Boosted Decision Tree (BDT) model for political analysis: Using machine learning to assess the *Anonymous* campaign against Islamic Extremists on Twitter

Elena Labzina[*]and George Yin[†]

Abstract

Social scientists now have unprecedented access to a wealth of information on human behavior, but "big data" pose unique analytical challenges. Classic regression models, which make rigid assumptions regarding the data generating process, are often unsuited for extracting information from big data. Machine learning (ML) models, in contrast, are ideal for this task because they are flexible. In this article, we review major families of ML models, with a focus on Boosted Decision Tree (BDT) models, which is the gold standard of ML models. Furthermore, we explain how to compute the "marginal effect" of any variable and uncertainty estimates for ML models. We illustrate our techniques with an analysis of an original dataset of 16,286 suspicious jihadist Twitter accounts reported by cyber activists. This article provides a guide on ML methods for political scientists, and contributes to our understanding of the politics of social media.

[*]Corresponding author. ABD, Department of Political Science, and Master's Candidate in Statistics, Department of Mathematics, Washington University at St. Louis. elena.labzina@wustl.edu.

[†]Dickey Fellow in U.S.Foreign Policy and International Security, Dartmouth College. george.yin1@gmail.com.

Word count: 8371

1 Introduction

In the era of social media, social scientists have unprecedented access to a wealth of information on human behavior, which poses unique analytical challenges (Boyd and Crawford 2012). When analyzing "big data", researchers are often ignorant about the underlying data generating process, e.g. complex interactions among covariates, nonlinearities, and discontinuities. Traditional regression models, which make strong assumptions regarding the data generating process, are therefore often unsuited for analyzing big data. This deficiency encourages political scientists to utilize machine learning (ML) technique for the analysis of large and complex datasets. ML, as a subfield of artificial intelligence and computer science, develops techniques to extract information from data by building a predictive model from sample inputs – the "training data-set" – without making strong assumptions about the data generating process (e.g. linearity in parameters). Indeed, as Arthur Samuel (one of the founding fathers of machine learning who first coined this term in 1959) puts it, ML ”gives computers the ability to learn without being explicitly programmed”[1].

Despite increasing interest in ML models among social scientists, the potential of ML for political science remains to be fully realized (De Marchi 2005, chapter 3; Lazer et al. 2009; Hazlett and Hainmueller 2014; Grimmer 2015, Alvarez 2016). No study in political science has yet to introduce and compare the performances of *different classes* of popular ML models, which has prevented political scientists from fully realizing the potential

[1] “ What Is Machine Learning?”. *IBM developerWorks*. [link]

of ML models.[2] Consequently, political science researchers often choose ML models for their classification/ prediction tasks haphazardly, if not overlooking major classes of ML models that might be more suitable for their analyses. Furthermore, no study in political science has yet explained how to correctly isolate the effect that a specific variable has on the quality of the prediction of an outcome of interest with an ML model, or how to estimate the uncertainty associated with any point estimate from ML models. Without answers to those questions, political scientists will be unable to fully benefit from ML analysis.

In this article, we address these issues in the following ways. First, we provide an overview of the most popular families of ML models, with a particular focus on Boosted Decision Trees (BDT), which is the "gold standard" of ML models today. We also explain how to select the optimal ML model for any classification task, and explicate two new (to political scientists) techniques to compute the "marginal effect" of a variable for a ML model and to compute the uncertainty estimates for ML results. One ambition of this article is to provide a straightforward and up-to-date guide on how to apply ML models for social scientists who are interested in analyzing complex and large datasets with computational methods.

We illustrate the prowess of BDT and the techniques we outlined with an analysis of an original dataset on 16,286 unique suspicious jihadist Twitter accounts based on more

[2] An important forthcoming paper (Montgomery and Olivella Forthcoming) analyzes the performances of different decision tree based ML models for applications in political science, but the paper does not discuss other families of ML models (e.g. neural net models).

than 450,000 reports from cyber activists under the leadership of the hacker collective *Anonymous*. Islamist terrorists – particular the Islamic State (IS) – are adroit at running online campaigns to gather support for their causes (Berger 2015, Klausen 2015), which has attracted the attention from scholars of terrorism and cyberpolitics. Chatfield, Reddick and Brajawidagda (2015) detail how Islamist terrorists utilize "information disseminators" in recruiting and distribution of jihadist propaganda. Winter (2015) analyzes IS propaganda on social media, which often focuses on highlighting the group's claim and its military prowess. Nielsen (2017) explicates why Muslim clerics radicalize with internet data on *fatwas* (rulings on Islamic law). No study so far – to the authors' knowledge – has investigated how governments and cyber activists combat Islamist terrorists on social media. We fill this gap. With BDT, we demonstrate a clear connection between *Anonymous* reporting, Islamist extremist affiliation, and suspension of jihadist Twitter accounts. Contrary to popular skepticism, we argue that the *Anonymous* campaign to identify and expose Islamic extremists has been successful based on three sets of results.[3] First, machine learning models that include variables associated with the Anonymous campaign (e.g. the number of times that a suspicious jihadist Twitter account has been reported by each of the detected Anonymous's activists) significantly outperforms machine learning models that do not include such variables in terms of predictive accuracy. Second, the

[3] During an interview with the *Daily Dot* in November 2015, a Twitter spokesperson claimed that the *Anonymous* list of suspicious jihadist twitter accounts is "wildly inaccurate" and that the company ignores the list.[4] Similarly, the technology information website *Ars Technica* claims to have reviewed an *Anonymous*-curated list of 4,000 suspicious jihadist Twitter accounts, and found them to include accounts that are "trolling" IS or are simply Arabic.[5] Furthermore, we know that IS supporters would submit spam reports of "false positives" to counter the *Anonymous* campaign.[6]

number of times that a suspicious jihadist account has been reported is the most powerful predictor of affiliation with Islamist extremism and suspension. Third, our BDT model shows that 87 percent of the Twitter accounts reported by Anonymous are likely to be associated with jihadism.

Before proceeding, we note that we decide to illustrate the ML techniques with an analysis of *Anonymous* and Islamist terrorists on Twitter not only because of its substantive significance, but also because the *Anonymous* operation against Islamist terrorists is interesting for political methodologists interested in crowd sourcing. In essence, the *Anonymous* campaign is the largest crowd-sourced multi-lingual text analysis in history; in March 2016 alone, *Anonymous* volunteers have publicized 120,000 reports on suspicious jihadist Twitter accounts. As Benoit et al. (2016) demonstrated, crowd workers can perform classification tasks as well as experts; our machine learning analysis provides support for Benoit et al's finding.

The rest of this article is structured as follows. We first discuss the strengths and weaknesses of various ML models and introduce our model of choice, BDT. Subsequently, we introduce an original dataset on *Anonymous* and Islamist terrorists on Twitter, before running a series of ML analyses to justify our choice of BDT for our classification task and training a BDT model to assess the informational value of *Anonymous* reporting. We conclude by discussing the implications of this study, in particular how this study lays the foundation for future researchers to build a supervised classification ML model that can chart a comprehensive distribution of jihadist presence on Twitter over time.

2 Machine learning for classification and prediction

2.1 Major families of Machine Learning Models

Different classes of ML models rely on distinct generic algorithms to automatically construct predictions for sample inputs, which are split into one part that is used for training the model and another part that is used for evaluating the prediction quality of the trained model. The power of ML models lies in their ability to automatically extract information from a subset of data to uncover patterns that are subsequently generalized to the rest of the dataset.

Of course, we can also rely on classic regression models for classification and prediction. However, classic models, such as Ordinary Least Squares (OLS) linear or logistic regression models, often make strong assumptions regarding the data generating process (e.g. OLS models assume that errors are uncorrelated with the regressors). In contrast, ML models make no assumption about the distribution of data. This allows ML models to readily take nonlinearities and complex interactions among predictive variables into account when researchers utilize them for classification/ prediction. Consequently, ML models can often classify/ predict at much higher levels of accuracy compared to classic regressions (Beck, King and Zeng 2000).

ML models' flexibility to analyze data with complex structure stems from its philosophy. Classic regressions produce the output (e.g. dependent variable) from the input (e.g. explanatory variables) given a predefined protocol concerning the data generating process. In contrast, ML models generate such protocols by analyzing the output and the input concurrently with the training set (Figure 1).

[Figure 1 goes about here]

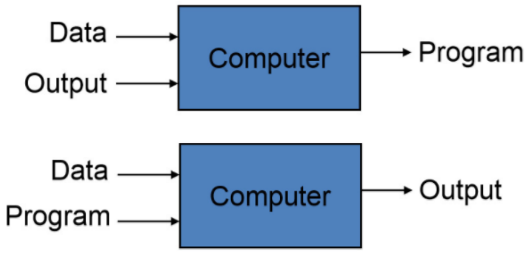


Figure 1: Machine learning (on top) and the “usual” statistical analysis (based on Barnes (2015))

There are three main classes of ML models based on distinct underlying principles (Murphy 2012).[7] The first class of ML models constructs a division that separates an *n*-dimensional (*n* is the number of features in the input) space into two subspaces that correspond to a binary outcome (e.g. jihadist and non-jihadist); the algorithm constructs that division as a function of the points in the training set (e.g. the "support vectors"). The simplest and most popular version of this class of models is the Support Vector Machine (SVM) models that use the hyperplane – the linear border of the dimension n-1 – as the separator. Importantly, the classical SVM is a non-probabilistic binary model, since it only

[7] In this paper, we focus on non-Bayesian ML models, because Bayesian ML models – e.g. Bayesian additive regression tree (Chipman et al. 2010) – require the researcher to assign joint prior distributions to all parameters. This can be difficult when the researcher is dealing with a large dataset with many parameters (as in the case of our dataset on suspicious jihadist Twitter accounts with more than 500 variables). Crucially, setting the priors as uniform does not indicate that the researcher is ignorant of the priors. As Syversveeen (1997) puts it, "not knowing the chance of mutually exclusive events and knowing the chance to be equal are two quite different states of knowledge".

predicts on "which side" of the estimated hyperplane an input point lies, but it does not provide any likelihood prediction. Researchers have used SVM to classify U.S. Congressional Bills (Purpura, Wilkerson and Hillard 2008), but political scientists are generally unfamiliar with this class of ML models.

Second, neural network models take inspiration from biological neural networks, where each "neuron" (node) applies a mathematical transformation to the "signal" (input) it receives before passing the transformed signal on to other neurons. Neural network models have multiple layers with various nodes (each layer is a system of nodes with the functions that transform the inputs).[8] The most basic version of a neural network model is the logistic regression, which is a neural network with only one layer with the logistic function. Neural network models, unlike SVM models, generate probabilistic predictions. Researchers of politics are more familiar with neural network models compared to SVM models; they have utilized neural network models for tasks ranging from predicting international conflict (Beck, King and Zeng 2000, De Marchi, Gelpi and Grynaviski 2004) to detecting political ideology in U.S. congressional debates (Iyyer et al. N.d.).

The third class of ML models is based on directed *acyclic graphs* (DAGs), such as decision trees (Montgomery and Olivella Forthcoming). A decision tree depicts a sequential series of binary decisions, and each of these decisions is related to one feature of the data. Formally, a decision tree depicts a sequence of pair (j, t), with j denoting a particular feature and t the rule that dictates which branch of the tree we should go down. Each pair

[8] Neural network models are often represented by a directed weighted graph where the nodes are functions.

corresponds to one node in “the tree”. In a binary tree, if the input’s value of j satisfies t, the next rule to look at is the left one down the tree, otherwise, the right one and so forth until we reach the terminal node that does not have any subsequent branches. To tentatively illustrate how a decision tree works, consider a dataset on Twitter accounts with only two binary variables: (1) mentioned "jihad" in the profile description; (2) received more than 10 reports from *Anonymous*. An example of a decision tree that seeks to classify whether an account is jihadist or not will first ask whether the account profile has mentioned jihad. If the answer is no, the account is not jihadist. If the answer is yes, we ask if the account has received more than 10 reports. If the account has received more than ten reports, the account is jihadist. Otherwise, the account is not jihadist.

There are two main categories of models based on decision trees, which rely on different methods to create an assembly of decision trees for classification. The first category of decision tree models is random forests, the canonical version of which is a generic algorithm that iteratively samples the training dataset with replacement to train decision trees, before aggregating the predictions (e.g. by averaging them) (Breiman 2001). Political scientists have used the model to investigate topics from civil war onsets (Muchlinski et al. 2015) to Russian military discourse (Grimmer and Stewart 2013). *Decision jungle*, on the other hand, is a new variation of random forests that utilizes general directed acyclic graphs (DAGs) as a weak learner instead of decision trees (Shotton et al. 2013). DAGs, compared to decision trees, allow for multiple paths to a node. Decision trees are a subtype of DAGs; also we can consider DAGs as "advanced" decision trees. Decision jungles, compared to random forests, utilizes computer memory more economically.

The second category of decision tree models is Boosted Decision Tree (BDT). BDT, as in the case of random forests, also creates an ensemble of decision trees for a classification task. They differ in how they select the observations from sample inputs to train each of the decision trees in the ensemble. Random Forrest bootstraps the observations, e.g randomly selects observations from the training set (with replacement). BDT, in contrast, selects the observations misclassified by the previous decision tree(s) to train a new decision tree. Computer scientists, for instance, Breiman (one of the founding fathers of ML), frequently consider BDT as the best ML model around (Murphy 2012). Most importantly, BDT is resistant to overfitting, which is the problem when a ML model learns the noise/ random fluctuation in the training dataset too well. A model with overfitting will perform poorly when analysts apply the model to analyze the test set.[9] Overfitting is a particular malaise that plagues many complex ML models (e.g. SVM), which can pick up almost any pattern from the training set. In the next section, we will discuss in detail why BDT is resistant to overfitting.

2.2 Boosted Decision Tree Model

As discussed briefly earlier, BDT is an ensemble machine learning method based on a succession of boosted decision trees where every subsequent tree corrects for the errors

[9] Underfitting, on the other hand, is the problem when a ML model does not model the training dataset, e.g. it does not extract sufficient information from the data. This is a problem often associated with simple ML models, e.g. logistics regression.

made by the previous tree ("boosting"). The trained prediction model employs the entire ensemble of the trained decision trees. Each decision tree on its own often has little predictive power, but a combination of them can have significant predictive power.

As in the case of all ML models that employ an ensemble of decision trees, BDT is well-suited to analyze a dataset where variables are highly inter-correlated and the number of the observations is large, e.g. when the researcher is dealing with big data. Furthermore, boosting enables BDT models to pick up patterns in the dataset without overfitting (Schapire et al. 1998), because the algorithm puts more weight on the most influential features and balances prediction over multiple trained weak learners, e.g. *shallow decision trees*. "*Shallow*" indicates that the number of decision tree leaves is restricted to a small number relative to the number of variables. For instance, in our dataset, we have more than 500 variables, but the trained decision trees in our final model have only twenty leaves. Each trained tree - a weak learner - does a fair job for slightly different subsets of the observations and combined they build a robust model - a strong learner.

Mathematically, the boosting algorithm “collects” the aggregating predictor function $f(x)$ for the outcome variable y, where x are the input features:

$$f(x) = f_0(x) + \frac{1}{M} \sum_{m=1}^{M} \hat{\phi}_m(x) \tag{1}$$

$\hat{\phi}_m(x)$ is the weak learner trained at the step m (where M is the total number of the decision trees in the assembly), and $f_0(x)$ is the prior prediction function, if there is any, otherwise we assume that $f_0(x) = \{\emptyset\}$. For the purpose of illustration, consider a dataset that consists of two observations. In the first round, let the weak learner correctly classify the outcome of observation 1 but misclassify the outcome of observation 2. Consequently, in the second round, the boosting algorithm will place more weight on the weak learner that correctly classifies observation 2 instead of observation 1. After these two rounds, the boosting algorithm generates a classification rule that is a linear combination of two distinctly optimized weak learners. Our BDT model combines the results from a collection of 1000 shallow decision trees.[10]

3 Empirical analysis

In this section, we first discuss the data collection process and supply background information on the *Anonymous* campaign against Islamist terrorists. Second, we show that Twitter accounts on the *Anonymous* list are suspended because they are related to Islamist extremism (and not for other violations of Twitter regulations); this allows us to claim that by predicting suspension of Twitter accounts on the *Anonymous* list, we are also predicting their jihadist affiliation. Third, we run a series of validation tests to show that BDT outperforms other ML models for our classification task. Fourth, we isolate the predictive power of *Anonymous* reporting with our BDT model with permutation analysis, which gives researchers some leverage to assess the individual effect of each parameter on the outcome variable. Fifth, we estimate the likelihood that each *Anonymous*-reported

[10] For a more technical discussion, see Appendix G.

Twitter account is jihadist with our ML model.

Before beginning, we want to stress that the following analysis is based on three assumptions, which we will discuss in more detail below. First, not all Twitter accounts in our dataset are related to Islamic extremism. *Anonymous* volunteers make mistakes. The level of expertise among *Anonymous* activists varies. Second, suspension of an *Anonymous*-reported account by Twitter indicates a high likelihood that the account is associated with Islamic extremism. Third, it can take some time for Twitter to suspend accounts given its review process. Our dataset, therefore, contains false negatives (jihadist accounts that Twitter has yet to suspend).

3.1 A new dataset on the *Anonymous* campaign against Islamic extremism: background

Anonymous is a loose international hacker collective formed around 2003 on *4chan,* an English-language imageboard site. The group professes to fight for freedom of speech on cyberspace, and it has garnered much notoriety with high profile cyber-attacks on the Church of Scientology (2008), *PayPal* (2011), Arab Dictatorships (2011), and U.S. government agencies (2012). Critics, such as the FBI, consider *Anonymous* "domestic terrorists"[11] while admirers see *Anonymous* as "Robin Hoods" in cyberspace.[12] In 2012, *Time* identified *Anonymous* as one of the top 100 most influential "people" in the world for its "taste for

[11] "FBI put Anonymous 'hacktivist' Jeremy Hammond on terrorism watchlist." *The Guardian.* [link]

[12] "From Anonymous to shuttered websites, the evolution of online protest." *CBC News.* [link]

shock humor and disdain for authority" and "an ever shifting enemies list".[13]

After the *Charlie Hebdo* attack on January 7, 2015, *Anonymous* declared war on the Islamic State (IS) and Islamic extremists on social media more broadly.[14] Subsequently, *Anonymous* initiated a series of campaigns including #OpISIS, #OpParis (in response to the November 13, 2015 Paris IS attack), and #OpBrussels (in response to the April 11, 2016 Brussels IS attack). The *Anonymous* campaign against Islamist terrorists consists of three components: (1) curating a list of suspicious jihadist Twitter accounts and reporting those accounts to Twitter; (2) hacking into jihadist websites and Twitter accounts and; (3) stealing bitcoins from Islamic extremists online. The first component of the campaign is at the heart of the *Anonymous* operation against Islamist terroristson social media, and has attracted the most media attention.

The *Anonymous* operation to identify and report Islamic extremist Twitter accounts consists of four stages. First, *Anonymous* mobilizes thousands of volunteers to identify accounts that promote Islamic extremism. The rookie volunteers locate suspicious jihadist Twitter accounts by searching for those accounts that tweet hashtags associated with Islamic extremism (e.g. #AllEyesOnISIS), before manually examining all the accounts that the suspicious accounts followed and the followers of the suspicious accounts. The "elite" volunteers, on the other hand, write computer programs to capture names of accounts that follow prominent IS members. Second, after identifying the suspicious accounts, *Anonymous* volunteers tip a team of around 30 core *Anonymous* activists (Twitter handles

[13] "The World's 100 Most Influential People: 2012". *Time*. [link]

[14] On jihadism, see Nielsen *forthcoming*.

nicknamed "CtrlSec"). Third, CtrlSec reviews the tips from the volunteers and publicizes the suspicious IS accounts on one of its official *Anonymous* Twitter accounts.[15] Fourth, thousands of *Anonymous* volunteers (often with the assistance of automated scripts) will then flag the suspicious accounts posted on the *Anonymous* list to Twitter for review. Fifth, after receiving the complaints, the Twitter surveillance team manually checks each reported account to decide whether to suspend it.

Our dataset consists of all profile information from accounts reported as jihadist by *Anonymous* between 3/16/2016 and 10/14/2016, in addition to the number of times that each account has been reported by *Anonymous*, and the *Anonymous* Twitter handle that report the account.[16] Each time an account was reported on any of the official *Anonymous* Twitter accounts, our computer program scrapes all the profile information associated

[15] *Nota bene*: it is unclear what criteria CtrlSec uses to determine whether an account is indeed jihadist, or how much effort CtrlSec has dedicated to the review process. *Mikro*, the hacker who leads CtrlSec, claims that "you [just] need two eyes and brain" to identify jihadist Twitter accounts in his interview with the *Atlantic*. [link]

[16] Our dataset only covered one of the three major *Anonymous* campaigns against IS, #OpISIS and #OpParis. Nonetheless, we argue that #OpBrussels deserves particular attention compared to the two earlier major *Anonymous* campaigns against IS on Twitter, for two reasons. First, while cyber security experts and Twitter have examined (although not in a rigorous fashion) the *Anonymous* list of suspicious jihadist accounts associated with #OpISIS and #OpParis, there has yet been no effort to analyze #OpBrussels. Second, *Anonymous* suffered heavy negative media coverages for its wildly inaccurate lists of suspicious jihadist Twitter accounts during #OpISIS and #OpParis, which included accounts associated with Al Jazeera, BBC news, Obama, Hillary Clinton, academics, journalists, and Arabic speakers. In response, CtrlSec redoubled its effort to review volunteer tips in 2016 and encouraged volunteers to also tip *Anonymous* for mistakes (even setting up a special portal on its website for the purpose). Consequently, #OpBrussels is arguably the most mature *Anonymous* campaign against IS, where the core volunteers have set up an infrastructure that would facilitate more accurate reporting of jihadist activities on Twitter.

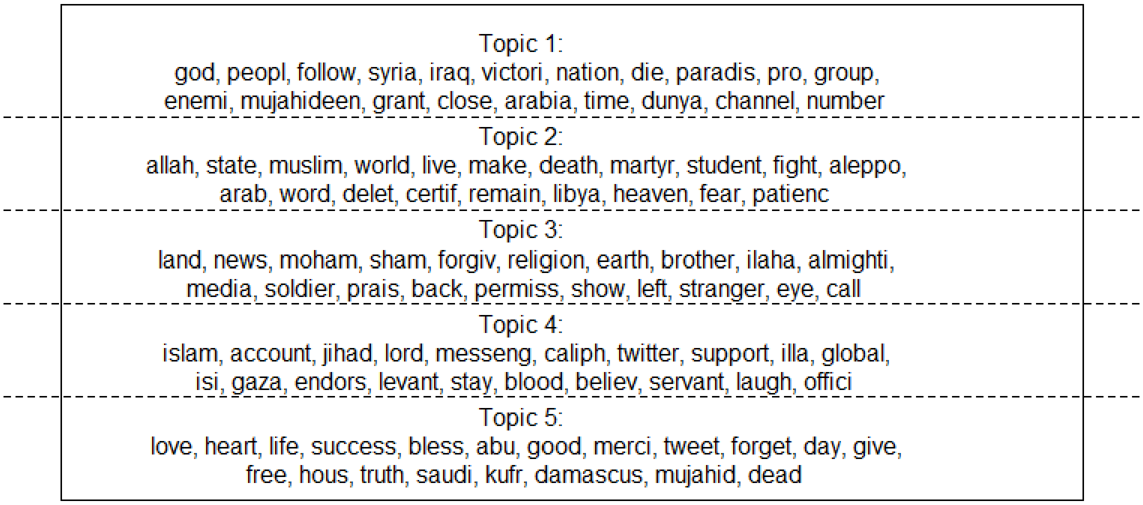
with that account in *real time*. For detailed information on the dataset, see Appendix Â§C. Roughly, our data collection process begins at #OpBrussels and ends at the *Anonymous* "civil war" over its operation against Islamist terrorists.[17]

3.2 Analyzing the profile descriptions of the reported Twitter accounts

We begin our analysis by analyzing the profile descriptions of the Twitter accounts reported by *Anonymous*. Semantic topic analysis reveals that the profile descriptions of *Anonymous*-reported Twitter accounts center on five topics: (1) Islamic cosmology; (2) martyrdom; (3) piety; (4) jihad; (5) religious blessings (see Figure 2); the labels above correspond to our interpretations of how key phrases cluster around a topic.

[Figure 2 goes around here]

[17] The *Anonymous* campaign against Islamic extremists fell into disarray as *Anonymous* fought over two issues. Is the *Anonymous* campaign developing too close of a relationship with the government and cyber security companies? Second, has the *Anonymous* campaign become a means for some to fulfill their desires for fame? See "Inside Anonymous' 'Civil War' Over Its Fight With ISIS", *Motherboard*, November 4, 2016. [link]



Topic 1:
god, peopl, follow, syria, iraq, victori, nation, die, paradis, pro, group, enemi, mujahideen, grant, close, arabia, time, dunya, channel, number

Topic 2:
allah, state, muslim, world, live, make, death, martyr, student, fight, aleppo, arab, word, delet, certif, remain, libya, heaven, fear, patienc

Topic 3:
land, news, moham, sham, forgiv, religion, earth, brother, ilaha, almighti, media, soldier, prais, back, permiss, show, left, stranger, eye, call

Topic 4:
islam, account, jihad, lord, messeng, caliph, twitter, support, illa, global, isi, gaza, endors, levant, stay, blood, believ, servant, laugh, offici

Topic 5:
love, heart, life, success, bless, abu, good, merci, tweet, forget, day, give, free, hous, truth, saudi, kufr, damascus, mujahid, dead

Figure 2: STM: Featuring words

Topic 1's featuring words include "dunya" (which refers to the temporary world" "god", and "paradis[e]", which are related to Islamic conception of the world. Topic 2's featuring words include "death", "martyr", "fight", "heaven", which are related to sacrifice for Islam and its rewards. Topic 3's featuring words include "moham" (short for Prophet Mohammad), "ilaha" (divinity of god) and "prais[e]", which pertain to the veneration of Islam. Topic 4's featuring words include "jihad", "caliph", and "global", which are related to the building of an Islamic empire. Topic 5's featuring words include "bless", "truth", "love", which refer to rewards that pious believers will enjoy.[18]. Note that although topics (1) and (5) look “harmless” upon first glance, but all topics' featuring words include

[18] Appendix F has more information about the optimization of the number of semantic topics (see Figure 3 in Appendix F)

mujahideen. Therefore, these topics may actually concern how dedicating oneself to exterminating the infidels fit Islamic theology. Among the reported accounts, martyrdom is the most popular topic, followed by piety, jihad, Islamic cosmology, and religious blessings (see Figure 3); our STM analysis reveals that almost all of the accounts are related to religion (and specifically Islam for topics 1-4). Last, profile descriptions of the suspended Twitter accounts compared to the active accounts are almost identical, except that the suspended accounts are more likely to talk about Islamic cosmology in their profile descriptions (see Figure 4).

[Figure 3 goes around here]

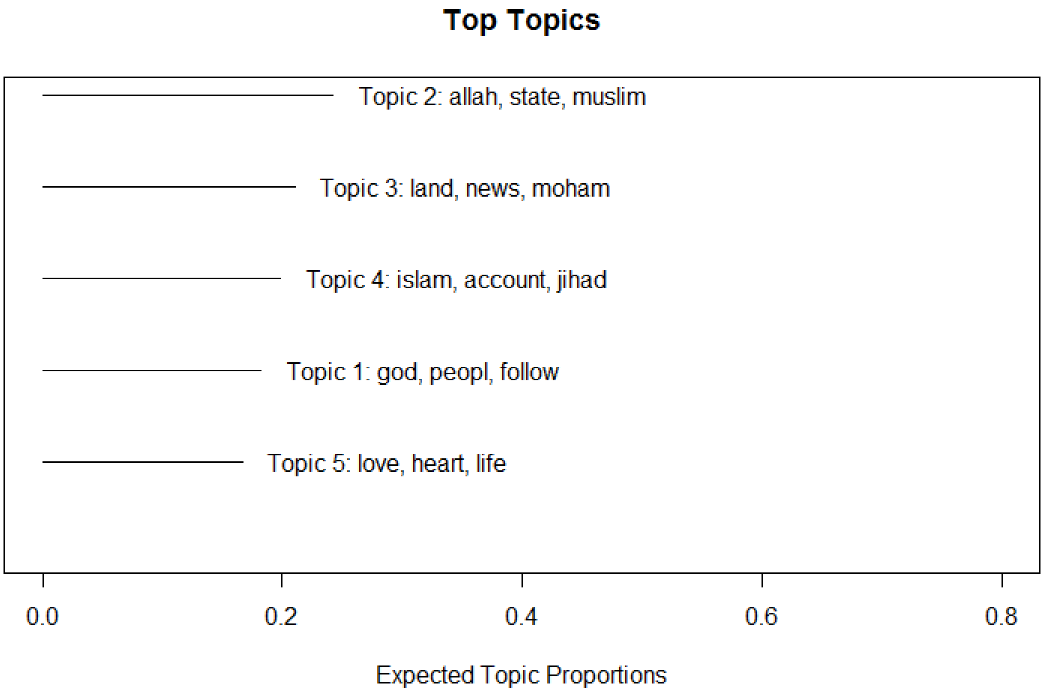


Figure 3: STM: Topic proportions

[Figure 4 goes around here]

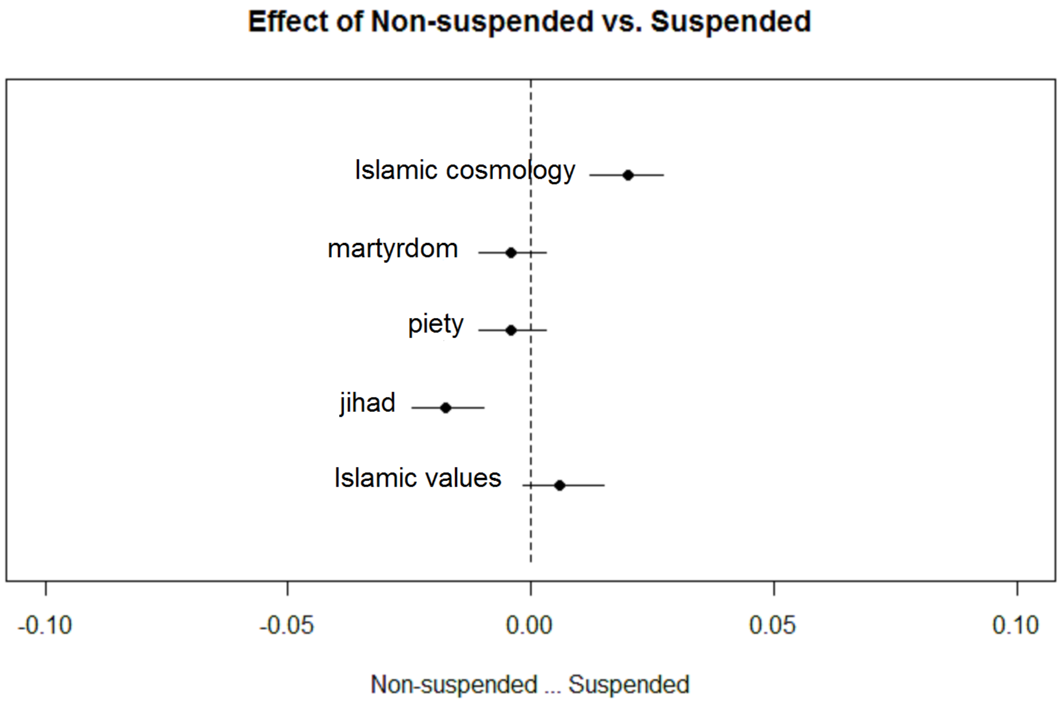


Figure 4: STM: Effect of the suspended status on the topic distribution

We know that Twitter suspends only three types of accounts: (1) bots; (2) hacked or compromised accounts; (3) abusive accounts.[19] Abusive accounts including accounts that spread child pornography or right-wing and religious extremism. Our analysis of profile descriptions shows that the accounts from the *Anonymous* list are actually related to Islam, and not either child pornography or ultra-right extremism. Furthermore, the share of users who might be bots in our sample is small, and their inclusion or exclusion does not drive the results (see Appendix E for more details). We can therefore assume that all suspended accounts from the Anonymous list are believed by Twitter to be jihadist accounts.

[19] The Twitter Rules. https://help.twitter.com/en/rules-and-policies/twitter-rules

We consider the Twitter review process an inter-coder reliability test. The *Anonymous* volunteers first label the suspicious jihadist accounts, while the Twitter surveillance team double checks the veracity of *Anonymous* reports before making a decision on suspension. Twitter takes a *passive* approach to blocking accounts with abusive content, e.g. Twitter only investigates an account for abuse after receiving a complaint from its users (for instance, the *Anonymous* volunteers). As an account receives more complaints, it moves up a queue of accounts that received complaint(s) for the Twitter surveillance team to review manually.

In brief, the key to evaluate the reliability of the *Anonymous* list is to the ability to predict which of the reported accounts will eventually get suspended, even if they have not been suspended yet (e.g. because Twitter has not got the chance to review them yet).[20]. For this task, we now turn to selecting and training a ML model for the classification/ prediction task.

3.3 Selecting the optimal machine learning model

To select the optimal ML model, we run a ten-fold cross-validation test where we utilize ten randomly selected subsets of data of equal size as test sets to compare model performances across a number metrics.[21] We fit five different ML models for the ten validation

[20] *Nota bene*: We define *prediction* in a classic machine learning sense - the possibility to train a model based on a random subset of data (the training set) to confidently predict the labels (suspension or not) for the rest of our data (the test set) (Murphy 2012)

[21] Algorithm to do the ten-fold cross-validation: 1) repeat 10 times: choose randomly 10% of the data into the test set, train the model on the 90% rest of data, evaluate the performance of the model on the test set, store the result 2) present the mean and the

tests to predict suspension. Our data include profile features of suspicious jihadist accounts (we use Π to denote the vector of these variables, e.g. language of the account, profile description, and number of friends) and data on how the *Anonymous* campaign targets each suspicious account (we use Γ to denote the vector of these variables, e.g. the number of times each account been reported by *Anonymous*, the *Anonymous* Twitter activist handle that reported the account).[22]

[Table 1 goes about here]

Table 1: Types of the outcomes in the prediction problem

	Predicted Positive	Predicted Negative
Actual True	TP	FN
Actual False	FP	TN

[Table 2 goes about here]

Table 2: Evaluation metrics

metric	definition
Accuracy	(TP + TN)/N
Precision	TP/(TP + FP)
Recall	TP/(TP + FN)
F1 Score	2*precision*recall/(precision+recall)
Area Under the Curve (AUC)	the area below the ROC[23]

Note: ROC = PRECISION/RECALL curve

We evaluate the fitted models by examining the following five metrics (see Table 2 and Table 1): *accuracy*, which is the share of correct predictions among all predictions made; *precision*, which is the share of correctly predicted suspension among all predicted

standard error of the 10 vectors with the stored results.

[22] For descriptive statistics of Π see Tables 8, 9 and 10 in Appendix C. For descriptive statistics on Γ see Table 11 in Appendix C.

suspensions; *recall*, which is the share of correctly predicted suspensions among all actual suspended accounts; *F1 Score*, which is a metric that combines *precision* and *recall*; *AUC*, the intuition of which is that it denotes the probability to correctly predict its actual status for a randomly selected Twitter account from the test set.

The ten-fold cross-validation test (Table 3) reveals that BDT outperforms all other ML models across almost all metrics except *recall*; Decision Jungle performs the best for *recall*, but its sub-optimal performance for other metrics – e.g. *accuracy* – suggests that it tends to over-predict the positive outcome. We will, therefore, rely on BDT for the classification task of this study.

[Table 3 goes about here]

Table 3: Model comparison: Averages from ten-fold cross-validation

model	Accuracy	Precision	Recall	F-Score	AUC
Random Forrest	0.666	0.643	0.808	0.716	0.734
	(0.010)	(0.015)	(0.018)	(0.007)	(0.010)
Decision Jungle	0.593	0.574	**0.844**	0.683	0.634
	(0.022)	(0.019)	(0.045	(0.023)	(0.026)
Support Vector Machine	0.732	0.722	0.718	0.717	0.795
	(0.012)	(0.053)	(0.053)	(0.026)	(0.018)
Neural Network	0.745	0.754	0.692	0.720	0.812
	(0.015)	(0.052)	(0.045)	(0.027)	(0.017)
Boosted Decision Tree	**0.794**	**0.801**	0.754	**0.776**	**0.863**
	(0.019)	(0.033)	0.052	0.030	(0.026)

Standard errors are given in parentheses. The highest result of a column – in bold.

3.4 The informational value of *Anonymous* reporting

We begin our analysis with BDT by evaluating the "value-added" of *Anonymous* reports for predicting suspension.

We have the following set of information on the *Anonymous* campaign (Γ) in our dataset. First, we know the number of times each suspicious jihadist Twitter account has been reported. Second, we have information on whether each account has been reported by one or more of one of the 250 active *Anonymous* Twitter handles (and the number of times each account has been reported by each handle).

To investigate whether *Anonymous* reporting has any informational value, we examine the prediction metrics of BDT models that: (1) include only information on profile characteristics Π (e.g. content of profile descriptions, number of friends); (2) include only information on *Anonymous* reporting (Γ); (3) include both Π and Γ (= D). Three points follow from the results Table 4). First, the results for Π provide a validation check – the profile features should (and did) predict affiliation with Islamist extremism. Second, Γ provides slightly less information than Π regarding suspension prediction: the accuracy for Π is 0.645 compared to 0.662 for Γ. Third, the full BDT model that includes both Γ and Π outperform the BDT model that includes only either Γ or Π across all performance metrics. In particular, the full BDT model shows approximately 20 percent improvement over the other two BDT models for accuracy, precision, and AUC. These results provide strong evidence that *Anonymous* reporting provides valuable additional information on whether a Twitter account is affiliated with Islamist extremism.

[Table 4 goes around here]

Table 4: Model prediction comparison from the Boosted Decision Tree Model

dataset	Accuracy	Precision	Recall	F-Score	AUC
Profile data	0.662	0.652	0.628	0.638	0.719
	(0.036)	(0.043)	(0.043)	(0.032)	(0.046)
Anonymous reports	0.645	0.646	0.704	0.674	0.691
	(0.007)	(0.009)	(0.015)	(0.005)	(0.014)
Combined	0.794	0.801	0.754	0.776	0.863
	(0.019)	(0.033)	(0.052)	(0.030)	(0.026)

Marginal Effects: Permutation Feature Index

To identify the most influential variables driving prediction of jihadist affiliation (and eventually suspension), we run a series of permutation feature index (PFI) analysis. By design, machine learning models do not provide explicit individual effect estimates the way the standard regression models do.

Consider a dataset with continuous inputs x_1, x_2 and x_3, and output y. We train a ML model that shows f(x_1, x_2, x_3) predicts y. To compute the marginal effect of x_1, we need to specify x_2 and x_3's values, since the marginal effect of x_1 depends on x_2 and x_3. In the case of logit, x_2 and x_3 would be set to their means for such computation. In the case of an ML model, we can theoretically calculate x_1's marginal effect in a similar manner, but this estimate is not meaningful. This is because ML models are designed to capture nonlinearities and complex interaction among inputs, which implies that the marginal effect of x_1 - $\frac{df(x_1,x_2=\bar{x_2},x_3=\bar{x_3})}{dx_1}$ - may be very sensitive to small changes in x_2 and x_3. In other words, if $\Delta > 0$ represent an infinitely small positive number, $\frac{df(x_1,x_2=\bar{x_2}+\Delta,x_3=\bar{x_3})}{dx_1}$ can be completely different compared to $\frac{df(x_1,x_2=\bar{x_2},x_3=\bar{x_3})}{dx_1}$; they might even have different signs.

Instead of calibrating x_i to obtain marginal effect, PFI assesses the influence of variable x_i on prediction by evaluating how sensitive predictions are to the permutations of x_i's value (Breiman 2001). This measure is denoted as *permutation feature importance* (pfi_i; i indexes each variable under consideration). If pfi_i for a variable x_i equals to 0.01, it indicates that the predictive accuracy of the model decreases by 0.01 if the values of this feature are randomly permuted within the test dataset. [24]

[Table 5 goes about here]

[24] Steps to calculate pfi_i:

1. Pick a trained machine learning model, a test dataset, and a evaluation metric (i.e accuracy);
2. Calculate the evaluation metric, p_b;
3. Pick feature i;
4. Randomly permute the values of feature i across the observations of the test dataset;
5. Calculate the evaluation metric for the test dataset after the permutation of feature i, $p_s i$;
6. Calculate $pfi_i = -(p_b - p_{si})$.

Table 5: Permutation feature importance > 0.01 among $\Pi \cup \Gamma$

feature	dataset	Score
number of Anonymous reports	Γ	0.160
days active	Π	0.052
profile description	Π	0.028
location	Π	0.024
account age (in days)	Π	0.020
reported by CtrlSec0	Γ	0.017
status count	Π	0.015
number of Twitter handles that reported	Γ	0.015
favorites count	Π	0.014
reported by CtrlSec	Γ	0.014
language	Π	0.012

Table 5 presents the list of variables that has an effect of more than 1% on our trained BDT model's accuracy. Two results stand out. First, the model's accuracy is the most sensitive to permutation in the total number of *Anonymous* reports each account has received. Second, three additional variables on the list are related to the *Anonymous* campaign: total number of *Anonymous* Twitter handle that reported the account and total numbers of reports from major *Anonymous* Twitter handle *CtrlSec* and *CtrlSec0*. Overall, our permutation analysis shows that variables associated with *Anonymous* reporting are powerful predictors of a Twitter account's jihadist affiliation and its eventual suspension.

3.5 Evaluating the quality of the *Anonymous* list

To assess the quality of the *Anonymous* list (are the accounts reported actually jihadist?), we use our trained *Boosted Decision Tree* model to calculate the probability that each *Anonymous* reported account faces suspension (if it is not already suspended at the time when we finished with data collection). If we assume that suspension indicates allegiance to

Islamist extremism – an assumption that we defended earlier in this paper – we can interpret the probability of suspension associated with each *Anonymous*-reported account as the likelihood that it is indeed jihadist, e.g. "jihadist score". To minimize the problem of overfitting, we calculate the jihadist scores for a randomly selected test set of 4,749 observations (30% of the sample, the rest of the observations are used for the training of the model). Figure 5 plots the distributions of the jihadist scores for the accounts that have already been suspended and accounts that are still active (at the end of our data collection time period).

[Figure 5 goes about here]

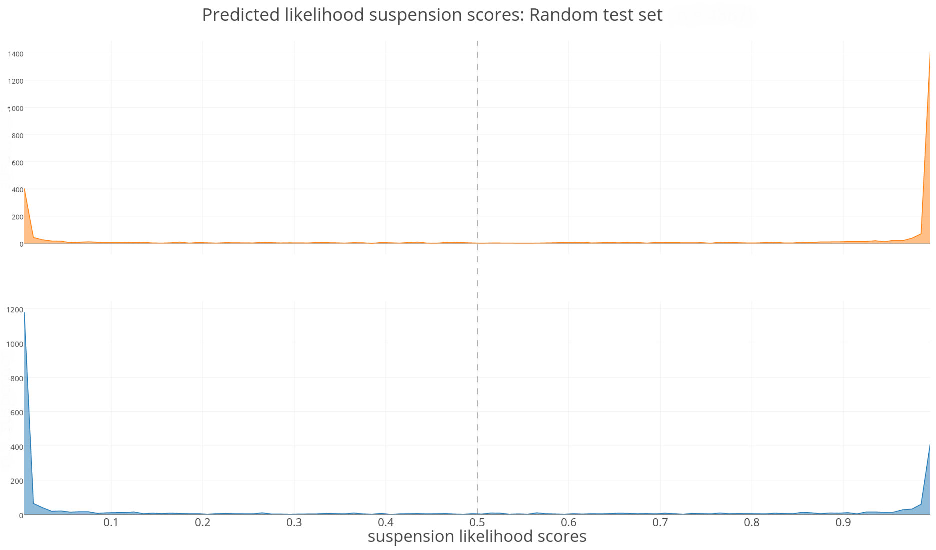


Figure 5: Boosted Decision Tree Model: Prediction scores for the suspended and active accounts (total n=4,749)

The mean jihadist score of the suspended accounts in our test dataset is 0.716, and the median is 0.998. The mean jihadist score of the active accounts in our test dataset is 0.336, and the median is 0.009; note that active accounts can also be IS-related, as the Twitter surveillance team might not yet have the opportunity to review those accounts. Both distributions spike at the edge: the jihadist scores of the suspended accounts spike at 1 (indicating close to 100 percent likelihood that these accounts are indeed jihadist), while the jihadist scores of the active accounts spike at 0 (indicating close to 0 percent likelihood that these accounts are jihadist).

To estimate the percentage of *Anonymous*-reported accounts that are jihadist, θ, we examine introduce the transformation $\Phi(.)$ as a function of the test set D_{test} and the predicted scores calculated with the trained BDT. For that we look at the 95-% one-side confidence interval of the empirical distribution of jihadist scores for the suspended accounts. The value corresponding to the empirical 95%-quantile for this distribution is $5.6 * 10^{-6}$:

$$P(BDTscore \geq \mathbf{5.6 * 10^{-6}} | suspended) \approx 0.95$$

Meanwhile, $P(BDTscore \geq 5.6 * 10^{-6} | active) \approx \mathbf{0.72}$, or 72 % of the active accounts in the test dataset satisfy this condition. If | s | $\subset$ D_{test} is the number of the suspended accounts in D_{test} and | a | $\subset$ D_{test} – of the active accounts, the model predicts that 87% of the observations in the test dataset is affiliated with Islamist extremism (at the 95% confidence level). Formally, the transformation, $\Phi(.)$, to calculate the estimate of θ:

$$\hat{\theta} = \Phi(D_{test}, BDTscores(D_{test})) = \frac{|s|}{|s| + |a|} + \frac{|a|a \geq s_{[|s|(0.95)]}|}{|s| + |a|} \tag{2}$$

where $s_{[|s|(0.95)]}$ is the value of the likelihood score for the [| s |(0.95)]th element in the test dataset arranged in the descending order.

Numerically, for this case:

$$\hat{\theta} = \frac{2,470 + (0.72)2,279}{4749} = 0.8656$$

In brief, our 87% estimate includes all accounts that have already been suspended and 72% of the active accounts in the test data (1,641 accounts).

Uncertainty estimation: a bootstrapping approach

So far we have focused on the point estimate of θ derived from the trained model introduced in 4.3. While the cross-validation test (see Table 4) suggests that our model's prediction is robust, we still need to estimate the uncertainty of θ explicitly. In this section we provide an algorithm (see Algorithm 1) that simulates the distribution θ and determines the upper and lower bounds for $\hat{\theta}$.

[Algorithm 1 goes about here]

```
Initialize $\hat{\Theta}$ - vector of m estimates of $\theta$;
for $m=1 : M$ do
    {Training Set (70%), Test Set (30%)} = Random Split($D$ => 0.7:0.3);
    Trained BDT = Train BDT(Training Set);
    Predicted Test Scores = Trained BDT(Test Set);
    $\hat{\theta}_m$ = $\Phi$(Predicted Test Scores, Test Set);
end
Ascending sort $\hat{\Theta}$
UpperBound$(\theta) = \hat{\theta}_{[M(0.975)]}$
LowerBound$(\theta) = \hat{\theta}_{[M(0.025)]}$
```

Algorithm 1: Uncertainty estimation for $\hat{\theta}$: The 95%-confidence interval

M is the number of simulations, and we start by initializing the vector $\hat{\Theta}$ that represents the simulated distribution of θ. For each simulation, we randomly split the dataset into the training set and the testing set, before training the BDT using the training set. With the trained BDT, we estimate the likelihood scores for the test set before calculating θ_m with $\Phi(.)$. After running M simulations, we arrange all elements in the simulated Θ in ascending order; $\hat{\theta}_{[M(0.975)]}$[25]. is the upper bound and $\hat{\theta}_{[M(0.025)]}$ is the lower bound of the 95-% confidence interval for the simulated distribution of θs.

Our algorithm for calculating the confidence intervals associated with our BDT model's point estimate resembles the bootstrapping (Efron 1992). The algorithm assumes that the sample is the population and obtains the confidence interval from the simulation without making any asymptotic assumption. The difference between our method and classic bootstrapping is that selection into the test set and the training set is performed with no replacement. In our case, we opt for no replacement because our sample is the population, and we do not want to loose any information by throwing out any observations or oversampling any observations.

We ran three analyses with 100, 1000, and 10000 simulations. For M=100, the 95%-confidence interval is [0.8536, 0.8840]; for M=1000, the confidence interval is [0.8532, 0.8817]; and for M=10000, the confidence interval is [0.8537, 0.8815]. These results confirm the robustness of $\hat{\theta} \approx 87\%$.

We wish to conclude this section by explaining why we focused on assessing the re-

[25] The square brackets refer to the integer part of a number

liability of the *Anonymous* list instead of examining how *Anonymous* reporting has contributed to the actual suspension of accounts by Twitter. We made this choice for two reasons. First, the reliability of *Anonymous* reporting is a first order question: without showing that *Anonymous* has identified actual jihadist accounts, it hardly makes sense to investigate whether Twitter has suspended *Anonymous*-reported accounts in the first place. It is possible that *Anonymous* has been reporting accounts that are likely to get suspended by Twitter, but nonetheless unrelated to religious extremism. Second, it is *unfair* for the analysis to assess the efficacy of the Anonymous campaign by using the number of suspended accounts on the *Anonymous* list as a metric. This is because *Anonymous* does not suspend problematic accounts. Twitter does. Thus *Anonymous*'s main contribution to the fighting against Islamist terrorists on social media is the provision of timely and accurate reporting of jihadist activities on Twitter.

4 Conclusion

ML models, despite their analytic advantages, are arguably still under-utilized in political science. This article provides an overview of the most popular classes of ML models in computer science and how to choose the most appropriate ML model. We also show that BDT models, which are widely considered the gold standard of ML models, outperform other ML models in classifying/ predicting jihadist affiliation of twitter accounts on the *Anonymous* list. Subsequently, we introduce new techniques to: (1) isolate the effect of any particular variable on the outcome of interest (permutation feature analysis) and to (2) calculate the confidence intervals associated with ML point estimates. This study provides a practical guide to political scientists on how to utilize the power of ML models (especially BDT) for classification/ prediction tasks.

Furthermore, this article also contributes to studies of terrorism and the politics of social media. With an advanced ML model, we demonstrate that the *Anonymous* campaign against Islamic extremists has been successful in tracking down jihadist presence on Twitter. We provide the first rigorous empirical analysis of the highly publicized campaign between hackers and the religious extremists on social media. Our finding speaks to studies on cyber security, religious terrorism, and the politics of social media, which have focused on social media use in electoral campaigns (e.g. Nulty et al. 2016), social movements (e.g. McCaughey and Ayers 2013), and the relationship between internet and political polarization (e.g. Negroponte 1995 and Bennett 2012). Scholars have paid less attention to how non-mainstream political entrepreneurs – whether they are hackers or terrorists – utilize the internet to advance their agenda.[26] This is an important omission, as the power of the internet lies in its ability to allow previously marginalized political actors to organize and make their voices heard (Kahn and Kellner 2004, Fuchs 2007 chapter 8).

This study also lays the foundation for future researchers to employ BDT in order to: (1) chart a comprehensive distribution of jihadist presence on Twitter over time and; (2) monitor the emergence of jihadist Twitter accounts real time. Our BDT model identifies

[26] With notable exceptions, see e.g. Rowe and Saif (2016), Klausen, Marks and Zaman (2016) and Mitts 2017.

a large number of accounts that are associated with Islamic extremism from the *Anonymous* list. By combining our data on jihadist Twitter accounts with an equal number of non-jihadist accounts randomly selected from the universe of Twitter accounts, future researchers can build a supervised ML model that would allow researchers evaluate whether any Twitter account is related to Islamic extremism. With this article, we hope to encourage political scientists to more fully realize the potential of ML models for social scientific analysis.

References

Alvarez, R Michael. 2016. *Computational Social Science*. Cambridge University Press.

Barnes, Jeff. 2015. *Azure Machine Learning Microsoft Azure Essentials*. Microsoft Press: Redmond, Washington.

Beck, Nathaniel, Gary King and Langche Zeng. 2000. "Improving quantitative studies of international conflict: A conjecture." *American Political Science Review* 94(1):21–35.

Bennett, W Lance. 2012. "The personalization of politics: Political identity, social media, and changing patterns of participation." *The ANNALS of the American Academy of Political and Social Science* 644:20–39.

Benoit, Kenneth, Drew Conway, Benjamin E Lauderdale, Michael Laver and Slava Mikhaylov. 2016. "Crowd-sourced text analysis: reproducible and agile production of political data." *American Political Science Review* 110(2):278–295.

Berger, JM. 2015. "Tailored online interventions: The islamic stateâs recruitment strategy." *CTC Sentinel* 8(10):19–23.

Boyd, Danah and Kate Crawford. 2012. "Critical questions for big data: Provocations for a cultural, technological, and scholarly phenomenon." *Information, communication & society* 15(5):662–679.

Breiman, Leo. 2001. "Random forests." *Machine learning* 45(1):5–32.

Chatfield, Akemi Takeoka, Christopher G Reddick and Uuf Brajawidagda. 2015. Tweeting propaganda, radicalization and recruitment: Islamic state supporters multi-sided twitter networks. In *Proceedings of the 16th Annual International Conference on Digital Government Research*. ACM pp. 239–249.

Chipman, Hugh A, Edward I George, Robert E McCulloch et al. 2010. "BART: Bayesian additive regression trees." *The Annals of Applied Statistics* 4(1):266–298.

De Marchi, Scott. 2005. *Computational and mathematical modeling in the social sciences*. Cambridge University Press.

De Marchi, Scott, Christopher Gelpi and Jeffrey D Grynaviski. 2004. "Untangling neural nets." *American Political Science Review* 98(2):371–378.

Efron, Bradley. 1992. Bootstrap methods: another look at the jackknife. In *Breakthroughs in statistics*. Springer pp. 569–593.

Fuchs, Christian. 2007. *Internet and society: Social theory in the information age*. Routledge.

Grimmer, Justin. 2015. "We are all social scientists now: how big data, machine learning, and causal inference work together." *PS: Political Science & Politics* 48(1):80–83.

Grimmer, Justin and Brandon M Stewart. 2013. "Text as data: The promise and pitfalls of automatic content analysis methods for political texts." *Political analysis* 21(3):267–297.

Hazlett, Chad J and Jens Hainmueller. 2014. Inference in tough places: essays on modeling and matching with applications to civil conflict PhD thesis Massachusetts Institute of Technology.

Iyyer, Mohit, Peter Enns, Jordan Boyd-Graber and Philip Resnik. N.d. Political ideology detection using recursive neural networks. In *Proceedings of the Association for Computational Linguistics*. pp. 1–11.

Kahn, Richard and Douglas Kellner. 2004. "New media and internet activism: from the "Battle of Seattle" to blogging." *New media & society* 6:87–95.

Klausen, Jytte. 2015. "Tweeting the Jihad: Social media networks of Western foreign fighters in Syria and Iraq." *Studies in Conflict & Terrorism* 38(1):1–22.

Klausen, Jytte, Christopher Marks and Tauhid Zaman. 2016. "Finding Online Extremists in Social Networks." *arXiv preprint arXiv:1610.06242* .

Lazer, David, Alex Sandy Pentland, Lada Adamic, Sinan Aral, Albert Laszlo Barabasi, Devon Brewer, Nicholas Christakis, Noshir Contractor, James Fowler, Myron Gutmann et al. 2009. "Life in the network: the coming age of computational social science." *Science (New York, NY)* 323(5915):721.

McCaughey, Martha and Michael D Ayers. 2013. *Cyberactivism: Online activism in theory and practice*. Routledge.

Montgomery, Jacob M and Santiago Olivella. Forthcoming. "Tree-based models for political science data." *American Journal of Political Science* .

Muchlinski, David, David Siroky, Jingrui He and Matthew Kocher. 2015. “Comparing random forest with logistic regression for predicting class-imbalanced civil war onset data.” *Political Analysis* 24(1):87–103.

Murphy, Kevin P. 2012. *Machine learning: a probabilistic perspective*. MIT press.

Negroponte, Nicholas. 1995. “L’homme numérique.”.

Nielsen, Richard A. 2017. *Deadly Clerics: Blocked Ambition and the Paths to Jihad*. Cambridge University Press.

Nulty, Paul, Yannis Theocharis, Sebastian Adrian Popa, Olivier Parnet and Kenneth Benoit. 2016. “Social media and political communication in the 2014 elections to the European Parliament.” *Electoral studies* 44:429–444.

Purpura, Stephen, John Wilkerson and Dustin Hillard. 2008. The US Policy Agenda Legislation Corpus Volume 1-a Language Resource from 1947-1998. In *LREC*.

Rowe, Matthew and Hassan Saif. 2016. Mining Pro-ISIS Radicalisation Signals from Social Media Users. In *ICWSM*. pp. 329–338.

Schapire, Robert E, Yoav Freund, Peter Bartlett, Wee Sun Lee et al. 1998. “Boosting the margin: A new explanation for the effectiveness of voting methods.” *The annals of statistics* 26(5):1651–1686.

Shotton, Jamie, Toby Sharp, Pushmeet Kohli, Sebastian Nowozin, John Winn and Antonio Criminisi. 2013. Decision jungles: Compact and rich models for classification. In *Advances in Neural Information Processing Systems*. pp. 234–242.

Winter, Charlie. 2015. “Documenting the virtual caliphate.” *Quilliam Foundation* 33.

“Boosted Decision Tree (BDT) models for political analysis: Using machine learning to assess the *Anonymous* campaign against Islamic Extremists on Twitter”: Online Appendix

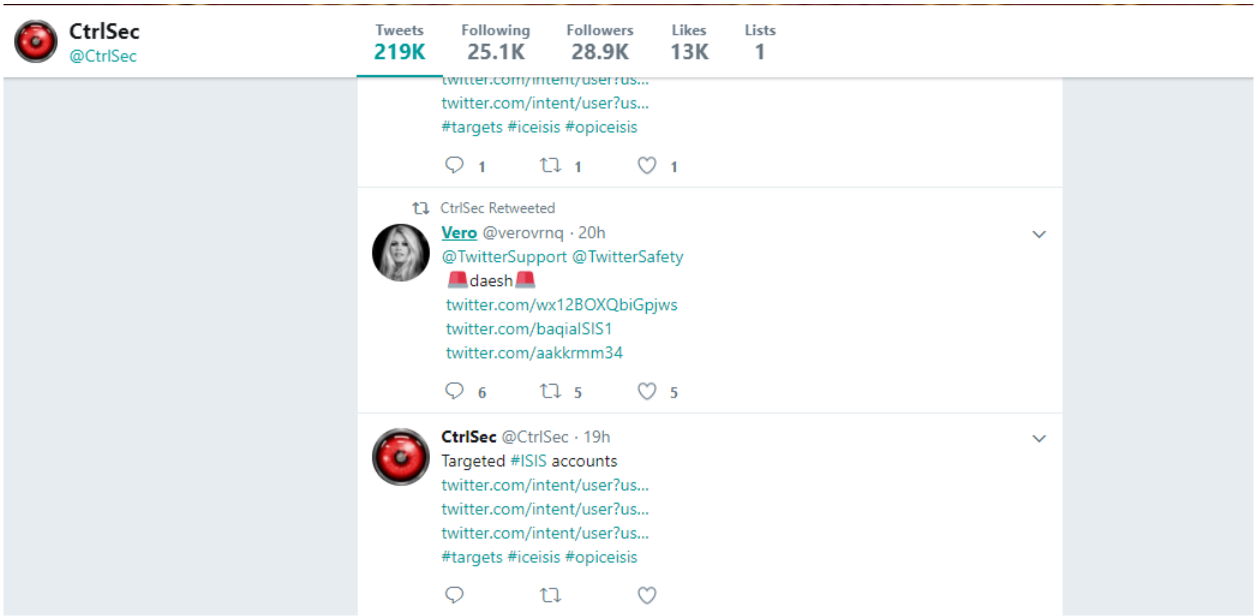
A. Software tools

The original dataset of Twitter accounts was collected with a Python script using the Twitter API wrapper - *tweepy* (Roesslein 2015). All dataset post-processing – data cleansing, aggregation, and translation – was performed in *C# .Net* in *Visual Studio 2015*. *The stm* R-package was used for the structural topic analysis (Roberts, Stewart and Tingley 2014). The machine learning analysis was performed in *Microsoft Azure Machine Learning Studio* (Barga et al. 2015, Microsoft 2017*b*), an online service developed by the Microsoft company that enables developing sophisticated dataflows involving data-processing (i.e SQL data-queries), training/evaluation of machine learning models, and the prediction of the data. We use Microsoft Azure Machine Learning Studio to run our machine learning analysis because the program is optimized for handling big data, user-friendly, and allows the user to utilize many new ML models (e.g. decision jungle) that are not available for R. We will provide codes and instructions on how to implement the analysis in this paper in an online appendix.

B. Monitoring *Anonymous* Twitter handles

We obtain the data for this paper by monitoring *Anonymous* activist accounts - @CtrlSec, @CtrlSec0, @CtrlSec1, @CtrlSec2, @CtrlSec_FR, and @CtrlSec_DE from 3/16/2016 to 10/14/2016 with Python scripts. Figure presents a screen from @CtrlSec's Twitter page.

Each tweet is either posted by the account itself (@CtrlSec in this case) or is a RT of a tweet from one of Anonymous activists. (Tables 1 and 2 show the full list of the detected activist over the period of observation.) Each tweet consists of a list of IS-suspected accounts.



twitter.com/intent/user?us...
twitter.com/intent/user?us...
#targets #iceisis #opiceisis

CtrlSec Retweeted
Vero @verovrnq · 20h
@TwitterSupport @TwitterSafety
🚨daesh🚨
twitter.com/wx12BOXQbiGpjws
twitter.com/baqialSIS1
twitter.com/aakkrmm34

CtrlSec @CtrlSec · 19h
Targeted #ISIS accounts
twitter.com/intent/user?us...
twitter.com/intent/user?us...
twitter.com/intent/user?us...
#targets #iceisis #opiceisis

Figure 1: Screenshot of @CtrlSec's Twitter page

Table 1: Detected Anonymous activist accounts I

___EnIgMa__
_010_isis
_anonsquadno035
PatJohnson
00NuclearBomb00
1nAW3ofGreatIAM
221BriggGarcia
aa_zbs
aa_zbz
AbouFMoiLeCul
abowlled38
adde9708
AdgaAeternus
akasha777
Alexseo10
allofmysoul
Amethyste2332
AndyCurtiss
Anon_Follower1
anon4paz
anonandmore
anonasrn
AnonBocaLeaks
AnonDroidNet
ANONGODESS
anonime1234
anonpaladin
AnonRastaFYB
anony_tetouani
Anonylox
ANONYMOUS_GREY1
Anonymousboss_
AnonymousKite
AntidjihadQcCa
antiharper101
Autarith
AzalCrow
Azkyll
AzureWren
BACFA
BeauLean13
bent__SA
BillBill7542
blabalade

braintwat
British_Ghost_
CallMeCiejeHero
carlste30
ceg1258
Ch3z15m3
Ch3zisMe
CharleneKaprole
ChezisMe
church_equality
Cookiesmeme1
Crisstti
Cruisingman88
CrumpetsGino
CtrlSec
CtrlSec_DE
Ctrlsec_FR
CtrlSec0
CtrlSec1
CtrlSec2
cu_mr2ducks
cyberahsokatano
CyberSec11
Damn_Lucky
DAMSASHH
danp1110
Darkstargoddess
DavidKnipp
DCIntelligence
de_kares
deadfrantz1313
deathonaplate
DeathOnAPlate1
DebashishHiTs
DecadentDissent
DELTA_SEC_OPS
DesireeGuasch
Dormez_Dormez
DreamStateWG
dudefindthebox
Eagle_Eyes01010
elisabettadovi2
eossipov
etabori

eviin__eviin
exoprotein
F________404
Fabrebirth
fayzalmleHan
fearless_war
FenrirReaper
FHD2ksa
Fistula222
ForceSecBot
FoxH2181
FrackenMother
Fred_PORTEFAIX
gingerkull
GLeutz
Global_hackers
GrandTheftPotOh
HalfSkulledHaxr
hapariciog1108
HappyAmazon
HardwayTactics
HeartOfAGypsy77
HippieThugg
holyghostpro
HoustonWelder
hysecotahufi
J0hnLarsen
ja9951
jaglouro
janeannakey
jazz1294u
JDKnowlse
JeffreyKahunas
jerome35800
JNYUTAH
Joaquin_CERO
jocamox1974
jodragon5
John_Conne
jrogj
JTheMagicRobot
JUDUPONT7
K1LL3RB07
KafirHulk

Table 2: Detected Anonymous activist accounts II

kafirkaty
KarenParker93
karjo2000
KawaAri1
Kittyoftheweb
kskaa22
KurtPzrLehr
LaPiky79
Law1Gloria
LilianeDurand
liline018
LokiRedd
lotyzaqokaje
lulzsecrbjb
M_R_TEMPEL
marty713
martydrinksbeer
MaxCUA
Merryman343
metaloona
michaelharrisdr
mkmknani
MLKrepublican
MontmartreClaud
MrB47351012
MrBates1012
muschifuss998
Mystic_2K
N3xCess
nathalie9209
NatvNewYorkr
navegand0
navy8r
nazimbalikci
ndon08
NeoProgressive1
neweraanonymous
NicholleMolly
Ninnin06690177
Noreth7
NotMeUs3434
o_orobo
old_mum
Op629tango
OpReaperSec
oRiGiNaL_ReTuRn
Otaims0o1
p0kN1k
P4sedBlog
Papaversomni
partizankur
pcarlmullan
PhotoTweetyScot
PrettyLaraPlace
PsychicHealerC
PunchyMcgregor
RalphSipani
rav3nsecbot1
redindo9
rektivikasyon
RevalationSaint
ReverantRevan
RigolaxPasDrole
riwired
robyns323
RosenthalEllery
roseOyuma
rxglenn
Saint_Wayne
SaintInan
Samael_StopIsiS
sampuzzo
SapphireKat13
satanic_N
Scarlett210
SEC_SAM
SecretaryMrs
SecularKafir
Segeltexter
sheeple101
shellieRNCEN
ShinyWingsLives
shoonn11
ShortbusMooner
Sin_Feris
SiyanVegan
SolBilgi
SpeKtryZ_
SrFrialdad
sscssjssi
SSN_Reborn
St34lthHunter
ST7757
stang289
StormyVNV
strategicpolicy
sunny_wantsome
SuperSSIAP3
sylvainraillard
TerryMcCracken
TheBigKhuna
Titiduvar
TMTalways
to0of404
TouchMyTweets
Tsipora777
Tviterovska
TweniCheeks
uncleSaul1
USAlivestrong
ushadrons
WANAGL
wantow
westerner222
windwens
XeqtiveCqrity
yetiforhire
zenquando

C. Features

Table 3 presents the original features obtained from the scraped Twitter profile descriptions or was calculated based on the scrapping Twitter profile. While most of the features' names are straight-forward, several clarifications may be required:

- The features of the boolean type are those variables that refer to the personal settings of the user that either turned off or turned on: "default profile","geo enabled", and "has extended profile". One more binary feature is "verified" that confirms the authenticity of identity and is provided by Twitter.

- "language" and "time zone" are categorical and during the analysis they are represented with 32 and 123 dummy variables respectively.

- "decription" and "description translated" (see more about the translation in Appendix.D) are the actual text and its translation from the field "decription" from an account profile; "location" and "location translated" are the self-declared (by the user) location and its translation.

- Table 4 and 5 present the word tokens, the number of occurrences of which in "description translated" and "location translated" are used as the features in the model as well.

Table 3: Features: Profile data (Π)

feature	comment	n variables
account id	integer	0
account age	account age in days	1
time observed	integer	1
default profile	boolean	1
favourites count	integer	1
followers count	integer	1
following count	integer	1
location	text	1
location translated	text	1
geo location enabled	binary	1
has extended profile	binary	1
language	dummies for each of 32 languages	32
profile use background image	boolean	1
statuses count	integer	1
time zone	dummies for each of 123 time zones	123
verified	boolean	1
bot index	followers count /following count	1
description	text	1
description translated	text	1
total		166

Table 4: Profile data: Words from the translated profiles (total = 399)

abdullah
abu
accept
account
accounts
activist
afghanistan
akbar
al
aleppo
ali
allah
allahumma
almighty
america
anbar
anti
approach
arab
arabia
arabian
arabic
army
avoid
awake
back
bagdad
baghdad
baghdadi
bakr
bear
beautiful
believers
belong
beware
bin
black
bless
blessed
blessing
blood
body
book
born
bring
brother
brothers
calculation
caliphate
call
called
care
center
certificate
channel
chest
city
close
closed
coming
continue
country
cross
damascus
dar
date
day
days
dead
dear
dearest
death
debt
defend
deletion
die
dm
dogs
dream
dunya
earth
east
egypt
el
end
endorsement
endorsements
enemies
enemy
engineer
epics
evil
expand
extremist
eye
eyes
face
faith
faithful
fallujah
false
falsehood
family
father
favorites
fear
feet
fight
fighting
find
fire
flag
folk
follow
followers
forget
forgive
forgiveness
foundation
france
free
freedom
front
full
gave
gaza
give
global
glory
god
good
grant
great
ground
group
groups
guide
hacked
hands
hard
hate
head
heart
hearts
heaven
heavens
helping
high
history
hit
holy
home
homs
honour
hope
hour
house
http
https
human
ibn
ilaha
illa
important
independent
indonesian
information
interested
iraq
iraqi
isis
islam
islamic
island
jacked
jerusalem
jihad
jihadist
journalist
kashmir
kill
killed
king
kingdom
knowledge
ksa
kufr
kuwait
la
land
laugh
law
leave
lebanon
left
levant
libya
life
light
live
living
london
long
lord
lost
love
lover
loving
made
make
man
mandate
martyr
martyrdom
martyrs
media
meet
men
merciful
mercy
messenger
met
middle
mind
mohamed
money
muhammad
mujahid
mujahideen
muslim
muslims
nation
news
nice
night
nineveh
nose
number
occupied
official
omar
open
opinion
oppressors
organization
page
pain
palestine
paradise
parents
party
path

Table 5: Profile data: Words from the translated profiles (total = 399)

patience
peace
peninsula
people
permission
person
personal
picture
place
planet
platform
pleased
political
power
praise
pray
prayer
pride
prisoners
private
pro
prophecy
prophet
proud
publish
purpose
put
qaeda
quran
rabbi
rahman
read
religion
remain
remains
remember
repent
replace
represent
researcher
reserve
return
revolution
righteous
rights
riyadh
road
room
rt
satisfaction
satisfied
satisfy
saudi
save
science
security
servant
servants
sham
sharia
shaykh
sheikh
show
sinai
sins
sister
sisters
slave
soldier
soldiers
somalia
son
soul
souls
special
spirit
splendor
state
states
stay
staying
stop
stranger
strangers
strength
student
subhan
succession
sunnah
sunni
support
supporters
sword
syria
syrian
talk
team
telegram
testify
throw
time
tire
tired
today
told
tomorrow
translate
trust
truth
tunisia
turkey
tweet
tweets
twitter
tyrants
ubayy
uk
ul
ummah
unbelievers
understanding
uniform
united
university
vast
victorious
victory
wa
wal
war
weird
wife
win
witness
word
words
work
world
worlds
worship
write
writer
wrong
ya
ye
year
yemen

Table 6 presents the features obtained based on the Anonymous report data. Total reports indicates the total number of the reports of a Twitter account. Number of distinct activist points to the total number of unique activist accounts reported a Twitter account. Then, for each detected activist account, there is a boolean feature, where it has been reported by this account and the integer feature of the total number of reports.

Table 6: Features: Anonymous account reports (n features = 502)

feature	n variables
total reports	1
number of distinct activists	1
reports from an activist account	250
dummy for each activist account	250

Table 7: PFI: Most influential features. Correlations

account age	observed	favourites	statuses	total reports	nbots	CtrlSec(n)	CtrlSec0(n)
1.000	0.364	0.122	0.240	0.272	0.193	0.235	0.267
0.364	1.000	0.098	0.102	0.408	0.239	0.340	0.394
0.122	0.098	1.000	0.326	0.122	0.101	0.113	0.118
0.240	0.102	0.326	1.000	0.097	0.065	0.085	0.095
0.272	0.408	0.122	0.097	1.000	0.714	0.956	0.963
0.193	0.239	0.101	0.065	0.714	1.000	0.681	0.675
0.235	0.340	0.113	0.085	0.956	0.681	1.000	0.895
0.267	0.394	0.118	0.095	0.963	0.675	0.895	1.000

D.Translation

The *description* and *location* fields of the Twitter accounts are translated to English via a cloud-based machine *Microsoft Translator* - using its API for C# .Net - that is part of Microsoft's bundle of *Cognitive Services* (Microsoft 2017*a*). Figure shows the detected languages in the profile descriptions in our dataset.

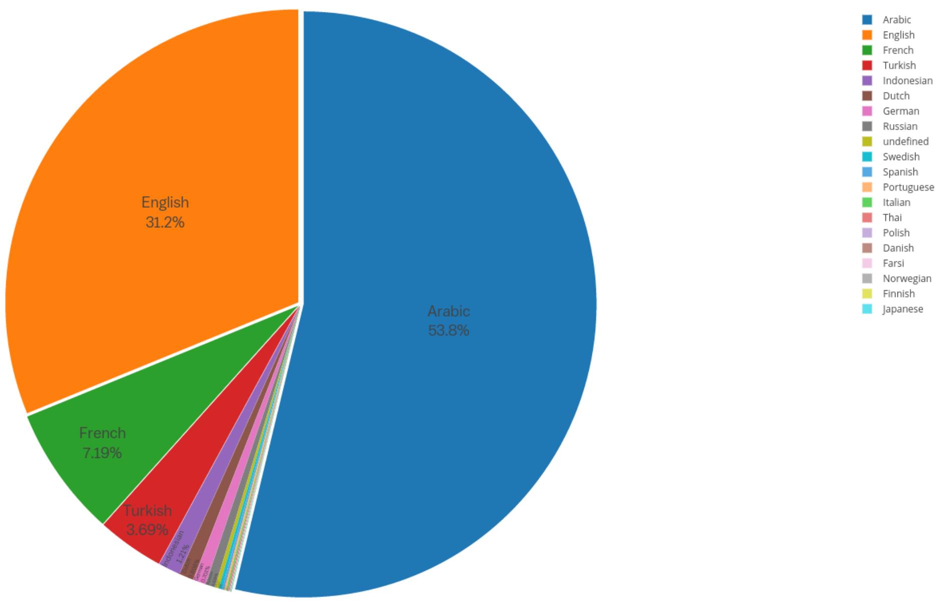


Figure 2: Detected languages of the profile descriptions

E. Bot users

For consistency of our argument in the main text of this paper, we need to address the issue of possible bots in our dataset. The selection of the non-bot accounts might be based on the ratio – followers/following. Indeed, bot users tend to follow significantly more accounts than they are followed. Importantly, bot users might have followers as well – primarily other bot users. To rule them out we apply the conservative rule: followers/following > 0.1.

In the main paper, we do not particular exclude such users to avoid the potential loss of information. Overall only 860 accounts in our sample satisfy the definition (including those who have no friends at all, for them we assume that their number of friends is 0.000001 to avoid division by zero).

Meanwhile, the prediction results not including the likely bots, all features, the boosted decision tree with the same specification as in the main text of the paper, are almost the same: *accuracy* = 0.696, *precision* = 0.660, *recall* = 0.804, *F1-score* = 0.725, and AUC = 0.786.

F. Structural Topic Models: Choosing number of topics

We used the R package *stm* to perform our topic analysis (Roberts, Stewart and Tingley 2014). To select the number of topics, we compare the convergence for three to twelve topics. All of our models converge within less than 75 iterations. Figure shows that the increase of the number of topics decreases the semantic coherence and decreases the residuals monotonically, meanwhile we see a local equilibrium in the held-out likelihood when the number topics is five. Being parsimonious, we decide to limit our analysis to five topics.

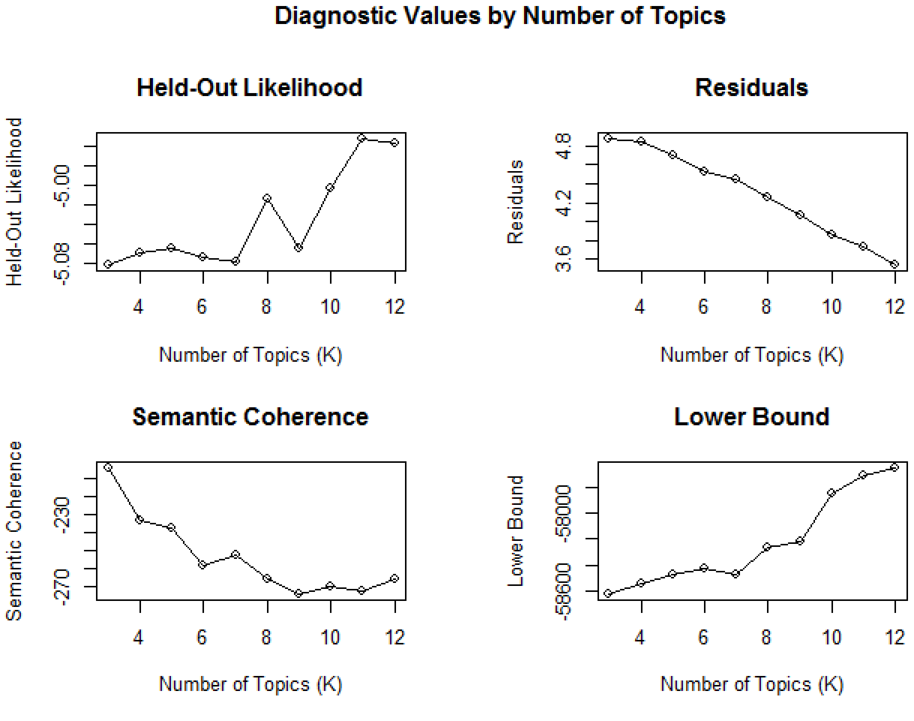


Figure 3: STM: Number of topics comparison

G. Machine learning to determine reporting accuracy

Boosting decision trees: A brief theory review

Boosting constructs the prediction function $f(\mathbf{x})$ as an *Adaptive basic model* (ABM); in case of BDT:

$$f(\mathbf{x}) = f_0(\mathbf{x}) + \nu \sum_{m=1}^{M} \phi_m(\mathbf{x}; \gamma_m) \tag{1}$$

$\phi_m(\mathbf{x}, \gamma_m)$ is *the weak learner* trained at step m of M iterations, where γ_m is the set of the parameters defining a decision tree. $\nu \in (0,1)$ is the shrinkage parameter that reflects how quickly the prediction function updates over the learning process. $f^*(\mathbf{x})$ solves the following optimization problem:

$$f^*(\mathbf{x}) = \underset{f(\mathbf{x})}{argmin} \sum_{m=1}^{M} L(y_i, f(\mathbf{x})) \tag{2}$$

the loss function is defined as $L(y_i, f(\mathbf{x}_i)) = |y_i - f(\mathbf{x}_i)|$ for *the functional gradient descent* employed by BDT.

```
Initialize $f_0(\mathbf{x};\gamma)$ s.t $\gamma = \underset{\gamma}{argmin} \sum_{i=1}^{N} L(y_i, \phi(\mathbf{x}_i;\gamma))$;
for $m$=1 : $M$ do
    Compute the gradient residual $r_{im} = -[\frac{\partial L(y_i, f(x_i))}{\partial f(x_i)}]_{f(x_i)=f_{m-1}(x_i)}$ ;
    Use the Decision Tree model to compute $\gamma_m = \underset{\gamma}{argmin} \sum_{i=1}^{N} (r_{im} - \phi(\mathbf{x}_i;\gamma_m))^2$;
    Update $f_m(\mathbf{x}) = f_{m-1}(\mathbf{x}) + \nu\phi(\mathbf{x}_i;\gamma_m)$;
end
Return $f(\mathbf{x}) = f_M(\mathbf{x})$
```

Algorithm 1: Gradient boosting for the Boosted Decision Tree model

Algorithm 1 describes the steps to build $f(\mathbf{x})$. First, we construct the initial decision tree by fitting the shallow decision tree that minimizes the loss. Then, for m iterations for each observation in the training set, we calculate the gradient residual. Next, we fit the decision tree to its gradient residual. Finally, we update the solution.

References

Barga, Roger, Valentine Fontama, Wee Hyong Tok and Luis Cabrera-Cordon. 2015. *Predictive analytics with Microsoft Azure machine learning*. Springer.

Microsoft. 2017*a*. “Cognitive Services.”.

URL: *https://www.microsoft.com/cognitive-services/en-us/apis*

Microsoft. 2017*b*. “Microsoft Azure Machine Learning Studio.”.

URL: *https://studio.azureml.net/*

Roberts, Margaret E, Brandon M Stewart and Dustin Tingley. 2014. “stm: R package for structural topic models.” *R package version 0.6* 1.

Roesslein, Joshua. 2015. “Tweepy: Documentation.”.