Following the Trail of Fake News Spreaders in Social Media: A Deep Learning Model

Antonela Tommasel
antonela.tommasel@isistan.unicen.edu.ar
ISISTAN (CONICET/UNCPBA)
Tandil, Argentina

Juan Manuel Rodriguez
juanmanuel.rodriguez@isistan.unicen.edu.ar
ISISTAN (CONICET/UNCPBA)
Tandil, Argentina

Filippo Menczer
fil@iu.edu
Observatory on Social Media
Indiana University
Bloomington, USA

ABSTRACT

Even though the Internet and social media are usually safe and enjoyable, communication through social media also bears risks. For more than ten years, there have been concerns regarding the manipulation of public opinion through the social Web. In particular, misinformation spreading has proven effective in influencing people, their beliefs and behaviors, from swaying opinions on elections to having direct consequences on health during the COVID-19 pandemic. Most techniques in the literature focus on identifying the individual pieces of misinformation or fake news based on a set of stylistic, content-derived features, user profiles or sharing statistics. Recently, those methods have been extended to identify spreaders. However, they are not enough to effectively detect either fake content or the users spreading it. In this context, this paper presents an initial proof of concept of a deep learning model for identifying fake news spreaders in social media, focusing not only on the characteristics of the shared content but also on user interactions and the resulting content propagation tree structures. Although preliminary, an experimental evaluation over COVID-related data showed promising results, significantly outperforming other alternatives in the literature.

CCS CONCEPTS

• **Information systems** → **Social networking sites**; • **Computing methodologies** → **Neural networks**.

KEYWORDS

User Profiling, Social Media, Fake News, Fake News Spreaders

ACM Reference Format:
Antonela Tommasel, Juan Manuel Rodriguez, and Filippo Menczer. 2022. Following the Trail of Fake News Spreaders in Social Media: A Deep Learning Model. In *Adjunct Proceedings of the 30th ACM Conference on User Modeling, Adaptation and Personalization (UMAP '22 Adjunct), July 4–7, 2022, Barcelona, Spain.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3511047.3536410

UMAP '22 Adjunct, July 4–7, 2022, Barcelona, Spain

ACM ISBN 978-1-4503-9232-7/22/07.
https://doi.org/10.1145/3511047.3536410

1 INTRODUCTION

In recent years, social media has profoundly changed how people consume information to the extent that it arises as one of the primary news sources [8]. One of the most valuable features of social platforms is their potential to propagate information on a large scale. However, the unmoderated nature of social media sites, and the potential for automation and fast propagation make it easy for users to share inaccurate or intentionally misleading information, thus threatening access to reliable and trustworthy information.

Fake or unreliable content can severely affect society, posing significant threats to democracies and economy. With the COVID-19 pandemic, health misinformation arose as a threat to public health, ranging from the viralization of harmful treatments to conspiracies. In turn, this can also affect how people perceive content [8]. First, repeated exposure can alter the likelihood of accepting fake content as truth, especially when the fake content aligns with internal beliefs [7]. Second, as fake content proliferates, the line between what is fake or not becomes more uncertain, inducing users to doubt the nature of all content and to think that all content is biased, hindering the differentiation between fake and authentic content [8]. In the long term, the trustworthiness of the entire news ecosystem might be at risk. Thereby, it is crucial to detect and mitigate the propagation of fake content.

Users play a fundamental role as creators and disseminators of fake content. Therefore, it is essential to detect both fake content and the users spreading it, as the latter will provide valuable information for the design of mitigation or intervention strategies to rapidly contain the spreading [21]. In this sense, the need to identify fake news spreaders on social media has never been more acute. This paper presents an initial proof of concept of a *deep learning model for identifying fake news spreaders in social media.* Our model includes not only features derived from the shared content, but also the content propagation trees and user community interactions. To support our proposal, we conducted a preliminary evaluation over a COVID-19 misinformation data collection. Results showed that the proposed model can effectively identify fake news spreaders when compared to traditional and state-of-the-art baselines.

2 RELATED WORK

The detection of fake content has been widely tackled in the literature. Initial works were usually based on linguistic characteristics [24]. However, as fake news are designed to be misguiding, it might be challenging to distinguish them by the text alone. Hence, some approaches included social context information in the detection [25], such as user profile and network features [17]. Recent

advances include using convolutional and recurrent neural networks to learn temporal representations from textual features and, in some cases, content propagation networks [11].

Similar techniques have been developed for fake news spreaders identification. For example, Sansonetti et al. [20] considered handcrafted user profile features, including screenname length, user description length, number of followers/friends, account's age, average number of daily statuses, and the overall sentiment score for all shared tweets. Giachanou et al. [5] proposed different content-based features, including word embeddings, LIWC categories, personality traits, communication style, sentiment, emotion and readability scores, and word embeddings. Sharma and Sharma [22] combined user features with an average word2vec representation of content for rumour spreader detection. Shrestha and Spezzano [23] considered demographics and behavioral features. Demographic features included age, gender and political orientation. As those features were not explicitly available, the authors inferred them from the content shared by users. Caution should be exercised when considering these types of features as they might introduce biases and reinforce stereotypes [2]. On the other hand, behavioural features were related to the moment of the day and week on which posts were shared. Ghanem et al. [4] divided user timelines into fixed-size chunks (i.e., a user could be represented by multiple chunks), which were represented by a combination of GloVE embeddings and semantic and stylistic features (e.g., emotions, morality, and style). Despite the different nature of the selected features, some works have reported that lexical and sentiment features were the most relevant for classification. In all cases, either traditional or simple neural network models were trained.

Other approaches have focused on network topology. Truong et al. [27] proposed variations of PageRank and HITS centrality metrics for ranking spreaders in different news sharing networks. Rath et al. [18] proposed an inductive representation learning framework based on community health assessment and interpersonal trust to detect spreaders in densely-connected communities.

As exposed, approaches in the literature have focused either on content, hand-crafted, or network topology features, disregarding their complementary nature. For example, as fake content is intentionally written to mislead users, detecting it based only on the shared content might be nontrivial. Fake content might also attempt to distort the truth by adopting similar linguistic styles to authentic content, thus affecting hand-crafted features [24]. Similarly, only considering network topology might lead to incomplete and noisy networks [26].

3 MODELING FAKE NEWS SPREADERS DETECTION

For a given user u_i and their past interactions ($\tau(u_i)$), the shared content (T_{u_i}), and the content propagation trees ($PrT(T_{u_i})$), the goal is to learn a function $F(\tau(u_i), T_{u_i}, PrT(T_{u_i})) \rightarrow \{1, -1\}$, where 1 indicates that u_i is a fake news spreader, and -1 otherwise. Note that while we define the task as a binary classification problem, it could be transformed into a regression problem, in which the goal is to learn a score of how likely the user is to be a spreader. An overview of the architecture is presented in Figure 1.

User representation. To model the heterogeneous nature of social media, the shared content, and the involved participants, user representation is divided into three components: user/tweet features, social interactions, and tweet propagation trees. Features are represented by a vector concatenating characteristics such as personality traits, readability scores, LIWC categories, sentiment and emotions[1]. When appropriate, features were standardized. Extreme values were clipped to the $[-2, 2]$ range, which works as a Gradient Clipping to help speed up training and reduce noise [28].

Previous works [25, 27] have shown the importance of network topology and interactions to identify both fake news and their spreaders. In this sense, user representation includes their social interactions modeled by Graph Convolutional Networks (GCNs). GCNs [10] allow representing nodes based on their characteristics and those of their interactions. Defining three concatenated GCNs allows including interactions from up to 3-hop neighbours, i.e., user community structures are characterized by considering both the direct and explicit user interactions and the neighbourhoods of such interactions. GCNs were implemented based on:

$$GCN(X, \hat{D}, \hat{A}) = f\left(\hat{D}^{-\frac{1}{2}}\hat{A}\hat{D}^{-\frac{1}{2}}XW + b\right) \quad (1)$$

Here, A represents the adjacency user matrix (obtained from the user graph), I is the identity matrix, $\hat{A} = A + I$, $\hat{D}_{ii} = \sum_j \hat{A}_{ij}$, f is the ReLU activation function, X is the user feature matrix, W is a matrix of trainable weights, and b is the trainable bias vector. For each user to classify, X includes their user feature vector and those of the users in their 3-hop neighbourhood. Then, the GCNs outputs a matrix representing all users in X, from which we keep the vector representing the user to classify.

Given the complex nature of information propagation, the definition of discriminant and hand-crafted features based on the content propagation structure might be complex and biased [13]. For example, features representing summary statistics such as centrality, cliques, or connected components might be too general to express the particularities of content cascades [14]. In this sense, to capture the semantics of propagation patterns, we represent each shared tweet based on a propagation tree derived from the triggered replies. Note that while the user graph focuses on users' complete set of interactions, these trees are concerned with the specific responses that each tweet originated, regardless of the users who wrote them. Each tweet is represented by its propagation tree, its pooled BERT [3] embeddings, and the pooled BERT embeddings of the tweets in its propagation tree. Then, the propagation trees (A in Eq. 1) and the embeddings (X in Eq. 1) are fed to a single GCN. The output of the GCN is fed to three parallel dense layers to generate the Query, Key, and Value that will be the input to a multi-headed self-attention mechanism, which aims at better learning how tweets interact with each other. For example, the mechanism is expected to discover relations between tweets that are similar in content or style but are not explicitly connected. Finally, the output of the attention mechanism is passed through a max-pooling layer to obtain the vector representing users' tweets.

Model prediction. Once the user and tweet representations are obtained, they are concatenated with the original user feature vector

[1]More details and the full set of characteristics can be found at https://github.com/tommantonela/umap2022-fake-news-spreaders.

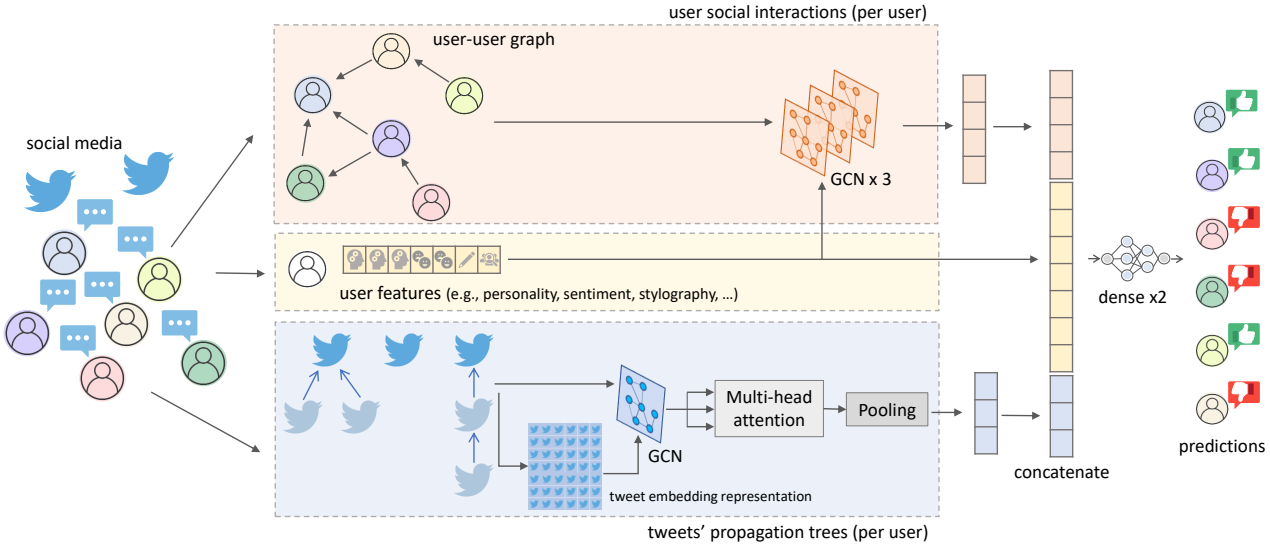

Figure 1: Schematic diagram of the proposed model

and passed through two dense layers to compute the output of the model. The first dense layer has a ReLU activation function, while the last one has a linear one. Due to hardware limitations, it might be unfeasible to make all estimations simultaneously, as the resulting adjacency matrices would require large amounts of memory. For this reason, predictions were made on mini-batches of 10 users.

Model training. The model was trained using a Hinge loss function with class weights (Eq. 2), where 1 is the error margin, y_t represents the true class of users, y_p is the prediction, and w_{y_t} is the balanced class weight. Depending on a margin allows the Hinge loss to select decision boundaries far from the instances of both classes, increasing its robustness [19]. y_t can be 1 or -1, where 1 represents the positive class (i.e., the user is a spreader), and -1 represents the negative class (i.e., the user is not a spreader). When y_t and y_p have the same sign and $|y_p| \geq 1$, the prediction is correct, and thus *loss* is 0. On the other hand, when they have opposite signs, the prediction is incorrect, and *loss* increases. Similarly, even when having the same sign, if $|y_p| < 1$, *loss* would also increase as the margin is not enough to deem the prediction as correct.

$$loss(y_t, y_p) = \frac{\sum w_{y_t} \cdot max(0, 1 - y_t \cdot y_p)}{N} \quad (2)$$

Class weights[2] were defined as $w_{y_t} = |samples|/(2 \cdot |y_t|)$ and aimed to compensate for unbalanced class distribution in training, which would induce a majority class prediction. In this sense, the model would prefer to misclassify instances of the minority class instead of moving predictions from the majority class inside the predefined margin.

4 EXPERIMENTAL SETTINGS

Data collection. Evaluation was based on `FibVid`, a COVID-related misinformation dataset [9][3]. The collection is based on news claims appearing in Politifact and Snopes. From each news claim, the authors extracted keywords that were then searched on Twitter to retrieve the associated content. Tweets were retrieved using the *Faking it!* tool[4]. The complete collection comprised 772 COVID-related news claims and 161,838 relevant tweets shared during 2020. From these claims, 26% and 74% were labeled as authentic and fake content, respectively, according to the Politifact and Snopes label of the associated news claim. Tweets' labels were used to determine whether users were fake news spreaders. To this end, we computed a user score based on the proportion of shared tweets associated with fake news. Then, users were deemed as spreaders if the proportion of shared fake content was higher than a certain threshold. For the purpose of the evaluation, we adopted a conservative definition of spreaders, setting the threshold to 0.5.

Based on the retrieved tweets, we built both a tweet and user graphs. In the tweet graph, nodes represent tweets and edges the reply, quote and retweet relations, following the information flow. In the user graph, nodes represent users, while edges were derived from the tweet graph and the user mentions in tweets. Edges were weighted according to the number of interactions between users (e.g., the number of mentions between two users). Finally, to ensure that the user graph is not disconnected, we kept users (and their tweets) belonging to the largest connected component of the user graph that shared more than one tweet. In summary, we kept 112,433 tweets belonging to 24,430 users. On average, each user shared 4.7 (± 15.78) tweets and established 4 (± 15) user relations[5].

Baselines. The performance of the proposed model was compared to several approaches. First, simple baselines consisting of traditionally used features[6]: **tweet stats** (e.g., the percentage of tweets with url, mentions or hashtags, the percentage of tweets shared during the night or the weekend, and the average tweet length, as

[2]Class weights were computed following the balanced sklearn strategy: https://scikit-learn.org/stable/modules/generated/sklearn.utils.class_weight.compute_class_weight.html

[3]The collection is originally available at https://doi.org/10.5281/zenodo.4441377

[4]*Faking it!*: https://github.com/knife982000/FakingIt

[5]The final retrieved set of tweet IDs (in accordance with Twitter TOS) and the resulting graphs are available at: https://bit.ly/37Topnx.

[6]The different sets of features can be found in the companion repository.

defined by Ghanem et al. [4] and other approaches for fake news detection [12]), **user and tweet stats** (e.g., follower/friend ratio, follower count, description length, whether the user is verified, age of the account, screenname length and digits in screenname, as defined by Sansonetti et al. [20]), **readability** (metrics to estimate the complexity of a text to determine the level of literacy needed to understand it, as defined by Giachanou et al. [5] and Shrestha and Spezzano [23]), **LIWC** (a set of LIWC psychologically-meaningful linguistic categories associated to personal pronouns, personal concerns, cognitive processes, informal processes and perceptual processes, as defined by Giachanou et al. [5]), **personality** (the scores for the Big 5 traits were computed following [15], as in [5]), **node2vec** [6], and a content-based representation of users based on **GloVe** [16] and **BERT** embeddings. We tested these baselines on different classifiers (Decision Tree, Random Forest, K-nn and SVM) and chose the configuration achieving the highest results. We also considered the closely related works of (see Section 2 for more details): i) **Sharma and Sharma [22]** (demographic features were not included), ii) **Sansonetti et al. [20]**, iii) **CheckerOrSpreader** [5], iv) **Shrestha and Spezzano [23]** and v) **FacTweet** [4].

When available, original implementations were used, and parameters were optimized according to the procedures described in the original studies. For each baseline, we selected the configuration achieving the highest results. Tweets were slightly pre-processed by replacing URLs and removing symbols and numbers.

Implementation details. The baselines and the proposed model were implemented in Python, with the support of sklearn, TensorFlow and PyTorch. The optimizer was set to Adam with a learning rate of $1e-3$, $\beta_1 = 0.1$ and $\beta_2 = 0.999$. Hyper-parameter optimization was focused on the size of GCNs, the self-attention mechanism, and the dense layers. The size of the user GCNs was set to 32 (we evaluated 32, 64 and 100), the size of the tweet GCN was set to 100 (we evaluated 100 and 200), the size of the self-attention mechanism was set to 15 (we evaluated 15 and 30), and the size of last two dense layers was set to 70 (we evaluated 70 and 150). Before selecting the weighted Hinge loss function, we also evaluated the performance of unweighted Hinge and Cross-Entropy. For all the layers, we evaluated both ReLU and linear activation functions. The learning process was stopped once no loss changes were observed, reaching convergence after 20 epochs[7]. The model was trained on an Asus Scar 15 with a Ryzen 9 5900HX and an NVidia GeForce RTX 3080. Training and evaluation for each epoch took approximately 1.40 minutes and 30 seconds, respectively.

Evaluation was performed in an offline setting based on a temporal user split, which allows emulating a scenario in which the model is trained with historical data and is used to predict current or future behaviours. Users were sorted considering the date of their first interaction, and then the first 70% users were selected as the training set and the last 30% as the test set. All evaluations were performed over the same data partitions. Performance was evaluated based on binary (focusing on the spreader class) and weighted precision and recall. Given the nature of the task, recall might be more relevant than precision. Finally, we also considered the AUC-ROC score.

[7]More details and implementation can be found at the companion repository.

5 EVALUATION

Table 1 presents the obtained results. For each metric, the best results are shown in bold, and the second-best are underlined. As observed, user and tweet statistics achieved better results than alternatives based on content, implying that tweets' statistics might be more helpful in identifying fake news spreaders than the shared content. In addition, the embedding content representation achieved similar results to the hand-crafted features. This situation seems to confirm that fake content tends to show similar characteristics and style to authentic content [24]. Similarly, the low results observed for the embeddings representation could be related to the topically focused nature of data, in which fake and authentic content will be naturally related, and differences between them might be too subtle for the embeddings to detect. On the other hand, network topology, as characterized by node2vec improved content-based representation, showing that despite being sparse, topology can be a strong indicator of fake news spreaders [27].

State-of-the-art baselines showed similar precision to the simple baselines but lower recall. The best precision/recall balance Was observed for Sansonetti et al. [20], confirming the tendency of tweet and user stats to provide relevant information for spreader detection. On the other hand, the worst results were achieved by FacTweet, which relied on characterizing users by dividing their timelines into small tweet chunks. In this case, it might be possible that chunks do not convey enough information to adequately characterize users. Despite the low precision/recall, it achieved higher AUC-ROC than the simpler baselines, showing that even when mistakenly predicting non-spreaders as spreaders, the confidence on spreaders detection was higher than for the non-spreaders (i.e., they were ranked higher).

Our model achieved the highest results, with average differences of 43% and 162%, and 50% and 54% for binary and weighted precision and recall. Although some baselines achieved similar precision results to our model, they achieved lower recall. In this sense, results confirm the importance of combining the different information sources for spreader detection. Nonetheless, an ablation study is still needed to fully assess their contributions. The differences in binary and weighted precision/recall are higher for the baselines that for our model. This could imply that the non-spreader class has a more significant impact on metrics for the baselines than for our model. In other words, our model was able to predict both classes with similar quality. Our model also outperformed all baselines in terms of the Matthews coefficient [1], with differences up to one order magnitude, confirming the capabilities of our model for correctly classifying instances of both classes.

Finally, we also evaluated performance in a 10-fold stratified cross-validation scenario. For most baselines, results showed similar tendencies than for the temporal split. The only exceptions were the feature sets based on tweet and user stats, CheckerOrSpreader and Shrestha and Spezzano [23]. The tweet and user stats feature sets achieved higher precision than our model, with differences up to a 9%. Nonetheless, they showed lower recall results with differences up to 34%. On the other hand, while still achieving lower results than our model, CheckerOrSpreader and Shrestha and Spezzano [23] largely increased their recall, which might hint their sensitivity

	Binary (Spreader class)		Weighted		AUC-ROC
	Precision	Recall	Precision	Recall	
our model	**0.851**	**0.84**	**0.839**	**0.838**	**0.878**
tweet stats	0.833	0.669	0.769	0.755	0.84
user stats & tweet stats	0.803	0.623	0.737	0.721	0.726
readability	0.573	0.448	0.542	0.534	0.539
LIWC	0.555	0.443	0.526	0.52	0.52
personality	0.537	0.413	0.511	0.504	0.509
node2vec	0.66	0.544	0.62	0.612	0.658
GloVe	0.539	0.436	0.512	0.507	0.51
BERT	0.534	0.411	0.508	0.501	0.506
Sharma and Sharma [22]	0.634	0.239	0.571	0.527	0.713
Sansonetti et al. [20]	0.467	0.642	0.397	0.426	0.688
CheckerOrSpreader [5]	0.809	0.119	0.662	0.521	0.544
Shrestha and Spezzano [23]	0.634	0.239	0.571	0.527	0.713
FacTweet [4]	0.436	0.177	0.524	0.561	0.507

Table 1: Best performance comparison for fake news spreaders detection

to class distribution. With only a few exceptions, differences were statistically significant with an alpha of 0.01, favoring our model.

6 CONCLUSIONS

This work presented a model for identifying fake news spreaders in social media by combining content and user features, the induced propagation trees, and features learned from user interactions. A preliminary evaluation showed the models' potential for accurately detecting fake news spreaders and the importance of combining the different aspects of user representation to achieve a more effective characterization of spreaders.

Several aspects could be tackled in future works. First, the preliminary evaluation only considered a small and sparse data collection with an unbalanced proportion of spreaders. Additional evaluations should be performed over different collections of varying scale and domains. Also, the model could be evaluated for other related tasks such as the detection of rumour, hoaxes or conspiracy spreaders. Second, perform an ablation study to assess the contribution (or effects) of the different components, particularly, the relevance of the hand-crafted features. Third, given the real-time nature of the tackled problem, the model could be adapted to the early detection of spreaders, and dynamic scenarios in which the user and tweet graphs are updated with new elements and interactions. Fourth, the definition of whether a user is a spreader relies on a threshold. In this sense, the problem could be transformed into a regression problem in which we aim to estimate how likely a user is to be a spreader. Finally, the model could be enriched to provide explanations and thus help users understand the effect of their activities and increase their decision awareness. Such explanations could also shed some light on the motivational aspects of fake news spreading [8].

ETHICAL STATEMENT

Research is based on publicly available Twitter data initially collected and tagged by third parties. No user personal information was included in the analysis, and no user identity is being disclosed. As per Twitter TOS, the shared data only includes user and tweet IDs and aggregated content features.

As an end goal, the presented model aims to provide users with tools to identify and quantify unwanted fake news spreaders in their social circles to increase their decision awareness. Nonetheless, the results of this study should not be used to publicly criticize users sharing fake news. Finally, the study can suffer from bias stemming, for example, from the data collection and tagging process. In this sense, biases should be considered before applying any derived result from this study in real-world settings.

REFERENCES

[1] Davide Chicco and Giuseppe Jurman. 2020. The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation. *BMC genomics* 21, 1 (2020), 1–13.

[2] Thomas Davidson, Debasmita Bhattacharya, and Ingmar Weber. 2019. Racial bias in hate speech and abusive language detection datasets. *arXiv preprint arXiv:1905.12516* (2019).

[3] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* (2018).

[4] Bilal Ghanem, Simone Paolo Ponzetto, and Paolo Rosso. 2020. FacTweet: profiling fake news twitter accounts. In *International Conference on Statistical Language and Speech Processing*. Springer, 35–45.

[5] Anastasia Giachanou, Bilal Ghanem, Esteban A Ríssola, Paolo Rosso, Fabio Crestani, and Daniel Oberski. 2022. The impact of psycholinguistic patterns in discriminating between fake news spreaders and fact checkers. *Data & Knowledge Engineering* 138 (2022), 101960.

[6] Aditya Grover and Jure Leskovec. 2016. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*. 855–864.

[7] Bohan Jiang, Mansooreh Karami, Lu Cheng, Tyler Black, and Huan Liu. 2021. Mechanisms and Attributes of Echo Chambers in Social Media. *arXiv preprint arXiv:2106.05401* (2021).

[8] Mansooreh Karami, Tahora H Nazer, and Huan Liu. 2021. Profiling Fake News Spreaders on Social Media through Psychological and Motivational Factors. In *Proceedings of the 32nd ACM Conference on Hypertext and Social Media*. 225–230.

[9] Jisu Kim, Jihwan Aum, SangEun Lee, Yeonju Jang, Eunil Park, and Daejin Choi. 2021. FibVID: Comprehensive fake news diffusion dataset during the COVID-19 period. *Telematics and Informatics* 64 (2021), 101688. https://doi.org/10.1016/j.tele.2021.101688

[10] Thomas N. Kipf and Max Welling. 2017. Semi-Supervised Classification with Graph Convolutional Networks. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net.

[11] Yang Liu and Yi-Fang Wu. 2018. Early detection of fake news on social media through propagation path classification with recurrent and convolutional networks. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 32.

[12] Jing Ma, Wei Gao, Zhongyu Wei, Yueming Lu, and Kam-Fai Wong. 2015. Detect rumors using time series of social context information on microblogging websites. In *Proceedings of the 24th ACM international on conference on information and knowledge management*. 1751–1754.

[13] Jing Ma, Wei Gao, and Kam-Fai Wong. 2017. Detect Rumors in Microblog Posts Using Propagation Structure via Kernel Learning. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, Vancouver, Canada, 708–717. https://doi.org/10.18653/v1/P17-1066

[14] Federico Monti, Fabrizio Frasca, Davide Eynard, Damon Mannion, and Michael M Bronstein. 2019. Fake news detection on social media using geometric deep learning. *arXiv preprint arXiv:1902.06673* (2019).

[15] Yair Neuman and Yochai Cohen. 2014. A vectorial semantics approach to personality assessment. *Scientific reports* 4, 1 (2014), 1–6.

[16] Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*. 1532–1543.

[17] Francesco Pierri, Carlo Piccardi, and Stefano Ceri. 2020. A multi-layer approach to disinformation detection in US and Italian news spreading on Twitter. *EPJ Data Science* 9, 1 (2020), 35.

[18] Bhavtosh Rath, Aadesh Salecha, and Jaideep Srivastava. 2020. Detecting fake news spreaders in social networks using inductive representation learning. In *2020 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*. IEEE, 182–189.

[19] Lorenzo Rosasco, Ernesto De Vito, Andrea Caponnetto, Michele Piana, and Alessandro Verri. 2004. Are loss functions all the same? *Neural computation* 16, 5 (2004), 1063–1076.

[20] Giuseppe Sansonetti, Fabio Gasparetti, Giuseppe D'aniello, and Alessandro Micarelli. 2020. Unreliable users detection in social media: Deep learning techniques for automatic detection. *IEEE Access* 8 (2020), 213154–213167.

[21] Karishma Sharma, Feng Qian, He Jiang, Natali Ruchansky, Ming Zhang, and Yan Liu. 2019. Combating fake news: A survey on identification and mitigation techniques. *ACM Transactions on Intelligent Systems and Technology (TIST)* 10, 3 (2019), 1–42.

[22] Shakshi Sharma and Rajesh Sharma. 2021. Identifying possible rumor spreaders on twitter: A weak supervised learning approach. In *2021 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 1–8.

[23] Anu Shrestha and Francesca Spezzano. 2021. Characterizing and predicting fake news spreaders in social networks. *International Journal of Data Science and Analytics* (2021), 1–14.

[24] Kai Shu, Amy Sliva, Suhang Wang, Jiliang Tang, and Huan Liu. 2017. Fake news detection on social media: A data mining perspective. *ACM SIGKDD explorations newsletter* 19, 1 (2017), 22–36.

[25] Kai Shu, Suhang Wang, and Huan Liu. 2019. Beyond news contents: The role of social context for fake news detection. In *Proceedings of the twelfth ACM international conference on web search and data mining*. 312–320.

[26] Jiliang Tang, Yi Chang, and Huan Liu. 2014. Mining social media with social theories: a survey. *ACM Sigkdd Explorations Newsletter* 15, 2 (2014), 20–29.

[27] Bao Tran Truong, Oliver Melbourne Allen, and Filippo Menczer. 2022. News Sharing Networks Expose Information Polluters on Social Media. *arXiv preprint arXiv:2202.00094* (2022).

[28] Jingzhao Zhang, Tianxing He, Suvrit Sra, and Ali Jadbabaie. 2020. Why Gradient Clipping Accelerates Training: A Theoretical Justification for Adaptivity. In *International Conference on Learning Representations*. https://openreview.net/forum?id=BJgnXpVYwS