Automated Detection of Sockpuppet Accounts in Wikipedia

Mostofa Najmus Sakib
Department of Computer Science
Boise State University
Boise, ID – USA
mostofanajmussak@u.boisestate.edu

Francesca Spezzano
Department of Computer Science
Boise State University
Boise, ID – USA
francescaspezzano@boisestate.edu

***Abstract*—This paper addresses the problem of identifying sockpuppet accounts on Wikipedia. We formulate the problem as a binary classification task and propose a set of features based on user activity and the semantics of their contributions to separate sockpuppets from benign users. We tested our system on a dataset we built (and released to the research community) containing 17K accounts validated as sockpuppets. Experimental results show that our approach achieves an F1-score of 0.82 and outperforms other systems proposed in the literature. Moreover, our proposed approach is able to achieve an F1-score of 0.73 at detecting sockpuppet accounts by just considering their first edit.**

***Index Terms*—Sockpuppetry, malicious activity, early prediction.**

I. Introduction

Wikipedia is a free Internet-based encyclopedia that is built and maintained via the open-source collaboration of a community of volunteers. Given its open-editing format, Wikipedia is highly vulnerable to malicious activity, including vandalism, spam, undisclosed paid editing, etc. [1]–[3]. Malicious users often use *sockpuppet accounts* to circumvent a block or a ban imposed by Wikipedia administrators on the person's original account. A sockpuppet is an "online identity used for the purpose of deception." [1] Usually, several sockpuppet accounts are controlled by a unique individual (or entity) called *puppetmaster*. Currently, suspected sockpuppet accounts are manually verified by Wikipedia administrators, which makes the process slow and inefficient.

In this paper, we address the problem of automatically detecting sockpuppet accounts on Wikipedia. We address the problem as a binary classification task and propose a set of new features to capture suspicious behavior that considers user activity and analyzes the contributed content. Specifically, content-based features have never been considered before and constitute the novelty of our work. We tested our approach on a dataset we collected containing 17K accounts validated by Wikipedia as sockpuppets. Experimental results show that our proposed approach is able to detect sockpuppet accounts with an F1-score of 0.82 (vs. a score of 0.77 achieved by the best competitor) by considering the user's first 20 edits and 0.73 by just considering the first edit (vs. a score of 0.68 achieved by the best competitor).

[1]https://en.wikipedia.org/wiki/Sockpuppet_(Internet)

IEEE/ACM ASONAM 2022, November 10-13, 2022
978-1-6654-5661-6/22/$31.00 © 2022 IEEE

II. Related Work

In the literature, several works have analyzed and detected sockpuppet accounts in online social networks and discussion forums [4]–[7]. Specifically to Wikipedia, Solorio et al. [8], [9] have addressed the problem of detecting whether or not two accounts are maintained by the same user by using text authorship identification features. They have extensively focused on the comments and edits on talk pages and considered features such as punctuation marks, use of emoticons, use of capitalization, and parts-of-speech to characterize the user writing style. Yamak et al. [10] have focused on classifying sockpuppet vs. genuine accounts using non-verbal behavior and considering editing patterns. Their analysis included Wikipedia-specific features, i.e., the number of edits, frequency of revert after each contribution in the same article, the time between registration and edits, etc. In continuation of this work, the same authors also addressed the grouping of detected sockpuppet accounts created by the same individual [11]. They developed relational graphs and combined them with community detection algorithms and account-focused attributes to catch sockpuppet groups. Tsikerdekis et al. [12] performed a Wikipedia-focused analysis to detect identity deception. Their experiment reflected on non-verbal behavior, including the number of total revisions on different Wikipedia pages (article, article discussion, user page, user discussion page), the average number of bytes added or removed, etc. Zheng [13] carried out a sockpuppet analysis by considering sockpuppets in both the same forum and cross-platform. They compared keyword-based similarity profiles for posts A1 and A2 in two different forums and evaluated the probability of being a sockpuppet pair. They assumed puppet masters tend to follow similar writing patterns even if they use multiple accounts.

Like Wikipedia, multiple account generation is prevalent in various online social media. For instance, Maitry et al. [14] analyzed sockpuppet accounts on Twitter and Swati Adhikari [15] performed a similar sockpuppet detection on Reddit data. Maitry et al. [14] emphasized real-time tweets and profile-focused features to identify accounts under the

same user in a quick time, whereas Swati [15] included Reddit users, their posts, subreddit, and their karma scores. However, both works are platform-dependent and cannot be generalized on other cross-platforms. A multiple online community-based analysis was conducted by Kumar et al. [7]. The authors analyzed sockpuppetry behaviors across nine different communities. They pointed out that sockpuppets follow unique linguistic traits (more singular first-person) and have more chances of posting on the same discussion in a short timeframe. In addition, they claimed sockpuppet pairs follow similar writing styles and patterns compared to regular contributors. Joshi et al. [3] investigated the use of sockpuppet accounts to perform undisclosed paid edits on Wikipedia. They found that sockpuppet accounts associated with undisclosed paid editors only work on a limited number of Wikipedia titles they are interested in promoting. In contrast, genuine users edit more pages related to their field of expertise. This shows that sockpuppets accounts' behavior in Wikipedia is different from sockpuppetry in online discussion communities, where sockpuppets' main goal is to interact with each other to deceive other users [7].

III. Dataset

This section describes how we built a dataset containing both sockpuppet and benign user accounts.[2] For collecting the data of sockpuppet accounts, we used the publicly available Wikipedia API.[3] We started by collecting all the usernames of the users grouped under the "Suspected Wikipedia sockpuppets" category [4] and all its subcategories (up to May 28th, 2022). We collected 17,180 accounts validated by Wikipedia as sockpuppets. Regarding the benign user accounts, we relayed on the set collected by Kumar et al. [1] which includes 16,496 Wikipedia accounts validated as benign users.

Next, for each of the considered accounts (sockpuppets and benign users), we retrieved their first 20 edits. We considered 20 edits for each user as our goal is to build an automated detection system that is able to identify sockpuppet accounts as early as possible. For each edit, we retrieved the following information: the page id, the parent page id, the page namespace (article, article discussion, user page, etc.), the page title, the edit timestamp, the text of the user contribution, and the size of the user contribution.

IV. Features for Sockpuppet Accounts Detection

In this section, we describe the features we propose to detect sockpuppet accounts. We consider account-based features and content-based features. The latter features are computed on the text of the edits contributed by each user.

A. Account-based Features

Username-based features: From previous literature, it is evident that characteristics of the chosen username are important to detect spammers, undisclosed paid editing, and sockpuppetry, other malicious behavior [2], [3], [16]. Hence we considered the following features that are derived from the username: (i) the number of digits in a username, (ii) the ratio of digits in a username, (iii) the number of leading digits in a username, and (iv) the unique character ratio in a username.

Average contribution length: Since benign users try to collaborate and contribute more, we expect the length of their contributions to be higher than the contribution length of sockpuppet accounts. Hence, we considered the average contribution length as one of our features.

Average title length: We considered the average length of the titles of the pages a user contributed to.

Average time difference between two consecutive edits: The behavior over time is an important feature for detecting any fraudulent activity [17]. Therefore, we considered the average time difference between two consecutive contributions as another feature.

B. Content-based Features

Traditionally, sockpuppetry detection tasks have been mostly focused on capturing the syntactic inheritance and stylistic of the content [8], [9]. A little emphasis has been put on semantics-focused features. Our major contribution through this research is to explore the semantics of the user contributions to analyze the user behavior further. We hypothesize that considering the semantics of the user edits would capture the deep-level pattern of the content that the same puppet master edits. If a puppet master focuses on a specific type of content or person, that account holder will edit or publish similar content from multiple accounts. To capture the content semantics, we consider the following features.

BERT embedding of user contribution: We used the BERT model to compute the embedding of each user contribution. Specifically, we used the BertTokenizer for tokenization and converting to tensors and the BERT "base" model trained on lower-cased English (12 Transformer layers, 12 self-attention heads, hidden size of 768) from the Hugging face library [18].

User contribution topics: To compute the topics in users' comments, we used the LDA model provided by the Gensim library (we used the WordNetLemmatizer and the bigram model). Specifically, we trained an LDA model with 20 topics on all the users' comments and then assigned to each comment the vector with the corresponding topic distribution.

V. Implementation and Experiments

In order to test the features we are proposing for the automated detection task, we considered different classifiers, namely Logistic Regression, Gaussian Naive Bayes, Decision Tree, Multilayer Perception (MLP), Classifier Random Forest, and a Long short-term memory (LSTM). To evaluate the performance, we considered the F1-score and implemented five-fold cross-validation.

For classical machine learning models, we considered all features described in Section IV-A plus the average vector

[2]Our dataset is available for download at https://github.com/Mostofa-Najmus-Sakib/Wikipedia-Sockpuppetry

[3]https://www.mediawiki.org/wiki/API:Main_page

[4]https://en.wikipedia.org/wiki/Category:Suspected_Wikipedia_sockpuppets

TABLE I
F1-SCORE COMPARISON OF DIFFERENT MACHINE LEARNING MODEL WITH OUR PROPOSED FEATURES IN INPUT TO PREDICT SOCKPUPPET ACCOUNTS. BEST SCORES ARE IN BOLD.

Classifier	F1-score
Logistic Regression	0.75
Gaussian NB	0.60
Decision Tree	0.75
MLP Classifier	0.77
Random Forest	**0.82**
LSTM	0.75

of the user contributions' BERT embeddings and the average vector of the user contributions' topics to capture the user semantics. For the LSTM model, we considered in input the sequence of features for each edit. For each edit, we considered the contribution length, the title length, the time difference between the current and previous edits, the BERT embedding of the contribution, and the vector of topics of the contribution. Finally, we concatenated the username-based features to the representation of the last cell of the LSTM and passed them to the classification layer. Results are shown in Table I. As we can see, among all the considered machine learning models, Random Forest achieves the best F1-score of 0.82. These models perform better than LSTM, which achieves a lower F1-score of 0.75.

A. Feature Analysis

To measure the feature importance, we performed feature ablation, i.e., for each group g of considered features, we removed g and performed the classification with the remaining features. The higher the drop in F1-score, the more important the group of features for the classification task. Results are shown in Figure 1. As we can see, the most important group of features is the one of LDA topics as removing it decreases the F1-score to 0.72. The second most important feature group contains the average contribution length, the average title length, and the average time difference between two consecutive edits. Removing this group of features decreases the F1-score to 0.81. By analyzing these features, we found that sockpuppet accounts do shorter contributions as compared to benign users (mean average contribution length of 27 vs. 31 characters), edit pages with longer titles (the mean average title length is 18 for sockpuppets vs. 17 characters for benign users), and edit more frequently (the mean average time difference between two consecutive edits is 3.5 days vs. 17 days for benign users). Username-based features and the BERT embedding of user comments are equally important, and removing them slightly decreases the F1-score. Removing both of them drops the F1-score to 0.81.

B. Comparison with Related Work

We compare our proposed features with the following:

ORES: The Objective Revision Evaluation Service (ORES) is a web service developed by Wikimedia Foundation that provides a machine learning-based scoring system for edits. More specifically, given an edit, ORES provides a probability

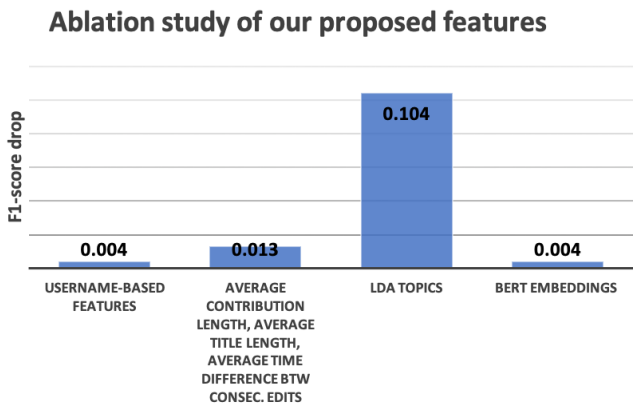

Fig. 1. Ablation study of our proposed features: drop in F1-score for each considered group of features.

distribution (draft quality scores) of being in one of the following four classes: spam, vandalism, attack, or OK. To compare our proposed approach with ORES, we retrieved the draft quality scores of each edit for each user by using the ORES publicly available API [5] and used them in input to a machine learning algorithms to predict sockpuppet accounts. We averaged the draft quality scores of all the edits of the same user when using classical machine learning algorithms, while we considered the sequence of the draft quality scores for the edits of the same user in input to the LSTM.

Yamak et al. [10]: Yamak et al. consider the following set of features for each edit: (i) the number of user's contributions by namespaces, (ii) the average of bytes added and removed from each revision, (iii) the average of contributions in the same article, and (iv) the interval between the user's registration and his first contribution (we did not consider this feature as we do not have this information in our dataset). [6] We used the abovementioned features in input to classical machine learning classifiers. At the same time, for the LSTM, we considered the following features to describe each contribution: namespace and bytes added or removed.

Solorio et al. [8]: Solorio et al. propose stylistic, grammatical, and formatting features to capture the writing style of each user by computing several features for each user contribution, including punctuation and emoticons count, parts of speech tags frequency, number of characters, sentences and tokens, first and third person pronouns frequency, and other features. We averaged all the features among the same user contributions when putting them in input to Random Forest, while we considered the feature sequence in input to LSTM.

The F1-score of our proposed approach and the considered competitors is shown in Table II, where we also compare the

[5]https://ores.wikimedia.org

[6]Yamak et al. [10] also included a feature that considers whether an edit has been reverted by another user, making the detection not completely automated as human input is required. As we propose an automatic detection approach that does not rely on human input, we did not include the reverted-based feature in our implementation of the Yamak et al. approach for a fairer comparison.

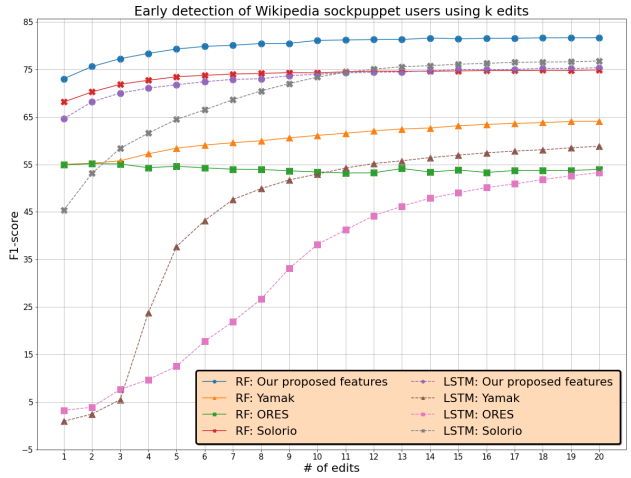

Fig. 2. Early detection of Wikipedia sockpuppet accounts.

TABLE II
F1-SCORE COMPARISON OF OUR PROPOSED FEATURES VS. RELATED WORK. WE COMPARE FEATURES IN INPUT TO RANDOM FOREST (RF – WHICH RESULTS THE BEST CLASSICAL MACHINE LEARNING ALGORITHM) AND LSTM. BEST SCORES ARE IN BOLD.

	F1-score
Our proposed features with RF	**0.82**
Our proposed features with LSTM	0.75
ORES with RF	0.54
ORES with LSTM	0.53
Yamak et al. [10] with RF	0.64
Yamak et al. [10] with LSTM	0.59
Solorio et al. [8] with RF	0.75
Solorio et al. [8] with LSTM	0.77

features in input to the best classical machine learning model (Random Forest also in the case of all competitors) and LSTM. As we can see, our proposed approach achieves a higher F1-score as compared to ORES with RF , Yamak et al. [10] with RF, and Solorio et al. [8] with LSTM, which achieve an F1-score of 0.54, 0.64, and 0.77, respectively.

C. Early Detection of Wikipedia Sockpuppet Accounts

We study the effect of the first-k edits made by the user on the prediction F1-score. Figure 2 shows the variation in F1-score when k is varied from 1 to 20. We show our features compared to related work features in input to Random Forest and LSTM. Our proposed set of features is able to detect a sockpuppet account with an F1-score of 0.73 by just considering the user's first edit (vs. 0.68 achieved by Solorio et al. [8]) and an F1-score of 0.80 by considering the first six edits. Moreover, Random Forest is always better than LSTM, especially for the early prediction. The only exception is given by Solorio et al. [8], where LSTM is slightly better starting from 12 edits.

VI. CONCLUSIONS

In this paper, we presented our proposed approach to address the problem of automatically identifying sockpuppet accounts on Wikipedia. We showed that our set of proposed features in input to a random forest classifier achieves an F1-score of 0.82, outperforms past work, and is able to detect sockpuppet accounts with an F1-score of 0.73 by just considering the first edit. We also showed that computing the topics of the user contributions is particularly important for detecting these types of malicious accounts. In future work, we plan to test our features on predicting whether two accounts belong to the same sockpuppet investigation.

REFERENCES

[1] S. Kumar, F. Spezzano, and V. S. Subrahmanian, "VEWS: A wikipedia vandal early warning system," in *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2015*. ACM, 2015, pp. 607–616.

[2] T. Green and F. Spezzano, "Spam users identification in wikipedia via editing behavior," in *Proceedings of the Eleventh International Conference on Web and Social Media, 2017*. AAAI Press, 2017, pp. 532–535.

[3] N. Joshi, F. Spezzano, M. Green, and E. Hill, "Detecting undisclosed paid editing in wikipedia," in *Proceedings of The Web Conference 2020*, 2020, pp. 2899–2905.

[4] B. Viswanath, A. Post, K. P. Gummadi, and A. Mislove, "An analysis of social network-based sybil defenses," *ACM SIGCOMM Computer Communication Review*, vol. 41, no. 4, pp. 363–374, 2011.

[5] Z. Bu, Z. Xia, and J. Wang, "A sock puppet detection algorithm on virtual spaces," *Knowledge-Based Systems*, vol. 37, pp. 366–377, 2013.

[6] D. Liu, Q. Wu, W. Han, and B. Zhou, "Sockpuppet gang detection on social media sites," *Frontiers of Computer Science*, vol. 10, no. 1, pp. 124–135, 2016.

[7] S. Kumar, J. Cheng, J. Leskovec, and V. S. Subrahmanian, "An army of me: Sockpuppets in online discussion communities," in *Proceedings of the 26th International Conference on World Wide Web, WWW 2017*, 2017, pp. 857–866.

[8] T. Solorio, R. Hasan, and M. Mizan, "A case study of sockpuppet detection in wikipedia," in *Proceedings of the Workshop on Language Analysis in Social Media at NAACL HTL*, 2013, pp. 59–68.

[9] ——, "Sockpuppet detection in wikipedia: A corpus of real-world deceptive writing for linking identities," *arXiv preprint arXiv:1310.6772*, 2013.

[10] Z. Yamak, J. Saunier, and L. Vercouter, "Detection of multiple identity manipulation in collaborative projects," *Proceedings of the 25th International Conference Companion on World Wide Web*, 2016.

[11] ——, "Sockscatch: Automatic detection and grouping of sockpuppets in social media," *Knowledge-Based Systems*, vol. 149, pp. 124–142, 2018.

[12] M. Tsikerdekis and S. Zeadally, "Multiple account identity deception detection in social media using nonverbal behavior," *IEEE Transactions on Information Forensics and Security*, vol. 9, no. 8, pp. 1311–1321, 2014.

[13] X. Zheng, Y. M. Lai, K. Chow, L. C. Hui, and S. Yiu, "Detection of sockpuppets in online discussion forums," Ph.D. dissertation, University of Hong Kong, 2011.

[14] S. K. Maity, A. Chakraborty, P. Goyal, and A. Mukherjee, "Detection of sockpuppets in social media," in *Companion of the 2017 ACM Conference on Computer Supported Cooperative Work and Social Computing*, 2017, pp. 243–246.

[15] S. Adhikari, "Detection of sockpuppet accounts on reddit," 2020.

[16] R. Zafarani and H. Liu, "10 bits of surprise: Detecting malicious users with minimum information," in *Proceedings of the 24th ACM International Conference on Information and Knowledge Management, CIKM 2015*, 2015, pp. 423–431.

[17] K. Lee, B. D. Eoff, and J. Caverlee, "Seven months with the devils: A long-term study of content polluters on twitter," in *Proceedings of the Fifth International Conference on Weblogs and Social Media, Barcelona, Catalonia, Spain, July 17-21, 2011*, L. A. Adamic, R. Baeza-Yates, and S. Counts, Eds. The AAAI Press, 2011.

[18] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.