Detecting Bot Accounts on Twitter by Measuring Message Predictability

Notebook for PAN at CLEF 2019

Piotr Przybyła

Institute of Computer Science, Polish Academy of Sciences
Warsaw, Poland
piotr.przybyla@ipipan.waw.pl

Abstract We present a method for deciding whether a given Twitter user is a bot or a human based on the textual content of their feed. Since messages published from the same bot account frequently follow simple, repetitive patterns, we propose to recognise such accounts by measuring message predictability. This is performed using two language modelling solutions: based on a linear model operating on hand-crafted features or a pre-trained neural language model (BERT). When evaluated within PAN 2019 bot detection shared task, our solution reaches an accuracy of 91% for tweets in English and 88% for tweets in Spanish.

1 Introduction

The problem of assessing credibility of content published and shared on the web remains one of the important challenges brought by the growing importance of the Internet. Many undesirable phenomena of the new communication platforms, such as social media, stem from the fact that their users have limited or no knowledge on the identity of their interlocutors. This anonymity combined with digital interfaces of online services makes it possible for computer programs to automatically generate messages that would look similar to those composed by human users.

Such applications, called *bots*, could serve good purposes as a means for automatic distribution of information valuable for a wider audience, e.g. a weather forecast. They could also be used to send undesired advertisement of products or services, similarly to email spam messages. Finally, some exploit bots to manipulate public opinion and heat up discussions on controversial issues for political purposes, as it has been demonstrated in the past [15,1,7,2]. In any of the above cases, it seems beneficial for the quality of online conversations to make their participants aware, which of the accounts they interact with are bots.

Several attempts have been made to use machine learning (ML) to automatically detect bots in popular social media services, especially Twitter. Most of these solution rely on observation of a given account in terms of social interactions specific to Twitter: number of following and followed users, frequency of messages, occurrences of mentions, retweets, hashtags etc. The textual content of the messages in the user's feed seems to be less commonly analysed in this task.

In our study we aim to contribute to this relatively under-explored direction by building a bot detection algorithm, which recognises bot accounts based solely on the textual content of the messages in their Twitter feed. The feature that is used to differentiate these accounts is message *predictability*. We notice that while bot users publish different types of content, the messages within one account usually follow a single, repetitive pattern. Therefore, the more easily a message could be predicted given the rest of the feed, the more likely the corresponding account is to be a bot one. Humans appear less predictable and post more diverse messages.

In order to measure message predictability, we employ two language models: one based on logistic regression using a collection of manually designed features, and one based on a pre-trained neural model. We aggregate the predictability values of all messages from an account to compute its feed predictability measures, which in turn are used as features for a final bot detection model. The model is trained and evaluated using the datasets of a *Bots and Gender Profiling* shared task [14] at PAN workshop at CLEF 2019. The source code and models of our approach are available online[1].

2 Related Work

Social bots have been investigated thoroughly [8] due to the role they play [17] in the proliferation of non-credible information through the web and social media [10] and increasing political polarisation [18], both processes motivating very active research. Nevertheless, accounts using bots remain challenging to identify, partly because the underlying applications actively try to prevent that (i.e. by simulating a human-like behaviour), as in many cases such identification would defeat their purpose.

Botometer [19] is the best known bot detection program. It uses a random forest and a plethora of features describing an account, including the associated user profile; number and user profiles of 'follower' and 'followed' accounts; contents and sentiment of messages in the feed; the network of retweets and mentions; tweeting time patterns. The tool runs as a service allowing internet users to check a selected Twitter account [20] and has been employed to analyse the influence of Russian bots on the discussion around vaccination [2] and the contribution of bots in spreading low-credibility content [17]. Other solutions [6] are based on the sentiment of messages from the analysed account compared to either its followers and or all users discussing the topic of interest (elections in India). Neural sequential models (LSTM) have also been applied to learn from a behaviour of the user, represented through sequence of words in messages and their posting times [3].

To sum up, the methods that were applied to the task so far achieve good performance by combining some representation of the content of a Twitter feed with its social context. Nevertheless, the research suggests the content component may be sufficient on

[1] https://github.com/piotrmp/bothunter

its own, as even a simple representation of it (through part-of-speech tags) constitutes a strong group of features in *Botometer* [19]. A content-based bot detection model could also be seen as a step towards a multi-platform solution, as it would be less dependent on Twitter-specific social features.

3 Task

The solution described in this paper is evaluated within the *Bots and gender profiling* shared task [14] of the PAN workshop [4] at the CLEF 2019 (*Conference and Labs of the Evaluation Forum*) conference[2]. We participate in the bot profiling problem, where the goal is to recognise whether a given Twitter feed was produced by a bot or a human. The task consists of two sub tasks, one involving tweets in English, the other one in Spanish.

Each author is represented through text of 100 messages from their feed, with no information on social media context, apart from the in-text mentions of other user names. Figure 1 shows fragments of two feeds from the training set from a human (left) and a bot (right) account. The whole training dataset includes 4120 such feeds in English and 3000 in Spanish. During the evaluation, participants of the shared task submit their software to TIRA infrastructure [12], where it is automatically executed on hidden test data.

4 Methods

In order to decide whether a given Twitter feed comes form a bot account, we perform a three-step procedure.

Firstly, we compute a predictability score for each message in the feed, which measures how well this message could be predicted in the context of the feed. This is essentially a language modelling task and is performed using two ML models: a logistic regression on hand-crafted features (LASSO) or a pre-trained and fine-tuned neural language representation model (BERT).

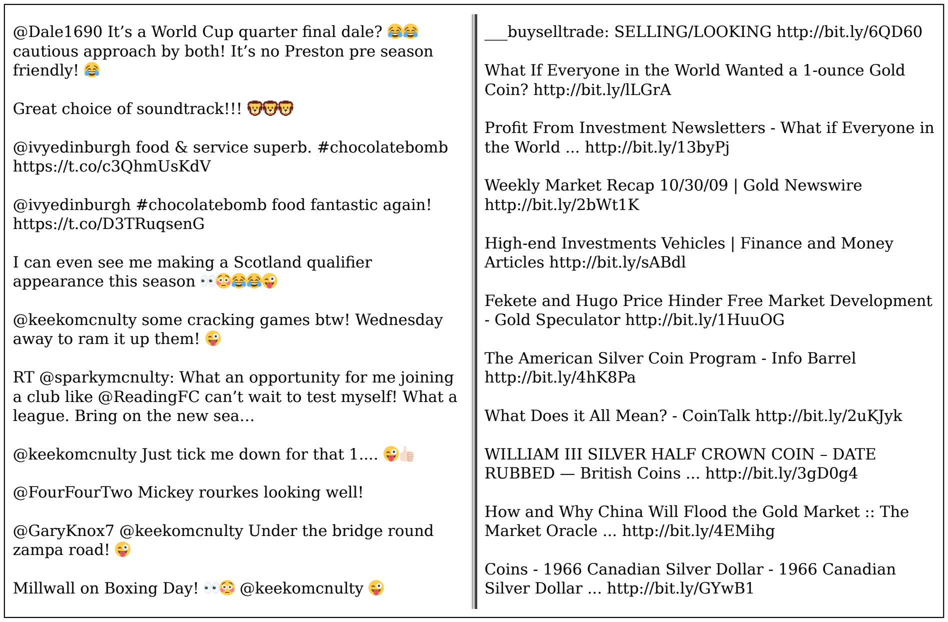
Secondly, we gather the predictability scores of all the messages in the feed and compute several measures of their distribution, such as mean or standard deviation, that describe the overall feed predictability. In case of bot accounts it could be expected that the messages will be similar and easy to predict, which corresponds to higher mean predictability.

Finally, the measured values are treated as features for a logistic regression model, trained on all gold-standard cases of bot and human feeds, which returns a score indicating the likelihood of a given feed coming from a bot account. The same procedure is applied to tweets in English and Spanish (with separate models).

4.1 LASSO Model of Predictability

Measuring message predictability using this approach requires converting all messages to features and building linear regression models, one for each feed, predicting whether

[2] http://clef2019.clef-initiative.eu/



@Dale1690 It’s a World Cup quarter final dale? 😂😂 cautious approach by both! It’s no Preston pre season friendly! 😂

Great choice of soundtrack!!! 🦁🦁🦁

@ivyedinburgh food & service superb. #chocolatebomb https://t.co/c3QhmUsKdV

@ivyedinburgh #chocolatebomb food fantastic again! https://t.co/D3TRuqsenG

I can even see me making a Scotland qualifier appearance this season 👀😳😂😂😜

@keekomcnulty some cracking games btw! Wednesday away to ram it up them! 😜

RT @sparkymcnulty: What an opportunity for me joining a club like @ReadingFC can’t wait to test myself! What a league. Bring on the new sea...

@keekomcnulty Just tick me down for that 1.... 😜👍🏻

@FourFourTwo Mickey rourkes looking well!

@GaryKnox7 @keekomcnulty Under the bridge round zampa road! 😜

Millwall on Boxing Day! 👀😳 @keekomcnulty 😜

___buyselltrade: SELLING/LOOKING http://bit.ly/6QD60

What If Everyone in the World Wanted a 1-ounce Gold Coin? http://bit.ly/lLGrA

Profit From Investment Newsletters - What if Everyone in the World ... http://bit.ly/13byPj

Weekly Market Recap 10/30/09 | Gold Newswire http://bit.ly/2bWt1K

High-end Investments Vehicles | Finance and Money Articles http://bit.ly/sABdl

Fekete and Hugo Price Hinder Free Market Development - Gold Speculator http://bit.ly/1HuuOG

The American Silver Coin Program - Info Barrel http://bit.ly/4hK8Pa

What Does it All Mean? - CoinTalk http://bit.ly/2uKJyk

WILLIAM III SILVER HALF CROWN COIN – DATE RUBBED — British Coins ... http://bit.ly/3gD0g4

How and Why China Will Flood the Gold Market :: The Market Oracle ... http://bit.ly/4EMihg

Coins - 1966 Canadian Silver Dollar - 1966 Canadian Silver Dollar ... http://bit.ly/GYwB1

Figure 1. Partial Twitter feeds of two training examples from the PAN training set, labelled as human (left) and bot (right).

a given message belongs to this feed or not. The predictability score for a message is obtained by computing the response of the model corresponding to the feed the message is taken from. High value of the score indicate that the message is easily recognised by the LASSO model of the feed and could be considered predictable.

Message Features The purpose of the prepared feature set was to enable the linear models to learn properties characteristic to each feed, but at the same time to avoid overfitting. The features describing a message include:

- length of the message text (number of characters),
- number of tokens in the message,
- number of occurrences of unigrams, bigrams and trigrams of word tags, excluding those with less than 10 occurrences in the corpus.

Splitting messages into tokens is performed using the *Stanford CoreNLP* [11] toolkit.

Because of the informal language style of tweets, we have decided that the role of word tags should not be played by part of speech categories, but we use a Twitter-specific tagging scheme instead. The scheme assigns each token one of the following tags:

- `@mention` for Twitter mentions,
- `#hashtag` for hashtags,
- `RT` for retweet indications,

- `URL` for web addresses,
- `number` for numerical expressions,
- `w` for individual lowercase letters,
- `W` for individual uppercase letters,
- `wooord` for alphabetic strings containing a character repeated three times,
- `word` for alphabetic strings in lower case,
- `WORD` for alphabetic strings in upper case,
- `Word` for alphabetic strings with only the first letter in upper case,
- `wOrD` for alphabetic strings with other casing scheme,
- `:` for colons,
- `(` for opening brackets (regular, square or curly),
- `)` for closing brackets (regular, square or curly),
- `-` for hyphens, minus signs, dashes and tildes,
- `'` for all quotation marks and apostrophes,
- `.` for full stops,
- `,` for commas,
- `!` for exclamation marks,
- `?` for question marks,
- `/` for slashes,
- `E` for individual emojis,
- `EE` for strings of emojis,
- $\star$ for any of the following characters: $, +, &, <, >, %, *, #,
- $\star\star$ for strings including the above characters,
- `...` for ellipsis (marking a tweet extending beyond a character limit),
- `w0rd` for alphanumeric strings,
- `<?>` for any token not matching any of the above.

Additionally, the tags `^` and `$` at the beginning and end, respectively, of the tag sequence, could also be included in the n-grams.

Computing Predictability Scores In order to assess the predicability of the i-th tweet from the j-th feed ($t_{i,j}$), the following procedure is followed:

1. Build a dataset consisting of all N tweets $t_{1,j} \ldots t_{N,j}$ from the feed in question and N additional tweets from the training set $t_1^* \ldots t_N^*$ (coming from other users than j), represented through their features,
2. Create a class variable y, such that $y(t_{i,j}) = 1$ and $y(t_i^*) = 0$,
3. Build a logistic regression model on the dataset using the `glmnet` [9] package in *R* [13], using L_1 regularisation (LASSO) and selecting the λ parameter through Bayesian information criterion [16],
4. Apply the model to the same dataset and collect its response $\hat{y}(t_{i,j})$ as predicability score for the i-th tweet.

4.2 BERT Model of Predictability

The second approach exploits a pre-trained deep language model BERT [5]. One of the tasks BERT was trained on is predicting whether two given sentences could follow each other in a text, which resembles our problem of assessing likelihood a message given other messages from the same feed. Therefore, we use a pre-trained BERT model and fine-tune it to the task by providing examples of subsequent messages from the training data. In order to measure how predictable a given feed is, we check how the tuned BERT model responds to pairs of messages from it. Similarly to the LASSO approach, high values of output mean that our model has found messages from the given account likely to occur in the same context and predictable.

Fine-tuning To fine-tune the BERT model, we first prepare a dataset including pairs of messages coming either from the same feed or from different ones. Specifically, for the i-th message from the j-th feed ($t_{i,j}$), two training cases could be generated:

- $x = (t_{i,j}, t_{i^*,j}), \quad y = 1$, comparing the message with another one (randomly selected) from the same feed ($i^* \neq i$),
- $x = (t_{i,j}, t_{i,j^*}), \quad y = 0$, comparing the message with one from another (randomly selected) feed ($j^* \neq j$).

In order to limit the fine-tuning time, the training cases are only generated for 20% of the messages.

Next, we load a BERT model from the checkpoints available online (for English: `uncased_L-12_H-768_A-12`, Spanish: `multi_cased_L-12_H-768_A-12`) and perform the fine-tuning process according to BRAT documentation[3] and using code shared as a *Google Colab* notebook[4].

Obtaining the Predictability Scores Using the BERT next sentence prediction model, now fine-tuned to detect Twitter messages coming from the same feed, we can assess the predictability of any given feed. To do that, we generate test cases according to the procedure described above, except only pairs from the same feed are used and no messages are skipped, and observe the model response. If the returned values are close to 1, it means it was easy for the model to recognise the similarities between the tweets. Therefore, the model response to the pair $(t_{i,j}, t_{i^*,j})$ is treated as predictability score for message $t_{i,j}$.

4.3 Feed Predictability Features

The message predicability scores are aggregated as feed predictability features using seven measures:

- mean,

[3] https://github.com/google-research/bert

[4] https://colab.research.google.com/github/tensorflow/tpu/blob/master/tools/colab/bert_finetuning_with_cloud_tpus.ipynb

- median,
- standard deviation,
- fraction of scores above 0.9,
- fraction of scores below 0.1,
- skewness (or 0, if undefined),
- kurtosis (or 100, if undefined).

4.4 Bot Detection Model

Finally, once each feed is described by predicability features, we can build a prediction model that would use them to decide whether the feed comes from a bot account or not. We use logistic regression in *R* [13] again, but this time the regularisation is not necessary due to low number of features, so we simply employ the `glm()` function from the *stats* package. The bot detection model could be built using feed predictability features computed based on LASSO, those coming from BERT, or the concatenation of both (LASSO+BERT).

The regression model is trained on the training data using gold-standard labels and, when applied to new data, return a continuous score between 0 (human) and 1 (bot). Given that the evaluation requires a single decision for each feed and the dataset is balanced, we apply a threshold of 0.5.

5 Evaluation

Our bot account detection method is evaluated in two scenarios:

- **PAN**: the basic evaluation scenario prepared by task organisers involves preparing models using the training data provided, uploading scoring software to *TIRA* infrastructure and applying it to unseen test data,
- **Internal**: additional evaluation is performed internally by dividing the training data into internal training and internal test in order to compare the performance of different approaches.

The training data provided by the organisers include 4120 feeds in the English subtask (2060 humans and 2060 bots) and 3000 feeds in the Spanish sub-task (1500 humans and 1500 bots). For the purpose of internal evaluation, the feeds are randomly divided into internal training (80% of feeds) and internal test (20% of feeds). The BERT finetuning and bot detection model inference are then performed on the internal training data and applied to the internal test. For the purpose of PAN evaluation, the LASSO features and the corresponding bot detection model are re-trained on the whole training set.

For internal evaluation, we execute three versions of our approach:

- using features based on message predictability from logistic regression (LASSO),
- using features based on message predictability from the BERT model (BERT),
- using both of the above sets of features (LASSO+BERT).

	Internal		PAN	
Feature set	English	Spanish	English	Spanish
LASSO	0.9090	0.8783	0.9155	0.8844
BERT	0.9260	0.8983	-	-
LASSO+BERT	0.9369	0.8933	-	-

Table 1. Bot detection accuracy of our approach in internal and PAN evaluation scenarios for English and Spanish tweets with respect to the feature set involved.

In each case, the output on the test is was converted to binary decision using an 0.5 threshold and compared to the gold standard labels through accuracy.

Unfortunately, we were unable to execute the BERT-based versions within the constraints of the PAN evaluation. Given the computational resources on the TIRA machines, calculating an output of a BERT model takes around 5 minutes per feed, which means scoring the whole test set would exceed the evaluation time limits imposed by the PAN organisers.

Table 1 includes the results of the evaluation. We can see that the BERT features provide better predictability estimates than LASSO, resulting in higher classification accuracy for both English and Spanish. In case of English the best results are achieved by combining the features from BERT and LASSO. In case of Spanish there is no benefit from combining the features and the overall results are substantially worse. This could be explained by smaller amount of training data available and using a multilingual BERT model instead of a language-specific one, like in case of English.

6 Conclusions

The performance of the algorithm, with the accuracy on unseen test data around 90%, seems satisfactory. Nevertheless, there remains an open question of how this results would translate to a real-life bot detection ability. The answer depends on how well the data of the PAN shared task reflect the overall population of bot and human users on Twitter. Representative datasets of this kind are typically challenging to obtain, as people and organisations publishing content through bot accounts do not intend to disclose this fact. Additionally, the distinction between ‘human’ and ‘bot’ may not be clear-cut in case of so-called ‘cyborg’ accounts, where some messages come from a bot program and some others are written by humans [2].

Our hope that the proposed approach could perform well on other datasets and related tasks is justified by the fact that it does not rely on any features specific to Twitter, but builds on the very nature of bots. Such programs will always be more predictable than humans, as long as they generate content through automatic processes. Moreover, our method casts the bot detection problem as a language modelling problem, which is a fundamental task in natural language processing. Further advances in the field could therefore be used to improve the predictability measurement and bot recognition accuracy.

Acknowledgements

This work was supported by the *Polish National Agency for Academic Exchange* through a *Polish Returns* grant number PPN/PPO/2018/1/00006.

References

1. Bessi, A., Ferrara, E.: Social Bots Distort the 2016 US Presidential Election Online Discussion. First Monday 21(11) (2016)
2. Broniatowski, D.A., Jamison, A.M., Qi, S.H., AlKulaib, L., Chen, T., Benton, A., Quinn, S.C., Dredze, M.: Weaponized health communication: Twitter bots and Russian trolls amplify the vaccine debate. American Journal of Public Health 108(10), 1378–1384 (2018)
3. Cai, C., Li, L., Zengi, D.: Behavior enhanced deep bot detection in social media. In: International Conference on Intelligence and Security Informatics: Security and Big Data (ISI 2017) (2017)
4. Daelemans, W., Kestemont, M., Manjavancas, E., Potthast, M., Rangel, F., Rosso, P., Specht, G., Stamatatos, E., Stein, B., Tschuggnall, M., Wiegmann, M., Zangerle, E.: Overview of PAN 2019: Author Profiling, Celebrity Profiling, Cross-domain Authorship Attribution and Style Change Detection. In: Crestani, F., Braschler, M., Savoy, J., Rauber, A., Müller, H., Losada, D.E., Heinatz, G., Cappellato, L., Ferro, N. (eds.) Proceedings of the Tenth International Conference of the CLEF Association (CLEF 2019). Springer (2019)
5. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. arXiv (2018), http://arxiv.org/abs/1810.04805
6. Dickerson, J.P., Kagan, V., Subrahmanian, V.: Using sentiment to detect bots on Twitter: Are humans more opinionated than bots? In: IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2014). pp. 620–627 (2014)
7. Ferrara, E.: Disinformation and Social Bot Operations in the Run Up to the 2017 French Presidential Election. First Monday 22(8) (2017)
8. Ferrara, E., Varol, O., Davis, C., Menczer, F., Flammini, A.: The rise of social bots. Communications of the ACM 59(7), 96–104 (2016)
9. Friedman, J., Hastie, T., Tibshirani, R.: Regularization Paths for Generalized Linear Models via Coordinate Descent. Journal of Statistical Software 33(1) (2010)
10. Kumar, S., Shah, N.: False Information on Web and Social Media: A Survey. arXiv (2018), http://arxiv.org/abs/1804.08559
11. Manning, C.D., Bauer, J., Finkel, J., Bethard, S.J., Surdeanu, M., McClosky, D.: The Stanford CoreNLP Natural Language Processing Toolkit. In: Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations (2014), http://aclweb.org/anthology/P14-5010
12. Potthast, M., Gollub, T., Wiegmann, M., Stein, B.: TIRA Integrated Research Architecture. In: Ferro, N., Peters, C. (eds.) Information Retrieval Evaluation in a Changing World - Lessons Learned from 20 Years of CLEF. Springer (2019)
13. R Core Team: R: A Language and Environment for Statistical Computing (2013), http://www.r-project.org/
14. Rangel, F., Rosso, P.: Overview of the 7th Author Profiling Task at PAN 2019: Bots and Gender Profiling. In: Cappellato, L., Ferro, N., Losada, D.E., Müller, H. (eds.) CLEF 2019 Labs and Workshops, Notebook Papers. CEUR Workshop Proceedings. CEUR-WS.org (2019)
15. Ratkiewicz, J., Conover, M., Meiss, M.R., Gonçalves, B., Flammini, A., Menczer, F.: Detecting and Tracking Political Abuse in Social Media. In: Proceedings of the Fifth International AAAI Conference on Weblogs and Social Media (2011)

16. Schwarz, G.: Estimating the Dimension of a Model. The Annals of Statistics 6(2), 461–464 (1978)
17. Shao, C., Ciampaglia, G.L., Varol, O., Yang, K.C., Flammini, A., Menczer, F.: The spread of low-credibility content by social bots. Nature Communications 9(1), 4787 (2018)
18. Tucker, J.A., Guess, A., Barberá, P., Vaccari, C., Siegel, A., Sanovich, S., Stukal, D., Nyhan, B.: Social Media, Political Polarization, and Political Disinformation: A Review of the Scientific Literature. Tech. rep., Hewlett Foundation (2018), https://hewlett.org/library/social-media-political-polarization-political-disinformation-review-scientific-literature/
19. Varol, O., Ferrara, E., Davis, C.A., Menczer, F., Flammini, A.: Online Human-Bot Interactions: Detection, Estimation, and Characterization. In: Proceedings of the Eleventh International AAAI Conference on Web and Social Media (ICWSM 2017) (2017)
20. Yang, K., Varol, O., Davis, C.A., Ferrara, E., Flammini, A., Menczer, F.: Arming the public with artificial intelligence to counter social bots. Human Behavior and Emerging Technologies 1(1), 48–61 (2019)