Annotated Dataset Creation for Fake News Detection on Online Social Networks

F. Batool, G. Lo Re, and M. Morana

Proceedings of the 1st International Workshop on Artificial Intelligence for Cybersecurity (AI-SEC-2025)

Publisher: Springer

Annotated Dataset Creation for Fake News Detection on Online Social Networks

F. Batool[1], G. Lo Re[2], and M. Morana[2]

Abstract Social media are powerful platforms for sharing news and opinions, but their use may also facilitate the rapid spread of false information. Supervised algorithms for fake news detection, ranging from traditional machine learning to deep learning methods, rely heavily on the quality of training data. This work proposes a semi-automatic dataset creation technique to support the validation of fake news detection algorithms. The system aims to generate annotated datasets containing tweets with detailed information (e.g., text, user data, and relationships between users and data) and to determine the ground truth by assigning truth values to each tweet in the dataset. The tests conducted showed the effectiveness of the annotation process as well as the limitations and strengths of the approaches considered.

1 Introduction

The rapid spread of Online Social Networks, platforms through which users have the possibility to share information on various topics, has favored the birth of new communication models. All social networks, X (formerly Twitter) in particular, constitute a powerful means through which users can instantly publish their thoughts and opinions [2]; on the other hand, this immediate access and the freedom of sharing that derives from it have over the years also led to an increase in disinformation

Farwa Batool
Scuola IMT Alti Studi Lucca, Lucca, Italy
e-mail: farwa.batool@imtlucca.it

Giuseppe Lo Re
Università degli Studi di Palermo, Palermo, Italy
e-mail: giuseppe.lore@unipa.it

Marco Morana
Università degli Studi di Palermo, Palermo, Italy
e-mail: marco.morana@unipa.it

phenomena based on the dissemination of false information, known as *fake news*. Fake news, considered a threat to democracy, negatively influences politics, economics, stock markets, journalism and decreases users' trust in institutions and real news. This is why fake news detection systems have been developed with the aim of distinguishing real information from false information. In the literature, there are several methods that allow to detect fake news in circulation including: traditional machine learning methods [5], deep learning methods [11], knowledge-based detection methods [12], propagation-based detection methods [8], source-based detection methods [18], which consider some linguistic, temporal, user information-based and interaction-based characteristics. These models are highly data-dependent, requiring extensive and diverse datasets to effectively capture the nuances and complexities of their respective domains. This data intensity is particularly crucial to achieve robust predictive accuracy and generalizability. The type of information collected from the datasets depends on the purpose of the application and may vary significantly between datasets. For example, some datasets focus on gossip facts while others include political statements. Furthermore, they also differ according to the type of content that is included (e.g. user responses, source of the statement, etc.), and the labels that are provided. Datasets are often used as training or validation models. This means that the quantity and quality of data in the dataset and the number of labels influence the classification algorithms for fake news detection. These considerations on the one hand highlight the significant influence of datasets on fake news detection algorithms, but on the other hand shows the limited quality of the information contained, signifying the need of a system for the mass collection of tweets. In this work, a system for collection of most knowledge about tweets and users is presented. Then the collected dataset is utilized to test unsupervised algorithms for fake news detection, which are known in literature, to analyze the validity of the proposed system.

The remainder of this paper is organized as follows: the related work is described in Section 2. Section 3 presents the proposed methodology. The fake news detection techniques and the related features are outlined in Section 4. Section 5 discusses the experimental results. And lastly the work is concluded and future indications are provided in Section 6.

2 Related Work

Over the past decade, approaches for detecting rumors and fake news predominantly based exclusively on traditional machine learning techniques. These methods include Decision Tree [1], Random Forest [7], Linear Regression, Bayesian algorithm [9], and SVM (Support Vector Machine) [7]. Recently, advances in deep learning have led to the use of more sophisticated methods, such as Convolutional Neural Networks (CNN) [3], Recurrent Neural Networks (RNN), Long Short Term Memory (LSTM) [10], Attention Mechanism [20], and Joint Learning [16]. Despite their predictive capabilities, these models have notable limitations in capturing the lin-

guistic intricacies of the news content. These limitations emphasize the need of development of new and advanced solutions.

A major challenge in this field is the quality of available data, which varies based on the dataset's objective, labels, and content. Some datasets rely on automatic labeling rather than manual verification, affecting algorithm performance. To address this, researchers use unsupervised learning [13]. For example, the study by Hosseinimotlagh [6], used tensor-based modeling and an ensemble method was also introduced to combine results from different tensor techniques, improving detection reliability and precision. Wang et al. [21] proposed an Expectation-Maximization (EM) algorithm to estimate source reliability and evaluate observation correctness in truth-finding. EM is an iterative algorithm used to estimate model parameters when data is missing. It is useful for determining the truthfulness of claims made by unreliable users, like those on X. EM starts by assuming initial reliability for each user, then calculates the probability of each claim being true. Based on these probabilities, it updates the users' reliability scores. The algorithm alternates between the E-step (estimating truthfulness) and M-step (updating reliability), refining estimates until convergence, where the truth of claims and user reliability are optimized. Enhanced methods like the Constrained EM (CEM) algorithm [14] improved accuracy by incorporating multi-modal data and heuristic penalties. This model introduces a constraint based on the number of independent features supporting a claim. The constraint ensures that the probability of a claim being true is higher when supported by multiple independent sources providing evidence. Shao et al. [13] introduced unsupervised approaches like EM-Multi, CEM-Multi, and PEM-MultiF, further boosting estimation accuracy by using multi-modal data and penalties. Building on their work, our study evaluates these models while incorporating detailed semantic processing to enhance their performance in fake news detection.

3 Methodology

The problems related to the evaluation of fake news detection methods and the need for annotated datasets, highlighted earlier have led to the experimentation of a system for the mass collection of tweets, with semi-automatic annotation. The proposed system includes a phase for collecting tweets, with all the related information, and also consists of a phase that allows for the attribution of a truth value to the collected tweets. The goal of the system is to create a dataset containing tweets and all the necessary information associated with them, with the aim of helping the research community by providing a tool to create annotated datasets and help in the definition of new techniques. In order for the performance of the algorithms to be evaluated, a procedure is needed that allows determining a Ground Truth, and therefore establishing whether each tweet contained in the dataset is fake news or real news.

The system is characterized by a modular architecture, and therefore includes a sequence of phases as outlined in Figure 1.

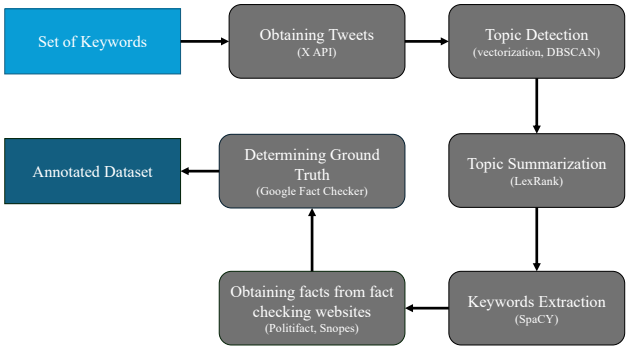

Fig. 1 System Overview.

The input to the system is a set of keywords related to certain topics, including politics, news, health and gossip. The X platform provides APIs through which it is possible to obtain certain tweets in a simple and unique way. The API provides access to a set of methods that allow you to obtain a large amount of information about tweets. The response is made up of all the related attributes, including: tweet text, tweet id, creation date, URL or mentions in the tweet, number of likes, number of retweets, media attached to the tweet, hashtags and much more. There is also information about the user who published the tweet: user name, user id, account creation date, number of followers, number of friends, user timeline etc. Tweets can be obtained either from their identifier, or from a word present in the tweet. In this case, tweets were obtained from a set of keywords, both because tweet IDs are not available, and because in this way tweets related to a specific topic can be obtained. So, for each keyword present in the input set, a request must be made to the X platform via the relevant API method. For data collection, the python library *tweepy* was used, which allows access to the X API in a simple way.

The obtained collection constitutes the input of the *topic detection* phase, which aims to group tweets by clustering semantically similar ones, representing the same claim. Clustering requires vectorizing tweets into numerical form, which is done using a binary matrix where rows represent tweets and columns represent words. The presence of a word in a tweet is marked with 1 using the CountVectorizer. Then, the DBSCAN (Density-Based Spatial Clustering of Applications with Noise) [4] algorithm is used. DBSCAN connects points based on density, identifying regions with similar densities and isolating outliers. It does not require a predefined number of clusters and can detect clusters of arbitrary shapes. The algorithm only needs two

parameters: a radius R and a minimum number N of points, which are determined by evaluating clustering performance through the Silhouette coefficient [19]. The highest Silhouette coefficient, which ranges from -1 to 1, is used to select the best parameters, with higher values indicating better clustering.

$$S_i = \frac{Dmin_i^{out} - Davg_i^{in}}{max(Dmin_i^{out}, Davg_i^{in})}$$

Here $Davg_i^{in}$ indicates the average distance of a point of a cluster from the points of its own cluster, while $Dmin_i^{out}$ represents the minimum average distance of a point from other clusters different from its own. The clustering thus provides the output containing; CLAIM, which represents the cluster identifier; NUM_OF_TWEETS, which represents the number of tweets that make up the cluster; LIST_OF_ID, which includes a list of tweet IDs assigned to the clusters and LIST_OF_TEXT, which includes a list of the texts of the tweets assigned to the cluster. This output is forwarded to the next component for the topic summarization.

Once the tweets have been grouped into clusters, the next step is to determine the claim that is represented by each individual cluster obtained. For this purpose, the **LexRank** approach is used which is based on the fact that a sentence similar to many other sentences in the text has a high probability of being important. The more frequent the sentence, the higher the rank, which constitutes the priority of being included in the summarized text. This approach is suitable for the purpose, since the most frequent sentences in the tweets are most significant for the fact represented. LexRank summarizes the text and provides following attributes; CLAIM representing the claim identifier; LIST_OF_ID having a list of tweet IDs assigned to the clusters; LIST_OF_TEXT, which includes a list of the texts assigned to the clusters; and LEX_RANK_SUMMARY_TEXT includes the the summary obtained from the LexRank model.

The next phase regards obtaining *facts*. To this aim, Google Fact Check Tools is used to perform queries by one or more keywords, through which you can obtain facts annotated by the most popular fact-checking sites. Since the search must be performed through keywords, a process is needed that, starting from the text of the claim determined by the summarization, allows you to obtain a set of keywords. When processing input, requiring all words to be present in the facts reduces the likelihood of finding relevant information, as each fact must include all input words. This phase aims to remove non-essential words from the claim texts while maintaining their original meaning. Keywords extraction significant from the claims was performed by means of spaCy library which returns INITIAL_KEYWORDS, which are the tokens resulting from the grammatical filtering and DEFINITIVE_KEYWORDS constituting the list of words obtained from the second and final filtering. These two attributes along with the CLAIM and LEX_RANK_SUMMARY_TEXT from the previous step are forwarded to the next step. Once a set of significant keywords has been obtained for each claim, a request is made to the Google Fact Check system specifying the set of keywords and Politifact.com and Snopes.com as the fact-checking sites. The response obtained from the system is a set of annotated facts.

For each fact-checking site considered, it is necessary to transform the ratings of the relative site into the two classic truth values: *True* or *False*. To assign a single truth value to a claim in the dataset, the following method is used: if only one fact is retrieved, its truth value is assigned to the claim. When multiple facts are obtained, the truth value of the fact most similar to the claim is used. Texts of all facts and the claim are vectorized using sklearn's CountVectorizer, converting them into binary vectors indicating word presence. The claim vector is then compared to fact vectors using Euclidean distance, and the truth value of the closest fact is assigned to the claim. Finally, the system-generated annotation is manually verified to ensure the accuracy of assigned truth values. If the fact corresponding to the claim aligns in terms of information content and meaning, the truth value associated with the fact is assigned to the claim. Conversely, the opposite truth value is assigned if the fact conveys the opposite meaning or denies the claim. In cases where the statements of the fact are unrelated or neutral (not attributable to either True or False), the claim in question is assigned the attribute *Undetermined*. Tweets with Undetermined or Unknown values are discarded, leaving only True and False tweets to evaluate the fake news detection algorithms. Additionally, a dataset of users' followers was collected to enable analysis of user relationships. The tweepy Python library's *api.friends()* method was used to gather the 20 most recent followers for each user.

4 Feature Analysis and Detection Techniques

To evaluate the proposed methodology, different established methods from the literature were selected namely EM [21], EM-MultiF [13], CEM [14] and PEM [13]. EM iteratively estimates user reliability and claim truthfulness. Starting with initial assumptions it calculates claim probabilities, and refines estimates through alternating E-steps and M-steps until convergence, optimizing both parameters. EM-MultiF incorporates additional features beyond source reliability, like whether the tweet contains an image or a URL. It aims to leverage these features to improve truth discovery. While PEM penalizes the probability of a claim being false if it has supporting features (like images or URLs). The idea is that claims with more supporting evidence are more likely to be true. Lastly, based on the number of independent features supporting a claim, CEM ensures that the probability of the claim being true is higher when supported by multiple independent sources providing evidence.

These methods exploit information about the associations between "subjects and claims", "features and claims", and the relationship between the claim author and his ancestor in the social graph. These features are represented using three matrices: **SC**, **FC**, and **D** respectively.

Association between Author and Claims: The SC matrix includes the authors of tweets, on the rows, and claims as column. The information needed to execute the algorithm is the informative content of the tweet, and therefore the semantic meaning. For example, considering the tweets "*Today is Brad Pitt's 57th birthday*" and "*57 years ago Brad Pitt was born*". Although the two sentences just reported are

different from a lexical point of view, the informative content of both is the same. Clustering is used to achieve this aim.

Associations between Features and Claims: The FC matrix contains features about i) presence of images, ii) presence of URLs, and iii) claim reported by at least two independent sources. Such a FC matrix represents the relationship between claims and features, with rows for features (three total) and columns for claims. If claim C_j contains feature F_k, the cell is set to 1; otherwise, it is 0. For example, if the claim in column 6 includes an image, $F_0C_5 = 1$. The same applies for URLs. For the third feature, independent sources reporting similar claims are marked with 1. The matrix uses binary indicators without checking the actual content of images or URLs.

Associations between Claim and Author's Ancestors: The D matrix reports the authors on the rows and the claims on the columns. In this case, the information is the relationship between the author's ancestor and the claim. The cell of the matrix D corresponding to the source S_i and the claim C_j will have the value 1 if there is at least one ancestor of S_i who has report the claim C_j.

5 Results and Discussion

The dataset collected using the proposed method consisted of 106,901 tweets and 4301 clusters representing claims. Among these, 57,413 tweets were classified as noise points and not associated with any claim, leaving the remaining tweets for further analysis. Of the 4,301 claims, the Google Fact Check system annotated 4,245 as *Unknown*, 9 as *Undetermined*, 6 as *True* and 41 as *False*. Therefore, 47 claims containing 667 tweets and 653 users were used for the analysis.

For the comparison of proposed method, this study employs two publicly available datasets. The first dataset consists of news articles from Politifact and Gossipcop[1], containing tweet IDs, URLs, and titles. The second dataset was collected from a Dropbox repository [17]. The former exhibits a significant class imbalance, while the later is balanced as outlined in Table 1. As mentioned above the fake news algorithms utilize features described in Section 4. The dataset generated by proposed method already consists of these features. For the public datasets, to retrieve detailed tweet data, the X API is used to extract tweet content, metadata, and user information such as ID, bio, and follower counts. For constructing the SC (source-claim), FC (feature-content), and D (dependency) matrices, the API helps map tweet-author relationships and build an influence graph. Retweets and follower-following data are combined to identify ancestor-descendant relationships between users. The *api.show_friendship()* function is used to verify these relationships, forming the basis for constructing the matrices. The dimensions of feature matrices and their values of each dataset are shown in Table 2, while Figure 2 shows the count of presence (1) and absence (0) of images and URLs in each dataset.

[1] https://github.com/KaiDMML/FakeNewsNet

Table 1 Information about the datasets adopted.

Dataset	Imbalanced	Balanced	Proposed
Users	1554	495	653
Tweets	4285	682	667
True Claims	168	101	6
False Claims	638	135	41
Total Claims	806	236	47
Features	3	3	

Table 2 Feature Matrices' dimensions and values.

	Imbalanced Dataset			Balanced Dataset		
Matrix	**Dim**	**Number of 1s**	**Number of 0s**	**Dim**	**Number of 1s**	**Number of 0s**
SC	1554 × 806	3246	1249278	495 × 236	591	116229
FC	3 × 806	1235	1183	3 × 236	303	405
D	1554×806	0	1252524	495×236	0	116820

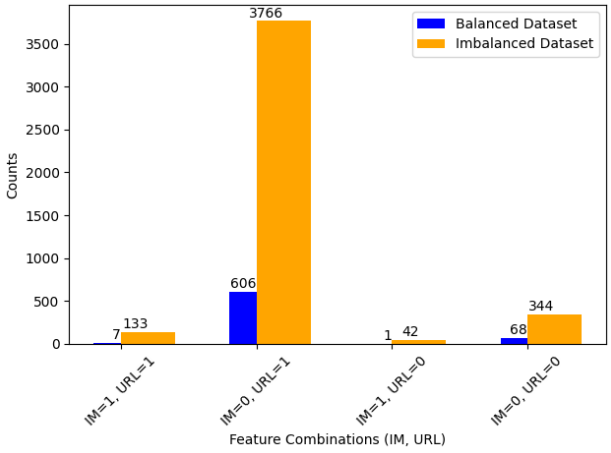

Fig. 2 Comparison of Images and URL Combinations in Datasets.

The four algorithms introduced in Section 4 were extensively tested to determine their optimal working parameters. In case of EM and EM-MultiF, *fraction* refers to a parameter that determines the subset of the data to be considered in each iteration of the algorithm. In this case, the fractions for EM and EM-MultiF were analyzed across a range of values from 0.2 to 0.9, with the optimal value found to be 0.9 for both algorithms. For PEM the parameter α controls the penalty applied during the estimation process, influencing how much emphasis is given to the reliability of sources. The optimal value of α found for PEM was 0.8 from the values ranging from 0.1 to 0.8. For CEM the parameter λ is related to the constraints imposed on the estimation process, affecting how much weight is given to these constraints. The optimal value of λ found for CEM was 0.8.

Table 3 Performance of models on three datasets.

Dataset	Algorithm	Accuracy	Precision	Recall	F1-score
Balanced (train)	EM	0.62288	0.61057	0.94074	0.74052
	EM-MultiF	0.66949	0.64467	0.94074	0.76506
	PEM	0.64406	0.65644	0.79259	0.71812
	CEM	0.65670	0.63910	0.91850	0.75370
Imbalanced (test)	EM	**0.799**	0.83147	**0.93573**	**0.88053**
	EM-MultiF	0.79528	0.83263	0.92789	0.87768
	PEM	0.77295	0.83604	0.88714	0.86083
	CEM	0.74193	**0.84789**	0.82131	0.83439
Proposed (test)	EM	**0.78723**	**0.87804**	**0.87804**	**0.87804**
	EM-MultiF	**0.78723**	**0.87804**	**0.87804**	**0.87804**
	PEM	0.76595	0.87500	0.85365	0.86419
	CEM	0.74468	0.87179	0.82926	0.85000

The models are applied to the two public datasets, as well as to the one generated by the proposed system. The balanced dataset was used for training the models and other two datasets i.e., imbalanced and proposed, for testing, reflecting the real-world imbalance typically observed in fake news scenarios. As illustrated in Table 3, all models show good performance on the testing datasets. However, the performance is sightly lower on the imbalanced dataset, which could be due to the class imbalance. While the proposed dataset, despite sharing a similar imbalance, includes features and patterns that enhance model performance, maintaining a clear balance between the metrics. A different view of the comparison between the results achieved on the testing datasets is proposed in Figure 3. It is worth highlighting also the effectiveness of the proposed data collection method, where all necessary data is readily available, eliminating the need to construct separate feature matrices for evaluating each model and to use Twitter APIs, which can be time-consuming and subject to rate limits. The main advantage of the system-generated dataset over traditional datasets is that it simplifies the process of creating annotated datasets for evaluating and testing fake news detection algorithms.

Additionally, this study offers valuable insights into the performance of fake news detection models based on the obtained results. When comparing EM (user reliability-based) with EM-MultiF (which includes images and URLs), there was little performance improvement from adding content features, suggesting that multimedia presence alone is not a strong indicator of truth. EM and EM-MultiF outperformed PEM and CEM, possibly due to the limitation of later in handling complexities in user-claim relationships. It is important to note that the user relationship matrix (D) lacked data, limiting the algorithms' ability to leverage user influence, thus affecting overall performance.

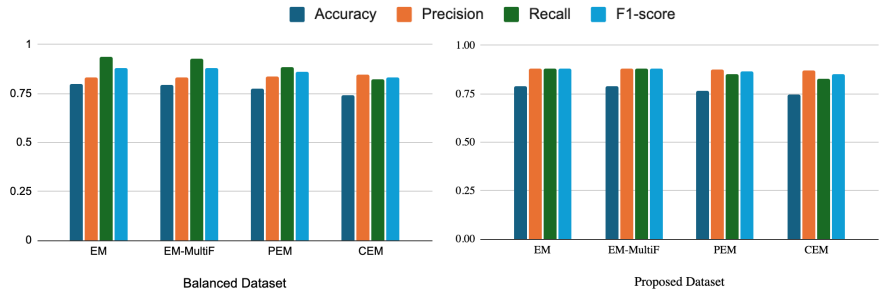

Fig. 3 Performance of models on public dataset and dataset obtained by proposed model.

6 Conclusion

The primary goal of this research was to create a comprehensive dataset for fake news detection algorithms to help them learn the nuanced patterns of data more effectively. The proposed approach provides a more accurate attribution of truth values which is a key factor for a reliable model evaluation.

The dataset was used to evaluate the performance of four unsupervised models: EM, EM MultiF, CEM, and PEM. For comparison, two publicly available datasets were also analyzed. The results demonstrated that the imbalanced dataset and the dataset generated by the proposed method yielded comparable outcomes, with the proposed dataset offering more consistent and balanced metrics. These findings confirm that the proposed method is not only effective for evaluating fake news detection models but also advantageous in producing reliable and balanced results. By streamlining the data collection and preprocessing process, this method enables researchers to directly utilize a complete and well-structured dataset, saving time and resources while enhancing the accuracy of evaluations.

Future work could focus on refining user relationship modeling by incorporating more engagement types (e.g., likes) and exploring alternative data sources. Enhancing data retrieval to include complete user-pair information could improve results.

Acknowledgments

This work was partially supported by the AMELIS project, within the project FAIR (PE0000013), and by the ADELE project, within the project SERICS (PE00000014), both under the MUR National Recovery and Resilience Plan funded by the European Union - NextGenerationEU

References

1. Castillo, C., Mendoza, M., Poblete, B. (2011, March). Information credibility on twitter. In Proceedings of the 20th international conference on World wide web (pp. 675-684).
2. Concone, F., Re, G. L., Morana, M., Ruocco, C. (2019, February). Twitter Spam Account Detection by Effective Labeling. In ITASEC.
3. De Sarkar, S., Yang, F., Mukherjee, A. (2018, August). Attending sentences to detect satirical fake news. In Proceedings of the 27th international conference on computational linguistics (pp. 3371-3380).
4. Ester, M., Kriegel, H. P., Sander, J., Xu, X. (1996, August). A density-based algorithm for discovering clusters in large spatial databases with noise. In kdd (Vol. 96, No. 34, pp. 226-231).
5. Gravanis, G., Vakali, A., Diamantaras, K., Karadais, P. (2019). Behind the cues: A benchmarking study for fake news detection. Expert Systems with Applications, 128, 201-213
6. Hosseinimotlagh, S., Papalexakis, E. E. (2018, February). Unsupervised content-based identification of fake news articles with tensor decomposition ensembles. In Proceedings of the Workshop on Misinformation and Misbehavior Mining on the Web (MIS2).
7. Kwon, S., Cha, M., Jung, K., Chen, W., Wang, Y. (2013, December). Prominent features of rumor propagation in online social media. In 2013 IEEE 13th international conference on data mining (pp. 1103-1108). IEEE.
8. Meyers, M., Weiss, G., Spanakis, G. (2020). Fake news detection on twitter using propagation structures. In Disinformation in Open Online Media: Second Multidisciplinary International Symposium, MISDOOM 2020, Leiden, The Netherlands, October 26–27, 2020, Proceedings 2 (pp. 138-158). Springer International Publishing.
9. Mihalcea, R., Strapparava, C. (2009, August). The lie detector: Explorations in the automatic recognition of deceptive language. In Proceedings of the ACL-IJCNLP 2009 conference short papers (pp. 309-312).
10. Popat, K., Mukherjee, S., Yates, A., Weikum, G. (2018). Declare: Debunking fake news and false claims using evidence-aware deep learning. arXiv preprint arXiv:1809.06416.
11. Raza, S., Ding, C. (2022). Fake news detection based on news content and social contexts: a transformer-based approach. International Journal of Data Science and Analytics, 13(4), 335-362.
12. Seddari, N., Derhab, A., Belaoued, M., Halboob, W., Al-Muhtadi, J., Bouras, A. (2022). A hybrid linguistic and knowledge-based analysis approach for fake news detection on social media. IEEE Access, 10, 62097-62109.
13. Shao, H., Sun, D., Yao, S., Su, L., Wang, Z., Liu, D., ... Abdelzaher, T. (2020). Truth discovery with multi-modal data in social sensing. IEEE Transactions on Computers, 70(9), 1325-1337.
14. Shao, H., Yao, S., Zhao, Y., Zhang, C., Han, J., Kaplan, L., ... Abdelzaher, T. (2018, April). A constrained maximum likelihood estimator for unguided social sensing. In IEEE INFOCOM 2018-IEEE Conference on Computer Communications (pp. 2429-2437). IEEE.
15. Shu, K., Mahudeswaran, D., Wang, S., Lee, D., Liu, H. (2020). Fakenewsnet: A data repository with news content, social context, and spatiotemporal information for studying fake news on social media. Big data, 8(3), 171-188.
16. Shu, K., Wang, S., Liu, H. (2019, January). 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 (pp. 312-320).
17. Shu, K. (2020). ICWSM2020-FakeNewsNet. Dropbox. Retrieved June 21, 2024, from https://www.dropbox.com/scl/fo/64eyk438rt55ozfel3y3h/ALuo-4XcDVCw3Ke3m_vqULI?rlkey=l4ibd9y6zrzndslfxeesrqrx8&e=1&dl=0
18. Sitaula, N., Mohan, C. K., Grygiel, J., Zhou, X., Zafarani, R. (2020). Credibility-based fake news detection. Disinformation, misinformation, and fake news in social media: Emerging research challenges and Opportunities, 163-182.
19. Steenari, J., Nurminen, J. K. (2023). Geospatial DBSCAN Hyperparameter Optimization with a Novel Genetic Algorithm Method.

20. Vaswani, A. (2017). Attention is all you need. Advances in Neural Information Processing Systems.
21. Wang, D., Kaplan, L., Le, H., Abdelzaher, T. (2012, April). On truth discovery in social sensing: A maximum likelihood estimation approach. In Proceedings of the 11th international conference on Information Processing in Sensor Networks (pp. 233-244).