How Many Users Are Enough? Exploring Semi-Supervision and Stylometric Features to Uncover a Russian Troll Farm

Nayeema Nasrin[1,2], **Kim-Kwang Raymond Choo**[1,3], **Myung Ko**[1,2], and **Anthony Rios**[1,2]
[1]Department of Information Systems and Cyber Security
University of Texas at San Antonio
San Antonio, TX 78249, USA
[2]{nayeema.nasrin, myung.ko, anthony.rios}@utsa.edu
[3]raymond.choo@fulbrightmail.org

Abstract

Social media has reportedly been (ab)used by Russian troll farms to promote political agendas. Specifically, state-affiliated actors disguise themselves as native citizens of the United States to promote discord and promote their political motives. Therefore, developing methods to automatically detect Russian trolls can ensure fair elections and possibly reduce political extremism by stopping trolls that produce discord. While data exists for some troll organizations (e.g., Internet Research Agency), it is challenging to collect ground-truth accounts for new troll farms in a timely fashion. In this paper, we study the impact the number of labeled troll accounts has on detection performance. We analyze the use of self-supervision with less than 100 troll accounts as training data. We improve classification performance by nearly 4% F1. Furthermore, in combination with self-supervision, we also explore novel features for troll detection grounded in stylometry. Intuitively, we assume that the writing style is consistent across troll accounts because a single troll organization employee may control multiple user accounts. Overall, we improve on models based on words features by ∼9% F1.

1 Introduction

Social media platforms, such as Twitter, can be helpful in monitoring events, particular for ongoing emergency events (i.e. time-critical situations) (Yin et al., 2015). For example, Twitter has been used to create earthquake monitoring systems by monitoring tweets in real-time (Sakaki et al., 2010). However, Twitter has also become the subject of public scrutiny regarding unwanted actors who are exploiting the social media platform to steer public opinion for their political gain.[1] Twitter, like many other social networking services, has both positive and negative sides of its rendered services. However, when it is used unfairly, malicious actors can manipulate Twitter to influence a potentially large audience by using fake accounts, or worse, by hiring troll farms (Zhang et al., 2016), organizations that employ people to provoke conflict via the use of inflammatory or provocative comments. In general, for this paper, we study models for classifying users as being part of a troll farm.

There has been many inquiries concerning the interference into the 2016 presidential election by the Russian government (Badawy et al., 2018). The Internet Research Agency (IRA)—a troll farm that positioned fraudulent accounts on major social accounts such as Facebook, YouTube and Twitter (Mueller, 2019)—engaged in an online campaign for Russian business and political interests. The IRA's accounts have been created in such a way that they are portrayed as real American accounts. Masking the sponsor of a message such that it appears to originate, and be supported by, grassroots participants is also known as astroturfing (Peng et al., 2017). Based on a 2018 Pew Report, 53% of the Americans participate in some form of civic or political activities on social media during the year (Anderson et al., 2018). Therefore, the magnitude of exploitation by troll farms in influencing opinion on social media is significant. With this growing concern, it is critical that the troll accounts are detected.

Given ground-truth troll farm accounts, researchers have studied if they can develop classifiers to find other members of the troll farm organizations (Im et al., 2019). Even though all the accounts in their dataset are no longer active on Twitter (i.e., they have been banned), based on their classifier, they find that accounts with similar characteristics are still active. However, while social media is swarming with troll accounts (Metaxas

[1] https://nyti.ms/2Uwr36y

Proceedings of the 2nd Workshop on NLP for Internet Freedom: Censorship, Disinformation, and Propaganda, pages 20–30
Hong Kong, China, November 4, 2019 ©2019 Association for Computational Linguistics

and Mustafaraj, 2012), building large datasets of real troll accounts is challenging, especially as new troll farms are formed with different political agendas. It is hard to annotate new troll accounts because they masquerade as citizens, news media outlets, or individual journalists on social media (Paul and Matthews, 2016). Without extensive domain expertise, and external knowledge regarding specific troll organizations, it is challenging for the research community to gather newly annotated users to train more predictive models.

In this paper, we study two specific issues related to troll farm classification. First, we analyze how three different sets of features impacts our classifier's performance. Specifically, we look at *content*, *behavioral*, and *stylistic* features. Based on the political agenda a troll farm is pushing, it is intuitive that there will be common tokens associated with the organization (e.g., #fakenews). However, it is possible that writing style can improve predictive performance. Intuitively, if we assume that certain employees at a troll organization control multiple accounts, then even if the topical information (i.e., content) varies across the accounts, the writing style should be similar. Thus, we hypothesize that features that are predictive for authorship attribution (Sari et al., 2018), can be applied to the troll farm domain.

Second, we study how the number of annotated trolls impacts the classifier's performance. While more data is generally better, there are still many interesting questions that need to be addressed. For example, how many annotated trolls do we need to build a classifier? Would adding more data significantly improve the performance? Can we achieve similar performance using few annotated accounts? What types of errors does the classifier make if we have limited ground-truth troll data? Manually verifying an account as a Russian troll at scale is not feasible. As a result, this leads to an open challenge in text classification i.e., how can we effectively leverage unannotated tweets to improve the classifier's performance. This necessitates the design of a novel/effective machine learning method to detect anonymous fake accounts. Moreover, detecting the bad actors on Twitter/social media that are trying to influence opinion of unaware users will be critical in the future to ensure unbiased elections, and to minimize the impact of information warfare.

Overall, our work is the most similar to Im et al. (2019). In contrast to Im et al. (2019), our work differs in two substantial ways. First, while they explored one set of stylistic features (e.g., stopwords), we ground our work by exploring state-of-the-art stylometric features originally developed for authorship attribution (Sari et al., 2018). Second, their work was focused on showing that troll accounts are likely still out there. Yet, in this manuscript, we are more interested in understanding classifier performance and behavior, not analyzing possible unseen troll accounts still active on Twitter. Moreover, via a detailed error analysis, we study possible biases the classifier has with regards to both false positives and false negatives. For example, the classifier trained using recent IRA data is biased against politically active conservatives, resulting in more false positives.

The contributions of the paper are listed below:

- Based on the hypothesis that a single troll organization employee can control multiple social media accounts, we introduce state-of-the-art stylometric and behavioral features, in combination with standard ngrams, to develop a novel troll detection method. Moreover, we compare content-based features against stylometric/behavioral features, analyzing which group has the biggest impact on classifier accuracy.
- We study how the number of annotated troll accounts affects classifier performance. We also show that simple methods that only use content-based features do not effectively make use of large quantities of training data as well as methods with stylistic and behavioral features. Furthermore, we use a simple, yet effective, semi-supervised method to improve performance in the presence of severe data scarcity.
- Finally, we perform a detailed error analysis across different training set sizes. From the error analysis, we investigate how to improve the model further, as well as analyzing the types of biases the models make, and whether the biases are reduced, or enhanced, by adding more training data.

2 Related Work

Overall, our work is related to three major research areas: Russian troll analysis, text classification, and semi-supervised learning.

Russian Trolls. Researchers have studied Russian propaganda on social media across various domains, including, but not limited to, politics and healthcare. The spread of propaganda is a form of information warfare (Denning, 1999). Broniatowski et al. (2018), for example, explained how Russian trolls discussed vaccine-relevant messages to promote discord. Specifically, they created divisive messages that legitimized the debate by polarization. Their work sought to understand the role played by trolls in the promotion of content related to vaccination. Stewart et al. (2018) studied how Russian trolls polarized topics using retweet network and community detection algorithms. Specifically, they showed that trolls aggravated the context of a domestic conversation surrounding gun violence and race relations. Badawy et al. (2018) explored the manipulation effects by analyzing users that re-shared tweets generated from Russian trolls during 2016 U.S. election campaign. Using bot detection techniques and text-analysis, they identified the percentages of liberal and conservative, showing that most of the tweets were conservative-leaning tweets in an attempt to help the presidential campaign.

Surprisingly, IRA linked accounts, which have been identified by Twitter as evidence and later on submitted to United States Senate Judiciary Subcommittee on Crime and Terrorism, have also been found to be associated with Brexit (Llewellyn et al., 2018). These accounts attempted to promote discord for various topics regarding the European Union and migration. Similarly, the IRA had also participated in the #BlackLivesMatter in accounts identified by Arif et al. (2018). Their work elaborated on how these bad actors impersonated real users to manipulate audiences in accordance to their political agenda.

Text Classification. There are several types of machine learning-based text classification methods available such as generative, discriminative, linear, kernel-based, and deep learning methods. In machine learning, generally text classification is a task of automatically assigning set of predefined categories to unstructured texts. Kim (2014) introduced convolutional neural network for text classification. Yang et al. (2016) introduced a hierarchical attention mechanism that simultaneously weights sentences and words based on their predictive importance. While neural networks have produced state-of-the-art results for a wide variety of tasks, the focus of this paper is on interpretable models with features grounded in stylometry combined with easy-to-understand behavior information.

With regards to interpretable models, Joulin et al. (2016) showed that in many cases linear classifiers still create strong baselines, and are faster than neural networks. Generally, linear classifiers are often faster and more efficient than neural network on large datasets. As we will discuss in Section 3, we use a dataset consisting of 700,000 Twitter users, with more than 17 million tweets. Therefore, for our task, efficiency is important. Moreover, given the recent concern of the carbon footprint of natural language processing models, linear models should continue to be studied (Strubell et al., 2019; Schwartz et al., 2019).

Recently, stylometry-grounded features have been used for authorship attribution, including in malware code authorship attribution (Kalgutkar et al., 2019). For example, Sari et al. (2018) explored the connection between topical (content) features combined with various stylistic features, including, but not limited to, capitalization and punctuation usage. Similarly, Abbasi and Chen (2008) introduced “writeprints”, method of identifying authorship across the internet. They combined traditional features such as lexical, syntactic, structural, content-specific, with idiosyncratic attributes (e.g., spelling mistakes). They utilized a transform-based technique that uses a pattern disruption algorithm to capture feature variations.

Semi-Supervised Text Classification. Finding training data to train a troll classifier is challenging in practice, and results in a needle-in-a-haystack situation. One of the aims of this paper is to study whether large quantities of unlabeled data can be automatically annotated to augment small amounts of training data to more accurately detect Russian trolls.

There has been a lot of work regarding semi-supervision, for both image, video, and text classification (Li et al., 2019; Mallinar et al., 2019). Wang et al. (2009), for example, applied semi-supervised learning algorithms for video annotation. They presented a technique that was developed based on the classical kernel density estimation approach using both labeled and unlabeled data to estimate class conditional probability densities. Habernal and Gurevych (2015) created a clustering-based semi-supervised method to annotate unlabeled text. The aim was to make the model better at identifying scene text with the semi-supervised learning from the unannotated dataset. Rajendran et al. (2016) proposed a semi-supervised algorithm for argument detection. In this work, we primary focus on methods previously developed for other tasks Rajendran et al. (2016, 2018). Specifically, we focus on self-supervision, a model agnostic method of automatically annotating unlabeled data.

3 Data

To be consistent with prior work, our data collection is similar to Im et al. (2019). We provide the basic statistics for our dataset in Table 1. In 2018, federal agents released 3,841 accounts found to be associated with the IRA. We focus on the 2,284 accounts that have selected English as the main language in their profile. Intuitively, we are interested in classifying bad actors that masquerade themselves as a normal user from the United States (US). Note that while most of the tweets are in English, there are occasional tweets in other languages. Furthermore, we collect each user's last 200 tweets, assuming that each user has that many available tweets. We limit to the last 200 tweets because this is the number of tweets we can collect for an active user with a single Twitter API call.

While we have ground-truth troll accounts, we do not have a standardized non-troll dataset. Therefore, we gathered a 701,614 random Twitter accounts constrained to the continental US. Tweets were collected from August 2018 to January 2019. Furthermore, for each account, we retrieved their last 200 tweets, as available. It is important to note that some users posted fewer than 200 times. The collected user's tweets represent our control, or not-troll accounts. Overall, the data is unbalanced, where the control makes up 99.676% of the total accounts, and the Russian troll accounts represent only 0.324% of the entire dataset. The imbalance matches the real-world assumption that troll accounts are rare (Im et al., 2019).

We split the dataset into four groups: Train, Validation, Test, and Unlabeled. Each group contains both troll and control accounts. The unlabeled set is used for training our model using a semi-supervised technique.

	Train	Val	Test	Unlab.	Total
Troll	924	206	229	925	2,284
Control	284,153	63,146	70,162	284,153	701,614

Table 1: Dataset statistics.

4 Method

Based on previous studies (Sari et al., 2018; Abbasi and Chen, 2008; Stamatatos, 2009; Im et al., 2019), in Section 4.1, we discuss the three groups of features we used in our model: stylistic, content and behavioral. Intuitively, we identify trolls by what they say (content) and how they say it (stylistic and behavioral). Furthermore, in Section 4.2 we explain the semi-supervised method (self-supervision) we used to analyze whether unlabeled data can be automatically annotated to improve our model performance.

4.1 Features

We use three groups of features: Content, Stylistic, and Behavioral. In this section, we describe each feature group in details.

Content Features (C). The content features represent the topics that people discuss on Twitter (Sari et al., 2018). To represent content, we use bag-of-words (BoW). This group of features was also used for troll detection in Im et al. (2019). Specifically, we use unigram word counts. Moreover, we limit the vocabulary to the 5000 most common unigrams. The reason we limit the vocabulary is to avoid overfitting. For instance, slight shifts in content may occur over time. However, the broad political agenda that trolls are perpetuating may stay relatively stable. For example, in the IRA dataset, there are many tweets regarding the #BlackLivesMatter movement to promote discord because it was a popular topic on the news at the time (Arif et al., 2018). Ideally, we want to detect when trolls promote discord, not simply remember a few specific topics discussed during a certain time period.

Stylistic Features (S). We adopt the following stylistic features from Sari et al. (2018): average word length, number of short words, percentage of digits, percentage of upper-case letters, frequency of alphabetic characters, frequency of each unique digit, richness of vocabulary, frequency of stop words and frequency of punctuation. These features are both of lexical and syntactic in nature. The number of short words is determined by counting tokens that contain no more than four characters. Richness of vocabulary was calculated by counting the number of hapax and dis legomena, i.e., the number of words that appear only once or twice in the corpus. We also count the frequency of stop words. We use the 179 stop words provided in the Natural Language Toolkit (BIRD and LOPER, 2004). The rest of the features are explained in Sari et al. (2018).

Behavioral Features (B). In a study on political communication on Twitter, it was shown that emotionally charged tweets are retweeted repeatedly and quicker than average neutral tweets (Stieglitz and Dang-Xuan, 2013). Earlier work has shown that hashtags, shared links, and user mention patterns are predictive of Russian trolls (Broniatowski et al., 2018; Im et al., 2019; Zannettou et al., 2019). For our model, we use three behavioral features. Specifically, we calculate the number of times a user adds hashtags, mentions, and links/URLs to their tweets. Intuitively, tweets that repeatedly share links, or use a large number of hashtags, could indicate bot activity, or someone promoting a specific agenda.

4.2 Self-Supervision (Self)

To address the question "How can we automatically annotate unlabeled data?", we use a technique called self-supervision. Intuitively, self-supervision is an iterative method that slowly adds unlabeled instances to the training data. First, the model is trained on the original annotated training dataset. Next, it is applied to the unlabeled dataset. The most confident Russian trolls, based on the classifier score, are added to the training dataset as new troll instances. The process is repeated for a fixed number of iterations. Furthermore, only a fixed number of unlabeled instances k are only added to the training dataset at each iteration. Only unlabeled examples with a score greater than t are added to the training dataset.

4.3 Implementation Details

As our base classifier, we use a linear support vector machine with L2 regularization. We grid-search over the C values 0.0001, 0.001, 0.01, 0.1, 1, and 10. The best C value is chosen using the validation dataset. We repeat the self-supervision process for 25 iterations. Moreover, k is set to 10. Therefore, no more than 10 examples are added during each iteration with a threshold t of 0.

	Precision	Recall	F1
C	0.635	0.738	0.683
CBS	0.745	0.764	0.754
CBS+Self	0.815	0.729	0.770

Table 2: Overall results on the test dataset. The results are generated from models trained on all of the Russian troll users in the training dataset.

	Precision	Recall	F1
Best Model (CBS)	0.761	0.772	0.766
- CB (without S)	0.668	0.723	0.695
- CS (without B)	0.785	0.602	0.681
- BS (without C)	0.595	0.578	0.586

Table 3: Ablation results using the validation dataset for the three major feature groups: Stylistic (S), Behavioral (B), and Content (C). The results are generated from the model trained on all of the Russian troll users in the training dataset.

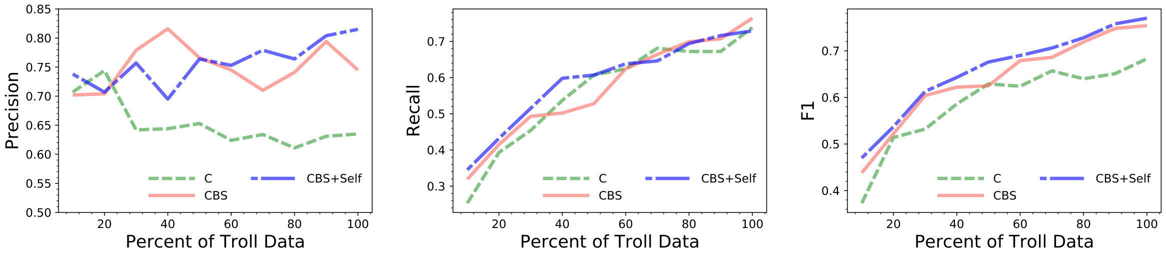
The self-supervision hyperparameters were chosen based on the validation dataset.

5 Results

In this section, we evaluate two of the major contributions of this paper: the stylometric features and self-supervision.

Stylometric Features. In Table 2, we compare our model (CBS+Self) trained using the entire troll dataset. We compare it to (CBS), our model without self-supervision, and to simply using content (C), without stylometric features. Overall, we find that the model CBS+Self outperforms the other two baselines, with an improvement of nearly 2% over CBS and 9% over C. While not directly comparable, we find that C performs comparably to the bag-of-words model presented in Im et al. (2019). Thus, implying that the control dataset may have similar data distributions. Moreover, compared to Im et al. (2019), we do not use any profile features nor do we extract information about the language, unless a language specific token was one of the 5,000 most common words when combined with the control group. Overall, we only rely on linguistic style, simple behavior information, and general topical content to make predictions.

Feature Ablation. We perform an ablation study across the three feature groups on the validation dataset in Table 3. Specifically, we analyze the



Precision

Recall

F1

Figure 2: Precision, recall, and F1 test results are plotted using different percentages of troll data during training.

loss in precision, recall, and F1 scores by excluding a feature set from the CBS model and recording its performance. Excluded features are indicated by the minus (-) symbol. Overall, we find that removing content features results in the largest drop in performance, with a 20% drop. This is expected given content features were also the most predictive in Im et al. (2019). The next largest drop is from removing behavioral features, followed by stylistic. However, removing stylistic features results in the second largest drop in precision, while behavioral features have the second largest drop in recall.

Self-Supervision. In Figure 2, we plot the precision, recall, and F1 for the three major models using different percentages of the troll training dataset. We observe that CBS outperforms C across all percentages of troll data with regards to F1. Similarly, CBS+Self consistently results in around a 2% F1 improvement over CBS. Interestingly, precision has a near linear improvement as more trolls are used for training. Yet, recall stays relatively consistent, or for C, slightly decreases. From the plots, we can make two important conclusions. First, adding more troll data improves overall prediction, at least based on F1. It seems that because of the diversity of topics discussed across troll accounts, it is not easy to detect a significant amount of trolls. Moreover, we find that adding more troll data results in a nearly linear increase in recall. Yet, precision is erratic, resulting in neither large improvements nor decreases. Second, while CBS results in consistent improvements over C, showing the positive impact of behavioral and stylistic features, more data does not necessarily help precision. This suggests that new information must be incorporated for further improvement. We examine the false positive and false negative errors in more detail in Section 6.1.

6 Discussion and Limitations

To address two questions, "What type of errors are reduced by adding behavioral and stylistic features?" and "What errors are reduced as more data is collected?". Specifically, we perform a manual error analysis and discuss our study's limitations.

6.1 Error Analysis

In order to assess the quality of our classifier, we analyze the false positive and false negative errors made by the models. Particularly, we study error differences between C and CBS. Moreover, we analyze the different errors made by classifiers trained on different percentages of troll accounts. For error analysis, one of the authors manually analyzed the errors and grouped them into semantic categories. Specifically, we selected a total of 100 false positives and negatives if available. Otherwise, if there were fewer than 100 errors, we annotated all of them. The aim of the analysis is twofold. First, we want to provide insights into what the models are unable to learn (i.e., weaknesses). Second, we want to provide insight for future avenues of work.

6.1.1 False Positives

The false semantic groups and counts of false positive errors are displayed in Table 4a. Overall, we grouped errors into four semantic classes: Bots, Political, Unknown Character, and Misc. None of the models had more than 100 false positives in the validation dataset.

Bots. A common source of false positives appear to fall into the "bot" category. We find that the number of bot-related false positives increases from 5 to 9. Intuitively, the C model fails to distinguish the repetitive nature of the troll accounts from Bots. Example of bot accounts includes users that repeatedly share links in every tweet.

	C		CBS	
	10%	100%	10%	100%
Bot	5	9	5	5
Political	10	20	10	13
Unknown Character	4	7	3	3
Misc.	11	27	11	29
Total Error	30	63	29	50

False Positives

	C		CBS	
	10%	100%	10%	100%
Support	14	6	14	6
Discord	14	10	13	7
Political Concealment	12	10	12	9
Unknown Character	19	13	19	5
Misc.	41	23	42	20
Total Error	100	62	100	47

False Negatives

Table 5: Manual analysis of false positives and false negatives for the Content (C) and Content+Behavioral+Style (CBS) models. We also analyze errors made by models trained on different percentages of the troll dataset (10% and 100%). The error analysis is based on the validation dataset.

One "bot" user repeatedly tweets the time of day.

> Example: "*It's 5 o'clock in Auckland. It's 5 o'clock in Apia. It's 5 o'clock in Juneau. It's 5 o'clock in Seattle. It's 5 o'clock in San Rafael. It's 5 o'clock in Yanacancha...*"

For the CBS model, the number of Bot-related false positives did not increase after adding more troll-related data (i.e., from 10% to 100%). Suggesting that the stylistic and behavior feature are able to distinguish a bot from troll. Yet, a substantial group of errors are still bot-related. Therefore, we believe future work should jointly learn to classify bots and trolls.

Political. The second category of errors are labeled as "political". These tweets are not essentially leaning towards democratic or republican ideologies. Rather they are politically active users, that are criticizing various issues or posting political updates on current events. The topic of the tweets included, but were not limited to, healthcare, Medicaid, Obamacare, and war. Tweets mentioned several political figures such as Donald Trump, Barrack Obama, Ivanka Trump, Ted Cruz, and Jeb Bush. Likewise, politically active users that were misclassified as trolls also used terms such as debate, campaign, and president.

> Example: "*...The **GOP** asked her to endorse **Rubio** NBC/WSJ knows that their recent poll is a fraud. It would have been better to say **JEB** polls **Rubio** was leading the nation wide poll The **Gop** pundits keep saying...*"

We did find a few false positives were also related to sexual abuse. Overall, with the C model, the number of false positives increased after adding more troll data. For CBS, there was a slight increase in errors (10 to 13), but the increase was not as dramatic. This suggests that stylistic and behavior information can distinguish between politically active users and trolls with a political agenda.

Unknown Characters. The third category only resulted in a few errors. We labeled this group as "unknown character" which groups users that have tweets with repetitive non-English characters along with repetitive mentions, in combination with shared links. Overall, because the content does not appear in or ngrams, the false positive is called because of the user's behavior (i.e., sharing many links).

> Example: "لصفلا نع اذه ثدحتنس ،ابحرم
> طباور دجتس .سيب نو اجنام نم ٩٣٢
> درجمب لفسلأاب ةدهاشملاو ليمحتلا
> يف لصفلا دعوم دجتسو لصفلا رودص
> لاقملا اذه.
> *https://t.co/XXXXXX*"

The CBS model only had 3 unknown character-related errors. Likewise, the number of errors did not increase, or decrease, by adding more trolls to the training dataset. Overall, many of the unknown characters are not in the top 5000 unigrams. Thus, we find that many of the false postives are caused by the behavior aspects of the tweets (e.g., sharing many hyperlinks).

Misc. The final category we developed for false positives are "misc." errors. These tweets did not contain political-related topics. The focus of the tweets ranged from religion to pop culture. Likewise, sometimes, users in this group shared links for marketing purposes. We find that this is the largest group of errors, and the number of misc-related errors increases dramatically as more troll data is added (e.g,. 11 to 29 for CBS). We observed a pattern in the ground-truth troll data in which they talk about Veterans Day, then heroes, Christmas, someone's birthday, and music. They then generally post a politically-related tweet.

> Example: "*Specials 3/28/19 Sandwich: turkey, bacon, avocado aioli and greens ... Sad note, today is chef Laurette's last day ... Specials 3/29/19 Sandwich: Parmigiana chicken breast ...* ***Also contains 20+ urls***"

Many of the errors are caused by the behavior of the user (e.g., sharing a large number of links). To fix these errors in future work, adding topic pattern over time could help. Intuitively, if a user never discusses any political topics, and is not likely to tweet one, based on temporal patterns that differ from known troll farms, then we may be able to reduce this group of false positives.

6.1.2 False Negatives

In Table 4b we display the counts of false negatives that fall into one of five groups: Support, Discord, Political Concealment, Unknown Character, and Misc. Overall, for both C and CBS, and unlike false positives, we find that the number of false negatives decreases as more data is added. This pattern is also evident in Figure 2 by the nearly linear increase recall as more data is added.

Discord. The model failed to detect Russian troll tweets gave an impression of "discord"—in our work we labeled accounts that were attempting discord about certain topics, e.g. black lives matter, immigration ban on Muslims, and racial degradation/issues.

> Example: "*@EdwardNiam Namaste Cops getting away with murder. Once again #TamirRice #Justice4Tamir #BlackLivesMatters #policebrutality https://t.co/XXXXX Love my city! #Cleveland #Blackycleveland #streetart #graffiti https://t.co/XXXXXX ...* ***Also contains 10+ urls*** "

For C, the number of errors dropped from 14 to 10 by adding more data. Likewise, for CBS, the errors dropped from 13 to 7. We find that behavioral and stylistic information takes better advantage of more data, with a nearly 50% drop in discord errors. Intuitively, CBS improves by a lot because many of the discord text contain many hyperlinks which the model correlates with troll behavior. Moreover, common topics are captured by the top 5000 ngrams as more troll data is added.

Political with Concealment. We refer to next group of errors as "political with concealment". The models failed to identify trolls that posted a large number of tweets that were not related to politics, compared to the political-related tweets. Examples of non-political topics include tweets about the Kardashians and Pamela Anderson. Generally, we found the transition into a political post are quite sudden. Political concealment is a major tactic used by troll organizations to masquerade themselves as US citizens. While CBS performed slightly better with more data (12 to 9) than C (12 to 10), political concealment errors still make up a large proportion of the false negatives.

> Example: "*... I was supposed to be flying from NY to San Antonio on business, but my wife got hurt the day before and I canceled my trip. #My911Story ...****Poland bans Russian "journalist"*** *from entering Schengen zone until 2020 https://t.co/XXXXX via .... RT @EjHirschberger: This is my daughter, Elizabeth Thomas, missing since Monday, March 13th. Please help me find...*"

Support. The "support" category is similar to political false negatives. Except, most of the tweets for a user consisted of messages which that heavily support Donald Trump, but they do not directly refer to him. The tweets mentioned anti-Muslim and anti-Hillary posts.

> Example: "*We don't allow "refugees" into this country until we help our homeless first #IslamKills*"

Generally, adding more data solves this issue. This suggests that the training data is not large enough to capture all the topics discussed by Russian trolls.

Misc. The largest portion false negatives are caused by users that either did not not tweet any political issues or tweeted political issues that are not common, thus not captured by the 5000 most frequent unigrams. We labeled this category as "misc". Most of these tweets did not have any specific focus which seemed to repeat. The length of the tweets was not long. Two uncommon political subjects kept recurring are about nuclear explosions and chemicals. For example, many of these users tweeted about #FukushimaAgain or #Fukushima2015, a nuclear disaster that occurred in Japan.

> Example: "*...* ***#FukushimaAgain Ukrainians*** *say it was the new Chernobyl! They are afraid! I wanna drown my sorrow http://t.co/XXXXX ... Bitterness is like drinking poison Chernobyl's reactor is going to explode again!...*"

Compared to the misc group for false positives, we found that the misc examples for false negatives did not always contain many distinguishing behavior or stylistic characteristics. Therefore, a large number of false negatives are produced by both the C (23 false negatives) and CBS (20 false negatives) models.

Unknown Character. Finally, we also have a category called "unknown character" for false negatives. Often those were related to non-English characters that are not commonly occurring within the continental US. Examples include Unicode characters from the Russian alphabet.

> Example: "Ковер на стене и бесконечные тосты. Что удивляет испанку в русских: *https://t.co/XXXXX*"

We find that most of these errors are handled by adding more troll data. For instance, CBS errors were reduced from 19 to 5 by increasing the troll data from 10% to 100%. We find that the behavior and stylistic features are important to handle the unknown character error type.

6.1.3 Error Analysis Discussion

Overall, we believe temporal patterns of topics could further reduce false negatives. For example, if we analyze a user's tweets over time, we may find that they repeatedly discuss the following topics in temporally: 1. pop culture 2. birthday wish 3. political 4. pop culture. Thus, temporal-topic patterns can be used as auxiliary features. If we use neural networks, the patterns can be used by a recurrent neural network. The topics can be learned automatically using topic modeling.

6.2 Limitations

There are two limitations to this study. First, the control dataset is not guaranteed to be troll-free. While we did not find any obvious trolls in our error analysis of false positives, this does not stop them from being part of the training, test, or unlabeled datasets. This can result in sub-optimal performance, either by incorrectly reported test results, or because of noisy training data. Second, the training dataset consisted of Twitter accounts that have selected English as their primary language. Thus, given the limitations, future work should provide more varied datasets. Specifically, data should be collected carefully to avoid contamination. Also, larger collections of bots and politically active users should be added to the dataset to increase the difficulty of the task. Furthermore, normal users that discuss non-political topics similar to the topics discussed by the trolls should be targeted to include in a new dataset. Finally, while we found that stylistic and behavior information can improve classification performance, sometimes this information resulted in more false positives (e.g., Misc false positives).

7 Conclusion

Social media platforms are likely to play a more important role in political discourse for both democratic and authoritative nations, as evidenced by recent world events. Hence, it is important that we develop approaches to identify malicious actors seeking to influence the outcomes or decision making of various stakeholders by manipulating social media platforms. Therefore, in this paper we presented a novel troll detection method, based on state-of-the-art stylometric and behavioral information. Moreover, because it is challenging to collect real troll accounts, we analyzed the use of self-supervision to automatically annotate unlabeled collections of data. Specifically, we showed that self-supervision improves detection performance with as few as 100 training users and with nearly 1,000 annotated trolls. Finally, we performed a detailed error analysis that provides insight for future model development. Future research includes, but is not limited to, new dataset development, detecting both bots and trolls, expanding the stylistic/behavioral features, and introducing temporal topic patterns as features.

Also, it is important to study the ethical implications of this technology, such as asking the question, "How could false positives, or false negatives, adversely impact real people?" Moreover, should black box models be used by government agencies, or social media companies, to monitor Russian troll activity? It is important to understand each of these questions before putting this work into production.

References

Ahmed Abbasi and Hsinchun Chen. 2008. Writeprints: A stylometric approach to identity-level identification and similarity detection in cyberspace. *ACM Transactions on Information Systems (TOIS)*, 26(2):7.

Monica Anderson, Skye Toor, Lee Rainie, and Aaron Smith. 2018. Activism in the social media age.

Washington, DC: Pew Internet & American Life Project. Retrieved July, 11:2018.

Ahmer Arif, Leo Graiden Stewart, and Kate Starbird. 2018. Acting the part: Examining information operations within# blacklivesmatter discourse. *Proceedings of the ACM on Human-Computer Interaction*, 2(CSCW):20.

Adam Badawy, Emilio Ferrara, and Kristina Lerman. 2018. Analyzing the digital traces of political manipulation: the 2016 russian interference twitter campaign. In *2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, pages 258–265. IEEE.

SG BIRD and E LOPER. 2004. Nltk: The natural language toolkit. Association for Computational Linguistics.

David A Broniatowski, Amelia M Jamison, SiHua Qi, Lulwah AlKulaib, Tao Chen, Adrian Benton, Sandra C Quinn, and Mark Dredze. 2018. Weaponized health communication: Twitter bots and russian trolls amplify the vaccine debate. *American journal of public health*, 108(10):1378–1384.

Dorothy Elizabeth Robling Denning. 1999. *Information warfare and security*, volume 4. Addison-Wesley Reading, MA.

Ivan Habernal and Iryna Gurevych. 2015. Exploiting debate portals for semi-supervised argumentation mining in user-generated web discourse. In *Proceedings of the 2015 conference on empirical methods in natural language processing*, pages 2127–2137.

Jane Im, Eshwar Chandrasekharan, Jackson Sargent, Paige Lighthammer, Taylor Denby, Ankit Bhargava, Libby Hemphill, David Jurgens, and Eric Gilbert. 2019. Still out there: Modeling and identifying russian troll accounts on twitter. *arXiv preprint arXiv:1901.11162*.

Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. 2016. Bag of tricks for efficient text classification. *arXiv preprint arXiv:1607.01759*.

Vaibhavi Kalgutkar, Ratinder Kaur, Hugo Gonzalez, Natalia Stakhanova, and Alina Matyukhina. 2019. Code authorship attribution: Methods and challenges. *ACM Computing Surveys*, 52(1):3:1–3:36.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751.

Yanchao Li, Yong li Wang, Dong-Jun Yu, Ye Ning, Peng Hu, and Ruxin Zhao. 2019. Ascent: Active supervision for semi-supervised learning. *IEEE Transactions on Knowledge and Data Engineering*.

Clare Llewellyn, Laura Cram, Adrian Favero, and Robin L Hill. 2018. Russian troll hunting in a brexit twitter archive. In *Proceedings of the 18th ACM/IEEE on Joint Conference on Digital Libraries*, pages 361–362. ACM.

Neil Mallinar, Abhishek Shah, Rajendra Ugrani, Ayush Gupta, Manikandan Gurusankar, Tin Kam Ho, Q. Vera Liao, Yunfeng Zhang, Rachel K. E. Bellamy, Robert Yates, Chris Desmarais, and Blake McGregor. 2019. Bootstrapping conversational agents with weak supervision. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019.*, pages 9528–9533.

Panagiotis T Metaxas and Eni Mustafaraj. 2012. Social media and the elections. *Science*, 338(6106):472–473.

RS Mueller. 2019. Report on the investigation into russian interference in the 2016 presidential election. *US Department of Justice*.

Christopher Paul and Miriam Matthews. 2016. The russian “firehose of falsehood” propaganda model. *Rand Corporation*, pages 2–7.

Jian Peng, Sam Detchon, Kim-Kwang Raymond Choo, and Helen Ashman. 2017. Astroturfing detection in social media: a binary n-gram-based approach. *Concurrency and Computation: Practice and Experience*, 29(17).

Pavithra Rajendran, Danushka Bollegala, and Simon Parsons. 2016. Contextual stance classification of opinions: A step towards enthymeme reconstruction in online reviews. In *Proceedings of the Third Workshop on Argument Mining (ArgMining2016)*, pages 31–39.

Pavithra Rajendran, Danushka Bollegala, and Simon Parsons. 2018. Is something better than nothing? automatically predicting stance-based arguments using deep learning and small labelled dataset. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 28–34.

Takeshi Sakaki, Makoto Okazaki, and Yutaka Matsuo. 2010. Earthquake shakes twitter users: real-time event detection by social sensors. In *Proceedings of the 19th international conference on World wide web*, pages 851–860. ACM.

Yunita Sari, Mark Stevenson, and Andreas Vlachos. 2018. Topic or style? exploring the most useful features for authorship attribution. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 343–353.

Roy Schwartz, Jesse Dodge, Noah A Smith, and Oren Etzioni. 2019. Green ai. *arXiv preprint arXiv:1907.10597*.

Efstathios Stamatatos. 2009. A survey of modern authorship attribution methods. *Journal of the American Society for information Science and Technology*, 60(3):538–556.

Leo G Stewart, Ahmer Arif, and Kate Starbird. 2018. Examining trolls and polarization with a retweet network. In *Proc. ACM WSDM, Workshop on Misinformation and Misbehavior Mining on the Web*.

Stefan Stieglitz and Linh Dang-Xuan. 2013. Emotions and information diffusion in social media—sentiment of microblogs and sharing behavior. *Journal of management information systems*, 29(4):217–248.

Emma Strubell, Ananya Ganesh, and Andrew McCallum. 2019. Energy and policy considerations for deep learning in nlp. *arXiv preprint arXiv:1906.02243*.

Meng Wang, Xian-Sheng Hua, Tao Mei, Richang Hong, Guojun Qi, Yan Song, and Li-Rong Dai. 2009. Semi-supervised kernel density estimation for video annotation. *Computer Vision and Image Understanding*, 113(3):384–396.

Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. 2016. Hierarchical attention networks for document classification. In *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: human language technologies*, pages 1480–1489.

Jie Yin, Sarvnaz Karimi, Andrew Lampert, Mark Cameron, Bella Robinson, and Robert Power. 2015. Using social media to enhance emergency situation awareness. In *Twenty-fourth international joint conference on artificial intelligence*.

Savvas Zannettou, Tristan Caulfield, William Setzer, Michael Sirivianos, Gianluca Stringhini, and Jeremy Blackburn. 2019. Who let the trolls out?: Towards understanding state-sponsored trolls. In *Proceedings of the 10th ACM Conference on Web Science*, pages 353–362. ACM.

Yubao Zhang, Xin Ruan, Haining Wang, Hui Wang, and Su He. 2016. Twitter trends manipulation: a first look inside the security of twitter trending. *IEEE Transactions on Information Forensics and Security*, 12(1):144–156.