IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 31 August 2022, accepted 6 October 2022, date of publication 25 October 2022, date of current version 2 November 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3216891

RESEARCH ARTICLE

Push-To-Trend: A Novel Framework to Detect Trend Promoters in Trending Hashtags

SOUFIA KAUSAR, BILAL TAHIR, AND MUHAMMAD AMIR MEHMOOD
Al-Khawarizmi Institute of Computer Science, University of Engineering and Technology, Lahore, Lahore 54890, Pakistan

Corresponding author: Bilal Tahir (bilal.tahir@kics.edu.pk)

This work was supported in part by the Higher Education Commission (HEC) Pakistan, and in part by the Ministry of Planning Development and Reforms under National Center in Big Data and Cloud Computing.

ABSTRACT Twitter trends have enabled the speedy dissemination of information with the ability to affect public opinion. Unfortunately, fake trends are also generated by malicious users to mislead the public. In general, Twitter users are studied in depth to identify humans, bots, spam, and fake accounts. However, artificial intelligence algorithms are not developed for the identification of 'trend promoters' generating fake trends. In this paper, we propose **Push-To-Trend** – a novel framework to detect 'trend promoters' in trending hashtags. For this purpose, first, we develop a dataset of TREP-21 containing 3, 900 users labelled into two categories of 'trend promoters' and 'normal users'. In addition, we design four discerning features of number of total tweets, duplicate tweets, overlapping ngram, and peak-to-mean ratio for trend promoters classification. Moreover, we thoroughly examine the features used for spam and bot accounts classification to filter three efficacious features for trend promoters identification. Leveraging these seven features, Push-To-Trend achieves the accuracy of 0.97 for TREP-21. Furthermore, we leverage our framework to identify and analyze trend promoters from the Urdu tweets repository ''Anbar'' which consists of 106.9 million tweets and 1.69 million users. The analysis of 602 most frequent hashtags in Anbar reveals that 15.7% of trend promoters generate 68.1% of total tweets related to hashtags. To the best of our knowledge, this is the first attempt to design machine learning models for the automatic classification of trend promoters. As such, our framework is generic and adaptable for tweets posted in different natural languages as it utilizes language-independent features.

INDEX TERMS Twitter trends, trend promoters, social media user classification, Twitter analytics.

I. INTRODUCTION

Trending panels of social media platforms have emerged as a vital source of information dissemination and real-world events monitoring. These trends are constantly monitored by journalists, media reporters, news and government organizations for latest news and events [1], [2]. However, these trends are not secure from manipulation and users generate fake trends for socio-economic benefits [3]. One major reason for creating fake trends is that trending hashtags/topics exhibit the domino effect after reaching the trending panel and gain the visibility of a large audience. In addition, these users have created networks to perform malicious activities in coordination with pre-defined goals such as user opinion manipulation and spreading political/social disinformation [4], [5]. This amount of attention and utilization of the trending panel demands the inspection of trending hashtags to confine the reach of malicious content.

A recent study unveils that 47% of Twitter trends in Turkey and 20% of global trends are generated by manipulation [6]. In addition, online advertisers earned more revenue by generating fake trends on the Twitter trending panel than Twitter itself in India [7]. Similarly, financially and politically motivated users perform coordinated efforts to trend their target hashtag by artificially increasing Twitter traffic [4]. Due to such economical benefits, a successful business model by different companies has emerged which offers services to create fake trends [8]. Prompted by the devastating impact of

The associate editor coordinating the review of this manuscript and approving it for publication was Juan Wang.

fake trends, Twitter has declared the usage of automated, multiple, or fake accounts to generate fake trends illegal. Twitter deactivates such accounts temporarily or permanently [9].

The analysis of malicious users discloses that human, cyborg, or bot accounts are used to manipulate the trending panel [4]. Activities of different types of users on the Twitter platform motivate the research community to design algorithms for the automatic examination of user behavior and Twitter trends. For instance, algorithms are introduced for the identification of trending topics and their classification into different categories [15], [16], [17]. In addition, Twitter content along with user behavior is examined to identify users such as bots [18], [19], [20], [21], spam users [22], [23], and fake accounts [24], [25]. Also, Twitter trending hashtags and topics are studied to understand traffic manipulation [4], political manipulation [26], astroturfing attacks [6], and role of bot accounts in manipulation [27]. Table 1 summarizes different types of users using the Twitter platform considered in literature. We notice two research gaps in the existing literature considering the Twitter accounts involved in trend manipulation. First, the classification of users into bots, humans, fake, spam, etc., did not explicitly identify the accounts performing manipulation. As such, trend manipulation is performed by bots as well as human accounts. Second, in literature, the focus of studies is to understand the manipulation of trends, not the users performing trend manipulation.

In this paper, we introduce '**Push-To-Trend**'– a novel framework for the automatic detection of trend promoters. Specifically, we are interested in detecting users who generate or promote Twitter trends. Therefore, we will use the term *trend promoters* for such users in our paper. For this purpose, first, we develop a labelled dataset of TREnd Promoters (TREP-21) containing 3, 900 users labelled as 'normal users' and 'trend promoters'. Next, we propose four features: i) tweet count, ii) duplicate tweet, iii) peak-to-mean ratio of tweets by a user, and iv) overlapping n-grams of tweets text to train machine learning models for the classification of users. We also perform experiments using correlation analysis and Recursive Feature Elimination (RFE) with features proposed for identification of bots and spam accounts to find the features suitable for classification of trends promoters. This process results in the selection of three valuable features i.e., count of retweets, hashtags, and intermediate time between tweets posted by a user. We notice that these three features achieve the Area Under Curve (AUC) value of 0.91 on TREP-21. Furthermore, augmentation of these features with proposed four features for trend promoters classification enhances the AUC value of the model by 6% (0.91 to 0.97). Finally, we utilize our framework for the empirical analysis of large-scale Urdu tweets repository Anbar. Our major contributions in this paper are as follows:

- We build the very first dataset of TREP-21 which contains 2, 061 users labelled as trend promoters and 1, 839 users labelled as normal users.
- After an in depth examination of Twitter users, we design four features for the identification of trend promoters. Moreover, a detailed evaluation of features used for spam and bot accounts classification is performed to find three features adaptable for trend promoters classification.
- Our framework achieves the accuracy of 0.97 for the classification of trend promoters using seven features related to user behavior and tweet content.
- We use Push-To-Trend for empirical analysis of the large-scale Urdu tweets dataset Anbar containing 106.9 million tweets. We identify $147K$ trend promoters and assign *promotion score* to hashtags according to the contribution of these users.

The rest of the paper is organized as follows: Section II describes the literature review while Section III presents the details of the developed dataset. Our framework is presented in Section IV and Section V describes the evaluation results. We present the case study of the Anbar dataset in Section VI. Finally, Section VII provides salient applications of our framework and Section VIII concludes our study.

II. RELATED WORK

Over the years, researchers had analyzed various aspects of social media users. For example, a variety of algorithms were introduced to detect users such as bots, spam, fake accounts, influencers, and cyborgs [18], [22], [28], [29]. These algorithms examined behavior, content, action, and interaction-based features for user classification [30].

User classification was an interesting topic of research that focused on exploring the different attributes of various types of users. In general, these algorithms made use of user profiles, tweeting behavior, tweet content, and social networking features for classification. For instance, Efthimion et al. examined the combination of user profile and tweet text similarity features for the identification of bot accounts [19]. Precisely, user profile features included friend/follower ratio, geo-location, number of followers, etc., while text similarity features were computed using Levenshtein distance between user tweets. Their analysis indicated that text similarity feature was computationally expensive and had low contribution to the bot classification task. Additionally, a study was conducted to identify bots from live streams of tweets with optimized and scalable computation of user profiles, tweeting behavior, and tweet text features [20]. These features included frequency of tweets, follower/following growth rate, length of name, screen name, etc. These features developed a generalizable and real-time model for bot detection with the F1-score of 0.94. In a similar vein to bot detection, user profile and textual features were used to evaluate six machine learning classifiers for spam account classification [23]. The detailed analysis of spam accounts indicated that an automated spam account posted at least 12 tweets per day at a specific time of day. Similarly, machine learning techniques were adopted for the identification of fake followers [25].

TABLE 1. Types of Twitter users in literature.

Sr#	User Type	Definition
1	**Bot**	Twitter accounts handled by a software that generate automated activity [10].
2	**Cyborg**	Twitter accounts with automated as well as human activity [11].
3	**Spammer**	Twitter accounts posting unsolicited or malicious content in their tweets [12].
4	**Influencer**	Twitter accounts with large following that can increase the rate of information diffusion and create an influence on other people [13].
5	**Fake Follower**	Twitter accounts that are set up to increase followers for other users on Twitter network [14].
6	**Trend Promoter**	Twitter accounts or users who make efforts to make a hashtag visible in Twitter trending panel.

TABLE 2. Sample tweets and features of users labelled as trend promoters.

User	Trend	Category	#Tweets	Duplicate	MaxLen	Guidelines	Sample Tweets
UserA	#StudentsKoInsafDo	Campaign	33	0	6	G1, G2, G4	Retweet and use this trend as much as possible to trend it on top.#StudentsKoInsafDo
							We are trending again. But this with more power. #StudentsKoInsafDo
							Today's trend related to online exams are following. Use these trends as much as possible to full fill your duty if you are not able to include them in protest. #StudentsKoInsafDo #StudentsWantOnlineExams #OnlineExamsOrWeProtest #StudentsRejectPhysicalExams #online_exams_only
UserB	#GulmitMainTeerChalega	Political	535	403	34	G1, G2, G3	@ZahoorKhaskhel5 @BBhuttoZardari #GulmitMainTeerChalega
							#GulmitMainTeerChalega
							#GulmitMainTeerChalega
UserC	#MultanSultans	Sports	1	0	0	G2, G3	#PSL5 #PSL2020 #PSLT20 #PSL #LahoreQalandars @lahoreqalandars #KarachiKings @KarachiKingsARY #multansultans @MultanSultans @TeamQuetta #quettagladiators #PhirSeTayyarHain #ptvsports @PTVSp0rts

Authors bought fake followers from online services to create a dataset of fake followers and legitimate users. Next, machine learning models were trained with user behavior features and 10-fold cross-validation.

User profiles, tweeting behavior, and tweet content features focused on attributes of a single user for classification. However, social network features computed the correlation between features of different accounts to classify online Twitter accounts. In Rodriguez-Ruiz et al., user behavior and tweet text-based features were clustered to identify the group of users with bot attributes [21]. Also, similarity in user behavioral patterns of tweeting time and text features of URLs, text, and hashtags were utilized to detect spambots [31]. Moreover, a Graph Convolutional Network (GCN) was designed to exploit follow relationships of users to detect bot communities [32]. The model achieved the accuracy and F1-score values of 0.84 and 0.87, respectively. Shifting to fake accounts, Mohammadrezaei et al. computed similarity measures of users like common friends, cosine, and Jaccard similarity from the corresponding graph of the social network [33]. The algorithm had an underlying assumption that fake accounts connect with each other to create a network. Such network of friends and followers provided the essential information needed for the identification of fake accounts. In addition, the QuickSquad framework detected fake Twitter accounts by optimizing graph-based algorithm with the 'divide and conquer' rule [34]. Precisely, social network graphs of Twitter users were divided into a heavy and light set based on out-degree vertices. Next, two detection methods of dSybilRank and dCOLOR were applied for the detection of fake accounts.

Twitter trends were analyzed to rank trends using a real-time stream of tweets, topic classification, real-time event detection, and manipulation estimation [6], [16], [35], [36], [37]. Examining trend manipulation, Zhang et al. argued that two important factors in determining Twitter trends were the volume and frequency of tweets [35]. Their analysis revealed that these two factors were exploited by manipulators to create fake trends. Similarly, the impact of deleted tweets on creating fake trends was analyzed [6]. The examination of tweets showed that fake, as well as compromised accounts, contributed in creating fake trends with astroturfing attacks. Moreover, the percentage of manipulation in Twitter trends was estimated after analyzing the features related to the frequency of tweets posted by a user [4]. Analysis showed that a small group of users posting a large number of tweets generated fake trends. Finally, in our previous study, we proposed a framework 'Manipify' to detect users involved in the manipulation of Twitter trends [38]. The prominent features used for classification were textual similarity among tweets posted by a user and the number of tweets before trending time. Furthermore, our analysis highlighted that both bot and human accounts participated in trend manipulation.

From the literature review, we notice that a plethora of techniques had been proposed for the classification of users into different categories including bots, spammers, and fake followers. However, only our previous study proposed the framework of Manipify for the identification of manipulators. However, we believe that 'Push-To-Trends' differs from 'Manipify' due to three major reasons. First, Manipify focused on identifying the users creating fake trends which requires information of the trending time of a hashtag. However, 'Push-To-Trends' identifies users involved in creating as well as promoting the trend. In addition, 'Push-To-Trends' uses completely different features from 'Manipify' which are

TABLE 3. TREP-21 – Statistics.

Item	Promoters	Normal	Overall
User	2061	1839	3900
User (%)	52.8	47.2	100
Verified users	17	30	47
Verified users (%)	0.82	1.63	1.20
Geo-tagged users	573	646	1219
Geo-tagged users (%)	27.8	35.1	31.2
# of unique hashtags	219	266	300
Average hashtags/tweet	2.55	2.22	2.39
Average users/hashtag	9.41	6.91	13.0
Max. users/hashtag	125	170	212
Min. users/hashtag	1	1	1
Total # of tweets	72272	3642	75914
Max. # of tweets/ user	691	80	691
Min. # of tweets/ user	1	1	1
Avg. # of tweets/ user	35.07	1.98	19.45
Max. # of Duplicate tweets/ user	616	4	616
Min. # of Duplicate tweets/ user	1	1	1
Avg. # of Duplicate tweets/ user	6.67	1.02	4.00

computationally optimized. Finally, 'Push-To-Trends' also identifies and adopts the useful features used for bot and spam accounts.

III. DATASET

The classification of 'trend promoters' and 'normal' users requires a labelled dataset to train machine learning models. Motivated by the absence of such dataset, we build our TREnd Promoters (TREP-21) dataset. To collect data for labelling, we make use of the observation that the ultimate aim of trend promoters is to gain the visibility of hashtags in the trending panel [39]. First, we leverage two datasets of PK-Nov-20 and PK-Jan-21 containing trending hashtags from Pakistan and their related tweets from November 8th to November 15th, 2020, and January 21st to January 28th, 2021, respectively. The details of these datasets are available in our previous study [38]. We fetch tweets related to trending hashtags along with meta-information using the open-source tool 'Twint' [40]. Note that Twint only scraps 'original' tweets, therefore, retweets and replies are not collected and considered during labelling. PK-Nov-20 and PK-Jan-21 cumulatively contain 477 unique trending hashtags and 3.5 million tweets posted by 1.4 million users.

To build TREP-21, we randomly select and label 4, 000 users. It must be noted that tweets by a user related to one hashtag are grouped into one sample. Users containing tweets related to multiple hashtags are processed as multiple samples for labelling. To mark a user as a trend promoter, we provide the following four guidelines to human annotators.

- G1: Users posting a large number of tweets or duplicate tweets containing trending hashtags
- G2: Repetition of the hashtag within a tweet or multiple tweets with no contextual text
- G3: Mentioning other users with hashtags only and no other text
- G4: Users who post tweets that contain phrases such as "tweet/retweet to trend this hashtag"

Leveraging these guidelines, a human expert examines the content of tweets, user profile, and activity to assign the labels. After labelling, we notice that 3, 900 users are labelled where 2, 061 and 1, 839 users are labelled as trend promoters and normal users, respectively. 100 users are not included in TREP-21 due to the duplicity of samples. Table 2 shows the sample users labelled as trend promoters along with prominent features used to assign the category. Table 3 presents the statistics of TREP-21 dataset. We notice that TREP-21 contains 75, 914 tweets posted by 3, 900 users where 72, 272 (95%) tweets are generated by trend promoters. In addition, on average, trend promoters and normal users post 35 and 2 tweets per user, respectively. Surprisingly, we noticed that 17 (0.8%) 'verified users' are labelled as trend promoters in our dataset. We inspect tweets and profiles of these users and notice that these accounts are related to political parties or sports organizations. During political or sports events, these users post a large number of tweets to generate trends of their hashtags for promotional purposes.

IV. PUSH-TO-TREND FRAMEWORK

In this section, first, we discuss spam and bot features tested for the identification of trend promoters. Next, we describe our proposed features for user classification. Finally, we provide the details of the feature selection method, classification models, and evaluation metrics.

A. SPAM AND BOT FEATURES

Figure 1 shows the process flow diagram of our proposed methodology for the identification of trend promoters. It consists of four distinct phases: i) feature extraction, ii) feature selection, iii) model training, and iv) evaluation. The crucial phase to classify promoters is to identify and extract distinctive features of 'trend promoters' and 'normal users'. We argue that literature lacks features focusing on the identification of such trend promoters. However, a plethora of work is done for the identification of bot and spam users having a pernicious impact on trending panel [18], [22]. First, we test features of spam and bot accounts for trend promoters classification labelled in TREP-21. Focusing on features of bot accounts, we adopt 47 user features presented in Efthimion et al. [19], Yang et al. [20], and Rodriguez-Ruiz et al. [21]. Efthimion et al. [19] propose 15 binary features related to user profile and activity to identify bot accounts. For example, authors argue that the absence of profile pictures, descriptions, and geo-tagged tweets indicates the behavior of bot accounts. In a similar vein, Yang et al. [20] utilize meta-information of user profile to compute 19 features such as profile verified or not, description length, frequency of tweets, and user name length. Finally, Rodriguez-Ruiz et al. [21] avail 13 features to filter bot

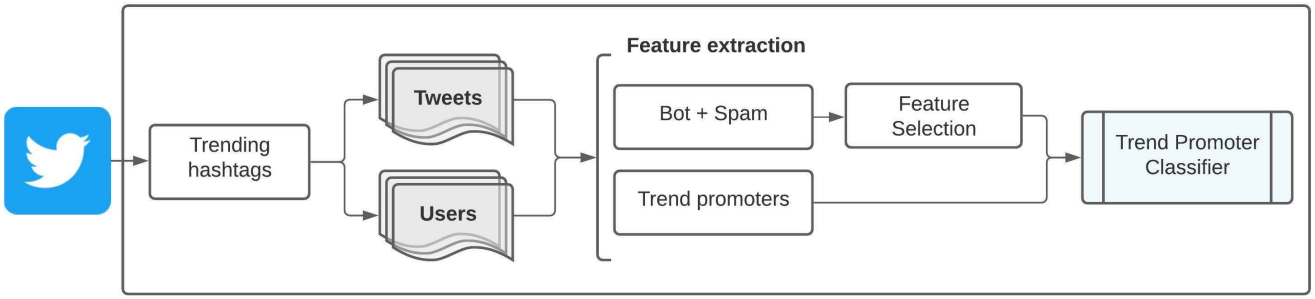

FIGURE 1. Push-To-Trend – process flow.

TABLE 4. Proposed feature set for trend promoters classification.

Sr#	Feature	Description
1	Tweet_count	Number of total original tweets with trending hashtags.
2	Duplicate_tweets	Number of duplicate tweets consisting the trending hashtag by a user.
3	Max_ngram	Length of maximum overlapping ngram among the user tweets.
4	Peak-to-mean	Ratio of maximum tweets by a user and average tweets.

accounts with the one-class classification approach. Similarly, Inuwa-Dutse et al. [23] leverage meta-information of user profiles and user engagement behavior to derive 24 features for the classification of spam accounts. Overall, we consider 71 features for the classification of promoters (refer to Appendix Table 11 for details of features related to spam and bot accounts). However, the only feature of lexical richness (f21) of Inuwa-Dutse et al. is not computed due to the unavailability of multilingual lexical resources. Currently, a lexical dictionary is available for the English language only.

B. TREND PROMOTER FEATURES

As described earlier, trend promoters participate in generating and amplifying a Twitter trend. However, we observe that these users are not necessarily bot or spam accounts. Hence, features of bot and spam accounts are not sufficient for the classification of trend promoters. In this context, we propose four features related to promoters after contemplating the tweet content and tweeting behavior of the trend promoter. Table 4 shows the proposed features along with brief description. These features are: (i) number of tweets with the trending hashtag (*Tweet_count*), (ii) number of duplicate tweets containing the trending hashtag (*Duplicate_tweets*), (iii) length of largest overlapping ngram among tweets of a user (*Max_ngram*), and (iv) peak-to-mean ratio of tweets (*Peak-to-mean*).

Our first proposed feature of *Tweet_count* is based on the observation that promoters post a large number of tweets containing the target hashtag to create a trend [41]. Similarly, *Duplicate_tweets* feature computes the volume of the same tweets posted by a user to increase the volume of tweets related to hashtag [42]. Moreover, posting tweets with similar content from one or multiple accounts is also a violation of Twitter rules [43]. Extending the duplicate tweets features, we focus on calculating *Max_ngram* which assesses the amount of duplicity in the content of tweets posted by a user. To calculate the feature, first, we extract all n-grams of tweets posted by a user related to a hashtag. Next, we compute the length of overlapping n-grams of all possible pairs of tweets. Finally, we use the highest length of overlapping n-gram from all pairs of tweets as *Max_ngram* feature.

Next, we hypothesize that trend promoters post a high volume of tweets in a small amount of time during their complete activity related to a hashtag. Figure 2 shows the example of the temporal frequency of tweets posted by trend promoters and normal users. To avail this attribute, first, we calculate the mean frequency of tweets posted by users. However, we notice that the mean value of tweets over time is not feasible to differentiate the activity of trend promoters and normal users due to indistinguishable values. To utilize this user behavior, we propose *Peak-to-mean* feature which is computed using Equation 1.

$$Peak\text{-}to\text{-}mean = \frac{Max(Tweets)}{Mean(Tweets)} \tag{1}$$

In the equation, *Max(Tweets)* is the maximum number of tweets posted by a user in one hour bin and *Mean(Tweets)* is an average number of tweets related to the hashtag. Figure 2 shows that trend promoters have peak-to-mean value greater than the mean due to the large number of tweets posted in lesser time. Consequently, the peak-to-mean and mean values for normal users are equal. We believe that the peak-to-mean feature is useful in identifying a trend promoter as these users

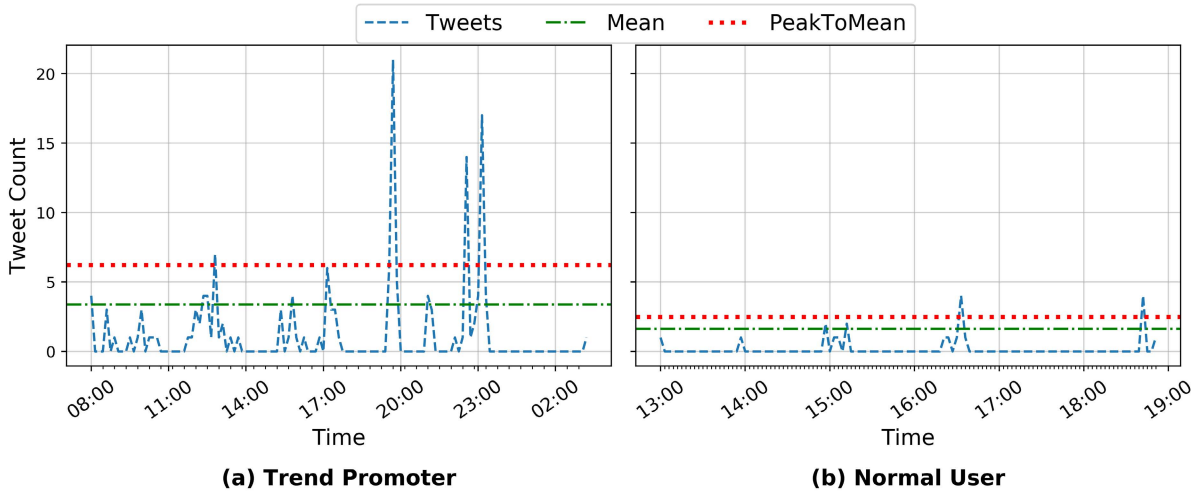

FIGURE 2. Comparison of peak-to-mean and mean values for trend promoter and normal user (bin size=1 hour).

TABLE 5. Pairs of highly correlated features.

Sr#	Features removed	Correlated features	Correlation
1	User_ffratio	Friendship	1.00
2	User_50ffratio	Friendship	1.00
3	User_100ffratio	Friendship	1.00
4	FollowerGrowth	FollowersCount	0.95
5	UniqueURLs	URLCount	0.93
6	FavouriteGrowth	Favourites	0.90
7	UniqueHashtags	HashtagCount	0.88
8	ListedGrowth	ListedCount	0.87
9	UniqueMentions	MentionCount	0.85

post tweets more than the average number of tweets in lesser time [41].

C. FEATURE SELECTION AND EVALUATION

Figure 3 shows our adopted process for feature elimination and selection. Feature selection is an iterative process of removing redundant and useless features. In this regard, first, we identify and remove 18 (out of 71) pairs of duplicate and redundant features proposed in more than one paper for bot and spam accounts classification. In addition, we calculate the values of the remaining 53 features for all samples in TREP-21. We notice that 5 binary features have zero variance for all samples in the dataset suggesting to remove these features. These features include the absence of: i) id, ii) profile picture, iii) screen name, iv) tweet, and v) description.

Next, we perform an analysis of feature correlation on 48 features using the Pearson correlation method to remove redundant features [45]. We notice that 9 pairs of features have a correlation value greater than 0.85. Table 5 shows 9 features removed due to high correlation and their correlation values (see Figure 9 in Appendix for correlation values of 48 features). Such a high value of correlation indicates that

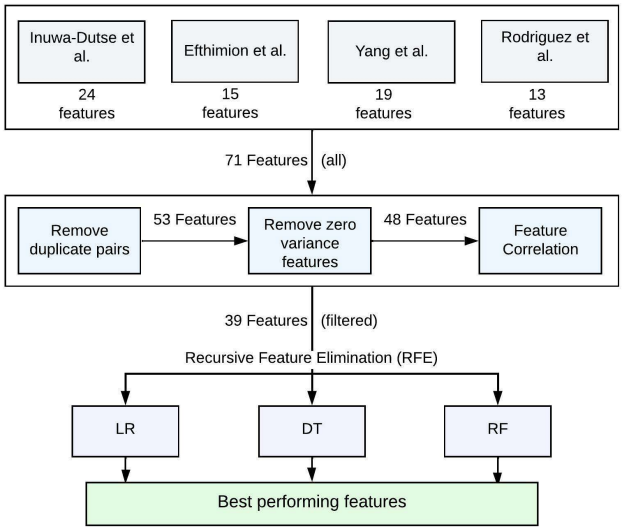

FIGURE 3. Feature selection and elimination process in Push-To-Trend framework.

both features present the same pattern in the data. Hence, we remove one feature from the pair of features for further processing. After removing redundant features, we focus on selecting the most informative features from the 39 (filtered) remaining features for the classification of trend promoters.

For this purpose, we use the Recursive Feature Elimination (RFE) method with three machine learning models [46]. RFE algorithm finds a subset of features by fitting the provided machine learning model with all features in the training dataset, successfully removing least important features, and re-fitting the model. We test the performance of Logistic Regression (LR) [47], Decision Tree (DT) [48], and Random Forest (RF) [49] models by varying the top number of features

TABLE 6. Performance comparison of various features sets on TREP-21 dataset.

Sr#	Feature Set	User	Features	Classifier	Measure				
					AUC	Precision	Recall	F1-score	Accuracy
1	**Inuwa-Dutse** *et al.* [23]	Spam	24	LR	0.65	0.65	0.65	0.65	0.65
				DT	0.69	0.70	0.69	0.70	0.70
				RF	0.80	0.80	0.80	0.80	0.80
2	**Efthimion** *et al.* [19]	Bot	15	LR	0.58	0.58	0.58	0.57	0.58
				DT	0.60	0.60	0.60	0.60	0.60
				RF	0.60	0.61	0.60	0.60	0.60
3	**Yang** *et al.* [44]	Bot	19	LR	0.64	0.64	0.64	0.64	0.64
				DT	0.65	0.65	0.65	0.65	0.65
				RF	0.72	0.72	0.72	0.72	0.72
4	**Rodriguez-Ruiz** *et al.* [21]	Bot	13	LR	0.90	0.90	0.90	0.90	0.90
				DT	0.93	0.93	0.93	0.93	0.93
				RF	0.94	0.94	0.94	0.94	0.94
5	**Combined (all)**	Bot + spam	71	LR	0.89	0.89	0.89	0.89	0.89
				DT	0.90	0.90	0.89	0.90	0.90
				RF	0.90	0.90	0.90	0.90	0.90
6	**Combined (filtered)**	Bot + spam	39	LR	0.73	0.73	0.73	0.73	0.73
				DT	0.87	0.87	0.87	0.87	0.87
				RF	0.89	0.89	0.89	0.89	0.89
7	**Combined (best performing)**	Bot + spam	31	LR	0.73	0.73	0.73	0.73	0.73
			3	DT	0.88	0.88	0.88	0.87	0.88
			3	RF	0.91	0.90	0.90	0.90	0.90
8	**Proposed features**	Trend promoters	4	LR	0.91	0.92	0.91	0.91	0.91
				DT	0.90	0.91	0.90	0.90	0.90
				RF	0.92	0.92	0.92	0.91	0.91
9	**Push-To-Trend**	Bot + spam + trend promoters	7	LR	0.95	0.95	0.95	0.95	0.95
				DT	0.96	0.96	0.96	0.96	0.96
				RF	**0.97**	**0.97**	**0.97**	**0.97**	**0.97**

TABLE 7. Best-performing features selected by three classifiers after applying RFE.

Sr#	Features	Model	Features
1	31	Logistic Regression (LR)	Interestingness, User_GeoLocated, ScreenName_digits, ScreenName_len, User_Followers, URL, EntropyDescription, Description_Len, RepliesC, Name_digits, Activeness, User_Description, TweetLen, Default_profile, Friends_growth, HashtagC, Name_Len, Friendship, Mentions, URLsRatio, EntropyScreenName User_ProfileImage, User_Tweets, MentionRatio, EntropyTweet, User_DescriptionURL,LexRichWithoutUU, NamesRatio, FavouriteC, RepliesC, Verified
2	3	Decision Tree (DT)	RetweetsC, HashtagC, Intertime
3	3	Random Forest (RF)	RetweetsC, HashtagC, Intertime

(k) from 1 to 39. The 'k' features providing the highest value of accuracy and AUC with a minimum number of 'k' are selected for further processing. It is important to note that a train test split of 70:30 is used for all classifiers during evaluation. Also, the implementation for classifiers is done using the scikit-learn library in Python programming language [50], [51]. Finally, the process is done on the machine with 2.4GHz x2 processors, 96GB RAM, 48 cores, and 2TB memory.

V. RESULTS

In this section, first, we provide the details of bot and spam features for the classification of TREP-21. Next, we dive into the performance of our proposed features of trend promoters.

A. SPAM AND BOT FEATURES

Table 6 presents the performance of spam and bot features for trend promoters classification using standard metrics of Area Under Curve (AUC), precision, recall, F1-score, and accuracy [52]. We start by focusing on the performance of 24 spam account features for the TREP-21 dataset [23]. We notice that classifiers of RF, LR, and DR achieve AUC values in the range of 0.65-0.80 with the highest value of RF classifier. Similarly, we observe that 15 features proposed by Efthimion et al. obtain AUC values ranging from 0.58 to 0.60 for three classifiers [19]. In addition, bot account features used by Yang et al. and Rodriguez-Ruiz et al. attain the maximum AUC values of 0.72 and 0.94, respectively. From

TABLE 8. Weights of features assigned by RF classifier.

Sr#	Feature	Category	Feature weight
1	Tweet_count	Trend promoters	0.359
2	Max_ngram	Trend promoters	0.186
3	Peak-to-mean	Trend promoters	0.147
4	HashtagC	Bots	0.136
5	Intertime	Bots	0.104
6	Duplicate_tweets	Trend promoters	0.035
7	RetweetsC	Bots	0.033

TABLE 9. Anbar dataset – Statistics.

Sr#	Feature	Anbar	Anbar-Hashtag
1	Size	855 GB	8.7 GB
2	Tweets	106.9 million	2.9 million
3	Users	1.69 million	0.27 million
4	Hashtags	16.72 million	602
5	Original Tweets	15.1 million	2.9 million
6	Retweeted Tweets	75.9 million	-

TABLE 10. Anbar-Hashtag dataset – User classification.

Sr#	Item	Promoters	Normal	Overall
1	Samples	147,785	926,141	1,073,926
2	Unique Users	43,251	266,764	275,899
3	Unique Users (%)	15.7	84.3	100
4	Tweets	2,694,833	1,259,437	3,954,270
5	Tweets (%)	68.1	31.9	100
6	Avg. Tweets	18.23	1.36	3.68
7	Unique Hashtags	602	602	602
8	Avg. Unique Hashtags	1.98	1.89	1.90
9	HashtagC	753253	2088147	2841401
10	Avg. HashtagC	5.09	2.25	2.65
11	RetweetsC	912671	4398201	5310873
12	Avg. RetweetsC	6.18	4.75	4.95

these results, we conclude that 13 bot account features used in Rodriguez-Ruiz et al. such as the number of retweets, replies, hashtags, URLs, the time interval between tweets, etc., are also informative features for trend promoters classification. Furthermore, interestingly, we notice that maximum values of AUC and accuracy are obtained with Random Forest (RF) classifier.

So far, we have inspected the performance of bot and spam features separately. Next, we combine these features and examine the classification performance of all 71 features as shown in Figure 3. We notice that LR, DT, and RF classifiers achieve the AUC values of 0.89, 0.90, and 0.90, respectively. Unsurprisingly, all features show performance improvement compared to spam and bot features used by Inuwa-Dutse et al., Yang et al., and Efthimion et al. However, all features achieve a lower value of AUC in comparison to Rodriguez-Ruiz et al. This decrease in the performance of all features is linked to the presence of redundant and duplicate features. In this regard, we filter 39 features after removing duplicate and highly correlated features as described in Section IV. Comparing the AUC values of the classifier trained with filtered (39) and all (71) features,

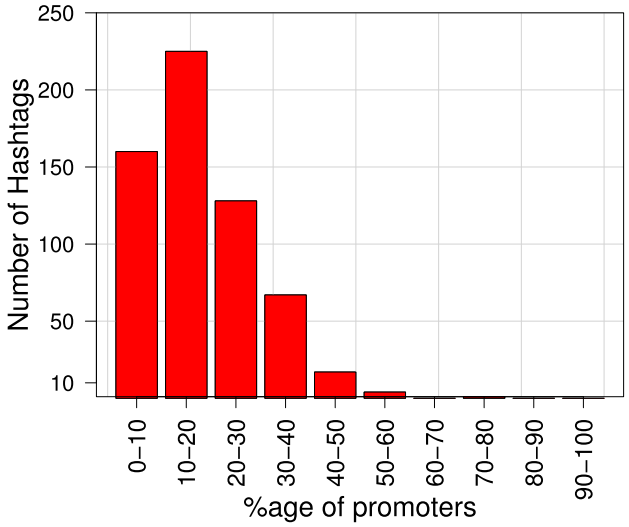

FIGURE 4. Percentage of trend promoters in Anbar-Hashtag.

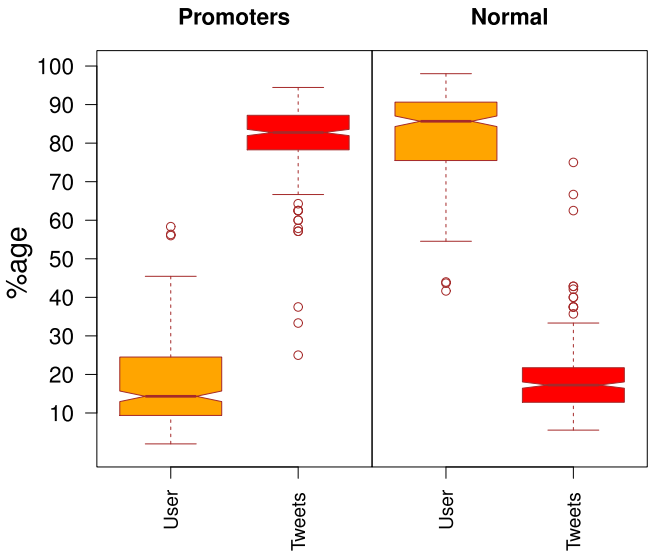

FIGURE 5. Percentage of tweets and users in Anbar-Hashtag.

we notice a decrease of 0.16, 0.03, and 0.01 for LR, DT, and RF, respectively. This result concludes that a set of filtered features contains useless and irrelevant features which adversely impact the performance. Therefore, we apply the RFE feature selection method to identify a combination of best-performing features for LR, DT, and RF. Table 6 provides results achieved using the best-performing combination of features. We notice that LR achieves 0.73 AUC with the top 31 features according to the RFE algorithm. However, DT and RF utilize only the top 3 features to obtain AUC values of 0.88 and 0.91, respectively. Table 7 provides the list of best-performing features for three classifiers. We notice the highest AUC of 0.91 by RF classifier with 3 top features of *RetweetsC*, *HashtagC*, and *Intertime*. Interestingly, all these features are selected from the bot feature set of Rodriguez-Ruiz et al. [21].

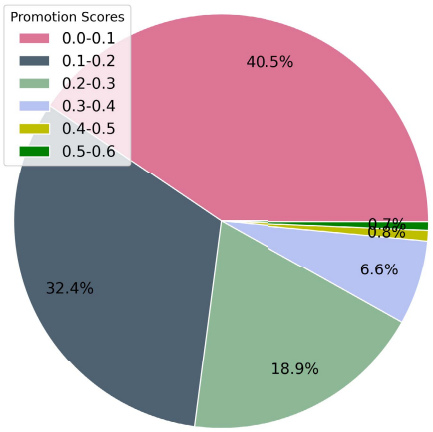

FIGURE 6. Distribution of *promotion score* of hashtags in Anbar-Hashtag dataset.

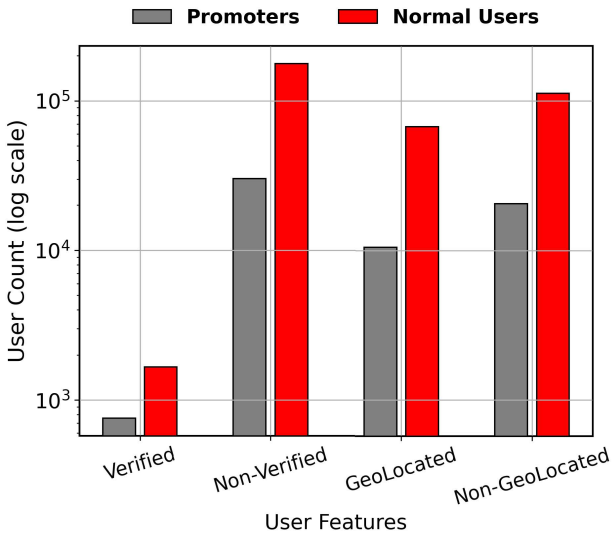

FIGURE 7. Distribution of verified and geo-located users.

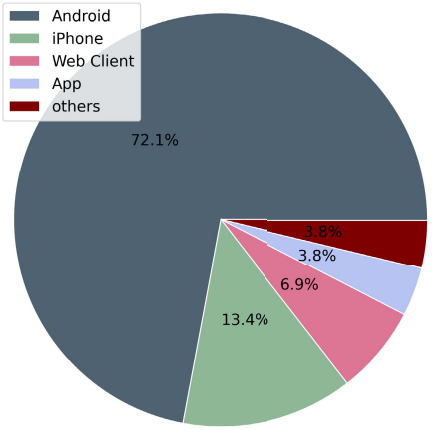

FIGURE 8. Device distribution in Anbar-Hashtag dataset.

B. TREND PROMOTER

From Table 6, we notice that the RF classifier attains 0.92 AUC with four trend promoters features. Next, we augment 3 best performing features (mentioned in Table 7) of bot accounts with our proposed 4 trend promoters features. We notice that AUC of RF classifier is improved from 0.92 to 0.97 with these 7 features. From this result, we conclude that combining the bot and trend promoter features is a beneficial approach for the classification of trend promoters. Interestingly, we observe the comparable performance of LR and DT classifiers with AUC values of 0.95 and 0.96, respectively. Next, we focus on an interesting question which features are important for the classification of trend promoters? To answer this question, we show feature importance assigned by the best performing RF classifier in Table 8. The examination of feature weights indicates that *Tweet_count* is the most important feature for trend promoter classification. This is because posting a large number of tweets by trend promoters creates a trending hashtag. The similar observation is described by Elmas et al. [6]. Also, the frequency of tweets related to a hashtag is one of the key factors in identifying the trend [35]. Moreover, the feature of *Max_ngram* is assigned the second-highest weight by the RF classifier. This observation represents that promoters have larger overlapping n-grams among their tweets. Twitter spam policy also prohibits posting such tweets with similar content [9]. In addition, the feature analysis reveals that our proposed features are assigned higher importance weights compared to features used for bots classification. This observation indicates the importance of proposed features for trend promoter identification. Among bot features, the *HashtagC* feature is allocated the highest importance depicting the frequent usage of hashtags by trend promoters.

VI. EMPIRICAL STUDY ON URDU CONTENT

In this section, first, we present the statistics of large-scale Urdu tweets repository Anbar. Next, we describe the empirical analysis of Anbar through the lens of Push-to-Trend framework.

A. DATASET

We leverage the Anbar [53] to satisfy the need for a large-scale dataset to conduct the empirical analysis. Anbar dataset contains Urdu language tweets fetched using the Twitter API [54]. In particular, Anbar dataset contains 106.9 million Urdu tweets posted by 1.69 million users. However, we apply three filters to develop Anbar-Hashtag dataset for the classification. First, we select 16.72 million (15.7%) tweets containing hashtags from the Anbar dataset. Additionally, note that Push-To-Trend uses only original tweets, therefore, we further filter original tweets from the dataset. Finally, we select tweets of hashtags containing at least 5000 related tweets. Overall, Anbar-Hashtag consists of 602 hashtags, 2.9 million related tweets, and 275K unique users. It is pertinent to mention that a classification sample is generated by user-hashtag pair. For instance, a user posting tweets related

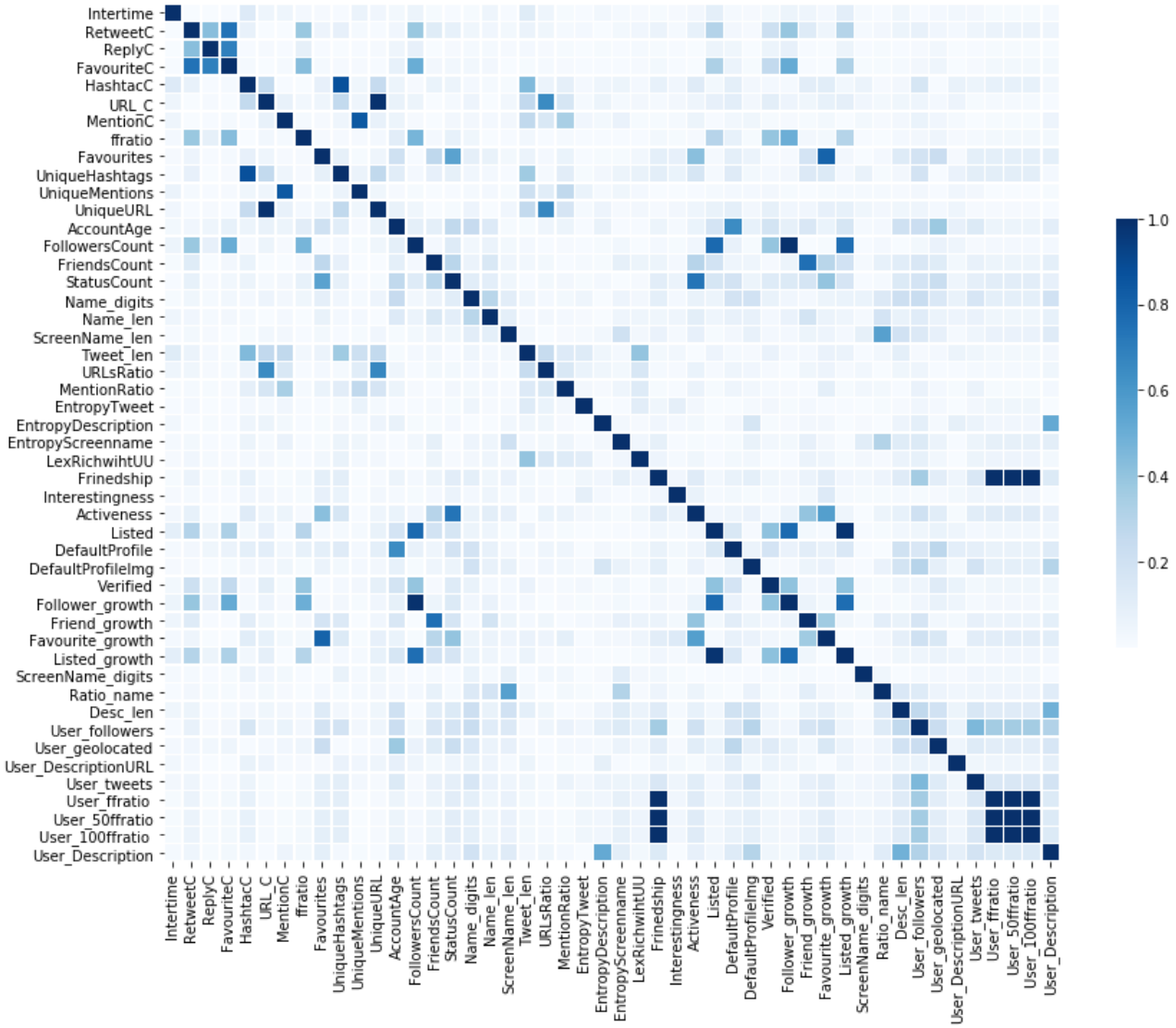

FIGURE 9. Feature correlation of 48 features of bot and spam accounts.

to multiple hashtags is considered as a separate sample for the classification. We notice that total samples in Anbar-Hashtag for classification are 1.07 million. Table 9 shows the statistics of Anbar and Anbar-Hashtag datasets.

B. ANALYSIS

We commence the analysis of the Anbar-Hashtag dataset by classifying the 1.07 million samples using our framework. Table 10 shows the statistics of classified users. We notice that only 147K (15.7%) users are marked as trend promoters. Interestingly, 15.7% of these trend promoters posted 68.1% of total tweets in the dataset. We further examine the characteristics of trend promoters. Figure 4 shows the percentage of trend promoters for 602 hashtags. For instance, 224 (37.2%) hashtags contain 10-20% of users labelled as trend promoters. We notice 597 (99.1%) hashtags have less than 50% trend promoters. Also, one hashtag of #*paknews* has 70% of users labelled as trend promoters. We further inspect this hashtag and observe that the #*paknews* hashtag is tweeted by only 21 users. However, the total tweets related to hashtag are 17, 472 where 17, 464 of these tweets are posted by 15 trend promoters. Such a large number of tweets posted by a small set of users depict the effort of malicious users. Furthermore, we observe that the average hashtag count for trend promoters in Anbar-Hashtag dataset is 5.09 while it is 2.25 for normal users.

Figure 5 presents the boxplot of percentage of trend promoters and their posted tweets related to hashtags. The graph exhibits a large difference between the median values of users and tweets. For instance, the median value for trend promoters is 14 while it is 86 for normal users. Similarly, in the case of trend promoters tweets, the median lies between 80-90 and for normal users tweets, this number lies in the range of 10-20. As expected, trend promoters post a large number of tweets to promote a hashtag [38]. Moreover, we assign the '*promotion score* ($Score_H$)' to each hashtag based on the

TABLE 11. Features proposed for spam and bot accounts.

Id	Features	Description	Id	Features	Description
Inuwa-Dutse *et al.* **[23]**					
f1	AccountAge	Days since account creation	**f13**	URLsRatio	Characters in URL / TweetLen
f2	FollowersCount	From user profile meta-data	**f14**	MentionRatio	Characters in mentions / TweetLen
f3	FriendsCount	From user profile meta-data	**f15**	NameSim	Similarity in UserName & ScreenName
f4	StatusesCount	From user profile meta-data	**f16**	LexRichWithUU	Type token ration (TTR)
f5	DigitsCountInName	Number of digits in username	**f17**	Friendship	FriendsCount / FollowersCount
f6	TweetLen	Number of characters in tweet	**f18**	Followership	FollowersCount / FriendsCount
f7	UserNameLen	Number of characters in user name	**f19**	Interestingness	FavouritesCount / StatusesCount
f8	ScreenNameLen	Number of characters in screen name	**f20**	Activeness	StatusesCount / AccountAge
f9	EntropyTweet	SE*(Tweet) / Tweet length	**f21**	LexRichWithoutUU	Lexical words / Total words
f10	EntropyDescription	SE*(Description) / Description length	**f22**	VerifiedAccount	From user profile meta-data
f11	EntropyUserName	SE*(UserName) / UserName length	**f23**	FavouritesCount	From user profile meta-data
f12	EntropyScreenName	SE*(ScreenName) / ScreenName length	**f24**	NamesRatio	ScreenNameLen / UserNameLen
Efthimion *et al.* **[19]**					
f1	User_id	Absence of id	**f9**	User_ffratio	2:1 friends/followers ratio
f2	User_profile	Absence of a profile picture	**f10**	User_followers1000	Has over 1,000 followers
f3	User_screenName	Absence of a screen name	**f11**	User_profileImage	Has the default profile image
f4	User_followers	Has less than 30 followers	**f12**	User_tweeted	Has never tweeted
f5	User_geoLocated	Not geo-located	**f13**	User_50ffratio	50:1 friends/followers ratio
f6	User_language	Language not set to English	**f14**	User_100ffratio	100:1 friends/followers ratio
f7	User_descriptionURL	Description contains a link	**f15**	User_description	Absence of a description
f8	User_tweets	Has sent less than 50 tweets			
Yang *et al.* **[20]**					
f1	Statuses_count	From user profile meta-data	**f11**	Favourites_growth	Favourites_count / user_age
f2	Followers_count	From user profile meta-data	**f12**	Followers_growth	Followers_count / user_age
f3	Friends_count	From user profile meta-data	**f13**	Friends_growth	Friends_count / user_age
f4	Listed_count	From user profile meta-data	**f14**	Listed_growth	Listed_count / user_age
f5	Default_profile	From user profile meta-data	**f15**	ffratio	Followers_count / friends_count
f6	Profile_background_image	From user profile meta-data	**f16**	ScreenName_len	Length of screen name
f7	Verified	From user profile meta-data	**f17**	ScreenName_digits	Digits in screen name
f8	Description_len	Characters in description	**f17**	Name_len	Length of user name
f9	ScreenName_likelihood	Likelihood of screen name	**f19**	Name_digits	Digits in user name
f10	Tweet_freq	Statuses_count / user_age			
Rodriguez-Ruiz *et al.* **[21]**					
f1	RetweetsC	Retweets count / Tweet count	**f8**	ffratio	Friends count / Followers count
f2	RepliesC	Reply count / Tweet count	**f9**	Favourites	Number of favourited tweets
f3	FavouriteC	Favourited tweets/ Tweet count	**f10**	Listed	Number of listed tweets
f4	HashtagC	Hashtag count / Tweet count	**f11**	UniqueHashtags	Unique hashtags / Tweet count
f5	URL	URL count / Tweet count	**f12**	UniqueMentions	Unique mention / Tweet count
f6	Mentions	Mention count / Tweet count	**f13**	UniqueURL	Unique URLs / Tweet count
f7	Intertime	Average seconds between tweets			

* Shannon Entropy

activities of trend promoters. First, we calculate the ratio of the number of trend promoters (*Promoters*) and the number of total users (*TotalUsers*) tweeting the hashtag. Next, we determine the ratio of tweets posted by trend promoters (*PromoterTweets*) and total users (*TotalTweets*). Finally, both ratios are combined using the multiplication operator as described in Equation 2. Using this equation, hashtags are assigned the promotion score in the range of 0 to 1.

$$Score_H = \frac{Promoters}{TotalUsers} \times \frac{PromoterTweets}{TotalTweets} \quad (2)$$

Figure 6 shows the distribution of promotion score assigned to 602 hashtags in equally sized bins. The highest percentage of hashtags are assigned a score in 0.0-0.1 range. Moreover, the maximum score of 0.55 is assigned to the #*paknews* hashtag.

Shifting towards trend promoters and normal users attributes, Figure 7 shows the distribution of verified accounts and geo-location of users. It is noticed that trend promoters have only 756 verified accounts whereas 1666 normal users have verified accounts. A similar pattern is observed for geo-located users. This investigation highlights that trend promoters prefer not to disclose their location. Moreover, Figure 8 shows the distribution of device usage by all users in the Anbar-Hashtag dataset. We notice that Android is the most frequently used device which is used by more than 50% of users. Furthermore, we include the devices with less than 2% contribution in the 'others' category. These sources include Facebook, TweetDeck, Instagram, BlackBerry, and other third-party applications and online services. From the empirical analysis of Anbar-Hashtag, we conclude that the Push-To-Trend framework successfully reveals the difference between various attributes like tweeting behaviour, tweet content, user account information, etc., of trend promoters and normal users.

VII. APPLICATIONS

The Push-To-Trend framework is a language-independent framework which can be adapted for the analysis of tweets

posted in different natural languages. In general, systems are language-dependent due to two critical steps of pre-processing and feature extraction. For instance, pre-processing method of stopword removal requires a dictionary containing stopwords of a particular language. However, our Push-To-Trend framework only removes emojis and extra spaces. In addition, textual features like vocabulary make models highly language-dependent which are not used in our framework. Push-To-Trend leverages the text of tweets to compute the number of *Duplicate_tweets* and *Max_ngram* instead of using the text as a feature. We also substantiate our claim by utilizing the framework on the Urdu tweets repository Anbar.

We believe that the Push-To-Trend framework has applications in a wide range of fields. For instance, it can be utilised to make the trending panel more organic by limiting Twitter usage of trend promoters. In addition, our framework can assist governments and social media researchers in detecting anti-state propaganda on Twitter. Moreover, our framework can be extended to determine the demographics of both trend promoters and normal users which is useful for journalists and sociologists. Furthermore, socio-informatics researchers can also utilise this work to determine the organic sentiment for a topic.

VIII. CONCLUSION

In this article, we present the Push-To-Trend – a novel framework to detect trend promoters in trending hashtags. First, we build a labelled dataset of TREP-21 containing 3, 900 users labelled as trend promoters and normal users. Additionally, we design four features of number of total tweets, duplicate tweets with the hashtag, overlapping ngram in tweets, and peak-to-mean ratio for trend promoters classification. Also, we evaluate the performance of 71 spam and bot accounts features using correlation and recursive feature elimination methods to discover three useful features for trend promoter classification. These three selected features are count of retweets, hashtags, and intermediate time between tweets posted by a user. Our framework achieves an accuracy of 0.97 after the augmentation of four designed and three selected features. In addition, we examine these selected seven features and notice that number of tweets and duplicate tweets are the most informative features for the classification of trend promoters. Furthermore, we avail Push-To-Trend for the empirical examination of a large-scale Urdu repository Anbar containing 106.9 million tweets. In particular, we analyze 602 most frequent hashtags and successfully identify $147K$ trend promoters. Finally, we exploit the trend promoter activities to assign the 'promotion score' to hashtags.

In future, we aim to extend our framework to detect communities of trend promoters performing coordinated activities. In addition, the framework will also be extended to differentiate between organic and in-organic Twitter trends. Finally, we plan to detect fake news from in-organic trends.

APPENDIX.

Figure 9 presents the correlation for 48 features using heat map graph. In addition, Table 11 shows the details of features related to user classification proposed in literature.

REFERENCES

[1] B. Arias. (2018). *How to Cover Breaking News on Twitter*. Accessed: Jan. 28, 2021. [Online]. Available: https://media.twitter.com/en/articles/best-practice/2018/how-to-cover-breaking-news-on-twitter.html
[2] G. Brena, M. Brambilla, S. Ceri, M. Di Giovanni, F. Pierri, and G. Ramponi, "News sharing user behaviour on Twitter: A comprehensive data collection of news articles and social interactions," in *Proc. AAAI Conf. Web Social Media*, vol. 13, 2019, pp. 592–597.
[3] F. B. Soares and R. Recuero, "Hashtag wars: Political disinformation and discursive struggles on Twitter conversations during the 2018 Brazilian presidential campaign," *Social Media Soc.*, vol. 7, no. 2, 2021, Art. no. 20563051211009073.
[4] B. Nimmo, "Measuring traffic manipulation on Twitter," Project Comput. Propaganda, Oxford, New York, NY, USA, Work. Paper 1, Jan. 2019. [Online]. Available: https://demtech.oii.ox.ac.uk/research/posts/measuring-traffic-manipulation-on-twitter/
[5] E. Chen, A. Deb, and E. Ferrara, "#Election2020: The first public Twitter dataset on the 2020 US presidential election," *J. Comput. Social Sci.*, vol. 5, no. 1, pp. 1–18, May 2022.
[6] T. Elmas, R. Overdorf, A. F. Ozkalay, and K. Aberer, "Ephemeral astroturfing attacks: The case of fake Twitter trends," in *Proc. IEEE Eur. Symp. Secur. Privacy (EuroSP)*, Sep. 2021, pp. 403–422.
[7] A. K. Mishra. (Apr. 2016). *How to Manufacture a Twitter Trend*. Accessed: Nov. 16, 2021. [Online]. Available: https://www.livemint.com/Consumer/FmndRnomzWZlXye0rFEfSK/How-to-manufacture-a-Twitter-trend.html
[8] E. Gallagher. (2016). *Manipulating Trends & Gaming Twitter*. Accessed: Mar. 25, 2021. [Online]. Available: https://erin-gallagher.medium.com/manipulating-trends-gaming-twitter-6fd31714c06c
[9] Twitter. *Twitter Trends FAQs*. Accessed: Jul. 8, 2021. [Online]. Available: https://help.twitter.com/en/using-twitter/twitter-trending-faqs
[10] S. Feng, H. Wan, N. Wang, J. Li, and M. Luo, "TwiBot-20: A comprehensive Twitter bot detection benchmark," in *Proc. 30th ACM Int. Conf. Inf. Knowl. Manage.*, Oct. 2021, pp. 4485–4494.
[11] N. Rahman, M. Maimuna, A. Begum, M. Ahmed, and M. S. Arefin, "A survey of data mining techniques in the field of cyborg mining," in *Soft Computing for Security Applications*. Singapore: Springer, 2022, pp. 781–797.
[12] W. Daffa, O. Bamasag, and A. AlMansour, "A survey on spam URLs detection in Twitter," in *Proc. 1st Int. Conf. Comput. Appl. Inf. Secur. (ICCAIS)*, Apr. 2018, pp. 1–6.
[13] Z. Z. Alp and Ş. G. Öğüdücü, "Identifying topical influencers on Twitter based on user behavior and network topology," *Knowl.-Based Syst.*, vol. 141, pp. 211–221, Feb. 2018.
[14] B. Jang, S. Jeong, and C.-K. Kim, "Distance-based customer detection in fake follower markets," *Inf. Syst.*, vol. 81, pp. 104–116, Mar. 2019.
[15] M. Adedoyin-Olowe, M. M. Gaber, C. M. Dancausa, F. Stahl, and J. B. Gomes, "A rule dynamics approach to event detection in Twitter with its application to sports and politics," *Exp. Syst. Appl.*, vol. 55, pp. 351–360, Aug. 2016.
[16] A. Zubiaga, D. Spina, R. Martínez, and V. Fresno, "Real-time classification of Twitter trends," *J. Assoc. Inf. Sci. Technol.*, vol. 66, no. 3, pp. 462–473, 2015.
[17] D. E. Cahyani and A. W. Putra, "Relevance classification of trending topic and Twitter content using support vector machine," in *Proc. Int. Seminar Appl. Technol. Inf. Commun. (iSemantic)*, Sep. 2021, pp. 87–90.
[18] A. Anwar and U. Yaqub, "Bot detection in Twitter landscape using unsupervised learning," in *Proc. 21st Annu. Int. Conf. Digit. Government Res.*, Jun. 2020, pp. 329–330.
[19] P. G. Efthimion, S. Payne, and N. Proferes, "Supervised machine learning bot detection techniques to identify social Twitter bots," *SMU Data Sci. Rev.*, vol. 1, no. 2, p. 5, 2018.
[20] K.-C. Yang, O. Varol, P.-M. Hui, and F. Menczer, "Scalable and generalizable social bot detection through data selection," in *Proc. AAAI Conf. Artif. Intell.*, vol. 34, no. 1, 2020, pp. 1096–1103.

[21] J. Rodríguez-Ruiz, J. I. Mata-Sánchez, R. Monroy, O. Loyola-González, and A. López-Cuevas, "A one-class classification approach for bot detection on Twitter," *Comput. Secur.*, vol. 91, Apr. 2020, Art. no. 101715.
[22] F. Masood, G. Ammad, A. Almogren, A. Abbas, H. A. Khattak, I. Ud Din, M. Guizani, and M. Zuair, "Spammer detection and fake user identification on social networks," *IEEE Access*, vol. 7, pp. 68140–68152, 2019.
[23] I. Inuwa-Dutse, M. Liptrott, and I. Korkontzelos, "Detection of spam-posting accounts on Twitter," *Neurocomputing*, vol. 315, pp. 496–511, Nov. 2018.
[24] S. Cresci, R. Di Pietro, M. Petrocchi, A. Spognardi, and M. Tesconi, "Fame for sale: Efficient detection of fake Twitter followers," *Decision Support Syst.*, vol. 80, pp. 56–71, Dec. 2015.
[25] A. Khalil, H. Hajjdiab, and N. Al-Qirim, "Detecting fake followers in Twitter: A machine learning approach," *Int. J. Mach. Learn. Comput.*, vol. 7, no. 6, pp. 198–202, Dec. 2017.
[26] A. Badawy, E. Ferrara, and K. Lerman, "Analyzing the digital traces of political manipulation: The 2016 Russian interference Twitter campaign," in *Proc. IEEE/ACM Int. Conf. Adv. Social Netw. Anal. Mining (ASONAM)*, Aug. 2018, pp. 258–265.
[27] A. Thieltges, O. Papakyriakopoulos, J. C. M. Serrano, and S. Hegelich, "Effects of social bots in the iran-debate on Twitter," 2018, *arXiv:1805.10105*.
[28] C. A. Davis, O. Varol, E. Ferrara, A. Flammini, and F. Menczer, "BotOrNot: A system to evaluate social bots," in *Proc. 25th Int. Conf. Companion World Wide Web WWW Companion*, 2016, pp. 273–274.
[29] A. B. Eliacik and N. Erdogan, "Influential user weighted sentiment analysis on topic based microblogging community," *Exp. Syst. Appl.*, vol. 92, pp. 403–418, Feb. 2018.
[30] M. Orabi, D. Mouheb, Z. Al Aghbari, and I. Kamel, "Detection of bots in social media: A systematic review," *Inf. Process. Manage.*, vol. 57, no. 4, Jul. 2020, Art. no. 102250.
[31] S. Cresci, R. Di Pietro, M. Petrocchi, A. Spognardi, and M. Tesconi, "Dna-inspired online behavioral modeling and its application to spambot detection," *IEEE Intell. Syst.*, vol. 31, no. 5, pp. 58–64, Sep./Oct. 2016.
[32] S. Feng, H. Wan, N. Wang, and M. Luo, "BotRGCN: Twitter bot detection with relational graph convolutional networks," in *Proc. IEEE/ACM Int. Conf. Adv. Social Netw. Anal. Mining*, Nov. 2021, pp. 236–239.
[33] M. Mohammadrezaei, M. E. Shiri, and A. M. Rahmani, "Identifying fake accounts on social networks based on graph analysis and classification algorithms," *Secur. Commun. Netw.*, vol. 2018, pp. 1–8, Aug. 2018.
[34] X. Jiang, Q. Li, Z. Ma, M. Dong, J. Wu, and D. Guo, "QuickSquad: A new single-machine graph computing framework for detecting fake accounts in large-scale social networks," *Peer-to-Peer Netw. Appl.*, vol. 12, no. 5, pp. 1385–1402, Sep. 2019.
[35] Y. Zhang, X. Ruan, H. Wang, H. Wang, and S. He, "Twitter trends manipulation: A first look inside the security of Twitter trending," *IEEE Trans. Inf. Forensics Security*, vol. 12, no. 1, pp. 144–156, Jan. 2017.
[36] A. H. Hossny and L. Mitchell, "Event detection in Twitter: A keyword volume approach," in *Proc. IEEE Int. Conf. Data Mining Workshops (ICDMW)*, Nov. 2018, pp. 1200–1208.
[37] H. U. Khan, S. Nasir, K. Nasim, D. Shabbir, and A. Mahmood, "Twitter trends: A ranking algorithm analysis on real time data," *Exp. Syst. Appl.*, vol. 164, Feb. 2021, Art. no. 113990.
[38] S. Kausar, B. Tahir, and M. Amir Mehmood, "Towards understanding trends manipulation in Pakistan Twitter," 2021, *arXiv:2109.14872*.
[39] A. Onuchowska, D. J. Berndt, and S. Samtani, "Rocket ship or blimp?–implications of malicious accounts removal on Twitter," in *Proc. Eur. Conf. Inf. Syst. (ECIS)*, Jun. 2019, pp. 1–10.
[40] C. Zacharias and F. Poldi. (2018). *Twint*. Accessed: Mar. 9, 2021. [Online]. Available: https://pypi.org/project/twint/
[41] D. Assenmacher, L. Clever, J. S. Pohl, H. Trautmann, and C. Grimme, "A two-phase framework for detecting manipulation campaigns in social media," in *Proc. Int. Conf. Hum.-Comput. Interact.* Cham, Switzerland: Springer, 2020, pp. 201–214.
[42] A. Johns and N. Cheong, "Feeling the chill: Bersih 2.0, state censorship, and networked affect on Malaysian social media 2012–2018," *Social Media Soc.*, vol. 5, no. 2, 2019, Art. no. 2056305118821801.
[43] Twitter. *Twitter Developer Community*. Accessed: Feb. 2, 2022. [Online]. Available: https://twitterdev.bevylabs.com/
[44] K. Yang, O. Varol, C. A. Davis, E. Ferrara, A. Flammini, and F. Menczer, "Arming the public with artificial intelligence to counter social bots," *Human Behav. Emerg. Technol.*, vol. 1, no. 1, pp. 48–61, Jan. 2019.
[45] J. Benesty, J. Chen, Y. Huang, and I. Cohen, "Pearson correlation coefficient," in *Noise Reduction in Speech Processing*. Berlin, Germany: Springer, 2009, pp. 1–4.
[46] I. Guyon, J. Weston, S. Barnhill, and V. Vapnik, "Gene selection for cancer classification using support vector machines," *Mach. Learn.*, vol. 46, nos. 1–3, pp. 389–422, 2002.
[47] D. R. Cox, "The regression analysis of binary sequences," *J. Roy. Stat. Soc., B Methodol.*, vol. 20, no. 2, pp. 215–232, 1958.
[48] X. Wu, V. Kumar, J. R. Quinlan, J. Ghosh, Q. Yang, H. Motoda, G. J. McLachlan, A. Ng, B. Liu, P. S. Yu, Z.-H. Zhou, M. Steinbach, D. J. Hand, and D. Steinberg, "Top 10 algorithms in data mining," *Know. Inf. Sys.*, vol. 14, pp. 1–37, Dec. 2008.
[49] T. K. Ho, "Random decision forests," in *Proc. 3rd Int. Conf. Document Anal. Recognit.*, vol. 1. Aug. 1995, pp. 278–282.
[50] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, A. Müller, J. Nothman, G. Louppe, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, Oct. 2012.
[51] M. F. Sanner, "Python: A programming language for software integration and development," *J. Mol. Graph Model.*, vol. 17, no. 1, pp. 57–61, 1999.
[52] N. Seliya, T. M. Khoshgoftaar, and J. Van Hulse, "A study on the relationships of classifier performance metrics," in *Proc. IEEE Int. Conf. Tools Artif. Intell.*, Nov. 2009, pp. 59–66.
[53] B. Tahir and M. A. Mehmood, "Anbar: Collection and analysis of a large scale Urdu language Twitter corpus," *J. Intell. Fuzzy Syst.*, vol. 42, no. 5, pp. 4789–4800, 2022.
[54] K. Makice, *Twitter API: Up and Running Learn How to Build Applications With the Twitter API*, 1st ed. Sebastopol, CA, USA: O'Reilly Media, 2009.

SOUFIA KAUSAR received the B.Sc. degree in computer science from the Lahore College for Women University, Lahore, Pakistan, in 2017, and the M.S. degree in computer science from the National University of Computing and Emerging Sciences (FAST-NU), Lahore, in 2019. Currently, she is working as a Research Officer with the Al-Khawarizmi Institute of Computer Science (KICS), University of Engineering and Technology (UET) Lahore, Lahore. Her research interests include machine learning, deep learning, computer vision, and text mining.

BILAL TAHIR received the B.Sc. degree in electrical engineering from the National University of Computer and Emerging Sciences (FAST-NU), Lahore, Pakistan, in 2014, and the M.S. degree in computer engineering from the University of Engineering and Technology (UET) Lahore, Lahore, in 2018. Since 2017, he has been working as a Senior Research Officer with the Al-Khawarizmi Institute of Computer Science (KICS), UET Lahore. His research interests include machine learning for images and text, natural language processing, deep learning, and information retrieval.

MUHAMMAD AMIR MEHMOOD received the Ph.D. degree in engineering from the Department of Electrical Engineering and Computer Science, Technische Universität Berlin, Deutsche Telekom Innovation Laboratories, Berlin, Germany, in 2012, under the supervision of Prof. Anja Feldmann. Currently, he is working as an Associate Professor at the Al-Khawarizmi Institute of Computer Science, University of Engineering and Technology Lahore, Pakistan. He has been the Head of the High-Performance Computing and Networking Laboratory (HPCNL), since 2013. His research interests include internet measurements, big data, information retrieval, and deep learning.