Naive-Bayesian Classification for Bot Detection in Twitter

Notebook for PAN at CLEF 2019

Pablo Gamallo[1] and Sattam Almatarneh[1,2]

[1] Centro de Investigación en Tecnoloxías Intelixentes (CiTIUS)
University of Santiago de Compostela, Galiza
[2] Computer Science Department, University of Vigo
Escola Superior de Enxeñaría Informática, Campus As Lagoas, Ourense 32004, Spain.
{pablo.gamallo,sattam.almatarneh}@usc.es

Abstract This article describes a system that participated in the Bots and Gender Profiling shared task at PAN 2019. The first objective of the task is to detect whether the author of a Twitter account is a bot or a human; and in case of human, the second objective is to identify the gender of the user account. For this purpose, we present a bayesian strategy based on features including specific content of tweets and automatically built lexicons. The best configuration of features reached 0.88 accuracy in the official Spanish test dataset and 0.81 in the English one for the bot/human classification. For gender profiling, the scores we obtained were lower, around 0.70.

1 Introduction

Social bots are programs to automate usual human activities such as messages generation. The main objectives of bots are to interact with humans, resend messages or pictures from other users, add likes to other messages, and so on. The rise of bots in online social networks have led to the emergence of malicious behavior including misinformation dissemination and any pollutant content such as malware spreaders or spammers. To identify social bots, machine learning techniques have been used successfully exceeding in several cases 95% accuracy, but with datasets built by the authors themselves [17,11]. In 2015, DARPA Social Media in Strategic Communications program conducted the *Twitter Bot Detection Challenge* [15], whose aim was to identify influence bots supporting a pro-vaccination discussion on Twitter. In this case, the final results of the competition were much more discrete as the best systems did not reach 50% accuracy.

Machine learning techniques for social bots detection in Twitter generally make use of a great variety of features. Among them, we have identified the following types: user profile, user friendship networks (following and followers), content of tweets and user history. One of the tasks in Bots and Gender Profiling Shared Task [14] in PAN 2019

[7], tries to give a new pulse to the studies of bot profiling. After having dealt with some facets of author profiling in social networks since 2013 (e.g., gender, age, gender and even language variety), the main aim in 2019 is to detect whether the author of a Twitter account is a bot or a human. In addition, in case of human, the second objective is to identify the gender of the user account. It is worth noting that the training datasets provided by the Shared Task organizers do not contain all the information required to extract all feature types enumerated above. More precisely, there is no information on user profile, user friendship networks, or user history.

Given the characteristic of the training dataset provided by the PAN Shared Task 2019, we design a machine learning strategy based on features only including content of tweets as well as automatically built lexicons. Therefore, features built from user profile and user history are beyond our scope. As there is small training data, we decided to use a basic Naive Bayes classifier, which performs well in this type of task as it was reported in [1].

The PAN Shared Task not only includes bot detection, but also gender identification. However, as our main objective is to identify bots from human accounts, we decided to reuse the features conceived for the bot/human task also for the female/male identification task, with slight differences concerning lexical features.

As it will be reported later, for the bot detection task in PAN 2019, our best feature configuration reached 0.88 and 0.81 accuracy in the official Spanish and English test dataset, respectively. These results are similar to those achieved during the development phase, even though they are below the state-of-the-art described in the related work section.

In the next section (2), we will describe other works and experiments just focused on bot detection (not gender profiling). Then, Section 3 describes the Twitter-based features used by our classification method. Experiments are reported in Section 4 and conclusions are addressed in Section 5.

2 Related Work

The possibility to collect tweets from user accounts and, thereby, build training datasets allowed researchers to design machine learning methods for social bot detection. The main idea behind these methods is to discover the key features of social bots to draw the border between a human actor and a machine. In the following, we introduce some selected works.

[17] try to characterize and understand Sybil account activity, that is fake accounts, in the Renren Online Social Network. Renren is a Chinese OSN similar to Facebook. They claimed that Sybils in social networks do not form close-knit communities, contrary to what was said in the structure-based approaches for bot detection [16]. The authors applied a SVM classifier on Renren 2,000 accounts (1,000 human accounts and 1,000 Sybils) achieving very high performance: about 99% accuracy.

[6] focused on the classification of human, bot, and cyborg accounts on Twitter, where cyborgs stand for either human assisted bots or bot assisted humans. Using a collection of 500,000 accounts, the authors studied the differences among bots, humans, and cyborgs, by considering features related to account properties, tweet content, and

tweeting behavior. They applied a Random Forest-Based classifier on a test dataset of 2,000 users, reaching 98% accuracy for humans, 91% for cyborgs, and 96% for bots.

[11] introduced the first strategy to filter out content polluters using social honeypots. They collected 23,869 polluters (bots or not) by making use of a small set of honeypots they created. Then, they developed a large variety of classification algorithms, such as naive bayes, logistic regression, support vector machine, or tree-based for distinguishing between content polluters and legitimate users. Random Forest produced the highest performance, reaching 98.42% accuracy.

In [1], the authors collected and manually labeled a dataset of Twitter accounts, including bots, human users, and hybrids (i.e., tweets posted by both human and bots). This dataset was used to train and test several types of classifiers. Random Forest and bayesian algorithms reached the best performance at both two-class (bot / human) and three-class classification (bot / human / hybrid).

[5] used a deep learning method to extract latent temporal patterns. To the best of our knowledge, the first system that applies deep neural network in bot detection. However, this method cannot be compared to the PAN Shared Task approach because the datasets provided by the organizers in the shared task do not provide explicit time axis information in the user accounts.

"Bot or Not?" [8] is one of the first social bot detectors publicly available for Twitter. The detection algorithm relies on more than 1000 features which are grouped into six types: user, network, temporal, content, friends, and sentiment. BotOrNot can be used via a website and APIs.[3]

Another system aimed at detecting bots on Twitter is SentiBot [9], which focuses on a number of sentiment-related factors that are key to the identification of bots. Therefore, Sentibot employs sophisticated sentiment analysis techniques to extract relevant features to train the classifier.

Unlike previously introduced systems, the method we propose has to adapt to the characteristics of the dataset provided by the shared task and, therefore, focus on features related to the linguistic content of tweets.

3 Types of Features

Feature extraction and selection is a critical process for any classification task. In the following, we describe the different types of features we used in the experiments reported later.

3.1 Social Network Features

These are specific characteristics of the language of social networks, which include textual elements that can only be found on Twitter. We used the following list of social network features:

- Ratio of the number of hashtags (i.e. number of hashtags used by a user account divided by total number of tweets sent from this account)

[3] http://botornot.co/

- Ratio of the number of user references.
- Ratio of the number of url links.
- Ratio of the number of retweets.
- Ratio of the number of textual emoticons, where textual emoticons are, for instance: ';)', ':)', and so on.
- Ratio of the number of emojis.
- Ratio of the number of onomatopoeia, where onomatopoeia are for instance: *haha* in English or *jeje* in Spanish.
- Ratio of the number of language abbreviations, where abbreviations are for instance: *b4* (*before*) or *btw* (*by the way*) in English, and *q* (*que*) or *xq* (*porque*), in Spanish.
- Ratio of the number of alliterations, that is, the repetition of vowel sounds.

3.2 Content-Based Features

These are features that can be extracted from any text message. The content features we used are the following:

- Ratio of the size of tweets.
- Ratio of the number of identical pairs of tweets.
- Lexical richness, defined as lemma/token ratio (LTR):

$$LTR = \frac{\parallel L \parallel}{\parallel T \parallel} \tag{1}$$

  where $\parallel L \parallel$ is the number of different lemmas appearing in the tweets of one user account, and $\parallel T \parallel$ is the total number of tokens. As grammatical words should not be taken into account, we only consider lexical lemmas and tokens, that is, $l \in L$ and $t \in T$ if l and t are nouns, adjectives, verbs or adverbs.
- Similarity between sequential pairs of tweets, t_1 and t_2, defined as follows:

$$Sim(t_1, t_2) = \frac{\parallel L_{t_1} \cap L_{t_2} \parallel}{\parallel L_{t_1} \cup L_{t_2} \parallel} \tag{2}$$

  where L_{t_1} and L_{t_2} are the lexical lemmas (nouns, adjectives, verbs, and adverbs) of tweets t_1 and t_2, respectively, where $t_1 \prec t_2$. To obtain the final similarity ratio associated with a user account, all Sim scores between pairs of sequential tweets are added, and the result is divided by the total number of tweets.

3.3 Lexical Features

Lexical features were derived from several domain-specific lexicons, in particular, four different weighted lexicons were automatically built for each language:

human-machine lexicon: a lexicon consisting of specific words belonging to two classes: the language of bots and the language of humans in Twitter.

female-male lexicon: a lexicon consisting of specific words belonging to two classes: women language and men language.

sentiment lexicon of human-machine: a lexicon consisting of polarity words (positive or negative) used by bots or humans.

sentiment lexicon of female-male: a lexicon consisting of polarity words (positive or negative) used by women or men.

Each lexicon was built by making use of the annotated corpora provided by the PAN organizers and a ranking algorithm. For instance, as the word *consent* appears frequently in the female discourse in Twitter, it will be added as a female word within the female-male lexicon. In addition, a weight is assigned to each word within a lexicon. The higher the weight the more intense the female or male value of the word. The same procedure was followed for building the human-machine lexicon. Concerning sentiment lexicons, we also used the same method but restricted with external polarity lexicons, that is, only words also appearing in external sentiment resources are considered. As in [9], we consider that a number of sentiment-related factors might be essential to the identification of bots.

We just considered words belonging to lexical categories, hence, only nouns, verbs, adjectives, and adverbs were selected. Besides lexical words, hashtags were also taken into account. PoS tagging for English and Spanish was carried out with the multilingual toolkit LinguaKit [10]. The polarity lexicon provided by Linguakit was also used as external resource to build sentiment lexicons of human-machine and female-male classes.

The method to build a domain-specif lexicon is somehow inspired by that reported in [3,2] for very negative opinions. The score of a word given a class (bot, human, female or male), noted C, is computed as follows:

$$C(w) = \frac{freq_{Total}(w)}{freq_C(w)} \tag{3}$$

where $freq_{Total}(w)$ is the number of occurrences of word w in the whole annotated corpus, and $freq_C(w)$ stands for the number of occurrences of the same word in the segments (tweets) annotated as belonging to this class, where C stands for bot, human, female or male. In addition to the class score C, it is also required to compute a threshold above which the word is considered as belonging to the class. So, we compute the difference between the use of a word within the given class and out of it:

$$DIFF(w) = freq_C(w) - freq_{-C}(w) \tag{4}$$

where $freq_{-C}(w)$ stands for the occurrences of w in segments that are not annotated as C. To insert a word in the lexicon, the value of $DIFF(w)$ must be higher than a threshold. In our experiments, this value for human-machine and female-male was 50. So, in these two lexicons, we only selected those words with $DIFF$ values higher than 50. In the case for sentiment lexicons the threshold was set to 10. Finally, words were ranked by their C score giving rise to weighted and ranked lexicons. The same procedure were carried out to build specific sentiment lexicons. Yet, for this purpose, we made use of general-purpose polarity lexicons to just extract polarity words.

features	**bot/human accuracy**	**male/female accuracy**
bow	0.73	0.77
lex	0.62	0.68
text	0.62	0.51
bow+text	0.73	**0.77**
lex+text	**0.83**	0.67
bow+lex+text	0.62	0.73

Table 1. Results obtained by using the English training and development datasets

features	**bot/human accuracy**	**male/female accuracy**
bow	0.85	0.67
lex	0.65	0.57
text	0.71	0.50
bow+text	0.83	**0.68**
lex+text	**0.90**	0.67
bow+lex+text	0.80	0.63

Table 2. Results obtained by using the Spanish training and development datasets

4 Experiments

In order to find the best feature configuration in a classification task, we have used a Bayesian algorithm. In addition to its simplicity and efficiency, Naive Bayes performs well in this type of task, as described in [1], where the Bayesian classifier obtained the best results in the bot/human classification. Our classifier was implemented with the NaiveBayes Perl module.[4] As it was mentioned before, in order to lemmatize and identify lexical PoS tags, tweets were processed using the multilingual toolkit LinguaKit [10].

In Tables 1 and 2, content and social network features are called textual features (*text*), while lexical features (*lex*) represent both human-machine and sentiment lexicons. It is important to point out that *text* features are the same for both bot/human and male/female classification. By contrast, *lex* features were specified for each subtask. In addition, we also consider traditional bag-of-words with term frequency (simplified as *bow*). Tables 1 and 2 show the results obtained by different combinations of those features configuring the bayesian classifier for English and Spanish, respectively.

Naive Bayes classifier with *bow* works acceptably but the combination of *bow* with other features drops the accuracy, as in the experiments for hate speech detection reported in [4]. By contrast, combining *lex* or *bow* with *text* peforms well. This could be explained by the fact that *lex-text* and *bow-text* are pairs of features that are conceptually independent while Naive-Bayes algorithm assumes that all features are conceptually independent. In Spanish, *lex-text* achieves 0.90 accuracy and 0.83 in English for bot/human detection. Concerning the gender profiling task, the best configuration in

[4] https://metacpan.org/pod/Algorithm::NaiveBayes

Language	bot/human accuracy	male/female accuracy
Spanish	0.88	0.71
English	0.81	0.72

Table 3. Results obtained by our best configurations with the Spanish and English official test set of PAN Shared Task: Bots and Gender Profiling.

both languages is *bow-text*, achieving 0.77 and 0.68 accuracy, respectively in Spanish and English. By contrast, *lex-bow* pair does not work well in any task since *lex* and *bow* seem to be quite dependent features. In fact, *lex* is a relevant subset of *bow*.

The most important observation to point out is that there are some feature combinations that improve the *bow* baseline. It is worth noting that in many classification tasks, the *bow* model is very difficult to overcome. So, the results we obtained seem to prove that the features described are useful and, therefore, it is worthwhile to go deeper into their improvement and use.

We selected the best configurations to be used with the official test dataset of PAN Shared Task. For bot detection in Spanish and English, we used *lex-text* pair of features, while for gender profiling in the two languages we used *bow-text*. The official results depicted in Table 3 are very similar to those obtained in the development phase, although they are a little lower for bot detection and a little higher for gender profiling. The experiments with the official test dataset were carried out in a Ubuntu 16-04 virtual machine by means of TIRA [12], which is a web service to facilitate software submissions to shared tasks. The software will be freely available.

It is worth noting than the Spanish accuracy for bot/human classification outperforms two of the baselines proposed by the organizers, namely that relying on word embeddings and that based on the low dimensionality model reported in [13]. By contrast, the n-grams baselines worked slightly better than our approach, which was the 16th best system out of 44 for this specific task.

5 Conclusions

In this study, we presented a basic classification method to bot detection focused on the extraction and selection of relevant features. The experiments showed that both linguistic features extracted from tweets and lexical information from external resources may help the classification process by improving baseline feature configurations. The experiments also showed that the selected features have a better behavior in the task of identifying bots than in gender profiling.

In current work, we are collecting political accounts from Twitter in order to analyze the influence of malicious bots in the different elections that are taken place in Spain in 2019. One of our aims is to build an annotated corpus with the aid of the best features we have identified in the present work. In future work, we will use those features as heuristics of an unsupervised system aimed at ranking Twitter accounts from *more* human to *less* human. This ranked list of accounts will be revised by annotators so that a reliable gold-standard dataset is obtained at the end.

Acknowledgments

This work has been partially supported by the DOMINO project (PGC2018-102041-B-I00, MCIU/AEI/FEDER, UE) and eRisk project (RTI2018-093336-B-C21). It also has received financial support from the Consellería de Cultura, Educación e Ordenación Universitaria (accreditation 2016-2019, ED431G/08) and the European Regional Development Fund (ERDF).

References

1. Alarifi, A., Alsaleh, M., Al-Salman, A.: Twitter turing test. Inf. Sci. 372(C), 332–346 (Dec 2016), https://doi.org/10.1016/j.ins.2016.08.036
2. Almatarneh, S., Gamallo, P.: Automatic construction of domain-specific sentiment lexicons for polarity classification. In: International Conference on Practical Applications of Agents and Multi-Agent Systems. pp. 175–182. Springer (2017)
3. Almatarneh, S., Gamallo, P.: A lexicon based method to search for extreme opinions. PloS one 13(5), e0197816 (2018)
4. Almatarneh, S., Gamallo, P., Pena, F.J.R.: CiTIUS-COLE at semeval - 2019 task 5: Combining linguistic features to identify hate speech against immigrants and women on multilingual tweets. In: the 13th international Workshop on Semantic Evaluation (2019)
5. Cai, C., Li, L., Zengi, D.: Behavior enhanced deep bot detection in social media. In: 2017 IEEE International Conference on Intelligence and Security Informatics (ISI). pp. 128–130 (July 2017)
6. Chu, Z., Gianvecchio, S., Wang, H., Jajodia, S.: Detecting automation of twitter accounts: Are you a human, bot, or cyborg? IEEE Trans. Dependable Secur. Comput. 9(6), 811–824 (Nov 2012), http://dx.doi.org/10.1109/TDSC.2012.75
7. 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)
8. Davis, C.A., Varol, O., Ferrara, E., Flammini, A., Menczer, F.: Botornot: A system to evaluate social bots. In: Proceedings of the 25th International Conference Companion on World Wide Web. pp. 273–274. WWW '16 Companion, International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, Switzerland (2016), https://doi.org/10.1145/2872518.2889302
9. Dickerson, J.P., Kagan, V., Subrahmanian, V.S.: Using sentiment to detect bots on twitter: Are humans more opinionated than bots? In: Proceedings of the 2014 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining. pp. 620–627. ASONAM '14, IEEE Press, Piscataway, NJ, USA (2014), http://dl.acm.org/citation.cfm?id=3191835.3191957
10. Gamallo, P., Garcia, M., Piñeiro, C., Martinez-Castaño, R., Pichel, J.C.: Linguakit: A big data-based multilingual tool for linguistic analysis and information extraction. In: 2018 Fifth International Conference on Social Networks Analysis, Management and Security (SNAMS). pp. 239–244 (2018)
11. Lee, K., Eoff, B.D., Caverlee, J.: Seven months with the devils: a long-term study of content polluters on twitter. In: In AAAI Int'l Conference on Weblogs and Social Media (ICWSM (2011)

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. Rangel, F., Franco-Salvador, M., Rosso, P.: A low dimensionality representation for language variety identification. In: Proceedings of the 17th International Conference on Computational Linguistics and Intelligent Text Processing (CICLing 2016). Springer-Verlag (2016)
14. Rangel, F., Rosso, P.: Overview of the 7th Author Profiling Task at PAN 2019: Bots and Gender Profiling. In: In: Cappellato L., Ferro N., Müller H, Losada D. (Eds.), CLEF 2019 Labs and Workshops, Notebook Papers. CEUR Workshop Proceedings. CEUR-WS.org (2019)
15. Subrahmanian, V.S., Azaria, A., Durst, S., Kagan, V., Galstyan, A., Lerman, K., Zhu, L., Ferrara, E., Flammini, A., Menczer, F.: The darpa twitter bot challenge. Computer 49(6), 38–46 (Jun 2016), https://doi.org/10.1109/MC.2016.183
16. Viswanath, B., Post, A., Gummadi, K.P., Mislove, A.: An analysis of social network-based sybil defenses. SIGCOMM Comput. Commun. Rev. 41(4), – (Aug 2010), http://dl.acm.org/citation.cfm?id=2043164.1851226
17. Yang, Z., Wilson, C., Wang, X., Gao, T., Zhao, B.Y., Dai, Y.: Uncovering social network sybils in the wild. ACM Trans. Knowl. Discov. Data 8(1), 2:1–2:29 (Feb 2014), http://doi.acm.org/10.1145/2556609