculate this, all the comments from the posts are embedded and these are then evaluated against each other based on cosine similarity. This is then averaged across all shared users on any two connected posts in the network. This calculation would estimate the overall reaction users had towards a post and could describe if two posts had similar or different sentiment. The edge weight would then be added to the network accordingly. From this, we assume that if two posts engage the same user, that they are likely to be similar. Furthermore, if two posts are engaging similar users at greater numbers (i.e. the edge weight between the two posts is greater), they are even more similar. Using hyperlink l_p, we are able to retrieve a standard reliability score r for the domain associated with l_p from the FACTOID dataset [8]. The reliability score is calculated based on the history of the website related to the domain, and how much misinformation they shared from 2018 -2021. We gather this from a dataset of news source domains that attaches a normalized reliability score to each. Using this score we can generate two disjoint classes. These classes are credible links and untrustworthy links. We generate these classes with the threshold of 0.55 reliability. That is, any link l_p under 0.55 reliability is untrustworthy, and every l_p above 0.55 is credible. Once we have formed the network and classified the nodes, we begin cleaning our data. We first filter our data by year, using only submissions from 2016 through 2020. When creating edges in our network, we only connect posts p and q if $|A_p \cap A_q| >= 20$. This means that we only create an edge between p and q if they share 20 similar commenters between the two posts. This is to account for bots that are commonplace in every reddit community (ie. u/stabbot, u/RemindMeBot, u/getvideobot, etc.) and to increase confidence that posts are connected only if real users comment on both of them. We also only consider posts if the score of p, s_p, is greater than 1. This is to ensure we use posts that generate at least mild interaction in r/Conservative. Lastly, we ensure $|c_p| >= 50$, meaning we only use posts with a minimum of 50 comments. Once we have a fully filtered graph, we begin generating text embeddings using BERT- specifically the bert-base-nli-mean-tokens model- on the titles of each post. BERT is an advanced method in natural language processing that interprets the context of words in a sentence by analyzing text in both directions simultaneously and outputs a set of embeddings representing the input text. These text embeddings are the input to the first layer of our model, seen in Figure 2.

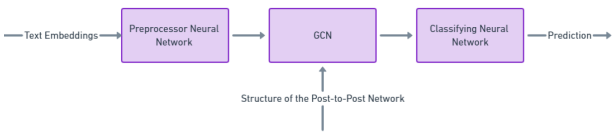


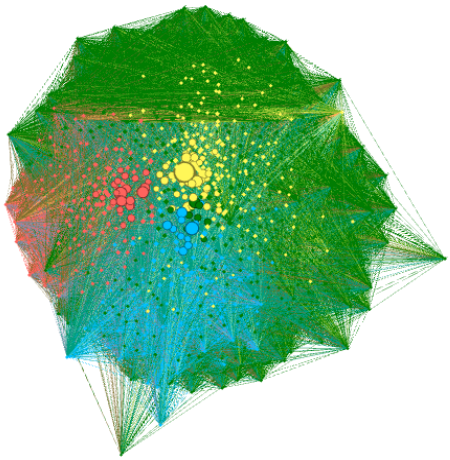
Figure 2: A flowchart of our model that predicts whether a URL shared in a post on r/Conservative is credible or not. It contains three layers: a preprocessing layer, a GCN, and a classifying layer.

We pass these embeddings into a preprocessing neural network (NN). The embeddings begin as size 768, but through the first NN they are condensed down to 64 in order to make them more digestible for our GCN. The output of this NN becomes the input to our GCN as well as the graph structure of our post-to-post network. Graph structure refers to an edge index in addition to the edge weights. This GCN is the second layer. Lastly, the output of this GCN is the input to our third layer, which is a classifying NN. The output of this is the class prediction of whether the post's URL is credible or untrustworthy. Our GCN approach has proven scalable thus far, as we have run initial tests of the classification on multiple years of data.

To determine the success of our model, we calculate the accuracy of the model's predictions. We calculate accuracy by comparing the model's predicted class to the target class of each post. Given the reliability score, we are able to classify the nodes into two classes for the testing set. We also output a scatterplot of the model's predictions to ensure the model is assigning each class at a similar rate.

IV. Experimental Setup and Results

We first created our post-to-post network in our dataset exploration ,enabling us to identify clear communities and the structure of our network. Using our graph approach on r/Conservative data from 2016, we created a network with 7278 nodes and 536,596 undirected edges. This network had an average degree of 147, representing high connectivity in our graph. This demonstrates that many of the posts had a large amount of shared commenters, and thus, high interaction on the post. Our graph also illustrated small-world properties, as we observed a clustering coefficient of 0.696 and an average path length of 1.5. This graph also demonstrated seven distinct communities that we identified from our subreddit. We were able to see a clear connection between the communities and pockets of users who were credible or untrustworthy. These are seen below in Figure 3(a) and 3(b).



(a)

(b)

Figure 3: (a) Graph of the post-to-post network for subreddit r/Conservative for posts from 2016. (b) Same network graph, but colored based on credibility of a post: blue meaning more credible while red is untrustworthy. The size correlates to the number of total comments on the post.

Figure 3(a) shows a graph partitioned into communities generated using Louvain's Method, where nodes are grouped based on maximizing modularity. Each color is a different community. Figure 3(b) is the same graph but instead of being colored by community, the nodes are colored to indicate the credibility of a post, blue indicating credible and red indicating untrustworthy. The size of the node represents the number of comments: the greater the size, the more total comments, the smaller the size, the less total comments. Using our GCN classifier on this network, we were able to achieve approximately a 70.13% classification accuracy score. The scatterplot of the predictions from our model is shown in Figure 4.

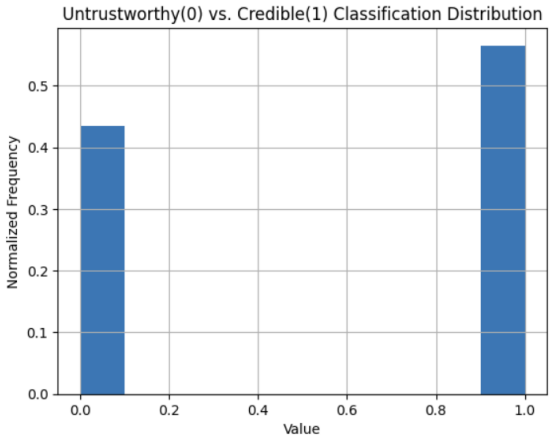
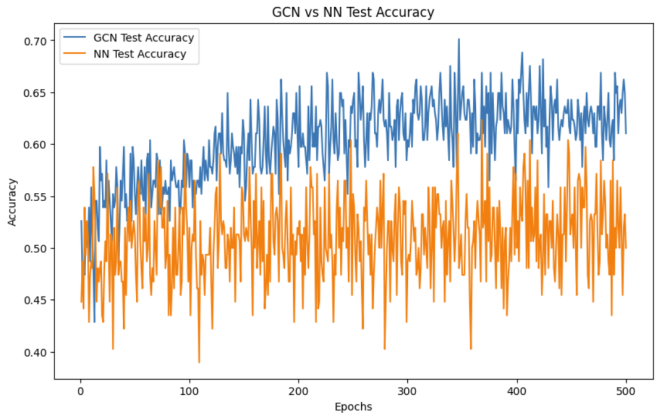
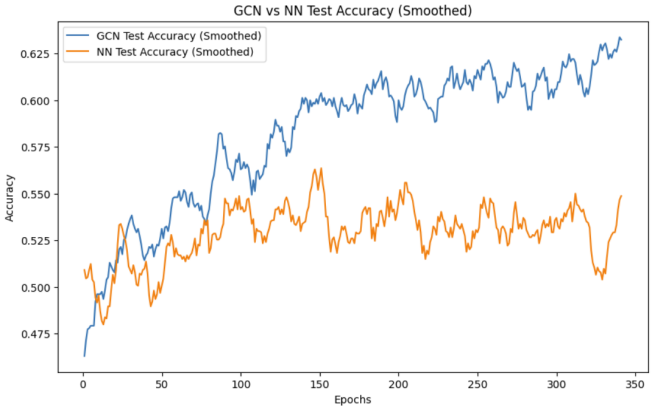


Figure 4: A scatterplot of the predictions from our GCN model with a ~70% accuracy score. Class 0 is untrustworthy, while class 1 is credible.

Without using graph weights in our classification method, we were only able to achieve classification accuracy of approximately 61%. Furthermore, when applying a convolutional neural network (CNN) on our data instead of our GCN model, we can see a significant drop in classification accuracy. A CNN does not use graph structure in its model, and thus, we can see that our post-to-post network structure and edge calculations enhanced the model. The results of the GCN vs. the CNN model are displayed in Figure 5.



(a)



(b)

Figure 5: (a) Graph of GCN vs. CNN testing accuracy across 500 epochs with all data points included. (b) Graph of GCN vs. CNN testing accuracy but with values smoothed to average data at every 10 epochs, reducing noise.

Since our graph weights represent the number of users interacting with two posts and their reactions to the posts, this appears to validate our assumption that if many users are interacting with the same two posts with similar sentiment, then these posts must be similar. This also points to the conclusion that there is a clear correlation in the way that people interact with posts whether they are sharing credible or untrustworthy news. This is further supported in the graphs from Figure 3. Between these two representations, the yellow community in Figure 3(a) clearly correlates to the pocket of blue nodes in the center of Figure 3(b). This illustrates a community of highly connected posts that are largely credible posts and shows that some users heavily interact with only credible posts. Contrarily, the red community in Figure 3(a) and the dark red cluster near the center left side of Figure 3(b) communicate the opposite of the first example. In this case, the graphs illustrate a strong community of mostly untrustworthy posts. Since the edges represent people interacting with similar posts, it can be postulated that there is a significant amount of people that only interact with untrustworthy posts. Both of these examples further support our conclusion that there is a clear correlation between how people interact with a post and if that post shares credible information. It is also interesting to note that our model had a training accuracy of upwards of 96%. This points to signs of overfitting, and although we attempted to mitigate this, we could not find a way to implement the graph structure without overfitting to some degree. This shows that our model was able to learn very effectively, but struggled to generalize to the data as a whole. While it still out-performed a comparable CNN, we would like to continue to explore the GCN's discrepancy between learning and generalizing in the future.

V. Conclusion and Future Work

With our post-to-post network and GCN classification model, we have shown promising results for classifying misinformation on Reddit, reaching an accuracy of 70.13%. Our research contributes to the evolving field of misinformation detection by introducing a novel approach that combines text embeddings with post-to-post graph structure in a GCN framework. The relationship observed between user interactions and the credibility of shared URLs underscores the potential for our model in identifying and containing misinformation within other online communities. As this method develops, we hope to influence the spread of misinformation on Reddit, providing the platform with a more robust algorithm than simply relying on human administrators. We would like to continue developing this approach in hopes to combat the challenges presented by overfitting. We hope to apply this to more areas of Reddit to understand the misinformation transmission on other types of communities. This more diverse data would hopefully help our model learn to generalize more effectively. Ultimately, it would be interesting to see this applied to more community-based social media platforms, and we hope this tool will prove useful as it continues to develop.

VI. Contributions and Lessons Learned

Through Milestone 3, our team has split work evenly, with both Shaunak and Stephen working on the development and training of our GCN model. Stephen focused more heavily on the cleaning and embedding while Shaunak focused on creating the graph and implementing graph

structure in the GCN module. During this project, we have learned a significant amount about GCN's, research design, and machine learning models as a whole. Prior to this project, Shaunak and Stephen did not have any sort of experience with GCN's. Through this project we were able to understand how they work and their purpose in machine learning. One large lesson learned was to ensure that in each step of training the model, that your test, train, and validation sets are completely separate. We ran into an issue with data leaking, and we originally believed that our model had a test accuracy of upwards of 90%. Because of this, we wasted more time trying to make this model better and experimenting with graph attention networks (GATs) instead of trying to figure out how to make our model generalize more efficiently. It is now extremely clear how important handling data effectively and efficiently is. Additionally, we gained insight into the importance of viewing individual components as interconnected parts of a network, rather than as isolated elements. From our CNN, we can see that when viewing each post as a single entity, it is rather difficult to see if that post is sharing fake news. However, if you provide that post more context, such as how people interact with that post and others, the broader perspective enhances our ability to classify and generalize. Lastly, the importance of a dataset's quality cannot be overstated. Initially, our project was based on a different dataset, leading to a slightly altered project scope. However, as we progressed in developing our model and refining our algorithm, it became evident that our existing data was insufficient to achieve what our project goal was. This caused us to completely restart, and highlights the critical need to verify a dataset's quality prior to starting our model and algorithm development. This loss of time cost us in productivity, which we could have used to further enhance our approach and model. Further, poor-quality data can lead to rather poor insights, and can potentially lead to an overall ineffective model. This is even more important if we were operating in a field such as financial forecasting or healthcare, as our model's output could have far reaching and detrimental impacts on real people. So, ensuring the quality of a dataset from the very beginning is not only important in maintaining efficiency for one's project, but is a requirement for the reliability and validity of the end results. This experience has firmly implanted this idea into both Stephen and Shaunak.

References

[1] Achimescu, V. et al. "Raising the Flag: Monitoring User Perceived Disinformation on Reddit"Information 12, 2021 no. 1: 4. https://doi.org/10.3390/info12010004

[2] Alonso, M.A. et al. "Sentiment Analysis for Fake News Detection." Electronics 2021, 10, 1348. https://doi.org/10.3390/electronics10111348

[3] Amaya, A. et al. "New Data Sources in Social Science Research: Things to Know Before Working With Reddit Data." Social Science Computer Review, 2021, 39(5), 943-960. https://doi.org/10.1177/0894439319893305

[4] Bond, R. et al. "Engagement with fact-checked posts on Reddit," PNAS Nexus, Volume 2, Issue 3, March 2023, pgad018, https://doi.org/10.1093/pnasnexus/pgad018

[5] de Oliveira, N.R. et al. "Identifying Fake News on Social Networks Based on Natural Language Processing: Trends and Challenges." Information 2021, 12, 38. https://doi.org/10.3390/info12010038

[6] Devlin, J. et al. "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," arXiv, 2018. https://arxiv.org/abs/1810.04805.

[7] Hangloo, S. et al. "Combating multimodal fake news on social media: methods, datasets, and future perspective." Multimedia Systems 28, 2391–2422 (2022). https://doi.org/10.1007/s00530-022-00966-y

[8] Hwang, JD. et al. "FACTOID: FAirness-aware Curation of Training Instances for Open Information extraction Dataset," in Proceedings of the 13th Language Resources and Evaluation Conference, Marseille, France, 2022, pp. 3036-3046. https://aclanthology.org/2022.lrec-1.345/.

[9] Meyers, M. et al. "Fake News Detection on Twitter Using Propagation Structures." Disinformation in Open Online Media. MISDOOM 2020. Lecture Notes in Computer Science(), vol 12259. 2020. Springer, Cham. https://doi.org/10.1007/978-3-030-61841-4_10

[10] Michail, D. et al. "Detection of fake news campaigns using graph convolutional networks,"International Journal of Information Management Data Insights, Vol. 2, Issue 2, 2022. https://doi.org/10.1016/j.jjimei.2022.100104.

[11] New America, "How Internet Platforms Are Combating Disinformation and Misinformation in the Age of COVID-19: Reddit," New America, 2020. https://www.newamerica.org/oti/reports/how-internet-platforms-are-combating-disinformation-and-misinformation-age-covid-19/reddit/.

[12] Patel, A. et al. "Fake News Detection on Reddit Utilising CountVectorizer and Term Frequency-Inverse Document Frequency with Logistic Regression, MultinominalNB and Support Vector Machine," 2021 32nd Irish Signals and Systems Conference (ISSC), Athlone, Ireland, 2021, pp. 1-6, https://ieeexplore.ieee.org/abstract/document/9467842

[13] Sager, M. et. al, "Identifying and Responding to Health Misinformation on Reddit Dermatology Forums With Artificially Intelligent Bots Using Natural Language Processing: Design and Evaluation Study"JMIR Dermatol 2021;4(2):e20975 https://derma.jmir.org/2021/2/e20975