Twitter Bots and Gender Detection using Tf-idf

Notebook for PAN at CLEF 2019

Asad Mahmood and Padmini Srinivasan

The University of Iowa
{amahmood1,padmini-srinivasan}@uiowa.edu

Abstract As the amount of unstructured data increases, value (and the number) of models that can infer information from this data also increases. This paper presents another such model that can perform bots and gender detection on Twitter using just the tweets from the respective Twitter user. We show that a simple frequency based approach with a machine learning algorithm i.e., SVM can achieve high accuracy if the preprocessing is done right. In English language. our model detects bots with an accuracy of 91% and gender with an accuracy of 82%. Main strength of this model is its simplicity along-with the ease with which it can be used with other languages.

Keywords: author profiling, bots detection, gender detection, Twitter

1 Introduction

We have seen a major shape shift in internet over the past two decades and all this has happened due to the advent of digital social media. Number of social media websites have risen a lot over time. Owing to which, it is now being said that the most valued commodity has changed from oil to data[1]. This data in its crude form, like oil, isn't of much benefit due to which researchers are constantly looking for ways to structurize this data.

One of the most researched online platform is Twitter[2]. Twitter mostly deals with the unstructured textual data which can be used to extract many characteristics of its author like gender, age, identity [5] etc. PAN [3] organizes many tasks targetting the identification of these characteristics [11,12,7,10]. For example, in PAN 2018 [11] participants were asked to identify the gender of Twitter users from their tweets. In PAN 2019 [9], the organizers have added one additional step on top of the previous challenge of gender detection i.e., bots detection.

In this paper, we use the ideas of gender detection from [2] and apply them to bots detection to solve the task of bots and gender detection [9].

[1] https://www.wired.com/insights/2014/07/data-new-oil-digital-economy/

[2] https://twitter.com/

[3] https://pan.webis.de/

This paper from here on is organized as follows. Section 2 explains the existing related work. Section 3 talks about the data provided for the task. Section 4 discusses the model we used to solve this challenge. Section 5 discusses the results of our approach and then in the end we discuss our conclusion in Section 6.

2 Related Work

Bots and gender detection task [9] in this year's PAN consists of two components. The first one is bots detection and the second one is gender detection. In this section we look at the previous work done in both tasks.

2.1 Bots Detection

Over time, bots detection has been done using both machine learning algorithms with hand crafted features and with deep learning on different online platforms.

A. Hall et al. [3] used some basic features with machine learning models to detect bots on Wikipedia . Specifically they used ensemble models like random forest classifier and gradient boosting classifier on behavioural features like time difference between edits made, time spent on the website etc. This method was able to detect bots with a precision of 0.88.

Sneha et al. [4] used contextual LSTMs on Twitter data to perform both account level bots detection (with accuracy up to 100%) and tweet level bots detection (with accuracy up to 90%). Tweets used in this system are preprocessed by steps like replacing hashtags, URLs, user mentions with some static token, changing all tokens to lower case etc.

2.2 Gender Detection

Gender detection from unstructured data whether it be images, metadata or text, is one of the most researched topic. It has been studied previously in PAN [11] as well. Researchers have tried to solve this challenge using both machine and deep learning based approaches.

Daneshvar et al. [2] extract features from preprocessed tweet text and then applies SVMs to detect the gender of a given tweet. Their model achieved an accuracy of 82% on English language.

Erhan et al. [13] use character embeddings with attention based Convolutional Neural Networks (CNNs) to detect gender from tweet text without any preprocessing. This model achieved accuracy of 70%.

3 Dataset

Training data for Twitter bots and gender detection task [9] in PAN 2019 consists of tweets from different Twitter users. This dataset was made available for two languages i.e., English (en) and Spanish (es).

Table 1. Number of Twitter users in training data for each language and class.

en			es		
Bots	Humans		Bots	Humans	
	Male	Female		Male	Female
2060	1030	1030	1500	750	750

Table 1 shows the number of Twitter users in each language (en, es) and in each class (bots, male and female). For each of these users, we were provided with one xml file containing 100 unprocessed tweets.

It is evident from Table 1 that training dataset has no bias towards any class i.e., number of Twitter users for bots and humans are same and number of Twitter users for male and female are same.

4 Model

The model used to perform bots and gender detection is inspired by [2] with a difference that our approach is more focused on text pre-processing and less focused on feature engineering and model selection. Figure 1 shows the pipeline of our approach. We used this same pipeline to do bots and gender detection for each respective language.

4.1 Preprocessing

Given tweets from a particular user, we apply the following sequence of preprocessing steps.

1. **Tokenization:** In this step, we tokenize the given tweet using TweetTokenizer[4] provided by NLTK library. During tokenization we change all alphabets to lower case and restrict all character sequences of length greater than 3 to the length of 3. Restricting the character sequence size helps us remove the redundant tokens. For example, people usually write the word 'yay' as 'yaaaaayyyy' with variable repetitions in character 'a' and 'y'.
2. **URL remover:** In this step, we replace all the urls found in the tweet with the token '<URLURL>'. A token is considered as a URL if it starts with either 'https://' or 'http://'.

[4] https://www.nltk.org/api/nltk.tokenize.html

Figure 1. Pipeline to perform bot and gender classification on respective languages.

3. **User mention remover:** In this step, we replace all the user mentions found in the tweet with the token '<UsernameMention>'. A token is considered as a username mention if it starts with '@'.
4. **Hashtag remover:** In this step, we replace all the hastags found in the tweet with the token '<HashtagMention>'. A token is considered as a hashtag if it starts with '#'.

4.2 Merge Tweets

Current dataset is completely balanced with every Twitter user having 100 tweets but in real world our intuition is that the amount of tweets posted by a human and a bot will be different . So, in order to capture that, after applying all the preprocessing steps on the tweets by a user, we combine them using the token '<LineFeed>' to represent the number of tweets posted by that particular user.

Once all the tweets are combined, we put the token '<EndOfTweet>' in the end. In the current dataset and experimental setup, there is no advantage of adding '<EndOfTweet>', but this can be useful when we are considering tweet chunks based on the time intervals in which they were posted.

4.3 Calculate Features

After preprocessing and merging tweets together, we use Term Frequency-Inverse Document Frequency (Tf-idf) to encode them. Tf-idf will assign more weight to tokens appearing frequently in tweets by one user, as compared to the tweets by other users in the same class. This will help in identifying unique words used by different users belonging to same class.

We didn't try to go for a more refined feature representation like [8] as the focus of our approach is mainly preprocessing.

We used scikit-learn[5] to first calculate the counts for each token using CountVectorizer and then calculate Tf-idf using TfidfVectorizer.

4.4 Classification

After creating tweet encodings for all users, we feed them to a machine learning classifier. For this purpose we use LinearSVM[6].

For each language, we trained two binary classifiers. One was used to detect whether a given Twitter user was bot or human. The other was used to detect whether the given Twitter user was male or female.

In the submitted version of our model, if the first classifier predicts the Twitter user to be human, we pass it onto the second classifier to predict the gender.

[5] https://scikit-learn.org/stable/

[6] https://scikit-learn.org/stable/modules/generated/sklearn.svm.LinearSVC.html

5 Results

To evaluate our approach before submission, we create our own train-test split with 80% training data and 20% test data. This train-test split is stratified i.e., it has equal representation of each class in both train and test set. This is done for both languages independently.

English language has 3296 users (divided equally among bots and humans) in its train set and 824 in its test set for bots v humans experiment. For male v female experiment, train set has 1648 users and test set has 412 users.

On the other hand, Spanish language has 2400 users (divided equally among bots and humans) in its train set and 600 in its test set for bots v humans experiment. For male v female experiment, train set has 1200 users and test set has 300 users.

Table 2. Accuracy results for experiments against our test set, official test set 1 (TS-1) and official test set 2 (TS-2).

	bots v humans			male v female		
dataset	our test set	official TS-1	official TS-2	our test set	official TS-1	official TS-2
en	98%	88%	91%	91%	76%	82%
es	97%	88%	92%	84%	72%	80%

As evident from Table 2, the results after training respective models on 80% data were pretty good as compared to the random chance level i.e., 50%. This was also shown in the results when our model was tested against the official test sets provided by PAN 2019 [1] using the TIRA platform [6]

6 Conclusion

We use the frequency based features from tf-idf along with SVMs to solve the bots and gender detection task in PAN 2019. The main component of this method is the preprocessing of data. High accuracies achieved in both tasks show the importance of preprocessing even when the features used are trivial.

This approach detects bots with an accuracy of 91% in English language and achieves an accuracy of 92% for the same in Spanish language. This shows how easy and effective it is to use this approach across different languages as compared to some other models which use pre-trained language word embeddings.

SVM used to solve this challenge show a good performance but we believe that ensemble models in this case could have done better.

References

1. 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., Heinatz, G., Cappellato, L., Ferro, N. (eds.) Proceedings of the Tenth International Conference of the CLEF Association (CLEF 2019). Springer (Sep 2019)
2. Daneshvar, S., Inkpen, D.: Gender Identification in Twitter using N-grams and LSA: Notebook for PAN at CLEF 2018. In: CEUR Workshop Proceedings. vol. 2125 (2018), http://ceur-ws.org/Vol-2125/paper_213.pdf
3. Hall, A., Terveen, L., Halfaker, A.: Bot Detection in Wikidata Using Behavioral and Other Informal Cues (2018), https://dl.acm.org/citation.cfm?id=3274333
4. Kudugunta, S., Ferrara, E.: Deep Neural Networks for Bot Detection. Information Sciences 467, 312–322 (2018)
5. Patra, B.G., Das, K.G., Das, D.: Multimodal Author Profiling for Twitter. In: Notebook for PAN at CLEF 2018. CEUR-WS.org (2018)
6. 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)
7. Rangel, F., Celli, F., Rosso, P., Potthast, M., Stein, B., Daelemans, W.: Overview of the 3rd Author Profiling Task at PAN 2015. In: CLEF 2015 Evaluation Labs and Workshop Working Notes Papers (2015)
8. Rangel, F., Franco-Salvador, M., Rosso, P.: A low dimensionality representation forÂălanguage variety identification. In: Gelbukh, A. (ed.) Computational Linguistics and Intelligent Text Processing. pp. 156–169. Springer International Publishing, Cham (2018)
9. 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., Müller, H. (eds.) CLEF 2019 Labs and Workshops, Notebook Papers. CEUR-WS.org (Sep 2019)
10. Rangel, F., Rosso, P., Chugur, I., Potthast, M., Trenkmann, M., Stein, B., Verhoeven, B., Daelemans, W.: Overview of the 2nd Author Profiling Task at PAN 2014. In: CLEF 2014 Evaluation Labs and Workshop Working Notes Papers, Sheffield, UK, 2014 (2014)
11. Rangel, F., Rosso, P., y Gómez, M.M., Potthast, M., Stein, B.: Overview of the 6th Author Profiling Task at PAN 2018: Multimodal Gender Identification in Twitter. CEUR-WS.org (2018)
12. Rangel, F., Rosso, P., Verhoeven, B., Daelemans, W., Potthast, M., Stein, B.: Overview of the 4th Author Profiling Task at PAN 2016: Cross-Genre Evaluations . In: Working Notes Papers of the CLEF 2016 Evaluation Labs. CEUR Workshop Proceedings/Balog, Krisztian [edit.]; et al. (2016)
13. Sezerer, E., Polatbilek, O., Sevgili, O., Tekir, S.: Gender Prediction From Tweets With Convolutional Neural Networks. In: Proceedings of the Ninth International Conference of the CLEF Association (CLEF 2018) (2018)