Proceedings of the OSACT 2022 Workshop @LREC2022, pages 12–22
Marseille, 20 June 2022

Detecting Users Prone to Spread Fake News on Arabic Twitter

Zien Sheikh Ali[1], **Abdulaziz Al-Ali**[1,2], **Tamer Elsayed**[1]
[1]Computer Science and Engineering Department, Qatar University
[2]KINDI Center for Computing Research, Qatar University
{zs1407404, a.alali, telsayed}@qu.edu.qa

Abstract

The spread of misinformation has become a major concern to our society, and social media is one of its main culprits. Evidently, health misinformation related to vaccinations has slowed down global efforts to fight the COVID-19 pandemic. Studies have shown that fake news spreads substantially faster than real news on social media networks. One way to limit this fast dissemination is by assessing information sources in a semi-automatic way. To this end, we aim to identify users who are prone to spread fake news in Arabic Twitter. Such users play an important role in spreading misinformation and identifying them has the potential to control the spread. We construct an Arabic dataset on Twitter users, which consists of 1,546 users, of which 541 are prone to spread fake news (based on our definition). We use features extracted from users' recent tweets, e.g., linguistic, statistical, and profile features, to predict whether they are prone to spread fake news or not. To tackle the classification task, multiple learning models are employed and evaluated. Empirical results reveal promising detection performance, where an F1 score of 0.73 was achieved by the logistic regression model. Moreover, when tested on a benchmark English dataset, our approach has outperformed the current state-of-the-art for this task.

Keywords: Misinformation, Social Media, Source Credibility, Fake News

1. Introduction

Twitter has evolved into a popular social media platform for news sharing. It allows tweets to reach a larger audience quickly through retweets and likes. The platform is commonly used by news outlets, governments, and public figures to communicate the latest news in a brief manner and engage with their followers (Vosoughi et al., 2018). While Twitter can be an effective tool to express thoughts and engage with authorities and organizations, it is also misused to generate fabricated information and occasionally manipulate the public opinion.

Misinformation can spread faster, deeper and wider in social networks compared to traditional media sources (Vosoughi et al., 2018). This wide spread of misinformation causes a serious impact on society and individuals. In the past few years, Arabic social media has been utilized to spread state propaganda, attack political parties, and mislead the society (Jones, 2019). Moreover, with the recent COVID-19 outbreak, health related misinformation has proliferated on Arabic social media (Jones, 2020). Spreading anti-vaccine misinformation has contributed in large public hesitancy, which is now hindering the national global efforts to fight the pandemic. Misinformation nowadays is not only used as a political weapon, but it also poses a serious risk to society and public health.

Previous studies have targeted misinformation on Arabic social media from a content-based perspective, by verifying the content of a single post or a tweet (El Ballouli et al., 2017; Nakov et al., 2021; Harrag and Djahli, 2022; Haouari et al., 2021a). However, only a few studies explored this task from a source-based perspective. The spread of misinformation can be effectively mitigated by identifying the credibility of the source of the information (Shu et al., 2020). In social media, users are contributing to the spread of fake news by retweeting and engaging with the information. It was found by Shao et al. (2018) that fake news tends to attract both malicious and normal users. The goal of malicious users is to achieve personal benefits, while normal users often spread misinformation unintentionally. Contrary to previous studies that target malicious users that intentionally spread misinformation (e.g., bots (Yang et al., 2020) and trolls (Mihaylov et al., 2015)), our work is concerned with users that are prone to spread fake news. We define them as *users that contribute in the diffusion and amplification of misinformation on Twitter, either intentionally or unintentionally*. Recognizing that type of users on Twitter is an important task that can be employed to combat the spread of fake news. For example, a tool to identify fake news spreaders can be an explicit addition to fake news detection systems.

In this paper, we aim to identify users prone to spread fake news on Arabic Twitter. Our objective is to classify a user as either prone to spread fake news, or not. Due to the lack of Arabic datasets for this task, we proposed a data collection pipeline to collect claims, tweets, and users for this task. We explored a range of different features extracted from the user timeline, such as textual, profile, statistical, and emotional features. Finally, we evaluated the performance of multiple learning models on our Arabic dataset, as well as publicly available English benchmark dataset.

The contributions of this paper are three-fold:

- We propose a method for constructing a user dataset using a set of previously-verified claims.

- We propose the first model to detect users prone to spread Arabic fake news.
- We made the source code and features used in our experiments publicly available for reproducibility and further research.[1]

The rest of the paper is organized as follows. Section 2 summarizes related work. Section 3 describes our user data collection process. Section 4 outlines our methodology. Section 5 shows our experimental evaluation and results, and Section 7 concludes.

2. Related work

In this section, we review the literature for related work on the task of detecting misinformation spreaders on social media. We discuss efforts on profiling users that spread misinformation (Section 2.1). Second, we review the different approaches that collect datasets of Twitter users to identify their role in spreading fake news (Section 2.2).

2.1. Classifying Misinformation Spreaders on Social Media

The task of classifying misinformation spreaders remains under-explored. Recent attention has focused on identifying social media users that spread fake news. Rangel et al. organized an Author Profiling Shared task at CLEF 2020 (Rangel et al., 2020). The task is defined as follows, given a Twitter user's recent tweets, determine whether they are keen to spread fake news or not. The authors provide a corpus of Twitter users and their recent 100 tweets. The languages covered are English and Spanish only. The task received 66 participants, and the highest Accuracy scores achieved were 75% on the English dataset and 82% on the Spanish dataset. It is worth mentioning that the highest performance was achieved using a stacked ensemble classifier of five machine learning algorithms; four of the base models use character n-grams as features, while the fifth model uses features based on statistics of the tweets such as the average length of the tweets (Buda and Bolonyai, 2020). All of the highest six participants in the task used a combination of n-grams and traditional machine learning approaches.

Rath et al. (2020) proposed a fake news spreader detection model using an inductive representation framework. Given a tweet and a directed social network, users that are more likely to spread misinformation are identified. They built a social graph of twitter users and defined modular communities using Community Health Assessment (CHA) model. Their approach identifies fake news spreaders based on a given tweet, while our approach identifies users independently.

Shu et al. (2019) investigated the role of user profiles for fake news detection. Their experiments show that user features such as registration time, account verification, political bias and personality type could make a significant impact in detecting fake news.

2.2. Annotated User Datasets

Current Twitter fake news datasets are catered for verification of tweets (tweet-level verification) (Haouari et al., 2021a). Limited datasets identify the role of users in the spread of fake news. We summarize the different approaches to collect Twitter users next.

Rangel et al. (2020) constructed a corpus of 500 English users and 500 Spanish for PAN 2020 shared task to detect users keen to spread fake news on Twitter. The corpus was constructed as follows: first, false claims debunked by fact-checking websites (e.g. PolitiFact and Snopes) are collected. Then, Twitter is searched to find tweets relevant to these claims, where the tweets are labeled as supporting a claim or not. After annotating the tweets, the users are labeled as keen to spread fake news or not based on whether they shared at least one tweet supporting a fake claim. Finally, users with the most annotated tweets were selected.

Labelling users based on annotated tweets was similarly adopted by Shao et al. (2018), where users who are super-spreaders are identified as users that continuously spread misinformation. Another contribution by Shu et al. (2019) used verified tweets from FakeNewsNet dataset (Shu et al., 2018) to label users as likely to spread fake-news or likely to spread real-news.

The studies presented thus far provide solutions for profiling users who try to spread misinformation. There is however insufficient research on addressing users spreading misinformation on Arabic Twitter. To fill this gap, our study focuses on identifying users that are prone to spread Arabic fake news. While previous work has focused on misinformation datasets for the task of tweet-level verification, very few studies worked on constructing datasets for user-level verification.

3. Data Collection

In this section, we describe the user data collection and annotation methodology. Our goal is to collect a set of users that are prone to spread fake news, and users that are not prone to spread fake news. To build the dataset, we modified the method used in the shared task at PAN 2020 for profiling fake news spreaders on Twitter, as described in Section 2.2. We constructed the dataset in three main stages. First, we collected sets of previously-verified Arabic claims from multiple resources. We then used those claims to find tweets that are spreading them. Finally, we identify users associated with those tweets and label them based on tweet frequencies. These stages are detailed in the next three subsections.

3.1. Claim Collection

In this stage, we aim to collect real claims from the Arab world and then search for tweets that are spread-

[1] https://gitlab.com/bigirqu/ArPFN

Figure 1: Example of a tweet obtained from AraFacts. The claim translates to "Dr. Mohammed Mashali has passed away." However, the tweet is only questioning whether the claim is true or not.

ing them. To do so, we leveraged two existing Arabic rumor datasets, namely, ArCOV19-Rumors (Haouari et al., 2021b), and AraFacts (Sheikh Ali et al., 2021).

ArCOV19-Rumors covers claims related to COVID-19 from multiple topical categories such as social, political, sports and, entertainment. The dataset contains 138 verified claims from fact-checking websites, and 9,414 tweets relevant to those claims.

AraFacts is the first large collection of Arabic naturally-occurring claims from 5 different Arabic fact-checking websites. The claims are annotated and verified by professional fact-checkers. It contains 6,222 claims that were posted between 2016 and 2021. Claims are crawled from each fact-checking website along with their factual label, description, and 10 additional meta-data. We selected claims from AraFacts that have the labels *True* and *False* only. Overall, we have collected 5,371 claims from both datasets with 299 of them being True and 5,072 being False.

3.2. Tweet Collection

After collecting the claims, the next step is to find tweets that are relevant to them. We utilized the manually-annotated tweets from ArCOV19-Rumors dataset, where only tweets labeled as *True* or *False* were kept, and the rest were discarded, resulting in 3,025 tweets.

In the AraFacts dataset, we used the claim URLs data field, which contains URLs to Web pages that spread each claim. We identified URLs pointing to tweets and obtained their tweet IDs. The tweets were then crawled using the Twitter API yielding 2,981 tweets that are related to 1,213 claims.

After collecting the tweets from AraFacts, we manually inspected a subset of 100 tweets to verify that the tweets are indeed relevant to their corresponding claims. Surprisingly, some of the tweets were not associated with their claims, or not expressing them. Figure 1 shows one such example. Arguably, a user that is questioning the correctness of a claim is neutral towards it and not spreading it. Out of the 100 tweets that we inspected, 9 were found to be irrelevant to their claims. This has prompted us to manually annotate all tweet-claim pairs to verify their relevancy to the claim. The annotation task was performed by one annotator who was asked to read the tweet and the claim, then label the tweet as: *Expressing the claim*, *Negating the claim* or *Other*. The detailed annotation guidelines can be found in Appendix 8.

The results of the annotation task are presented in Table 1. Evidently, 95% of the tweet-claim pairs were labeled correctly by the fact-checkers and 4.5% tweets were labeled as *Other*; meaning they are not relevant or not spreading the claim. We unexpectedly identified 7 tweets negating the claim which could have been added erroneously by the fact-checkers.

Annotation	Number of tweets
Expressing the claim	2,474
Other	125
Negating the claim	7

Table 1: Results for tweet-claim annotation task.

Once annotation was complete, we eliminated the tweets that are labeled as *Other* and changed the label of tweets that are labeled as *Negating* (i.e., a tweet that negates a True claim is labeled False and vice versa). Finally, we collected retweets of all the verified tweets from AraFacts using Twitter API.[2] Unlike AraFacts, the retweets for ArCOV19-Rumors are publicly available.[3] The total number of collected retweets is **35,698**.

3.3. User Collection

Since our annotated tweet collection is limited to only ArCOV19-Rumors tweets and a small subset of AraFacts claims (only 1,213 claims have annotated tweets), this step aims to capture more associated claims to each user by searching the users' timelines for occurrences of other claims from our collection.

We started by using the collected tweets to identify unique users with at least 1 tweet in ArCOV19-Rumors or AraFacts. Consequently, 4,176 unique users were found. For each user, we used Twitter API to collect their timelines. The maximum number of tweets that can be crawled per user is 3,200 tweets.

We then searched the users' timelines for claims using all 5,371 claims from our collection. For each user timeline, we used the ElasticSearch engine[4] to retrieve tweets that have high similarity with the claims' text or description. The retrieved tweets, with BM25 similarity score above 15, were manually annotated using the same annotation guidelines mentioned in Section 3.2, and then appended to the tweet collection.

Table 2 summarizes our tweet collection statistics. We also visualize our collection of verified tweets (tweets

[2] https://developer.twitter.com/en/docs/twitter-api/tweets/retweets/introduction

[3] https://gitlab.com/bigirqu/ArCOV-19/-/blob/master/ArCOV19-Rumors/tweet_verification

[4] https://www.elastic.co/elasticsearch/

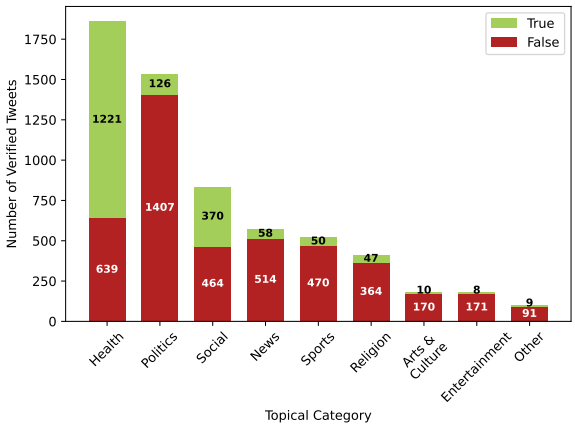

Figure 2: Distribution of the verified tweets and their topical categories and veracity labels.

related to verified claims) in Figure 2 by demonstrating all 9 topical categories and their label distributions. Notably, the majority of tweets are related to Health or Politics and most Fake claims are political.

Tweet Source	True Tweets	False tweets
ArCOV19-Rumors	1,625	1,948
AraFacts	191	2,431
Manually annotated tweets	133	114
All	1,949	4,493

Table 2: Summary of the verified tweets collection.

Next, we investigate the user collection after adding the newly annotated tweets. We count the number of users in terms of their number of verified tweets or retweets. Table 3 summarizes our user collection.

Table 3: Number of users in terms of the number of verified tweets and retweets (RTs) they shared.

Users that shared ...	X			
	1	2	3	4
at least X true tweets or RTs and 0 false tweets or RTs	**1,005**	204	71	35
at least X false tweets or RTs	3,171	**541**	166	73

To construct our final *labeled* user dataset, we identify users as **prone to spread fake news** if they *shared at least two false tweets or retweets*. On the other hand, users are considered **not prone to spread fake news** if they *shared at least one true tweet or retweet and have no record of spreading false tweets or retweets*. The assumption is that users associated with frequent false tweets or retweets are more likely to be prone to spread fake news than others. Although we choose a threshold of *two* false tweets or retweets for users prone to spread fake news, this threshold can be adjusted by the practitioner to suit the task at hand. The threshold for users not prone to spread fake news was set to at least one true tweet. Admittedly, this criterion may introduce noise to this class, as we do not have enough evidence that those users did not spread any fake news that are not included in our verified set of claims.

4. Methodology

In this section, we describe our features and models used to automatically identify users that are prone to spread fake news on Twitter.

4.1. Feature Extraction

For each user, we obtain recent tweets and user's metadata using Twitter API. Features that capture information about the user's activity, popularity, and linguistic style are extracted. These features can be classified into the following five main categories:

4.1.1. Textual Features

To obtain textual features, the user's recent tweets are first concatenated as one "document". We then performed light pre-processing on the text. In particular, we removed all non-alphanumeric characters, replaced URLs or media links with #URL# and #MEDIA#, and used tashaphyne library[5] to clean the text by removing any figuration and normalizing elongated words.
From each user's document, we derived tf-idf word n-grams, and eliminated words that appear in less than 50 documents (across all users). Additionally, we tested multiple n-gram ranges (unigrams, bigrams, and unigrams and bigrams) as a hyper-parameter for each trained model. Using n-grams as textual features was proven to be effective in PAN author profiling task (Rangel et al., 2020).

4.1.2. Contextualized Embeddings

We used contextualized embeddings to represent each user's recent 100 tweets. The use of contextualized embeddings as features is motivated by the work of An et al. (2021) to predict hateful users on Twitter. They obtain a user-level representation by computing Sentence-BERT (S-BERT) (Reimers and Gurevych, 2019) embeddings for each user's tweet then averaging all tweet embeddings into one 768-dimensional vector. In our experiments, we used transformer models to generate embeddings, namely, the different variations of Bidirectional Encoder Representations from Transformers (BERT) to compute embeddings. Three different BERT-based models that support Arabic were tested: AraBERT (Antoun et al., 2020), MARBERT (Abdul-Mageed et al., 2021), and S-BERT.

[5] https://github.com/linuxscout/tashaphyne/

4.1.3. Statistical Features

This category of features is derived from users' recent 3,200 tweets. We used features that describe the user impact, motivated by the work proposed by Lampos et al. (2014), in addition to timeline features that describe the user's activities. The proposed statistical features are listed below. The last three features are newly proposed in this work.

- Proportion of tweets with hashtags.
- Average number of hashtags per tweet.
- Proportion of tweets with mentions.
- Number of unique mentions in user's timeline.
- Proportion of tweets that are replies to other users.
- Proportion of tweets that contain URLs.
- Proportion of tweets that contain media, e.g., images or videos.
- Proportion of tweets that are retweets.
- Proportion of tweets that are quote retweets.
- Average engagement of the user, computed as the average number of retweets and likes per tweet.
- Average number of days between each two consecutive tweets.

4.1.4. Profile Features

For each user, we used some meta-data from the user's JSON object as features and derived 10 additional features related to the user. The features used have been implemented in previous studies that profile users (Shu et al., 2019; Yang et al., 2020; Castillo et al., 2011). Table 9 summarizes the extracted profile features, their type, and description.

4.1.5. Emotional Features

Several researchers have utilized emotional signals for credibility assessment (Ghanem et al., 2020; Zhang et al., 2021). Moreover, multiple participants in PAN author profiling task (to detect users keen to spread fake news) used emotional signals to address the task (Rangel et al., 2020; Fersini et al., 2020; Moreno-Sandoval et al., 2020). We similarly extracted emotional signals from the text of each user's recent 100 tweets. For the Arabic experiments, we used the emotion functionality in ASAD tool (Hassan et al., 2021). The extracted 11 features are: anger, anticipation, disgust, fear, joy, love, optimism, pessimism, sadness, surprise, and trust. For the English experiments, we used NRC emotion Lexicon.[6] Specifically, we used the python library NRCLex[7] to retrieve raw emotions count given a text. The extracted features include eight basic emotions (anger, fear, anticipation, trust, surprise, sadness, joy, and disgust) and two sentiments (positive and negative).

[6] https://saifmohammad.com/WebPages/NRC-Emotion-Lexicon.htm

[7] https://pypi.org/project/NRCLex/

4.2. Models

We trained multiple machine learning models over a combination of the features described in Section 4.1. The models we experiment with are known to achieve high performance in different text classification tasks (Shwartz-Ziv and Armon, 2022; Islam et al., 2019), namely, XGBoost (HGB), Random Forests (RF), Logistic Regression (LR), and Feed-forward Neural Networks (NN).

5. Experimental Evaluation

In this section, we conduct experiments to answer the following research questions:

RQ1 How effective are traditional machine learning methods in automatically detecting users that are prone to spread fake news on Arabic Twitter?

RQ1.1 How effective are the existing baselines for the task?

RQ1.2 Which feature category combination exhibits the best performance?

RQ1.3 How does the classifier perform when contextualized embeddings are used instead of word n-grams?

RQ2 What is the effect of increasing the number of user tweets, considered for feature extraction, on the performance of the classifier?

RQ3 How effective is our methodology on an English dataset?

5.1. Experimental Setup

5.1.1. Datasets

The datasets used in our experiments are listed below and summarized in table 4.

- Arabic Dataset (**ArPFN**): This is our Arabic user dataset described in Section 3.3.
- English Dataset (**EN_PAN**): We use the English dataset constructed for PAN author profiling task to predict users keen to spread fake news.[8] The provided dataset consists of hashed user ids, the text of their recent 100 tweets, and the user label.

Dataset	PFN	NPFN	Total Users
ArPFN	541	1,005	1,546
EN_PAN	250	250	500

Table 4: Datasets used in our experiments. PFN/NPFN denotes the number of users that are prone/not prone to spread fake news.

[8] https://zenodo.org/record/4039435#.YlV0g-hBw2x

5.1.2. Training and Evaluation Measures

We evaluated our models using Positive-F_1 (F_1^+) score, where users prone to spread fake news constitute the positive class. We additionally report Macro-F_1 score.
Experiments on **ArPFN** were performed using nested 10-fold cross validation to tune the hyper-parameters of model. For that, we optimized for F_1^+ score. Since the dataset is imbalanced and the positive class is the minority, we over-sampled the positive class in training folds only. The reported results on **ArPFN** are the average over the 10-folds used in cross validation.
For the experiments on **EN_PAN**, we used the same data splits provided PAN for easy comparisons. Additionally, we evaluated our models using 10-fold cross validation to be able to perform significance tests.
For statistical significance tests, we performed two-tailed paired t-test on F_1^+ score, using the scores over the 10 folds, with a 5% significance level.

5.1.3. Baselines

We compare the performance of our models against the following baselines:

1. **Majority:** A classifier that always predicts the label of the majority class.
2. **PAN_2020**: The winning participation at PAN author profiling task (Buda and Bolonyai, 2020). They proposed an ensemble of five machine learning models. They replaced the typical majority voting with a logistic regression classifier that takes the outputs of the ensemble models as the input vector. The first four models (Logistic Regression, Support Vector Machine, Random Forest and XGBoost) use word n-grams as features, while the fifth model (XGBoost) uses statistical features. All features are derived from the user's recent 100 tweets only. We used the authors' implementation.[9]
3. **PAN_2020+**: An improved version of **PAN_2020** that we proposed. First, we eliminated the XGBoost model from the ensemble, as it was shown by Buda and Bolonyai (2020) that it has the least impact on the performance as per the Logistic Regression coefficients. Additionally, for the remaining models that use only tf-idf as features, we expand the feature vector by including emotional signals. We trained four models individually with the same feature vector of word n-grams and emotions, then we stack the four models into a Logistic Regression ensemble as done in **PAN_2020**.

5.2. Classification of Users Prone to Spread Arabic Fake News (RQ1)

To address **RQ1**, we trained our baselines and individual models to predict if a user is prone to spread fake news or not. We tried four models: Random Forest (RF), XGBoost (XGB), Logistic Regression (LR), and Feed-forward Neural Networks (NN). We used the Arabic dataset **ArPFN** for this experiment, where the textual features are extracted from each user's recent 100 tweets.
Table 5 summarizes the performance of the baseline models. We note that **PAN_2020** and **PAN_2020+** clearly outperform majority baseline. Moreover, **PAN_2020+** performs slightly better than **PAN_2020**; however, the difference is not statistically significant. To answer **RQ1.1**, the baseline models identify users prone to spread fake news with a F_1^+ score of 0.63.

Model	F_1^+	Macro-F_1
Majority	0.00	0.40
PAN_2020	0.61±0.05	0.71
PAN_2020+	**0.63**±0.06	**0.73**

Table 5: Baseline performance on **ArPFN**.

Next, we perform an ablation study to evaluate the impact of different feature category combinations and find the best combination for this task. We tried the following combinations:

- Textual features only.
- Non-textual-based features (profile and statistical).
- Textual, profile, and statistical features.
- All feature categories.

Table 6 summarizes the results of these experiments. We performed significance tests to compare the performance of each combination with respect to **PAN_2020+**. We use the $*$ symbol to denote a statistically significant improvement over that baseline.
The results clearly show that training the models using textual features only produces similar (in case of XGB and NN) to or better results (in case of RF and LR) than the baseline model. Moreover, RF still yields even better performance with non-textual-based features than the baseline and also the other models. However, the improvements were not statistically significant.
Interestingly, we observe that when textual features are combined with profile and statistical features, XGB, LR, and NN models outperformed the baseline with statistically-significant improvements. Moreover, adding the emotional features (i.e., using all the feature categories) yield an even further improvement in both F_1^+ and Macro-F_1 for the XGB model.
In conclusion, to answer **RQ1.2**, combining textual and non-textual features yields better results in general. More specifically, the best achieved performance (F_1^+= 0.70) is obtained when the XGB classifier is trained on all feature categories.

[9] https://github.com/pan-webis-de/bolonyai20

Features	Model	F_1^+	Macro-F_1
-	**PAN_2020+**	0.63±0.06	0.73
Textual	RF	0.64±0.05	0.75
	XGB	0.63±0.05	0.74
	LR	0.65±0.05	0.75
	NN	0.63±0.05	0.74
Profile +Statistical	RF	0.65±0.05	0.76
	XGB	0.63±0.05	0.73
	LR	0.60±0.03	0.59
	NN	0.63±0.04	0.66
Textual +Profile +Statistical	RF	0.66±0.06	0.76
	XGB	0.68*±0.04	0.78
	LR	0.68*±0.04	0.77
	NN	0.67*±0.05	0.78
Textual +Profile +Statistical +Emotions	RF	0.67±0.05	0.77
	XGB	**0.70***±0.05	**0.79**
	LR	0.68*±0.04	0.76
	NN	0.64±0.06	0.75

Table 6: Performance on **ArPFN** with different feature category combinations. The asterisk (*) indicates statistically-significant improvement over the baseline model.

Lastly, we investigate the performance of the classifiers when contextualized embeddings are used as features instead of word n-grams. We used the 768-dimensional embeddings vector that represents the average of the embeddings of each user's tweet. We concatenate the embeddings vector to the profile, statistical, and emotional features. Figure 3 compares the F_1^+ score of using *different* embeddings (i.e., generated from different pre-trained language models) in training our four models. The figure also illustrates the performance of the models trained with all feature categories when the textual features are word n-grams (same scores as in Table 6) for the sake of comparison.

The figure shows that S-BERT embeddings yield the best performance among all other types of embeddings. However, the models trained on the embeddings are all outperformed by the models trained on the word n-grams. Answering **RQ1.3**, the replacement is then deemed ineffective, at least in the way we generated the embeddings vector as the average of the embeddings vectors of the individual user's tweets.

5.3. Effect of Considering Longer User's Timeline (RQ2)

We explore the effect of using more tweets from the user's timeline on classifying the users. Identifying the ideal number of tweets is important in time-sensitive applications, as it determines the number of requests using Twitter API, which allows the retrieval of 100 tweets per request with a rate limit of 900 requests within a 15-minute window.[10]

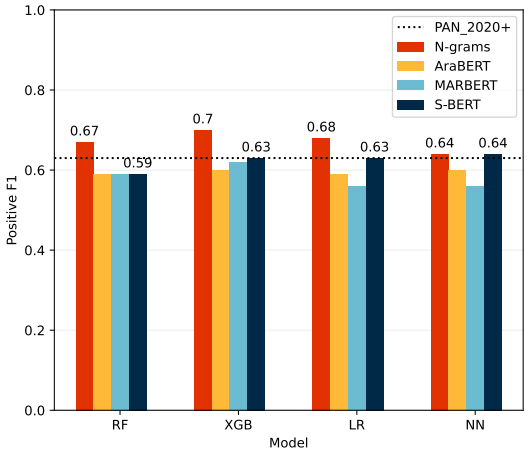

Figure 3: Performance of different models trained using mean-pooling BERT embeddings with profile, statistical, and emotional features on **ArPFN**.

We conduct experiments by gradually increasing the number of tweets per user and evaluating the performance of each model. We test the model performance on 100, 500, 2,000 and 3,200 user tweets. The experiments are conducted only on **ArPFN**, as **EN_PAN** is only limited to 100 tweets and the usernames are hashed so we were unable to expand it.

For these experiments, we chose the best models from Table 6, namely XGB and LR, trained on all features (with word n-grams). Figure 4 shows the performance after increasing the number of tweets for both models. The figure clearly shows that increasing the number of considered tweets of the timeline results in a monotonically-improving performance for both models. The most notable improvement (which is also statistically-significant) was achieved by the LR model whose performance jumped from F_1^+ score of 0.68 with 100 tweets to 0.73 with 3,200 tweets, yielding the highest performance in all of our experiments. Answering **RQ2**, considering more tweets in extracting the textual features yield better performance; however this requires more API requests, hence more time.

5.4. Performance on English (RQ3)

We aim to validate the effectiveness of our methodology by testing it on datasets of other languages. To this end, we used **EN_PAN** dataset to conduct our experiments on English. **EN_PAN** is limited to the text of the recent 100 tweets from each user, and the usernames were hashed to maintain their privacy. So, we were unable to extract all the features we described in Section 4.1. In this experiment, we compare the performance

[10] https://developer.twitter.com/en/docs/twitter-api/rate-limits

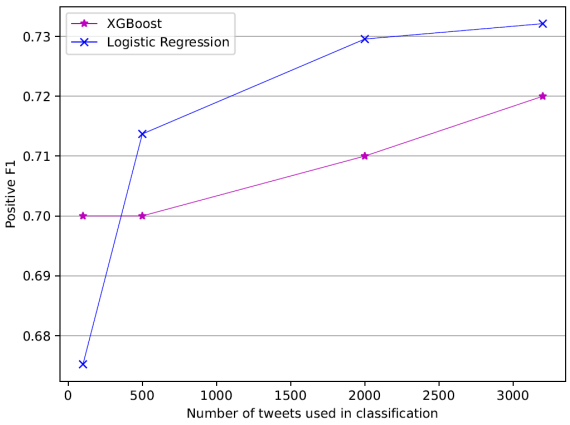

Figure 4: Performance on **ArPFN** after increasing the number of user's tweets for classification.

of the main baseline **PAN_2020** and the improved baseline **PAN_2020+**.

Table 7 reports the results of our experiment on the same PAN data splits. It is shown that our methodology of combining textual features with emotional signals has improved the F_1^+ scores by 5 points. To validate our results, we also perform 10-fold cross validation. The results of that setup are summarized in Table 8, showing that our improved baseline **PAN_2020+** outperforms **PAN_2020**, which constitute the current state-of-the-art. However, the improvement was not statistically-significant.

Model	F_1^+	Macro F_1
PAN_2020	0.74	0.73
PAN_2020+	0.79	0.77

Table 7: Performance on **EN_PAN** using PAN train-test splits.

Model	F_1^+	Macro F_1
PAN_2020	0.73±0.05	0.73
PAN_2020+	0.75±0.03	0.75

Table 8: Performance on **EN_PAN** using 10-fold cross validation.

6. ArPFN Dataset Release

To enable further research, we have made the extracted features of all 1,546 users in **ArPFN** publicly available. Additionally, we shared the folds used in our experiments to enable the reproducibility of our experimental results. To maintain the confidentiality of the users, and in accordance to Twitter content redistribution policy,[11] we do not share the text of the tweets.

[11] `https://developer.twitter.com/en/developer-terms/agreement-and-policy`

7. Conclusion

In this paper, we explored the task of identifying users who are prone to spread fake news in Arabic Twitter. While most related work on fake news detection systems focus on tweet verification, we instead explore the source of the tweet. We constructed the *first* Arabic users dataset **ArPFN** for this task by leveraging two Arabic misinformation datasets, ArCOV19-Rumors and AraFacts. We also proposed the *first* Arabic-specific classifier to identify users prone to spread fake news on Arabic Twitter. Our experiments showed that combining all feature categories yields the best classification performance. Moreover, we established that increasing the number of considered user tweets increases detection accuracy. The best model has achieved an average F_1^+ score of 0.73 using 10-fold cross validation on our Arabic dataset. We also showed that our method is effective even on English datasets, as it has outperformed the current state-of-the art and achieved an F_1^+ score of 0.79.

This study offers important insights on the subject of user credibility on Twitter, a topic that undoubtedly has ethical consequences. As a result, the use of any such prediction system to assess an individual's credibility must be done with caution. We would like to emphasize that the user labeling heuristic in this paper was established by taking the opinion of multiple individuals rather than one. Ultimately, the choice of heuristics to label users is subjective and may differ based on the use case of the target application.

Acknowledgements

This work was made possible by NPRP grant No.: NPRP11S-1204-170060 from the Qatar National Research Fund (a member of Qatar Foundation). The statements made herein are solely the responsibility of the authors.

8. Bibliographical References

Abdul-Mageed, M., Elmadany, A., et al. (2021). Arbert & marbert: Deep bidirectional transformers for arabic. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 7088–7105.

An, J., Kwak, H., Lee, C. S., Jun, B., and Ahn, Y.-Y. (2021). Predicting anti-asian hateful users on twitter during covid-19. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 4655–4666.

Antoun, W., Baly, F., and Hajj, H. (2020). Arabert: Transformer-based model for arabic language understanding. In *Proceedings of the 4th Workshop on Open-Source Arabic Corpora and Processing Tools, with a Shared Task on Offensive Language Detection*, pages 9–15.

Buda, J. and Bolonyai, F. (2020). An ensemble model using n-grams and statistical features to identify fake news spreaders on twitter. In *CLEF*.

Castillo, C., Mendoza, M., and Poblete, B. (2011). Information credibility on twitter. In *Proceedings of the 20th international conference on World wide web*, pages 675–684.

El Ballouli, R., El-Hajj, W., Ghandour, A., Elbassuoni, S., Hajj, H. M., and Shaban, K. B. (2017). Cat: Credibility analysis of arabic content on twitter. In *WANLP@ EACL*, pages 62–71.

Fersini, E., Armanini, J., and D'Intorni, M. (2020). Profiling fake news spreaders: Stylometry, personality, emotions and embeddings. In *CLEF (Working Notes)*.

Ghanem, B., Rosso, P., and Rangel, F. (2020). An emotional analysis of false information in social media and news articles. *ACM Transactions on Internet Technology (TOIT)*, 20(2):1–18.

Haouari, F., Hasanain, M., Suwaileh, R., and Elsayed, T. (2021a). ArCOV-19: The First Arabic COVID-19 Twitter Dataset with Propagation Networks. In *Proceedings of the Sixth Arabic Natural Language Processing Workshop*, pages 82–91.

Haouari, F., Hasanain, M., Suwaileh, R., and Elsayed, T. (2021b). ArCOV19-Rumors: Arabic COVID-19 Twitter Dataset for Misinformation Detection. In *Proceedings of the Sixth Arabic Natural Language Processing Workshop*, pages 72–81.

Harrag, F. and Djahli, M. K. (2022). Arabic fake news detection: A fact checking based deep learning approach. *Transactions on Asian and Low-Resource Language Information Processing*, 21(4):1–34.

Hassan, S., Mubarak, H., Abdelali, A., and Darwish, K. (2021). Asad: Arabic social media analytics and understanding. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*, pages 113–118.

Islam, M. Z., Liu, J., Li, J., Liu, L., and Kang, W. (2019). A semantics aware random forest for text classification. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, pages 1061–1070.

Jones, M. O. (2019). The gulf information war—propaganda, fake news, and fake trends: The weaponization of twitter bots in the gulf crisis. *International journal of communication*, 13:27.

Jones, M. O. (2020). Disinformation superspreaders: the weaponisation of covid-19 fake news in the persian gulf and beyond. *Global Discourse: An interdisciplinary journal of current affairs*, 10(4):431–437.

Lampos, V., Aletras, N., Preoţiuc-Pietro, D., and Cohn, T. (2014). Predicting and characterising user impact on twitter. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 405–413.

Mihaylov, T., Georgiev, G., and Nakov, P. (2015). Finding opinion manipulation trolls in news community forums. In *Proceedings of the nineteenth conference on computational natural language learning*, pages 310–314.

Moreno-Sandoval, L. G., Del Puertas, E. A. P., Quimbaya, A. P., and Alvarado-Valencia, J. A. (2020). Assembly of polarity, emotion and user statistics for detection of fake profiles. In *CLEF (Working Notes)*.

Nakov, P., Da San Martino, G., Elsayed, T., Barrón-Cedeño, A., Míguez, R., Shaar, S., Alam, F., Haouari, F., Hasanain, M., Mansour, W., et al. (2021). Overview of the clef–2021 checkthat! lab on detecting check-worthy claims, previously fact-checked claims, and fake news. In *International Conference of the Cross-Language Evaluation Forum for European Languages*, pages 264–291. Springer.

Rangel, F., Giachanou, A., Ghanem, B., and Rosso, P. (2020). Overview of the 8th author profiling task at pan 2020: Profiling fake news spreaders on twitter. In *CLEF*.

Rath, B., Salecha, A., and Srivastava, J. (2020). Detecting fake news spreaders in social networks using inductive representation learning. In *2020 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, pages 182–189. IEEE.

Reimers, N. and Gurevych, I. (2019). Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992.

Shao, C., Ciampaglia, G. L., Varol, O., Yang, K.-C., Flammini, A., and Menczer, F. (2018). The spread of low-credibility content by social bots. *Nature communications*, 9(1):1–9.

Sheikh Ali, Z., Mansour, W., Elsayed, T., and Al-Ali, A. (2021). AraFacts: The First Large Arabic Dataset of Naturally-Occurring Professionally-Verified Claims. In *Proceedings of the Sixth Arabic Natural Language Processing Workshop*. Association for Computational Linguistics.

Shu, K., Mahudeswaran, D., Wang, S., Lee, D., and Liu, H. (2018). FakeNewsNet: A data repository with news content, social context and spatialtemporal information for studying fake news on social media. *arXiv preprint arXiv:1809.01286*.

Shu, K., Zhou, X., Wang, S., Zafarani, R., and Liu, H. (2019). The role of user profiles for fake news detection. In *Proceedings of the 2019 IEEE/ACM international conference on advances in social networks analysis and mining*, pages 436–439.

Shu, K., Bhattacharjee, A., Alatawi, F., Nazer, T. H., Ding, K., Karami, M., and Liu, H. (2020). Combating disinformation in a social media age. *Wiley*

Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 10(6):e1385.

Shwartz-Ziv, R. and Armon, A. (2022). Tabular data: Deep learning is not all you need. *Information Fusion*, 81:84–90.

Vosoughi, S., Roy, D., and Aral, S. (2018). The spread of true and false news online. *Science*, 359(6380):1146–1151.

Yang, K.-C., Varol, O., Hui, P.-M., and Menczer, F. (2020). Scalable and generalizable social bot detection through data selection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 1096–1103.

Zhang, X., Cao, J., Li, X., Sheng, Q., Zhong, L., and Shu, K. (2021). Mining dual emotion for fake news detection. In *Proceedings of the Web Conference 2021*, pages 3465–3476.

Appendix

The annotation guidelines used to annotate the AraFacts tweets are described below:
Given a claim X, label the tweet T as:

- **Expressing the same claim**: if the author of the tweet is sharing, restating, or rephrasing the same claim X. In other words, the author is believing the claim and participating in sharing it. (i.e., T = X).
- **Negating the same claim**: if the author of the tweet is disagreeing or denying the claim. In other words, the author is debunking the claim and stating that it is incorrect. (i.e., C = not X).
- **Other**: if it is not one of the above, for example:
  - Author of the tweet is sharing the claim and questioning whether it is true or fake
  - Author of the tweet is sharing multiple claims including the main claim
  - The tweet is referring to a deleted image or video and the text of the tweet is insufficient to annotate the claim

Annotation steps:

1. Read claim text.
2. Read the tweet text.
3. Determine if tweet is expressing the claim, negating the claim or neither.

Notes:

- If the claim is related to an image or video, we recommended to check the URL of the claim and the URL of the tweet to compare if both links refer to the same image or video.
- We recommend considering the claim publication date and tweet posting date into considerations. If the tweet is posted after the claim has been verified, make sure that the tweet is still relevant to the same claim and that the claim is still holding the same label when it was verified.

Table 9: Profile features extracted from users' profiles. Features marked with $\star$ are the 10 features derived using fields from the User's JSON meta-data, while the remaining features are fields from the user's JSON object without modifications.

Feature	Type	Description
default_profile	Boolean	If the user has changed the default theme or background of their profile or not.
verified	Boolean	If the user has a verified account or not.
followers_count	Integer	Number of followers the account has.
following_count	Integer	Number of users that the account is following.
favourites_count	Integer	Number of tweets that were liked by the user.
listed_count	Integer	Number of lists the user has been added to.
statuses_count	Integer	Number of tweets posted by the user.
tweet_frequency$\star$	Float	Frequency of the users tweets, calculated as *tweets_count* divided by the account age in months.
follower_ growth_rate$\star$	Float	Rate of followers growth, calculated as *followers_count* divided by the account age in months.
following_ growth_rate$\star$	Float	Rate of followings growth, calculated as *following_count* divided by the account age in months.
listed_ growth_rate$\star$	Float	Rate of lists growth, calculated as *lists_count* divided by the account age in months.
followers_ following_ratio$\star$	Float	Number of followers compared to the number of following
screen_name_length$\star$	Integer	Number of characters in the users screen name.
digits_in_screen_name$\star$	Integer	Number of digits in the users screen name
name_length$\star$	Integer	Number of characters in the name of the user.
digits_in_name$\star$	Integer	Number of numerical digits in the name of the user.
description_length $\star$	Integer	Number of characters in the user's description (biography).