Bot-Detective: An explainable Twitter bot detection service with crowdsourcing functionalities

Maria Kouvela*
mvkouvela@csd.auth.gr
Informatics Department, Aristotle University of Thessaloniki
Thessaloniki

Ilias Dimitriadis
Informatics Department, Aristotle University of Thessaloniki
Thessaloniki, Greece
idimitriad@csd.auth.gr

Athena Vakali
Informatics Department, Aristotle University of Thessaloniki
Thessaloniki, Greece
avakali@csd.auth.gr

ABSTRACT

Popular microblogging platforms (such as Twitter) offer a fertile ground for open communication among humans, however, they also attract many bots and automated accounts "disguised" as human users. Typically, such accounts favor malicious activities such as phishing, public opinion manipulation and hate speech spreading, to name a few. Although several AI driven bot detection methods have been implemented, the justification of bot classification and characterization remains quite opaque and AI decisions lack in ethical responsibility. Most of these approaches operate with AI black-boxed algorithms and their efficiency is often questionable. In this work we propose *Bot-Detective*, a web service that takes into account both the efficient detection of bot users and the interpretability of the results as well. Our main contributions are summarized as follows: *i)* we propose a novel explainable bot-detection approach, which, to the best of authors' knowledge, is the first one to offer interpretable, responsible, and AI driven bot identification in Twitter,*ii)* we deploy a publicly available bot detection Web service which integrates an explainable ML framework along with users feedback functionality under an effective crowdsourcing mechanism; *iii)* we build the proposed service under a newly created annotated dataset by exploiting Twitter's rules and existing tools. This dataset is publicly shared for further use. In situ experimentation has showcased that Bot-Detective produces comprehensive and accurate results, with a promising service take up at scale.

CCS CONCEPTS

• **Human-centered computing** → *Collaborative and social computing systems and tools*; • **Computing methodologies** → *Artificial intelligence*; • **Information systems** → **Social networks**.

KEYWORDS

social influence, social networks, social bots, bot detection, explainable AI

*All authors contributed equally to this research.

MEDES '20, November 2–4, 2020, Virtual Event, United Arab Emirates

ACM ISBN 978-1-4503-8115-4/20/11…$15.00
https://doi.org/10.1145/3415958.3433075

ACM Reference Format:
Maria Kouvela, Ilias Dimitriadis, and Athena Vakali. 2020. Bot-Detective: An explainable Twitter bot detection service with crowdsourcing functionalities. In *12th International Conference on Management of Digital EcoSystems (MEDES '20), November 2–4, 2020, Virtual Event, United Arab Emirates.* ACM, Abu Dhabi, UAE, 9 pages. https://doi.org/10.1145/3415958.3433075

1 INTRODUCTION

Social networks have quickly become an omnipresent and a default part of our lives, boosting online human interactions and media sharing among peers. Among the most popular social media platforms, Twitter has a crucial role in such interactions and content sharing, since it connects millions of active users. Twitter has quickly turned into a major medium for the viral dissemination of views and information, largely transforming many sectors such as digital advertising, marketing, journalism, etc. However, its large social and economic impact in these sectors, has attracted a dark side of users who act maliciously with efforts to manipulate and influence people and their decisions.

Social bots, i.e. code functions which operate as human accounts, proceed with automated actions that resemble those that people would do [1], are widely used and deployed in Twitter. The large impact of social bots is evident by a recent research which has shown that almost one third of the content shared in Twitter is bot-generated [4], two thirds of the URLs shared are posted by automated accounts [49] and that 9-15% of all Twitter users are not actually human, but bots [47]. At the same time, the fact that users tend to believe that bot generated content is credible [18] turns the bots phenomenon into a challenging and critical one in terms of trust establishment in social networks. Certainly, bots are not always malicious, since many of them are used for good purposes. For example, botwiki[1] includes multiple such accounts that promote news, art or funny, yet not insulting, content. Relative research has showcased the dual sides of social bots which might either have a positive impact [32],[41] or show a benevolent behavior which impacts opinion distortion [46],[5].

The vast majority of research has focused on the on-growing "community" of malicious bots since social spam remains a crucial open challenge over the last decades and it has been intensified by the online social bots [20]. Twitter bots were initially studied almost a decade ago [45],[31],[15],[10], where scientists profiled the potential and activities of bots and addressed the problem of efficiently identifying them. There is already a broad range of malevolent activities associated with harmful bots. Most of them relate with the phenomena of : dissemination of phishing links; spreading fake news; inaccurate information [27],[8]; and deceiving human users

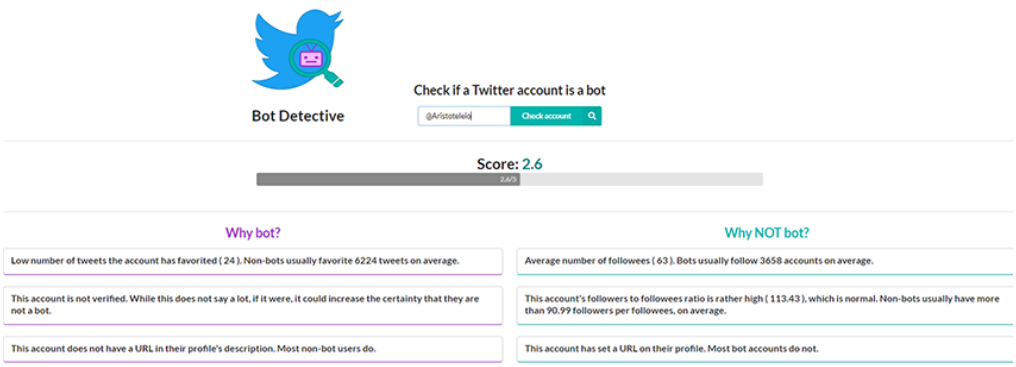

Figure 1: An illustrative example of the Bot-Detective service. The user enters a Twitter handle and receives a bot-likelihood score and the explanations behind this reasoning

[28]. What is more interesting, is the fact that people are still not able to recognize bots and do not understand why an account has been identified as such, because they simply don't know how and why [3]. In the most recent work, more intense challenges have been addressed since social bots were used to build up terrorist hypes [6], cyberbullying [14], stimulate actions in the stock market [21], increase polarity for sensitive subjects [12], and even influence political elections and results [9], [38]. Obviously, the problem of identifying bots and bot generated content has become very critical and essential. As the evolution of bots raised concern and drew public's attention [43] (e.g. role of bots in the USA 2016 elections discussion [42]), Twitter platform has proceeded to the removal of millions of bot accounts [40]. It was not just public's concern that stimulated Twitter to act accordingly [44] but also the high economical costs associated with the phenomenon. According to recent studies, social bots cost companies around 17 billion dollars annually [11],[22].

Bot detection has become a cutting edge research topic and it has been addressed by efforts which favor advanced Machine Learning (ML) and Artificial Intelligence (AI) techniques [17],[24]. Despite the recent strong emphasis on bot detection many issues remain challenging and open. According to [50], even well established AI bot detection tools, like Botometer[17], perform bot classification but provide limited explanations, thus leading to fundamental misunderstandings. This fact is actually imprinted in the latest data and analytics technology trends [34] where Explainable Artificial Intelligence (XAI),a way to present the result of a black box model in human readable terms [26], ranks on top five research directions.

This work places emphasis on delivering explainable and comprehensive bot detection characterizations based on the principle that providing meaningful explanations is not just a matter of result understanding or gaining user's trust, but rather it is a matter of social responsibility, which is also imposed by EU laws as well [2]. According to the authors knowledge, there is no such tool that apart from identifying bots, also provides explanations on the reasons behind this decision. In an attempt to fill this bot detection explainability gap, this paper proposes *Bot-Detective*[1], an online social bot detection service which combines explainable machine learning techniques and crowdsourcing functionalities in a publicly available web tool. Unlike most existing work, *Bot-Detective* places extra effort on providing detailed explanations which reveal the reasons behind the classification of a Twitter account as a bot or not. The proposed approach conforms with the growing need for responsible data science and collects crowdsourcing data, aiming to improve user acceptance, results interpretability and efficiency in the overall bot detection pipeline.

In summary, the main contributions of this paper are the following:

(1) We propose a novel **explainable bot-detection service**, which, to the best of authors' knowledge, is the first one to offer interpretable, responsible, and AI driven bot identification in Twitter. Bot-Detective uses a Random Forest classifier and follows an explainable AI state of the art method to provide justifications of high granularity, since explanations are offered for all relevant features that have contributed in the *Bot-Detective*'s scoring.
(2) We provide a publicly available **bot detection Web service** which integrates an explainable machine learning framework along with users feedback functionality under an effective crowdsourcing mechanism. This service covers the growing demand for bot detection services and it offers extended crowdsourcing functionalities and XAI capabilities which advance existing state of the art tools, such as the Botometer.
(3) We share a **new labelled dataset**, annotated by exploiting Twitter's rules and existing tools, which we use to build the proposed service. The dataset consists of thousands of tweets collected using Twitter's official API and labelled with the use of Botometer and by taking into account that many of the authors of the posting accounts were, at a later time, deactivated by Twitter.

The remaining of this paper is organized as follows: Section 2 discusses the related work. Section 3 outlines our approach and methodology, while Section 4 presents the experimental results of using Bot-Detective. Finally, Section 5 includes the conclusions of this paper and future potentials.

[1]bot-detective.csd.auth.gr

2 RELATED WORK

The significance of the bot-detection challenge has drawn the attention of many scientists, leading to numerous publications proposing different methods. However, almost every approach does not include methods that provide explainable results. In the next sections, we provide a brief overview of already available methodologies regarding bot detection and upcoming challenges.

2.1 Bot Detection Methodologies

Most of bot detection methods rely on supervised ML techniques, favoring the use of one (or more) labelled datasets to train their machine learning algorithms and build an efficient model. These labelled datasets are usually human-annotated, however there are also datasets that have been created by using already available established models, crowdsourcing or automated methods.

Lee et al. [31] created a mixture of honeypot accounts i.e. accounts which are designed to lure spammers to interact with them and logged information about their profiles. They created a dataset of spammers and enriched it with normal users to train a classification model using both user and content features. A similar approach was also used by Stringhini et al. [45] who additionally tried to identify group of spammers (botnets) which were operated by the same user. Wang et al. [48] used crowdsourcing methods to identify bots in Facebook and although the method seemed to produce promising results, the fact that bots are not finite but keep rising limited the scalability of this effort. Crowdsourcing has been used within data annotation tasks in other approaches that we will present below.

The most popular approach[17], called BotOrNot and later on Botometer, builds on the dataset provided by [31], enriched with updated tweets for each labelled account. The model was used in a bot detection web service and currently receives more than 500 daily visits, supporting over 250,000 requests daily [50]. The breakthrough of this approach was the huge number of different features (user,sentiment,content,friends,network and temporal) that were used to train the model. The authors proposed a framework to extract this massive feature set and validated their results on a new annotated dataset in [47]. The results validated the efficiency of the proposed model and pointed out specific drawbacks. For example, the model performance was inferior in the new dataset, due to the fact that the model was trained on older bots, which apparently had different behavior and characteristics than the updated ones. In [50], the authors present the various available labelled bot datasets and show how the crowdsourcing functionalities of the Botometer service are used in order to retrain the model.

Other simpler, yet reliable, approaches that use a much lower volume of features to identify bots have been proposed such as Stweeler [24],[23] which uses a click-bait approach and collect user and tweet features to detect bots. Another methodology [7], identifies bots by checking the randomness of the account's screen name, while others [19] show that even with a careful selection of ten features, the trained model is quite efficient. Most of the work that has been already mentioned use simple ML algorithms, usually Random Forest, SVM, Decision Trees, etc., but there are also approaches that use Deep Learning or more sophisticated algorithms. Cai et al. [13] propose a behavior enhanced deep model on users which is then applied to a deep learning framework comprised of Convolutional and Long Short-Term memory (LSTM) neural nets to detect bots. Similarly, the authors of [30] propose the use of LSTM network exploiting the tweet's content and metadata along with contextual user features to identify if a tweet was posted by a bot. On the other hand Grimme et al.[25], propose a whole different approach regarding bot detection, stressing out the need to identify orchestrated attacks rather than lone players (distinct bot accounts). In summary, the above methodologies exploit a wide range of features and ML techniques, but despite that, seem to lack in addressing other challenges, which are further discussed in the following section.

2.2 Challenges and contributions

Even though there are many scientific efforts that have delivered methods which efficiently detect online social bots, there are still many open challenges as highlighted in the aforementioned research.

As we discussed above, feature engineering holds a very important role in ML-based bot detection methodologies. Despite the fact that we have reached a point where certain methodologies use more than 1,000 features to train their model [17], it is still uncertain if an increased number of features actually improves the trained model's efficiency. Moreover, such a large set of features, deeply affects the scalability of the bot detection systems as highlighted by Yang et al. [51]. At the same work, it is pointed that using different subsets of all the publicly available labelled datasets[2] improves the model's generalization, which designates another interesting issue. The performance of all the ML bot detection models is not the same to all existing datasets. There is a constant need for more datasets in an effort to make our training data cover all bot behavioral characteristics. The same observation is reached in [50] and in [16], which proceed to a more fine-grained classification of bots, providing separate datasets for each type of bots (celebrities, porn bots,etc.). Thus, a critical issue in the online social bot detection is the challenge to understand what characteristics actually constitute a social bot.

From the above related work discussion it is evident that there is no common definition of online social bots in the literature since characterizing an account as bot (or not) depends in varying features and views in the different existing efforts. This challenge is addressed by *Bot-Detective* which provides a publicly available service, which apart from providing an estimation of an account's probability of being a bot, it offers explanatory bot characterization details. Moreover, *Bot-Detective* introduces a new dataset and a new model, with specific and detailed human readable explanations of the reasons behind the tool's bot scoring estimation. The need for delivering human understandable justification in terms of online social bots scoring is along with the earlier research work [50],[51] but also with the AI research responsibility directives [33],[39]. Moreover, the feedback framework of *Bot-Detective*, supports the need of appraising the users' aspect on responsible data science and can also be used to improve the interpretability of the proposed models. Finally, the implemented *Bot-Detective* web service (bot-detective.csd.auth.gr) aims to highly contribute to the on-going

[2] https://botometer.iuni.iu.edu/bot-repository/datasets.html

battle against online social bots, which has already triggered global strong interest (evident by the wide adoption and high usage statistics of Botometer).

3 THE BOT-DETECTIVE APPROACH

In this section, we introduce *Bot-Detective* approach for designing and implementing an explainable bot-detection tool, which delivers interpretable and updateable results. As outlined in Figure 2, *Bot-Detective* consists of three major modules : (a) its ML model which is trained on a new labelled dataset and provides a bot estimation score in a scale 0-5; (b) the integration of an explanation model tailored to the specific use case, which aims to provide the justification behind the previous estimation in human-readable language and (c) the crowdsourcing module which is responsible for collecting users' feedback in concern to the estimated scores and the explanations that accompany them.

3.1 Bot-Detection Model

Feature selection and extraction is the cornerstone of the model implementation process. Previous work has shown that there is no magic number or perfect set of features. The performance of a model trained using a specific set varies from one dataset to another. Unlike many of the already available models, we leverage a smaller set of tweet-based features to train a model that will provide an estimation about whether this tweet has been posted by a bot or a human. The use of such a set is justified by the following reasons:

- The trained model will be used in a publicly available tool. This means that we need to save time at every possible operation to retrieve a near real-time score estimation.
- We leverage the Twitter API to collect and extract the tweet-based features. The rate limits imposed by Twitter already narrow down the real-time response of our service. Retrieving tweet-objects rather than user-objects is the most efficient way to exploit Twitter's API.
- Feature extraction is a time costly operation. The time needed for this process depends on the number of features (more features - more time).
- Previous research has proven to be quite efficient using even a set of ten features [19].

Based on relevant research, the final selection consists of 36 user and content features, see Table 1, which can be extracted or calculated using the Twitter's API so-called tweet object.

In order to train our model we use a new annotated dataset, which is presented in detail in Section 4. The whole process has been implemented in Python and using the Sklearn framework[36]. The purpose of Bot-Detective is predicting a score in the range of [0-5] for every instance. For the creation of our model we used an ensemble classifier and more specifically Random Forest. The interpretability of the classifier output is non-trivial for people with no relevant scientific background. The results do not refer specifically to the probability of an account being a bot, thus in order to resolve this we follow a similar approach to existing work [50] and continue with the calibration of our model. The calibration is necessary, due to the fact that our model seems to shift the predicted probabilities away from 0 and 1 [35], creating a sigmoid like shape as presented in Figure 6. The calibration method that we followed is that of Platt's [37], an out of the shelf method also known as sigmoid calibration. The final calibrated model returns the actual probability of an account being a bot. It takes as input an array of features and returns a final score, which is a linear transformation of the calibrated probability prediction that describes in a scale of 0-5 the probability of an account being a bot (5 means that most probably the account is a bot). More information with respect to the calibration and classification results can be found in Section 4.

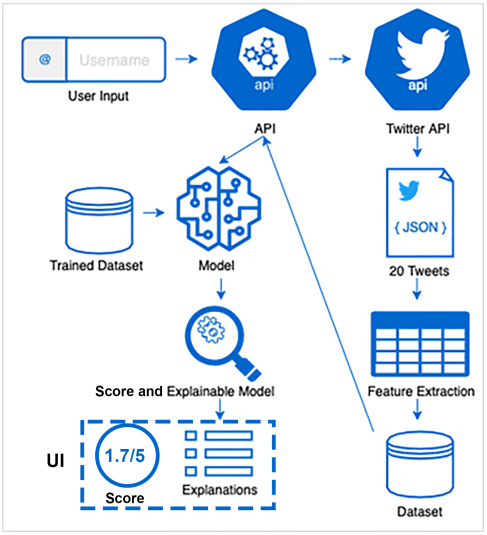

Figure 2: Bot-Detective's overall architecture.

3.2 Explainable Method

As already presented, the interpretability of bot-detection tools is a challenge that has not yet been addressed. Among others, *Bot-Detective* focuses on this exact challenge. The goal of the explainability module is to convert the predictions of the classifier into justifications understandable by humans.

Our module is based on a state-of-the-art method[26] called LIME[39]. LIME is an agnostic explanatory model, which means that it implements a function able to explain the results of our classifier, regardless the type of data or ML model. Using the trained model's prediction function, and a given point in its hyperplane, a local linear approximation is being performed, by sampling other instances around that point. By weighting samples around a given point, a linear model could approximate the generic model well, in a small vicinity. One can think of this process, as trying to use a form of linear interpolation around known instances, to find out where the probability function of the model changes the most. The process of retrieving these explanations is described below.

To build the explainer, we provide as input the trained dataset instances with their respective scores, the labels of the features and the indexes of the features that are categorical. The output of the explainer is an array with the weights of the features around the predicted instance. The weight values can either be negative, zero or positive. A negative value indicates that the feature affects the model into predicting a low score (non-bot account), while a positive

Table 1: List of the selected features used in our approach. All the features are either extracted or calculated by the tweet-object retrieved from the Twitter API.

Features								
Type:[**C**:*Content* - **U**:*User*] - Value:[**N**:*Number* - **B**:*Boolean*- **R**:*Ratio*]								
Name	**Type**	**Value**	**Name**	**Type**	**Value**	**Name**	**Type**	**Value**
URLs	C	N	Words	C	N	Numeric Characters	C	N
Hashtags	C	N	Symbols	C	N	Mentions	C	N
Times favourite	C	N	URLs-Words	C	R	Hashtags - Words	C	R
Times Retweeted	C	N	Media	C	N	Characters	C	N
Sensitive Tweet	C	B	Followers	U	N	Followees	U	N
Followers-Followees	U	R	Tweets	U	N	Lists	U	N
Favourite Tweets	U	N	Def. Profile	U	B	Profile Description	U	B
Verified	U	B	Def. Image	U	B	Profile location	U	B
Profile URL	U	B	Username Characters	U	N	URLs in description	U	N
Screen name characters	U	N	Characters in description	U	N	Bot word in username	U	B
Bot word in screen name	U	B	Bot word in description	U	B	hashtags in username	U	N
Numeric chars in username	U	N	Numeric chars in screename	U	N	hashtags in description	U	N

value indicates that the feature affected the model into predicting a high score (bot account). Features with a zero-weight value do not affect the model about the prediction of the specific account's score. However, in the direction of providing actual explanations,so that the computed features can have a clear meaning, we used a function that maps each feature to manually engineered explanations written in natural language. This rule based approach matches specific sentences to each positive or negative weight value accordingly. The sentences have been manually generated and most of them include detailed information regarding the average of this feature value for bot or non-bot accounts in comparison to the feature value of the examined tweet. A simplified example is presented in Figure 3.

3.3 Bot-Detective as a Web Service

The deployment of an explainable bot-detection service is among the main contributions of *Bot-Detective*. The web service that will be presented below aims to advance the already established tools and provide an alternative for future use. The architecture of the developed service follows the client-server model. The client provides as input the Twitter handle of the account he/she wants to check and the *Bot-Detective* web service makes a call to the *Bot-Detective* API, which using the official Twitter API, retrieves the last twenty tweets of the account. At this point, it should be mentioned that for the moment this service does not require every user to have a Twitter account, but uses specific Twitter Application credentials.

As we have already mentioned, the tweet-object contains information for both the content of the tweet and the user as well. The call to the *Bot-Detective* API, actually activates a single endpoint which is responsible for the tweet-object collection, the feature extraction for each tweet, the feature explanations and the average score prediction. Due to the fact that our model is tweet-based, the final bot-score is the mean score of the tweets that have been collected previously. In the case of feature-based explanations we follow a different approach. Considering that each one of the collected tweets contains different content and taking into consideration that the user information may vary from one tweet to another (for example, a user may change the screen name of his account at any time), the array with the negative or positive feature weights produced by the explainer, may also be different for every distinct tweet. For this reason, we only take into account the feature weights that are present in at least half of the collected tweets. The final output is a JSON file containing all the information as presented above.

The system's User Interface was developed using the JavaScript web programming language, and the ReactJS library. The one-screen clean design of the application opts for a better user experience. The screen includes an input form, where users can input Twitter handles. When submitting an input, the application makes a request to the server on the background. The server returns the results in JSON format, that are then presented to the user in the screen.

In addition to the explainability extension, the web service also provides advanced crowdsourcing functionalities. More specifically, the user can provide feedback for the estimated score and ,unlike existing work, also provide feedback for the quality of the interpretation of the results. The available feedback form collects information about the validity of the prediction; the users can share their own opinion and classify the account as Human, Bot, Human using automation or as an Organization. Moreover, they can indicate whether they agree with the explainable results, assess the quality of the explanations and even provide in simple text how these can be improved. The collected feedback is stored for future improvements of the model and the explainer as well.

4 DATASET AND EXPERIMENTATION

This section includes a detailed presentation of a new dataset, on which we have trained the model responsible for the bot-score estimation, details about our model and the results of an in-situ evaluation of the interpretability of *Bot-Detective*.

4.1 Dataset

With respect to the constant need for new datasets in this research area, we decided to create, use and share a new one, with the help of

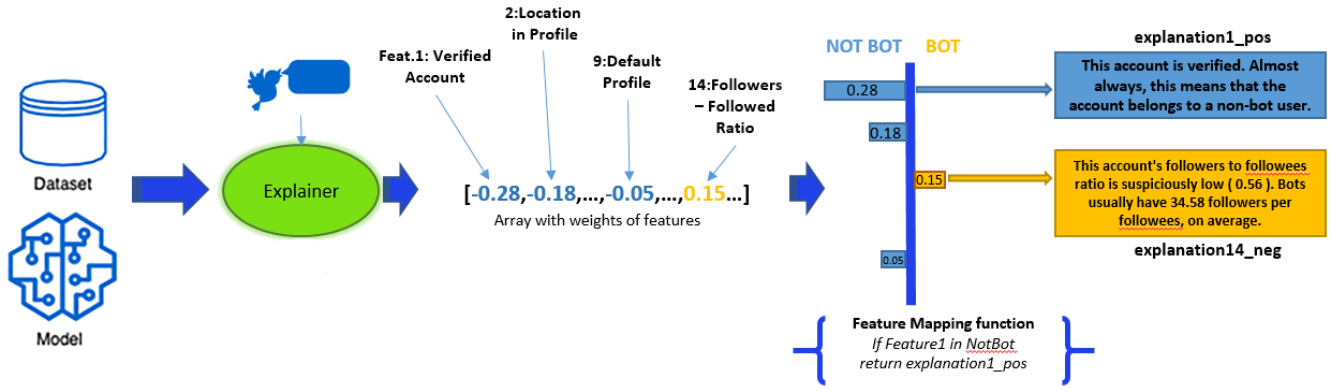

Figure 3: Explanable AI module outline : The stored explainer takes as input the features of each tweet. It returns an array of weights for the available features. The feature mapping function, depending on the weight values, matches each feature to a manually engineered explanation.

which we trained our model. To that end, using the official Twitter API, we started collecting public tweets for a period of 20 days, focused on the topic of cryptocurrencies. More specifically, we used the hashtags #crypto, #cryptocurrency and #ico as filtering terms. The main motivation behind this selection was the proliferation of cryptocurrency giveaway scams [29]. Apart from this, it is also a very trending topic in Twitter, which most probably would provide a sizeable pool of bots. After collecting the tweets from the Twitter API, we filtered out the tweets that were not in English. Combining multiple languages would complicate the feature extraction process, since the performance of most available NLP solutions is inferior for other languages. The next step was to find a way of evaluating the state of each tweet's author. Since our resources could not allow the manual investigation of 2 million tweets and their authors, the idea was to use a third-party app that would provide a fairly confident ground truth about the likelihood of a user being a bot. For this purpose, we used Botometer which provides a publicly available API[3] which has proven to be quite reliable. The process of labelling the collected tweets using the Botometer API was completed in ten days, using three different accounts to speed up the process. For a large number of tweets, the API could not provide answers about their authors' scores since Twitter had already identified these accounts as spam and deactivated them. Assuming that these accounts would receive a high bot probability score from Botometer, they were assigned with a score in range of [4,5] and with an additional 'deleted' label to distinguish them from other accounts. Since Botometer also linearly transforms the bot likelihood score in a [0,5] scale it is safe to assume that accounts with score higher than 2.5 have a higher probability to be bots. The distribution of bot-scores is presented in Figure 4.

4.2 Model Evaluation

In order to build our ML model we experimented with a wide range of supervised learning algorithms. More specifically, our experiments included the following: AdaBoost, Support Vector Machines,

[3]https://botometer.osome.iu.edu/api

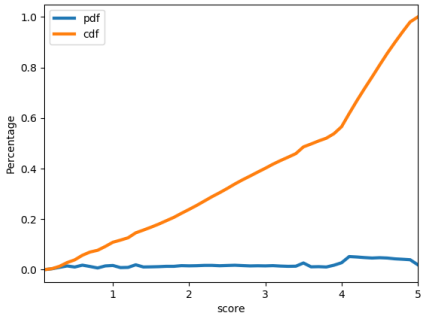

Figure 4: Cumulative and Probability distribution of the dataset bot scores

Multi-Layer Perceptrons, Naive Bayes, Random Forest. However, the classifier that we have decided to use is an ensemble supervised learning Random Forest which produced the most accurate results and which is also used from other state of the art approaches, see[17]. It has been trained on the dataset that we presented in the previous section, after splitting it into train and test data. In order to define the parameters of the classifier we used an exhaustive search method with 5-fold cross validation (GridSearchCV), setting a range of [100,200] in steps of 10 for the number of trees and the sequence of [None,5,10,15] for their maximum depth. Experiments shown that the best combination was using 150 trees and maximum depth equal to the default value, which means that nodes are expanded until all leaves are pure or until all leaves contain less than 2 samples. Our model seems to perform great, scoring a value of 0.99 for the AUROC metric (Area Under the Receiver Operating Characteristic curve), see Figure 5. However, since our training data are focused on a specific topic, with possibly bot-users having a similar behavior, it is still under question if the model's performance will be similar in other datasets as well. As described

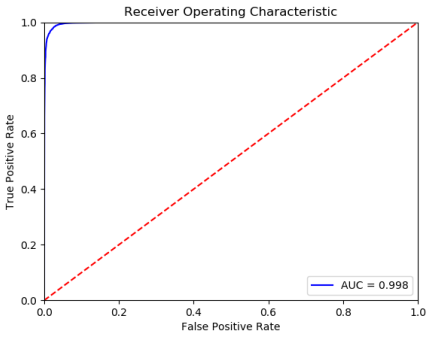

Figure 5: ROC curve of our model, measured using the AUROC metric

in subsection 3.1 we also applied a calibration method to our model, in order to receive actual probabilities as output. The calibration results are presented in Figure 6, where we can observe difference between the non-calibrated and calibrated results.

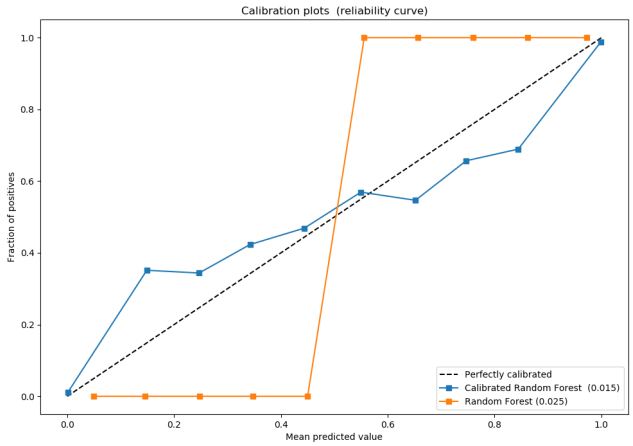

Figure 6: Our model is much more reliable, after the calibration process

4.3 Interpretability of results

The main improvement of *Bot-Detective* over other available similar solutions is that it provides human-readable explanations with respect to the bot estimation score. To achieve this, we have used an explainable AI method as presented in section 3.2. Towards the evaluation of the rule-based interpretations of the result, we conducted a small-scale survey in our department. We asked 50 Computer Science students (post-graduates and PhDs) to fill in a short questionnaire, assessing the comprehension and the quantity of the provided explanations. The results are presented in Figure 7. Based on the answers of the participants, it is obvious that the number of visualized explanations is rather high. This happens because we have decided to include all the features whose weights, as calculated by the explainer, are positive or negative (we omit

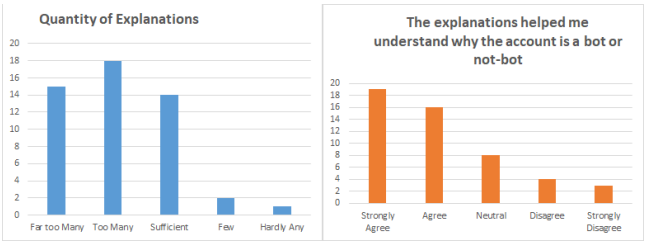

Figure 7: Results of the survey, regarding the efficiency of the provided explanations. Most users agreed that their quantity is rather high and that they helped them comprehend whether an account is a bot or not.

the neutral ones). However, it seems that a better approach would be to include only those features,and as a consequence the explanations as well, whose weight is above a certain threshold. When the participants were questioned whether the explanations helped them though, the results are quite optimistic; 70% of the involved users stated that the explanations helped them understand why an account is a bot or not.

Finally, every participant was asked to use our model's bot-estimation score for one Twitter account each. The results were recorded and compared to the ones produced by Botometer. Figure 8 shows the differences in the bot-score estimation for the 50 accounts. We selected the top differences in the accounts that had

Figure 8: Difference between Botometer and Bot-Detective scores. The green watermarked scores are bots included in botwiki misclassified by Botometer, unlike Bot-Detective which makes a correct prediction.

been checked and asked the users to provide feedback regarding the state of these accounts. 85% of the accounts were misclassified by the Botometer app; Most of them were bot accounts that can be found in botwiki.org , a pair of them belonged to the users themselves (they were classified as bots) and one to a news-agency which was misclassified by Bot-Detective. At this point we should stress out that the sample is really small and can only indicate a better performance for these specific accounts.

5 CONCLUSIONS AND FUTURE WORK

In this paper, we propose *Bot-Detective*, a novel online Twitter bot-detection service focused on providing explainable and accurate

results. To the best of our knowledge, Bot-Detective is the first bot-detection service to provide these extended functionalities. In our approach, we use a state-of-the-art explainable AI method tailored to the needs of the specific challenge and we introduce a new annotated dataset on which we train our ML model. Our method conforms with the emerging need for explainable machine learning models, more annotated datasets and alternative bot-detection services to the already established ones. We also introduce a crowdsourcing approach with respect to the reasoning behind our results, hoping to lay the groundwork of improved and human-readable ML interpretations. Although the system has been deployed and is publicly available, we prioritized on preparing the ground for integrating precise interpretations on accurate ML models. Our results are a starting point and can definitely be improved extensively.

Throughout our study, we have identified many challenges that need to be addressed in the future. First of all,As in other methodologies, our research also showcases that the segregation of accounts to bots and non-bots is not sufficient, although we are making an effort to provide understandable justifications in the process of bot identification. We need to extend the volume of our training data, starting by using and evaluating datasets that have also been utilized by state-of-the-art approaches. We also have to identify different selections of features which may perform better, depending on the type of bot. The explanations provided by our service, need to be further improved in order to be more generalized or more focused, where needed. Furthermore, taking into consideration the wide variety of explainable AI methods, in the future we plan to use, evaluate and compare some of them in the specific use case. Finally, focusing on our solution as a service, there are multiple forthcoming improvements. Initially the service will start using each user's Twitter credentials to proceed with the evaluation. Also, based on the monitored traffic, we may need to improve the scalability of our service. Future plans, also include changes in the UI in order to improve the user experience and the development of an open API that will provide bot-score estimations and explanations accordingly.

ACKNOWLEDGMENTS

This research has been co-financed by the European Union and Greek national funds through the Operational Program Competitiveness, Entrepreneurship and Innovation, under the call RESEARCH – CREATE – INNOVATE (Project Code: T2EDK-03898), as well as from the H2020 Research and Innovation Programme under Grant Agreement No. 875329.

REFERENCES

[1] [n.d.]. https://botwiki.org/.
[2] 2016. General Data Protection Regulation - Right to explanation. https://www.privacy-regulation.eu/en/r71.htm.
[3] 2020. Twitter Co-Founder Jack Dorsey Answers Twitter Questions From Twitter | Tech Support | WIRED. https://youtu.be/de8wRd2TQQU?t=99.
[4] Norah Abokhodair, Daisy Yoo, and David W McDonald. 2015. Dissecting a social botnet: Growth, content and influence in Twitter. In *Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing*. 839–851.
[5] Luca Maria Aiello, Martina Deplano, Rossano Schifanella, and Giancarlo Ruffo. 2012. People are strange when you're a stranger: Impact and influence of bots on social networks. In *Sixth International AAAI Conference on Weblogs and Social Media*.
[6] Jonathon M Berger and Jonathon Morgan. 2015. The ISIS Twitter Census: Defining and describing the population of ISIS supporters on Twitter. *The Brookings project on US relations with the Islamic world* 3, 20 (2015), 4–1.
[7] David M Beskow and Kathleen M Carley. 2019. Its all in a name: detecting and labeling bots by their name. *Computational and Mathematical Organization Theory* 25, 1 (2019), 24–35.
[8] Alessandro Bessi, Mauro Coletto, George Alexandru Davidescu, Antonio Scala, Guido Caldarelli, and Walter Quattrociocchi. 2015. Science vs conspiracy: Collective narratives in the age of misinformation. *PloS one* 10, 2 (2015), e0118093.
[9] Alessandro Bessi and Emilio Ferrara. 2016. Social bots distort the 2016 US Presidential election online discussion. *First Monday* 21, 11-7 (2016).
[10] Yazan Boshmaf, Ildar Muslukhov, Konstantin Beznosov, and Matei Ripeanu. 2013. Design and analysis of a social botnet. *Computer Networks* 57, 2 (2013), 556–578.
[11] Russell Brandom. 2017. One in five ad-serving websites is visited exclusively by fraud bots. https://www.theverge.com/2017/5/24/15681080/ad-fraud-websites-traffic-bots-white-ops-report.
[12] David A Broniatowski, Amelia M Jamison, SiHua Qi, Lulwah AlKulaib, Tao Chen, Adrian Benton, Sandra C Quinn, and Mark Dredze. 2018. Weaponized health communication: Twitter bots and Russian trolls amplify the vaccine debate. *American journal of public health* 108, 10 (2018), 1378–1384.
[13] Chiyu Cai, Linjing Li, and Daniel Zengi. 2017. Behavior enhanced deep bot detection in social media. In *2017 IEEE International Conference on Intelligence and Security Informatics (ISI)*. IEEE, 128–130.
[14] Despoina Chatzakou, Nicolas Kourtellis, Jeremy Blackburn, Emiliano De Cristofaro, Gianluca Stringhini, and Athena Vakali. 2017. Mean birds: Detecting aggression and bullying on twitter. In *Proceedings of the 2017 ACM on web science conference*. 13–22.
[15] Zi Chu, Steven Gianvecchio, Haining Wang, and Sushil Jajodia. 2012. Detecting automation of twitter accounts: Are you a human, bot, or cyborg? *IEEE Transactions on Dependable and Secure Computing* 9, 6 (2012), 811–824.
[16] Stefano Cresci, Roberto Di Pietro, Marinella Petrocchi, Angelo Spognardi, and Maurizio Tesconi. 2017. The paradigm-shift of social spambots: Evidence, theories, and tools for the arms race. In *Proceedings of the 26th international conference on world wide web companion*. 963–972.
[17] Clayton Allen Davis, Onur Varol, Emilio Ferrara, Alessandro Flammini, and Filippo Menczer. 2016. Botornot: A system to evaluate social bots. In *Proceedings of the 25th international conference companion on world wide web*. 273–274.
[18] Chad Edwards, Autumn Edwards, Patric R Spence, and Ashleigh K Shelton. 2014. Is that a bot running the social media feed? Testing the differences in perceptions of communication quality for a human agent and a bot agent on Twitter. *Computers in Human Behavior* 33 (2014), 372–376.
[19] Emilio Ferrara. 2017. Disinformation and social bot operations in the run up to the 2017 French presidential election. *arXiv preprint arXiv:1707.00086* (2017).
[20] Emilio Ferrara. 2019. The history of digital spam. *Commun. ACM* 62, 8 (2019), 82–91.
[21] Emilio Ferrara, Onur Varol, Clayton Davis, Filippo Menczer, and Alessandro Flammini. 2016. The rise of social bots. *Commun. ACM* 59, 7 (2016), 96–104.
[22] Juliette Ferraro. 2019. Battling The Bots On Twitter. https://blog.thomasnet.com/social-media-marketing-twitter-bots.
[23] Zafar Gilani, Reza Farahbakhsh, and Jon Crowcroft. 2017. Do bots impact Twitter activity?. In *Proceedings of the 26th International Conference on World Wide Web Companion*. 781–782.
[24] Zafar Gilani, Liang Wang, Jon Crowcroft, Mario Almeida, and Reza Farahbakhsh. 2016. Stweeler: A framework for twitter bot analysis. In *Proceedings of the 25th International Conference Companion on World Wide Web*. 37–38.
[25] Christian Grimme, Dennis Assenmacher, and Lena Adam. 2018. Changing perspectives: Is it sufficient to detect social bots?. In *International Conference on Social Computing and Social Media*. Springer, 445–461.
[26] Riccardo Guidotti, Anna Monreale, Salvatore Ruggieri, Franco Turini, Fosca Giannotti, and Dino Pedreschi. 2018. A survey of methods for explaining black box models. *ACM computing surveys (CSUR)* 51, 5 (2018), 1–42.
[27] Aditi Gupta, Hemank Lamba, and Ponnurangam Kumaraguru. 2013. $1.00 per rt# bostonmarathon# prayforboston: Analyzing fake content on twitter. In *2013 APWG eCrime researchers summit*. IEEE, 1–12.
[28] Tim Hwang, Ian Pearce, and Max Nanis. 2012. Socialbots: Voices from the fronts. *interactions* 19, 2 (2012), 38–45.
[29] Ryan Mac Jane Lytvynenko. 2018. General Data Protection Regulation - Right to explanation. https://www.buzzfeednews.com/article/janelytvynenko/twitter-cryptocurrency-scams-verified-accounts-russia-target.
[30] Sneha Kudugunta and Emilio Ferrara. 2018. Deep neural networks for bot detection. *Information Sciences* 467 (2018), 312–322.
[31] Kyumin Lee, Brian David Eoff, and James Caverlee. 2011. Seven months with the devils: A long-term study of content polluters on twitter. In *Fifth international AAAI conference on weblogs and social media*.
[32] Tetyana Lokot and Nicholas Diakopoulos. 2016. News Bots: Automating news and information dissemination on Twitter. *Digital Journalism* 4, 6 (2016), 682–699.
[33] Don Monroe. 2018. AI, explain yourself.

[34] Susan Moore. 2019. Gartner Identifies Top 10 Data and Analytics Technology Trends for 2019. https://www.gartner.com/en/newsroom/press-releases/2019-02-18-gartner-identifies-top-10-data-and-analytics-technolo.

[35] Alexandru Niculescu-Mizil and Rich Caruana. 2005. Predicting good probabilities with supervised learning. In *Proceedings of the 22nd international conference on Machine learning*. 625–632.

[36] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* 12 (2011), 2825–2830.

[37] John Platt et al. 1999. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in large margin classifiers* 10, 3 (1999), 61–74.

[38] Jacob Ratkiewicz, Michael D Conover, Mark Meiss, Bruno Gonçalves, Alessandro Flammini, and Filippo Menczer Menczer. 2011. Detecting and tracking political abuse in social media. In *Fifth international AAAI conference on weblogs and social media*.

[39] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. " Why should i trust you?" Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*. 1135–1144.

[40] Jon Russell. 2018. Twitter is (finally) cracking down on bots. https://techcrunch.com/2018/02/22/twitter-is-finally-cracking-down-on-bots/.

[41] Saiph Savage, Andres Monroy-Hernandez, and Tobias Höllerer. 2016. Botivist: Calling volunteers to action using online bots. In *Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work & Social Computing*. 813–822.

[42] Chengcheng Shao, Pik-Mai Hui, Lei Wang, Xinwen Jiang, Alessandro Flammini, Filippo Menczer, and Giovanni Luca Ciampaglia. 2018. Anatomy of an online misinformation network. *PloS one* 13, 4 (2018), e0196087.

[43] Galen Stocking and Nami Sumida. 2018. Social media bots draw public's attention and concern. *Pew Research Center, Washington, DC* (2018).

[44] Simone Stolzoff. 2018. The problem with social media has never been about bots. It's always been about business models. https://qz.com/1449402/how-to-solve-social-medias-bot-problem/.

[45] Gianluca Stringhini, Christopher Kruegel, and Giovanni Vigna. 2010. Detecting spammers on social networks. In *Proceedings of the 26th annual computer security applications conference*. 1–9.

[46] Milena Tsvetkova, Ruth García-Gavilanes, Luciano Floridi, and Taha Yasseri. 2017. Even good bots fight: The case of Wikipedia. *PloS one* 12, 2 (2017).

[47] Onur Varol, Emilio Ferrara, Clayton A Davis, Filippo Menczer, and Alessandro Flammini. 2017. Online human-bot interactions: Detection, estimation, and characterization. In *Eleventh international AAAI conference on web and social media*.

[48] Gang Wang, Manish Mohanlal, Christo Wilson, Xiao Wang, Miriam Metzger, Haitao Zheng, and Ben Y Zhao. 2012. Social turing tests: Crowdsourcing sybil detection. *arXiv preprint arXiv:1205.3856* (2012).

[49] Stefan Wojcik, Solomon Messing, Aaron Smith, Lee Rainie, and Paul Hitlin. 2018. Bots in the Twittersphere. *Pew Research Center. Retrieved May* 22 (2018), 2018.

[50] Kai-Cheng Yang, Onur Varol, Clayton A Davis, Emilio Ferrara, Alessandro Flammini, and Filippo Menczer. 2019. Arming the public with artificial intelligence to counter social bots. *Human Behavior and Emerging Technologies* 1, 1 (2019), 48–61.

[51] Kai-Cheng Yang, Onur Varol, Pik-Mai Hui, and Filippo Menczer. 2019. Scalable and generalizable social bot detection through data selection. *arXiv preprint arXiv:1911.09179* (2019).