2022

Twitter Sentiment Analysis on Russia-Ukraine Conflict

A COMPARATIVE STUDY

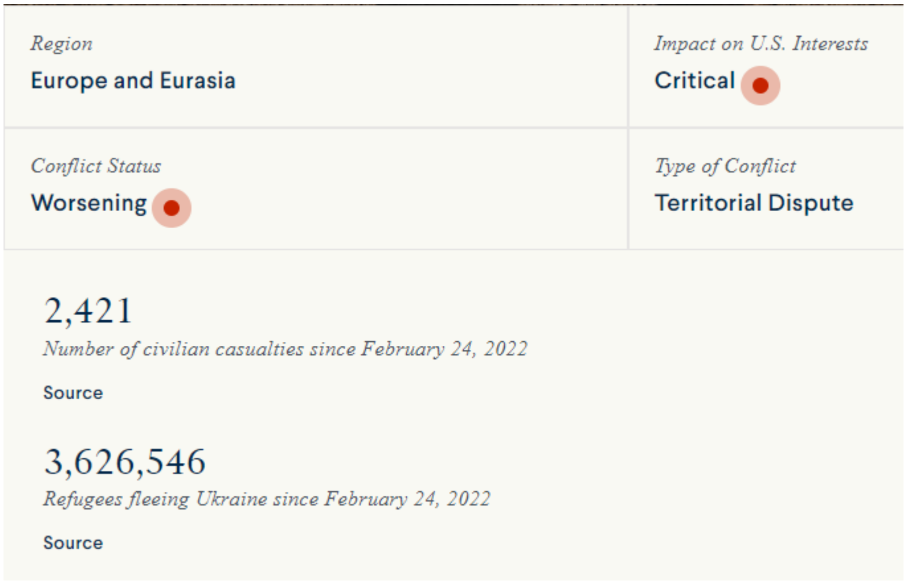
GROUP NO: 12

Contents

Introduction

For over a decade, digital platforms have played an important and increasingly growing role in crisis, conflicts, and war. People all over the world use social media platforms to document human rights issues, abuses, conflicts, condemn atrocities, appeal to the international community for action and to extend measures of relief. These platforms are also where governments, political leaders and others coordinate actions, recruit fighters and might also spread disinformation and incite violence. The Russo-Ukraine war is no exception.

In October2021, Russia began moving troops and military equipment near Ukraine borders, reigniting concerns over potential invasion. On February 24, 2022, during a United Nations Security Council meeting to dissuade Russia from attacking Ukraine, Putin announced the beginning of a full-scale land, sea and air invasion of Ukraine targeting military assets all over Ukraine.[1]



Region
Europe and Eurasia

Impact on U.S. Interests
Critical ●

Conflict Status
Worsening ●

Type of Conflict
Territorial Dispute

2,421
Number of civilian casualties since February 24, 2022
Source

3,626,546
Refugees fleeing Ukraine since February 24, 2022
Source

Fig. 1: Ukraine- Russia Conflict Tracker dashboard

After Russia's invasion of Ukraine, social media was flooded with pictures of bombings, people fleeing from their houses, cities destroyed, building burning etc. Social media was also flooded with sentiments of people from all over the world, be it that of fear and anguish from Ukraine, Pride and Joy from Russia and sympathy from people all over the world for Ukraine and anger towards Russia.

At the time of writing, the conflict is still ongoing. People worldwide have been using social media to share their opinions regarding this conflict. Online Social Networks (OSNs) have been a prominent source of data in studying prior large-scale information discourse during crises and social movements particularly in terms of 'information warfare' [2]

Through this project, we aim at analyzing the sentiments of people all over the world through the Tweets that they have been sharing on Twitter. Twitter is widely used by government agencies, companies, news channels and even common citizens.

Literature Review:

Sentiment analysis for Twitter data is a known technique that many organizations and companies actively implement to understand how the people are receiving and responding to certain events, products, or ideas.

In their paper, 'A Study on Sentiment Analysis Techniques of Twitter Data '[3] , Abdullah et al. (2019) have discussed the three levels of sentiment analysis namely Document level, Sentence level and Aspect and feature level. They have discussed various supervised Machine learning approaches for Twitter sentiment analysis using classification methods like Naive Bayes , Maximum Entropy algorithm and Support Vector Machine. They have also explored Lexicon based approaches (unsupervised methodology) and also a few hybrid methods. They found that supervised learning techniques SVM and Multinomial Naive Bayes produced the best precision results.

Bhumika et al. (2017) [4] has explored techniques for Twitter Sentiment analysis using Python Libraries. They have explained in detail how the NLTK (natural language toolkit) library in python can be used for text processing and classification. Operations such as tokenization, tagging, filtering, text manipulation can be performed with the use of NLTK. They also used SCIKIT Learn which provides many machine learning classification algorithms, efficient tools for data mining and data analysis, from a statistical standpoint. They experimented with various classification models and found that DAN2 gave the highest classification accuracy 0f 86.06%

In 'A Clustering-based Approach on Sentiment Analysis' Gang Li et al. [5] have proposed a clustering-based approach for sentiment analysis. They experimented with the famous movie review data and performed clustering using K-means algorithm in MATLAB Toolkit, using cosine distance as the distance method. Post which, they applied the TF-IDF weighting approach to evaluate the term frequency in a given document.

$Wi=tfi*log(D/dfi)$

In this expression, tfi is the frequency of term i in a document, D is the number of documents in the corpus, and dfi is the document frequency or number of documents containing erm i. Thus, log (D/dfi) is the inverse document frequency. By applying this weighting method, they found that importance increases proportionally to the frequency of a term in a document. In order to obtain more stable clustering results, they designed a voting mechanism. They found that the size of the document has a great influence on the outcome.

Kim et al (2015) in their paper ‘Survey on Aspect- Level Sentiment Analysis’ [6] have discussed the three processing steps that can be distinguished when performing aspect-level sentiment analysis: identification, classification, and aggregation. While in practice, not every method implements all three steps or in this exact order, they represent major issues for aspect-level sentiment analysis. The first step is concerned with the identification of sentiment-target pairs in the text. The next step is the classification of the sentiment-target pairs. The expressed sentiment is classified according to a predefined set of sentiment values, for instance positive and negative. Sometimes the target is classified according to a predefined set of aspects as well. At the end, the sentiment values are aggregated for each aspect to provide a concise overview. The actual presentation depends on the specific needs and requirements of the application.

Samar et al. (2020) experimented with Randomized clustering Cuckoo Search Algorithm which has proved effective in solving global optimization problems like minimizing cost problems and maximizing the profit, output, performance, and efficiency problems. One of the advantages of this algorithm is to have two ways of searching: global search and local search that can be controlled by switching probability parameters. They used the RCCS algorithm to find the optimal number of clusters to describe the data and developed a method that can deal with any type of data (text, audio, images etc.) [7]

Research Question:

It is established that in this on-going Russia-Ukraine conflict, people from all over the world are sharing their opinions and views about this situation. Through this project, we aim at gauging the sentiments that people are expressing through their tweets. A general hypothesis based on recent reported news is that people from Ukraine are feeling ‘fear’ and ‘anguish’ and people from Russia are expressing ‘pride’ and people all over the world are having mixed feelings. We aim to analyze and validate this hypothesis. We will use Python’s NLTK library and compare and contrast the results of 2 sentiment analysis techniques – TextBlob , Vader and Flair. As this is an unsupervised learning task, we will be using Latent Dirchlet Allocation(LDA) technique for topic modeling and divide the tweets into similar topics to get deeper insights into the textual features.

Methodology:

Data Crawling:

We have implemented 2 methods to scrape tweets from Twitter.

Method 1 - Recent (past 7 days) Tweets

We created a list of highly trending hashtags related to the Russia - Ukraine war and found that the following hashtags were majorly used by people to address their opinions.

Serial No.	Hashtags
1	#ukraineunderattack
2	#RussiaUkraineWar
3	#standWithUkraine
4	#PutinsWar
5	#WarInUkraine
6	#usasupportUkraine
7	#Insolidaritywithukraine
8	#supportukraine
9	#SupportRussia
10	#StandWithRussia
11	#stopwar
12	#ukraineconflict
13	#UkraineRussianWar
14	#SanctionRussiaNow
15	#WorldWar3
16	#RussiaUkraineCrisis
17	#UkrainiansWillResist

18	#ChangeInRussianUkrainianSituation
19	#UkraineIssue
20	#Kharkiv

Step 1 : Importing libraries and Authentication

We use the pandas library to create a dataframe and store the scraped tweets in this dataframe. Using the tweepy module and keys obtained from Twitter through elevated access for Twitter developer portal , we establish a secure connection with twitter server. [8]

Making use of Tweepy Library on Python, using the search_tweets API, we were able to mine recent public tweets from all over twitter pertaining to these 20 hashtags.

API.search_tweets(q, *, geocode, lang, locale, result_type, count, until, since_id, max_id, include_entities)

Step 2: Scraping using python script

We created a scraping script that takes 2 inputs - Hashtag, Since- date.
Using this script, we were able to collate a dataset of recent tweets. Irrespective of the since- date, this script will always fetch recent (current timestamp) data.
Tweepy script

Method 2 - To fetch past Tweets (creation< todays' date - 7 days)

Due to the limitation of the tweepy's and twitter's API.search_tweets recent 7 days limit, we had to implement another approach to mine historic or rather past tweets, considering the speculations started in October 2021 and the war started on February 24, 2022.

We used the TWINT module to tackle this issue.

Step 1: Installing packages and importing libraries

Twint is an advanced Twitter scraping tool written in Python that allows for scraping Tweets from Twitter profiles without using Twitter's API. Twint utilizes Twitter's search operators to let you scrape Tweets from specific users, scrape Tweets relating to certain topics, hashtags & trends etc [9]

Step 2: Data mining using python script

We created a script that fetches tweets using the twint library. The function takes a hashtag and a date as input and returns tweets for that particular day, with a limit of 2000.
We used the following script to mine data around Feb 24, 2022, and April 18, 2022.

Twint script

Data Preprocessing:

In order to extract important and useful features from textual data, it is essential to preprocess it. Pre-processing makes the raw data ready for analysis through machine learning algorithms. If data is noisy and raw, there is a possibility that the analysis won't be accurate.

We followed the following steps to process and clean the textual data obtained from tweets for TextBlob and Flair.

- Convert tweet text into lower case
- Remove urls
- Remove ‘rt’ , retweet tags and ‘cc’
- Remove hashtags
- Remove @mentions
- Remove emoticons
- Remove HTML tags
- Remove extra spaces
- Remove Punctuations

We followed the following steps to process and clean the textual data obtained from tweets for Vader. We use a different approach for Vader because Vader uses capitalization, emoticon etc to gauge the sentiments of the text.

- Remove retweet handles
- Remove users handles
- Remove URL links
- Remove punctuations

Exploratory Data Analysis:

Exploratory Data Analysis refers to the critical process of performing initial investigations on data so as to discover patterns, to spot anomalies, to test hypotheses and to check assumptions with the help of summary statistics and graphical representations.

We managed to capture 2 datasets, using 2 different approaches. We performed exploratory data analysis on both these data sets and following are the results.

Part 1: Recent Tweets EDA

In order to understand which attributes, have complete values and which attributes have missing data value, we did a missing value analysis and found that 'location' attributes have maximum values missing. Among the location values fetched, we found that people from following 20 locations are actively sharing their opinions on the ongoing Russia-Ukraine conflict. Due to restricted services of twitter in Russia, we can observe that tweets from Russia are limited.[10]

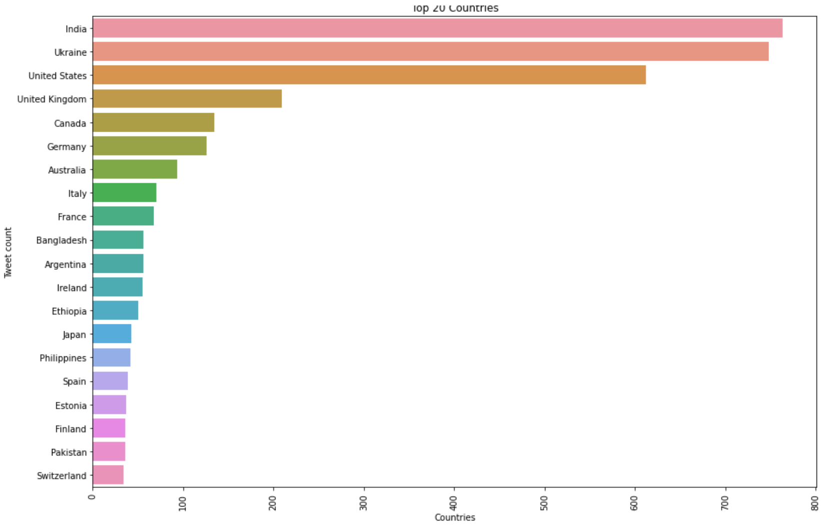


Fig. 2: Bar graph showing top 20 locations with maximum activity

Top 30 words from Ukraine:

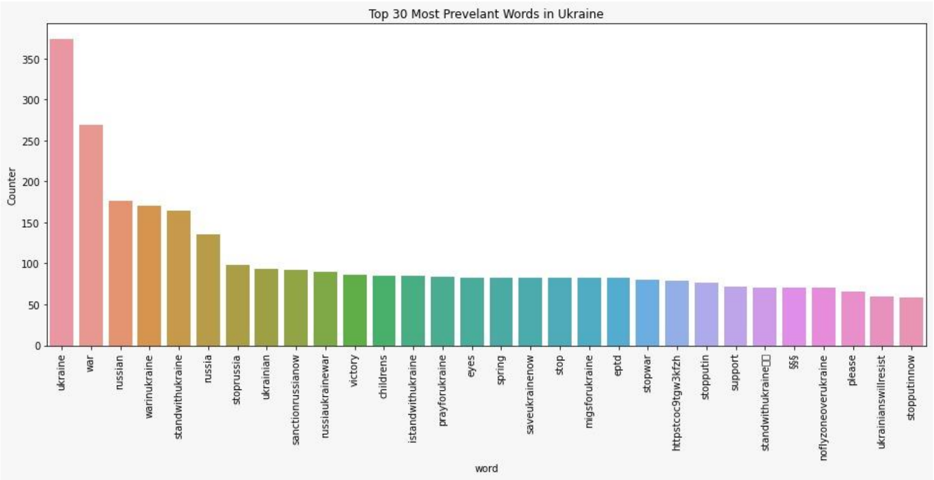


Fig. 3: Bar graph showing the frequency of top 30 words in tweets coming from Ukraine

Top 30 words from Russia:

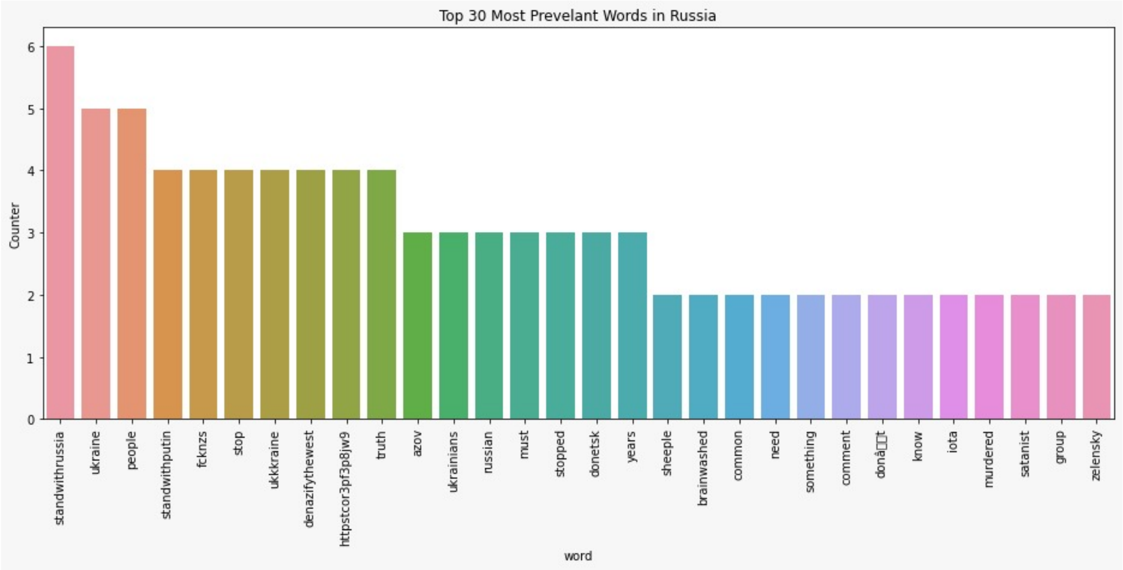


Fig. 4: Bar graph showing the frequency of top 30 words in tweets coming from Russia

It is very interesting to know that tweets from Ukraine have 'Ukraine' as the most frequently used word and tweets coming from Russia have 'Russia' as the most frequently used word respectively.

Word Cloud: Using a tank image for masking, we were able to make a tank shaped word cloud for better and more appealing visualization.

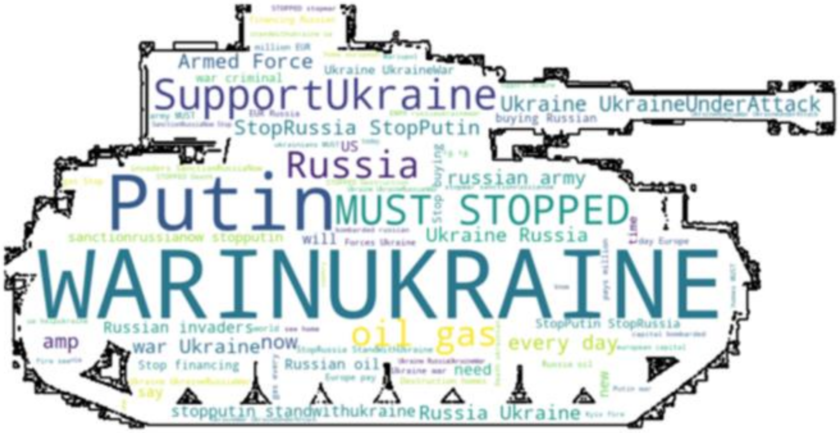


Fig. 5: Word cloud for top words from all tweets

Part 2: Past Tweets EDA

We observed that tweets and opinions have been posted from all parts of the world in various languages. Even though we were not able to capture concrete location data values in the past tweets, we were able to capture the language values of the tweets and found that after English, German was the highest used language for tweets, followed by UK English, Italian etc.

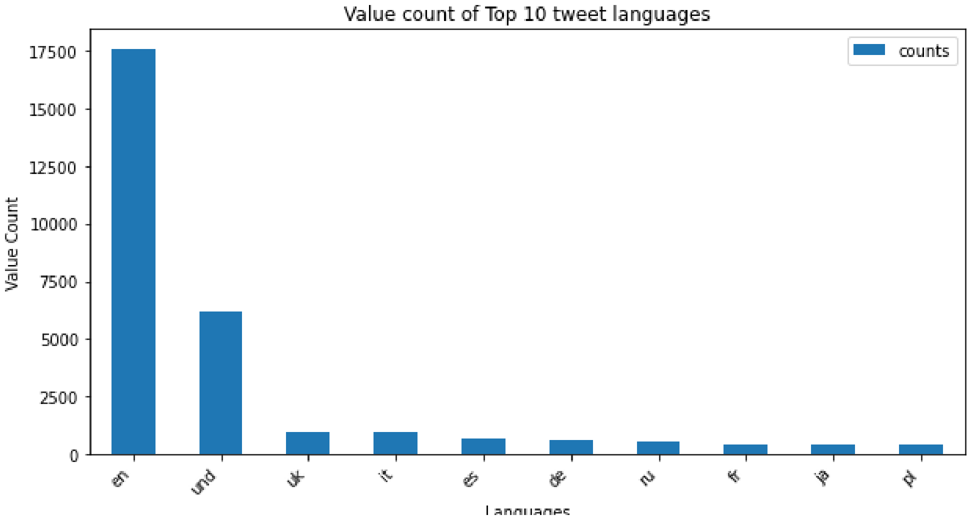


Fig.6: Bar graph showing top 10 languages for the tweets

Top 50 words in English language:

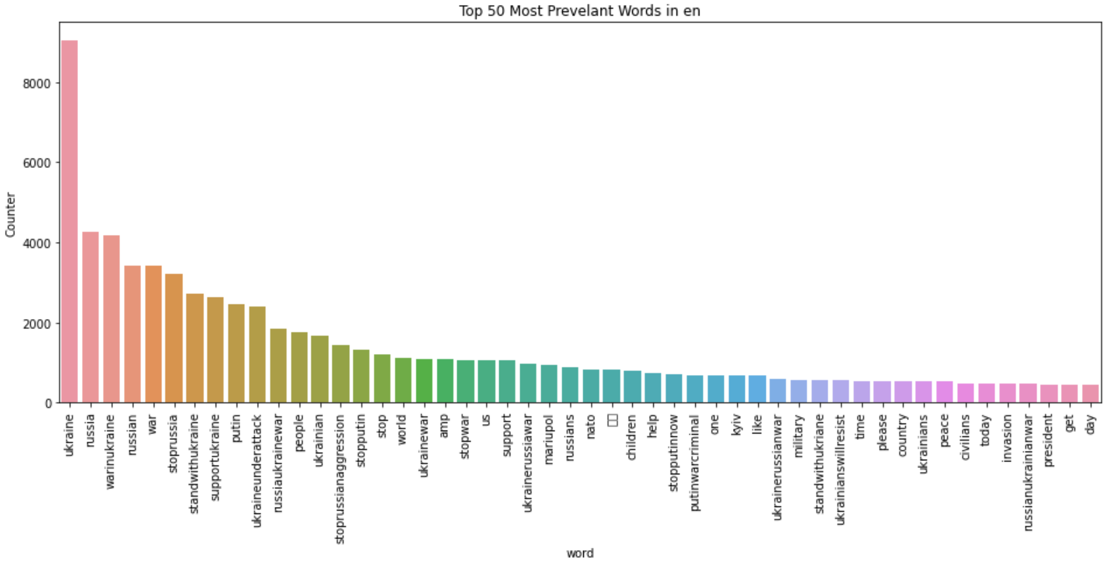


Fig. 7: Bar graph showing Top 50 words and their respective frequency

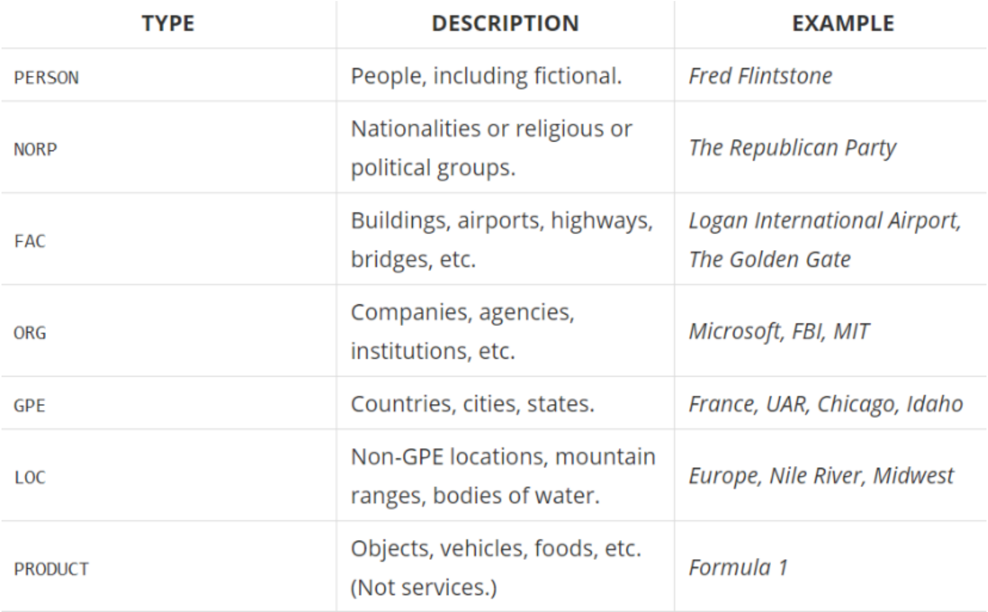
Observations:

- Most of the tweets encompassing this subject are coming from USA, Ukraine, UK, India, and Canada are the top countries .
- People are tweeting in various languages apart from English, like German and Italian etc. indicating that there is a lot of information and opinions being expressed, However, since our study will concentrate only on textual data In English, we will not be able to capture sentiments from other languages.
- Words from Ukraine are majorly talk about 'war in Ukraine' and 'stand with Ukraine'. However, those from Russia are like 'Stand with Russia' and 'Stand with Putin' outlining the very evident polarity from both sides.

Spacy for Named Entity Recognition

Spacy is a relatively new framework in the Python Natural Language Processing. Spacy is an open-source Python library that provides capabilities to conduct advanced natural language processing analysis and build models that can underpin document analysis, chatbot capabilities, and all other forms of text analysis. It provides models for Part of Speech tagging, Named Entity Recognition and Dependency Parsing. It's becoming increasingly popular for processing and analyzing data in NLP. Unstructured textual data is produced at a large scale, and it's important to process and derive insights from unstructured data. To do that, you need to represent the data in a format that can be understood by computers.

Spacy has built-in methods for Named Entity Recognition. Spacy has a fast statistical entity recognition system. We can use spacy very easily for NER tasks. [11]



TYPE	DESCRIPTION	EXAMPLE
PERSON	People, including fictional.	*Fred Flintstone*
NORP	Nationalities or religious or political groups.	*The Republican Party*
FAC	Buildings, airports, highways, bridges, etc.	*Logan International Airport, The Golden Gate*
ORG	Companies, agencies, institutions, etc.	*Microsoft, FBI, MIT*
GPE	Countries, cities, states.	*France, UAR, Chicago, Idaho*
LOC	Non-GPE locations, mountain ranges, bodies of water.	*Europe, Nile River, Midwest*
PRODUCT	Objects, vehicles, foods, etc. (Not services.)	*Formula 1*

Fig.8: Table briefly explaining the various NERs

Results from NER:

Top 10 entities that are mentioned in the tweets.

In order to dig deeper as to what context, news or events are being talked about in the tweets data collected, we run an analysis on the top 10 entities that are being talked about. We find that NORP (nationalities, political groups etc.) is the topmost entity. We verified this manually as well and find the mentions like ‘Russian’, ‘Ukrainian’ etc. mentions in a lot of tweets. This can also be confirmed when we try to identify the top 10 NORP elements.

We also notice that GPE(geographical elements) are the second most talked about entities. We then have Cardinal entities, which are essentially numerical values that do not fall under any other NER category.

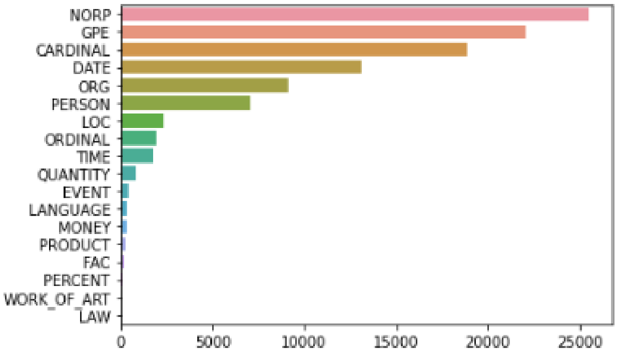


Fig. 9 : Top 10 entities from the tweets recognized by spacy

Top 10 NORP and GPE elements

After close examination of the top 10 NORP elements we find that ‘Russian’, ‘Ukrainian’ and ‘European’ have the most mentions in the tweets. Which completely aligns with the fact that the crisis is currently affecting these 3 nationalities. It’s very interesting to know that when it comes to GPE elements, we also have a lot of mentions of India and USA, along with Russia and Ukraine, indicating the global impact.

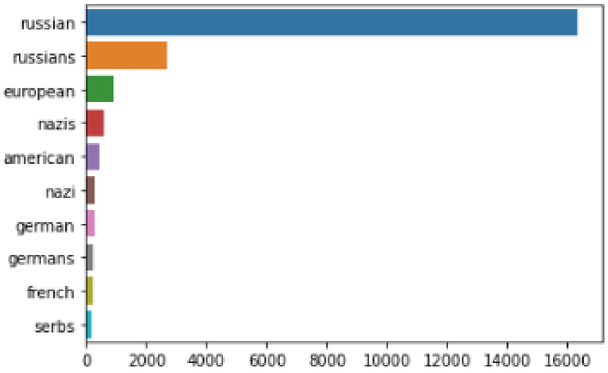


Fig. 10: Top 10 NORP elements

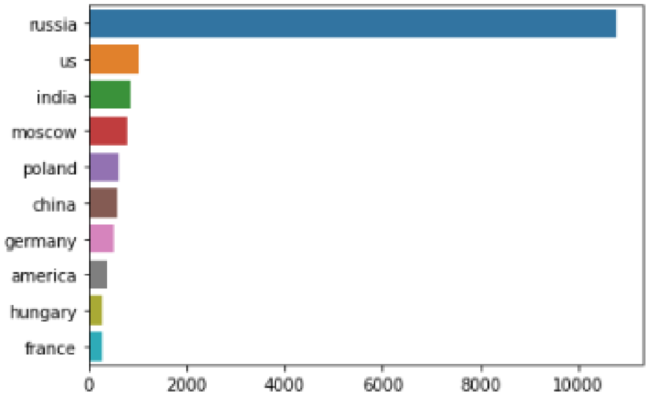
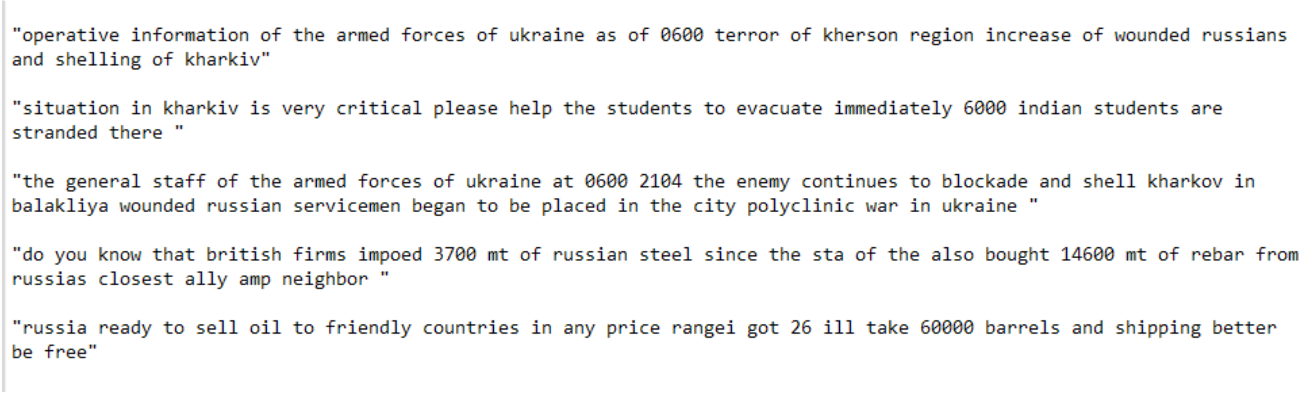


Fig. 11: Top 10 GPE elements

Cardinal Entities

At first, cardinal entities look like stray numeric values. Upon close evaluation of tweets , we find that these numerical values give more information about the situations or circumstances being talked about.



```
"operative information of the armed forces of ukraine as of 0600 terror of kherson region increase of wounded russians
and shelling of kharkiv"

"situation in kharkiv is very critical please help the students to evacuate immediately 6000 indian students are
stranded there "

"the general staff of the armed forces of ukraine at 0600 2104 the enemy continues to blockade and shell kharkov in
balakliya wounded russian servicemen began to be placed in the city polyclinic war in ukraine "

"do you know that british firms impoed 3700 mt of russian steel since the sta of the also bought 14600 mt of rebar from
russias closest ally amp neighbor "

"russia ready to sell oil to friendly countries in any price rangei got 26 ill take 60000 barrels and shipping better
be free"
```

Sentiment Analysis

Vader

VADER is a lexicon and simple rule-based model for sentiment analysis. It can efficiently handle vocabularies, abbreviations, capitalizations, repeated punctuations, emoticons, etc. usually adopted on social media platforms to express one’s sentiment, which makes it a great fit for social media sentiment text analysis. [12]

We make use of SentimentIntensityAnalyzer() from Vadersentiment python library to analyze the sentiments in each tweet.

SentimentIntensityAnalyzer() function provides positive, negative, neutral, and compound scores for each tweet. Based on the values of compound scores, we can classify the tweet as positive, negative, or neutral. The Compound score is a metric that calculates the sum of all the lexicon ratings which have been normalized between -1(most extreme negative) and +1 (most extreme positive).

- Positive sentiment: (compound score >= 0.05)
- Neutral sentiment: (compound score > -0.05) and (compound score < 0.05)
- Negative sentiment: (compound score <= -0.05)

TextBlob

TextBlob is a python library for Natural Language Processing (NLP).TextBlob actively used Natural Language Toolkit (NLTK) to achieve its tasks. NLTK is a library which gives an easy access to a lot of lexical resources and allows users to work with categorization, classification, and many other tasks.

TextBlob is a simple library which supports complex analysis and operations on textual data. TextBlob returns polarity and subjectivity of a sentence. Polarity lies between [-1,1], -1 defines a negative sentiment and 1 defines a positive sentiment. Negation words reverse the polarity.[13]

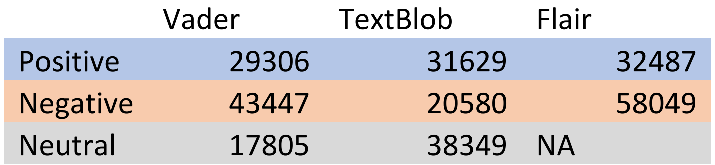
Flair

Flair is a simple natural language processing (NLP) library developed and open-sourced by Zalando the sentiment of a sentence is negative, the score is negative

The flair sentiment classifier was originally trained on IMDB movie review data. This difference may explain some of the limitations of this classifier, but it also appears that the TextBlob library and Vader captures other dimensions of "sentiment" that the flair NLP library just doesn't have.[14]

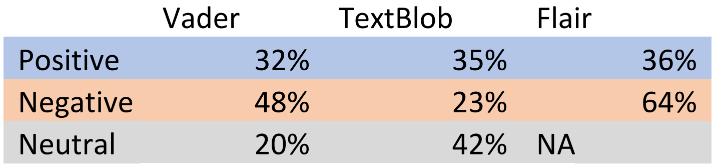
Results – Sentiment Analysis

When we compare and contrast the results of all the three sentiment analysis techniques , we find that Vader and Flair classify most of the tweets as negative in nature , following by positive. However, TextBlob classifies most of the tweets as neutral, followed by positive and then negative.



	Vader	TextBlob	Flair
Positive	29306	31629	32487
Negative	43447	20580	58049
Neutral	17805	38349	NA

Sentiment classification – Value Counts



	Vader	TextBlob	Flair
Positive	32%	35%	36%
Negative	48%	23%	64%
Neutral	20%	42%	NA

Sentiment classification - Percentage

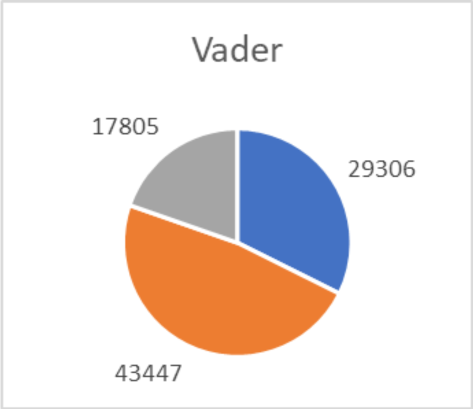


Fig. 12: Pie chart showing sentiments classification results - Vader

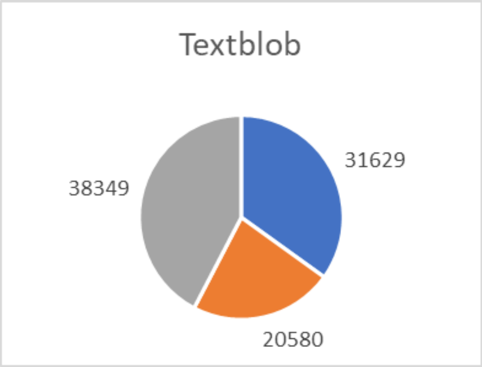


Fig. 13: Pie chart showing sentiments classification results - TextBlob

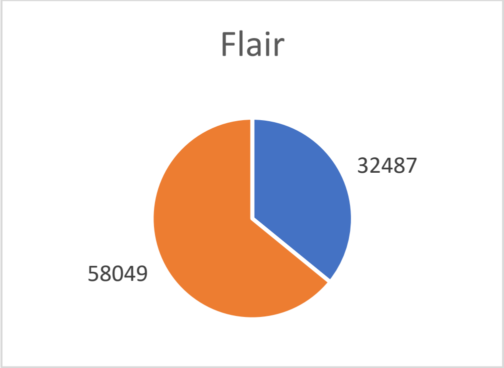


Fig. 14: Pie chart showing sentiments classification results - Flair

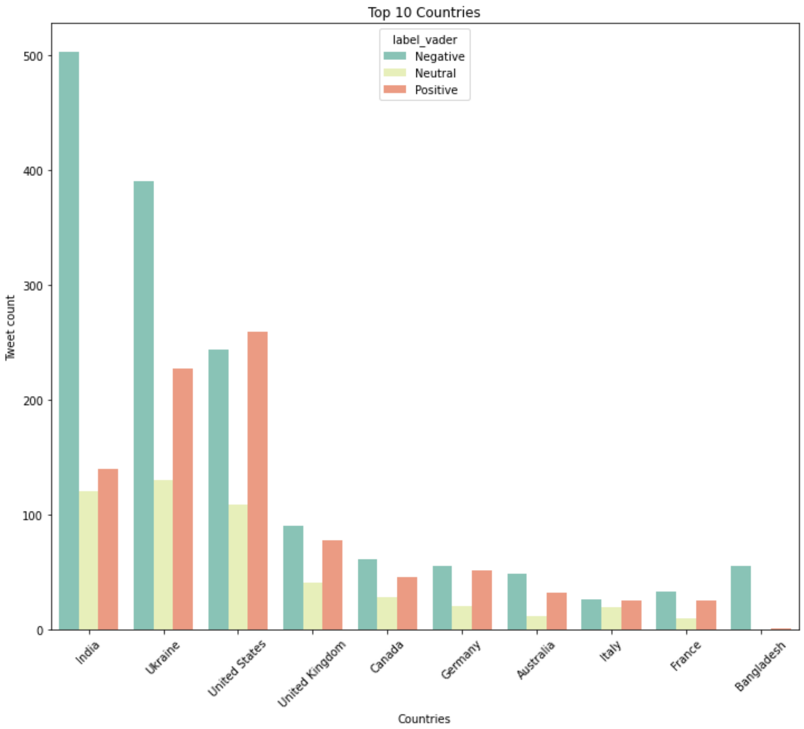


Fig. 15: Bar chart -Vader Sentiment distribution based on user location(country)

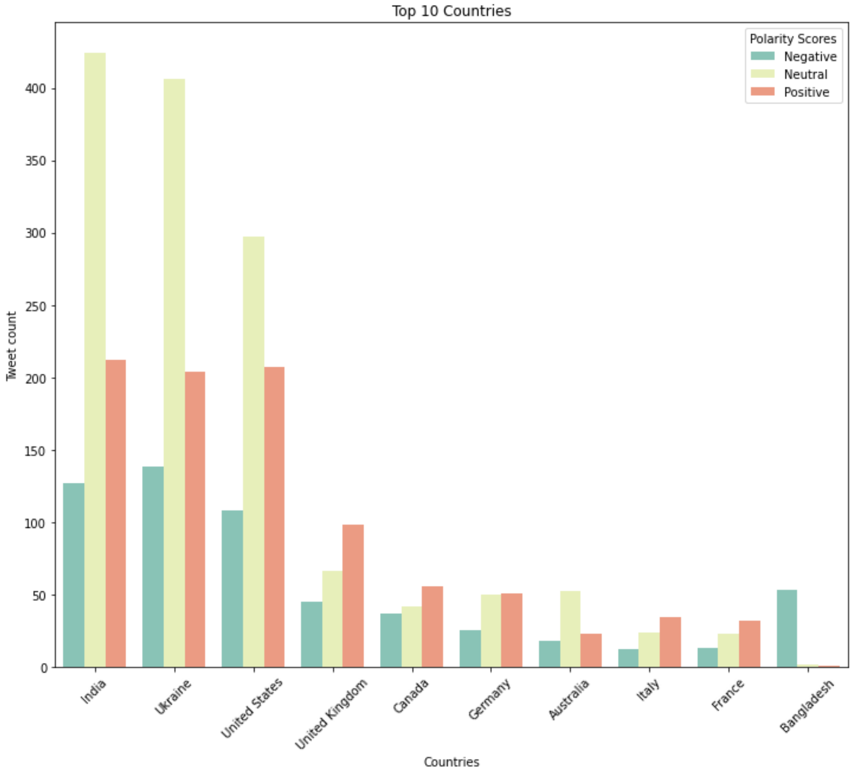
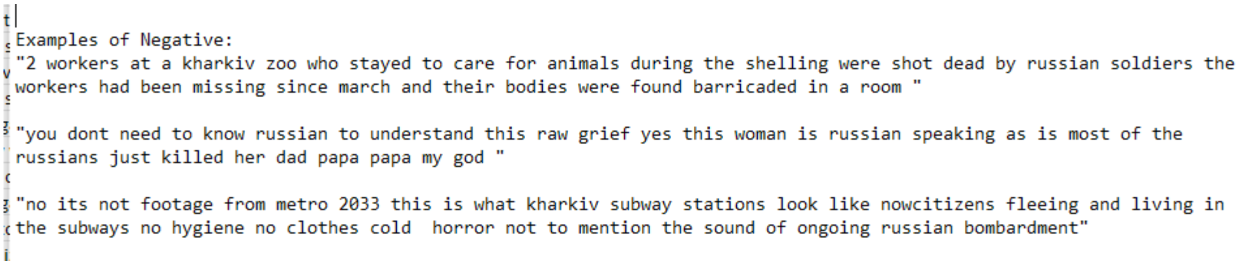


Fig 16: Bar- chart :TextBlob Sentiment distribution based on user location(country)

Manual Inspection of Sentiment analysis Results

It is given that since the subject itself comprises of war and conflict between Russia and Ukraine, major sentiments expressed are mostly negative. Upon closer evaluation of these tweets, we find the incidents and events that people are tweeting about.

Common negative tweets:



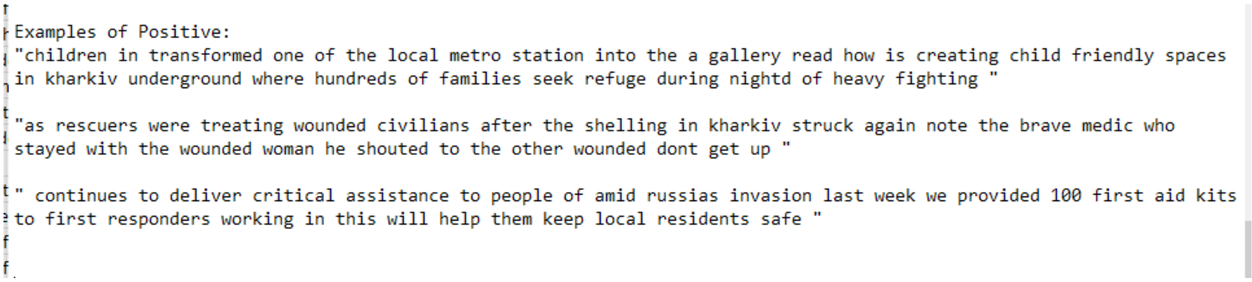
```
Examples of Negative:
"2 workers at a kharkiv zoo who stayed to care for animals during the shelling were shot dead by russian soldiers the
workers had been missing since march and their bodies were found barricaded in a room "

"you dont need to know russian to understand this raw grief yes this woman is russian speaking as is most of the
russians just killed her dad papa papa my god "

"no its not footage from metro 2033 this is what kharkiv subway stations look like nowcitizens fleeing and living in
the subways no hygiene no clothes cold  horror not to mention the sound of ongoing russian bombardment"
```

In first example we can clearly see that user is talking about 2 zookeepers being shot dead by Russian soldiers. In second tweet, we see that user is talking about a Russian woman losing her dad in the war, indicating that the war is having an adverse effect on Russian lives as well.

Common positive tweets:



```
Examples of Positive:
"children in transformed one of the local metro station into the a gallery read how is creating child friendly spaces
in kharkiv underground where hundreds of families seek refuge during nightd of heavy fighting "

"as rescuers were treating wounded civilians after the shelling in kharkiv struck again note the brave medic who
stayed with the wounded woman he shouted to the other wounded dont get up "

" continues to deliver critical assistance to people of amid russias invasion last week we provided 100 first aid kits
to first responders working in this will help them keep local residents safe "
```

First positive tweet indicates that children transformed a local metro station in Kharkiv into a gallery, making it a child friendly space. This tweet has been marked positive by Vader, TextBlob and Flair, indicating its optimistic nature. Similarly, second tweet talks about a 'brave medic' who stayed with a wounded woman. Third tweet is about some organization being able to provide aid kits, indicating a positive gesture.

In general, Twitter requires users to stick to character limits. It is also overwhelmingly used on mobile devices, which creates oddities in the way tweets are written and they may not be written in proper language, with proper spellings. The character limit often means that users tend to abbreviate words, which can create problems. Considering the text classifier used in Flair is pretrained on IMDB data, we can tell that it doesn't work well on the dataset that we have collected.

Upon closer manual inspection, we find that Vader(using capitalizations, emoticons etc.) provides the closest classification results to true sentiments of the tweets. In the digital and textual form of communication, where people use exclamations, emoticons etc. to express the intensity of the emotion, it is important that they are considered as well , towards gauging the sentiment of textual data. Vader clearly offers that ability.

Topic Modeling using LDA

Topic modeling is an unsupervised machine learning technique that's capable of scanning a set of documents, detecting word and phrase patterns within them, and automatically clustering word groups and similar expressions that best characterize a set of documents.

Latent Dirichlet Allocation (LDA) is a popular topic modeling technique to extract topics from a given corpus. The term latent conveys something that exists but is not yet developed. In other words, latent means hidden or concealed. Latent Dirichlet Allocation with online variational Bayes algorithm.

Now, the topics that we want to extract from the data are also "hidden topics". It is yet to be discovered. Hence, the term "latent" in LDA. The Dirichlet allocation is after the Dirichlet distribution and process.

Latent Dirichlet Allocation (LDA) does two tasks: it finds the topics from the corpus, and at the same time, assigns these topics to the document present within the same corpus. The below schematic diagram summarizes the process of LDA well: [15]

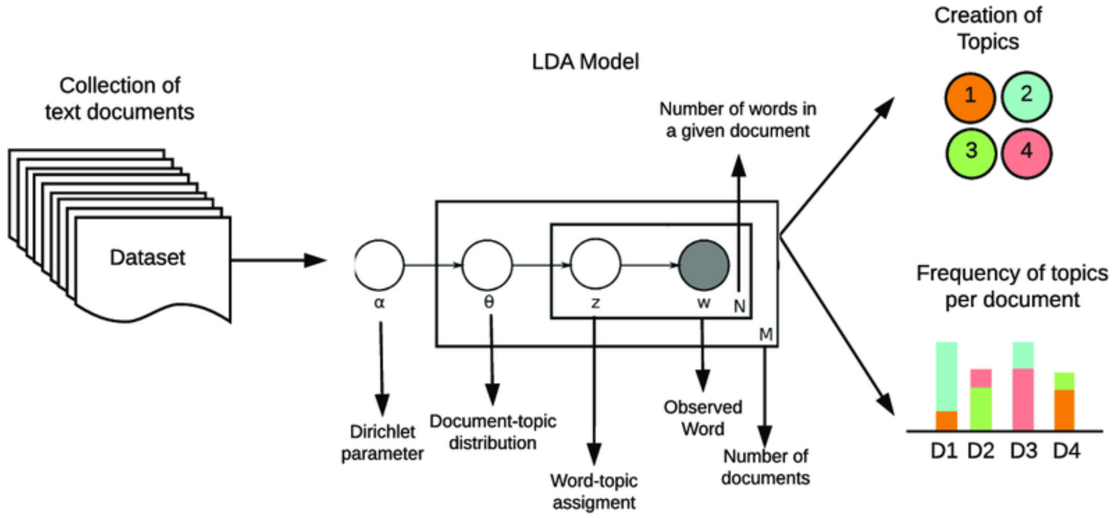


Fig.17: Schematic Diagram of LDA process

We use sklearn.decomposition.LatentDirichletAllocation algorithm to model our dataset into topics. [16]

Steps in Topic Modeling:

Step 1 – Create word vectors using CountVectorizer from sklearn.feature_extraction.text
We use the CountVectorizer method to convert our tweets into word vectors. The hyperparameters considered were as follows:

- Stop words - 'english'(case 1), ‘english’ + ‘russian’ + ’ukraine’(case 2)
- analyzer - 'word'
- min_df = 100 (minimum occurrence of word)
- lowercase = true
- token pattern > 3 char
- max_features = 5000 (max number of unique words)
-

Step 2: Fit transform the vectorizer on tweets to form a data matrix

Step 3: Defining the LDA model

- Hyperparameters for the model:
- No of topics = 2 to 9
- random state = 20
- n_jobs = -1

Step 4: Fit transform the LDA model on data matrix

Step 5: Calculate perplexity for all number of topics.
Perplexity is a statistical measure of how well a probability model predicts a sample. As applied to LDA, for a given value of no. of topics, estimate the LDA model. Perplexity as well is one of the intrinsic evaluations metrics and is widely used for language model evaluation. It captures how surprised a model is of new data it has not seen before and is measured as the normalized log-likelihood of a held-out test set.

LDA includes the function perplexity() which calculates this value for a given model.

Results of LDA modeling:

We observe that as number of topics increase the perplexity score decreases. It is also interesting to know that when we consider ‘russia’ and ‘ukraine’ as stop words(due to their frequent occurrences in the tweets) we find that the perplexity score increases in general, which is not desirable.

Case 1: english stopwords
We observe a local minimum at point 3 and 6. After 6, there is a very gradual decrease in the perplexity value. Thus, we consider number of topics for our analysis as 6 for our further evaluation.

Case 2: english stopwords + ‘russia’ , ‘ukraine’

Due to the frequent occurrences of words ‘russia’ , ‘ukraine’, we wanted to understand how the LDA model will converge if we consider them as stop words. However, we find that model finds it tough to converge , based on increased perplexity score, when these 2 words are added to stop words list. This can indicate that these 2 words add context to the topics and hence, are essential to be considered in the corpus.

	English stopwords + ‘russia’, ‘ukraine’	english stopwords
No of Topics	Perplexity	Perplexity
2	906.09	836.26
3	887.36	791.03
4	888.62	792.77
5	843.86	787.16
6	833.57	763.59
7	837.85	760.79
8	820.15	756.25
9	817.2	753.63

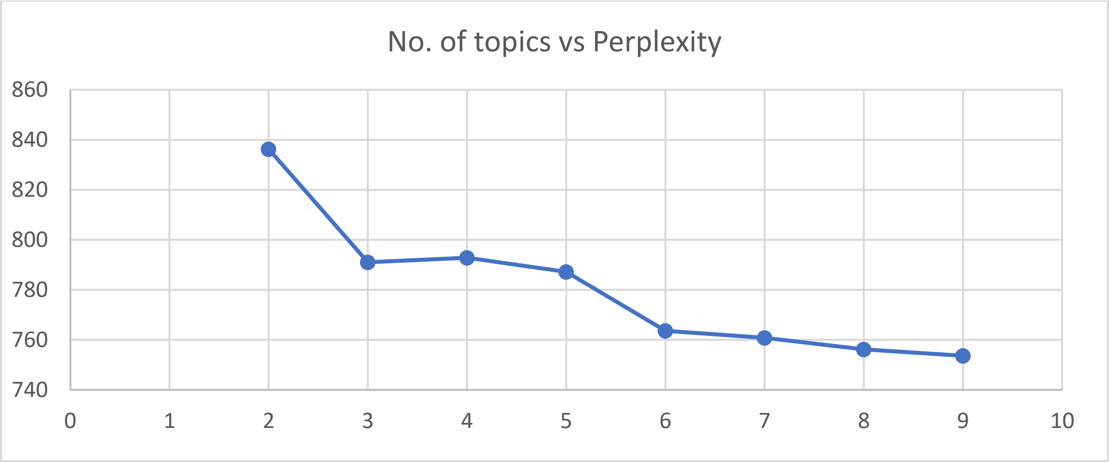


Fig. 18: Perplexity with english stop words

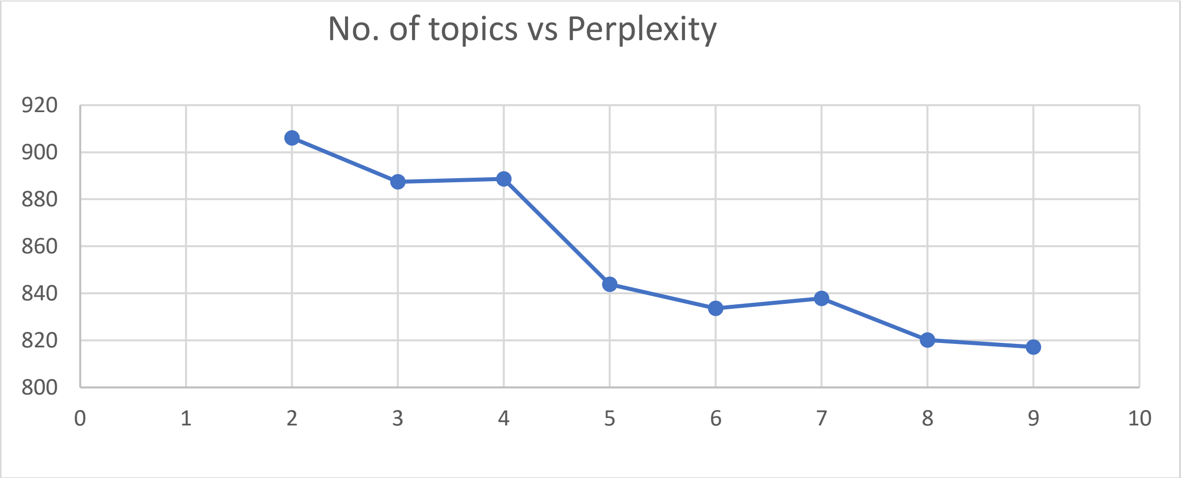


Fig. 19: Perplexity with english stop words + ‘russia’, ‘ukraine’

Visualization and Evaluation of LDA results:

We use pyLDAvis to visualize the results from our LDA. PyLDAvis is designed to help users interpret the topics in a topic model that has been fit to a corpus of text data. The package extracts information from a fitted LDA topic model to inform an interactive web-based visualization. [17]

Text data, when transformed into numeric tabular data, usually is high-dimensional. On the other hand, visualizations on a screen is two-dimensional (2D). Thus, a method of dimension reduction is required to bring the number of dimensions down to 2. In order to achieve this, we select multidimensional scaling as tsne -(t-distributed Stochastic Neighbor Embedding)

After close examination of the inter-topic visualizations, we observe that all 6 topics are very much separated from each other with no overlap. The further the bubbles are away from each other, the more different they are. However, on close examination, we find a bunch of common high frequency words.

When we manually try to understand the context of each topic, we realize that this is a very tough task. This is actually the step where domain knowledge is very useful. However, it seems that the tweets are so general in addressing the issue, it is a challenge trying to gauge context of each topic.
Number of topics = 6

- Topic 1 has words like 'please, thank, peace, right, good' which might indicate that this one is about tweets mentioning peace and probably few positives happening in this crisis.
- Topic 2 indicates the words around war in Mariupol region and children being killed
- Topic 3 mentions russia, ukraine war and new invasion

- Topic 5 mentions 'news, sanctions' indicating the sanctions imposed on Russia
- Topic 6 mentions 'must ,stop ,russia’

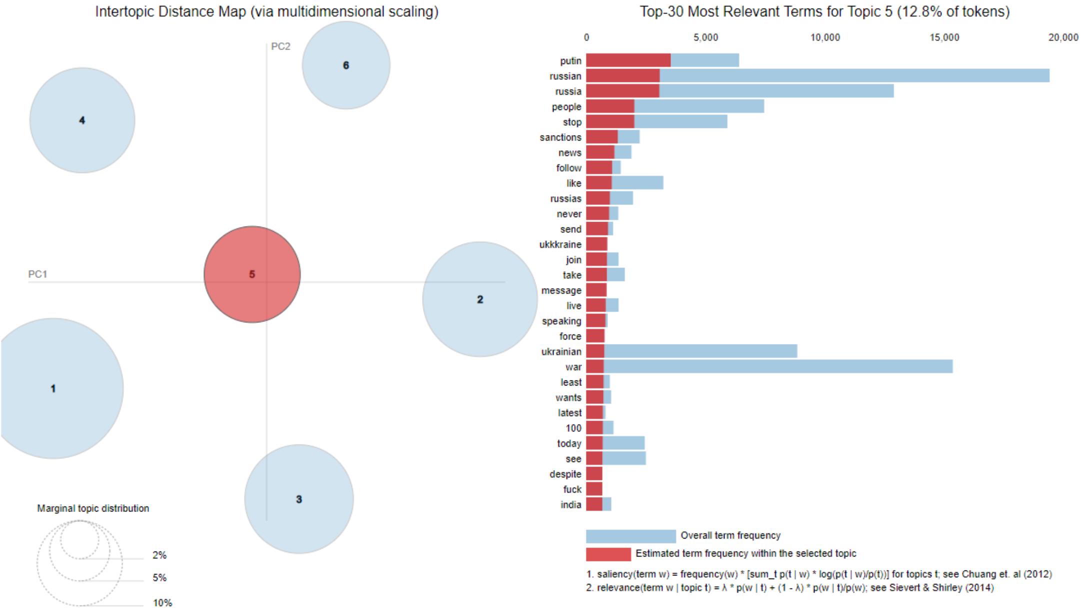


Fig. 20: Intertopic distance map for 6 topics- highlighting topic 5

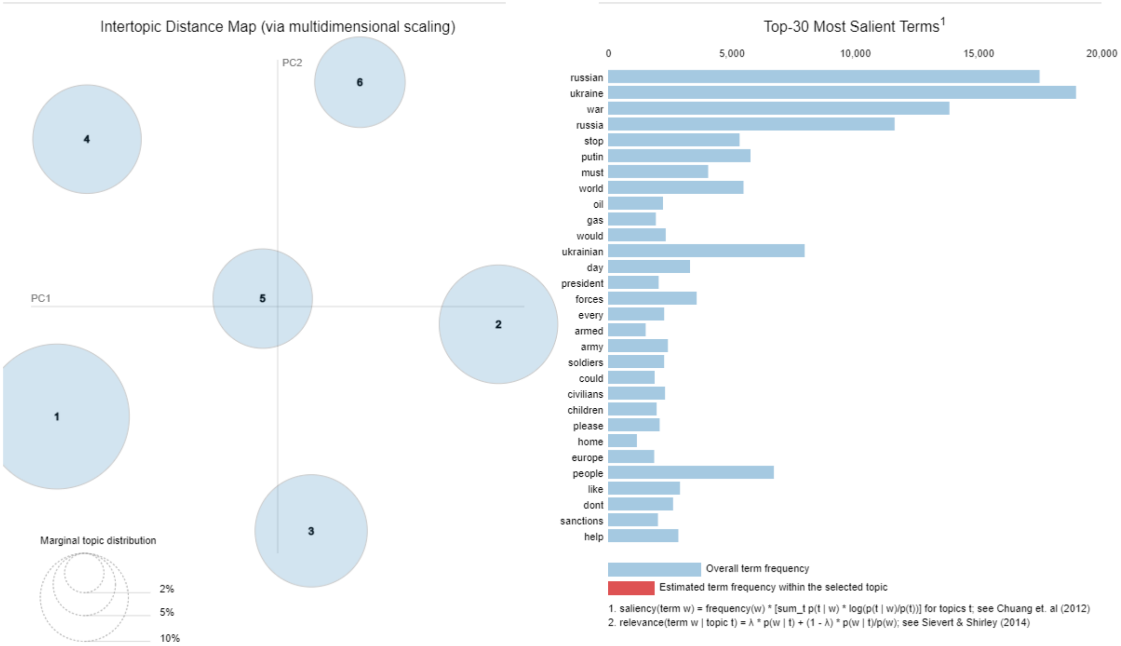


Fig. 21: Intertopic distance map for 6 topics

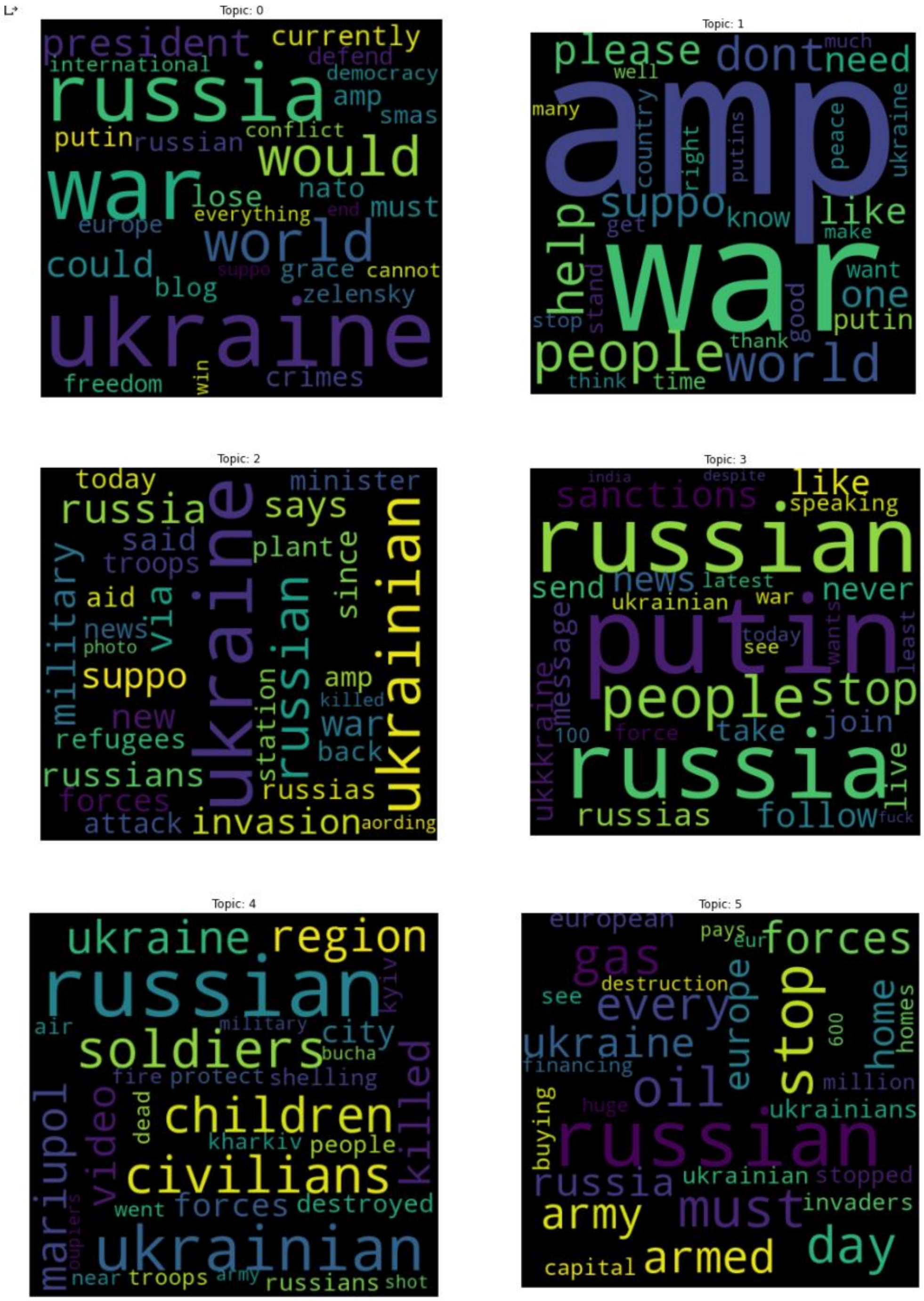


Fig 22: word cloud for each topic

Topics and their Sentiments :

In order to get deeper and more structured insights, we combine the results from both sentiments analysis via Vader and topic modeling(assigning a topic to each tweet based on LDA) to form a heat map.

This heat map helps us understand the sentiments expressed in each topic.

- Topic 1 comes across as a balanced on with both positive and negative sentiments with almost equal intensity.

- Topic 0, 2, 3 are majorly inclined towards negative sentiments than positive.

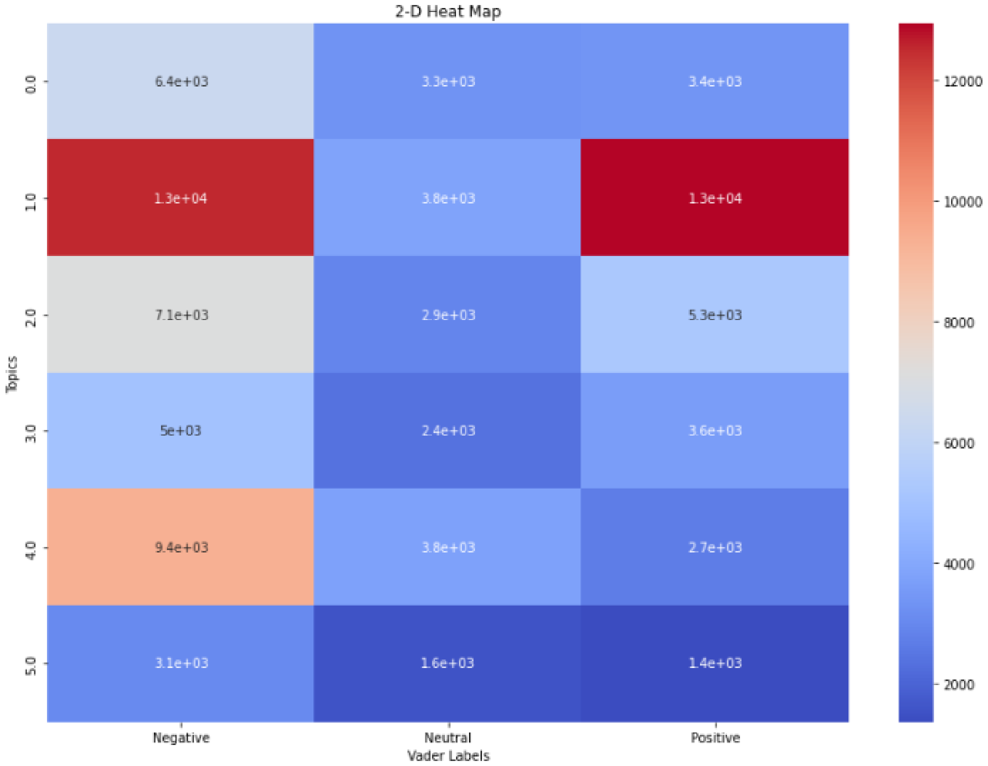


Fig. 23: Heat map indicating sentiments per topic

Challenges:

1. The data scraped from twitter had a lot of missing values like location(this is tracked as per user preference). Even though we were able to get plots for ‘sentiments per country’ , missing location values on a lot of tweets kept us from getting better or more accurate results. We had to make do with the values available.

2. Tweets are character limited and hence, often user use abbreviations, which the machine does not understand. One such usage found was that of word ‘amp’. The user used it in particular context, which is almost impossible for the system to measure. This is a limitation of NLP techniques.
3. Based on the word vectors and data matrix, we created our LDA model for 6 topics. However, understanding the context of each topic is a challenge and requires manual inspection.
4. Some tweets are empty tweets(only contain mentions @s and maybe images). These tweets add noise to data.
5. LDA modeling is time consuming, especially for a large dataset. Model takes a lot of time to fit and transform and provide result for each instance.
6. Hyperparameter tuning is also a time-consuming affair.

Conclusion:

- Russia, Ukraine are the most frequent words in majority of the tweets. These words also contribute to the GPE entities. Similarly, we have 'Russian' and 'Ukrainian' as high frequency words, which contribute to the NORP entity of the data.
- There are a lot of numbers and dates used in the tweets to express time of incidents and events or even number of people killed or no. of soldiers etc., which were considered as Cardinal entities.
- India, Ukraine, USA, UK and Canada remain the top 5 countries from where most of the tweets related to Russia-Ukraine conflicts are coming in.
- For the data that we collected, Vader Sentiment analyzer works the best and provides closer to true sentiment classification.
- Considering the topic itself encompasses the Russia- Ukraine conflict, majority of the sentiments are negative in nature. We found a few positives as well upon close evaluation of the tweets.
- The optimal number of topics for our dataset was found to be 6, with english stop words and perplexity value of 763.59. If we go on increasing the number of topics, there is a possibility that the model will converge to a lower perplexity value. Future scope might include tuning the hyper-parameters of the model and vectorizer to obtain a better converging model.
- We found that when we consider 'Russia' and 'Ukraine' as stop words, the perplexity scores surprisingly increase. We believe that these words add context to the topics and once they are removed, the model finds it little tough to converge as it takes away the similarity or likeness within data.
- We decided to put together the results of both sentiment analysis and topic modeling together to get deeper insights into our analysis. We did so by plotting a heatmap, between topics and Vader sentiments and found that topic 1 is a well balance of both negative and positive sentiments. All other topics are mostly inclined to negative sentiments than positive.

Google drive link – data and scripts

References:

[1] “Global Conflict Tracker - Conflict in Ukraine ”- Updated daily https://www.cfr.org/global-conflict-tracker/conflict/conflict-ukraine

[2] Ehsan-Ul Haq, Gareth Tyson, Lik-Hang Lee, Tristan Braud, and Pan Hui “Twitter Dataset for 2022 Russo-Ukrainian Crisis” , March 6, 2022 , published on Researchgate
https://www.researchgate.net/publication/359079402_Twitter_Dataset_for_2022_Russo-Ukrainian_Crisis

[3] Abdullah Alsaeedi , Mohammad Zubair Khan ,“A Study on Sentiment Analysis Techniques of Twitter Data ”, (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 10, No. 2, 2019
https://thesai.org/Downloads/Volume10No2/Paper_48-A_Study_on_Sentiment_Analysis_Techniques.pdf

[4] Bhumika Gupta, Monika Negi, Kanika Vishwakarma,Goldi Rawat , “Study of Twitter Sentiment Analysis using Machine Learning Algorithms on Python ”, May 2017, International Journal of Computer Applications 165(9):29-34
https://www.researchgate.net/publication/317058859_Study_of_Twitter_Sentiment_Analysis_using_Machine_Learning_Algorithms_on_Python

[5] Gang Li; Fei Liu, “A clustering-based approach on sentiment analysis ” , Published in: 2010 IEEE International Conference on Intelligent Systems and Knowledge Engineering,
DOI: 10.1109/ISKE.2010.5680859
https://ieeexplore.ieee.org/document/5680859

[6] Kim Schouten, Flavius Frasincar , “Survey on Aspect-Level Sentiment Analysis”, January 2015 IEEE Transactions on Knowledge and Data Engineering 28(3):1-1
DOI:10.1109/TKDE.2015.2485209
https://www.researchgate.net/publication/282436170_Survey_on_Aspect-Level_Sentiment_Analysis

[7] Samar Hesham, Khaled Wassif,emad nabil , “Clustering Based Sentiment Analysis Using Randomized Clustering Cuckoo Search Algorithm” , July 2020, published on researchgate.
https://www.researchgate.net/publication/343726430_Clustering_Based_Sentiment_Analysis_Using_Randomized_Clustering_Cuckoo_Search_Algorithm

[8] Twitter, Tweepy official documentation

https://developer.twitter.com/en/docs/twitter-api, https://docs.tweepy.org/en/stable/

[9] Twint official documentation
https://github.com/twintproject/twint

[10] “Russia blocks access to Facebook and Twitter” https://www.theguardian.com/world/2022/mar/04/russia-completely-blocks-access-to-facebook-and-twitter

[11] NLP Application: Named Entity Recognition (NER) in Python with Spacy , Prateek Majumder — June 16, 2021
https://www.analyticsvidhya.com/blog/2021/06/nlp-application-named-entity-recognition-ner-in-python-with-spacy/

[12] A brief intro to NLP and VADER Sentiment Analysis, Aryan Bajaj — June 17, 2021
https://www.analyticsvidhya.com/blog/2021/06/vader-for-sentiment-analysis/

[13] Sentiment Analysis using TextBlob, Parthvi Shah -Jun 27, 2020
https://towardsdatascience.com/my-absolute-go-to-for-sentiment-analysis-textblob-3ac3a11d524

[14] Introduction to Flair for NLP: A Simple yet Powerful State-of-the-Art NLP Library, Sharoon Saxena — February 11, 2019
https://www.analyticsvidhya.com/blog/2019/02/flair-nlp-library-python/#:~:text=Flair%20is%20a%20simple%20natural,deep%20learning%20frameworks%20out

[15] Topic Modeling and Latent Dirichlet Allocation (LDA) using Gensim and Sklearn, seth neha — June 28, 2021
https://www.analyticsvidhya.com/blog/2021/06/part-2-topic-modeling-and-latent-dirichlet-allocation-lda-using-gensim-and-sklearn/#:~:text=Latent%20Dirichlet%20Allocation%20(LDA)%20is,are%20also%20%E2%80%9Chidden%20topics%E2%80%9D.

[16] Latent Dirichlet Allocation by sklearn official documentation
https://scikit-learn.org/stable/modules/generated/sklearn.decomposition.LatentDirichletAllocation.html

[17] Plydavis library official documentation
https://pyldavis.readthedocs.io/en/latest/readme.html