# Part 1: Text Processing and Exploratory Data Analysis

Malena Broner, Francesc Corden and Aleydis Galindo
u188309, u189643 and u189642

GitHub: https://github.com/AleydisGalindo/RIAW-G_101_4.git
TAG: IRWA-2023-Part-1

We are given a document corpus (JSON data file) which is a set of tweets related to the Russo-Ukrainian War. The data includes information such as tweet text, tweet ID, creation date, hashtags, likes, retweets, and URLs.

## PRE-PROCESS OF THE DOCUMENT

Before normalizing the documents, we created the following functions to clean the tweets, such as: remove_emoticons and remove_links. We didn't want to have any emoticons nor links since they are not relevant to the analysis, they even messed it up.

```
def remove_emoticons(text):
    # Define a pattern to find all the emoticons
    emoticon_pattern = re.compile("[" u"\U0001F600-\U0001F64F" u"\U0001F300-\U0001F5FF"
                                  u"\U0001F680-\U0001F6FF" u"\U0001F1E0-\U0001F1FF"
                                  u"\U00002500-\U00002BEF" u"\U00002702-\U000027B0"
                                  u"\U000024C2-\U0001F251" u"\U0001f926-\U0001f937"
                                  u"\U00010000-\U0010ffff" u"\u2640-\u2642"
                                  u"\u2600-\u2B55" u"\u200d"
                                  u"\u23cf" u"\u23e9"
                                  u"\u231a" u"\ufe0f"
                                  u"\u3030" "]+", re.UNICODE)

    # Replace emoticons with an empty string
    text_without_emoticons = emoticon_pattern.sub('', text)

    return str(text_without_emoticons)
```

```
def remove_links(text):
    # Define a pattern to match URLs
    url_pattern = re.compile(r'https?://\S+|www\.\S+')

    # Replace URLs with an empty string
    text_without_links = url_pattern.sub('', text)

    return str(text_without_links)
```

In order to be able to use all the information, we need to pre-process all the data and store it in a dictionary ("tweet_information") with tweet IDs as keys.

To pre-process it, we created a function called “build_terms” that processes each tweet's text by:

- Transforming text to lowercase
- Tokenizing the text into a list of terms
- Removing non-words and non-whitespaces → We decided to exclude the '#' symbol to avoid altering the tweet's intended message, as some users integrated the words into the messaging of their tweets.
- Eliminating stop words
- Performing stemming
- Joining the processed tokens into a string
- Removing double whitespaces

```
def build_terms(line):

    filtered_line = line.lower() ## Transform in lowercase
    filtered_line = filtered_line.split() ## Tokenize the text to get a list of terms
    filtered_line = [re.sub(r'[^\w\s]', '', word) for word in filtered_line] # Removing non-words and non-whitespaces

    # Removing stop words
    stop_words = set(stopwords.words("english"))
    filtered_line = [word for word in filtered_line if word not in stop_words]  ## Eliminate the stopwords

    # Stemming
    stemmer = PorterStemmer()
    filtered_line = [stemmer.stem(word) for word in filtered_line] ## Perform stemming

    # Joining the processed tokens back into a string
    processed_line = ' '.join(filtered_line)

    # Remove double whitespaces
    processed_line = re.sub(" {2,}", " ", processed_line)
    processed_line = re.sub(" +\n", "\n", processed_line).strip()

    return processed_line
```

Finally, tweet IDs are mapped to document IDs for evaluation purposes.

```
# Map tweet IDs with document IDs for evaluation stage
tweet_document_ids_map = {}
with open('PART 1/IRWA_data_2023/Rus_Ukr_war_data_ids.csv', 'r') as map_file:
    doc = csv.reader(map_file, delimiter='\t')
    for row in doc:
        doc_id, tweet_id = row
        tweet_document_ids_map[tweet_id] = doc_id
```

Now to store the information, as we previously mentioned, we created tweet_information with tweet IDs as keys to be able to perform the exploratory data analysis.

```python
# Clean the text
tweet_text = tweet_data['full_text']
tweet_text = remove_emoticons(tweet_text)
tweet_text = remove_links(tweet_text)

# Extract relevant information
tweet_id = tweet_data['id_str']
tweet_date = tweet_data['created_at']
tweet_times.append(datetime.datetime.strptime(tweet_date, '%a %b %d %H:%M:%S %z %Y'))
hashtags = [hashtag['text'] for hashtag in tweet_data['entities']['hashtags']]
likes = tweet_data['favorite_count']
retweets = tweet_data['retweet_count']
twitter_username = tweet_data['user']['screen_name']
tweet_url = f"https://twitter.com/{twitter_username}/status/{tweet_id}"

processed_tweet = build_terms(tweet_text)
```

We first cleaned the documents and then extracted the most relevant information. The most relevant decisions we made in this part were:

- Time → We store this to be able to perform an analysis that sees the evolution of the number of tweets over time.
- Tweet URL → All tweets have the section URL in entities. However, not all of them have a URL. This is why we decided to create the URLs ourselves for all tweets. To do this, we followed the structure:
  https://twitter.com/{twitter_username}/status/{tweet_id}

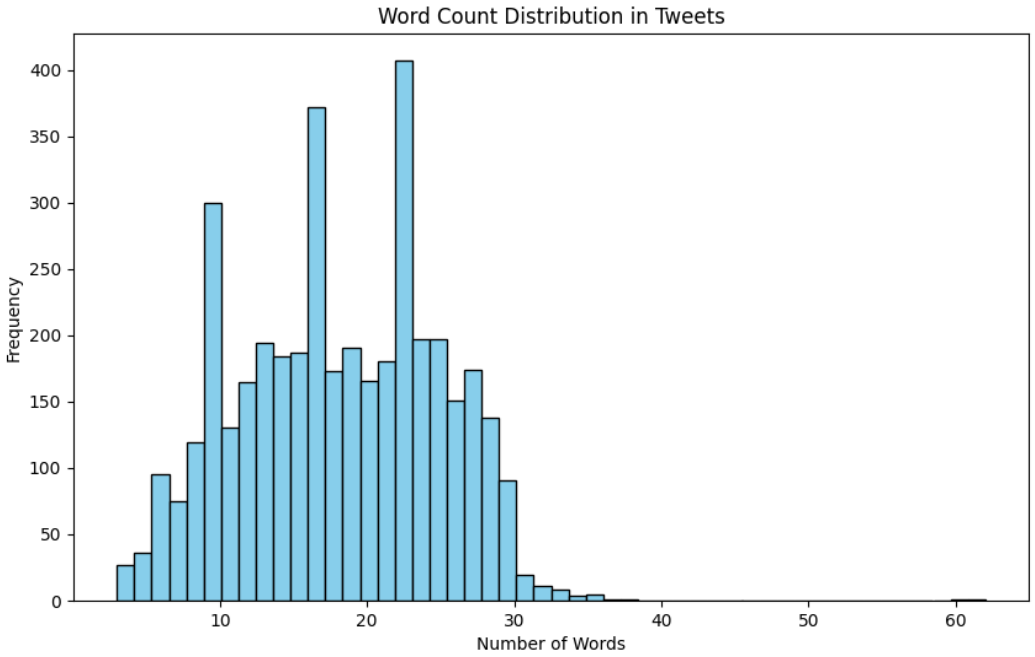
Now we store all the tweet information in the dictionary:

```python
# Store all the tweet information
tweet_information[tweet_id] = {
    'Tweet ID': tweet_id,
    'Tweet Text': tweet_text,
    'Processed Tweet': processed_tweet,
    'Tweet Date': tweet_date,
    'Hashtags': hashtags,
    'Likes': likes,
    'Retweets': retweets,
    'Tweet_url': tweet_url,
    'Entities': entities
}
```

## EXPLORATORY DATA ANALYSIS

The exploratory data analysis provides insights into the characteristics of the Twitter data, including word count distribution, average sentence length, vocabulary size, visualization of word frequency through a word cloud, ranking of tweets most liked and retweeted, entity recognition, and the evolution of the number of tweets over time. The analysis helps in understanding the content and patterns within the dataset.

- **Extract the number of words per tweet**




As we can observe from the previous plot, most tweets have relatively low word count: in the interval of 5 to 30 words with notable peaks at around 10, 18 and 22. This suggests that Twitter users tend to express brief and to the point tweets.

- **Calculate average sentence length**

```
Average Sentence Length: 2.01 sentences
```

The average sentence length can give us a way to quantitatively measure how users structure their messaging. A sentence usually conveys one idea, and additional sentences can be used to complement or challenge the previous.

With the average sentence length being under 3, it seems that Twitter users write short tweets (as understood by the previous analysis) that transmit information that a single sentence is unable to.

- **Extract unique words from all tweets**

```
Vocabulary Size: 8207 words
```

The extraction of unique words from all tweets is a way to obtain the vocabulary size. A larger vocabulary size may be an indicator of a diverse range of topics, since there would be many specific words for each specific topic.

As we can observe, the vocabulary size of our documents (in total) is just 8207 words. This indicates that, as was expected, the tweets are all just about one same topic: the Russo-Ukrainian War.

- **Display top 5 most retweeted tweets and 5 most liked tweets**

```
Top 1 Retweeted Tweet:
Tweet ID: 1575775162674212865
Retweets: 646
Processed Tweet: situat around lyman sep 30 1100 ua forc liber yampil advanc north ru troop re
portedli abandon posit drobyshev exit rout lyman within fire rang ua forc ukrainerussiawar

Top 2 Retweeted Tweet:
Tweet ID: 1575396903252025351
Retweets: 338
Processed Tweet: uniqu rare photo ukrainian forward command post offens kharkiv oblast news re
port arent usual invit place seem except ukrainerussiawar

Top 3 Retweeted Tweet:
Tweet ID: 1575181552170201088
Retweets: 283
Processed Tweet: oper interflex ukrainian recruit continu master skill guidanc british canadia
n instructor uk ukrainerussiawar

Top 4 Retweeted Tweet:
Tweet ID: 1575625313446289409
Retweets: 251
Processed Tweet: follow countri urg citizen leav updat govern make similar statement ukraineru
ssiawar annexationofukrain nafo poland estonia latvia itali unit state bulgaria romania taiwan
 canada portug

Top 5 Retweeted Tweet:
Tweet ID: 1575742923068813314
Retweets: 247
Processed Tweet: russian shell outskirt zaporizhzhia hit civilian humanitarian convoy head tow
ard occupi part 23 peopl kill dozen wound ukrainerussiawar
```

```
Top 1 Liked Tweet:
Tweet ID: 1575775162674212865
Likes: 3701
Processed Tweet: situat around lyman sep 30 1100 ua forc liber yampil advanc north ru troop re
portedli abandon posit drobyshev exit rout lyman within fire rang ua forc ukrainerussiawar

Top 2 Liked Tweet:
Tweet ID: 1575396903252025351
Likes: 2685
Processed Tweet: uniqu rare photo ukrainian forward command post offens kharkiv oblast news re
port arent usual invit place seem except ukrainerussiawar

Top 3 Liked Tweet:
Tweet ID: 1575181552170201088
Likes: 2155
Processed Tweet: oper interflex ukrainian recruit continu master skill guidanc british canadia
n instructor uk ukrainerussiawar

Top 4 Liked Tweet:
Tweet ID: 1575391586422243333
Likes: 1631
Processed Tweet: czech volunt ongo offens ukrainian forc kharkiv oblast ukrainerussiawar

Top 5 Liked Tweet:
Tweet ID: 1575896773511434240
Likes: 1407
Processed Tweet: ukrainian forc liber drobyshev donetsk oblast ukrainerussiawar
```

Identifying the most popular tweets (based on retweets and likes) helps us understand the topics that resonate most with the audience. It is also a way to see which topics have caused more controversy. The more likes or retweets, the more audience the tweets have reached and have been interacted with.

Moreover, it is also a way to possibly know which twitter users have a bigger audience. It is more probable that users with a big audience have a lot of interaction, rather than those “smaller” users.

As we can observe, in our data, the 3 most liked tweets are also the 3 most retweeted tweets. This is to be expected since it is common that “viral” tweets have higher interactions (likes, retweets, comments…).

As we can observe, some of the words in the processed tweets don’t make sense (eg. *situat*, *uniqu*, *forc*, …). This is due to the stemming we previously implemented, which reduces related words or those of the same family to the same root or base form, also known as the ‘stem’, which is beneficial for text processing tasks.

- **Generate and display word cloud**


We have generated a word cloud in order to see the prevalent themes and topics discussed in the tweets. With this, we are able to see what words people tend to use more frequently to express themselves.

As we can observe, the most frequently used words are: *ukrainerussiawar*, *ukrain*, *russia*, *russian*, *putin*, *ukrainewar…* Some of these correspond to hashtags, which makes sense as people tend to widely use them to reach a wider audience.

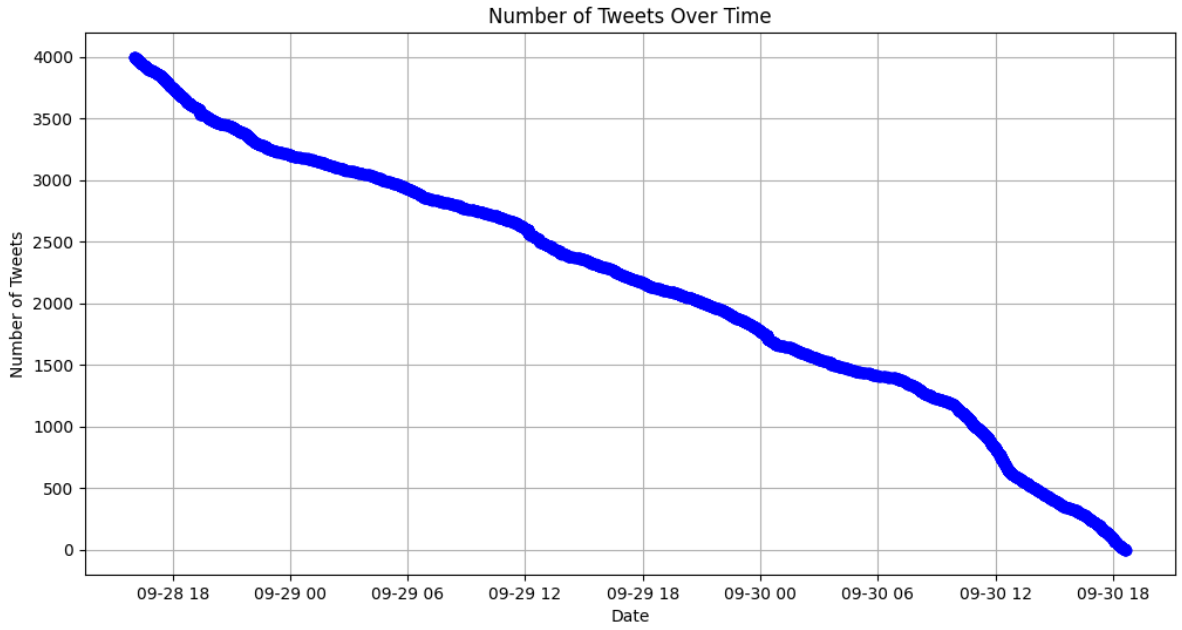
- **Entity recognition**

```
Top 5 Entities:
('russia', 'GPE'): 1777 times
('russian', 'NORP'): 1325 times
('nato', 'ORG'): 359 times
('putin', 'PERSON'): 237 times
('zelenski', 'PERSON'): 190 times
```

We have decided to also implement entity recognition as we believe it's a good way to extract new information. By identifying entities in the tweets, we can gain a better understanding of the key information and relationships within the content. We have displayed the top 5 entities, which correspond to:

- russia → Geopolitical Entity (GPE)
- russian → Nationality of Religious or Political Groups (NORP)
- nato → Organization (ORG)
- putin → Person
- zelenski → Person

- **Plotting the number of tweets over time**




By visualizing the tweet activity over time we are able to identify some patterns or trends.

The dataset covers tweets from Wednesday, September 28th 2022 at 16:03:38 to Friday, September 30th 2022 at 18:39:17. We can observe that at the beginning, there was a substantial surge in tweets (4000) but it decayed quite linearly until there were basically no tweets.

This could potentially be linked with the fact that Tuesday September 27th 2022 was the final day of voting in a series of referendums on joining Russia. After this, more and more people started tweeting about the war. However, as it is expected, news doesn't stay relevant on Twitter for a long period of time. That is why there is a big decline in Twitter activity in the following 2 days.

## REFERENCES

Macias, A., & Ellyatt, H. (2022, September 28). *Nord Stream pipelines hit by suspicious leaks in possible sabotage; Russia says it has "a right" to use nuclear weapons*. CNBC. https://www.cnbc.com/2022/09/27/russia-ukraine-live-updates.html