An Exploratory Analysis of the Behavior and Communication Patterns of Interfering Agents

Nicholas E. Vogt

University of Michigan - Dearborn

Submitted to Professor Luis E. Ortiz

in partial fulfillment of the requirements

of the Winter 2018 Term

of CIS 579: Artificial Intelligence

Abstract

Russian interference in the 2016 United States presidential election has proven to be be a contentious public issue in recent years. The Office of the Director of National Intelligence believes social media facilitated the efforts of these interfering agents. Recently, Twitter deleted over 200,000 tweets associated with accounts "tied to 'malicious activity' from Russia-linked accounts." In this report, we investigate the behavior of these interfering agents by conceptualizing the Twitter social network as nodes of users connected by mentions. We apply community detection to show that interfering agents interact with common users, but rarely interact with each other. Secondarily, we explore latent Dirichlet allocation as a method to understand how topics are related between different communities.

Introduction

Foreign interference in the 2016 United States Presidential Election has come to emblemify a trend of mistrust in democratic processes. Since confirmation of foul play by the Office of the Director of National Intelligence [1], public and private entities are cooperating with US investigators to uncover the scope of Kremlin-supported efforts. In October 2017, Twitter joined these efforts by submitting the names of 3,814 "potentially Russian [state]-linked accounts" [2]. The tweets linked with these accounts have been removed from public view as of April 2018; however, Ben Popken of NBC News worked with anonymous sources to publish the identified users' and their tweets for the public domain [3].

The Popken dataset provides a unique view on how agents behave with the unified goal of interfering in a democratic election and subsequent government policy. This report applies state-of-the-art methods in community detection and their more computationally

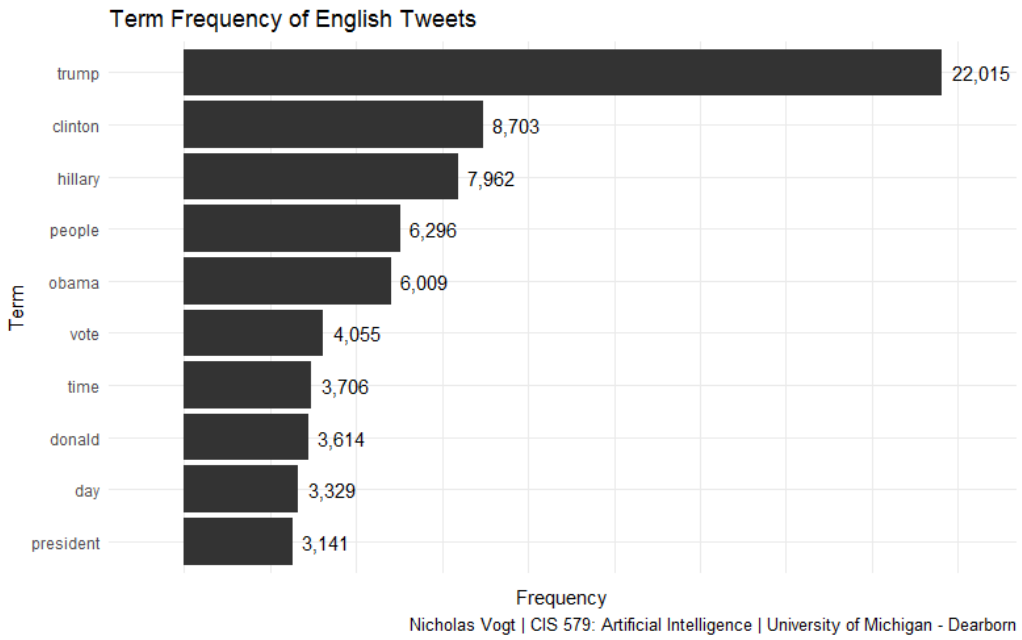
efficient advancements. We further explore elementary topic modeling on of corpus of about 150,000 documents as a curiosity to explore how different communities speak.

Data

The data for this study is credited to Ben Popken of *NBC News* [3]. The Popken dataset includes a table, *Tweets*, of user-tagged, time-stamped tweets and a second table, *Users*, of accounts identified by Twitter as possible interfering agents. We expand and refine the Popken dataset in two meaningful ways:

- The Popken corpus includes tweets in several languages, including English, German, and Mandarin. Non-English tweets are identified using the R textcat package and removed from the corpus. 143,455 tweets were classified as an English dialect.
- Some tweets mention accounts which are not included in *Users*. We identify these accounts and append them to the *Users* data. The expanded dataset includes 14,273 users.

We begin our investigation of interfering agents communications by investigating common terms with the hope that some distinct topics could be identified by the most frequently occurring terms. Interestingly, five of the top 10 terms directly reference US politicians, TRUMP, CLINTON, HILLARY, OBAMA, and DONALD in descending order. TRUMP is far and away the most frequent term in the corpus with 22,015 occurrences; the second most common term is CLINTON with 8,703 occurrences.



Methodology

Community detection in networks has the expressed goal of finding classes of nodes which are most similar to others in its class. How one best defines their similarity metric, or *centrality measure*, is a well-researched topic [4]. We will examine algorithms utilizing two centrality measures: edge betweenness and spectral partitioning.

M. Girvan and M. E. J. Newman[1] coined the Girvan-Newman algorithm which uses *edge betweenness* as it's centrality measure [12-4]. For undirected, unweighted networks, edge betweenness is defined as

$$g(x) = \sum_{a \neq b \neq x} \frac{\sigma_{ab}(x)}{\sigma_{ab}}$$

where σ_{ab} is the total number of shortest paths from node *a* to node *b* and $\sigma_{ab}(x)$ is the number that pass through node *x*. The Girvan-Newman algorithm identifies communities of nodes "in

[1] M. E. J. Newman is a physics professor at University of Michigan. Close to home!

tightly-knit groups between which there are only looser connections" [4]. More specifically, it identifies nodes with the highest edge betweenness and labels communities on either side of that node. Girvan-Newman has the immeasurable benefit of being easily explained to laypeople. It's also very applicable to social networks of "tightly-knit" communities.

An alternative to the edge betweenness centrality measure is *spectral partitioning* employed by Newman's leading eigenvector method [5]. Spectral partitioning is a complex method requiring more background in linear algebra than the scope of this report allows; however, its core purpose is to identifies communities based on the most connected nodes.

Both methods have the regrettable quality of intractability on larger networks. Spectral partitioning requires several expensive matrix computations (many running in polynomial time), and Girvan-Newman algorithm computes the shortest path between every node in the network. While Newman defined a method for computing edge betweenness in $O(mn)$ time for m edges and n vertices [7-6], the algorithm is not suitable on large networks and ill-equipped machines. Of course, social networks can be very large. The Popken dataset includes only 14,273 distinct users and less than 143,455 edges,[2] but in the interest of education, we seek methods refined for very large social networks.

Wakita's and Tsurumi's algorithm is optimized to evaluate faster than most other methods and is well suited to large networks [7]. Like Girvan-Newman, it evaluates on the metric of node betweenness. Its computational gains are achieved by choosing more optimal data structures (which, again, are beyond the scope of this report).

[2] Only 392 accounts are given by Popken. There are an additional 13,881 distinct, non-suspect agents mentioned in the tweets posted by the original 392.

We now turn our attention to topic modeling with latent Dirichlet allocation (LDA). LDA is a generative topic model conceived by Blei et. al in 2003 [8] to identify similarities between unlabeled documents. In the authors' words, "documents are represented as random mixtures over latent topics, where each topic is characterized by a distribution over words." Topic modeling is a large and complex field of natural language processing. We chose to implement LDA, or all choices, for its explanatory power, breadth of use cases, and conceptually simple premise.

We implement latent Dirichlet allocation (LDA)

$$p(T|\alpha,\beta) = \prod_{t=1}^{M} \int p(\theta_t|\alpha) \left(\prod_{n=1}^{N_t} \sum_{z_{tn}} p(z_{tn}|\theta_t) p(w_{tn}|z_{tn},\beta) \right) d\theta_t$$

On a corpus of English tweets *T*. We evaluate the model using the metric of *perplexity*, also defined by Blei et al [8] as

$$perplexity(T) = exp\left\{ -\frac{\sum_{t=1}^{M} logp(\mathbf{w}_t)}{\sum_{t=1}^{M} N_t} \right\}$$

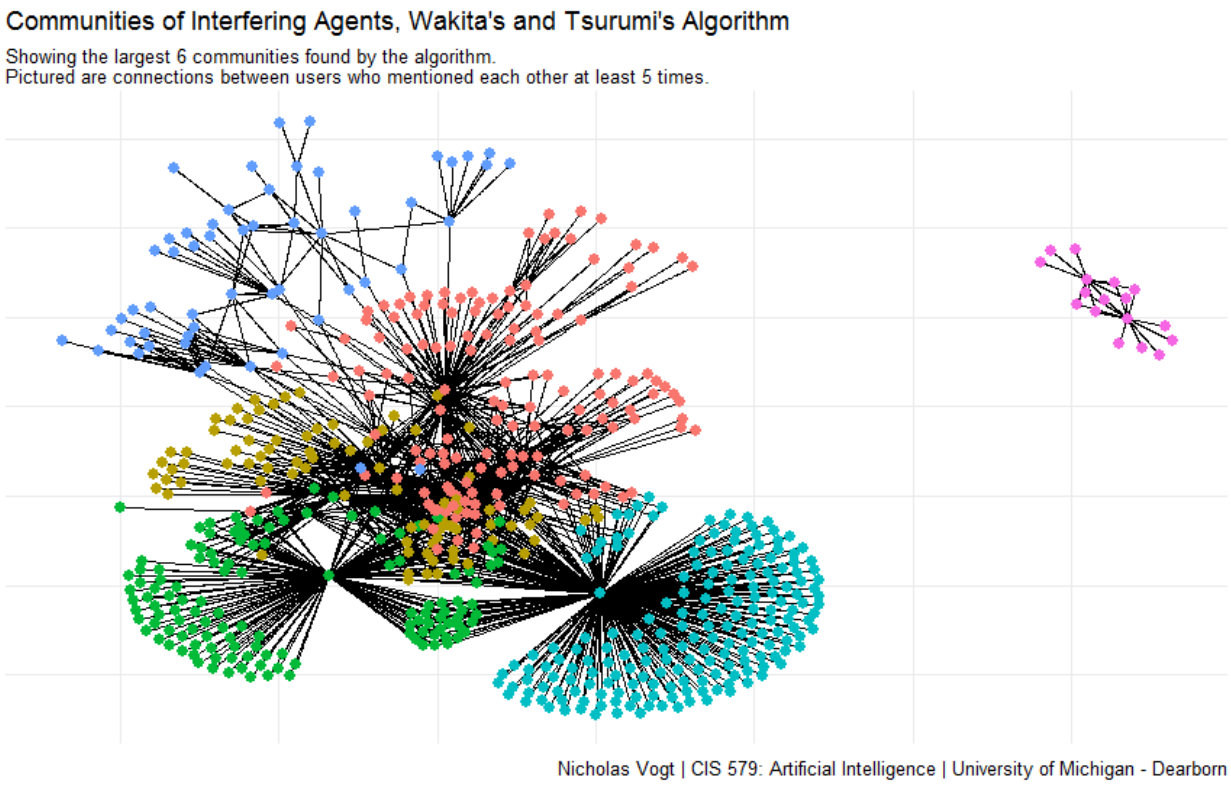
where $\boldsymbol{w}_t$ is the vector of words in tweet *t*, *M* is the number of tweets used in training, and N_t is the number of words in tweet *t*. Per the authors, a lower test perplexity signals a more fitted model.

We held out 20% of tweets from each cluster for testing perplexity and trained the LDA model on the remaining 80%. The authors describe an *elbow method* for choosing the correct number of topics (*k*); we choose *k* where the marginal perplexity with respect to *k* is sufficiently small. With the proper number of topics in mind, we intuit common topics between clusters by examining common keywords to each topic.

Results

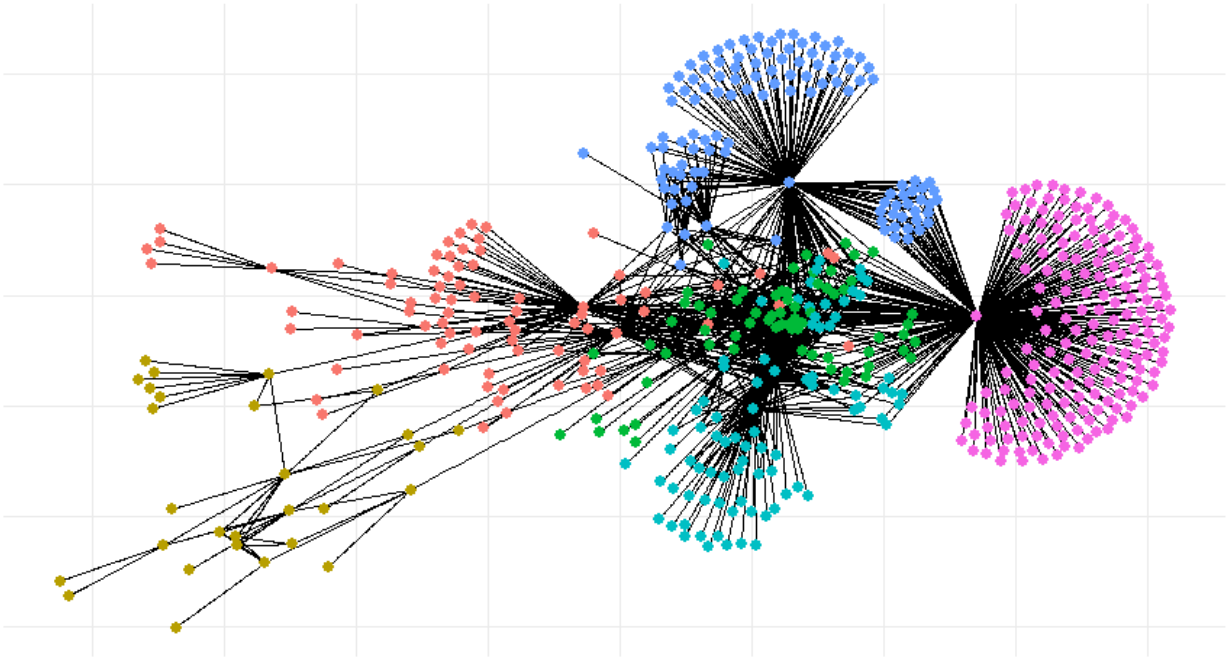
We apply three community detection algorithms: Wakita's and Tsurumi's algorithm, Girvan-Newman algorithm, and Newman's leading eigenvector algorithm. Each yields similar communities. In particular, we observe that the most connected nodes also have the highest edge betweenness measure. These central nodes tend to be politicians and celebrities (such as @RealDonaldTrump).

Surprisingly, interfering agents appear to interact with common users, but rarely interact with each other.



Communities of Interfering Agents, Girvan-Newman Algorithm

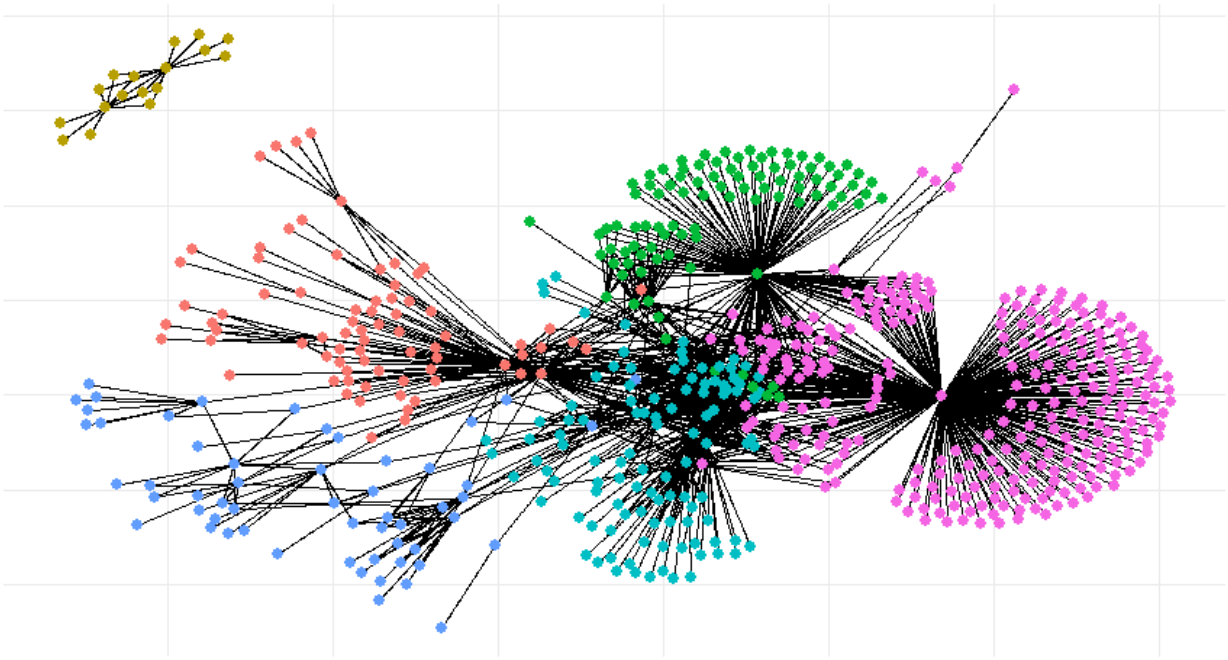
Showing the largest 6 communities found by the algorithm.
Pictured are connections between users who mentioned each other at least 5 times.



Nicholas Vogt | CIS 579: Artificial Intelligence | University of Michigan - Dearborn

Communities of Interfering Agents, Newman's Leading Eigenvector Algorithm

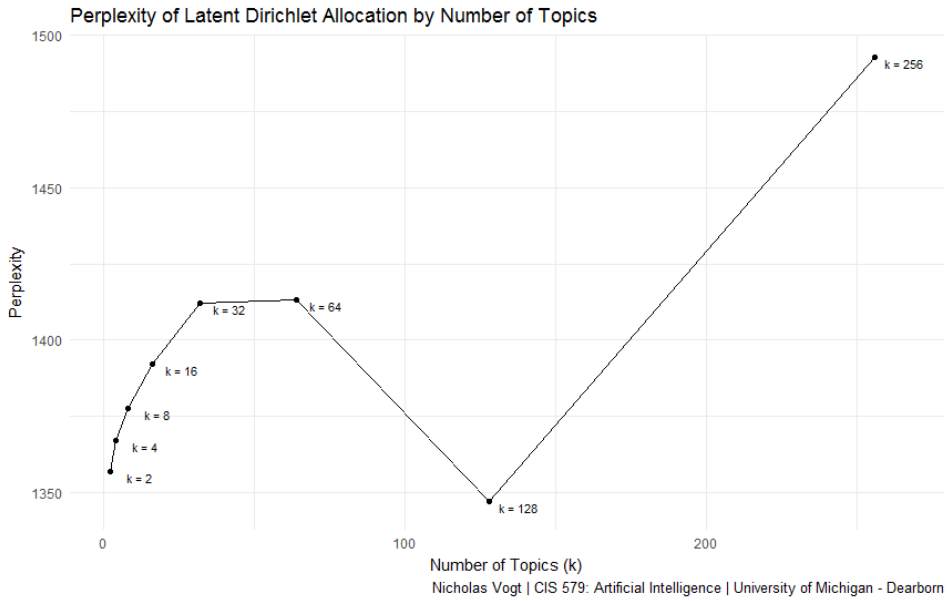
Showing the largest 6 communities found by the algorithm.
Pictured are connections between users who mentioned each other at least 5 times.



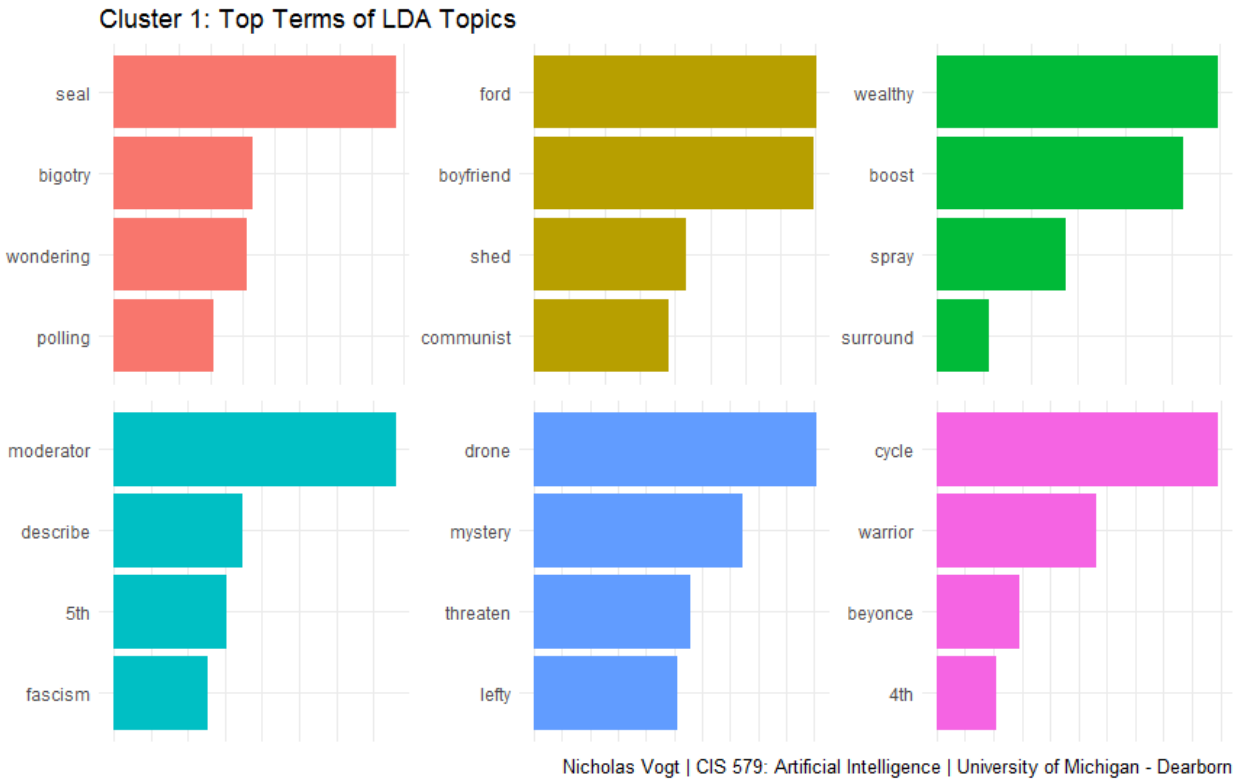
Nicholas Vogt | CIS 579: Artificial Intelligence | University of Michigan - Dearborn

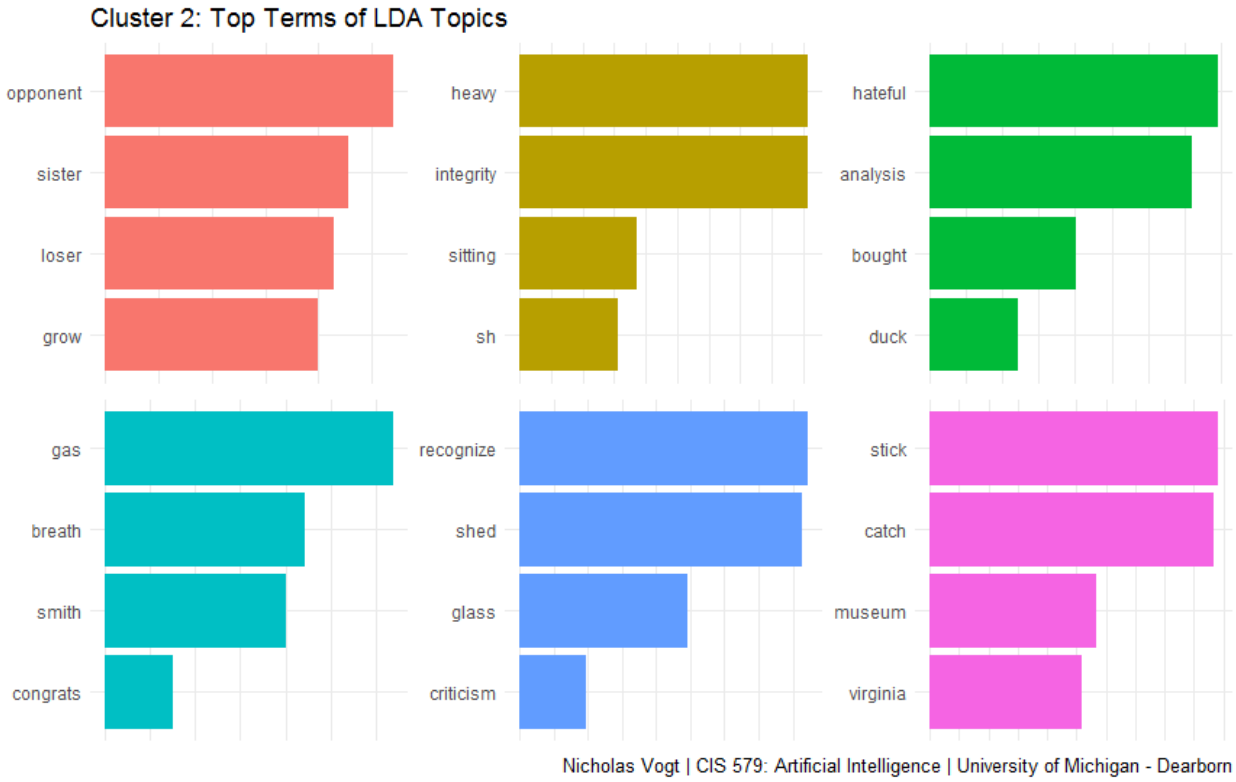
LDA proves to be sensitive to noise in the data. To reduce variance in the data and to focus on interesting words, we remove words which occurred fewer than 50 times in the English subset of tweets. We also remove tweets which consisted exclusively of those words. The resulting corpus consists of 143,455 English tweets and 2,825 distinct terms.

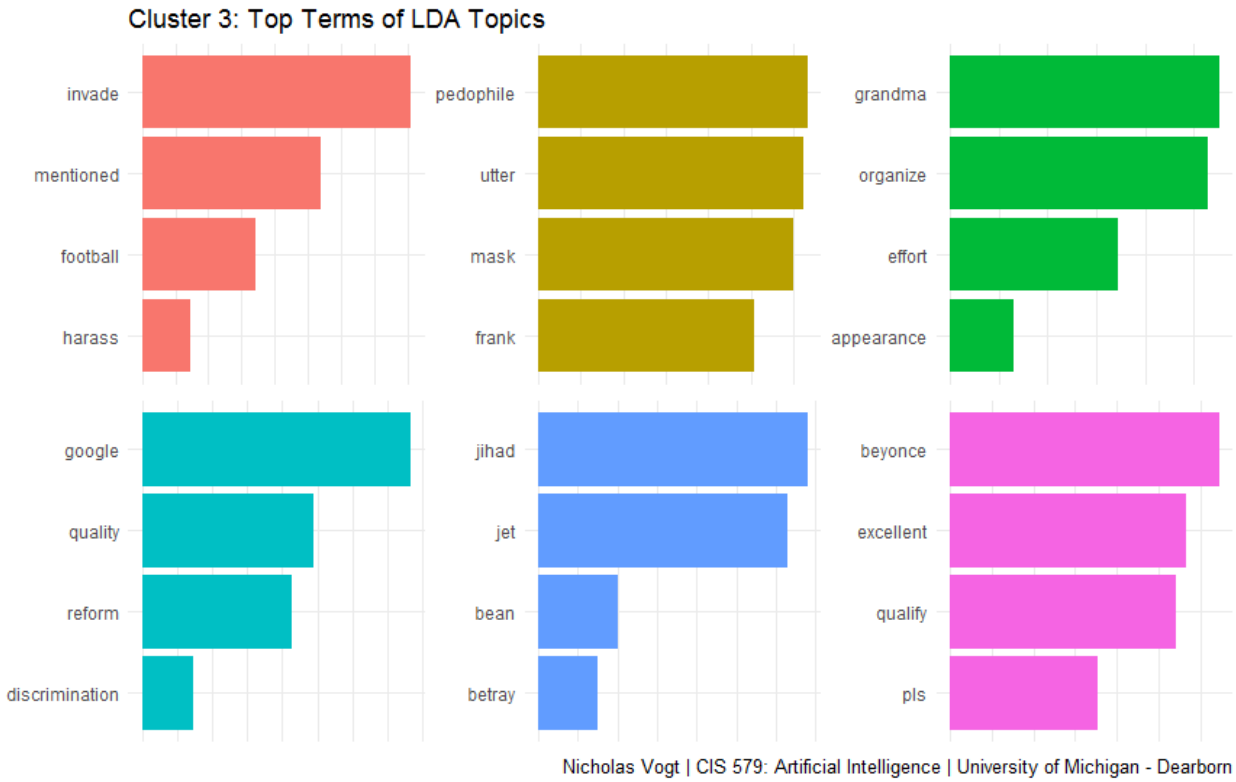
To determine how many topics to model, we create eight models with 2^k topics for the k^{th} model using a training sample of 80% of the data. With the remaining 20% of observations, we identify the test perplexity of each model. The model with the minimum test perplexity has k = 128 topics and forms a local minimum compared the models with k = 64 and k = 256 topics.[3] We choose to continue with 128 topics, but are limited by local computing resources.[4] For ease of interpretation and comparison between communities, as is the goal of this report, we continue with six topics.

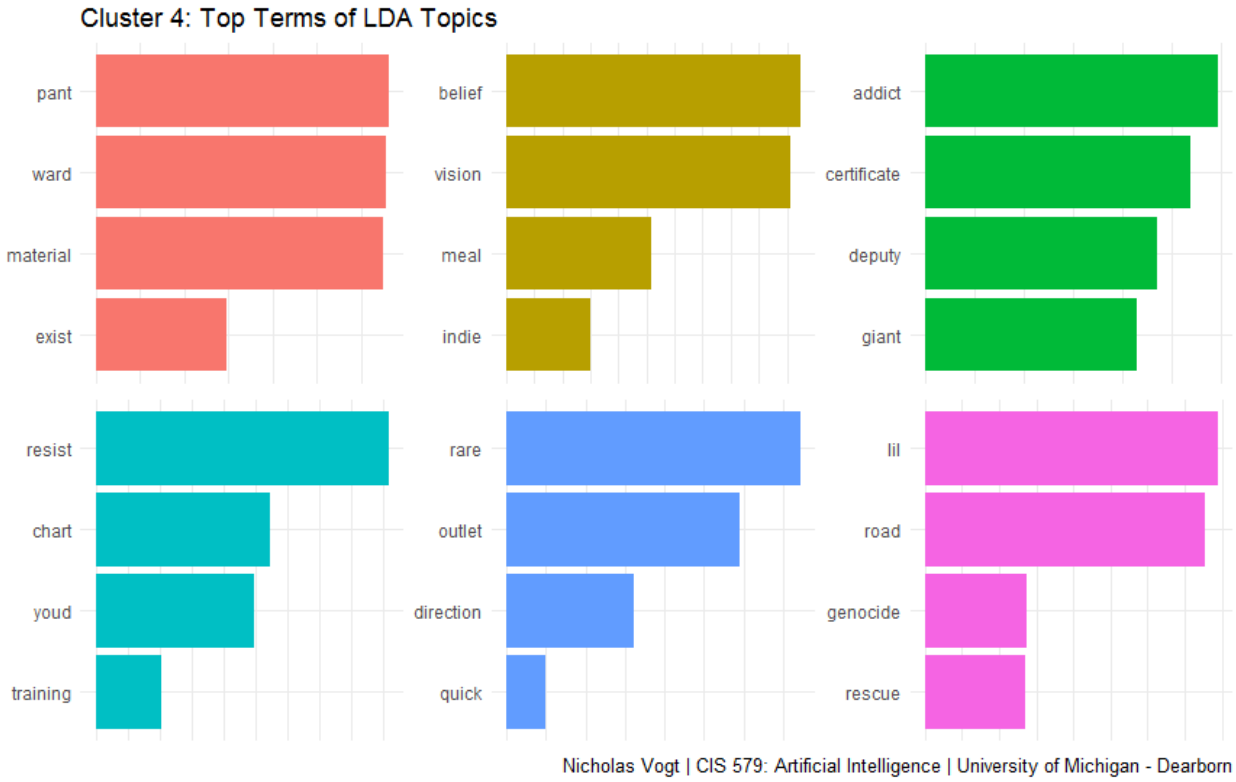


[3] Our findings contradict those of Blei et al. [8] state that perplexity is monotonically decreasing as the number of topics k increases.
[4] Computations exceeded our machine's 8 GB RAM.


Cluster 1: Top Terms of LDA Topics
seal
bigotry
wondering
polling
ford
boyfriend
shed
communist
wealthy
boost
spray
surround
moderator
describe
5th
fascism
drone
mystery
threaten
lefty
cycle
warrior
beyonce
4th
Nicholas Vogt | CIS 579: Artificial Intelligence | University of Michigan - Dearborn


Cluster 2: Top Terms of LDA Topics
opponent
sister
loser
grow
heavy
integrity
sitting
sh
hateful
analysis
bought
duck
gas
breath
smith
congrats
recognize
shed
glass
criticism
stick
catch
museum
virginia
Nicholas Vogt | CIS 579: Artificial Intelligence | University of Michigan - Dearborn


Cluster 3: Top Terms of LDA Topics
invade
mentioned
football
harass
pedophile
utter
mask
frank
grandma
organize
effort
appearance
google
quality
reform
discrimination
jihad
jet
bean
betray
beyonce
excellent
qualify
pls
Nicholas Vogt | CIS 579: Artificial Intelligence | University of Michigan - Dearborn


Cluster 4: Top Terms of LDA Topics
pant
ward
material
exist
belief
vision
meal
indie
addict
certificate
deputy
giant
resist
chart
youd
training
rare
outlet
direction
quick
lil
road
genocide
rescue
Nicholas Vogt | CIS 579: Artificial Intelligence | University of Michigan - Dearborn

We identify the top four terms contributing to each topic found by our LDA model. Each topic is plotted with its top four terms measured by its contribution to the topic. Color has no meaning other than to denote different topics and to aid readability.

Conclusions

Community detection yields an unexpected result: interfering agents have common interactions with popular users, but rarely interact with each other. This is remarkably ordinary behavior; we expect popular users like politicians and celebrities to have interactions from several different communities. If an interfering agent's goal is not to be identified as such, then this behavior seems to be an optimal strategy. That said, interfering agents also have the expressed goal of changing voting behavior of American users.[5]

The results of the three community detection algorithms are very similar. Girvan-Newman and Wakita's and Tsurumi's algorithms both use edge betweenness as their primary metric, whereas Newman's leading eigenvector focuses on finding the most central, connected node. In our case, the largest communities center around a single, highly connected node whose neighbors share no other common neighbors between them. The most central node then becomes the node with the highest betweenness, thus explaining the similar results.

The results of LDA were disappointing. We believe three factors contributed to the poor performance of LDA: First, our inexperience with common preprocessing tools and techniques in natural language processing, and second, a relatively small corpus and short documents minimize which words can be identified together by the model.

[5] From a philosophical point of view, this finding lends credence to the power of a single person's actions in a global, internet community.

The data used in the model had numerous glaring errors, and we struggled to implement satisfactory solutions to solve them. These problems include identifying and correcting misspellings (e.g. RALLY from RALY), abbreviations (e.g. COALITION from COAL), concatenations (e.g. HILLARY THE from HILLARYTHE), and acronyms (e.g. BLACKLIVESMATTER from BLM). Twitter users, in particular, may purposefully use space-saving, irregular patterns to communicate fully under a constraining character limit. The difficulty of these problems do not stem from identifying candidate replacements; RALY clearly corrects to RALLY. Rather, the difficulty lies in identifying suitable replacements. One would expect COAL and COALITION fall under separate topics.

One plausible solution is to correctly identify aliases for the most common terms and phrases. MAGA, MAKE AMERICA GREAT AGAIN, and #MAGAMAGAMAGA all reference Donald Trump's campaign slogan with little ambiguity. It may be sufficient to identify terms and phrases that contribute most strongly to different topics.

It's entirely plausible that LDA was the wrong model for the data. LDA relies on a rich corpus to infer topics, and, on its face, Twitter does not lend itself to that. Tweets are short; the median tweet had 73 characters after removing hashtags and mentions. With a corpus of 143,455 tweets, each with few distinct, meaningful words, we suspect the dataset had insufficient statistical power to identify a meaningful relationship with LDA. To this point, Zhao et al. found promising results applying LDA to Twitter data with over one million tweets [9], a corpus almost five time as large. Unfortunately, we have no knowledge of a method to test the power of LDA. To circumvent the possible issue of lower power, we could leverage *chained tweets*; that is, multiple tweets posted together as a single message to

surpass the 140 character limit. This would increase document density and may improve the model.

Notwithstanding, we suspect LDA is a valid approach toward understanding the communication patterns of interfering agents on the Popken dataset. We believe the crux of our non-result is that we did not have the acumen in NLP to properly leverage the method. This project was our first foray into NLP and many common transforms and preprocessing required more time and energy than anticipated. Coupled with the unique challenges of Twitter data, a significant result requires more resources than we have available for the report.

On the whole, the project successfully identified some behavior of interfering agents with a limited view of the network, and explored the potential of topic modeling on the corpus of tweets. Next steps are to include variables (hashtags, user bios and locales, and post time) and original tweets from other users in the analysis.

References

1. Office of the Director of Nat'l Intelligence, Background to “Assessing Russian Activities and Intentions in Recent US Elections”: The Analytic Process and Cyber Incident Attribution , 6 Jan. 2017; www.dni.gov/files/documents/ICA_2017_01.pdf .
2. United States Senate Committee on the Judiciary, Subcommittee on Crime and Terrorism. Testimony of Sean J. Edgett Acting General Counsel, Twitter, Inc. https://www.judiciary.senate.gov/download/10-31-17-edgett-testimony
3. Popken, Ben. "Twitter Deleted Russian Troll Tweets. So We Published More than 200,000 of Them." NBC News. February 14, 2018. https://www.nbcnews.com/tech/social-media/now-available-more-200-000-deleted-russian-troll-tweets-n844731.
4. Girvan, M., and M. E. J. Newman. 2002. “Community Structure in Social and Biological Networks.” Proceedings of the National Academy of Sciences of the United States of America 99 (12): 7821–26.
5. Newman, M. E. J. "Finding community structure using the eigenvectors of matrices." Physical Review E 74 (2006). doi:10.1103/PhysRevE.74.036104
6. Newman, M. E. J. "Scientific Collaboration Networks. II. Shortest Paths, Weighted Networks, and Centrality." Physical Review E 64 (2006). doi: 10.1103/PhysRevE.64.016132
7. Wakita, Ken, and Toshiyuki Tsurumi. “Finding Community Structure in Mega-Scale Social Networks.” Proceedings of the 16th International Conference on World Wide Web - WWW 07, 2007, doi:10.1145/1242572.1242805.

8. Blei, David M., Andrew Y. Ng, and Michael I. Jordan. 2003. “Latent Dirichlet Allocation.” Journal of Machine Learning Research: JMLR 3 (January): 993–1022.
9. Zhao, Wayne Xin, Jing Jiang, Jianshu Weng, Jing He, Ee-Peng Lim, Hongfei Yan, and Xiaoming Li. "Comparing Twitter and Traditional Media Using Topic Models." Lecture Notes in Computer Science Advances in Information Retrieval, 2011, 338-49. doi:10.1007/978-3-642-20161-5_34.

Source Code & Packages

Csardi, Gabor and Tamas Nepusz. "The igraph software package for complex network research." InterJournal Complex Systems, 1695 (2006). http://graph.org

Grün, Bettina, and Kurt Hornik. "Topicmodels: An R Package for Fitting Topic Models." Journal of Statistical Software 40, no. 13 (2011). doi:10.18637/jss.v040.i13.

Hornik, Kurt, Patrick Mairl Johannes Rauch, Wilhelm Geiger, Christian Buchta, and Ingo Feinerer. "The textcat Package for n-Gram Based Text Categorization in R." Journal of Statistical Software 52, no. 6, 1-17. doi: 10.18637/jss.v052.i06.

Silge, Julia and David Robinson. "Tidytext: Text Mining and Analysis Using Tidy Data." JOSS 1, no. 3 (2016). doi:10.21105/joss.00037.

Wickham, Hadley. "Tidyverse." Github. https://github.com/tidyverse/tidyverse.

Vogt, Nicholas. “Russian Troll Tweets.” GitHub. https://github.com/vogt4nick/russian-troll-tweets.