Disinformation and the War in Ukraine: A Sentiment Analysis of Tweets

Michael Grodecki

Image borrowed from: https://www.cnn.com/2020/07/30/tech/2020-election-russia-disinformation/index.html

Machine Learning
CSCI 5622, Fall 2022
University of Colorado Boulder
Professor Ami Gates

INDEX

1. Introduction

1.1 *Background*

The full-scale land invasion of Ukraine initiated by Russian president Vladimir Putin on February 24th 2022 is one of the biggest threats to international security and peace, reigniting state-on-state conflict, something that has not taken place on the European continent since the end of the Second World War. This bloody and devastating war has uprooted the lives of millions, creating an urgent humanitarian crisis with over 7 million people forced to flee their homes into neighboring countries under the constant barrage of Russian bombs and missiles. With no clear end in sight, this conflict has resulted in unprecedented death and destruction across the nation of Ukraine with far-reaching effects on a global level. An international energy crisis, an arms race to defend Ukrainian territory, a global food shortage, and the growing threat from the potential use of nuclear weapons are just some of the far-reaching effects of this war. Near-universal condemnation by the international community of this unjust and unnecessary conflict have turned Russia into a global outcast, with sanctions crippling the Russian economy and impacting the lives of Russian citizens. The result of this war will be critical for the survival of Putin's regime, and possibly for the survival of the Russian state itself. Facing mounting losses, an ill-equipped army, and difficulties in producing new weaponry, the prospect of an all-out Russian defeat is looking increasingly likely. What's at stake in this conflict is incredibly high for the authoritarian regime of the Russian state, and the decimation of Ukrainian allies is a key priority. In the past, the Russian propaganda machine has attempted to sway public opinion abroad through the dissemination of government-sponsored disinformation across social media platforms and there is evidence that the Russian state is spreading harmful propaganda about this conflict. Precautionary measures have been taken to lessen the impact of bot farms and other machine-driven algorithms that have inundated social media sites with harmful posts and comment threads in the past but these methods have not been completely effective. Disinformation continues to spread and the results can be incredibly far reaching.

1.2 *State-sponsored Disinformation and Social Media*

Russia began to use platforms such as Twitter, YouTube, and Facebook to spread disinformation around a decade ago as social media continued to gain prominence. Russia's Internet Research Agency is a government institute that directed the spread of disinformation and propaganda across social media during the Crimean conflict in 2014 and the 2016 US presidential elections. Millions of people were exposed to false and misleading information masquerading as legitimate news and the full impact this had on

influencing public opinion is still largely unknown. Since then, significant measures have been taken to tackle Russian web brigades and bots, although the presence of disinformation still remains on social media platforms such as Twitter.

Image borrowed from: https://www.prweek.com/article/1756368/ukraine-war-changed-pr-landscape

1.3 *Disinformation on Twitter*

The events that are taking place in Ukraine have been closely monitored by the global community with the Russian invasion making news headlines on a daily basis, and apart from making the news, the War in Ukraine is a popular discussion topic among users of social media who continue to share their thoughts about this bloody and devastating conflict. Twitter, in particular, being one of the largest social media forums with 450 million active users, contains a massive amount of information related to the ongoing conflict in Ukraine. In the past, Twitter has been scrutinized for allowing disinformation to proliferate on its platform, with its biased recommendation algorithms leading users down an echo chamber of false and misleading information. This effect of disinformation on social media platforms like Twitter have had far-reaching consequences in the past, and it is of critical importance to identify sources of disinformation being spread on the platform about the bloody and devastating conflict in Ukraine.

The Russian propaganda machine continues to spread false information aimed at confusing and weakening its targets. Politically-aimed disinformation is a source of chaos and confusion that cannot be ignored in a world that increasingly relies on social media as a means of communication, especially given the fact that this disinformation has far-reaching social and political implications. The reverberating effects of public disinformation can have dangerous consequences. An analysis of tweets related to the conflict in Ukraine using machine learning methods can be a powerful tool to identify potential sources of misinformation and prevent them from being spread to a wider audience.

1.4 *Identifying Potential Disinformation*

Frequently occurring words in the large dataset of tweets with information pertaining to the war in Ukraine can be categorized according to their importance and relevance to the topic, as well as their sentiments. Sentiment is the meaning attached to a given word, text, or text fragment and can be classified on a spectrum as being positive, negative, or neutral. The sentiment attached to specific words and groupings of words can shed light on the general user sentiment about specific people, places, or events that relate to the war. Initial research in this project has shown that the war is incredibly unpopular among Twitter users, with overwhelming condemnation of the actions being taken by the Russian state. Given that there is a common recurrence of sentiments about certain topics pertaining to the war, a deviation from popular opinion in a set of tweets can be used to help identify accounts that are spreading harmful disinformation. Categorizing frequently occurring keywords by sentiment can then shed light on specific tweets that deviate from popular sentiment, implying that they can be a source of Russian propaganda. Similar research being done on the topic currently has uncovered Russian disinformation being spread on Twitter relating to the war in Ukraine, mainly targeting Western countries, former Soviet nations, and NATO allied nations. A deeper analysis of tweet text content, hashtags, account names, and user locations will hopefully lead to the discovery of potential sources of disinformation.

2. Analysis

The initial analysis involved cleaning and reformatting the raw data in order to prepare the data for further analyses, such as regression analysis, neural networks and supervised learning methods. Once the data was cleaned and formatted, exploratory analysis yielded graphs, charts and clusters that helped visualize the word content of small sample groups of tweets. One of the main goals of the analysis was to attach sentiments to individual words and tweets, in order to help identify the general user sentiment on Twitter about certain words and topics pertaining to the war in Ukraine. Once trends were established and the dataset was labeled, additional analyses were performed on the tweet metadata, including hashtags and user locations.

2.1 *Data Sources*

The data used for the analysis in this project was gathered primarily from the Twitter API, along with the Ukraine Conflict Twitter Dataset from Kaggle. The Ukraine Conflict Twitter Dataset from Kaggle contains information gathered from around 50,000 tweets a day since the start of the war in Ukraine, making it a valuable source for analysis. The Kaggle dataset is updated daily, and since this war is ongoing and developing, major events can significantly impact tweet data. This project was approved for advanced access to the Twitter API, meaning that up to 2 million tweets can be gathered from the API per month, giving access to a massive amount of information to be used in analysis. At various points in the analysis, the size of the dataset had to be limited due to the computational limits of the machine on which analyses were performed. News headlines from NewsAPI.org were also gathered as a supplementary source. The NewsAPI.org data was used early in the project to help determine which words related to the war were trending in news headlines. The main analyses were performed on information gathered from both the Twitter API and the Ukraine Conflict Twitter Dataset from Kaggle.

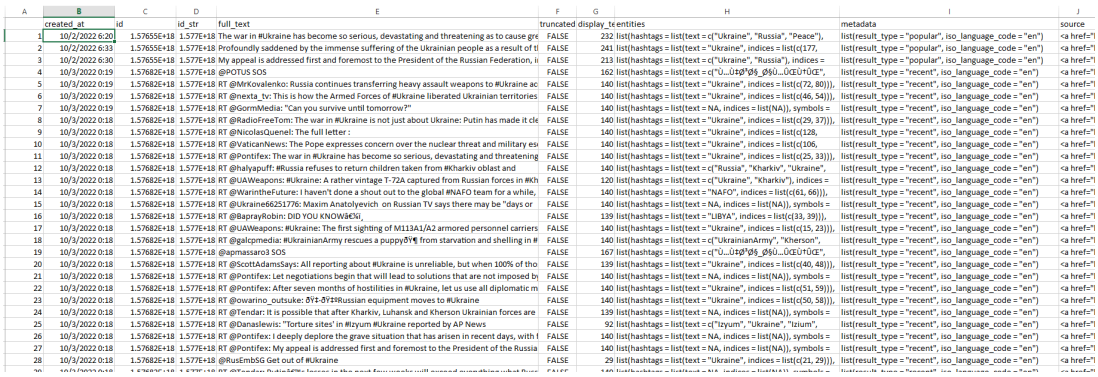
2.2 *Raw Data*

When the Twitter data was first accessed from the Twitter API and downloaded into a readable csv format, significant cleaning and parsing had to be done before the data was available to be used in further analyses. Data cleaning and preparation was an important phase in the project, and much was discovered about the text data during this phase. The main task that was undertaken during the data cleaning phase was to remove redundant and unnecessary information from the tweets, considering that over forty variables of tweet data and metadata were gathered for each individual tweet. Many of these variables had empty or missing values or contained irrelevant / uninterpretable information. The NewsAPI data was reformatted in a similar way to the data accessed

from the Twitter API. The data from the Ukraine Conflict Twitter Dataset on Kaggle had mostly been cleaned before being used in this project and only went through some minor reformatting changes.

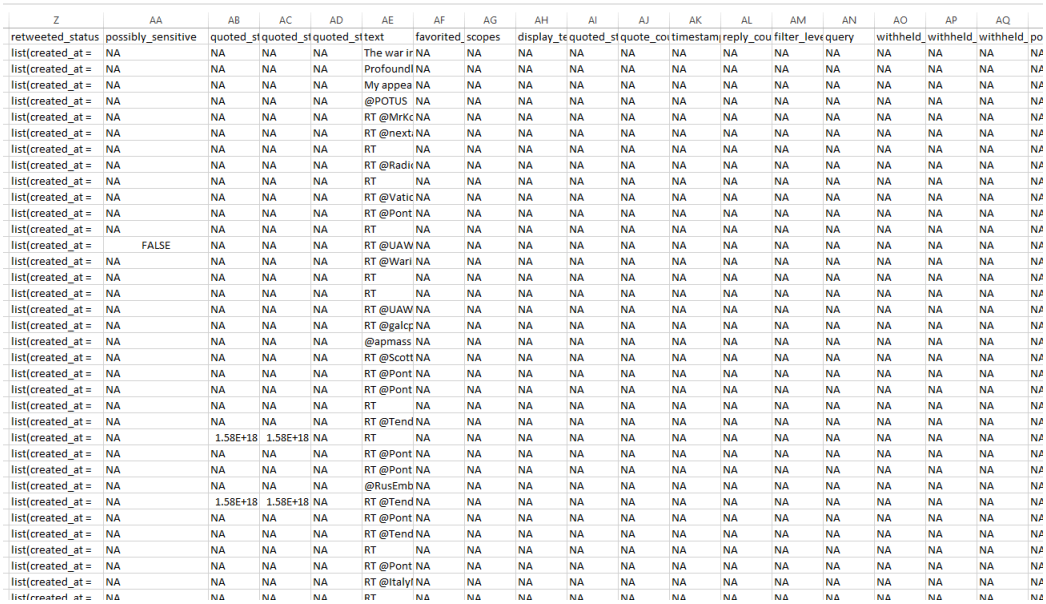
a) *Twitter Data Before and After Cleaning*

When the Twitter data was accessed from the API and imported into a csv file, the file contained large amounts of redundant and useless information. Each individual tweet was imported as a row in the dataset, with 44 variables of data attached to it. The removal of some of the unneeded variables helped reduce the size of the dataset and helped speed up the computational speed of the analyses that were performed. Columns that contained empty values, NA's, duplicate data, uninterpretable numerical metadata, and useless information were removed. Stopwords, punctuation, and foreign language text was also removed from the data. This helped with memory and computational speed, but also made the dataset more readable, since only the variables with relevant information were kept.



	A	B	C	D	E	F	G	H	I	J
		created_at	id	id_str	full_text	truncated	display_te	entities	metadata	source
1		10/2/2022 6:20	1.57655E+18	1.57E+18	The war in #Ukraine has become so serious, devastating and threatening as to cause gre	FALSE	232	list(hashtags = list(text = c("Ukraine", "Russia", "Peace"),	list(result_type = "popular", iso_language_code = "en")	<a href="h
2		10/2/2022 6:33	1.57655E+18	1.57E+18	Profoundly saddened by the immense suffering of the Ukrainian people as a result of t	FALSE	241	list(hashtags = list(text = "Ukraine", indices = list(c(177,	list(result_type = "popular", iso_language_code = "en")	<a href="h
3		10/2/2022 6:30	1.57655E+18	1.57E+18	My appeal is addressed first and foremost to the President of the Russian Federation, i	FALSE	213	list(hashtags = list(text = c("Ukraine", "Russia"), indices =	list(result_type = "popular", iso_language_code = "en")	<a href="h
4		10/3/2022 0:19	1.57682E+18	1.57E+18	@POTUS SOS	FALSE	162	list(hashtags = list(text = c("Ù…Ù‡Ø§Ø¬Ù…ÙˆÙ†",	list(result_type = "recent", iso_language_code = "en")	<a href="h
5		10/3/2022 0:19	1.57682E+18	1.57E+18	RT @MrKovalenko: Russia continues transferring heavy assault weapons to #Ukraine ac	FALSE	140	list(hashtags = list(text = "Ukraine", indices = list(c(72, 80))),	list(result_type = "recent", iso_language_code = "en")	<a href="h
6		10/3/2022 0:19	1.57682E+18	1.57E+18	RT @nexta_tv: This is how the Armed Forces of #Ukraine liberated Ukrainian territories	FALSE	140	list(hashtags = list(text = "Ukraine", indices = list(c(46, 54))),	list(result_type = "recent", iso_language_code = "en")	<a href="h
7		10/3/2022 0:19	1.57682E+18	1.57E+18	RT @GormMedia: "Can you survive until tomorrow?"	FALSE	140	list(hashtags = list(text = NA, indices = list(NA)), symbols =	list(result_type = "recent", iso_language_code = "en")	<a href="h
8		10/3/2022 0:18	1.57682E+18	1.57E+18	RT @RadioFreeTom: The war in #Ukraine is not just about Ukraine: Putin has made it cle	FALSE	140	list(hashtags = list(text = "Ukraine", indices = list(c(29, 37))),	list(result_type = "recent", iso_language_code = "en")	<a href="h
9		10/3/2022 0:18	1.57682E+18	1.57E+18	RT @NicolasQuenel: The full letter :	FALSE	140	list(hashtags = list(text = "Ukraine", indices = list(c(128,	list(result_type = "recent", iso_language_code = "en")	<a href="h
10		10/3/2022 0:18	1.57682E+18	1.57E+18	RT @VaticanNews: The Pope expresses concern over the nuclear threat and military es	FALSE	140	list(hashtags = list(text = "Ukraine", indices = list(c(106,	list(result_type = "recent", iso_language_code = "en")	<a href="h
11		10/3/2022 0:18	1.57682E+18	1.57E+18	RT @Pontifex: The war in #Ukraine has become so serious, devastating and threatening	FALSE	140	list(hashtags = list(text = "Ukraine", indices = list(c(25, 33))),	list(result_type = "recent", iso_language_code = "en")	<a href="h
12		10/3/2022 0:18	1.57682E+18	1.57E+18	RT @halyapuff: #Russia refuses to return children taken from #Kharkiv oblast and	FALSE	140	list(hashtags = list(text = c("Russia", "Kharkiv", "Ukraine",	list(result_type = "recent", iso_language_code = "en")	<a href="h
13		10/3/2022 0:18	1.57682E+18	1.57E+18	RT @UAWeapons: #Ukraine: A rather vintage T-72A captured from Russian forces in #Kh	FALSE	120	list(hashtags = list(text = c("Ukraine", "Kharkiv"), indices =	list(result_type = "recent", iso_language_code = "en")	<a href="h
14		10/3/2022 0:18	1.57682E+18	1.57E+18	RT @WarintheFuture: I haven't done a shout out to the global #NAFO team for a while,	FALSE	140	list(hashtags = list(text = "NAFO", indices = list(c(61, 66))),	list(result_type = "recent", iso_language_code = "en")	<a href="h
15		10/3/2022 0:18	1.57682E+18	1.57E+18	RT @Ukraine66251776: Maxim Anatolyevich on Russian TV says there may be "days or	FALSE	140	list(hashtags = list(text = NA, indices = list(NA)), symbols =	list(result_type = "recent", iso_language_code = "en")	<a href="h
16		10/3/2022 0:18	1.57682E+18	1.57E+18	RT @BaprayRobin: DID YOU KNOWâ€¼ï¸	FALSE	139	list(hashtags = list(text = "LIBYA", indices = list(c(33, 39))),	list(result_type = "recent", iso_language_code = "en")	<a href="h
17		10/3/2022 0:18	1.57682E+18	1.57E+18	RT @UAWeapons: #Ukraine: The first sighting of M113A1/A2 armored personnel carriers	FALSE	140	list(hashtags = list(text = "Ukraine", indices = list(c(15, 23))),	list(result_type = "recent", iso_language_code = "en")	<a href="h
18		10/3/2022 0:18	1.57682E+18	1.57E+18	RT @galcpmedia: #UkrainianArmy rescues a puppyðŸ¶ from starvation and shelling in #	FALSE	140	list(hashtags = list(text = c("UkrainianArmy", "Kherson",	list(result_type = "recent", iso_language_code = "en")	<a href="h
19		10/3/2022 0:18	1.57682E+18	1.57E+18	@apmassaro3 SOS	FALSE	167	list(hashtags = list(text = c("Ù…Ù‡Ø§Ø¬Ù…ÙˆÙ†",	list(result_type = "recent", iso_language_code = "en")	<a href="h
20		10/3/2022 0:18	1.57682E+18	1.57E+18	RT @ScottAdamsSays: All reporting about #Ukraine is unreliable, but when 100% of tho	FALSE	139	list(hashtags = list(text = "Ukraine", indices = list(c(40, 48))),	list(result_type = "recent", iso_language_code = "en")	<a href="h
21		10/3/2022 0:18	1.57682E+18	1.57E+18	RT @Pontifex: Let negotiations begin that will lead to solutions that are not imposed b	FALSE	140	list(hashtags = list(text = NA, indices = list(NA)), symbols =	list(result_type = "recent", iso_language_code = "en")	<a href="h
22		10/3/2022 0:18	1.57682E+18	1.57E+18	RT @Pontifex: After seven months of hostilities in #Ukraine, let us use all diplomatic m	FALSE	140	list(hashtags = list(text = "Ukraine", indices = list(c(51, 59))),	list(result_type = "recent", iso_language_code = "en")	<a href="h
23		10/3/2022 0:18	1.57682E+18	1.57E+18	RT @owarino_outsuke: ðŸ‡ºðŸ‡¸Russian equipment moves to #Ukraine	FALSE	140	list(hashtags = list(text = "Ukraine", indices = list(c(50, 58))),	list(result_type = "recent", iso_language_code = "en")	<a href="h
24		10/3/2022 0:18	1.57682E+18	1.57E+18	RT @Tendar: It is possible that after Kharkiv, Luhansk and Kherson Ukrainian forces are	FALSE	139	list(hashtags = list(text = NA, indices = list(NA)), symbols =	list(result_type = "recent", iso_language_code = "en")	<a href="h
25		10/3/2022 0:18	1.57682E+18	1.57E+18	RT @Danaslewis: "Torture sites' in #Izyum #Ukraine reported by AP News	FALSE	92	list(hashtags = list(text = c("Izyum", "Ukraine", "Izium",	list(result_type = "recent", iso_language_code = "en")	<a href="h
26		10/3/2022 0:18	1.57682E+18	1.57E+18	RT @Pontifex: I deeply deplore the grave situation that has arisen in recent days, with t	FALSE	140	list(hashtags = list(text = NA, indices = list(NA)), symbols =	list(result_type = "recent", iso_language_code = "en")	<a href="h
27		10/3/2022 0:18	1.57682E+18	1.57E+18	RT @Pontifex: My appeal is addressed first and foremost to the President of the Russia	FALSE	140	list(hashtags = list(text = NA, indices = list(NA)), symbols =	list(result_type = "recent", iso_language_code = "en")	<a href="h
28		10/3/2022 0:18	1.57682E+18	1.57E+18	@RusEmbSG Get out of #Ukraine	FALSE	29	list(hashtags = list(text = "Ukraine", indices = list(c(21, 29))),	list(result_type = "recent", iso_language_code = "en")	<a href="h

Figure 1. The raw data as downloaded from the API shows each tweet as a row in the dataset. There are 44 columns containing information such as the full tweet text, user information, geographic location, retweet status, date, etc. Only a select few of the 44 total variables for each tweet contain useful information.



Z	AA	AB	AC	AD	AE	AF	AG	AH	AI	AJ	AK	AL	AM	AN	AO	AP	AQ
retweeted_status	possibly_sensitive	quoted_st	quoted_st	quoted_st	text	favorited	scopes	display_te	quoted_st	quote_cou	timestam	reply_cou	filter_leve	query	withheld_	withheld_	withheld_
list(created_at =	NA	NA	NA	NA	The war ir	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
list(created_at =	NA	NA	NA	NA	Profoundl	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
list(created_at =	NA	NA	NA	NA	My appea	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
list(created_at =	NA	NA	NA	NA	@POTUS	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
list(created_at =	NA	NA	NA	NA	RT @MrKc	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
list(created_at =	NA	NA	NA	NA	RT @nexti	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
list(created_at =	NA	NA	NA	NA	RT	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
list(created_at =	NA	NA	NA	NA	RT @Radi	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
list(created_at =	NA	NA	NA	NA	RT	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
list(created_at =	NA	NA	NA	NA	RT @Vatic	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
list(created_at =	NA	NA	NA	NA	RT @Pont	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
list(created_at =	NA	NA	NA	NA	RT	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
list(created_at =	FALSE	NA	NA	NA	RT @UAW	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
list(created_at =	NA	NA	NA	NA	RT @Wari	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
list(created_at =	NA	NA	NA	NA	RT	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
list(created_at =	NA	NA	NA	NA	RT	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
list(created_at =	NA	NA	NA	NA	RT @UAW	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
list(created_at =	NA	NA	NA	NA	RT @galcp	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
list(created_at =	NA	NA	NA	NA	@apmass	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
list(created_at =	NA	NA	NA	NA	RT @Scott	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
list(created_at =	NA	NA	NA	NA	RT @Pont	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
list(created_at =	NA	NA	NA	NA	RT @Pont	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
list(created_at =	NA	NA	NA	NA	RT	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
list(created_at =	NA	NA	NA	NA	RT @Tend	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
list(created_at =	NA	1.58E+18	1.58E+18	NA	RT	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
list(created_at =	NA	NA	NA	NA	RT @Pont	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
list(created_at =	NA	NA	NA	NA	RT @Pont	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
list(created_at =	NA	NA	NA	NA	@RusEmb	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
list(created_at =	NA	1.58E+18	1.58E+18	NA	RT @Tend	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
list(created_at =	NA	NA	NA	NA	RT @Pont	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
list(created_at =	NA	NA	NA	NA	RT @Tend	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
list(created_at =	NA	NA	NA	NA	RT	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
list(created_at =	NA	NA	NA	NA	RT @Pont	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
list(created_at =	NA	NA	NA	NA	RT @Italyl	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
list(created_at =	NA	NA	NA	NA	RT	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA

Figure 2. Most of the variables were filled with NAs or contained information that would've proved of no use in the analysis. The data cleaning for the Twitter API data focused on first removing the unnecessary variables and determining what would be the most useful in future analyses. The cleaned tweet dataset contains 18 variables out of the original 44. The Kaggle data was also drawn from the Twitter API, and thus was formatted in an identical format. The same unnecessary variables containing NA's were removed from the Kaggle dataset, significantly reducing the amount of variables.

b) *Raw NewsAPI Data*

The NewsAPI dataset had much fewer variables than the data accessed from Twitter, with only five variables total (label, date, source, title, headline). The label variable was specified when accessing the API to search out keywords in the headlines. A few labels were chosen to create an initial exploratory dataset to determine which topics and words were most relevant to the current state of the war. Data cleaning consisted mainly of removing empty values and NA's.

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	LABEL	Date	Source	Title	Headline								
2	kherson	11/9/2022	bbc-news	Kherson L	city residents feel mounting dread wondering what occupiers will they leave								
3	kherson	11/11/2022	bbc-news	Kherson b	Moscow pullback from Kherson caps stunning three month change fortune Ukrainians								
4	kherson	10/28/2022	bbc-news	Russia en	eight operation ends Kherson city Russian officials prepare Ukrainian attack								
5	kherson	11/9/2022	bbc-news	Ukraine w	Russia retreating from Kherson only regional capital captured during invasion								
6	kherson	11/24/2022	bbc-news	Ukraine w	Engineers starting rebuild infrastructure destroyed during Russia occupation Kherson								
7	kherson	11/12/2022	bbc-news	Ukraine w	Jubilant scenes continuing after Ukraine took back city Kherson from Russia								
8	kherson	11/17/2022	bbc-news	Ukraine w	speaks people held Russians Kherson amid fresh reports atrocities there								
9	kherson	11/19/2022	bbc-news	Tears as fi	Crowds gathered bridge welcome train families reunited after months apart								
10	kherson	11/13/2022	bbc-news	Kherson I	Jeremy Bowen reports from city newly liberated from Russian control								
11	kherson	11/11/2022	bbc-news	Ukraine w	official says Ukraine troops almost fully control after Russia completed retreat								
12	kherson	11/10/2022	bbc-news	Kherson A	learnt believe word Russians resident tells Jeremy Bowen								
13	kherson	11/10/2022	bbc-news	Ukraine w	Kyiv reports rapid advances seven kilometres after Moscow said would leave city								
14	kherson	11/16/2022	bbc-news	Ukrainian	They able visit their homes relatives first time since Russian invasion began								
15	kherson	11/14/2022	bbc-news	Why did Z	Since Russia retreat from region memes fruit have been widely shared online								
16	kherson	11/5/2022	bbc-news	Ukraine w	Russian leader approves civilians leaving dangerous areas city southern Ukraine								
17	kherson	11/11/2022	bbc-news	Ukraine s	Russia withdrew troops from region Moscow officials								

Figure 3. The raw data from NewsAPI.org went through a cleaning process where punctuation and spurious characters were removed and the final dataset was created with five variables (label, date, source, title and headline). The *label* variable consists of a keyword in a news headline. A few keywords such as Putin, Missile, Zelensky, Kherson, etc. were chosen as labels to create an initial dataset with information relevant to the war.

2.3 *Data Exploration*

A few initial visualizations were created from the cleaned datasets in order to further explore the data and determine which steps should be taken next. Graphs showing favorites vs followers count and favorites vs retweets were created from the Twitter and Kaggle datasets to see if there is any correlation between these variables. A list of the top followers and unique geographic locations was created to see the general distribution of users from the Twitter API data from a small sample dataset of 100 tweets. A word cloud was created from the NewsAPI.org headlines to see what were some of the most frequently appearing keywords presented across world news headlines. Initial exploration of the data showed some interesting associations worthy of further research. During the data cleaning process, a large quantity of tweets were visually inspected for their text content, helping to identify key words that could be used for sentiment analysis.

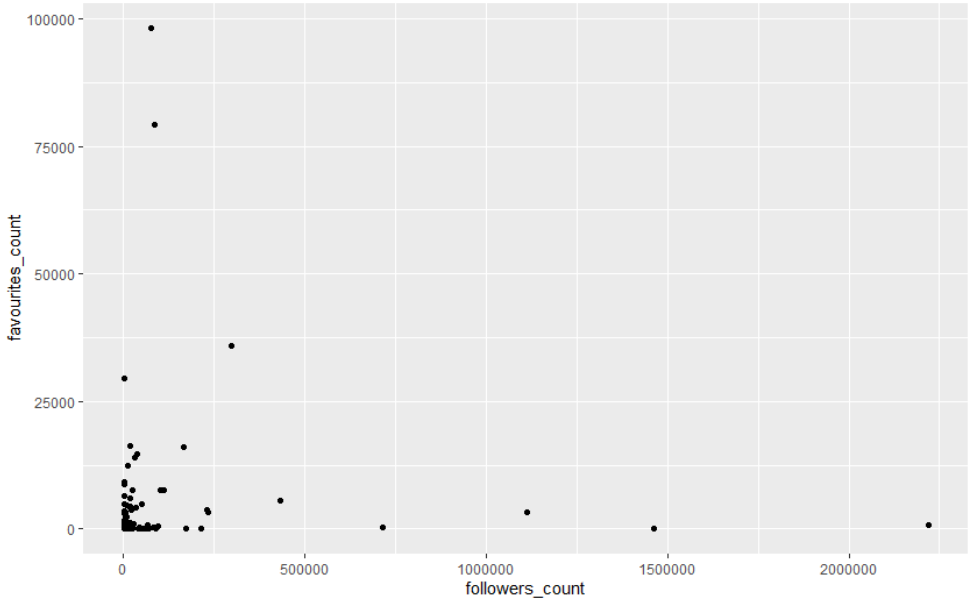


Figure 4. Graph showing the association between the amount of followers and the amount of favorites for a given user in the Twitter dataset. A graph of these variables shows that there is little correlation between them. The amount of followers a given Twitter user has does not indicate that they have a large amount of favorites.

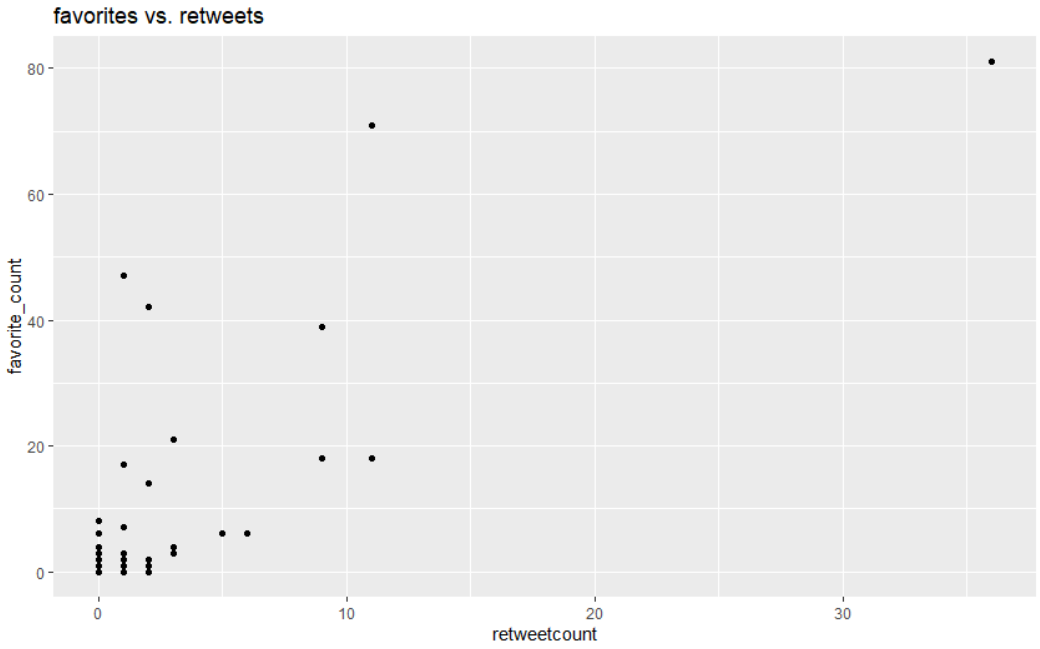


Figure 5. Graph showing the association between the amount of retweets and the amount of favorites that a given user has, drawn from the Kaggle dataset. There is a correlation between the amount of favorites and the amount of retweets that a given user has. In general, a larger retweet count indicates that the user will have a lot of favorite tweets.

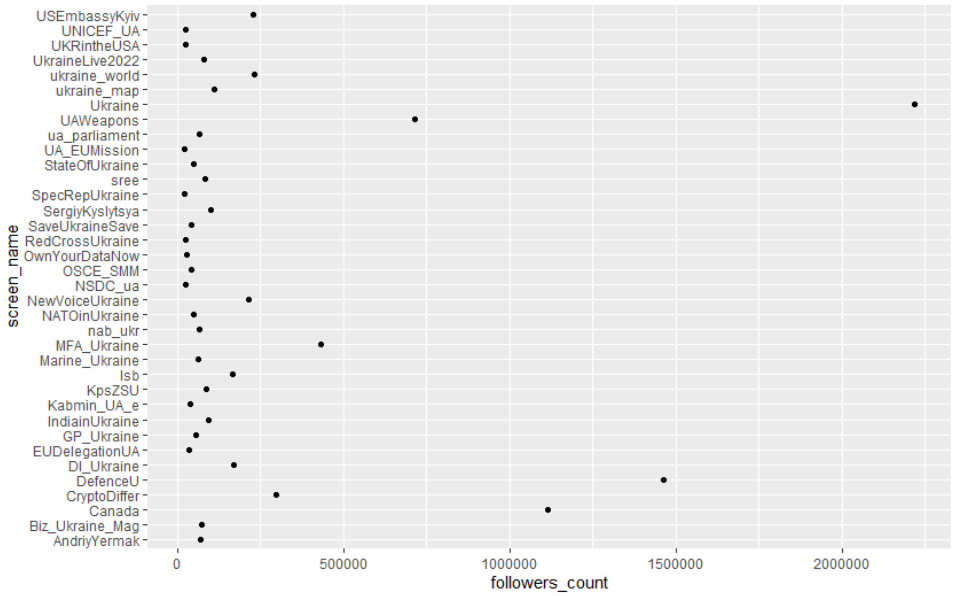


Figure 5. A small chart was created, using the Twitter dataset to show which of the users had the most followers. Further investigation of these accounts will be useful in finding information that is pertinent to the war and keeping track of ongoing developments. As the war continues, certain keywords may pop up that can be used in the analysis to help find any potential sources of disinformation that is being spread about the topic.

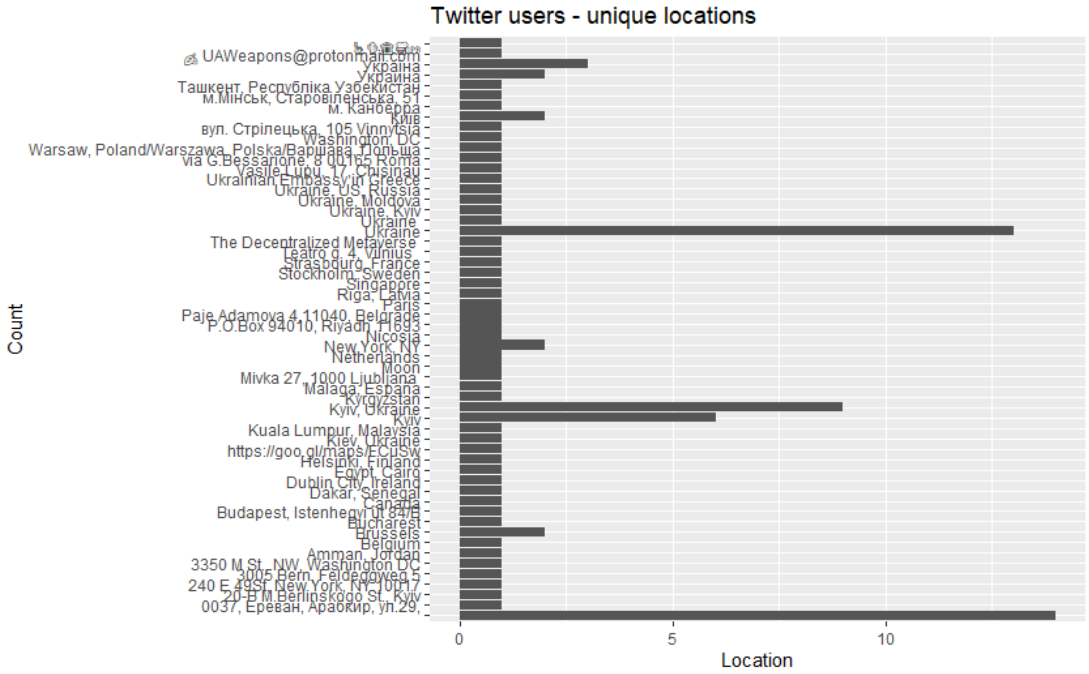


Figure 6. A chart of unique locations was created from the Twitter dataset. Many locations are found in Ukraine and surrounding countries. There are a few interesting outliers, notably “The Decentralized Metaverse” , “UAWeapons@protonmail.com” and “M.Minsk, Starovilenska 51”, an address from Belarus, a country that is allied with Russia. Further investigation into these locations might yield some interesting information.

Figure 7. A wordcloud was created from the NewsAPI.org news headlines. Popular words such as “Kherson” , “Missile” and “Zelensky” in the example above were designated as the topic word, finding which other words were commonly associated with these keywords in the headline text. This analysis will further help identify keywords that can be used for further studies such as ARM. It is evident that there are many “useless” words such as common articles and nouns that contain no meaningful information pertaining to the war. There will have to be data cleaning done in all of the datasets to prevent these words from cluttering the datasets.

2.4 *Sentiment Analysis*

One of the central goals of the analysis was to determine the sentiment that was attached to individual keywords present in the tweet text data. One general trend that was discovered during visual inspection of the datasets was that tweets that report on the same topic generally have a similar sentiment attached to that topic. For example, the sentiment is overwhelmingly positive when describing actions being undertaken by Ukrainian forces while the sentiment is overwhelmingly negative when describing the actions of the Russian army. A hypothesis was posed based on these observations, that sentiments attached to certain topics that are opposing the sentiments of the majority of Twitter users might point to possible disinformation.

Sentiments were classified into three categories, as positive, negative and neutral. Positive and neutral sentiments were labeled as 1 and negative sentiments were labeled as 0 in the data analyses. In order to label individual tweets with the sentiment attached to them, a text classification model known as roBERTa was used to determine the sentiment attached to individual words and groupings of words. The version of roBERTa used in the sentiment analysis was especially designed for Twitter data, and was trained on a large dataset of 122 million tweets. roBERTa classification was run on over one million tweets collected by Kaggle in September and October of 2022. A subset of the tweets labeled by ro-BERTa were further analyzed for their word content, and individual words were ranked by their importance and frequency in the dataset and were labeled with a TF-IDF score (term frequency inverse-document frequency). Several analyses were performed on the labeled data, using machine learning methods such as clustering, association rule mining, naive Bayes, data trees, support vector machines, linear regression and neural networks.

2.5 *Clustering*

Clustering is an unsupervised method used to discover if data in the dataset fits into any type of group, category or class. By recording the count of words contained in individual tweets, clustering was used to discover if there was any association between the tweets and how groupings of words can be categorized. Because the dataset that was used was high dimensional, this data was difficult to visualize and lots of data cleaning (such as removing stopwords, punctuation, foreign-language words, and other words) had to be done in order to make the data more workable. Clustering may be based on density or on distance, and in this analysis only the distance-based clustering was performed. Density-based clustering was not performed due to the lack of well-defined clusters of similar sizes that would allow for the use of density-clustering methods like DBSCAN. A larger dataset with common associations ruled out might allow for the use of

density-based clustering in the future, once meaningful and significant clusters are discovered. Both the Twitter API dataset and the Kaggle dataset were used for clustering, with the first 100 tweets from the Twitter API and the first 200 rows from the Kaggle dataset.

a) *Distance Measures*

The distance measure defines the calculation of the similarity of two elements (x, y), implies how similar they are, and how it will influence the shape of the clusters. The distance measure defines the separation between the clusters. Short distances with respect to each other form a cluster. Euclidean, normalized and cosine distances were used in the analysis. Euclidean distance is the square root of the sum of squares of the differences between two points. Euclidean distance is more effective for small datasets, and is less effective for a large dataset like the ones being used in this analysis. The cosine distance finds the angle between two data points, and is more effective for larger datasets. The normalized squared euclidean distance gives the squared distance between two vectors where lengths have been scaled to unit form. The squared distance can be useful when the direction of the vector is meaningful but the size of the vector is not.

b) *Hierarchical Clustering*

Hierarchical clustering plots were created using Euclidean distance, normalized cosine similarity and cosine similarity measures. Hierarchical clustering produces a set of nested clusters organized as a hierarchical tree, and is another method used to cluster data into potential groups. Hierarchical clustering can be categorized into two main categories: Agglomerative clustering (AGNES), which works in a bottom-up approach, and divisive hierarchical clustering (DIANA), which works in a top-up approach when creating the hierarchy.

c) *K-Means Clustering*

In the analysis, distance maps were created using cosine similarity, Euclidean, Manhattan, Pearson, Canberra, and Spearman distance measures. As the next step, k-means clustering was performed using the same distance measures. K-means clustering is a partitional clustering approach that assigns a centroid to each cluster and connects the centroid to the closest centroid of the surrounding clusters. The goal of k-means clustering is to connect the *k* amount of total centroids in such a way that the sum of the total connected distances is minimized.

d) *Density Clustering*

Density based clustering is used to detect areas where points are concentrated and to find out how data is connected and distributed. Defined Distance (DBSCAN) is a common method that finds clusters of similar size, and removes outlying data between the clusters (noise). DBSCAN requires clusters of similar size and distribution in order to be most effective, and given the varying size and distribution of clusters in the dataset, density clustering was not performed in this analysis.

2.6 *Association Rule Mining (ARM)*

Association rule mining (ARM) is a method used to find interesting and meaningful associations between different points of data in a dataset. For example, given text data, ARM can be used to find if it's probable that the presence of one word or group of words can indicate the presence of another word or group of words within a data sample, such as a tweet. ARM uses three metrics; confidence, lift, and support, to determine how likely it is that different rules are meaningful. Confidence describes the likeliness of occurrence of a word *X* (in the case of text data) given the word *Y.* The variables *X* and *Y* can also be pairs or groupings of words. Lift is a measure of importance of a rule. Support shows how frequent a word is given all the words in a dataset. High confidence implies that an association is correct often, high lift implies that an association is not just coincidental, and high support implies that an association applies to a large number of cases. In this analysis, ARM methods were used to determine associations between a variety of keywords related to the subject of the war in Ukraine. Data from the Twitter API was not used for ARM and all data for these analyses were derived from the Kaggle Ukrainian Conflict dataset, with the first 80 rows of tweets from the dataset being used. The text of each tweet was tokenized into words and output to a cleaned .csv file. The cleaned file was then read as a basket of tweet transactions. For the initial analyses, both clustering and ARM were used to discover the most basic associations between words / groupings of words. Rules and associations that contain redundant or useless information were rooted out and eliminated. This analysis was performed on 10/1/22 and all the information in the analysis is from tweets prior to that data. Due to the constantly changing nature of tweet contents, the date accessed has a critical impact on the data that is derived.

2.7 *Decision Trees*

Analyses were performed on the Twitter dataset using supervised learning methods, including decision trees, Naive Bayes, and support vector machines (SVM). These methods produced results of varying degrees of accuracy for sentiment analysis based on the size of the tweet data that was used. Before the data was analyzed through supervised learning methods, data cleaning and labeling had to be performed on the raw dataset. This process was done by using a modified version of a Jupyter Notebooks Python code borrowed from Kaggle. The supervised learning methods were run in R using this labeled and cleaned dataset. Several graphs and charts were generated, outlining the results that these different supervised learning methods produced.

(https://www.kaggle.com/code/ssaisuryateja/eda-and-sentiment-analysis/notebook).

tweetid	text	hashtags	Sentiment
1.56875E+18	http Adds 19 Popular ERC-20 And DeFi Tokens To Its Platform	['trading', 'ripple', 'russia', 'ELON'	neutral
1.56875E+18	Saw #Ukraine is trending with almost 800k tweets. Time for coffee and then surely a few har	['Ukraine', 'Russian']	neutral
1.56875E+18	Drone footage of Pakistan Tehreek e Insaaf Jalsa Gujranwala ðŸ‡µðŸ‡°ðŸ«	['GujranwalaJalsa', 'TikTok', 'Amir	positive
1.56875E+18	@user @user @user @user @user @user @user @user @user @user @user @user @user @	['Putin', 'NATO']	negative
1.56875E+18	Keep fighting and kicking those bullies out of there #Ukriane http	['Ukriane']	negative
1.56875E+18	ðŸ“ˆ The Reserve Bank of #Australia raised the cash rate by 50bps to 2.35% on Tuesday, #Russ	['Australia', 'Russian', 'payment',	neutral
1.56875E+18	@user @user @user @user @user @user @user @user Maybe non-stop shelling of #Kharkh	['Kharkhiv', 'KharkivOffensive', 'R	negative
1.56875E+18	@user he could stop playing it series b actor!	['Ukraine', 'warisnotgame', 'humi	negative
1.56875E+18	#Putler #kills civilians in #UkraineRussiaWar !!	['Putler', 'kills', 'UkraineRussiaWa	negative

Figure 1. Twitter data from the cleaned csv file containing the tweetid, corpus text, hashtags and the sentiment label. The supervised learning methods performed analysis on only the text data.

A decision tree is a tool used for classification and categorization of information by passing data through a tree-like structure. Decision trees are divided into the basic structure of nodes, branches and leaves. Each internal node denotes a test on a certain attribute, each branch represents the outcome of this test, and the leaf nodes (the last nodes on the tree) hold the labeled and categorized data. All data flows from the root node (top node) and is sorted into the terminal leaf nodes.

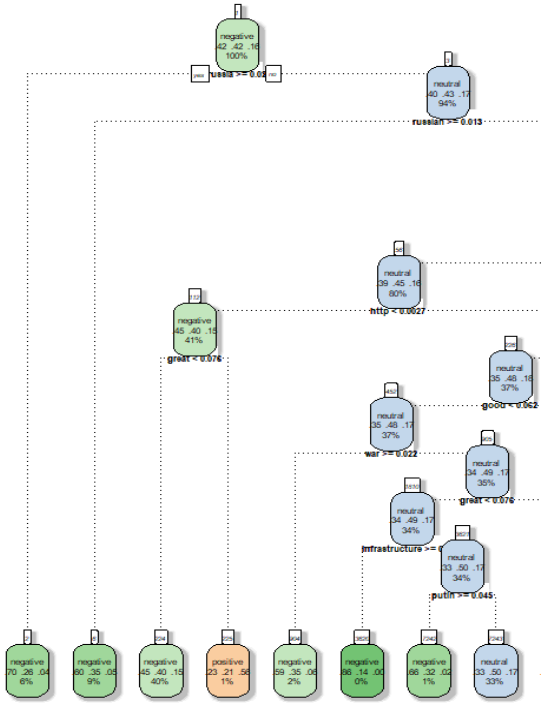


Figure 2. A sample from a DT that was generated in the tweet sentiment analysis. Nodes higher up in the tree show words that were found more frequently in the dataset. These words are split on a given set of criteria and are split by sentiment into the lower nodes (positive, negative, neutral).

a) *Parameters*

The Gini index and the entropy are two important parameters that are used to determine the splitting criteria in a decision tree. The Gini index is a measure between 0 and 1, where 0 signifies that all the observations belong to one class and 1 signifies that the data is random and disorganized. The goal is to have a Gini index as low as possible in order to have more accurate splitting in the decision tree. The entropy is a measure of the uncertainty in a group of

observations. The Gini index and the entropy can be used to calculate the information gain and the gain ration, which determine how effectively the information was split. The complexity parameter, which is used to determine the optimal size of a tree, was another parameter that was adjusted.

b) *Sentiment Analysis through DT's*

TF-IDF (term frequency inverse document frequency) is a numerical statistic that was gathered for each word in the text corpus. The amount of unique words in the text corpus was counted individually and a TF-IDF score was assigned to each unique word. The TF-IDF score denotes the most frequently repeating and important words in the entire text corpus. Multiple decision trees were generated and the parameters were adjusted for each decision tree, such as the complexity parameter, the size of the analyzed dataset, and the splitting criteria. The goal of creating the decision trees was to find the most frequently occurring words in the dataset and to split them into three categories based on the sentiment attributed to those words (positive, negative, neutral).

2.8 *Naive Bayes*

Naive Bayes is an algorithm used for classification problems that makes a naive assumption that all features are independent. The naive assumption states that the presence of any unique feature in a set means that this feature is unrelated to any other features in the set. This algorithm is a modification of the Bayes Theorem, which is used to determine the probability of *Y* given *X*.

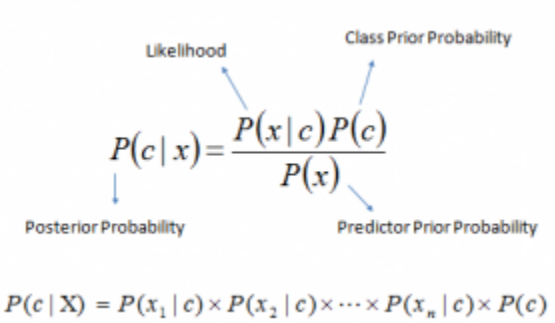


Figure 3. The Naive Bayes algorithm calculates the posterior probability of a class *c* given a set of attributes *x*. The posterior probability of *c* is calculated using the class prior probability, the likelihood of *c* given the presence of *x* and the predictor prior probability.

Naive Bayes classification is being trained to recognize whether the sentiment in a tweet is positive, negative, or neutral. The labeled training data is used to train the model using the Naive Bayes algorithm. This model is run against the test data, which is unlabeled, to determine the accuracy of the algorithm at predicting the labels. The accuracy of these results is presented as a confusion matrix.

2.9 *Support Vector Machines (SVM)*

A support vector machine (SVM) is a type of algorithm that performs supervised learning to classify labeled data. Known and labeled data, such as the text corpus from the tweets are transformed into a higher dimensional space. In this higher dimensional space the data points are separated by a hyperplane to separate groups according to patterns. SVM is run to separate individual occurrences of words in the tweets by sentiment. Groupings of words by sentiment were found, dividing the points by sentiment into positive, negative or neutral.

2.10 *Neural Networks*

As a very generalized overview, neural networks use a connected network of functions where they take data as an input, perform a given set of tasks to manipulate that data, and then return the data as an output. These networks of functions are called *neural* due to the similarities they have between the connectivities of the human brain. In the initial phase of neural network processing, labeled examples are processed by the neural network in the training phase to "learn" what characteristics of the input data are needed to accurately construct the output data. Neural networks learn with experience, and in general, the more data a neural network processes in the training phase, the more accurate results it will produce. When inputs are transmitted between neural network "neurons" the weights are applied to the input and are passed into an activation function along with the bias. Neural networks also typically have hidden layers, which perform non-linear transformations on the input data. Neural networks can have an almost unlimited amount of parameters and hidden layers, giving rise to processes of incredible complexity.

Neural networks are derived from the foundational algorithms in linear regression. Linear regression using gradient descent and a sigmoid activation function illustrates the mathematical basis of neural networks. In linear regression, a line of best fit is found between a group of points, such that the loss between the line and individual points is minimized. The sigmoid activation function is used to apply a non-linear transformation to the data, if the data points are too scattered. Below are sample equations for a linear regression model using sigmoid activation for a two input model.

2D input X (x_1 and x_2)

Loss Function: $L = \frac{1}{2n} \; SUM(\hat{y} - y)^2$

Linear Equation: $z = w_1x_1 + w_2x_2 + b$

Weight vector W (with w_1 and w_2) and bias b.

Sigmoid Activation Function:

$$\hat{y} = \sigma(z) = \frac{1}{1+e^{-z}}$$
$$\hat{y} = \sigma(w_1x_1 + w_2x_2 + b)$$

In the Ukrainian war tweet analysis, it is important to be able to determine the overall sentiment that is present in an individual tweet. This project attempts to find disinformation in tweet data about the war in Ukraine, and it is important to determine what the user sentiment is across a large sample of tweets in order to perform analysis. If certain tweets are labeled with a sentiment that deviates from the majority, it is oftentimes likely that these tweets are associated with disinformation. The neural network that was created predicts the sentiment label for a tweet (positive, neutral, or negative) based on the text that is present in the tweet. Tweets with a positive or neutral sentiment have a label of 1, while tweets with a negative sentiment have a label of 0. A training corpus of 5000 tweets labeled by their overall sentiment was used to train the network with 4750 tweets and 250 test tweets from the same corpus were run through the network to label them by sentiment. The input layer had 194 nodes with one hidden layer with 16 nodes in a fully connected network. A sigmoid function was used as an activation function in the hidden layer and in the output layer.

2.11 *Hashtag Analysis*

A discovery about tweet hashtags was made using ARM in the earlier stages of the analysis that brought up some interesting discoveries. In the top rules for lift, confidence, and support from a small sample of the tweet data, the most popular association rules were all in Mandarin, despite the search filters having been set to keep English-only text tweets in the dataset. Further investigation showed that these foreign-language tweets were tagged as English language in the tweet metadata. The exact reason for this is unknown, but the discovery of this anomaly brought up the hypothesis that the tweet metadata might have valuable data that could point to potential sources of disinformation.

An experiment was set up where a large sample of tweets was filtered by known pro-Russia hashtags such as “UkraineWarCriminal” , “ukrainiannazi” and “NaziUkraine.” Tweets were also filtered using words that were found to have a strong negative sentiment from the previous analyses that were conducted. The tweets containing these hashtags were filtered by unique locations and unique usernames.

3. Results

The result led to some interesting conclusions about the presence of disinformation on Twitter. Originally, it was expected that sentiment analysis of words and tweets would help identify tweets that have a positive or negative sentiment towards certain topics and that unusual sentiments might point to disinformation. This attempt was largely unsuccessful, although it did help lead to some interesting results.

Some interesting results appeared through association rule mining, which accidentally discovered a dataset that was mislabeled in its language metadata, and possessed a text corpus that pointed to potential disinformation. Based on these findings, it was decided that analysis of hashtags, user location and other tweet metadata would potentially lead to some tweets that were disinformation. In this process, categorizing words by sentiment helped identify some keywords with a strongly negative sentiment attached to them, which was helpful in identifying the tweets that have disinformation in their text contents. A large number of tweets containing disinformation was discovered by filtering specific hashtags and locations.

3.1 *Sentiment Analysis*

Sentiment analysis was achieved mainly through supervised learning methods such as decision trees and support vector machines, as well as neural networks. Sentiments were assigned to keywords that were previously identified in the datasets, but this did not directly lead to the identification of tweets that contained disinformation. The sheer volume of tweets that were present in the dataset resulted in computational difficulties with processing such large amounts of information through the machine learning analyses. Apart from the volume of tweets in the dataset, the tweets that were analyzed contained an incredible diversity of words, many of which had obscure or unknown meanings when separated from the context in which they were used. The most frequently appearing words that were present in the dataset pertaining to the subject of the war, such as “Ukraine”, “Putin” and “Russia” were labeled according to their sentiment. This did not yield anything that was of significance that would lead to potential disinformation. The results of the sentiment analysis prompted further research into findings that were discovered with ARM, and a deeper dive into the tweet metadata led to some interesting findings.

3.2 *Clustering*

The clustering analysis was informed by the distance maps for each of the distance measures under consideration. The distance map plots are shown below.

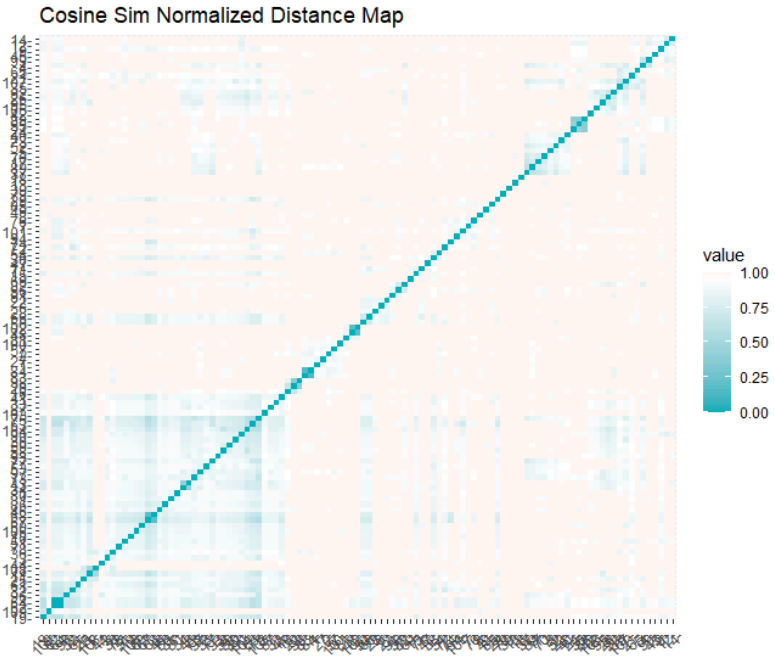


Figure 3. Map of cosine similarity normalized distance.

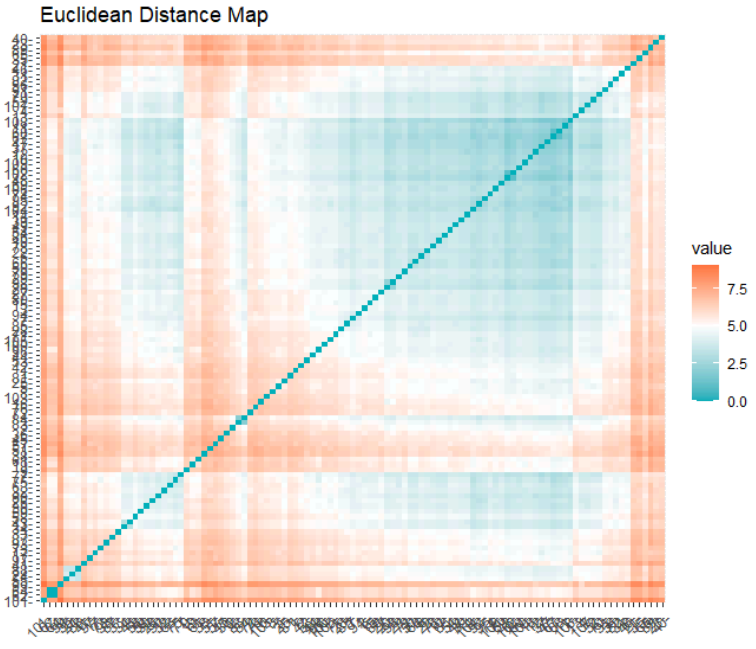


Figure 4. Map of Euclidean distance.

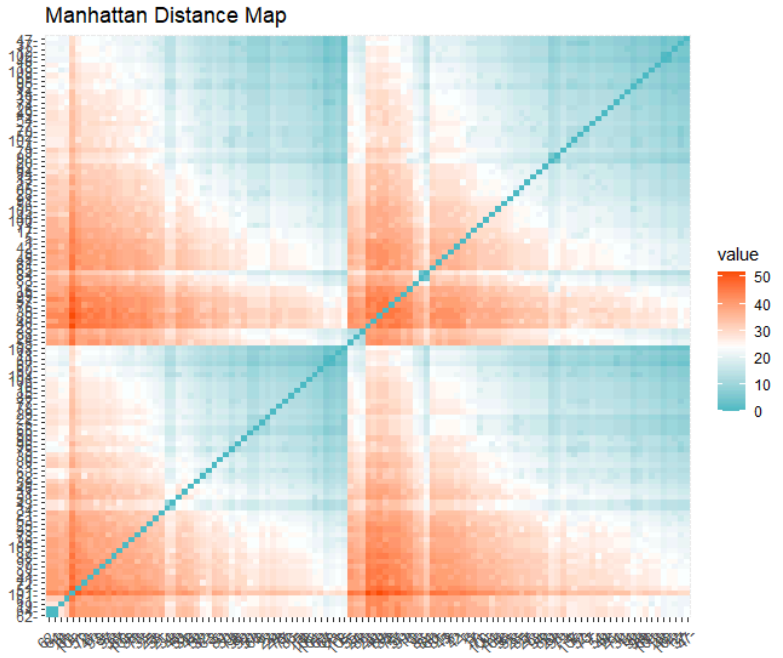


Figure 5. Map of Manhattan distance.

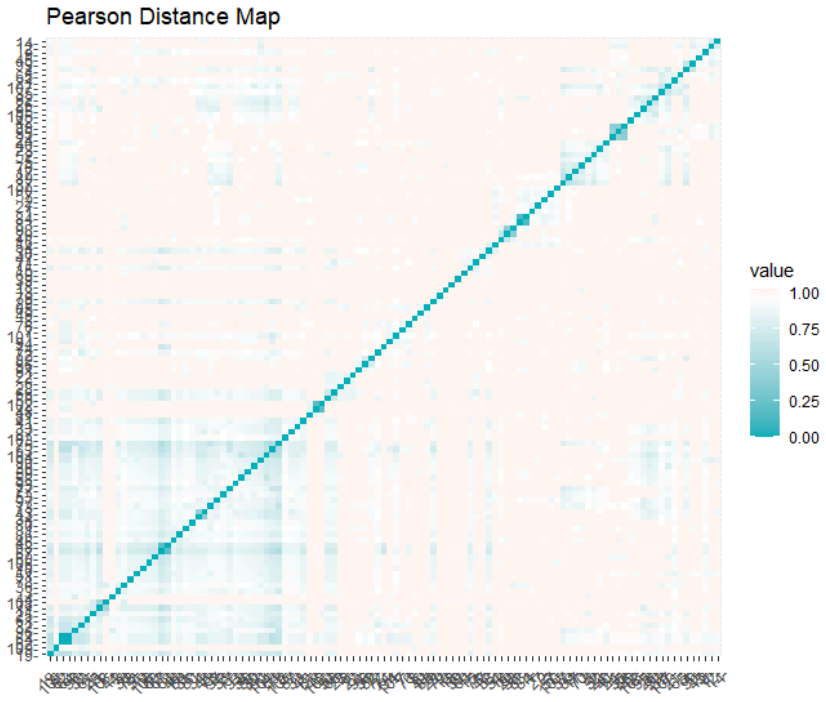


Figure 6. Map of Pearson distance.

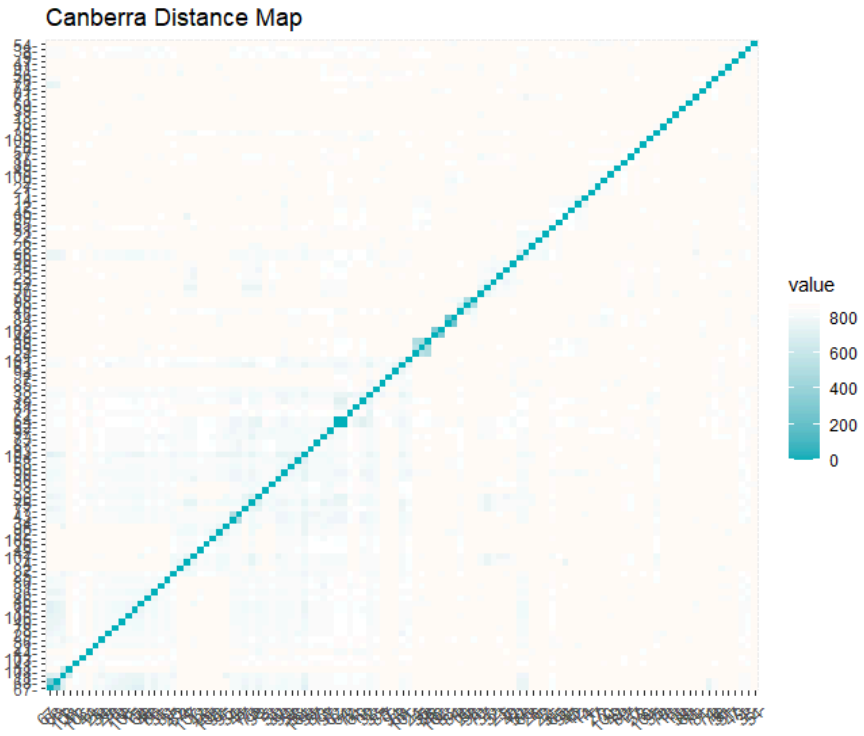


Figure 7. Map of Canberra distance.

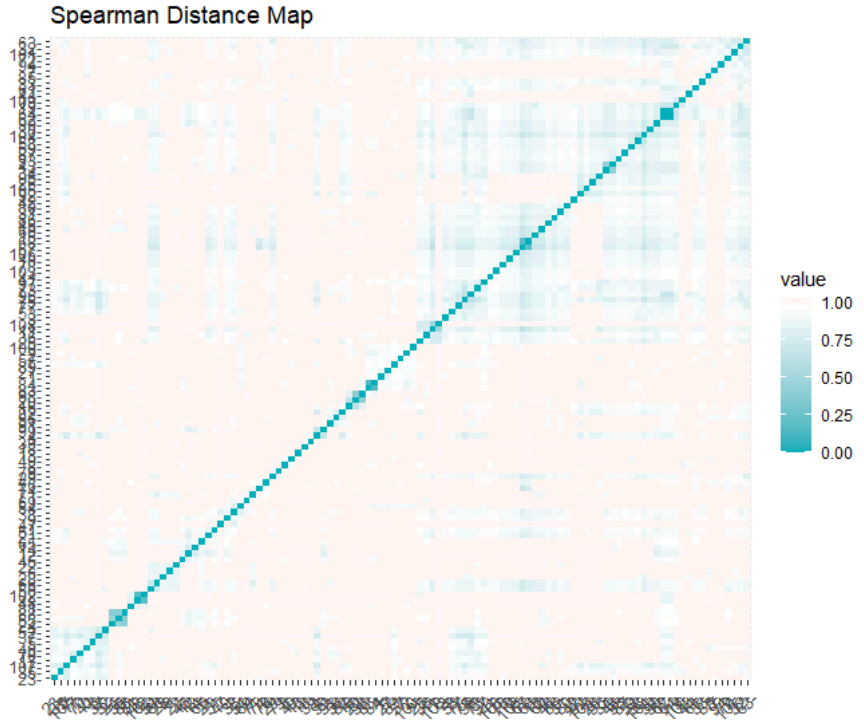


Figure 8. Map of Spearman distance.

Out of the various distances that were used, it appears that only the Manhattan, euclidean and cosine sim normalized distances produced clusters that were meaningful, based on the shading of the plots on the graph. The graphs that were produced are too dense to determine exactly what these associations are and what they are related to. The dataset that was used to perform clustering analysis shows an incredible diversity of words and phrases, and clustering groupings of words into categories did not produce meaningful results. Since only the first 80 columns from the dataset were used to perform this analysis, a larger number of perhaps several million tweets would yield completely different results. Due to computational difficulties, only 80 were able to be used.
K-means clustering was also performed as part of the clustering analysis and was performed with the number of centers varying from 2 to 5. The clustering results are depicted below.

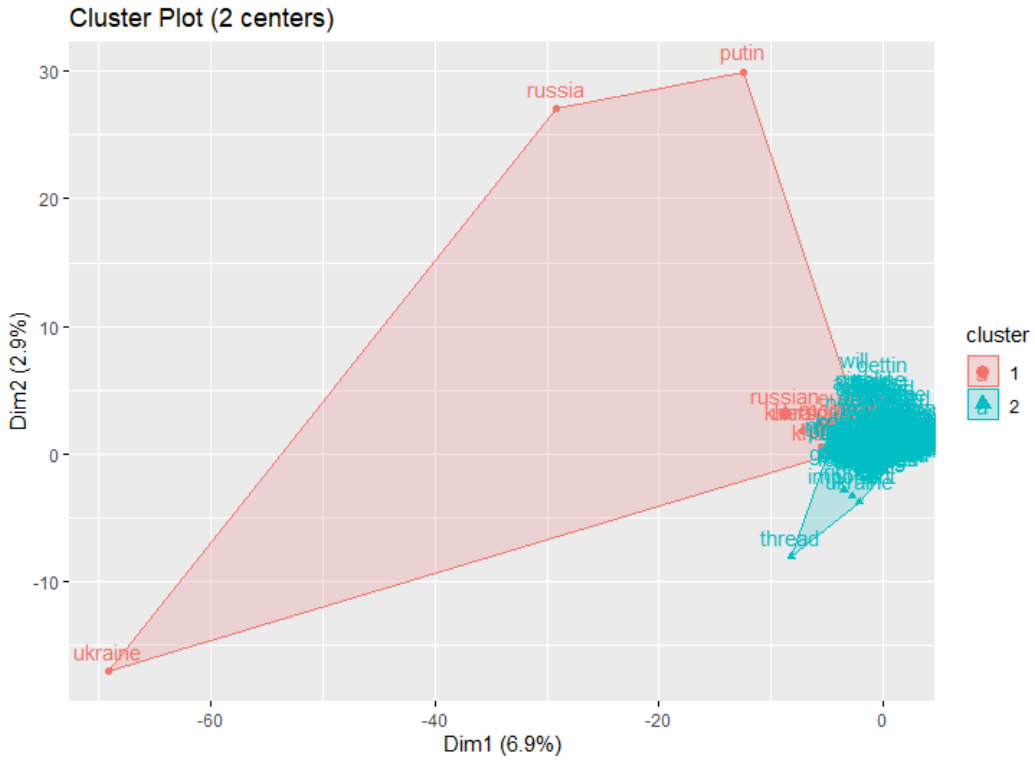


Figure 9. K-means cluster plot with two centers.

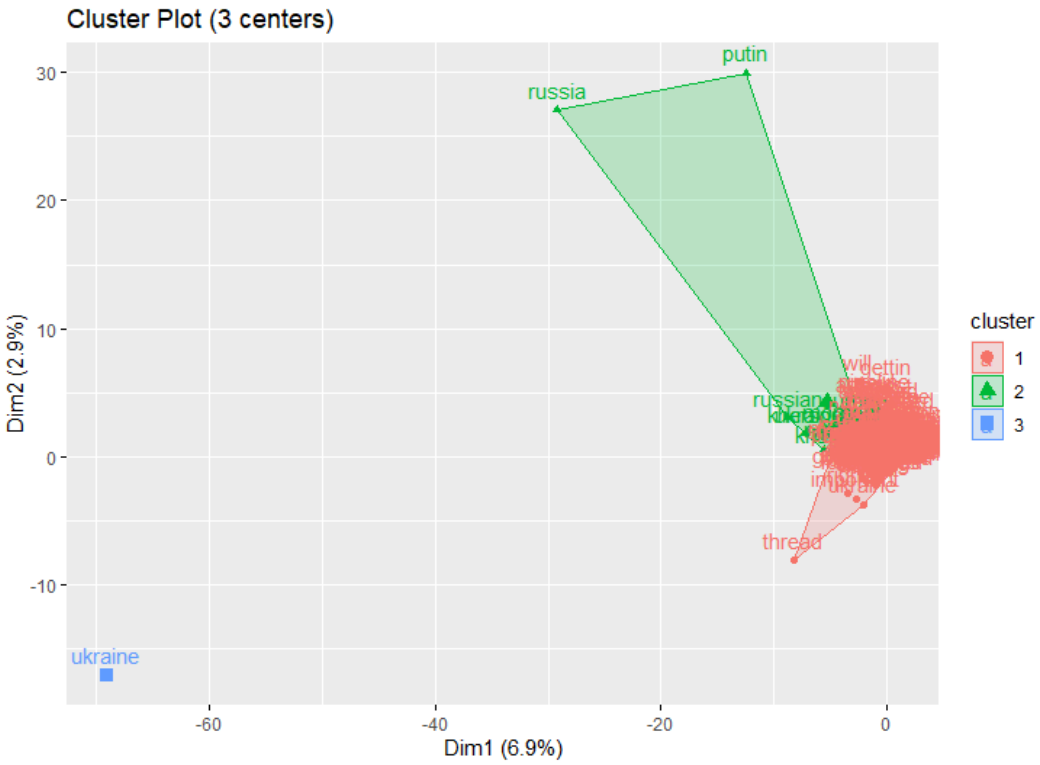


Figure 10. K-means cluster plot with three centers.

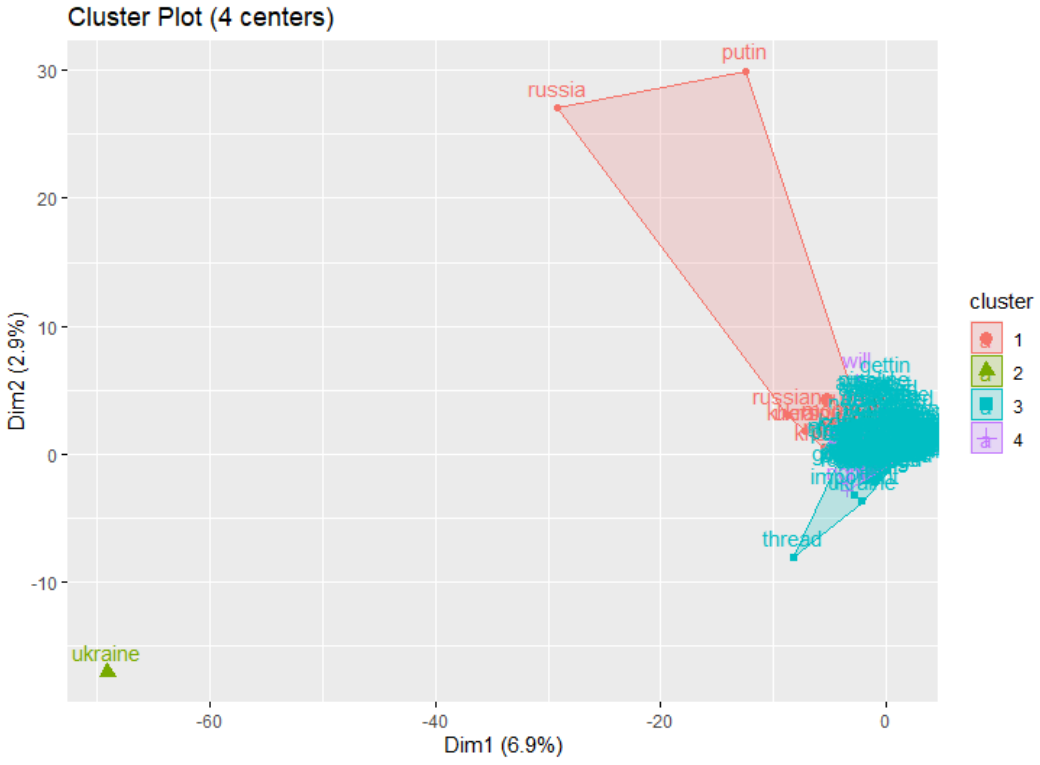


Figure 11. K-means cluster plot with four centers.

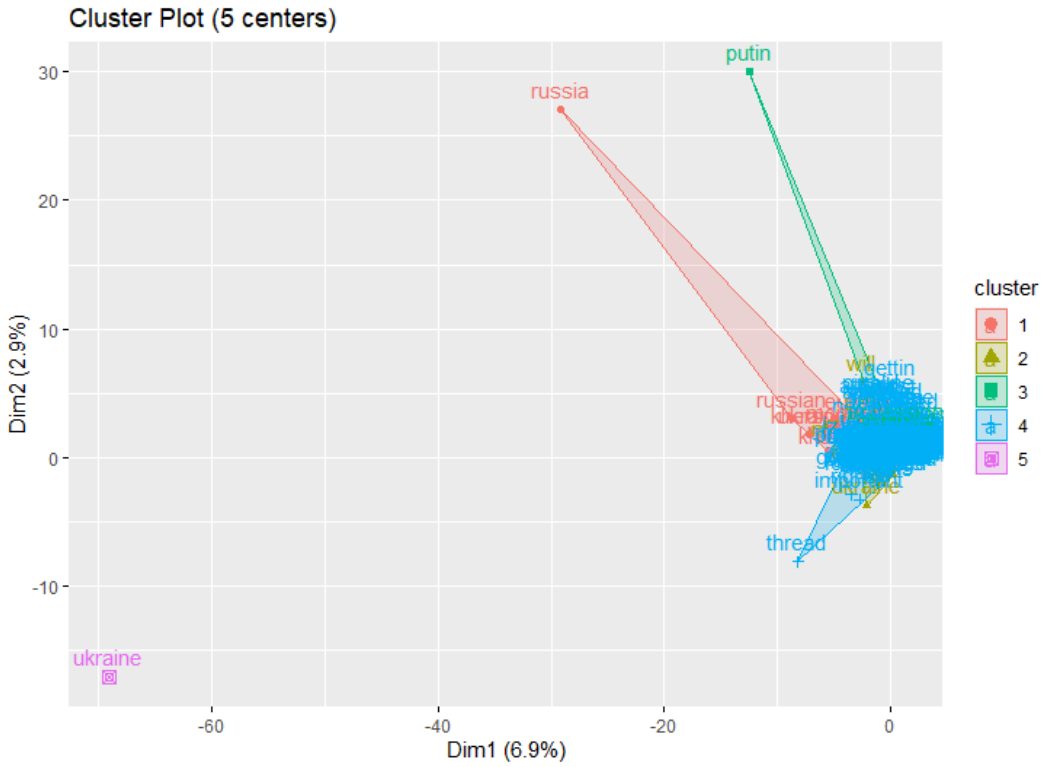


Figure 12. K-means cluster plot with five centers.

K-means clustering using a smaller number of centers tends to produce clearer results. Hierarchical clustering was performed using three different methods: cosine similarity, normalized cosine similarity, and Euclidean. Using cosine similarity and normalized cosine similarity produced identical results. This is most likely due to the relatively small sample size being considered in this study. The results generated using the cosine similarity and the Euclidean metrics are shown below. In both cases hierarchical clustering reveals close associations between different tweets that can be used as the input data for further analysis.

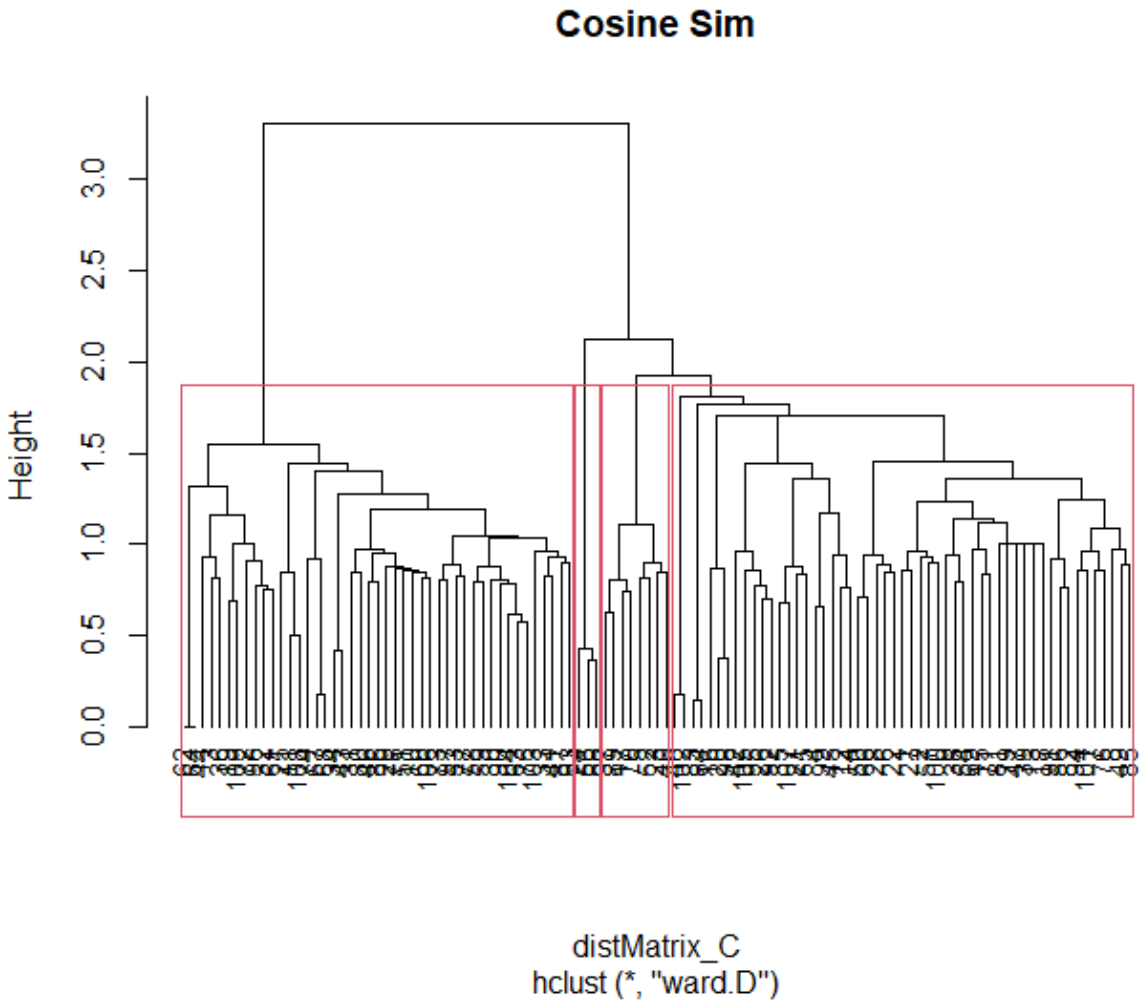


Figure 13. Cosine similarity with hierarchical clustering.

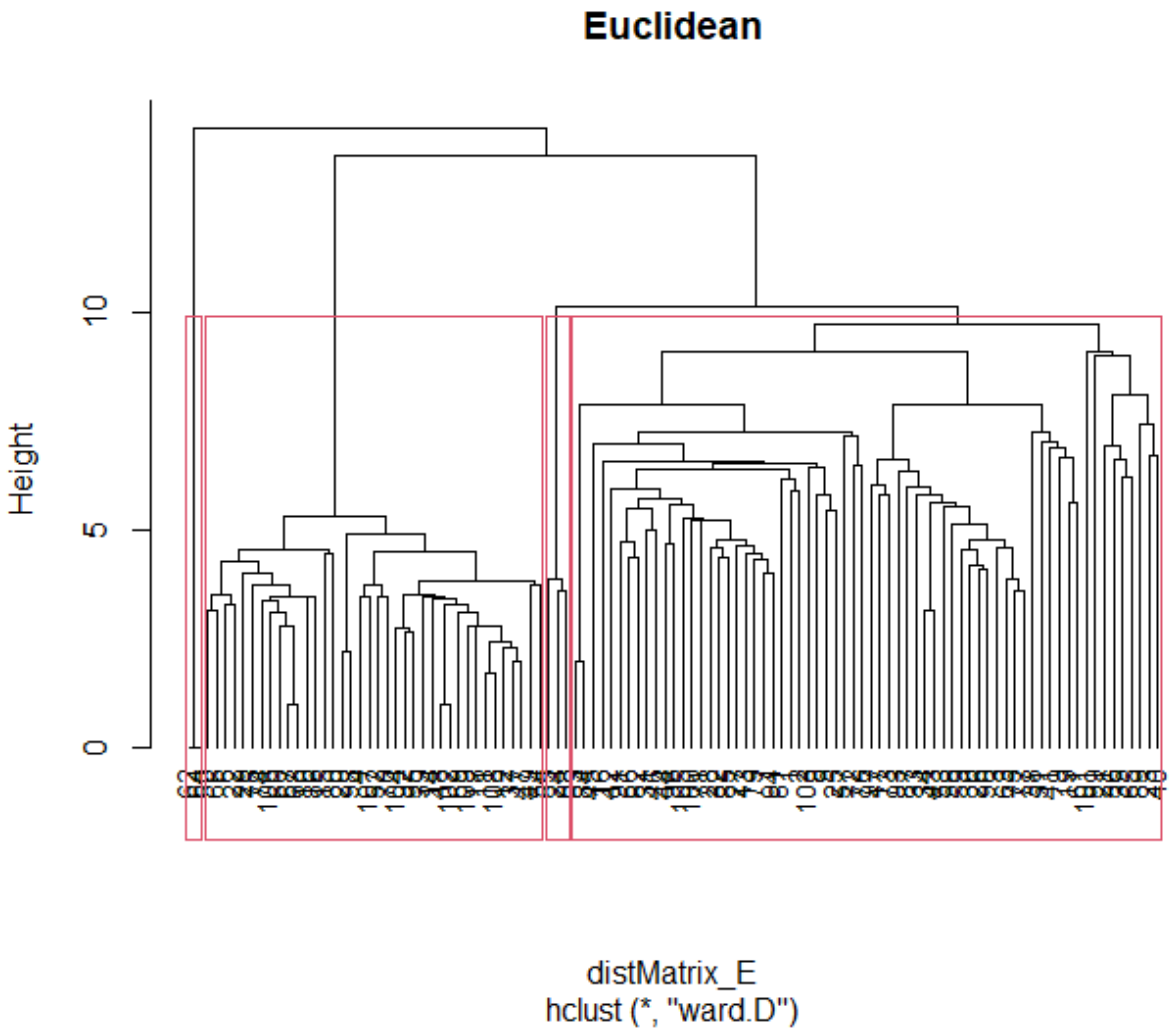


Figure 14. Euclidean distance with hierarchical clustering.

Hierarchical clustering produced various associations between words found in different tweets. The exact associations are not discernable from the graphs that were generated from either the cosine similarity or the euclidean distance graphs. Based on the graphs of the various clusters that were created, none of them contain any relevant or discernable information about the words in the tweets. The text data is incredibly diverse in terms of words and content and no meaningful clusters could be derived from a relatively small sample of the dataset.

3.3 *Association Rule Mining (ARM)*

Lift, confidence and support are three of the most important metrics through which important and meaningful associations between data points can be derived. The top rules derived from the dataset of 80 tweets are represented. See below the results for the top 15 rules for support, the top 15 rules for confidence, and the top 15 rules for lift.

Top 15 rules for support:

```
     lhs                            rhs       support    confidence   coverage
lift count
[1]  {russia}                    => {ukraine} 0.13207547 0.5833333  0.22641509
1.717593 7
[2]  {ukrainerussiawar}          => {ukraine} 0.09433962 0.6250000  0.15094340
1.840278 5
[3]  {now}                       => {ukraine} 0.05660377 1.0000000  0.05660377
2.944444 3
[4]  {right}                     => {ukraine} 0.05660377 1.0000000  0.05660377
2.944444 3
[5]  {ukrainewar}                => {russia}  0.05660377 1.0000000  0.05660377
4.416667 3
[6]  {ukrainewar}                => {ukraine} 0.05660377 1.0000000  0.05660377
2.944444 3
[7]  {west}                      => {putin}   0.05660377 1.0000000  0.05660377
6.625000 3
[8]  {lyman}                     => {russia}  0.05660377 1.0000000  0.05660377
4.416667 3
[9]  {russia, ukrainewar}        => {ukraine} 0.05660377 1.0000000  0.05660377
2.944444 3
[10] {ukraine, ukrainewar}       => {russia}  0.05660377 1.0000000  0.05660377
4.416667 3
[11] {russia, ukrainerussiawar}  => {ukraine} 0.05660377 1.0000000  0.05660377
2.944444 3
[12] {ukraine, ukrainerussiawar} => {russia}  0.05660377 0.6000000  0.09433962
2.650000 3
[13] {俄罗斯}                    => {乌克兰}  0.03773585 1.0000000  0.03773585
26.500000 2
[14] {乌克兰}                    => {俄罗斯}  0.03773585 1.0000000  0.03773585
26.500000 2
[15] {俄罗斯}                    => {中國}    0.03773585 1.0000000  0.03773585
26.500000 2
```

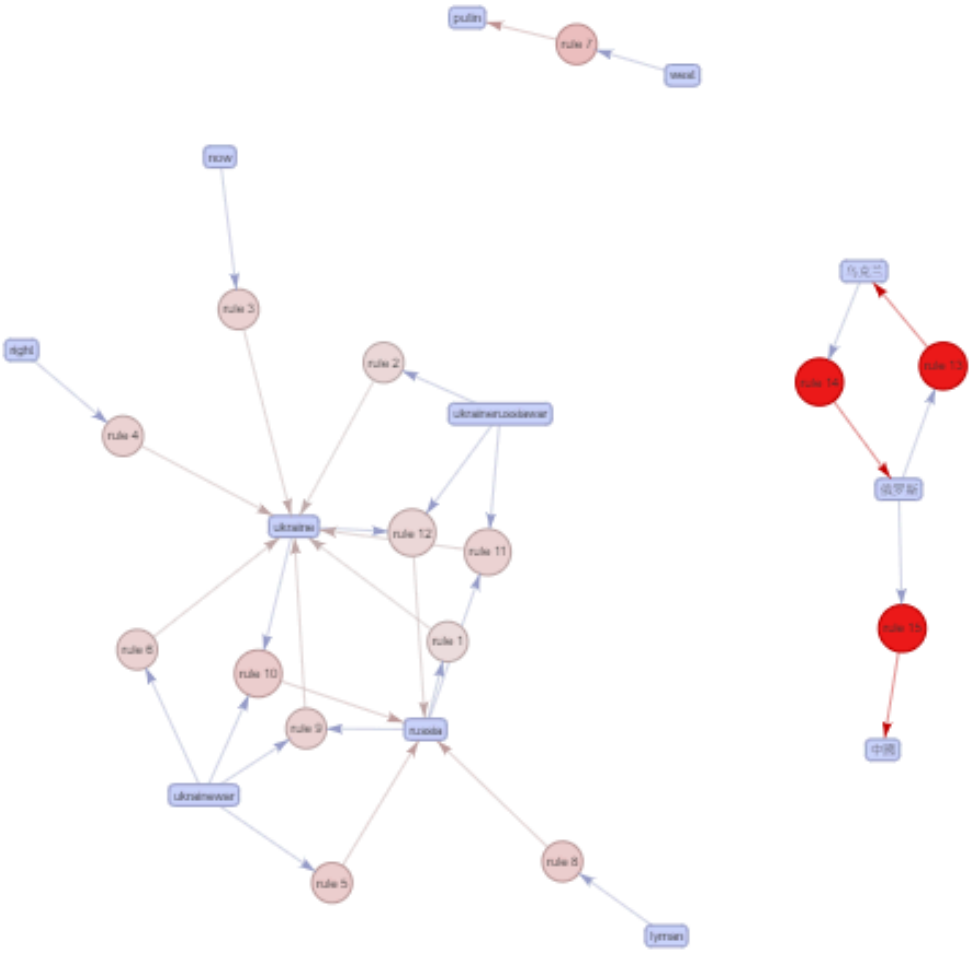


Figure 15. Graph of top 15 rules for support.

The top 15 rules for support have some obvious associations such as russia and ukraine but also some of the less obvious such as lyman and russia, the latter most likely due to the recent ukrainian offensive in the Kharkiv region and the battle for the city of Lyman at around the end of September. The dataset used in this analysis contains data from tweets that were created prior to 10/1/22. Not surprisingly, there is also a strong association between “war” and “putin”, which is seen as a two-node graph in the upper portion of the figure above. The last 3 rules contain non-english characters that must have crept into the data due to the incorrect language encoding of some of the tweets. These associations are depicted as a separate detached graph. These words are in traditional Chinese and can be translated as: [13] {Russia} => {Ukraine}[14] {Ukraine} => {Russia}[15] {Russia} => {China}.

Top 15 rules for confidence:

	lhs		rhs	support	confidence	coverage	lift	count
[1]	{俄罗斯}	=>	{乌克兰}	0.03773585	1	0.03773585	26.500000	2
[2]	{乌克兰}	=>	{俄罗斯}	0.03773585	1	0.03773585	26.500000	2
[3]	{俄罗斯}	=>	{中國}	0.03773585	1	0.03773585	26.500000	2
[4]	{中國}	=>	{俄罗斯}	0.03773585	1	0.03773585	26.500000	2
[5]	{俄罗斯}	=>	{sanktionengegendieusa}	0.03773585	1	0.03773585	26.500000	2
[6]	{sanktionengegendieusa}	=>	{俄罗斯}	0.03773585	1	0.03773585	26.500000	2
[7]	{俄罗斯}	=>	{maga}	0.03773585	1	0.03773585	26.500000	2
[8]	{maga}	=>	{俄罗斯}	0.03773585	1	0.03773585	26.500000	2
[9]	{俄罗斯}	=>	{kriegserklaerung}	0.03773585	1	0.03773585	26.500000	2
[10]	{kriegserklaerung}	=>	{俄罗斯}	0.03773585	1	0.03773585	26.500000	2
[11]	{俄罗斯}	=>	{ukrainewar}	0.03773585	1	0.03773585	17.666667	2
[12]	{俄罗斯}	=>	{russia}	0.03773585	1	0.03773585	4.416667	2
[13]	{俄罗斯}	=>	{ukraine}	0.03773585	1	0.03773585	2.944444	2
[14]	{乌克兰}	=>	{中國}	0.03773585	1	0.03773585	26.500000	2
[15]	{中國}	=>	{乌克兰}	0.03773585	1	0.03773585	26.500000	2
		=>	{中國}	0.03773585	1.0000000	0.03773585	26.500000	2

Top 15 rules for lift:

	lhs		rhs	support	confidence	coverage	lift	count
[1]	{俄罗斯}	=>	{乌克兰}	0.03773585	1	0.03773585	26.5	2
[2]	{乌克兰}	=>	{俄罗斯}	0.03773585	1	0.03773585	26.5	2
[3]	{俄罗斯}	=>	{中國}	0.03773585	1	0.03773585	26.5	2
[4]	{中國}	=>	{俄罗斯}	0.03773585	1	0.03773585	26.5	2
[5]	{俄罗斯}	=>	{sanktionengegendieusa}	0.03773585	1	0.03773585	26.5	2
[6]	{sanktionengegendieusa}	=>	{俄罗斯}	0.03773585	1	0.03773585	26.5	2
[7]	{俄罗斯}	=>	{maga}	0.03773585	1	0.03773585	26.5	2
[8]	{maga}	=>	{俄罗斯}	0.03773585	1	0.03773585	26.5	2
[9]	{俄罗斯}	=>	{kriegserklaerung}	0.03773585	1	0.03773585	26.5	2
[10]	{kriegserklaerung}	=>	{俄罗斯}	0.03773585	1	0.03773585	26.5	2
[11]	{乌克兰}	=>	{中國}	0.03773585	1	0.03773585	26.5	2
[12]	{中國}	=>	{乌克兰}	0.03773585	1	0.03773585	26.5	2
[13]	{乌克兰}	=>	{sanktionengegendieusa}	0.03773585	1	0.03773585	26.5	2
[14]	{sanktionengegendieusa}	=>	{乌克兰}	0.03773585	1	0.03773585	26.5	2
[15]	{乌克兰}	=>	{maga}	0.03773585	1	0.03773585	26.5	2

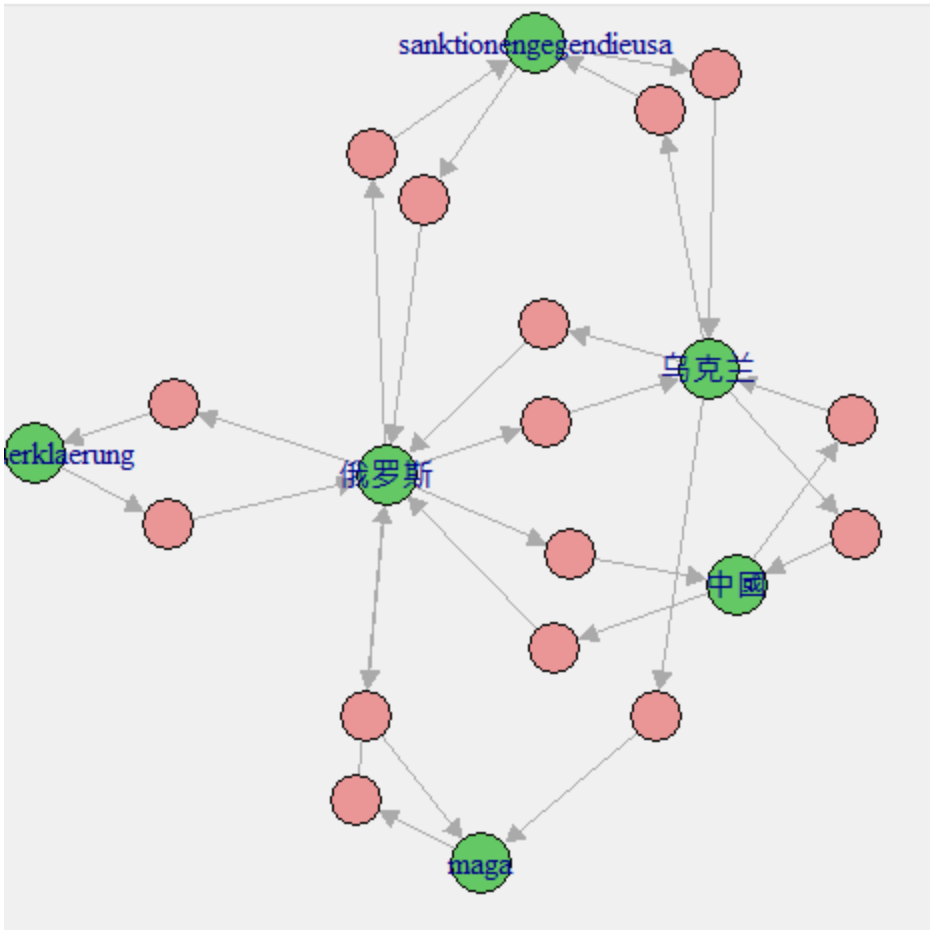


Figure 16. Graph of top 15 rules for lift.

The top rules for confidence and lift are overwhelmed by traditional Chinese text. The translation for the confidence rules is as follows:

[1] {Russia} => {Ukraine}
[2] {Ukraine} => {Russia}
[3] {Russia} => {China}
[4] {China} => {Russia}
[5] {Russia} => {sanktionengegendieusa}
[6] {sanktionengegendieusa} => {Russia}
[7] {Russia} => {maga}
[8] {maga} => {Russia}
[9] {Russia} => {kriegserklaerung}
[10] {kriegserklaerung} => {Russia}
[11] {Russia} => {ukrainewar}
[12] {Russia} => {russia}
[13] {Russia} => {ukraine}

[14] {Ukraine} => {China}
[15] {China} => {Ukraine}

Some German and English words are included in these associations. “Kriegserklaerung” is translated as “declaration of war” and “sanktionen gegen die usa” is translated as “sanctions against the USA.” The top 15 rules for lift also contain this same phenomenon. The translation for the lift rules is as follows:

[1] {Russia} => {Ukraine}
[2] {Ukraine} => {Russia}
[3] {Russia} => {China}
[4] {China} => {Russia}
[5] {Russia} => {sanktionengegendieusa}
[6] {sanktionengegendieusa} => {Russia}
[7] {Russia} => {maga}
[8] {maga} => {Russia}
[9] {Russia} => {kriegserklaerung}
[10] {kriegserklaerung} => {Russia}
[11] {Ukraine} => {China}
[12] {China} => {Ukraine}
[13] {Ukraine} => {sanktionengegendieusa}
[14] {sanktionengegendieusa} => {Ukraine}
[15] {Ukraine} => {maga}

Upon translating the texts, three associations in the confidence and lift stand out in particular. {Russia} => {maga}, {maga} => {Russia}, and {Ukraine} => {maga}. The meaning of this is unclear, as well as the origin of this Chinese text anomaly, and it should’ve been ruled out when only English language text was kept in the data. The meaning of “maga” is also unclear, although it is likely that this is a reference to the 2016 presidential campaign slogan of “Make America Great Again.” This Chinese text anomaly is unusual, and it was completely unexpected that Chinese text would dominate both the top 15 confidence rules and the top 15 lift rules, especially considering that foreign language text should’ve been eliminated from the dataset. The data was cleaned by filtering removing non-English language encoding from a variable in the tweet column data. Upon reinspecting the data, the Chinese language tweets were encoded as English language tweets in the metadata. The reason for this is unclear, although it does present an interesting anomaly that was discovered through ARM methods.

Both the ARM and the clustering methods generated results that were rather inconclusive. The graphs and charts that were created did not generate any meaningful

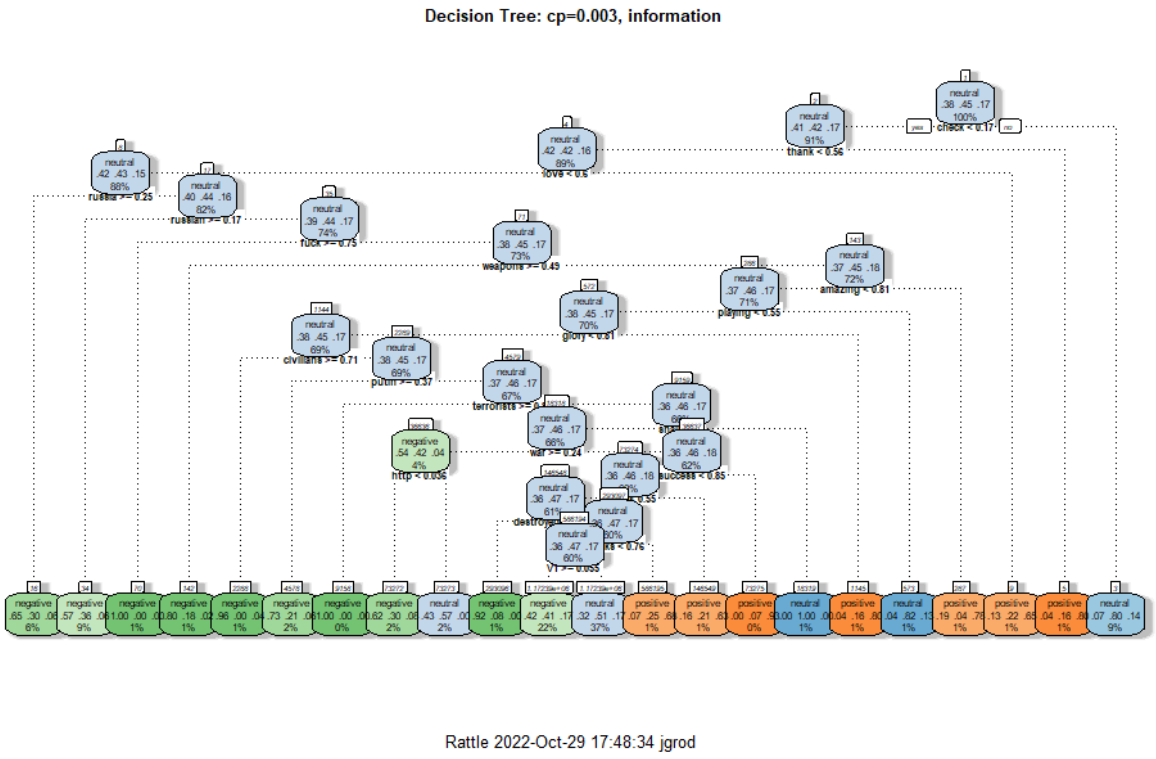
rules that would imply any significant associations between words that have any significance. Among the most common associations between words are: {russia} => {ukraine}, {ukrainerussiawar} => {ukraine}, {russia, ukrainewar} => {ukraine}. These associations by themselves do not imply anything significant that could be of use in discovering potential sources of misinformation about the war. Since these associations are commonly found throughout the tweets, they can be disregarded in future analyses. What this says is that "Russia" and "Ukraine" are words that are commonly found together in tweets pertaining to the war, a fact which has no useful meaning in of itself.

As an interesting aside, some of the most frequent associations that were found were in traditional Chinese, even though foreign language tweets that were not in English were supposed to have been ruled out from the datasets. The samples from the Kaggle dataset were drawn consecutively (the first 80 entries) and a large amount of these tweets were mislabeled with regards to their language status. This implies that the datasets will need further cleaning before they can be rerun for the same analyses. Three associations between the words "Russia" and "maga" and "Ukraine" and "maga" were discovered in the Chinese text. The meaning of this is unclear, although it is possible that "maga" refers to the 2016 presidential campaign slogan of "Make America Great Again." These associations raised many questions, and a further investigation of tweet hashtags and metadata was conducted as a result of these findings.

The biggest limitation in the analyses was the small size of the dataset (around 100 to 200 tweets from both the Twitter API and the Kaggle dataset). But even with the small sample of data, obvious associations were found between words, and these associations were ruled out when conducting future research. The results likely would've been improved with access to software that can handle a much larger dataset, where the same methods can be used to generate rules and associations between words. Once associations can be established, sentiment analysis can be used to find both positive and negative sentiments between different words on a variety of subjects pertaining to the war. For example, positive sentiments about Russia or negative sentiments about Ukraine can be discovered using sentiment analysis on a large dataset, and these sentiments can imply that there are certain sources of disinformation about the war. A deeper investigation of these sentiments can be used to gather information about user accounts, user locations, and other user information which can be used to further assess the nature of these tweets and whether or not they are potential sources of misinformation or disinformation. Accessing foreign languages such as Russian, Chinese, and Ukrainian could significantly increase the amount of significant associations that are discovered between words that are pertinent to misinformation / disinformation, and further research must be done to see whether there are tools that could be used to translate foreign language terms into English within the dataset.

Since the results of this analysis were inconclusive, no meaningful conclusions can be drawn about discovering potential types or sources of misinformation / disinformation about Twitter content pertaining to the War in Ukraine. The same analyses will need to be redone in different software that can handle larger datasets. Once these larger datasets are used, a much clearer picture will be established with regards to the associations between words.

3.4 *Decision Trees*

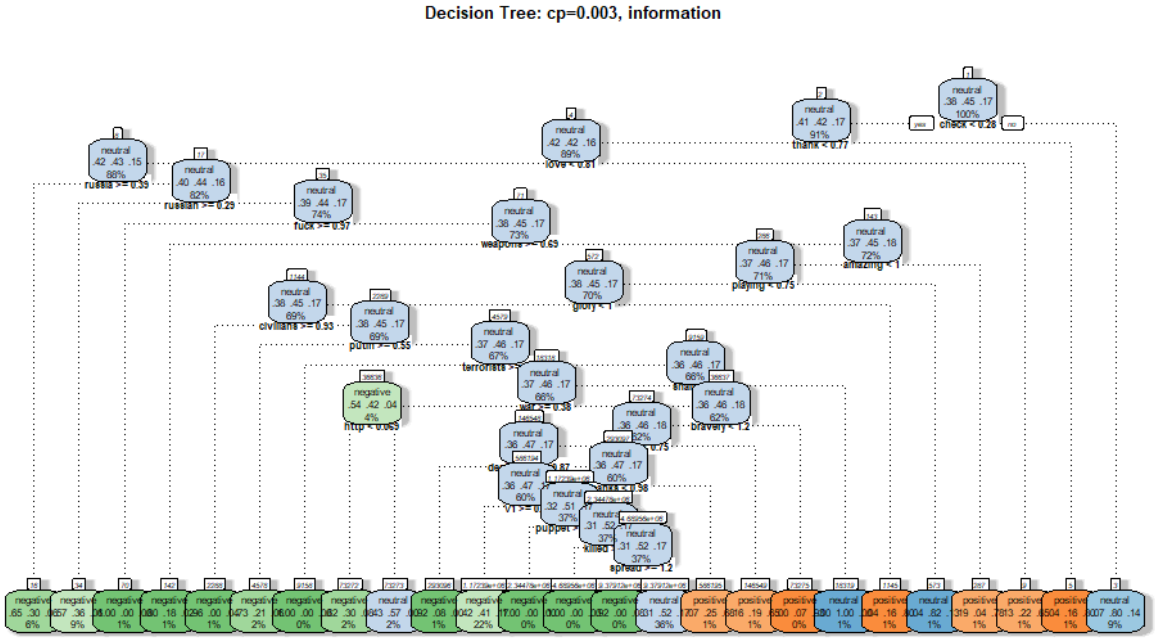


Confusion Matrix

Prediction	negative	neutral	positive
negative	63	35	8
neutral	34	72	25
positive	2	2	9

Accuracy 0.576

Figure 4. Decision tree made from dataset with 5,000 tweets and 300 features, entropy / information gain adjusted. Confusion matrix and accuracy.

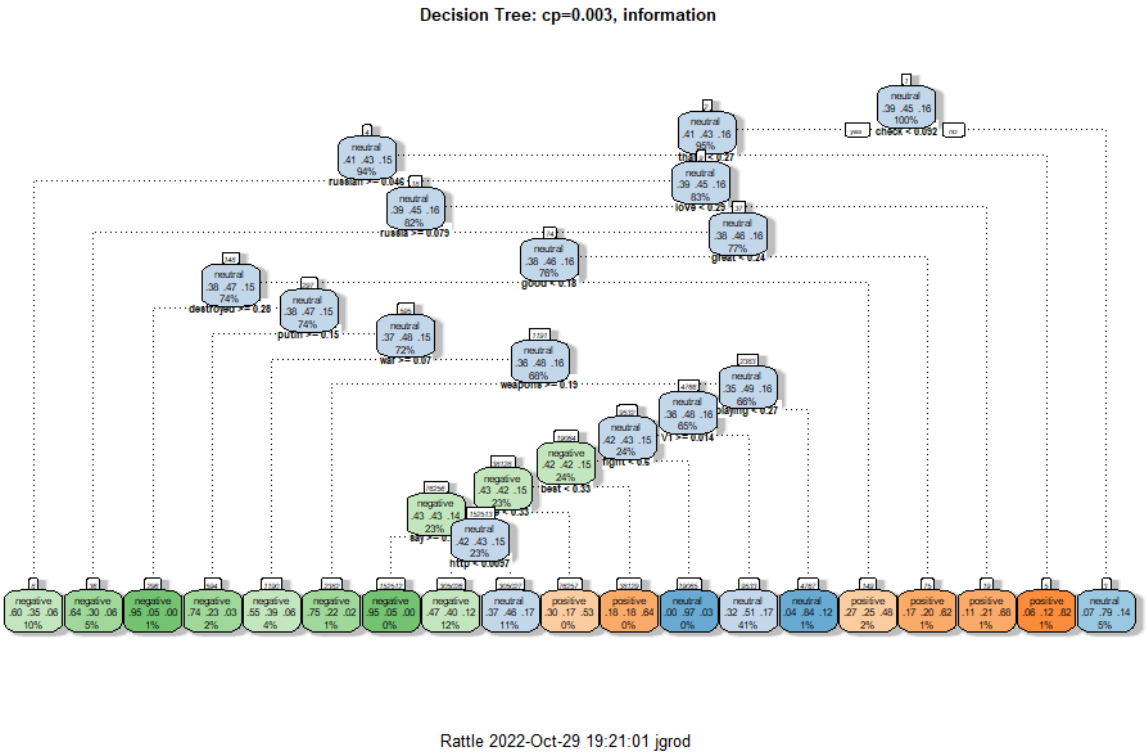


Confusion Matrix

Prediction	negative	neutral	positive
negative	63	35	8
neutral	34	72	25
positive	2	2	9

Accuracy 0.576

Figure 5. Decision tree made from dataset with 5,000 tweets and 600 features, entropy / information gain adjusted. Confusion matrix and accuracy.



Confusion Matrix

Prediction	negative	neutral	positive
negative	59	34	11
neutral	39	73	26
positive	1	2	5

Accuracy 0.548

Figure 6. Decision tree made from dataset with 5,000 tweets and 150 features, entropy / information gain adjusted. Confusion matrix and accuracy.

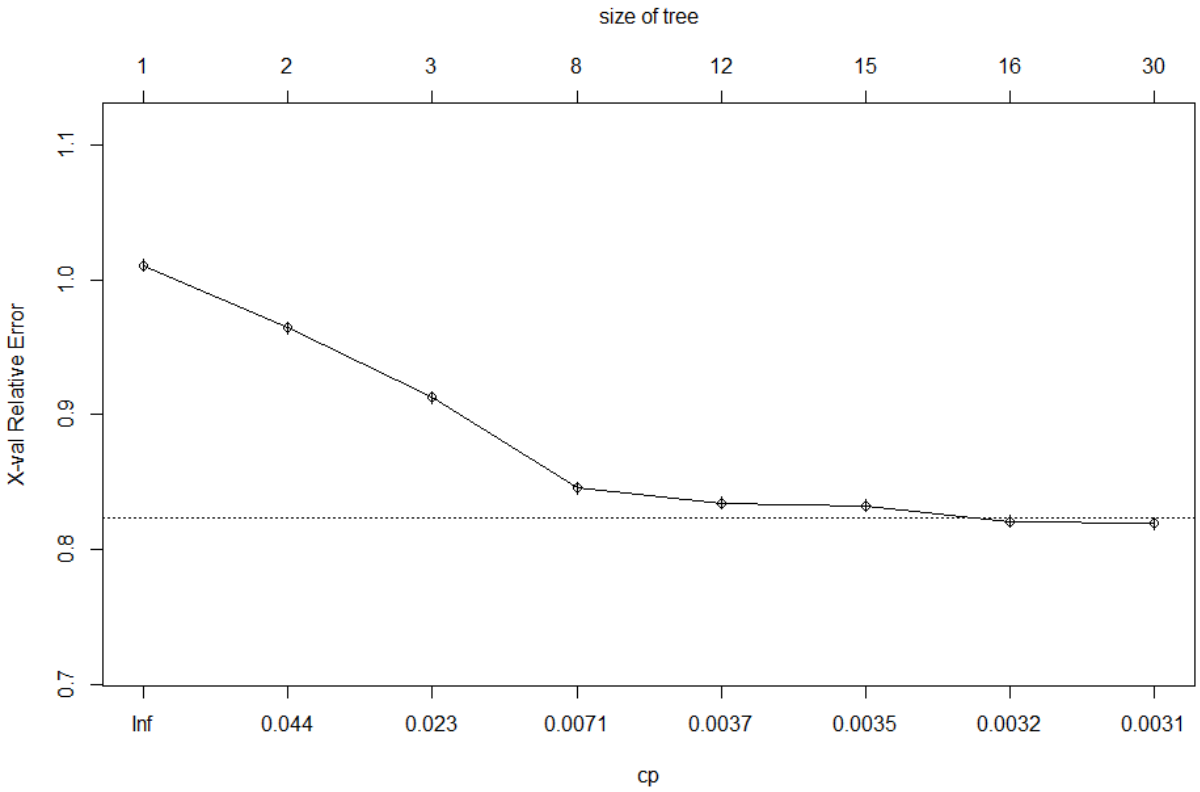


Figure 7. Correlation between value of complexity parameter, relative error and the size of the tree. Data used from decision tree made from dataset with 5,000 tweets and 300 features, entropy / information gain adjusted. For most of the graphs that were made for this comparison, the optimal cp was around 0.003 where the relative error stabilized.

The results for the three most accurate decision trees are illustrated, with the confusion matrices included. The ideal complexity parameter (cp) was chosen to be 0.003 for all of the decision trees that were generated. Decision trees were generated with varying amounts of tweets and features. The most relevant and popular words from the tweet dataset were chosen and classified according to the sentiment that was attributed to them. There are still some non-useful words such as “http” in the decision tree, which indicates that there is still some work to be done in the data cleaning and preparation process. Popular words such as “Russia” , “Russia” , and “Ukraine” were categorized according to the sentiment attributed to them.

The decision trees where GINI was a factor produced slightly less accurate results (by about 1/10 of a percent) than in the trees where entropy / information gain was adjusted. Discretization produced a data tree with two levels, with nonsensical results. The origins of this anomaly were not determined and are possibly due to a source of error in the code. There is a skewed distribution in the numerical values in the feature matrix of tweets vs. features, which might be a possible explanation. The matrix of tweets vs.

features is sparse and populated largely by 0's, which might explain the uninterpretable data that was generated.

3.5 *Naive Bayes*

Confusion Matrix

NB_Pred	negative	neutral	positive
negative	314	136	31
neutral	169	298	35
positive	313	452	252

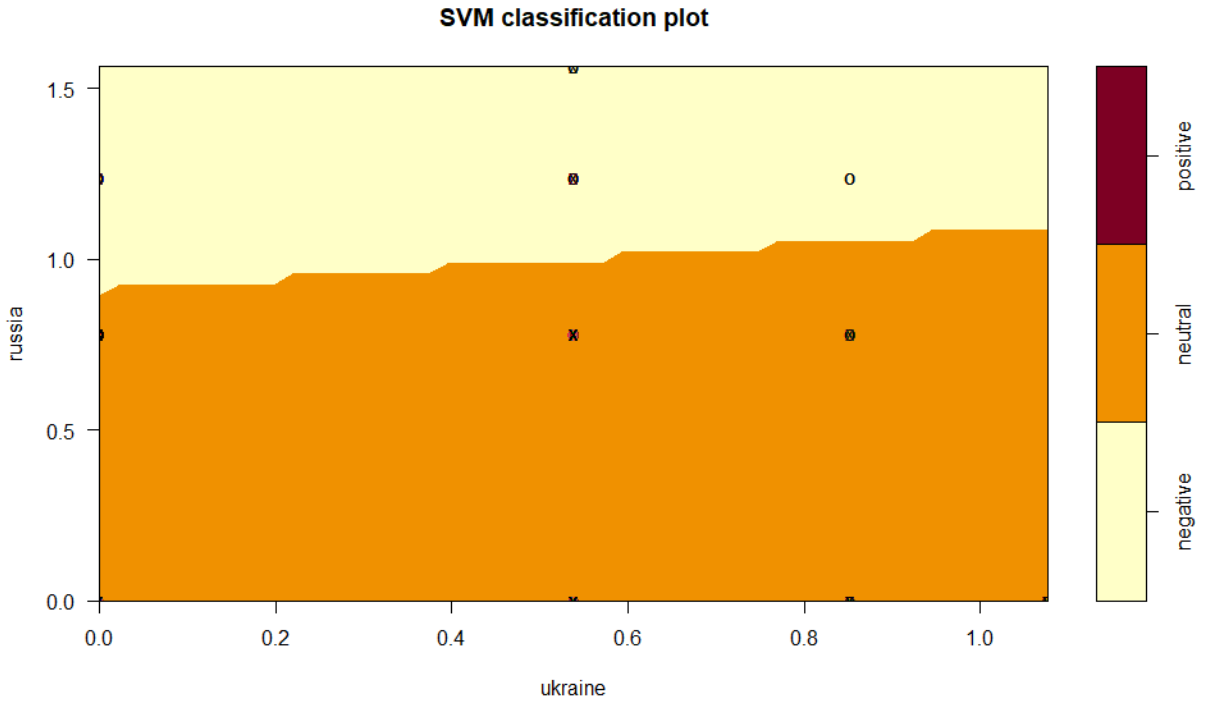
Accuracy = 0.432

Figure 8. Confusion matrix and accuracy for a naive bayes comparison generated from dataset of 40,000 tweets using 150 features.

The naive bayes method was run on a total of 16 tries, with various adjustments made to the size of the data set and the amount of features that were analyzed. The confusion matrix and the accuracy are shown for the most accurate combination of 40,000 tweets and 150 features which resulted in an accuracy of 0.432. This is a relatively low rate of accuracy when compared to the other two supervised learning methods (decision trees and SVMs), where the accuracy with the same adjustments was between 0.5 and 0.7. Overall, this method uses a less sophisticated algorithm and is less accurate when there are complex decision boundary conditions.

3.6 *Support Vector Machines (SVM)*

This section shows graphs and representations of sentiment analysis run on the dataset through SVM's and the other two supervised learning methods, followed by an analysis and explanation for each method.

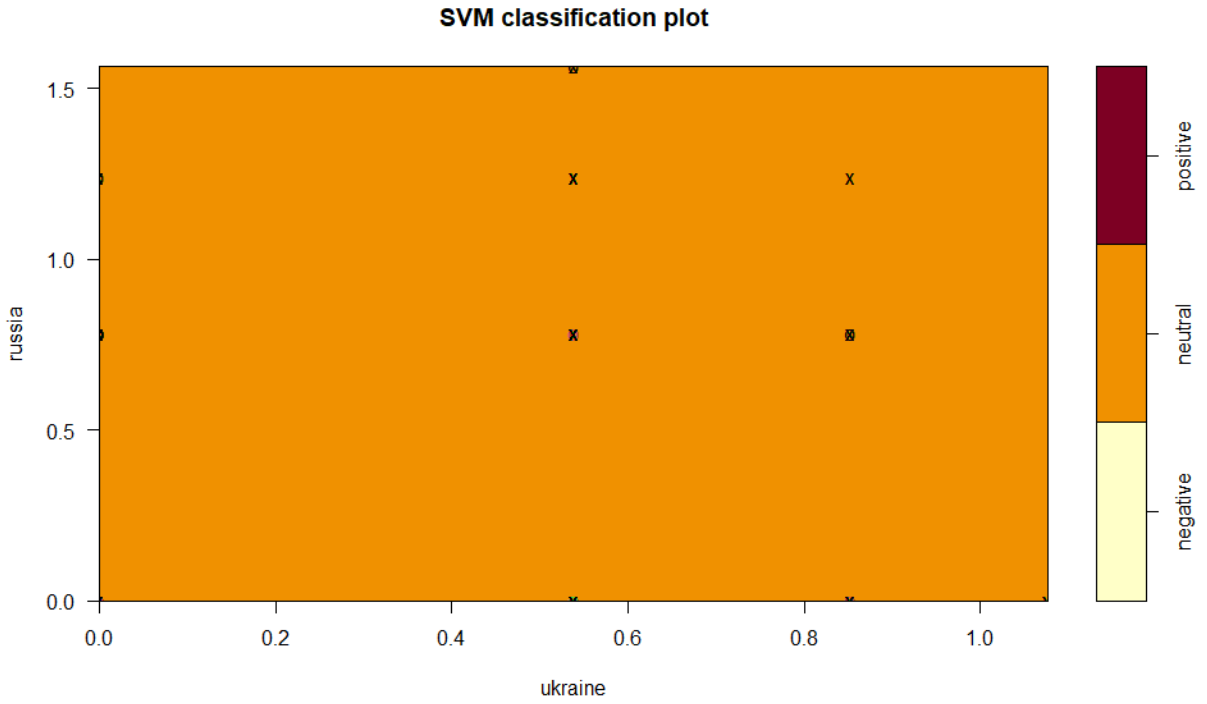


Confusion Matrix

pred_svm	negative	neutral	positive
negative	60	13	4
neutral	32	88	16
positive	7	8	22

Accuracy 0.68

Figure 9. SVM figure, confusion matrix and accuracy for a linear SVM comparison, cost = 1.

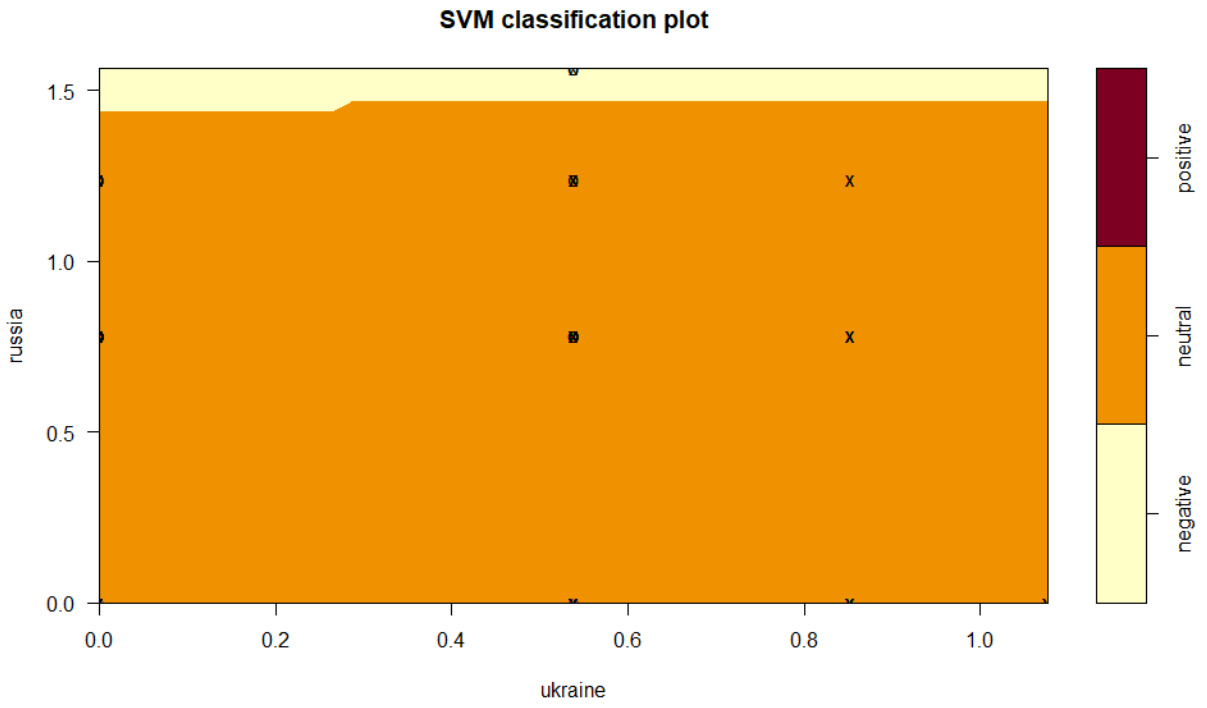


Confusion Matrix

pred_svm	negative	neutral	positive
negative	19	1	0
neutral	80	108	37
positive	0	0	5

Accuracy 0.528

Figure 10. SVM figure, confusion matrix and accuracy for a polynomial SVM comparison, cost = 1.



Confusion Matrix

pred_svm	negative	neutral	positive
negative	71	18	12
neutral	28	87	19
positive	0	4	11

Accuracy 0.676

Figure 11. SVM figure, confusion matrix and accuracy for a radial SVM comparison, cost = 1.

A cost of 1 gave the most accurate results, when compared to a cost of 0.01, 0.1, 1, 10, 100. As such, a cost of 1 was used in all of the graph generations. The linear SVM was the most accurate, to a small degree more accurate than the radial SVM. The polynomial SVM method was the most inaccurate, and categorized a large amount of information as neutral compared to the other sentiments.

The three supervised learning methods produced results of varying degrees of accuracy when comparing the training sets to the test sets. Overall the SVM methods and the decision trees produced the most accurate results, with naive bayes predictions being significantly less accurate at sentiment analysis than the other two. In both the decision trees and the SVM, the rates of accuracy were higher when using a smaller dataset of 5,000 tweets when compared to larger datasets going up incrementally from 5,000 tweets to 80,000 tweets. Increasing the amount of features that were analyzed also had an effect on increasing the accuracy of the decision tree and SVM methods. In naive bayes, increasing the amount of features actually led to a significant decrease in accuracy. Increasing the amount of tweets increased the accuracy and overall large datasets with a small amount of features produced the most accurate predictions when run through the naive bayes algorithm. There were computational limits that prevented a higher amount of tweets or a higher amount of features from being run using these methods. But overall, running the individual tweet keywords through the supervised learning methods helped determine which methods were the most effective and the amount of features / the size of the tweet dataset that produced the most accurate predictions.

Decision Tree (GINI) - Accuracy					
# of features # of tweets	150	300	600	1200	2400
5000	0.52	0.568	0.568	0.568	0.568
10000	0.536	0.542	0.55	0.55	
20000	0.516	0.522	0.521		
40000	0.514	0.515	0.515		
80000	0.517	X	X		

Decision Tree (information) - Accuracy					
# of features # of tweets	150	300	600	1200	2400
5000	0.548	0.576	0.576	0.588	0.576
10000	0.52	0.54	0.552	0.552	
20000	0.516	0.524	0.527		
40000	0.512	0.536	X		
80000	0.512	X	X		

Naïve Bayes - Accuracy					
# of features # of tweets	150	300	600	1200	2400
5000	0.372	0.32	0.248	0.168	0.168
10000	0.392	0.342	0.298	0.174	
20000	0.426	0.422	0.388		
40000	0.432	0.412	0.416		
80000	0.43	X	X		
SVM - Accuracy					
# of features # of tweets	150	300	600	1200	2400
5000	0.588	0.64	0.676	0.692	0.696
10000	0.598	0.628	0.688	0.69	
20000	0.594	0.624	0.66		
40000	0.604	0.634	0.672		
80000	0.604	X	X		

Figure 12. A chart illustrating the accuracy of the three supervised learning models. The amount of features used are on the column variable and the size of the dataset used (amount of tweets) is on the row variable. Decimal number between 0 and 1 shows the percentage of accuracy between the training set and the testing set. The SVM uses the radial kernel and the cost is equal to 1.

Supervised learning methods were used to classify individual words from a large dataset of tweets with information pertaining to the war in Ukraine. The most frequently occurring words found in this large corpus of tweets were categorized by their importance and sentiment analysis was run by three supervised learning methods (decision trees, naive bayes, and SVM). These frequently occurring words were categorized according to sentiment (positive, negative, neutral). This process helped categorize keywords found in the tweets by labeling them according to their sentiment. A comparison of these three supervised learning methods showed that each method produced results of varying accuracy when comparing the training sets to the testing sets. Overall, SVMs and decision trees produced the most accurate results in this process.

The categorization of words was quite predictable, given the overall public sentiment about the war for a majority of Twitter users writing about the topic. Some of the most popular keywords such as “Russia” and “Putin” were labeled with an overwhelming negative sentiment, while keywords such as “glory”, “slava” and “Ukraine” were labeled with an overwhelming positive sentiment. These results were expected before the supervised learning methods were run, and these results in of themselves do not reveal much about accounts that could be spreading potential disinformation. The discovery and labeling of keywords from a dataset of over one million recent tweets pertaining to the war however, is an important step in this process.

In the association rule mining phase in the analysis, an anomaly was found where a set of around eighty tweets was mislabeled by language causing Chinese text to seep into the dataset after it was run through the English-only text filter. Although this anomaly was not confirmed as a source of misinformation, a deeper dive into tweet metadata might yield interesting information that could help point to disinformation. An analysis of these tweets by account and user location could possibly lead to more discoveries, such as whether large amounts of these tweets are being sent from a specific geographic location or whether there is a set of accounts that is creating and retweeting similar posts.

3.7 *Neural Networks*

GA	GB	GC	GD	GE	GF	GG	GH	GI	GJ	GK	GL	GM	GN	GO	GP	GQ	GR	GS
using	via	victory	video	want	war	watch	way	weapons	well	west	western	win	world	years	zelensky	X....	X.....1	Sentiment
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	neutral
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	neutral
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	neutral
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	negative
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	negative
0	0	0	0	0	0.3477264715	0	0	0	0	0	0	0	0	0	0	0	0.5079989107	positive
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	negative
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	neutral
0	0	0	0	0	0	0	0	0	0	0	0	0	0.603612221	0	0	0	0	negative
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	negative
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	neutral
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	negative
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	negative
0	0	0	0	0	0.3477264715	0	0	0	0	0	0	0	0	0	0	0	0	neutral
0	0	0	0	1.016387405	0.3477264715	0	0	0	0.8273225202	0	0	1.187346281	0	0	0	0	0	negative
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	negative
1.3370535949	0	0	0	0	0.3477264715	0	0	0	0	0	0	0	0	0	0	0	0	negative

A sample of the training data. Each tweet is a row and tweet data is labeled by sentiment as 1 or 0, with TF-IDF scores for each individual word. The sentiment of each tweet is labeled in the last column.

Predicted Label	True Label
[1]	[1]
[0]	[1]
[0]	[1]
[0]	[0]
[1]	[0]
[1]	[1]
[1]	[0]
[1]	[1]
[1]	[0]
[1]	[0]
[1]	[1]
[1]	[0]
[1]	[0]
[0]	[1]
[0]	[0]
[1]	[0]
[0]	[0]
[1]	[1]
[1]	[0]
[0]	[1]
[0]	[0]
[1]	[0]
[1]	[1]
[1]	[0]
[1]	[1]
[1]	[0]
[1]	[0]
[0]	[0]
[1]	[0]

Results of the test data after passing through the neural network. Differences between predicted sentiment label and actual sentiment label for the tweet are listed for all 250 tweets in the test dataset.

Confusion Matrix:

[[53 32]
 [46 119]]

Sentiment labels (positive, neutral, and negative) with 0 being a positive or neutral sentiment, while 1 being a negative sentiment, were predicted for the test set of 250 tweets. Overall, the accuracy was high with a test loss of 78 and a test average loss of 0.312. This shows that the neural network can accurately predict the overall sentiment of a tweet. The accuracy rate could be significantly optimized with a larger training set. Considering that the Twitter API offers access to millions of tweets on the subject pertaining to the project, the accuracy can be significantly improved when test data is run against an expanded training set. Additional hidden layers can also help fine-tune the rate of accuracy. The sentiment by itself does not imply much about whether a given text is disinformation or not, but it can help direct towards data that can potentially be disinformation.

Confusion Matrix:

[[62 29]
[37 122]]

The same sentiment analysis was run on the test dataset using an SVM in an earlier phase of the project. The accuracy and confusion matrix were both similar to those generated by the simple neural network.

3.8 *Hashtag Analysis*

Hashtag analysis led to the most significant findings, where tweets containing actual disinformation about the war in Ukraine were discovered by filtering the tweets by hashtags. Initial findings based on the words and the sentiment attached to them yielded results that generally did not point to any sources of misinformation. Methods such as generating data trees, association rule mining (ARM), support vector machines (SVM) and neural networks (NN) did help to significantly narrow down the pool of data and brought light to which words were frequently showing up in the tweets and what sentiment they had attached to them. The accidental finding of the mislabeled language metadata in a subset of the tweets posed a hypothesis that there might be more interesting data to be found in the labels and metadata attached to each tweet. Since this particular set of tweets had a missing location label, an analysis was conducted on hashtags and location labels.

Using a large subset of over a million tweets available from the dataset, a roBERTa sentiment analysis was performed on the hashtag data in order to determine which words in the tweet had the most negative sentiment attached to them. A manual inspection of these negatively associated hashtags and the text data of the tweets they were attached to led to a more detailed analysis. A total of 15 hashtags with potential disinformation were chosen for further analysis: 'MAGA' , 'maga', 'MAGAForever', 'NaziUkraine', 'ukrainiannazi', 'ZelenskyWarCriminal', 'MaidanCoup', 'ProxyWar', 'Fascism', 'Nazism', 'Burisma', 'Corruption', 'Corrupt', 'MySonHunter','AmericaFirst'.

Disinformation

A subset of all the tweets containing these hashtags was generated, along with a list of unique locations attached to these tweets. The list of hashtags was further expanded to 6277 “pro-Russia” hashtags which were identified as being associated with the tweets containing 15 hashtags tied to potential disinformation. The location data that was generated using a subset of the 15 hashtags tied to potential disinformation yielded some surprising results. A very large subset of these tweets had location data that was intentionally mislabeled with misleading or garbage locations. Some of the most notable “locations” that were found include: “Somwher in ze midle of nowhere”, “Anti-Social Media Jail”, “Behind the enemy lines”, “People's Republic of Albonesia”, “ ‘Merica” , “Quahog” , “Free Moskow” , “Planet Clown World”, “Back from the Big Tech Gulag”, “Right behind you pedo. RUN.” among other nonsensical locations. Some location tags were directly attributed to locations in Russia or the occupied territories of Ukraine including “Санкт-Петербург, Россия” (St. Petersburg, Russia), “🇺🇦 Севастополь 🇺🇦” (Sevastopol, occupied Crimea), “Moscow, Russia” , “Lugansk, LPR” (occupied territory of Ukraine). The text data that these hashtags were attached to were blatant disinformation and pro-Russian propaganda, in many cases. Examples of meaningful tweets that were found include:

@zelenskyyua talks about freedom , while hes acts like a #dictator in #ukraine . #zelenskywarcriminal #humanrightsviolations #nuremberg2now #crimesagainsthumanity .

🔞🔞🔞 the whole "adequate" world supports this. everyone who supports zelensky and his punishers must be destroyed. #russia #ukrainewar #русија #usa #europe #ukrainenazi #zelenskywarcriminal #bidenwarcriminal #stopzelensky #stopbiden

raped women, disappeared young women, death, mental trauma by the fascists of #ukraine. #war #russia #ukrainerussiawar #nazi #naziukraine #usa #cia #pentagon#nato n #eu r #america|n #warcrimes #warcrimesofukraine

#zelenskywarcriminal was forced upon #ukraine in a us backed coup. entire opposition along with many journalists were put in jail. no one asked why the opposition political leaders are still in jail?

president zelenskyy is a fascist dictator and neo nazi that must be stopped #zelenskywarcriminal #russianukrainianwar

@dmytrokuleba the same fake photo that #zelenskywarcriminal posted yesterday.we have a recent video of patrick lancaster,and not only his,speaking w/ residents of #izyum.surprisingly,they want to joint to savior #russia & none of them told about crimes,except for #ukraine shelling their homes.

Image borrowed from: https://www.ft.com/content/9efbeac4-8dde-4e51-bb47-d4ae9d504d92

4. Conclusion

4.1 *Initial Findings and Challenges*

The initial phase of data analysis resulted in findings that were inconclusive and generally uninteresting. Given the massive amount of tweet data that is available on the subject, frequent words had to be singled out and labeled in order to perform sentiment analysis on words and phrases that carry some kind of meaning related to the war in Ukraine. Frequently occurring nouns and articles such as “the”, “week” , “at” had to be labeled and removed due to the lack of meaning that they have attached to them, while frequently occurring words related to the subject such as “war” , “Ukraine” and “Putin” had to be labeled and noted as well, since they generally don’t contain any relevant information directly associated with them. The sheer quantity of unique words found in a dataset of several million tweets presented a logistical challenge in looking through each individual word and determining whether it was attached to any potentially interesting information. Rare words with a small count of occurrences in the tweet dataset had to be removed, while the search for meaningful words related to the subject was largely done by hand. In the initial phases of the analysis, some of the findings were largely accidental, such as the set of tweets that was discovered by chance to have a missing location and mislabeled language label. This particular set of tweets was written in Mandarin and was labeled as English text data, with text information in Mandarin discussing Russia and newly imposed sanctions. Whether accidental or intentional, this mislabeling was

discovered in a small sample set of around 100 tweets that was used during the earlier phases of the analysis when there were issues in efficiently computing large amounts of data. The top association rules for lift, confidence and support in the dataset were in Mandarin even though only English text had been selected. Further analysis into the tweet labels led to some interesting discoveries later on in the analysis.

Twitter is a highly active and evolving platform and the information that is available on the subject of the war in Ukraine is constantly changing as new developments in the war arise. Analysis that was conducted a few weeks back in time may result in findings that are irrelevant to the current information that is being generated on the platform. As time progresses, new text data is generated, new associations between words are created, and labels that might not have existed a few days ago are being attached to the tweet data. The methods used in the analysis have to be rerun with an updated dataset containing the most recent tweet data available in order for the findings to stay updated and relevant.

4.2 *Associations and Sentiment Analysis*

Initial findings based on the words and the sentiment attached to them yielded results that generally did not point to any sources of misinformation. Methods such as generating data trees, association rule mining (ARM), support vector machines (SVM) and neural networks (NN) did help to significantly narrow down the pool of data and brought light to which words were frequently showing up in the tweets and what sentiment they had attached to them. The accidental finding of the mislabeled language metadata in a subset of the tweets posed a hypothesis that there might be more interesting data to be found in the labels and metadata attached to each tweet. Since this particular set of tweets had a missing location label, an analysis was conducted on hashtags and location labels.

Using a large subset of over a million tweets available from the Kaggle dataset, a roBERTa sentiment analysis was performed on the hashtag data in order to determine which words in the tweet had the most negative sentiment attached to them. A manual inspection of these negatively associated hashtags and the text data of the tweets they were attached to led to a more detailed analysis. A total of 15 hashtags with potential disinformation were chosen for further analysis: 'MAGA' , 'maga', 'MAGAForever', 'NaziUkraine', 'ukrainiannazi', 'ZelenskyWarCriminal', 'MaidanCoup', 'ProxyWar', 'Fascism', 'Nazism', 'Burisma', 'Corruption', 'Corrupt', 'MySonHunter','AmericaFirst'.

4.3 *Disinformation*

A subset of all the tweets containing these hashtags was generated, along with a list of unique locations attached to these tweets. The list of hashtags was further expanded to 6277 “pro-Russia” hashtags which were identified as being associated with the tweets containing 15 hashtags tied to potential disinformation. The location data that was generated using a subset of the 15 hashtags tied to potential disinformation yielded some surprising results. A very large subset of these tweets had location data that was intentionally mislabeled with misleading or garbage locations. The text data that these hashtags were attached to were blatant disinformation and pro-Russian propaganda, in many cases.

Image borrowed from: https://www.shrmonitor.org/exploiting-historical-grievances-and-fears-disinformation-narratives-in-central-europe/

4.4 *Further Steps*

Many of the tweets containing misinformation that were discovered using an analysis on pro-Russian hashtags and unique locations had intentionally mislabeled location metadata, an interesting fact that was noted. This might be to potentially avoid Twitter detection filters that search for Russian-based tweets and users. Another interesting fact was that many locations found in the pro-Russian hashtag analysis were in the US South, with many of those tweets containing obvious English grammatical errors that would not be said by an native English speaker. This implies that there are accounts that attempt to appeal to a particular subset of the population in the US, particularly in locations where "MAGA" has a strong support and voter base. Most of the accounts attributed to these tweets containing disinformation have been suspended and the tweets have been removed, implying that Twitter is actively censoring and removing this disinformation. The dataset where this analysis was performed contains tweets from the past three months. It is possible that more recent tweets are yet to be discovered and have not been identified and removed yet.

The identification of misinformation would be significantly improved if pro-Russian and other hashtags implying disinformation could be self-identified without manual inspection, considering that this is a very time consuming process. The currentness of the dataset is also another issue considering that new information appears quickly and new associations have to be made in order to find sources of disinformation. Logistical challenges of processing massive amounts of tweet data also have to be taken into account in order for disinformation to be accurately and quickly identified so that it reaches as few people as possible. As a suggestion for Twitter, it would probably be of use to have location metadata attached to tweets derived from IP addresses, not allowing the location to be manipulated by the user.

5. Code

Links to all code and data are given below:

The zip folder contains six .R files in which the analyses were performed and in which the data was cleaned from the Twitter API and Kaggle sources.

Twitter_csv_ARM_v2.R — Code to read the Kaggle csv file and perform ARM
ARM_Kaggle_Twitter.py — Code to read the Kaggle csv file and perform ARM
read_Kaggle_twitter_csv..R — Code to read the Kaggle csv file and perform clustering
read_twitter_csv.R — Code to read files generated from the Twitter API access and perform clustering
Twitter_API_2_csv.R — Authenticating the Twitter API access, reading tweets, cleaning data and writing the data to .csv files.
Twitter_Kaggle_NaiveBayes_SVM_DecisionTrees.R — Supervised machine learning with decision trees, SVM, and Naive Bayes.
Negative_hashtags.ipynb — negative hashtag analysis
Regression_2D.py — 2D regression analysis
Regression_2D_Sigmoid.py — 2D regression analysis with sigmoid activation
NN_FF_and_BP_Module_4_Mods.py — 2D regression analysis with simple NN
NN_FF_and_BP_Module_5.py — sentiment labeling with simple NN

0930_UkraineCombinedTweetsDeduped.csv —- Raw Kaggle data
Users_cleaned.csv — Cleaned Twitter API user data
Users_complete.csv — Raw, complete Twitter API user data
Tweets_cleaned.csv — Twitter API tweets, cleaned data
Tweets_complete.csv — Twitter API tweets, raw complete data

The other files are .png. These are the images and graphs / charts that were generated from running the different types of clustering analyses and ARM.