Chapter 1

Feature Engineering for Social Bot Detection

Onur Varol

Indiana University, Bloomington, IN, USA

Clayton A. Davis

Indiana University, Bloomington, IN, USA

Filippo Menczer

Indiana University, Bloomington, IN, USA

Alessandro Flammini

Indiana University, Bloomington, IN, USA

1.1 Introduction

Social media serve as a medium to disseminate information and a platform to connect millions of individuals. Properties of social media make it the ideal tool for communication, however, entities with malicious intentions have strong motives to abuse online social networks to profit or gain power by boosting their popularity, manipulating online discussions, and targeting certain groups to attack [29, 27, 74].

Increasing evidence suggests that social platforms like Twitter accommodate an increasing number of autonomous entities known as social bots [29, 4]. A recent study estimates between 9% to 15% of the accounts on Twitter display bot-like behaviors [78]. These autonomous entities are controlled by software that generates content and establishes interactions with other accounts. It is fair to point out that not all bots have malicious intentions; many are used for benign tasks, such as dissemination of news and publications [56, 41] and coordination of volunteer activities [73]. But there is a growing record of vicious applications of social bots.

Examples of malicious social-bot use include emulating human behavior to manufacture fake grassroots political support [72, 34], promoting terrorist propaganda and recruitment [5, 31, 7, 43], manipulating stock and advertisement markets [19, 27], and disseminating rumors and conspiracy theories [6].

The magnitude of the problem is underscored by a social bot detection challenge recently organized by DARPA to study information dissemination mediated by automated accounts and to detect deceptive activities carried out by these bots [76]. Researchers also point to the possibility of social bot involvement in online discourse about the US presidential election in 2016 [7, 43].

1.2 Social bot detection

Discussion of social bot activity, the broader implications for social network platforms, and the detection of these accounts are becoming central research avenues [50, 12, 83, 8, 20, 29]. Previous research categorized various types and *modus operandi* of social bots [63, 66, 46].

Mainstream research efforts have focused on three approaches to detect social bots: holistic, pairwise, and egocentric analysis. Each approach presents its own advantages and disadvantages to analyze the activities of users.

1.2.1 Holistic approach

In terms of performance and accuracy, the holistic approach performs better than other methodologies, since it captures more information about accounts and their interactions. However, capturing a complete picture of social networking systems is not practical outside of the companies that own the platforms themselves.

Having complete information about social network structure, user interactions, and online activities allows social media companies to build operational systems. Examples of studies discussing holistic solutions focus on clustering behavioral patterns of users [80] and classifying accounts using supervised learning techniques [83, 50]. For instance, Beutel *et al.* extract behavioral similarities by decomposing event data in time, user, and activity dimensions [8].

Advantages of this approach over other methodologies come from the availability of complete data. Other methodologies lack full knowledge of social ties that are hard to collect due to their dynamic nature. Companies can also track user behaviors such as impressions on each posts, time spent on user profiles, and usage statistics of the website to extract useful features. Such behavioral features have been studied to measure the credibility of online information [33] and purchasing behaviors [40].

A limitation of this approach is the computational complexity of analyzing such massive data in real or near-real time with limited resources. Recent advances in deep learning and reinforcement learning may help mitigate these limitations.

1.2.2 Pairwise account comparison

Evidence of so-called *botnets* — coordinated collectives of software-controlled fake accounts — has been observed in support of the Syrian War [1] as well as in seemingly aimless activities [26]. The comparisons of temporal or content patterns among pairs of accounts can reveal significant similarities that are unlikely to emerge organically.

The idea is to enumerate all elements of certain account features, such as friends, followers, URLs, hashtags, and so on. Pairwise comparison between sets defined for each user can then be used to compute account similarities. Such methodology has also been applied to cluster memes on social media [28, 44].

The pairwise comparison methodology has been used to detect abnormally correlated user activities [16]. An advantage of this approach is that the computed similarity matrix can be employed in both supervised and unsupervised learning frameworks [77, 62]. However, the computation of pairwise similarities in huge networks is very costly without some heuristics to narrow down the possible pairs.

1.2.3 Egocentric analysis

Egocentric analysis captures and evaluates information about a single user at a given point in time. When performed by users, as opposed to the platform owners themselves, one is usually restricted to collecting the public subset of information about other accounts. However the trade-off between computational complexity and accuracy favors this simpler approach in many cases.

In the literature, we have observed several examples of system designed to operate with limited information resources by considering single accounts [17, 18, 20, 50]. Most of the research in this direction relies on annotations by experts and crowd-sourcing workers to train supervised learning algorithms and evaluate the consistency and effectiveness of different detection systems.

1.3 Online bot detection framework

In this section, we present an online bot detection system, Botometer (`botometer.iuni.iu.edu`), that is freely available for academic and public use as part of the Observatory on Social Media (OSoMe) project [23]. Our system extracts 1150 features from a collection of tweets related to a given Twitter account and uses them in a machine-learning framework to classify the account as being operated by a bot or a human [78, 24]. Accessible via a website and an API, our system served over 30 million requests in the first several months after public release in 2016, as shown in Fig. 1.1.

Our desire to build a bot detection system for public use informed the choices of criteria for building feature sets and training classifiers. As a publicly-available service, we require the system to be simultaneously fast and reliable. Single-request speed is important in order for the website to feel responsive, while reliability is critical for API users submitting requests in bulk.

With single-request speed in mind, we took computational efficiency into account as well as accuracy when selecting a feature set. Details of the features implemented are discussed in Sec. 1.3.1.

Additionally we limit analysis to only the most recent activity from a given account. This is a result of strict rate limits on the Twitter API; each evaluation by Botometer only requires a single call to each Twitter API endpoint used, thus maximizing the number of account analyses possible per unit time and minimizing the total time required to classify a single account.

Besides speed and reliability, public availability necessitates that our system be useful without any special data or permission from the platform owners. As such, we only use public data from the Twitter API according to their terms of service.

Considering factors of computational efficiency, performance, and infor-

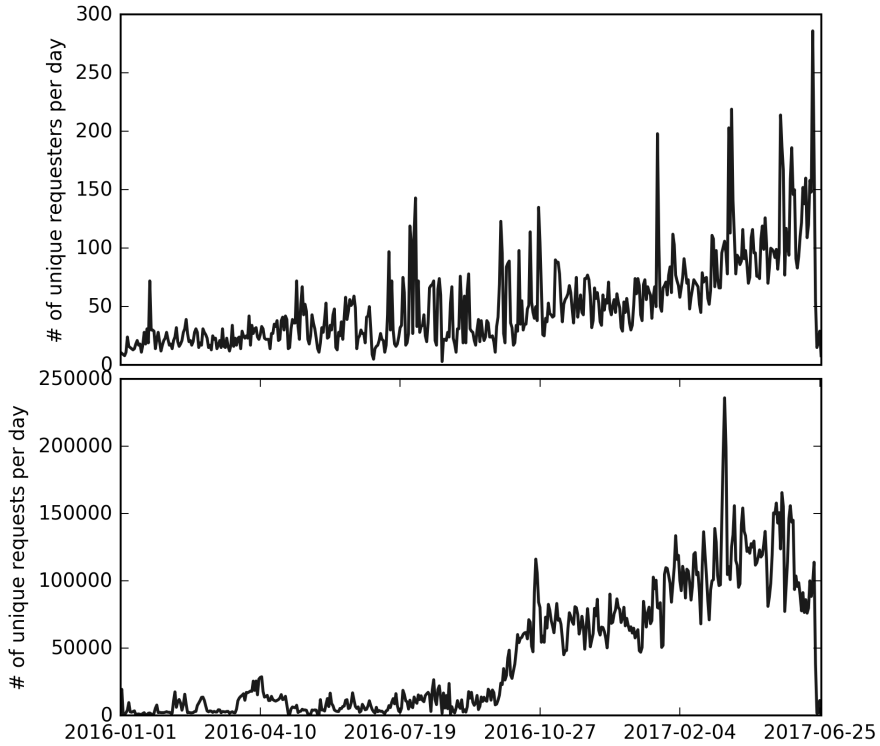

FIGURE 1.1: Number of daily requests (bottom) and unique requesters (top) served by the Botometer system.

mation access, the egocentric approach to classification best fits our design goals.

1.3.1 Feature extraction

Our approach to egocentric classification of a given Twitter user leverages three main types of data obtained from the Twitter REST API[1]: the user profile, tweets produced and broadcasted (retweeted) by the user, and tweets authored by other accounts mentioning the target user.

Quotas on the number of requests per unit time to each Twitter API endpoint present a trade off between accuracy and classification volume: more API calls per user yield more data, which may improve accuracy, but proportionally decrease the number of accounts one can analyze before reaching the quota. Our system harnesses the most recent 200 tweets by the user and 100 tweets mentioning the user. These quantities correspond to the maximum number of tweets that can be collected with a single API request to each endpoint (each endpoint is individually rate-limited). Therefore this choice is a natural trade-off between data volume (performance) and accuracy.

The public data and meta-data about the target user, collected using the Twitter API, is distilled into 1,150 features. These features are roughly categorized as friends, tweet content and sentiment, network patterns, and activity time series. Next we present details about the individual features in each class.

1.3.1.1 User-based features

As with other systems analyzing Twitter users and behavior, we leverage user meta-data features extracted from meta-data [60, 29]. First we count the length and number of digits in the user's `screen_name` and `user_name` (these can differ on Twitter). Users can also provide a textual `description` of themselves; we consider the length of this field as well as the number of unique descriptions observed in tweets from users connected via retweet, mention, *etc.*

User activity and connectivity in its simplest form can also provide signals for classification. We extract numerical features about number of `friends` and `followers`, as well as of different activity types such as `tweet`, `retweet`, `mention`, and `reply`. We consider both the total number of tweets in each type as well as their temporal rate. Further discussion of these social relations and tweet types can be found in the next section.

When a new account is created on Twitter, default values are used for some profile fields, such as profile image, until changed by the user. We use binary features indicating whether or not a given account has each of these `default properties`. We also extract features about `account age` and `time-zone`. For a complete list of features in this category see Table 1.1.

[1] `dev.twitter.com/rest/public`

TABLE 1.1: List of features extracted from user profile.

List of `user-features`
Screen name length
Number of digits in screen name
User name length
Time offset (sec.)
Default profile (binary)
Default picture (binary)
Account age (days)
Number of unique profile descriptions among connected users
(*) Profile description lengths for connected users
(*) Number of friends distribution
(*) Number of followers distribution
(*) Number of favorites distribution
Number of friends (signal-noise ratio and rel. change)
Number of followers (signal-noise ratio and rel. change)
Number of favorites (signal-noise ratio and rel. change)
Number of tweets (per hour and total)
Number of retweets (per hour and total)
Number of mentions (per hour and total)
Number of replies (per hour and total)
Number of retweeted (per hour and total)

(*) Distribution types. Following statistics are computed and used as individual features: min, max, median, mean, std. deviation, skewness, kurtosis, and entropy.

TABLE 1.2: List of features extracted from neighbors of user. We consider four types of users: retweeting, mentioning, retweeted, and mentioned.

List of `friend-features`
Number of distinct languages
Entropy of language use
(*) Account age distribution
(*) Time offset distribution
(*) Number of friends distribution
(*) Number of followers distribution
(*) Number of tweets distribution
(*) Description length distribution
Fraction of users with default profile
Fraction of users with default picture

(*) Distribution types. Following statistics are computed and used as individual features: min, max, median, mean, std. deviation, skewness, kurtosis, and entropy.

1.3.1.2 Friend features

Twitter actively encourages social connectivity via `following` other accounts. Users can follow any public profile, and the follow relation need not be reciprocal. The reverse of the `follower` relation is the `friend` relation: if A follows B, then B is a friend of A.

In this setting content can disseminated by users rebroadcasting other users' tweets via `retweets`. A tweet can be addressed to one or more specific users by `mention`ing the target users' screen names. We consider four types of links: retweet, mention, being retweeted, and being mentioned. Grouping tweets by their link type, we extract *friend-features* for each group separately.

Groupwise distributions of user meta-data are then extracted for accounts in the group. We compute distributions for number of friends, followers, and tweets, length of profile description, account age, and time-zone offset. For each distribution we compute mean, maximum, minumum, and median values, along with skewness, kurtosis, and entropy.

We also consider the number of unique languages represented in the group as well as entropy of language use as features. The fraction of those users with default profile information is also included. All of the features in this category are listed in Table 1.2.

1.3.1.3 Network features

Network structure can contain information useful for characterizing different types of communication. Network features have notably been leveraged in the context of astroturf detection [72]. Our system constructs three types of networks: `retweet`, `mention`, and `hashtag co-occurrence`.

Retweet and mention networks are represented as weighted, directed networks with users as nodes and retweets/mention tweets as links. The link

TABLE 1.3: List of features extracted from interaction and hashtag co-occurrence networks. We consider three types of network: retweet, mention, and hashtag co-occurrence networks.

List of `network-features`
Number of nodes
Number of edges (also for reciprocal)
(*) Strength distribution
(*) In-strength distribution
(*) Out-strength distribution
Network density (also for reciprocal)
(*) Clustering coeff. (also for reciprocal)

(*) Distribution types. Following statistics are computed and used as individual features: min, max, median, mean, std. deviation, skewness, kurtosis, and entropy.

direction corresponds to the information flow: toward the user retweeting or being mentioned. The edge weight represents the frequency of interaction.

A hashtag is a word prefixed with the hash (#) symbol, and is used in Twitter as a topic identifier. Hashtag co-occurrence networks are weighted, undirected networks with hashtags as the nodes. Two hashtags are linked when they occur together in a given tweet, and the edge is weighted according to the frequency of the co-occurrence in tweets.

Given the local nature of egocentric data collection, we utilize simple network features that quantify local interactions. These measures also happen to be the least expensive to compute. The most straight-forward features we consider are number of nodes and edges, as well as the density of the network. We also include features extracted from distributions of local clustering coefficients and (in-/out-)strength, or weighted degree. Subgraphs of the retweet and mention networks that contain only reciprocal links are additionally considered and used for feature extraction. The complete list of features in this category can be found in Table 1.3.

1.3.1.4 Content and language features

Content and linguistic analysis of tweets have been used for a wide variety of applications [21, 58, 64, 13, 51, 22]. The simplest content features we use come from word counts and text entropy.

Other content features are extracted by applying the *Part-of-Speech* (POS) tagging technique, which identifies different types of natural language components. We consider 9 types of POS tags: verbs, nouns, adjectives, modal auxiliaries, pre-determiners, interjections, adverbs, wh-, and pronouns. Distributions of POS tag occurrences are used to extract features to reflect use of different language styles [15]. For a complete list of features in this category see Table 1.4.

TABLE 1.4: List of features extracted from content of tweets.

List of `content-features`
(*,**) Frequency of POS tags in a tweet
(*,**) Proportion of POS tags in a tweet
(*) Number of words in a tweet
(*) Entropy of words in a tweet

(*) Distribution types. Following statistics are computed and used as individual features: min, max, median, mean, std. deviation, skewness, kurtosis, and entropy.
(**) Part-of-Speech (POS) tag. There are nine POS tags: verbs, nuns, adjectives, modal auxiliaries, pre-determiners, interjections, adverbs, wh-, and pronouns.

1.3.1.5 Sentiment features

Sentiment analysis is used to describe the emotions conveyed by a piece of text, or more broadly, the attitude or mood of an entire conversation. Sentiment extracted from social media conversations has been used to forecast offline events including financial market fluctuations [11] and is known to affect information spreading and social structure [61, 32, 10].

Our framework leverages several sentiment extraction techniques to generate various sentiment features:

- `ANEW`: Arousal, valence and, dominance scores are selected for analysis of mood and sentiment based on theoretical foundations of these dimensions [81]. Crowd-sourcing is used to annotate over 14k words along each of the three dimensions.

- `Happiness`: To quantify happiness in a text, we use a dataset of over 10k words identified and annotated by researchers [48]. This word list contains the most frequent words collected from Google books, New York Times articles, music lyrics, and Twitter messages.

- `Polarization and strength`: This measure identifies a phrase as neutral or polar and then disambiguates the polarity of the polar expressions. [82].

- `Emoticon`: Pictorial representations of different facial expressions are popular on social media. We used a lexicon of such symbols and character sequences to identify positively and negatively associated text [2]. This does not include emoji, although recent work exploring the popularity of emojis in social media [59, 3] suggests that their inclusion would be possible.

All these techniques rely on a lexicon to compute scores for each content and there exist several alternatives one could consider [36]. One could extend this analysis by adopting machine learning models trained solely to extract features about sentiment. The complete list of sentiment features is found in Table 1.5.

TABLE 1.5: List of features extracted from sentiment analysis of content.

List of `sentiment-features`
Mean of happiness scores of aggregated tweets
Standard deviation of happiness scores of aggregated tweets
(***) Happiness scores of aggregated tweets
Mean of valence scores of aggregated tweets
Standard deviation of valence scores of aggregated tweets
(***) Valence scores of aggregated tweets
Mean of arousal scores of aggregated tweets
Standard deviation of arousal scores of aggregated tweets
(***) Arousal scores of aggregated tweets
Mean of dominance scores of aggregated tweets
Standard deviation of dominance scores of aggregated tweets
(***) Dominance scores of single tweets
(*) Happiness score of single tweets
(*) Valence score of single tweets
(*) Arousal score of single tweets
(*) Dominance score of single tweets
(*) Polarization score of single tweets
(*) Entropy of polarization scores of single tweets
(*) Positive emoticons entropy of single tweets
(*) Negative emoticons entropy of single tweets
(*) Emoticons entropy of single tweets
(*) Positive and negative score ratio of single tweets
(*) Number of positive emoticons in single tweets
(*) Number of negative emoticons in single tweets
(*) Total number of emoticons in single tweets
Ratio of tweets that contain emoticons

(*) Distribution types. Following statistics are computed and used as individual features: min, max, median, mean, std. deviation, skewness, kurtosis, and entropy.

(***) For each feature, we compute mean and std. deviation of the weighted average across words in the lexicon.

We note that both content and sentiment features are language-specific and were trained on corpora of English-language tweets. Language-agnostic evaluation is possible by using models trained without these two categories.

1.3.1.6 Temporal features

Temporal signatures are shown to be useful in the context of analyzing content production and consumption, identification of online campaigns, and evolution of online discussion [35, 30, 16, 79].

Basic temporal features indicate how frequently an account is active; a human is unlikely to tweet hundreds of times per day. These features are listed in the user class. More sophisticated temporal features are extracted using distributions of time intervals between consecutive tweets, retweets, and mentions. Table 1.6 lists the features in this category.

TABLE 1.6: List of features extracted from temporal information.

List of `temporal-features`
(*) Time between two consecutive tweets
(*) Time between two consecutive retweets
(*) Time between two consecutive mentions

(*) Distribution types. Following statistics are computed and used as individual features: min, max, median, mean, std. deviation, skewness, kurtosis, and entropy.

1.3.2 Possible directions for feature engineering

As Twitter introduces new functionality and usage patterns evolve over time, one may consider creating new features to leverage these additional behaviors. For example, Twitter has recently introduced *quoted tweets*, which are essentially a retweet with additional user-supplied commentary.

Modern machine learning techniques can also be applied to extract more sophisticated features from the existing data. Deep learning and vector embeddings are promising technologies that one can employ to extract features for network structure [39, 69], language and sentiment [37, 25, 57]. Research in this area may lead to features that capture not only basic statistics but also semantics expressed by textual content [42, 49]. Of course, these more sophisticated analyses come with computational costs that must be weighed against one's desire for fast classification results. We discuss implications of recent technologies using deep learning more in detail in Sec. 1.4.

1.3.3 Feature analysis

With such a wide range of signals from various domains of available data and meta-data, we want to quantify the interactions among features. Upon examination of the pairwise correlations between features, we do notice that some of these features are correlated and thus possibly redundant in the context of social bot detection.

The magnitude of pairwise correlations is shown in Fig. 1.2. Features in this representation are grouped by classes and sorted by average correlation within each group.

The degree of correlation varies depending on the feature category. On average we observe 0.21 correlation among friends features, which is largely due to dependencies between profile meta-data. Content and network features also exhibit some redundancy.

These correlated features demonstrate the importance of feature selection: they suggest that we may be able to retain accuracy while extracting only a subset of our features. The next section introduces different feature selection methods and examines how they perform in identifying a subset of representative features.

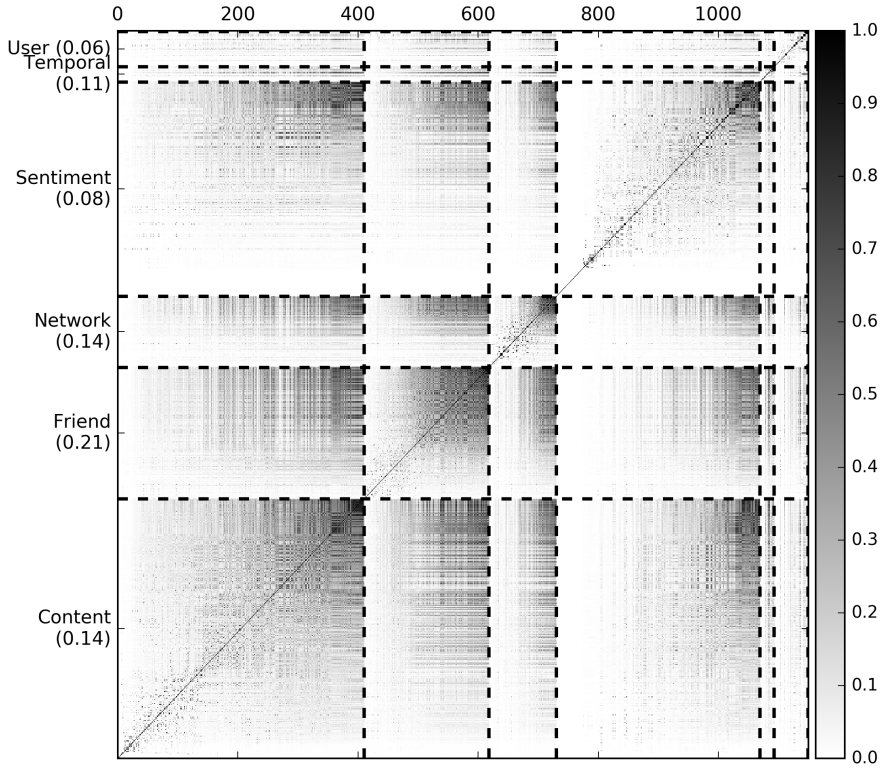

FIGURE 1.2: Intensities of pairwise correlation between feature values across the dataset. Average pairwise correlations are also reported for the features within each class.

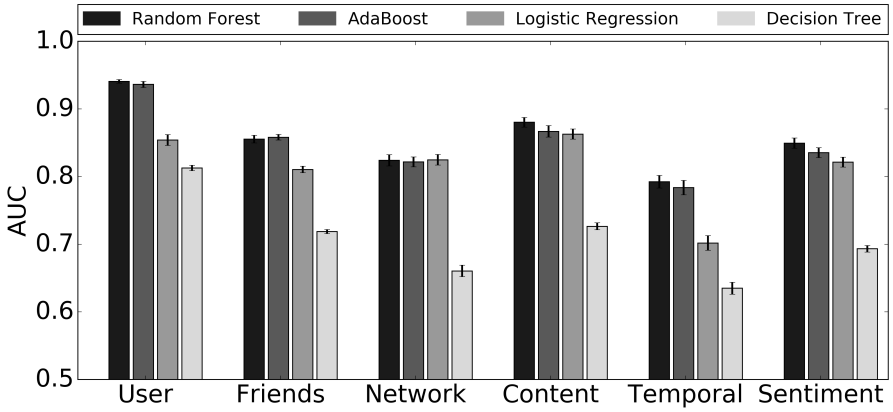

FIGURE 1.3: Performance comparison for different feature classes and classification methods.

1.3.4 Feature selection

We built a pipeline to evaluate classification models using subsets of our features. This pipeline uses several off-the-shelf benchmark algorithms provided in the *scikit-learn* library [67]. Our models are trained and evaluated using two different datasets: a honeypot dataset collected by Lee *et.al* [50] and a manually annotated collection [78]. We combine these two datasets to capture both simpler and more sophisticated bot behaviors, along with examples of humans accounts, from different time intervals. A model's accuracy is evaluated by measuring the Area Under the receiver operating characteristic Curve (AUC) with 5-fold cross validation, and computing the average AUC score across the folds using Random Forests, AdaBoost, Logistic Regression, and Decision Tree classifiers.

1.3.4.1 Feature classes

To compare the discriminatory power of the different feature types, we trained models using each class of features alone. We repeated the performance evaluation experiments considering independently only the user, friends, network, content, temporal, and sentiment feature classes. Fig. 1.3 presents the performance of the classifiers using the different feature subsets in isolation.

We achieved the best performance with user meta-data features. Content features are also effective. Both yielded AUC above 0.9. Other feature classes yielded AUC above 0.8. In addition to giving the best overall accuracy when all features are considered, Random Forest models produce scores at least as good as every other method when restricting to single feature classes.

TABLE 1.7: Top features according to Random Forests algorithm.

User number of friends
User number of favorites
Mentioned friends' mean tweet count
User number of follower
User account age
Mentioned friends' mean account age
Mention network mean edge strength
Mentioned friends' mean profile description length
Tweet content mean adjective count
Mentioned friends' mean number of followers

1.3.4.2 Top individual features

Given the performance of Random Forest models as compared to the other models, as well as its interpretability and robustness to overfitting, we use Random Forests in the rest of this analysis. This is also the algorithm used in the production Botometer service.

The Random Forest method builds a number of decision trees. In each tree, nodes represents a single condition about a feature value, designed to split the dataset into two so that similar response values end up in the same set. The split criterion can be either Gini impurity or information gain/entropy.

To enrich our understanding about important features, we extend our analysis beyond studying classes of features. To compute the importance of a single feature in Random Forests, one can average across trees the contribution of that feature in reducing impurity. We list the top features identified using this method in Table. 1.7.

Below we briefly describe a few additional feature selection methods inspired by information theory. Further details can be found in a recent review by Li *et.al* [54].

- **CIFE**: Conditional Informative Feature Extraction introduces class-relevant redundancy to maximize the joint class-relevant information by explicitly reducing the class-relevant redundancies among features [55].
- **FCBF**: Fast Correlation Based Feature solution tries to identify pairs of features correlating with each other [84]. Once a group of correlated features is identified, this method selects the subset of these features that have smaller inter-dependencies.
- **MRMR**: This method aims to achieve feature selection by controlling quality of features that satisfy Maximum dependency and Relevance, and Minimum Redundancy [68].

Fig. 1.4 shows that the best accuracy can be obtained using as few as 20 features selected by CIFE or Random Forests (RF). In terms of accuracy,

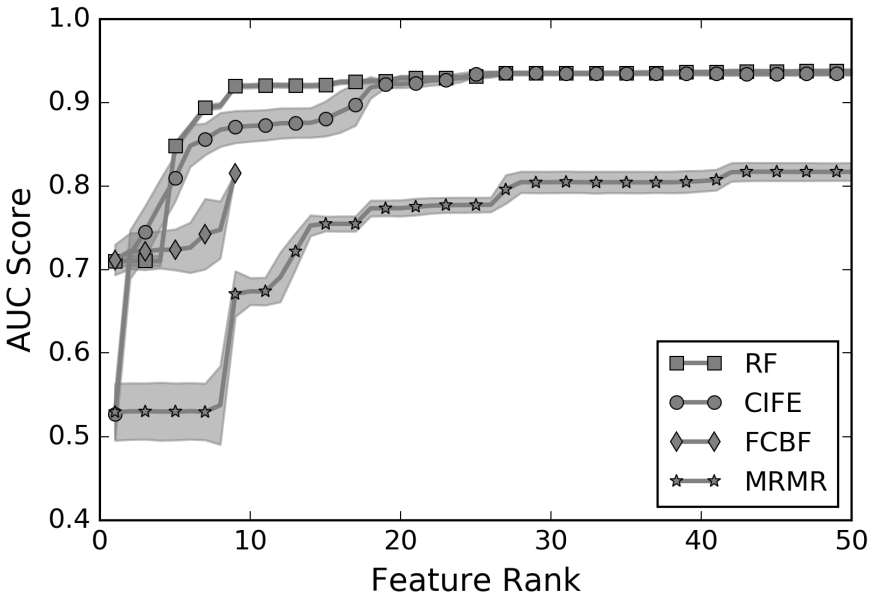

FIGURE 1.4: Classifier accuracy given by AUC of models trained using top N features, as ranked by different feature selection methods.

RF and CIFE perform equally well. The MRMR and FCBF methods do not perform as well. Comparing the top features chosen by each algorithm, CIFE algorithms tend to pick network, content, and friend features; MRMR and FCBF prefer content and sentiment features. Since FCBF is a subset selection method, only a small set of feature is identified by the algorithm. The Random Forest model ranks friend and user features more highly, as listed in Table 1.7.

1.4 Conclusions

While bots can be harmless or even useful parts of the social media ecosystem, bot accounts that are not clearly identified as such can be used for nefarious purposes. Since social bots can broadcast at a high rate and in coordination with other bots, they can skew the online conversation by amplifying the "volume" of the bot controller's message and creating the appearance that the message is coming from many independent sources. This can in term influence public opinion by overwhelming our capacity to discriminate quality information [70] and by leveraging cognitive biases that lead people to pay

attention to what is popular [65] and to trust content that seems to be shared by social connections [45] or in a social group [47]. Given the amount of public discourse taking place on social media platforms, it becomes crucial for users and platforms to be able to distinguish such activities as early and as accurately as possible.

Research on social bot detection aims to provide tools for identifying autonomous entities. While the arms race between humans and deceptive bots is likely to continue for years, advances in feature engineering and in the identification of weaknesses of different classes of social bots will be key to preserving our stance against malicious bot activities.

In this chapter, we presented the most common approaches used in systems for identifying social bots. We focused on egocentric analysis methodology due to its advantages with respect to data collection and algorithmic complexity. Our system, Botometer, analyzes public information about a Twitter account, extracting over a thousand features describing the account and its neighbors [78]. Using these features, we created a classifier that scores an account's likelihood of being a bot. We examined the extracted features in terms of their contribution to overall performance and redundancy within the feature set.

Feature selection is as essential as feature engineering for improving the performance of bot detection systems, especially when taking into consideration trade-offs between accuracy and computational speed. Some machine learning methods such as Random Forests can measure the importance of features intrinsically by using ensembles of weak learners [14]. We analyzed the top features identified by the Random Forest algorithm and also evaluated other feature selection mechanisms in the recent literature [54]. Our analysis points out that Random Forests can achieve over 90% accuracy, as measured by AUC, using fewer than 20 features.

Let us discuss some future directions that one can pursue to design better and semi-autonomous systems for social bot detection. Deep learning presents natural extensions to some of our feature extraction methods [49]. Architectures of deep neural networks (DNNs) can capture important patterns and use those as features for learning algorithms. As such, DNNs may be useful in identifying increasingly sophisticated bots.

Research into the use of these modern techniques for bot detection becomes even more critical when considering how they may be used by bot *creators*. Recent advances in DNN technologies accelerate fake persona generation [52, 9] and conversation models for social bots [75, 53]. Generative adversarial nets can be used to simultaneously learn generative models for social bots and how to trick detection systems [38, 71].

The task of social bot detection exhibits the characteristics of an arms race. Both bot creators and the bot detection community work towards improving their existing systems and try to exploit weaknesses of the adversary group. It is our hope that the work presented here will provide a key advantage in the

arms race against deceptive bot creators. By working together and sharing public tools and data[2] we won't have to fight this battle alone.

[2] A public repository of social bot datasets and tools is available on the Botometer website and we invite the community to contribute.

Bibliography

[1] Norah Abokhodair, Daisy Yoo, and David W McDonald. Dissecting a social botnet: Growth, content and influence in twitter. In *Proc. of the ACM Intl. Conf. on Computer Supported Cooperative Work & Social Computing*, pages 839–851. ACM, 2015.

[2] Apoorv Agarwal, Boyi Xie, Ilia Vovsha, Owen Rambow, and Rebecca Passonneau. Sentiment analysis of Twitter data. In *Proc. of the Workshop on Languages in Social Media*, pages 30–38. ACL, 2011.

[3] Wei Ai, Xuan Lu, Xuanzhe Liu, Ning Wang, Gang Huang, and Qiaozhu Mei. Untangling emoji popularity through semantic embeddings. In *Proc. of the Intl. Conf. on Web and Social Media*, pages 2–11, 2017.

[4] Luca Aiello, Martina Deplano, Rossano Schifanella, and Giancarlo Ruffo. People are strange when you're a stranger: Impact and influence of bots on social networks. In *Proc. of the Intl. Conf. on Web and Social Media*, 2012.

[5] J Berger and Jonathan Morgan. 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.

[6] Alessandro Bessi, Mauro Coletto, George Alexandru Davidescu, Antonio Scala, Guido Caldarelli, and Walter Quattrociocchi. Science vs conspiracy: Collective narratives in the age of misinformation. *PLoS ONE*, 10(2):e0118093, 02 2015.

[7] Alessandro Bessi and Emilio Ferrara. Social bots distort the 2016 us presidential election online discussion. *First Monday*, 21(11), 2016.

[8] Alex Beutel, Wanhong Xu, Venkatesan Guruswami, Christopher Palow, and Christos Faloutsos. Copycatch: stopping group attacks by spotting lockstep behavior in social networks. In *Proc. ACM Intl. Conf. on World Wide Web*, pages 119–130, 2013.

[9] Parminder Bhatia, Marsal Gavalda, and Arash Einolghozati. soc2seq: Social embedding meets conversation model. Preprint 1702.05512, arXiv, 2017.

[10] Johan Bollen, Bruno Gonçalves, Ingrid van de Leemput, and Guangchen Ruan. The happiness paradox: your friends are happier than you. *EPJ Data Science*, 6(1):4, 2017.

[11] Johan Bollen, Huina Mao, and Xiaojun Zeng. Twitter mood predicts the stock market. *Journal of Computational Science*, 2(1):1–8, 2011.

[12] Yazan Boshmaf, Ildar Muslukhov, Konstantin Beznosov, and Matei Ripeanu. The socialbot network: when bots socialize for fame and money. In *Proc. of the Annual Conf. on Computer Security Applications*, 2011.

[13] Federico Botta, Helen Susannah Moat, and Tobias Preis. Quantifying crowd size with mobile phone and twitter data. *Royal Society open science*, 2(5):150162, 2015.

[14] Leo Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001.

[15] R Sherlock Campbell and James W Pennebaker. The secret life of pronouns flexibility in writing style and physical health. *Psychological science*, 14(1):60–65, 2003.

[16] Nikan Chavoshi, Hossein Hamooni, and Abdullah Mueen. Identifying correlated bots in twitter. In *Proc. Intl. Conf. on Social Informatics*, pages 14–21, 2016.

[17] Zi Chu, Steven Gianvecchio, Haining Wang, and Sushil Jajodia. Who is tweeting on twitter: human, bot, or cyborg? In *Proc. of the Annual Conf. on Computer Security Applications*, pages 21–30, 2010.

[18] Zi Chu, Steven Gianvecchio, Haining Wang, and Sushil Jajodia. Detecting automation of twitter accounts: Are you a human, bot, or cyborg? *IEEE Tran Dependable & Secure Comput*, 9(6):811–824, 2012.

[19] Eric Clark, Chris Jones, Jake Williams, Allison Kurti, Michell Nortotsky, Christopher Danforth, and Peter Dodds. Vaporous marketing: Uncovering pervasive electronic cigarette advertisements on twitter. Preprint 1508.01843, arXiv, 2015.

[20] Eric Clark, Jake Williams, Chris Jones, Richard Galbraith, Christopher Danforth, and Peter Dodds. Sifting robotic from organic text: a natural language approach for detecting automation on twitter. *Journal of Computational Science*, 16:1–7, 2016.

[21] Cristian Danescu-Niculescu-Mizil, Robert West, Dan Jurafsky, Jure Leskovec, and Christopher Potts. No country for old members: user lifecycle and linguistic change in online communities. In *Proc. of the Intl. Conf. on World Wide Web*, pages 307–318, 2013.

[22] Abimanyu Das, Sreenivas Gollapudi, Emre Kiciman, and Onur Varol. Information dissemination in heterogeneous-intent networks. In *Proc. ACM Intl. Conf. on Web Science*, 2016.

[23] Clayton A Davis, Giovanni Luca Ciampaglia, Luca Maria Aiello, Keychul Chung, Michael D Conover, Emilio Ferrara, Alessandro Flammini, Geoffrey C Fox, Xiaoming Gao, Bruno Gonçalves, et al. Osome: the iuni observatory on social media. *PeerJ Computer Science*, 2:e87, 2016.

[24] Clayton Allen Davis, Onur Varol, Emilio Ferrara, Alessandro Flammini, and Filippo Menczer. BotOrNot: A system to evaluate social bots. In *Proc. of the Intl. Conf. Companion on World Wide Web*, pages 273–274, 2016.

[25] Cícero Nogueira Dos Santos and Maira Gatti. Deep convolutional neural networks for sentiment analysis of short texts. In *Proc. Intl. Conf. on Computational Linguistics (COLING)*, pages 69–78, 2014.

[26] Juan Echeverría and Shi Zhou. The 'star wars' botnet with >350k twitter bots. Preprint 1701.02405, arXiv, 2017.

[27] Emilio Ferrara. Manipulation and abuse on social media. *SIGWEB Newsletter*, Spring(4):1–9, 2015.

[28] Emilio Ferrara, Mohsen JafariAsbagh, Onur Varol, Vahed Qazvinian, Filippo Menczer, and Alessandro Flammini. Clustering memes in social media. In *Proc. IEEE/ACM Intl. Conf. on Advances in Social Networks Analysis and Mining*, pages 548–555. IEEE, 2013.

[29] Emilio Ferrara, Onur Varol, Clayton Davis, Filippo Menczer, and Alessandro Flammini. The rise of social bots. *Communications of the ACM*, 59(7):96–104, 2016.

[30] Emilio Ferrara, Onur Varol, Filippo Menczer, and Alessandro Flammini. Detection of promoted social media campaigns. In *Proc. of the Intl. Conf. on Web and Social Media*, 2016.

[31] Emilio Ferrara, Wen-Qiang Wang, Onur Varol, Alessandro Flammini, and Aram Galstyan. Predicting online extremism, content adopters, and interaction reciprocity. In *Proc. of the Intl. Conf. on Social Informatics*, pages 22–39, 2016.

[32] Emilio Ferrara and Zeyao Yang. Quantifying the effect of sentiment on information diffusion in social media. *PeerJ Computer Science*, 2015.

[33] Andrew J Flanagin and Miriam J Metzger. The role of site features, user attributes, and information verification behaviors on the perceived credibility of web-based information. *New Media & Society*, 9(2):319–342, 2007.

[34] Michelle C Forelle, Philip N Howard, Andrés Monroy-Hernández, and Saiph Savage. Political bots and the manipulation of public opinion in Venezuela. Technical Report 2635800, SSRN, 2015.

[35] Rumi Ghosh, Tawan Surachawala, and Kristina Lerman. Entropy-based classification of retweeting activity on twitter. In *Proc. of KDD workshop on Social Network Analysis*, August 2011.

[36] CJ Hutto Eric Gilbert. Vader: A parsimonious rule-based model for sentiment analysis of social media text. In *Proc. of the Intl. Conf. on Weblogs and Social Media*, 2014.

[37] Xavier Glorot, Antoine Bordes, and Yoshua Bengio. Domain adaptation for large-scale sentiment classification: A deep learning approach. In *Proc. of the Intl. Conf. on Machine Learning*, pages 513–520, 2011.

[38] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

[39] Aditya Grover and Jure Leskovec. node2vec: Scalable feature learning for networks. In *Proc. of the ACM SIGKDD Intl. Conf. on Knowledge Discovery and Data Mining*, pages 855–864. ACM, 2016.

[40] Angela V Hausman and Jeffrey Sam Siekpe. The effect of web interface features on consumer online purchase intentions. *Journal of Business Research*, 62(1):5–13, 2009.

[41] Stefanie Haustein, Timothy D Bowman, Kim Holmberg, Andrew Tsou, Cassidy R Sugimoto, and Vincent Larivière. Tweets as impact indicators: Examining the implications of automated bot accounts on twitter. *Journal of the Association for Information Science and Technology*, 67(1):232–238, 2016.

[42] Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. Teaching machines to read and comprehend. In *Advances in Neural Information Processing Systems*, pages 1693–1701, 2015.

[43] Philip N Howard and Bence Kollanyi. Bots,#StrongerIn, and #Brexit: computational propaganda during the UK-EU Referendum. Technical Report 2798311, SSRN, 2016.

[44] Mohsen JafariAsbagh, Emilio Ferrara, Onur Varol, Filippo Menczer, and Alessandro Flammini. Clustering memes in social media streams. *Social Network Analysis and Mining*, 4(1):1–13, 2014.

[45] T. Jagatic, N. Johnson, M. Jakobsson, and F. Menczer. Social phishing. *Communications of the ACM*, 50(10):94–100, October 2007.

[46] Yuede Ji, Yukun He, Xinyang Jiang, Jian Cao, and Qiang Li. Combating the evasion mechanisms of social bots. *Computers & Security*, 58:230–249, 2016.

[47] Youjung Jun, Rachel Meng, and Gita Venkataramani Johar. Perceived social presence reduces fact-checking. *Proceedings of the National Academy of Sciences*, 114(23):5976–5981, 2017.

[48] Isabel M Kloumann, Christopher M Danforth, Kameron Decker Harris, Catherine A Bliss, and Peter Sheridan Dodds. Positivity of the english language. *PLoS ONE*, 7(1):e29484, 2012.

[49] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015.

[50] Kyumin Lee, Brian David Eoff, and James Caverlee. Seven months with the devils: A long-term study of content polluters on twitter. In *Proc. of the AAAI Intl. Conf. on Web and Social Media*, 2011.

[51] Adrian Letchford, Helen Susannah Moat, and Tobias Preis. The advantage of short paper titles. *Royal Society Open Science*, 2(8):150266, 2015.

[52] Jiwei Li, Michel Galley, Chris Brockett, Georgios P Spithourakis, Jianfeng Gao, and Bill Dolan. A persona-based neural conversation model. Preprint 1603.06155, arXiv, 2016.

[53] Jiwei Li, Will Monroe, Alan Ritter, Michel Galley, Jianfeng Gao, and Dan Jurafsky. Deep reinforcement learning for dialogue generation. Preprint 1606.01541, arXiv, 2016.

[54] Jundong Li, Kewei Cheng, Suhang Wang, Fred Morstatter, Trevino Robert, Jiliang Tang, and Huan Liu. Feature selection: A data perspective. Preprint 1601.07996, arXiv, 2016.

[55] Dahua Lin and Xiaoou Tang. Conditional infomax learning: An integrated framework for feature extraction and fusion. In Aleš Leonardis, Horst Bischof, and Axel Pinz, editors, *Proc. 9th European Conference on Computer Vision (ECCV), Part I*, pages 68–82. Springer, 2006.

[56] Tetyana Lokot and Nicholas Diakopoulos. News bots: Automating news and information dissemination on twitter. *Digital Journalism*, 4(6):682–699, 2016.

[57] Andrew L Maas, Raymond E Daly, Peter T Pham, Dan Huang, Andrew Y Ng, and Christopher Potts. Learning word vectors for sentiment analysis. In *Proc. of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 142–150, 2011.

[58] Julian McAuley and Jure Leskovec. From amateurs to connoisseurs: modeling the evolution of user expertise through online reviews. In *Proc. 22nd Intl. ACM Conf. World Wide Web*, pages 897–908, 2013.

[59] Hannah Miller, Jacob Thebault-Spieker, Shuo Chang, Isaac Johnson, Loren Terveen, and Brent Hecht. Blissfully happy or ready to fight: Varying interpretations of emoji. *Proc. of the Intl. Conf. on Web and Social Media*, 2016, 2016.

[60] Alan Mislove, Sune Lehmann, Yong-Yeol Ahn, Jukka-Pekka Onnela, and J Niels Rosenquist. Understanding the demographics of Twitter users. In *Proc. of the Intl. AAAI Conf. on Weblogs and Social Media*, 2011.

[61] Lewis Mitchell, Kameron Decker Harris, Morgan R Frank, Peter Sheridan Dodds, and Christopher M Danforth. The geography of happiness: Connecting Twitter sentiment and expression, demographics, and objective characteristics of place. *PLoS ONE*, 8(5):e64417, 2013.

[62] Pabitra Mitra, CA Murthy, and Sankar K. Pal. Unsupervised feature selection using feature similarity. *IEEE transactions on pattern analysis and machine intelligence*, 24(3):301–312, 2002.

[63] Silvia Mitter, Claudia Wagner, and Markus Strohmaier. A categorization scheme for socialbot attacks in online social networks. In *Proc. of the ACM Intl. Conf. on Web Science*, 2013.

[64] Delia Mocanu, Andrea Baronchelli, Nicola Perra, Bruno Gonçalves, Qian Zhang, and Alessandro Vespignani. The Twitter of Babel: Mapping world languages through microblogging platforms. *PLoS ONE*, 8(4):e61981, January 2013.

[65] Azadeh Nematzadeh, Giovanni L. Ciampaglia, Filippo Menczer, and Alessandro Flammini. How algorithmic popularity bias hinders or promotes quality. Preprint 1707.00574, arXiv, 2017.

[66] Richard J Oentaryo, Arinto Murdopo, Philips K Prasetyo, and Ee-Peng Lim. On profiling bots in social media. In *Proc. Intl. Conf. on Social Informatics*, pages 92–109. Springer, 2016.

[67] 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. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[68] Hanchuan Peng, Fuhui Long, and Chris Ding. Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Transactions on pattern analysis and machine intelligence*, 27(8):1226–1238, 2005.

[69] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. Deepwalk: Online learning of social representations. In *Proc. of the Intl. Conf. on Knowledge discovery and data mining*, pages 701–710. ACM, 2014.

[70] Xiaoyan Qiu, Diego F. M. Oliveira, Alireza Sahami Shirazi, Alessandro Flammini, and Filippo Menczer. Limited individual attention and online virality of low-quality information. *Nature Human Behavior*, 1:0132, 2017.

[71] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. Preprint 1511.06434, arXiv, 2015.

[72] J. Ratkiewicz, M. Conover, M. Meiss, B. Goncalves, A. Flammini, and F. Menczer. Detecting and tracking political abuse in social media. In *Proc. Intl. Conf. on Weblogs and Social Media ICWSM)*, pages 297–304, 2011.

[73] Saiph Savage, Andres Monroy-Hernandez, and Tobias Höllerer. Botivist: Calling volunteers to action using online bots. In *Proc. of the Intl. Conf. on Computer-Supported Cooperative Work & Social Computing*, pages 813–822. ACM, 2016.

[74] Chengcheng Shao, Giovanni Luca Ciampaglia, Onur Varol, Alessandro Flammini, and Filippo Menczer. The spread of fake news by social bots. Preprint 1707.07592, arXiv, 2017.

[75] Alessandro Sordoni, Michel Galley, Michael Auli, Chris Brockett, Yangfeng Ji, Margaret Mitchell, Jian-Yun Nie, Jianfeng Gao, and Bill Dolan. A neural network approach to context-sensitive generation of conversational responses. Preprint 1506.06714, arXiv, 2015.

[76] VS Subrahmanian, Amos Azaria, Skylar Durst, Vadim Kagan, Aram Galstyan, Kristina Lerman, Linhong Zhu, Emilio Ferrara, Alessandro Flammini, Filippo Menczer, et al. The DARPA Twitter Bot Challenge. *IEEE Computer*, 6(49):38–46, 2016.

[77] Wei Tang, Zhengdong Lu, and Inderjit S Dhillon. Clustering with multiple graphs. In *Proc. IEEE Intl. Conf. on Data Mining*, pages 1016–1021. IEEE, 2009.

[78] Onur Varol, Emilio Ferrara, Clayton A Davis, Filippo Menczer, and Alessandro Flammini. Online human-bot interactions: Detection, estimation, and characterization. In *Proc. of the Intl. Conf. on Web and Social Media*, 2017.

[79] Onur Varol, Emilio Ferrara, Filippo Menczer, and Alessandro Flammini. Early detection of promoted campaigns on social media. *EPJ Data Science*, 6(1):13, 12 2017.

[80] Gang Wang, Tristan Konolige, Christo Wilson, Xiao Wang, Haitao Zheng, and Ben Y Zhao. You are how you click: Clickstream analysis for sybil detection. In *Proc. USENIX Security*, pages 1–15. Citeseer, 2013.

[81] Amy Beth Warriner, Victor Kuperman, and Marc Brysbaert. Norms of valence, arousal, and dominance for 13,915 english lemmas. *Behavior research methods*, pages 1–17, 2013.

[82] Theresa Wilson, Janyce Wiebe, and Paul Hoffmann. Recognizing contextual polarity in phrase-level sentiment analysis. In *Proc. of the Intl. Conf. on Human Language Techn & Empirical Methods in NLP*, pages 347–354. ACL, 2005.

[83] Zhi Yang, Christo Wilson, Xiao Wang, Tingting Gao, Ben Y Zhao, and Yafei Dai. Uncovering social network sybils in the wild. *ACM Trans. Knowledge Discovery from Data*, 8(1):2, 2014.

[84] Lei Yu and Huan Liu. Feature selection for high-dimensional data: A fast correlation-based filter solution. In *ICML*, volume 3, pages 856–863, 2003.

1.5 Glossary

Social bot: A social bot, also known as a sybil account, is a computer algorithm that automatically produces content and interacts with humans on social media.

Botnet: Coordinated collectives of software-controlled fake accounts operating on social media.

ROC: Receiver Operating Characteristic curve serves as a tool to visually evaluate the performance of a binary classifier as the value of threshold is varied.

AUC: A measure of quality for a classification system by computing the area under the ROC curve.