Exploring Author Profiling for Fake News Detection

This paper was downloaded from TechRxiv (<https://www.techrxiv.org>).

LICENSE

CC BY 4.0

SUBMISSION DATE / POSTED DATE

14-12-2021 / 16-12-2021

CITATION

Rathod, Shloak (2021): Exploring Author Profiling for Fake News Detection. TechRxiv. Preprint.
<https://doi.org/10.36227/techrxiv.17197655.v1>

DOI

[10.36227/techrxiv.17197655.v1](https://doi.org/10.36227/techrxiv.17197655.v1)

Exploring Author Profiling for Fake News Detection

Shloak Rathod

Aditya Birla World Academy

Mumbai, India

abwa.shloak.rathod2021@abet.co.in

Abstract—The proliferation of online media allows for the rapid dissemination of unmoderated news, unfortunately including fake news. The extensive spread of fake news poses a potent threat to both individuals and society. This paper focuses on designing author profiles to detect authors who are primarily engaged in publishing fake news articles. We build on the hypothesis that authors who write fake news repeatedly write only fake news articles, at least in short-term periods. Fake news authors have a distinct writing style compared to real news authors, who naturally want to maintain trustworthiness. We explore the potential to detect fake news authors by designing authors’ profiles based on writing style, sentiment, and co-authorship patterns. We evaluate our approach using a publicly available dataset with over 5000 authors and 20000 articles. For our evaluation, we build and compare different classes of supervised machine learning models. We find that the K-NN model performed the best, and it could detect authors who are prone to writing fake news with an 83% true positive rate with only a 5% false positive rate.

Index Terms—Fake News Detection, Machine Learning, Feature Extraction, Author Profiling, Internet Security

I. INTRODUCTION

The dawn of the information era has facilitated exponential growth in online news consumption. One factor that has led to the shift is the increase in accessibility and availability offered by online news. About 86% of Americans say they choose to read news online “often” or “sometimes” [1]. Indeed, online news has become the go-to news source for a vast majority of the population. While online news adds to accessibility, it leads to the rise of a potent threat: fake news. We define fake news as news that is intentionally and verifiably false. The rapid dissemination of unmoderated news online invariably allows authors to promote their political, financial, or personal biases, thereby decreasing trust in online media. A 2021 survey by Oxford University and The Reuters Institute found that only 44% of respondents responded by saying “they trust most news most of the time” [2]. Another consequence of fake news is that it manipulates readers into believing false information and making decisions based on it. Hence, to mitigate these adverse effects, we need methods to detect fake news automatically.

Detecting fake news is a daunting challenge. Fake news includes ambiguity and is often presented as a “story” rather than just news, therefore, adding biases and building fallacies. Popular media companies such as FactCheck.org¹ and PolitiFact² debunk fake news using professional fact-checkers.

However, this approach is labor-intensive, expensive, and time-consuming.

This paper proposes an automated approach that focuses on author profiling to detect authors who write fake news. We hypothesize that certain authors primarily and repeatedly write fake news articles. A study by Toby Hopp et al. also finds that a select group of authors repeatedly publish fake news articles [3]. Along the same lines, a series of experiments by Sacha Altay et al. find that writing an article “stamped” as fake can impact an author’s reputation in a manner that is difficult to rectify [4]. Most authors are motivated to maintain a good reputation by writing trustworthy news articles. However, authors who are primarily engaged in writing fake news do not have an incentive to maintain a good reputation. Therefore, authors who write fake news follow a distinct writing pattern in their articles. Solely because fake and real news authors have differing intentions and writing styles, we believe that profiling authors can be a reliable method to capture these differences and distinguish between authors who write fake and real news. For that reason, we attempt to establish and evaluate writing-based features to detect fake news authors.

In this paper, we make the following contributions:

- We propose and analyze features relating to stylometry, sentiment and emotion, named entities, and co-authorship patterns to detect fake news authors.
- We measure and study our proposed features using a publicly available dataset with over 5000 authors and 20000 articles.
- We plan and execute an experiment to answer the question, “which feature group is the most effective in detecting fake news authors?”
- We evaluate our features using supervised machine learning models. In addition, we compare the models to gauge which model works best with our approach and features.

The structure of our paper is as follows; Section II offers a review of past work. Section III highlights major fake news events and their implications. Section IV describes the dataset and the data preprocessing steps that we take. Section V introduces and analyzes our features. We then implement and evaluate our features in Section VI using supervised machine learning models. Section VI also presents an experiment to find the most influential feature group in detecting fake news authors. Finally, Section VII discusses future work, and Section VIII concludes our approach.

¹<https://www.factcheck.org>

²<https://www.politifact.com>

II. RELATED WORK

Fake news detection using machine learning has recently received significant research attention. Among the approaches to detect fake news, researchers have relied on word linguistics, author profiling, knowledge, and word embeddings to detect fake news.

Linguistics based approaches focus on an author's writing style, by analyzing emotion and stylometry. Anastasia Giachanou et al. [5] introduce emoCred, an LSTM-based neural network that extracts emotion scores from the text. Their approach counts the number of emotional words and calculates the emotional intensity expressed. One of their models, EmoCred-emoReact, achieves an F1-score of 61.7%. Similarly, another approach combines emotion-based and behavioral-based profiling (average hashtags per Tweet and average Retweets per Tweet) [6] to detect fake news. The approach evaluates the features using a random forest model, achieving an accuracy of 74%. Likewise, an approach by Martin Potthast et al. [7] incorporates commonly used stylistic features (n-grams of characters and readability scores) and some news domain-specific features (ratios of quoted words and external links, the number of paragraphs, and average sentence length). The approach implements a random forest classifier to evaluate the features, achieving an F-score of 81%. However, in our paper, we combine some of the aforementioned stylistic features with other feature groups to build an author's profile.

Author profiling is employed to build an author's reputation. Carlos Castillo et al. [8] propose a model which extracts user-based features (the number of followers and the number of Tweets posted), message-based features (the number of hashtags and sentiment), propagation-based features (Retweets), and topic-based features to profile an author. They train a supervised machine learning classifier, achieving an F1-score of 84.9%. Similarly, Jakab Buda et al. [9] pair an n-grams model with features such as the mean length of Tweets, the number of Retweets, and hashtags per author, achieving an accuracy of 75%. On the other hand, we implement an author profiling approach on articles rather than social media platforms, which is an area of study that is yet to be explored.

Knowledge-based fake news detection is the most straightforward method today. Media companies like Snopes³ employ an expert-oriented approach to detect fake news. Similarly, Fiskkit⁴, a media company, relies on a crowdsourcing-oriented approach to detect fake news. However, computational knowledge-based approaches differ. Ciampaglia et al. [10] propose an approach that finds the shortest paths between concept nodes and knowledge graphs, e.g., ("Socrates," "is a," "person"). They propose a metric to determine how truthful a statement is by analyzing path lengths between the concepts in a sentence. Our approach incorporates a knowledge-based feature referred to as entities. However, we combine it with other feature groups to build an author's profile.

³<https://www.snopes.com>

⁴<https://www.fiskkit.com>

Word embeddings is a popular approach among researchers. Anurag Goswami et al. [11] propose FakeBERT. The approach employs word embeddings using Glove to develop features. To detect fake news, they combine convolutional neural networks (CNN) with bidirectional encoder representations from transformers (BERT), achieving an accuracy of 98.9% and outperforming the current state-of-the-art model. Similarly, Prateek Agrawal et al. [12] pair linguistic and context-based features to develop WELFake. They implement one-hot encoding and term frequency-inverse document frequency (TF-IDF) to extract context-based features. They then pair a voting classifier with a model using TF-IDF and one-hot encoding. Finally, they achieve an experimental accuracy of 96.7% on their dataset. Our approach does not focus on a word embedding approach; instead, we profile authors. We choose to profile authors since the complexities of fake news detection cannot be solely understood by natural language processing (NLP) alone. Hence, a NLP deep learning model is bound to have limitations. On the other hand, our author profiling approach entails mining data from articles.

III. CASE STUDIES

This section highlights two instances where fake news has reportedly inflicted real and large-scale consequences on society. We know that the problem of fake news has persisted for a long time, dating to 1475 [13]. However, the dawn of the internet and advancements in technology have amplified fake news due to echo chambers and filter bubbles. The most extensive study on fake news finds that falsehoods are 70% more likely to be Retweeted than the truth [14].

One example where fake news had a real impact was during the 2016 US election when fake news defaced political campaigns [15]. Figure 1 by BuzzFeed shows that the top 5 fake news stories gained more engagement than mainstream media stories on Facebook. Furthermore, a research article published in Nature Communications finds that 25% of Tweets spread either fake or highly biased news [16]. The same research work also shows that fake news is highly polarizing. The implications of these findings suggest that fake news can potentially alter political outcomes.

Fig. 1. The top 5 fake news and mainstream news stories and their Facebook engagement three months prior to the 2016 US election day. BuzzFeed created the figure [15].

A second example where fake news had a sizable impact is the spread of reliable information during the COVID-19

pandemic. Oxford University and The Reuters Institute found that 54% of people say they have seen misinformation about COVID-19 [2]. Another study shows that over 25% of the most viewed YouTube videos on COVID-19 contained misleading information, totaling over 62 million views worldwide [17]. Fake news has led to a human toll, too; according to the American Journal of Tropical Medicine and Hygiene, an estimated 5,800 people have been admitted to the hospital after they followed advice found online which was later flagged false [18]. These examples elucidate how fake news can affect the flow of trustworthy information in times of uncertainty and stress.

IV. DATA

This section describes the dataset we use for feature exploration and experimentation. We also present a brief analysis of the dataset and explain the data preprocessing we conduct.

A. Dataset

We use a public dataset from a Kaggle competition for fake news detection [19]. The dataset was made available in 2019, and it has 26000 articles and 5733 unique authors. The dataset contains the author name, article title, article text, and label (i.e., *fake* or *real*). We divide the dataset into 20800 articles and 4202 authors for training and 5200 articles and 1531 authors for testing. The training dataset contains 10413 fake articles and 10387 real news, indicating a balanced training dataset. Interestingly, we find that only a minority group of authors from our training dataset have published both fake and real news articles. This finding corroborates our hypothesis that authors tend to write either fake news or real news articles.

We base our approach on compelling differences between fake and real news authors. For example, Comrade Torcer, a fake news author from our dataset, writes, “In Hillary’s America, email server scrubs you. Obama transfers his Nobel Peace Prize to anti-Trump rioters.” The author also makes false claims such as “Clinton Foundation in foreclosure as foreign donors demand refunds.” Overall, the author’s tone is exaggerated, polarizing, and shocking. Whereas Jerry Garrett, a real news author, reports articles in a formal tone. The author writes, “The 3.4 million U. S. citizens on the Caribbean island of Puerto Ricans will vote today, for the fifth time, on a referendum to become independent or to apply to be America’s 51st state, despite already declaring bankruptcy last month.” It is lucid that the author is reporting the news rather than adding any bias or a surprising tone. We use these preliminary findings as an impetus to engineering our features in Section V.

B. Data Preprocessing

First, we drop all rows that contain NaN (not a number) values in 3 columns: author name, text, and label (i.e., *fake* or *real*). Second, we drop the minority group of authors who published both fake and real news articles to investigate our hypothesis: a definite group of authors primarily and

repeatedly write fake news articles. Finally, we create feature vectors for every author. The feature vector consists of the mean feature values for every feature. For instance, to calculate the feature value for polarity we use:

$$\frac{\text{sum of polarity for all articles written by an author}}{\text{total number of articles written by an author}}$$

We create an author’s profile (feature vector) using this method. To compute feature values, we use various Python libraries such as Spacy [20], Natural Language Tool Kit (NLTK) [21], Text2emotion [22], Lexicalrichness [23], Textblob [24], and Stanford Core NLP [25]. These libraries conduct further data preprocessing by removing stop-words and normalizing the text. We use this preprocessed data frame in Section V to build our features.

C. Preliminary Data Analysis

This section presents some initial data analysis that motivated us to build and assess authors’ profiles.

First, we find that real news authors publish more articles than fake news authors. We plot an Empirical Distribution Function (ECDF) comparing the number of articles published by fake and real news authors in our dataset. Figure 2 distinctly shows that about 60% of real news authors publish more articles than fake news authors.

Fig. 2. The ECDF compares the distribution of the number of articles published per fake and real news author.

Second, we conduct topic analysis to find the common topics in our dataset using Latent Dirichlet Allocation (LDA) [26]. We present the three major topics for fake news, real news, and all news (fake and real) in Table I. We find that the topics are all related to the 2016 US election. Additionally, we find that fake news is mainly about Hilary Clinton. This notion is also corroborated by Figure 1, where fake news articles on Facebook targeted Hilary Clinton during the 2016 US election.

TABLE I
TOPICS OF ARTICLES

News type	Topic 1	Topic 2	Topic 3
Fake news	Russia	Hilary Clinton	War
Real news	Donald Trump and women	Tax	School shooting
All news	China	Election	Police

The common topics of articles in our dataset for three types of news (fake, real, and all) are shown using Latent Dirichlet Allocation.

V. APPROACH: BUILDING AN AUTHOR'S PROFILE

This section examines four features groups stylometry, named entities, sentiment and emotion, and authorship, as shown in Figure 3.

Fig. 3. The four features groups we extracted are stylometry, named entities, sentiment and emotion, and authorship.

A. Authorship Features

In this group, we attempt to capture the co-authorship patterns of an author. An author's co-authors and reputation reflect the author's goal: to publish real or fake news articles. We measure two features in this group: the number of co-authors and the number of articles written by an author. We create two lists to calculate the number of co-authors: unique authors and author pairs. To calculate the number of co-authors, we identify author pairs containing a particular author and count the number of unique authors in that pair; we make sure not to count any author twice. Similarly, to calculate the number of articles written by an author, we count the number of times an author's name occurs in the author column.

Fig. 4. A bar plot displaying how many co-authors [0,5] fake and real news authors have worked with in terms of percentage.

After normalizing the data, we find that real news authors publish more articles than fake news authors. Our analysis also shows that real news authors have more co-authors than fake news authors. Figure 4, a bar plot, plots the percentage of authors against the number of co-authors they have. The figure shows that 47% of real news authors have worked with at least one co-author. In comparison, just 7% of fake news authors have worked with at least one co-author.

B. Entities

In this group, we count the number of named entities that an author refers to in their article. Named entities include people, places, countries, and political parties. They give us an insight into the real-world connections authors make in their articles. We believe that a fake news article will probably contain fewer entities since it is harder to falsify real-world connections. Conversely, real news authors can easily make real-world references as these events actually take place. To calculate the average number of entities per author, we use the Spacy Python library [20]. The Spacy library employs a statistical entity recognition system that assigns labels (entities) to tokens (words).

Fig. 5. The Empirical distribution function (ECDF) compares the distribution of the number of people mentioned by fake and real news authors. The number of entities was computed using the Spacy Python library [20].

Figure 5 is an Empirical distribution function comparing the number of people mentioned by fake and real news authors. We observe that 50% of real news authors mention 0 to 25 people. In comparison, 50% of fake news authors mention 0 to 7 people. Thus, we conclude that real news authors mention more people than fake news authors.

C. Sentiment and Emotions

Research at MIT shows that fake news supports the “novelty hypothesis,” which states that novelty attracts human attention because it is surprising and emotionally arousing [14]. In our effort to profile authors, we compute emotions and compound sentiment.

We use the Text2emotion library [22] to compute five emotions: surprise, happy, sad, fear, and anger. The library checks and counts the number of words that express emotions or feelings to score every emotion. Likewise, we use the Valence Aware Dictionary and sEntiment Reasoner (VADER) library to calculate compound sentiment [21]. VADER sentiment analysis relies on a dictionary, which maps lexical features to emotional intensities called sentiment scores.

Figure 6 is a box plot comparing the computed values of surprise between real and fake news authors. First, we notice that the median surprise score for real news authors is lower than that for fake news authors. We notice that the interquartile range (IQR) of real news authors, about 0.08, is significantly lower than that of fake news authors, about 0.14. The difference in IQR leads us to believe that real news

Fig. 6. The box plot compares the surprise score between real and fake news authors.

authors have a more consistent writing style when compared to fake news authors. Our findings also support the “novelty hypothesis.” Fake news, indeed, is more surprising than real news. We conduct further analysis on other emotion scores, and we find that the IQR for all emotions among real news authors is significantly lower than the IQR for all emotions among fake news authors. Therefore, we believe that real news authors have a more consistent writing style than fake news authors.

Example: To showcase the practical effect of the surprise score, we select two authors with differing surprise scores and analyze why they receive those scores. Terry Coleman, an author with a surprise score of 0.9, writes, “You are as big a liar as Hillary!” The author is making a shocking and polarizing statement. Unsurprisingly, his articles are labeled as fake. In contrast, Thom Shanker, an author with a surprise score of 0, writes, “Although global tensions and terrorist threats have shown few signs of diminishing, the total size of the global arms trade dropped to around \$80 billion in 2015 from the 2014 total of \$89 billion, the study found.” Evidently, his factual article maintains a consistent, formal, and reporter-like tone with the sole purpose to report rather than persuade or surprise. Unsurprisingly, his articles are labeled as real. We, therefore, realize how the tone (emotion scores) affects the credibility of an author.

Fig. 7. The empirical distribution function (ECDF) compares the distribution of compound sentiment $[-1, 1]$ between fake and real news authors.

Figure 7 is an ECDF comparing the compound sentiment between fake and real news authors. A negative compound sentiment score indicates negativity; similarly, a positive compound sentiment score indicates positivity. We notice that

about 31% of fake news authors have a compound sentiment of less than or equal to -0.5 . At the same time, only 19% of real news authors have a compound sentiment of less than or equal to -0.5 . Moreover, 49% of real news authors have a compound sentiment of more than 0.5 ; on the contrary, just 32% of fake news authors have a compound sentiment of more than 0.5 . Therefore, we conclude that fake news authors are clearly more negative in nature than real news authors. Again, we can relate this to the “novelty hypothesis,” as negative news may be more surprising.

D. Stylometry

Stylometric features represent an author’s writing profile. Fake news authors often have malicious intent, which may influence their writing style. In our attempt to extract stylometric features, we compute the polarity score, subjectivity score, the number of quotations, the number of words, and the number of unique words.

We use the Textblob library to compute polarity and subjectivity [24]. Textblob scores polarity within the range $[-1.0, 1.0]$ and subjectivity within the range $[0.0, 1.0]$. In the English language, polarity refers to positive and negative clauses. Subjectivity relates to anything open to interpretation. To count the number of quotes, we use Stanford’s core NLP library [25]. Similarly, to count the number of words and the number of unique words, we use the Lexicalrichness library [23].

Fig. 8. The Empirical distribution function (ECDF) compares the distribution of polarity between fake and real news authors on a scale of -1 to 1 .

Figure 8 is an ECDF comparing the polarity of fake and real news authors. Our analysis of Figure 8 shows that both fake and real news articles are concentrated around a polarity score of 0. However, the polarity score range of 2 for fake news authors is significantly greater than that of 0.5 for real news authors. This difference in range insinuates that real news authors have a more consistent writing style when compared to the writing style of fake news authors.

Example: We further analyze our data by comparing authors who have a polar and nonpolar writing style. Justin Sawchuk, a fake news author, receives a -1 polarity rating; the author writes, “hope he nukes the hell out of CHINA’s disgusting race of people.” Similarly, Kenan Lott, a fake news author, writes, “They also said that Obama care would save you money, and that Benghazi murders were because of a movie, and that the election is not rigged! More Obama lies.”

Both authors employ highly polarizing writing styles. Kenan Lott’s article implies that the Democratic Party rigged the election, which is not true. However, when we analyze Mike Isaac’s article, which receives a polarity rating of 0, we notice his articles use highly formal language and include several statistics and quotations. The author writes, “‘To unmask an anonymous speaker online, the government must have a strong justification,’ said Nathan Freed Wessler, an A.C.L.U. attorney involved in the case.” Unsurprisingly, his articles are deemed as real. Once again, we notice a stark difference in the writing style of real and fake news authors.

Finally, we explore the hypothesis that fake news authors tend to cite fewer sources in their articles than real news authors. We confirm this hypothesis by measuring indicators such as the frequency of the word “said” and the number of quotations. After removing stop words, we plot a frequency distribution graph as shown in Figure 9. The graph shows fake and real news authors’ top 5 most used words. In fact, we notice a distinct difference in the frequency of the word “said” between real and fake news authors. Real news authors use the word “said” more frequently than fake news authors. We also realize that the word “said” relates to the number of quotes in an article (e.g., the senator said “health reforms are an urgent need”). Using Stanford’s core NLP library, we find that, on average, real news authors write approximately 17 quotes per article; whereas, fake news authors write approximately 7 quotes per article. Based on these observations, we confirm our hypothesis that fake news authors cite fewer sources in their articles than real news authors. Moreover, this once again links back to Section V-B, where we claim it is easier for real news authors to make real-world connections since they are referencing factual events that have actually happened.

Fig. 9. The frequency distributions graph compares the most used words by fake and real news authors on a per author basis.

VI. EVALUATION

In this section, we evaluate our features using various machine learning models. We investigate the extent to which our approach can differentiate between authors who write fake news and authors who write real news. We train and test four groups of models, specifically, decision tree, logistic regression, clustering, and support-vector classifier. We evaluate the models using multiple performance metrics and analyze their results. Lastly, we also conduct two experiments. The first

experiment answers the question, “which feature group is the most effective in detecting fake news authors?” The second experiment explores the question, “what if we deployed our models in a setting where users could assess the profile of each author?”

A. Background: Machine Learning Models

This section offers a background on the types of models we use for performance evaluation.

Decision Trees. The decision tree group includes extreme gradient boosting (XGBoost) and decision tree (DT) classifiers. XGBoost refers to an ensemble machine learning algorithm used for binary classification. The decision tree classifier is a decision support tool that employs a tree-like model of decisions and possible consequences. It is a method to approximate discrete-valued target functions. The algorithm learns these functions by constructing them top-down.

Clustering. We implement clustering using k-nearest neighbor (K-NN). K-NN is a statistical classification model that classifies using a plurality vote of its neighbors based on the Euclidean distance from its nearest neighbor.

Support Vector Classifier. The support vector classifier (SVC) computes a hyperplane in a high or infinite-dimensional space. The hyperplane classifies the data, by dividing it into two classes. When the hyperplane has a sizable functional margin, it achieves optimum separation.

Logistic Regression. We use two logistic regression models, Gaussian Naive Bayes (GNB) and logistic regression (LR). Logistic regression is a statistical model that uses a logistic function to model a binary dependent variable. The logistic regression models the output probability; it is not a classifier. However, we use a threshold of 0.5 as a cutoff to make it a binary classifier. Likewise, GNB is a probabilistic classifier based on Bayes’ theorem with strong (naive) independence assumptions between the features. GNB uses a generative approach, applying the Bayes Rule.

B. Training and Testing

We train and test our models based on the preprocessed vector data frame described in section IV. We randomly split our dataset into training and testing with 20800 articles and 4202 authors for training and 5200 articles and 1531 authors for testing.

C. Model Tuning

Tuning models have a sizable impact on a model’s results. We use a grid search approach to find the optimum hyperparameter values for every model. Grid search is an exhaustive method that builds a model with a range of values for each parameter. The method trains and tests the model and outputs the most optimum combination of parameters.

D. Evaluation Metrics

1) *TP and FP rates:* We assign fake and real news authors as positive and negative, respectively. Therefore, a true positive (TP) is a fake news author labeled as fake, a false positive (FP)

is a real news author labeled as fake, a true negative (TN) is a real news author labeled as real, and a false negative (FN) is a fake news author labeled as real. Finally, using the metrics mentioned above, we compute the following metrics:

- **Accuracy:** Accuracy is the percentage of correctly classified authors and is calculated using:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (1)$$

- **Precision:** Precision is defined as the percentage of correctly identified fake news authors out of all the predicted fake news authors and is calculated using:

$$Precision = \frac{TP}{TP + FP} \quad (2)$$

- **Recall:** Recall measures how many fake news authors are actually labeled as fake and is calculated using:

$$Recall = \frac{TP}{TP + FN} \quad (3)$$

- **F1-score:** F1-score computes the balance between recall and precision and is calculated using:

$$F1 \text{ score} = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall} \quad (4)$$

TABLE II
TARGET F1-SCORE, ACCURACY, PRECISION, AND RECALL

Model	F1-score	accuracy	precision	recall
DT	75%	79%	72%	80%
LR	70%	68%	57%	92%
K-NN	83%	85%	76%	92%
SVC	71%	70%	59%	90%
XGBoost	71%	71%	59%	88%
GNB	71%	78%	75 %	68%

The F1-score, accuracy, precision, and recall scores for the 6 models we train: DT, LR, K-NN, SVC, XGBoost, and GNB.

To evaluate and compare the effectiveness of the proposed models, we calculate the F1-score, accuracy, precision, and recall (results found in Table II). The results show that the K-NN model predicts 92% of all fake news authors and 76% of the fake news author predictions are correct. Indeed, the K-NN is our most accurate model in terms of F1-score and accuracy with scores of 85% and 83%, respectively. The DT model identifies 80% of all fake news; however, only 72% of its fake news author predictions are correct, indicating slight over prediction. Likewise, the LR, SVC, and XGBoost have a high recall; hence, indicating they identified most fake news authors. However, they heavily over-predict fake news authors since only 57%, 59%, and 59% of their fake news author predictions are correct, respectively. Lastly, the GNB model has a relatively low recall indicating it predicts less number of fake news authors. Moreover, it achieves a moderate precision, indicating that its fake news author predictions are correct 3out of 4 (75%) times. We ultimately rank our models from best to worst: K-NN, DT, GNB, after which SVC, LR, andXGBoost are least effective.

2) **ROC Curves:** Graph 10 is a Receiver Operating Characteristic (ROC) curve that plots the true positive rate (TPR) against the false positive rate (FPR).

- **True positive rate:** The true positive rate is the recall or probability of detection and is defined as:

$$FPR = \frac{FP}{TP + FN} \quad (5)$$

- **False positive rate:** The false positive rate is the probability of a false alarm and is defined as:

$$TPR = \frac{TP}{TP + FN} \quad (6)$$

Fig. 10. The Receiver Operating Characteristics (ROC) curve plots the true positive rate (TPR) against the false positive rate (FPR) for the six machine learning models we test.

The area under the curve (AUC) equals the probability that a classifier will rank a randomly chosen fake news author higher (more false) than a randomly chosen real news author. The AUC is significant since the decision threshold can be used to control the trade-off between the FPR and the TPR. In general, the higher the AUC, the better the model performs. It is worth mentioning that AUC is more statistically consistent and more discriminating than accuracy [27].

Graph 10 compares the ROC curve for the six models we test. Distinctly, K-NN has the highest AUC of 93%, followed by GNB and XGBoost. We also find that SVC is suboptimal when the FPR is less than 10%.

TABLE III
5-FOLDS ACCURACY

Model	iteration 1	iteration 2	iteration 3	iteration 4	iteration 5
DT	76%	78%	72%	74%	72%
LR	72%	73%	70%	68%	70%
K-NN	85%	83%	80%	82%	81%
SVC	62%	67%	68%	64%	70%
XGBoost	72%	71%	70%	78%	78%
GNB	77%	78%	72%	73%	72%

The table shows the accuracy of DT, LR, K-NN, SVC, XGBoost, and GNB over 5 iteration, using the 5-folds accuracy method.

3) **K-fold Cross-Validation:** To further evaluate our model, we use K-fold Cross-Validation, a resampling procedure used to evaluate models on a limited data sample. Moreover, K-fold Cross-Validation indicates how well a model will perform

on unseen data. Cross-Validation also flags problems such as over-fitting and selection bias. Overfitting occurs when a model fits the limited set of data points too closely. When this happens, the model cannot perform accurately against unseen data, decreasing the accuracy. Selection bias occurs when proper randomization of data is not achieved, thereby the training data is not representative of the total population intended to be analyzed.

We conduct a 5-fold Cross-Validation test and present our results in Table III. We observe that the K-NN model has a relatively consistent accuracy between 80% to 85%, which shows it is robust. The K-NN also performs better than the rest of the models, showing no signs of over-fitting. Likewise, the DT, GNB, and LR models have a consistent accuracy, indicating robustness. However, XGBoost and SVC show significant variance in accuracy; thus, we believe these models experienced a selection bias or over-fit the training dataset.

4) *Comparing Feature Groups*: This section evaluates the four feature groups: authorship, named entities, sentiment and emotion, and stylometry. These groups are described in section V. We aim to answer the question, “which feature group is the most effective in detecting fake news authors?” To investigate this question, we set up an experiment.

We create a feature vector for every feature group. For instance, to test the authorship feature group, we create a feature vector with only the authorship features. We then train a K-NN model based on that feature vector. We conduct this procedure on all four feature groups and plot a ROC curve to analyze our findings and conclude.

Fig. 11. The Receiver Operating Characteristics (ROC) curve compares the true positive rate (TPR) against the false positive rate (FPR) for four feature groups: authorship, named entities, sentiment and emotion, and stylometry.

Figure 11 plots ROC curves for the four feature groups. We observe that all feature groups build models better than the suboptimal level. Thus, all feature groups contribute to detecting fake news authors. However, to answer our question, we notice that the stylometry feature group outperforms all other feature groups, achieving a 93% AUC score. Second, the entity feature group achieves an 83% AUC score. Lastly, the sentiment and emotion feature group achieves a 65% AUC score, followed by the authorship group, which achieves a 62% AUC score. The results of this experiment are comprehensive; the stylometry feature group proves most effective in detecting fake news authors.

5) *Hypothetical Deployment and Operational Points*: We test our models considering a hypothetical deployment scenario. We explore the question, “What if we deployed our models in a setting where users could assess the profile of each author?” Falsely classifying news authors as fake has a dire repercussion on the author’s reputation; hence, a type I error (false positive) must be avoided if possible. Thus, we set our alpha to 0.05 that means we want the probability of a type I error, or a false positive, to be less than 5%. We measure the corresponding TPR for each model based on a 5% FPR rate and our ROC curve. We represent our results in Table IV.

TABLE IV
TP AND FP RATES IN A HYPOTHETICAL DEPLOYMENT SCENARIO

Model	True Positive Rate
DT	27%
LR	12%
K-NN	83%
SVC	4%
XGBoost	41%
GNB	40%

We fix the FPR at 5% and present the corresponding TPR value. The K-NN model performs best.

We distinctly observe that the K-NN model outperforms all other models with a TPR of 83%. The other models perform less accurately in a hypothetical deployment scenario and cannot be deployed in the real world. Overall, we conclude that the K-NN model will be effective in a hypothetical situation.

VII. FUTURE WORK

Fake news detection is an emerging and popular research area. As for future work, we lay out a brief plan to implement our approach in a hypothetical deployment situation. We propose creating an author wise database with their credibility record; we can use this database to predict the credibility of the author, using our model. Our proposed approach will allow for more real-time results. Other than that, to investigate the most influential feature group further, we propose an experiment where one feature group will be dropped each time, and the results are plotted and compared using ROC curves. The experiment will elucidate which group is the least effective in detecting fake news authors.

VIII. CONCLUSION

Fake news has been a real problem that gained researchers’ attention, especially after the 2016 US elections. There have been various instances when fake news had a lasting impact on individuals and society.

In this paper, we analyze news articles to build authors’ profiles. We analyze and engineer features to build authors’ profiles based on the hypothesis that specific authors primarily and repeatedly write fake news. Primarily, our features are from four groups, stylometry, named entities mentioned, sentiment and emotion, and authorship. Lastly, we evaluate our features using six different machine learning techniques and multiple evaluation metrics. The K-Nearest Neighbors model

performs best, achieving a 93% area under the curve score and an 83% true positive rate in a hypothetical deployment environment (5% false positive rate).

ACKNOWLEDGMENT

We would like to extend our sincere thanks to Dr. Maria Konte for her feedback on this paper.

REFERENCES

- [1] S. Elisa and M. Amy, "News use across social media platforms in 2020," 2021, (Last Accessed: 19-Nov-2021). [Online]. Available: <https://www.pewresearch.org/journalism/2021/01/12/news-use-across-social-media-platforms-in-2020/>
- [2] N. Newman, R. Fletcher, A. Schulz, S. Andi, C. T. Robertson, and R. K. Nielsen, "Reuters institute digital news report 2021," *Reuters Institute for the Study of Journalism*, 2021. [Online]. Available: <https://ssrn.com/abstract=3873260>
- [3] T. Hopp, P. Ferrucci, and C. J. Vargo, "Why Do People Share Ideologically Extreme, False, and Misleading Content on Social Media? A Self-Report and Trace Data-Based Analysis of Countermedia Content Dissemination on Facebook and Twitter," *Human Communication Research*, vol. 46, no. 4, pp. 357–384, 2020. [Online]. Available: <https://doi.org/10.1093/hcr/hqz022>
- [4] S. Altay, A.-S. Hacquin, and H. Mercier, "Why do so few people share fake news? it hurts their reputation," *New Media & Society*, 2020. [Online]. Available: <https://doi.org/10.1177/1461444820969893>
- [5] A. Giachanou, P. Rosso, and F. Crestani, "Leveraging emotional signals for credibility detection," in *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR'19. New York, NY, USA: Association for Computing Machinery, 2019, p. 877–880. [Online]. Available: <https://doi.org/10.1145/3331184.3331285>
- [6] M. Cardaioli, S. Cecconello, M. Conti, L. Pajola, and F. Turrin, "Fake news spreaders profiling through behavioural analysis," in *Conference and Labs of the Evaluation Forum*, vol. 2696, no. 113, 2020.
- [7] M. Potthast, J. Kiesel, K. Reinartz, J. Bevendorff, and B. Stein, "A stylometric inquiry into hyperpartisan and fake news," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Melbourne, Australia: Association for Computational Linguistics, Jul. 2018, pp. 231–240. [Online]. Available: <https://aclanthology.org/P18-1022>
- [8] C. Castillo, M. Mendoza, and B. Poblete, "Information credibility on twitter," in *Proceedings of the 20th International Conference on World Wide Web*, ser. WWW '11. New York, NY, USA: Association for Computing Machinery, 2011, p. 675–684. [Online]. Available: <https://doi.org/10.1145/1963405.1963500>
- [9] E. Katona, J. Buda, and F. Bolonyai, "Using n-grams and statistical features and to identify hate speech spreaders on twitter," in *CLEF*, vol. 2936, no. 176, 2021.
- [10] G. L. Ciampaglia, P. Shiralkar, L. M. Rocha, J. Bollen, F. Menczer, and A. Flammini, "Computational fact checking from knowledge networks," *PLOS ONE*, vol. 10, no. 6, pp. 1–13, 06 2015. [Online]. Available: <https://doi.org/10.1371/journal.pone.0128193>
- [11] R. K. Kaliyar, A. Goswami, and P. Narang, "Fakebert: Fake news detection in social media with a bert-based deep learning approach," *Multimedia Tools and Applications*, vol. 80, no. 8, pp. 11 765–11 788, 2021.
- [12] P. K. Verma, P. Agrawal, I. Amorim, and R. Prodan, "Welfake: Word embedding over linguistic features for fake news detection," *IEEE Transactions on Computational Social Systems*, vol. 8, no. 4, pp. 881–893, 2021.
- [13] J. Soll. (2016) The long and brutal history of fake news. [Online]. Available: <https://www.politico.com/magazine/story/2016/12/fake-news-history-long-violent-214535/>
- [14] S. Vosoughi, D. Roy, and S. Aral, "The spread of true and false news online," *Science*, vol. 359, no. 6380, pp. 1146–1151, 2018. [Online]. Available: <https://www.science.org/doi/abs/10.1126/science.aap9559>
- [15] C. Silverman. (2016) This analysis shows how viral fake election news stories outperformed real news on facebook. [Online]. Available: <https://www.buzzfeednews.com/article/craigsilverman/viral-fake-election-news-outperformed-real-news-on-facebook>
- [16] A. Bovet and H. Makse, "Influence of fake news in twitter during the 2016 us presidential election," *Nature Communications*, vol. 10, 01 2019.
- [17] H. O.-Y. Li, A. Bailey, D. Huynh, and J. Chan, "Youtube as a source of information on covid-19: a pandemic of misinformation?" *BMJ Global Health*, vol. 5, no. 5, 2020. [Online]. Available: <https://gh.bmj.com/content/5/5/e002604>
- [18] M. S. Islam, T. Sarkar, S. H. Khan, A.-H. M. Kamal, S. M. M. Hasan, A. Kabir, D. Yeasmin, M. A. Islam, K. I. A. Chowdhury, K. S. Anwar, A. A. Chughtai, and H. Seale, "Covid-19-related infodemic and its impact on public health: A global social media analysis," *The American Journal of Tropical Medicine and Hygiene*, vol. 103, no. 4, pp. 1621 – 1629, 2020. [Online]. Available: <https://www.ajtmh.org/view/journals/tpmd/103/4/article-p1621.xml>
- [19] "Fake-news," 2015. [Online]. Available: <https://www.kaggle.com/c/fake-news/data>
- [20] M. Honnibal and I. Montani, "spacy 2: Natural language understanding with bloom embeddings, convolutional neural networks and incremental parsing," *To appear*, vol. 7, no. 1, pp. 411–420, 2017.
- [21] E. Loper and S. Bird, "Nltk: The natural language toolkit," in *Proceedings of the ACL-02 Workshop on Effective Tools and Methodologies for Teaching Natural Language Processing and Computational Linguistics - Volume 1*, ser. ETMTNLP '02, 2002, p. 63–70. [Online]. Available: <https://doi.org/10.3115/1118108.1118117>
- [22] B. Amey, B. Karan, G. Aman, and S. Shivam, "Text2emotion python library," <https://github.com/aman2656/text2emotion-library>, 2020.
- [23] Y. S. Shen, "Lexicalrichness python module," <https://github.com/LSYS/LexicalRichness>, 2019.
- [24] S. Loria, P. Keen, M. Honnibal, R. Yankovsky, D. Karesh, E. Dempsey et al., "Textblob: simplified text processing," *Secondary TextBlob: simplified text processing*, vol. 3, 2014.
- [25] C. Manning, M. Surdeanu, J. Bauer, J. Finkel, S. Bethard, and D. McClosky, "The Stanford CoreNLP natural language processing toolkit," in *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*. Baltimore, Maryland: Association for Computational Linguistics, Jun. 2014, pp. 55–60. [Online]. Available: <https://aclanthology.org/P14-5010>
- [26] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *the Journal of machine Learning research*, vol. 3, pp. 993–1022, 2003.
- [27] C. X. Ling, J. Huang, and H. Zhang, "Auc: A statistically consistent and more discriminating measure than accuracy," in *Proceedings of the 18th International Joint Conference on Artificial Intelligence*, ser. IJCAI'03. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2003, p. 519–524.