# Great Learning – Great Lakes Institute of Management

## Final Report

-------------------------------------------------------------------------------------------------------------------------

# Comparison of News Popularity on Different Social Media Platforms

------------------------------------------------------------------------------------------------------------------------

**Authors:**

Aujasvi

Jithin James

Karthik Banka

Karthik Pinnu

**Mentor:**

Mr. Animesh Devarshi

*"Submitted towards partial fulfilment of the criteria forward of PGPDSE by GLIM"*

**March, 2019**

# ACKNOWLEDGEMENT

We express the deep sense of gratitude to our Project Mentor **Mr Animesh Devarshi** for providing precious guidance, inspiring discussion and constant supervision throughout this project. Their timely help, constructive criticism made it possible to present the work contained in this report.

We would like to express my gratitude to members of Great learning for their kind co-operation and encouragement, which help me in completion of this project.

We also thank all the course faculty of the DSE program for providing us a strong foundation in various concepts of analytics and machine learning.

Our thanks and appreciations also go to my colleagues in developing the project and people who have willingly helped me.

# CERTIFICATION OF COMPLETION

I hereby certify that the project titled "Comparison of News Popularity on Different Social Media Platforms" was undertaken and completed under my guidance and supervision by Aujasvi, Jithin James, Karthik Banka, and Karthik Pinnu, students of the Nov 2018 batch of the Post Graduate Program in Data Science Engineering, Bangalore.

**Mr. Animesh Devarshi**

Date: 25th March 2019

**Table of Contents:**

# EXECUTIVE SUMMARY

### Background & Need to study:

The popularity score is basically taken from the platforms like the Facebook, Google plus and LinkedIn. The dataset contains the news items like the title of the news article and headline of the news article. The given dataset contains the Four topics like the Obama, Palestine, Microsoft, Economy. The dataset further contains the Sentiment for the Title and Headline. It has the source which contains the information about the source in which the news article was released. We have several sub files which is related to the main file by the popularity score of the social media platforms in the main file. The subfiles contain the timestamp for each 20 minutes from the point of release of the article. Totally it contains the 144 Timestamps which means that we can know how the Popularity of the news article is changing according to the Timestamps within 2 days.

### Scope & Objectives:

The scope is to compare the popularity of news items on Microsoft versus Palestine. We have to see both types of news have the same popularity across the different platforms. We have to check how soon they are reaching the final level of popularity and also need to find out which words are more likely to create the popularity.

### Approach & Methodology:

We have observed some missing values and we dropped them as it was very few compared to the number of records of the dataset. We have observed some outliers in the social media platforms like Facebook, Google Plus, LinkedIn which we didn't remove them as it was important for further approach. Further we have done EDA in which we have done like Data exploration, Data visualization, Feature Extraction, Topic modelling. Since our target variable are the Social Media Platforms like Facebook, Google Plus, LinkedIn which was a continuous Variable We can build a predictive Model like the Linear Regression or XGBoost or Random Forest Regressor. These Machine Learning models help us to predict the popularity based on the given features and which features are created an impact on the Target Variable. The Model are evaluated using the Performance Metrics like the R2_score, RMSE, Adjusted R2 and the most relevant model is chosen.

### Key Learnings:

Interestingly we observed that the Popularity score for the social media Platforms in the main file are the Final values in the subfiles in their respective Files. Surprisingly we found that the Sentiment Headline, Sentiment Title, Source, Month, Day are not creating any impact on the popularity score of the Social Media Platforms. The Headline which is having the major impact on the popularity score of the Social Media Platforms. Different Models are used to compare the popularity of news items on Microsoft versus Palestine.

# Chapter 1: Project Overview

The study is about the popularity of news items on Microsoft versus Palestine. The data is collected from the different social media platforms and different sources with their respective news articles. To compare the popularity of news items we have to see which words are creating the popularity and how soon they are reaching their final level of popularity and do these topics are having the same behavior across the different platforms.

**Project Objective:** Objective is to compare the popularity of news items on Microsoft versus Palestine.

### Description:

The dataset contains news items and their respective social feedback on multiple platforms such as Facebook, Google Plus and LinkedIn. The collected data relates to a period of 8 months, between November 2015 and July 2016, accounting for about 100,000 news items on four different topics: Economy, Microsoft, Obama and Palestine.

### Domain:

Media and News

### Data Source:

This Data source has been taken from UCI Machine Learning Repository

### Variables of the News Dataset:

- IDLink (numeric): Unique identifier of news items
- Title (string): Title of the news item according to the official media sources
- Headline (string): Headline of the news item according to the official media sources
- Source (string): Original news outlet that published the news item

- Topic (string): Query topic used to obtain the items in the official media sources

- PublishDate (timestamp): Date and time of the news items' publication

- SentimentTitle (numeric): Sentiment score of the text in the news items' title

- SentimentHeadline (numeric): Sentiment score of the text in the news items' headline

- Facebook (numeric): Final value of the news items' popularity according to the social media source Facebook

- GooglePlus (numeric): Final value of the news items' popularity according to the social media source Google+

- LinkedIn (numeric): Final value of the news items' popularity according to the social media source LinkedIn

### Variables of the Social Feedback Data:

- IDLink (numeric): Unique identifier of news items

- TS1 (numeric): Level of popularity in time slice 1 (0-20 minutes upon publication)

- TS2 (numeric): Level of popularity in time slice 2 (20-40 minutes upon publication)

- TS... (numeric): Level of popularity in time slice ...

- TS144 (numeric): Final level of popularity after 2 days upon publication

# Chapter 2: Data Cleaning

## Cleaning up the Data:

We have imported the dataset into the jupyter notebook environment. Initially, the given data set is a combination of six files and a final dataset which contains 11 features and 93239 datapoints. News items has been obtained from different sources like journals, newspapers, television, online sites etc. It may also provide an indication of the item's relevance according to the official media source, denoted by its ranking position. The second, social media, is a medium used to measure the attention received by news items i.e. Popularity.

Here, the dataset consists of four different topics: Obama, Economy, Microsoft and Palestine. Our objective is to compare the popularity of news items based on Microsoft versus Palestine across multiple platforms such as Facebook, Google Plus and LinkedIn. Hence, we separated the rows which are having only the topics like the Microsoft and Palestine and dropped the other respective rows related to Obama and Economy.

We observed that there is a total of 30701 rows and 11 columns after the separation from the original dataset (main file).

## Dealing with missing values:

Missing data in the training data set can reduce the power / fit of a model or can lead to a biased model because we have not analyzed the behavior and relationship with other variables correctly. It can lead to wrong prediction.

let's identify the reasons for occurrence of these missing values. They may occur at two stages:

**1.Data Extraction**: It is possible that there are problems with extraction process. In such cases, we should double-check for correct data with data guardians. Some hashing procedures can also be used to make sure data extraction is correct. Errors at data extraction stage are typically easy to find and can be corrected easily as well.

**2.Data collection:** These errors occur at time of data collection and are harder to correct. They can be categorized in four types:

- **Missing completely at random:** This is a case when the probability of missing variable is same for all observations. For example: respondents of data collection process decide that they will declare their earning after tossing a fair coin. If a head occurs, respondent declares his / her earnings & vice versa. Here each observation has equal chance of missing value.
- **Missing at random:** This is a case when variable is missing at random and missing ratio varies for different values / level of other input variables. For example: We are collecting data for age and female has higher missing value compare to male.

- **Missing that depends on unobserved predictors:** This is a case when the missing values are not random and are related to the unobserved input variable. For example: In a medical study, if a particular diagnostic causes discomfort, then there is higher chance of drop out from the study. This missing value is not at random unless we have included "discomfort" as an input variable for all patients.
- **Missing that depends on the missing value itself:** This is a case when the probability of missing value is directly correlated with missing value itself. For example: People with higher or lower income are likely to provide non-response to their earning.

**Treatment of missing values:**

**Deletion:**

It is of two types: List Wise Deletion and Pair Wise Deletion.

- In list wise deletion, we delete observations where any of the variable is missing. Simplicity is one of the major advantages of this method, but this method reduces the power of model because it reduces the sample size.
- In pair wise deletion, we perform analysis with all cases in which the variables of interest are present. Advantage of this method is, it keeps as many cases available for analysis. One of the disadvantages of this method, it uses different sample size for different variables.
- After checking for the missing values, we have found that there are few missing values in the columns of Headline and Source columns, they are few columns with values. There are 4 missing values in the Headline column and 239 missing values in the Source column. Since, our Dataset is having the 30209 rows, we can drop those rows having the missing values and considering only the remaining data.
- The subfiles contains columns named TS1 to TS144 which denotes the timestamp for each 20 minutes from the point of publication. There are –1 value in these columns, which means on that Timestamp the news has not entered the social media platform.

# Chapter 3: Exploratory Data Analysis

## Exploratory Data Analysis

### Assumptions:

- We have made two assumptions. The articles which are having the final values of popularity on social media platforms as '–1' means those articles are not published in respective platforms because either it is not present in the respective subfiles or if it is present, for all the timestamps it is having '–1' value.

- The articles which are having the final values of popularity on social media platforms having '0' means those articles are present but having no popularity because the timestamps for a particular article in the respective subfiles either it is having '-1' or '0'.

### Data Exploration:

- There are about 707 articles which are related to source Win Beta which was related to the topic Microsoft which is the highest number of articles published among all the sources.

- We observed that the most of the articles are published in the month of March, January and December and very few articles are published in the month of September and August.

- The number of articles published on Monday and Thursday are more as compared to the articles published on Saturday and Sunday.

- In order to get a better understanding of the data we have separated the data which are having the rows which follows a condition like where all the three platforms like Facebook, Google Plus, LinkedIn are having the final values of the news items greater than '0'.

- Surprisingly, there are only 7820 articles which having the popularity among the three platforms which are greater than zero. Further if we deep dive, we found that there are only 32% of the total Microsoft articles which are gaining popularity and, in the Palestine, there are only 8% of the total articles which are gaining popularity.

- Further, when we have divided the dataset having the Final value of Facebook, LinkedIn, Google plus popularity less than or equal to zero. We found that Facebook is having the better popularity than the Google Plus and LinkedIn because Facebook is having less rows which are less than zero rather than Google Plus and LinkedIn.

- In the above divided dataset, interestingly we found that the articles releasing at the time hour zero (midnight) are having no popularity at all, which was around 2000 articles in both the Microsoft and Palestine.

**Data Visualization:**

Data Visualization involves the creation and study of visual representation of the data. It helps to communicate the information clearly and efficiently. We can use various statistical tools plots, graphs to the get the information and insights from the data.

- The sentiment for the title is spread between –0.95 to +0.71 which is the range having most of the data spread between –0.25 and 0.25 approximately.

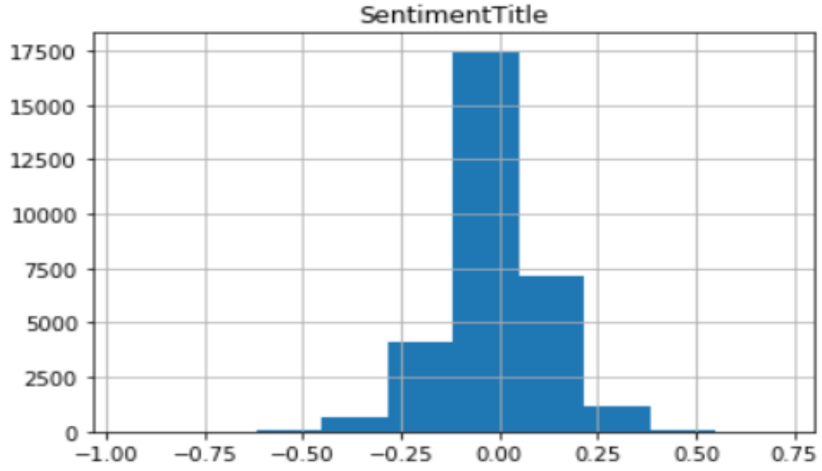


Fig1. Sentiment Title Histogram

- The Sentiment for Headline is spread between –0.73 to +0.9624 which is the range having most of the data spread between –0.35 to 0.26 approximately.

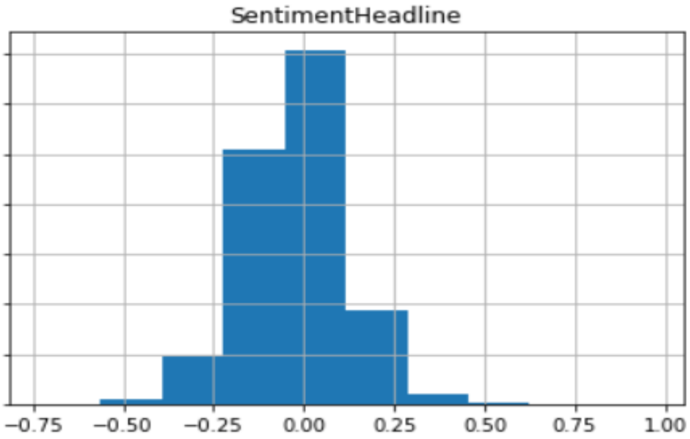


Fig2. Sentiment Headline Histogram

- After separating the dataset from the main dataset. We observed that there are around 21858 news articles for Microsoft and 8843 news articles for the Palestine Topic which is having more no of the news articles are published for Microsoft compared to Palestine in that particular timestamp

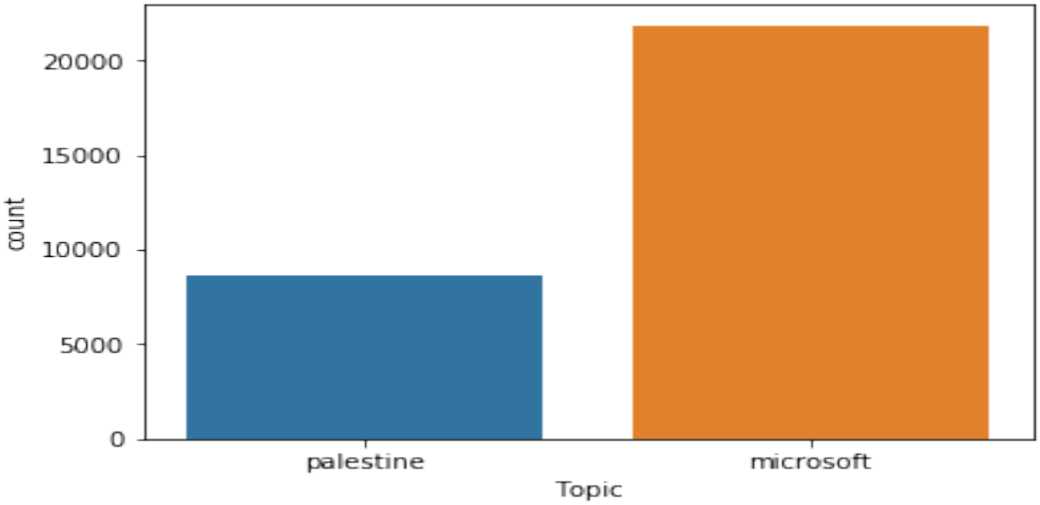


Fig3. Count plot for Topic

- The sentiment for the title Microsoft and Palestine is plotted separately, here data is spread between –0.95 to +0.71 w

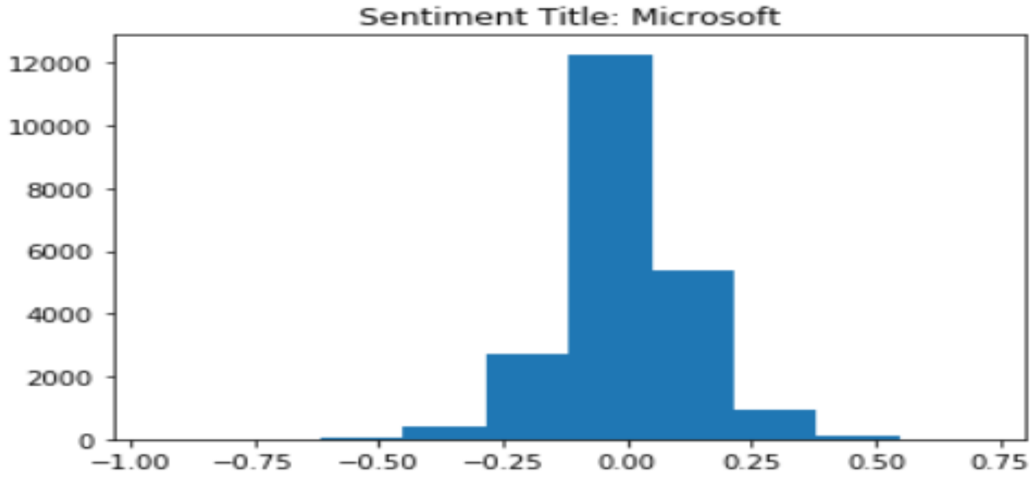


Fig4. Sentiment Title Histogram for Microsoft

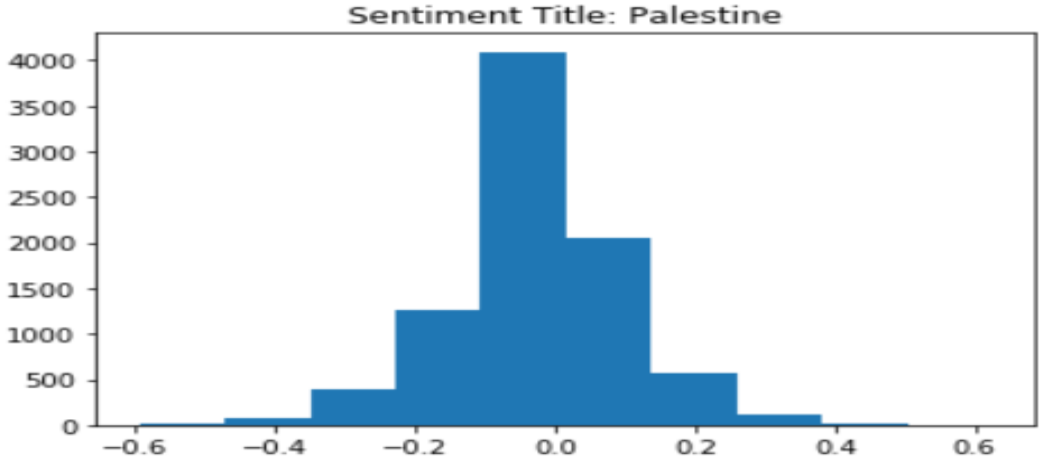


Fig5. Sentiment Title Histogram for Palestine

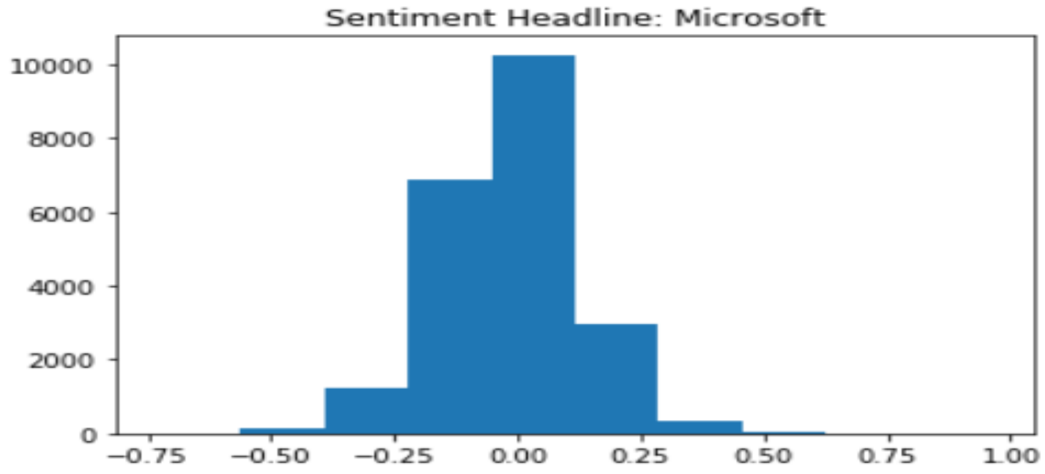


Fig6. Sentiment Headline Histogram for Microsoft

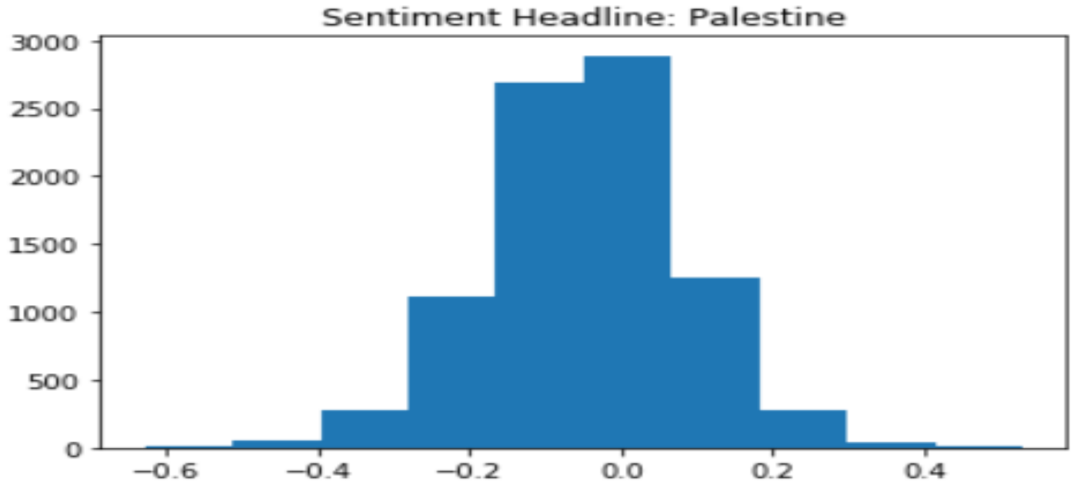


Fig7. Sentiment Headline Histogram for Palestine

- The Following plots shows the number of news published in each month form Nov 2015 to Jul 2016.Here we can see that for the Microsoft there are large number of articles published compared to Palestine and in the month of January there is a slight peak in both the topics. Interestingly both the topics are having a slight dip in the month of February.

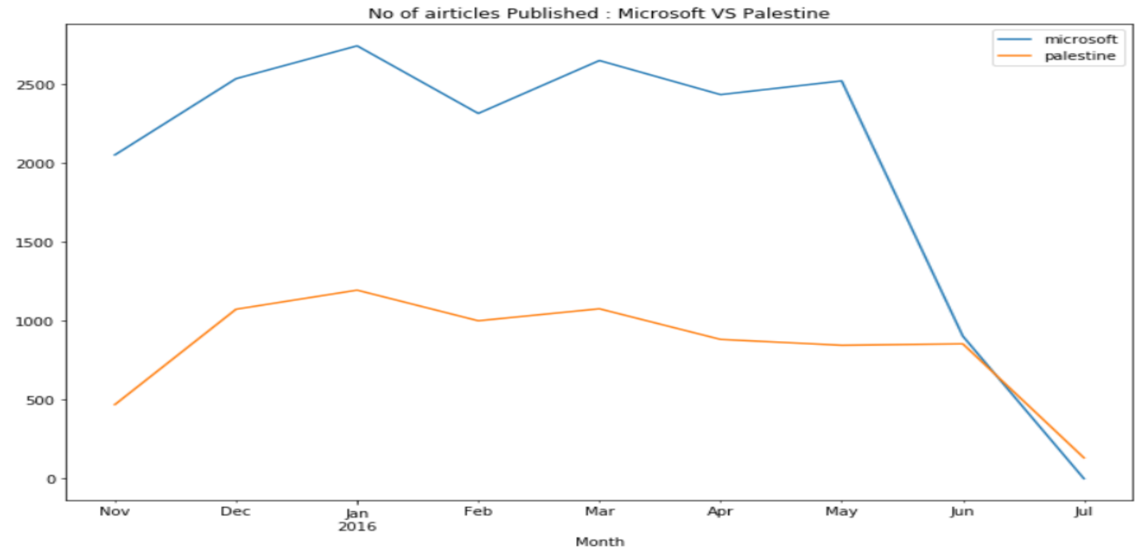


Fig8. No of Articles Published: Microsoft vs Palestine

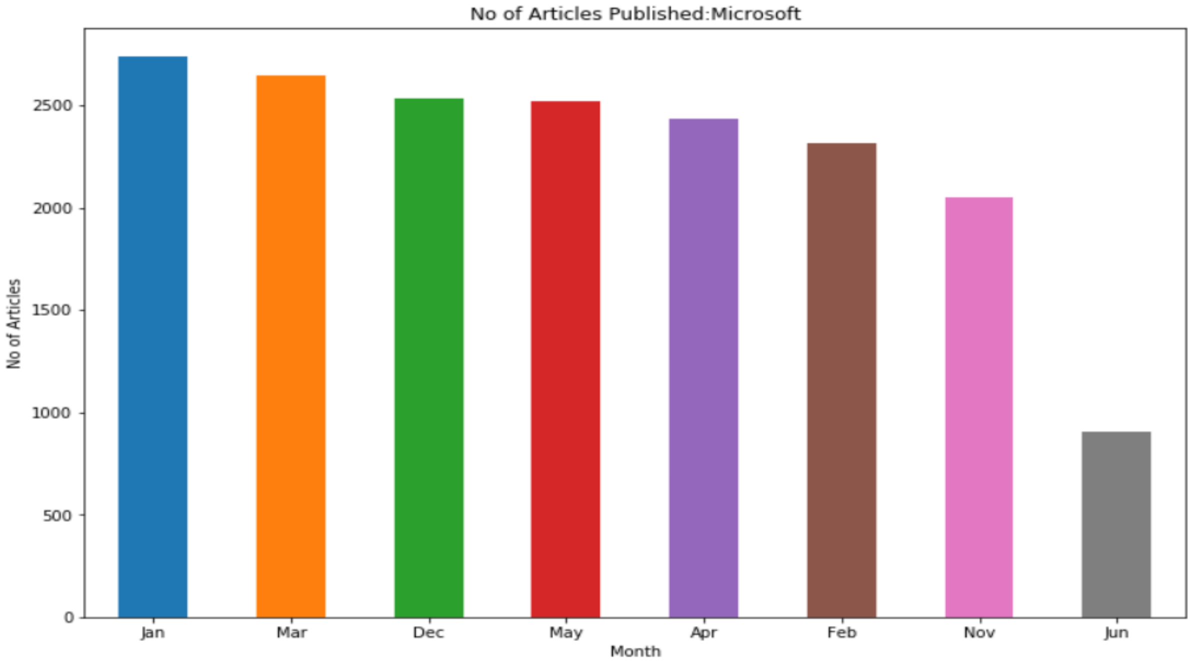


Fig9. No of Articles Published: Microsoft

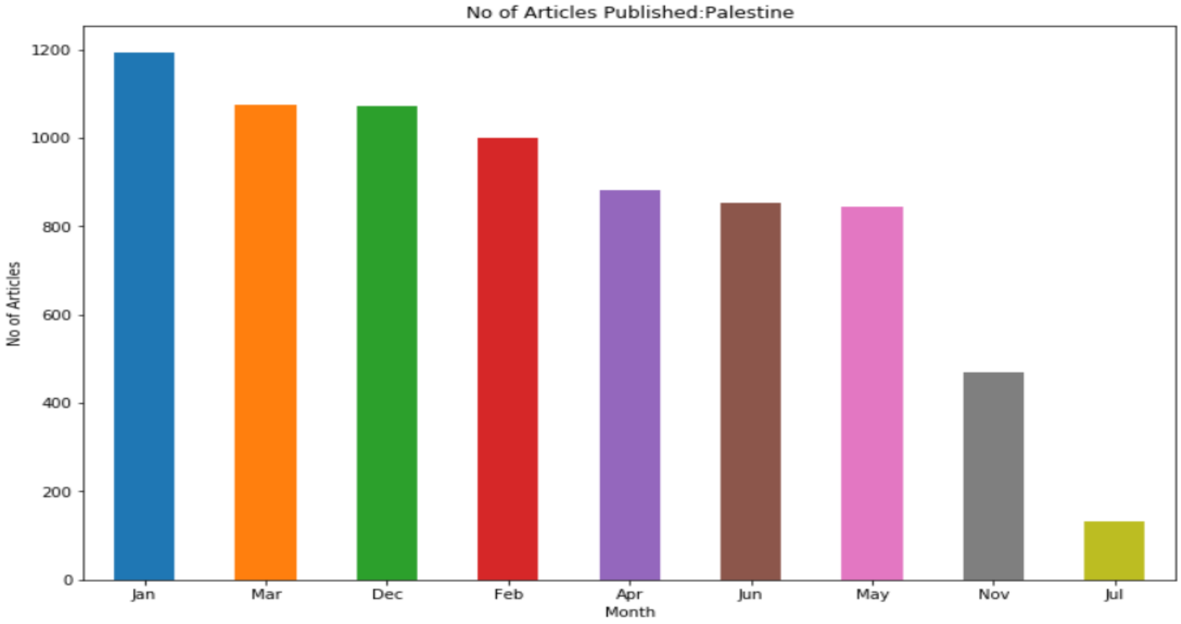


Fig10. No of Articles Published: Palestine

- The Following plot shows the Sentiment Score of Title for Microsoft and Palestine, we can see that Microsoft has a better sentiment score than Palestine

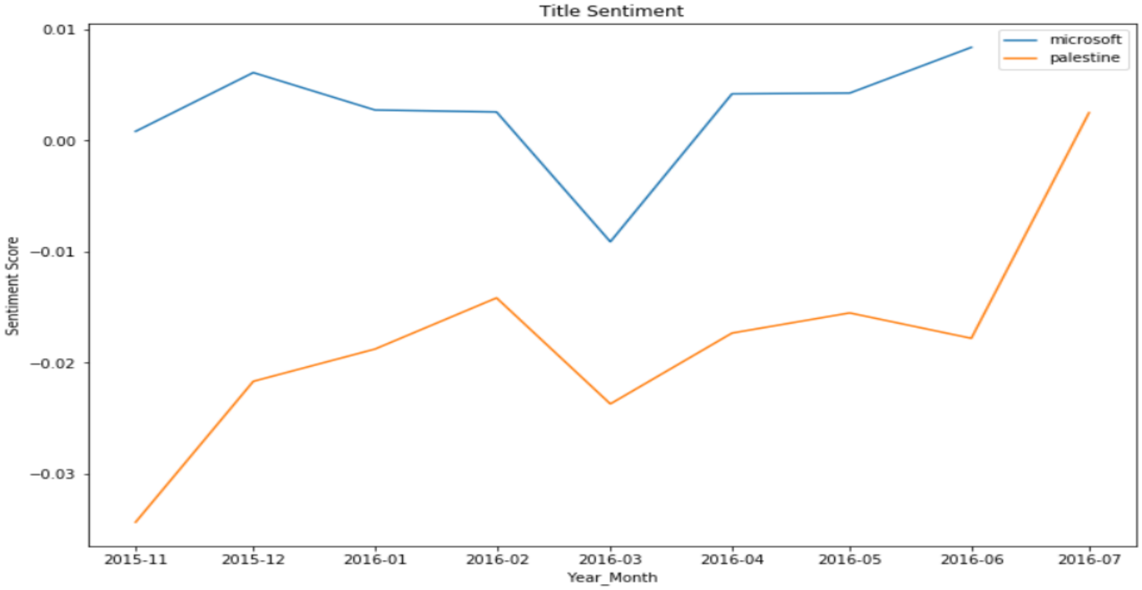


Fig13. Title Sentiment: Microsoft Vs Palestine

- The Following plot shows the Sentiment Score of Heading for Microsoft and Palestine, we can see that Microsoft has a better sentiment score than Palestine

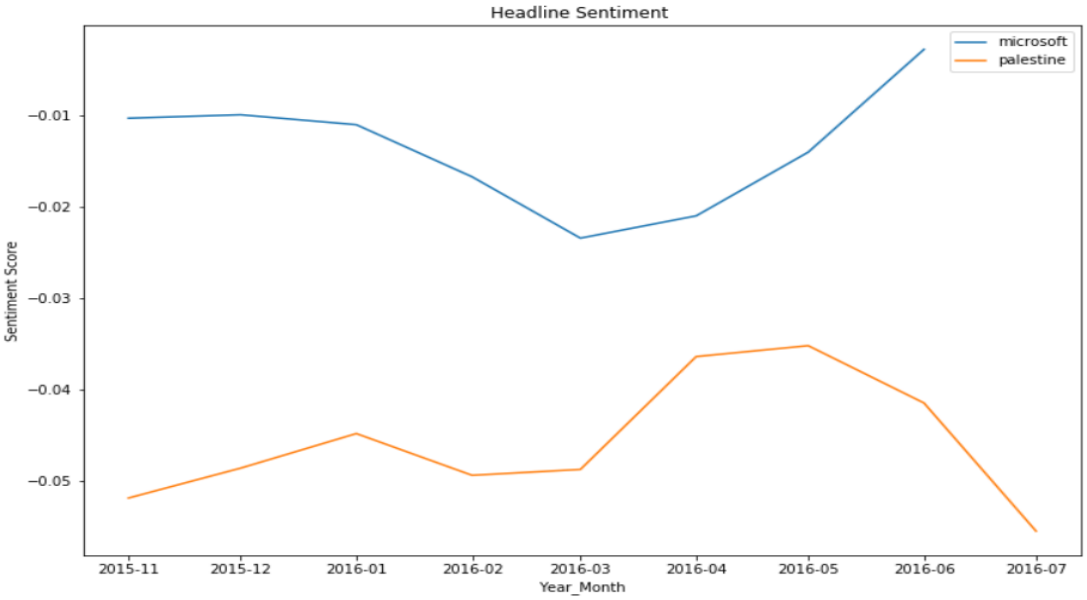


Fig14. Headline Sentiment: Microsoft vs Palestine

- The following plots shows the popularity of Microsoft news and Palestine News, from all the plots we can see that Palestine News has a delay in achieving popularity on all three Social Media Platform i.e. Facebook, LinkedIn, Google Plus. From these plots we can say that on an average the Microsoft is having the better popularity than the Palestine in all the three platforms

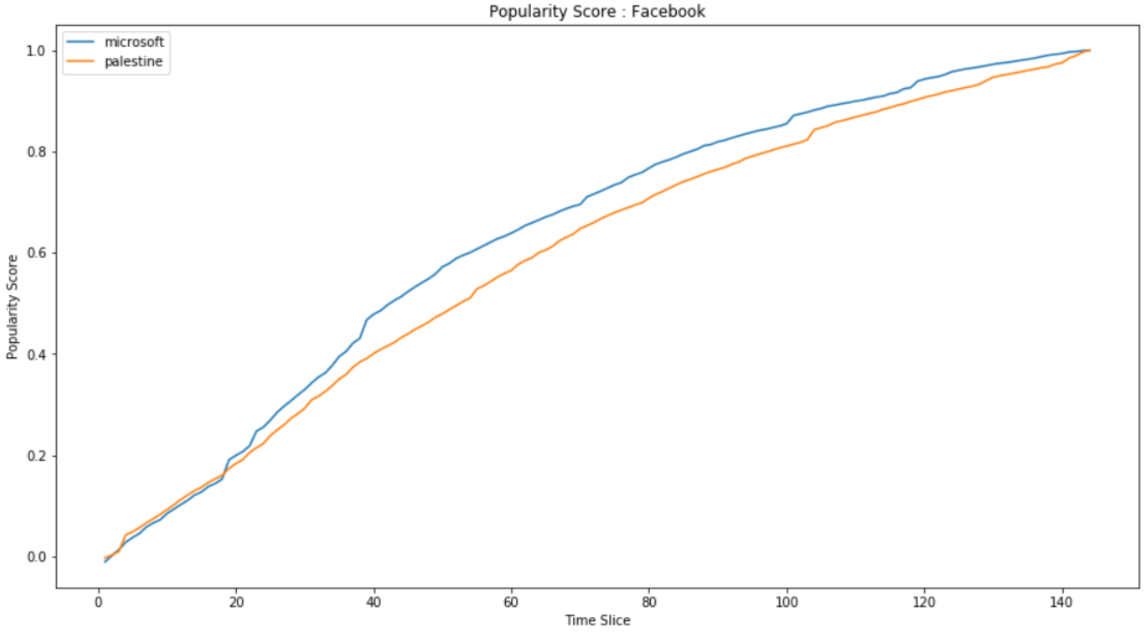


Fig15. Popularity Score: Facebook

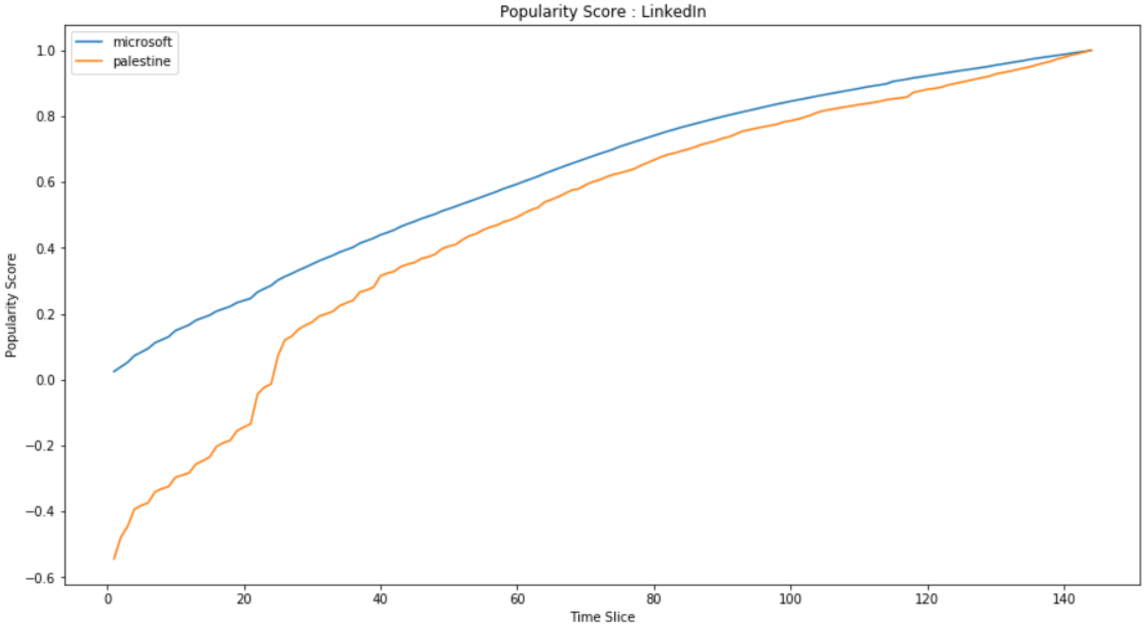


Fig16. Popularity Score: LinkedIn

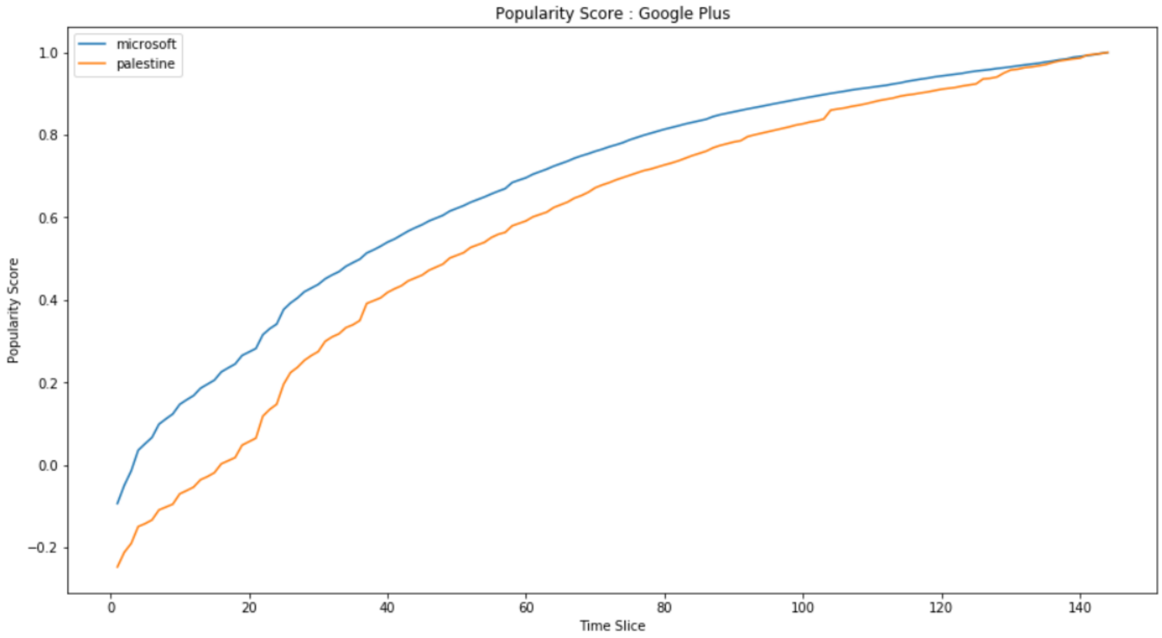


Fig17. Popularity Score: Google Plus

- The Following plots shows as the most commonly used words in Microsoft and Palestine News, in Title and Headline

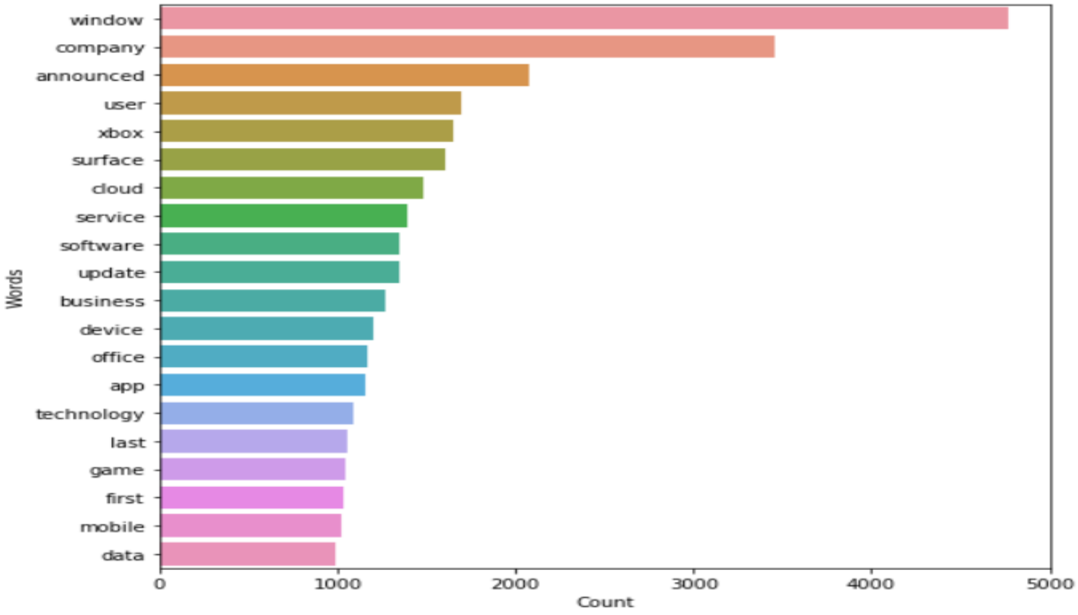


Fig18. Word count in Title for Microsoft

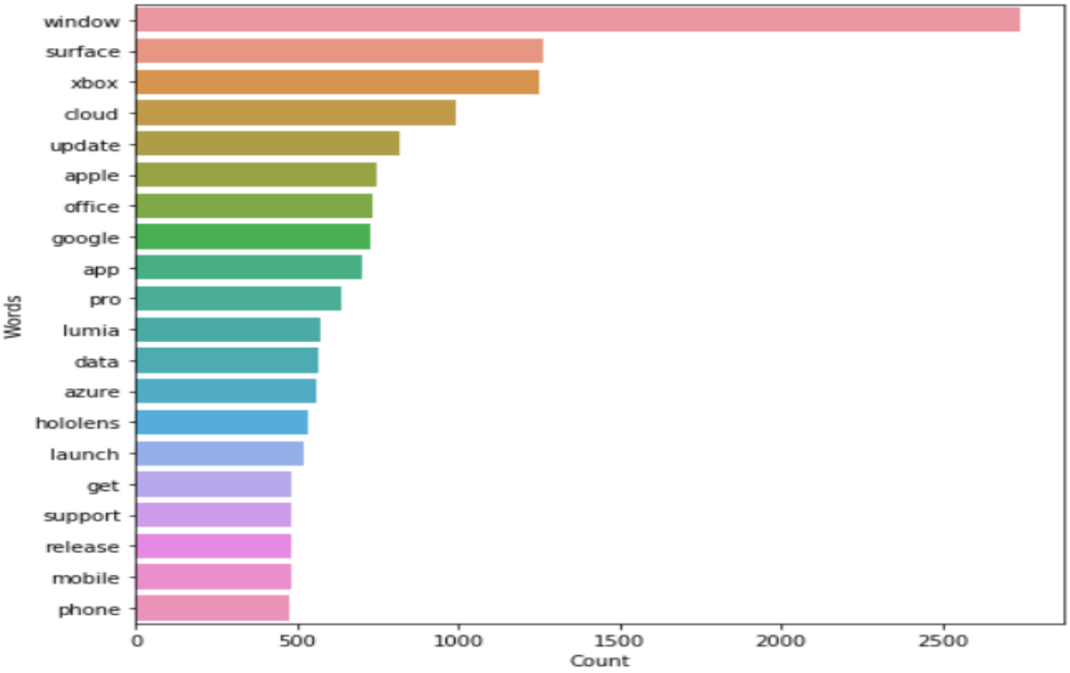


Fig19. Word count in Headline for Microsoft

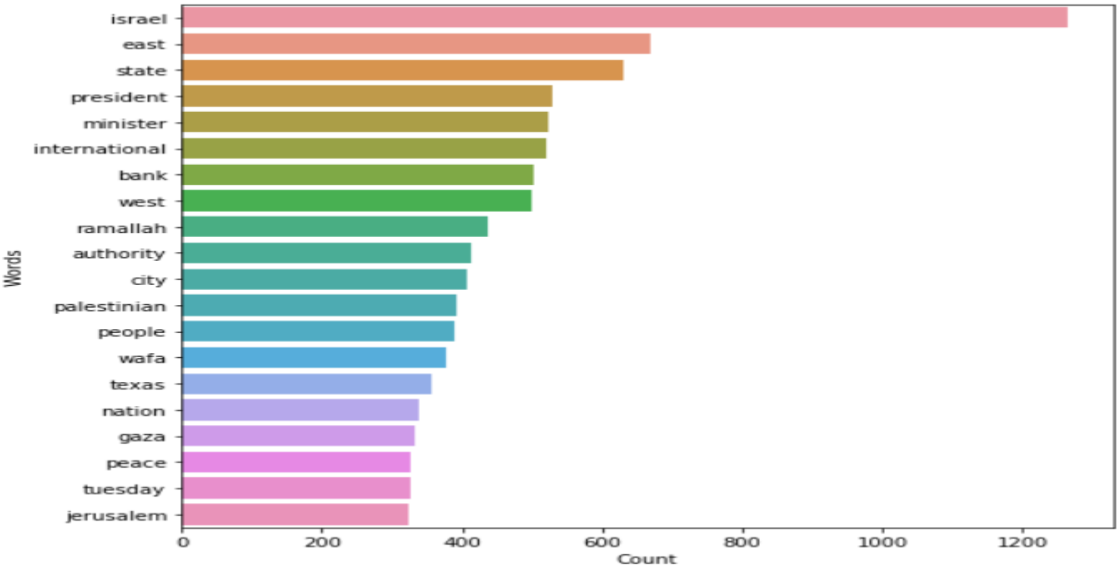


Fig20. Word count in Title for Palestine

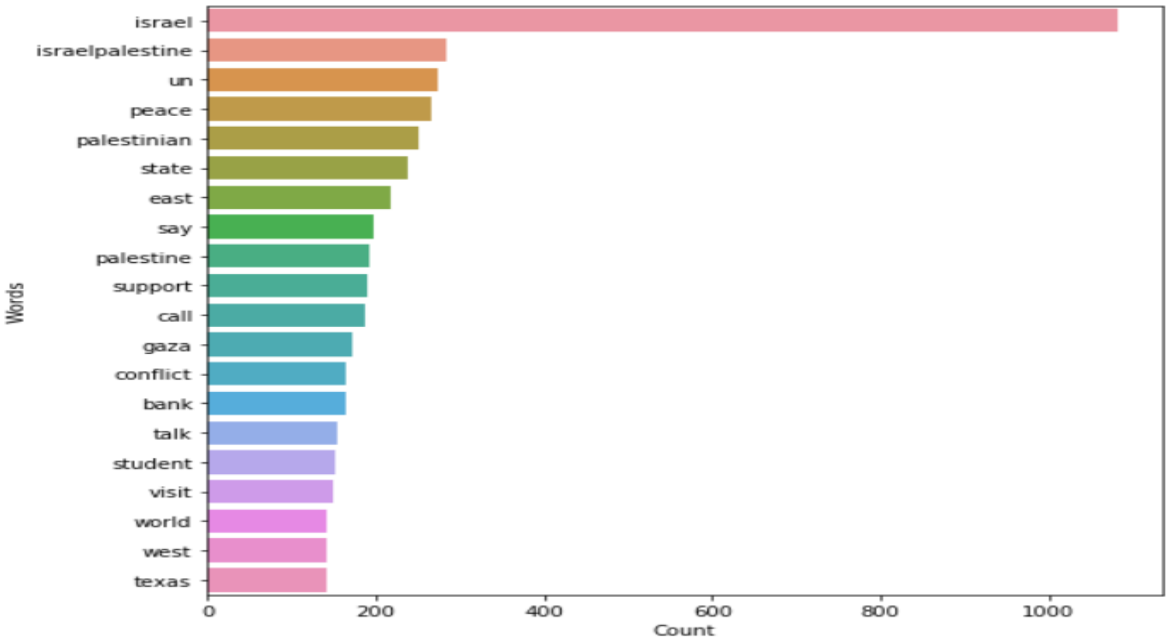


Fig21. Word count in Headline for Palestine

**WordCloud:**

A word cloud is a graphical representation of frequently used words in a collection of text files. The height of each word in this picture is an indication of frequency of occurrence of the word in the entire text.

- We have created word clouds for the title column for topic Microsoft. We found that in the Microsoft Title the words which are very frequently occurring are Xbox, cloud, Microsoft Office, HoloLens, support, Google, Windows mobile, surface pro, Microsoft Lumia

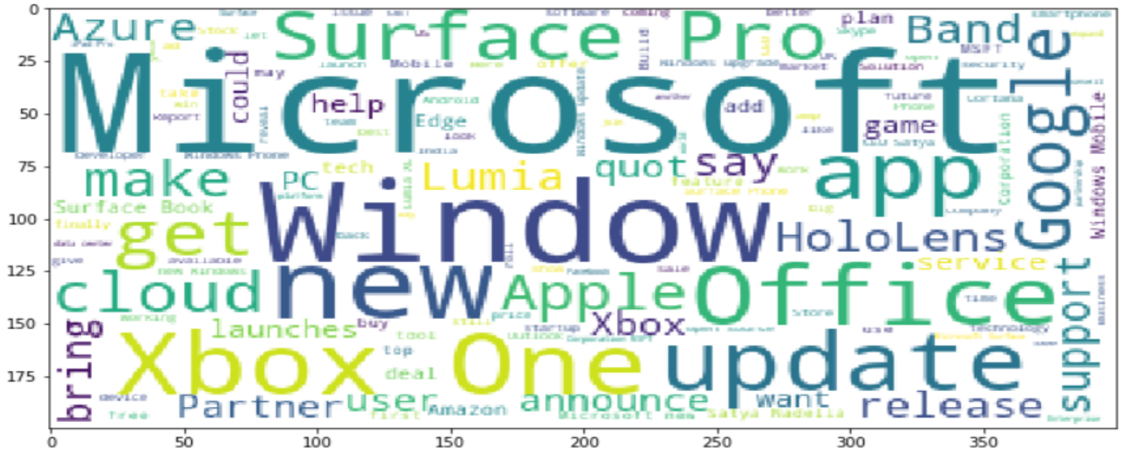


Fig22. Microsoft Title Wordcloud

- Similarly, we also created word clouds for the title for topic Palestine. Words which are very frequently occurring are Israel, Gaza, Palestinian, Israeli, US, UN, Israeli Forces, Israel Conflict, Activist, peace, Syria, Middle East, President, West Bank.

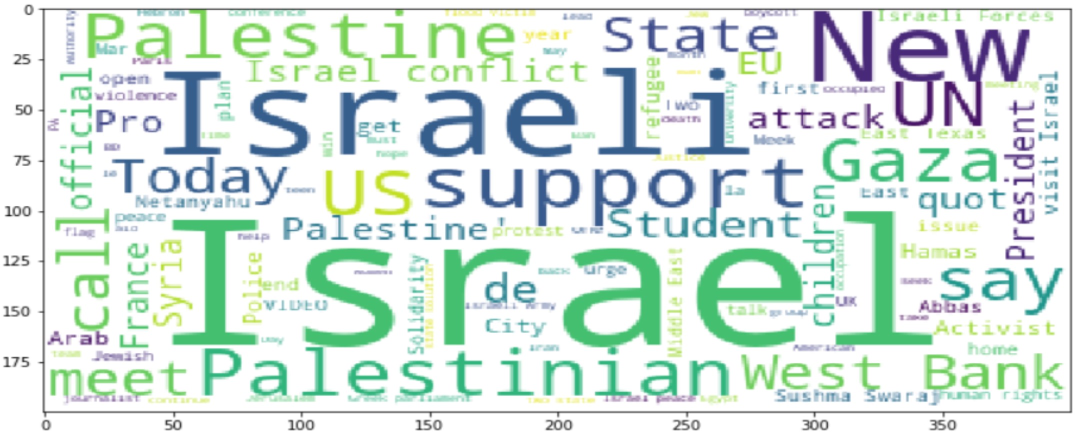


Fig23. Palestine Title Wordcloud

- Also, we have created a word cloud for the Headline column for each topic. We found that in the Microsoft Headline the words which are very frequently occurring are Software, Device, Sathya Nadella, Xbox, Products, Application, Surface Pro, NASDAQ, MSFT, Support, Microsoft.

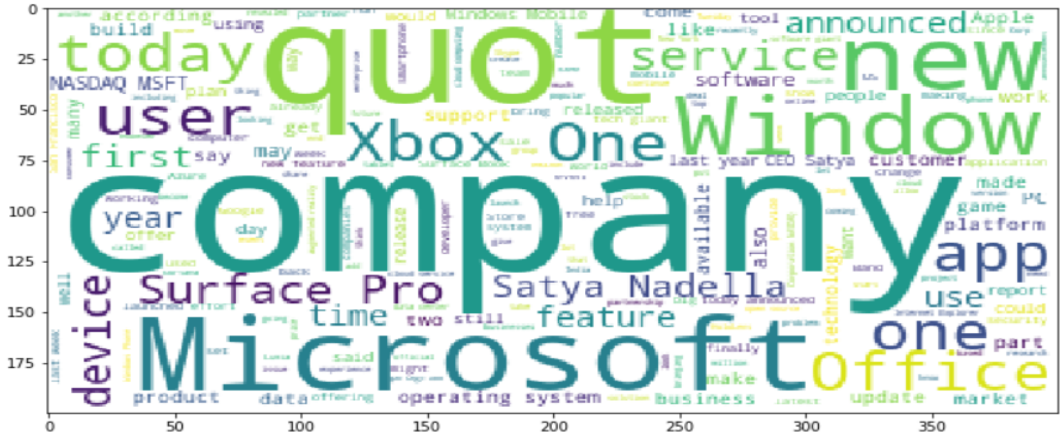


Fig24. Microsoft Headline Wordcloud

- We found that in the Palestine Headline the words which are very frequently occurring are Israel, Palestinian, quot, Middle East, West Bank, Israeli, Mahmoud Abbas, People, United Nation, President, Secretary General, Jewish, Friday Media center, External Affairs, Prime Minister, Gaza, Human Rights, Israeli conflict.

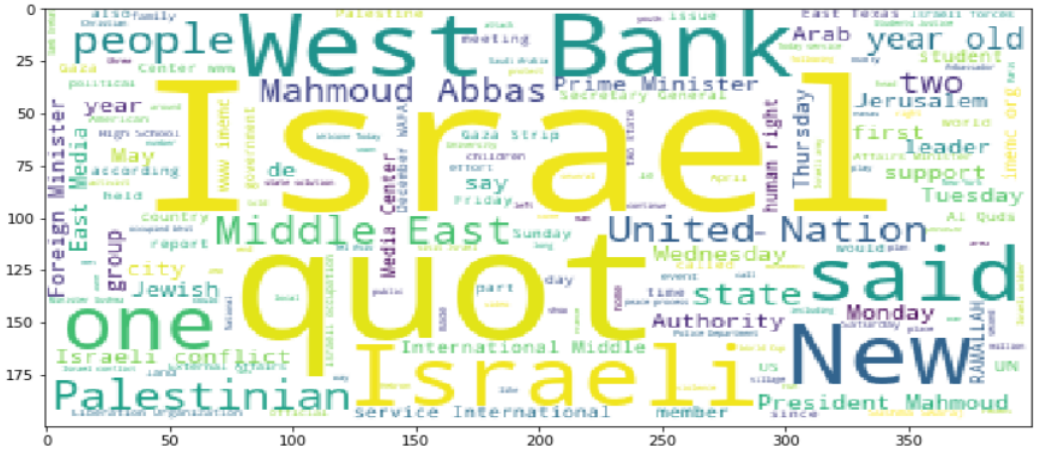


Fig25. Palestine Headline Wordcloud

## Feature Extraction

- As the PublishDate column is of datatype object, we have converted them into the datetime datatype in order to extract the new columns to have the better understanding of the data.
- From the PublishDate column we have extracted the columns like the Date, Month, Year and hour of publication of data.
- As per our objective It is necessary to create the columns like the count, min, max, sum, mean, median for each social media platform so that we can say that for a particular column like Source we can compare which social media platform is having the highest and lowest popularity.

## Hypothesis Testing

We have used Anova and T-Test to find out whether Microsoft news is having high popularity than Palestine News.

**Anova Test:**

The null hypotheses for each of the sets are given below.

1) The population means of the first factor (Platform) are equal.

2) The population means of the second factor (Topic) are equal.

3) There is no interaction between the two factors - (Platform) and (Topic)

Alternative Hypothesis:

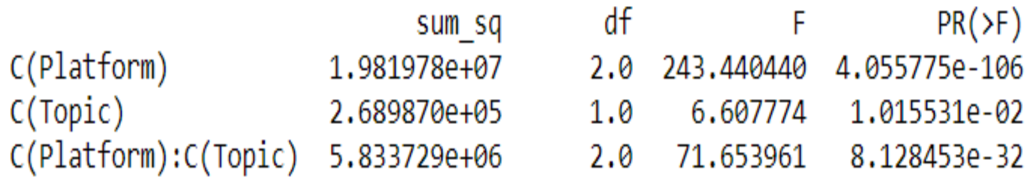
1) The population means of the first factor (Platform) are not equal.

2) The population means of the second factor (Topic) are not equal.

3) There is an interaction between the two factors - (Platform) and (Topic)

We decided the significance level to be 0.05

**Result**



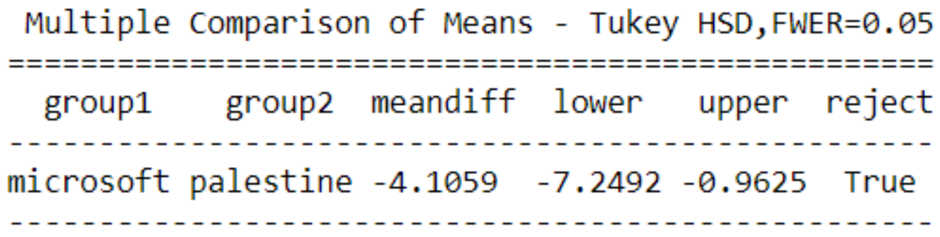
```
                           sum_sq     df           F          PR(>F)
C(Platform)          1.981978e+07    2.0  243.440440  4.055775e-106
C(Topic)             2.689870e+05    1.0    6.607774   1.015531e-02
C(Platform):C(Topic) 5.833729e+06    2.0   71.653961   8.128453e-32
```

p value for Platform is 4.055775e-106 and < 0.05 so we reject the null hypothesis (1) and conclude that the Platform is having an effect on Popularity.

p value for Topic is 1.015531e-02 and < 0.05 so we reject the null hypothesis (2) and conclude that the Topic is having an effect on Popularity.

p value for interaction (Platform: Topic) is 8.128453e-32 and < 0.05 so we reject the null hypothesis (3) and conclude that the interaction (Platform: Topic) is having an effect on Popularity.

We use Tukey-krammer HSD test to detect which mean(s) is / are different.



```
 Multiple Comparison of Means - Tukey HSD,FWER=0.05
=====================================================
  group1    group2   meandiff  lower   upper  reject
-----------------------------------------------------
microsoft palestine -4.1059  -7.2492 -0.9625  True
-----------------------------------------------------
```

**Inference**

Microsoft Vs Palestine: Since the result - reject is true, mean popularity is significantly different between Microsoft news and Palestine News.

We did a T Test to check whether Microsoft is more popular than Palestine

**T-Test (Unpaired):**

Null Hypothesis: Mean Popularity Score of Microsoft<= Mean Popularity Score of Palestine

Alternate Hypothesis: Mean Popularity Score of Microsoft > Mean Popularity Score of Palestine

Significance level was decided to be 0.05

The Above hypothesis was applied to Facebook, Google Plus, LinkedIn

In Facebook the p value is 3.6680742807462585e-08 and < 0.05 so we reject the null hypothesis and conclude Microsoft news is more popular than Palestine in Facebook

In Google Plus the p value is 0.025645880358880848 and < 0.05 so we reject the null hypothesis and conclude Microsoft news is more popular than Palestine in Google Plus

In LinkedIn the p value is 9.661871644071797e-22 and < 0.05 so we reject the null hypothesis and conclude Microsoft news is more popular than Palestine in LinkedIn

# Chapter 4: Modelling Technique

## Modelling Technique

Topic Modelling is a type of statistical modelling for discovering the abstract “topics” that occur in the collection of documents. Here, we are using the LDA (Latent Dirichlet Allocation) model which is used to classify the text in a document to particular topic. It builds a topic per document model and words per document model, modeled as Dirichlet distributions.

## Data Preprocessing for Topic Modelling:

We will perform the following steps:

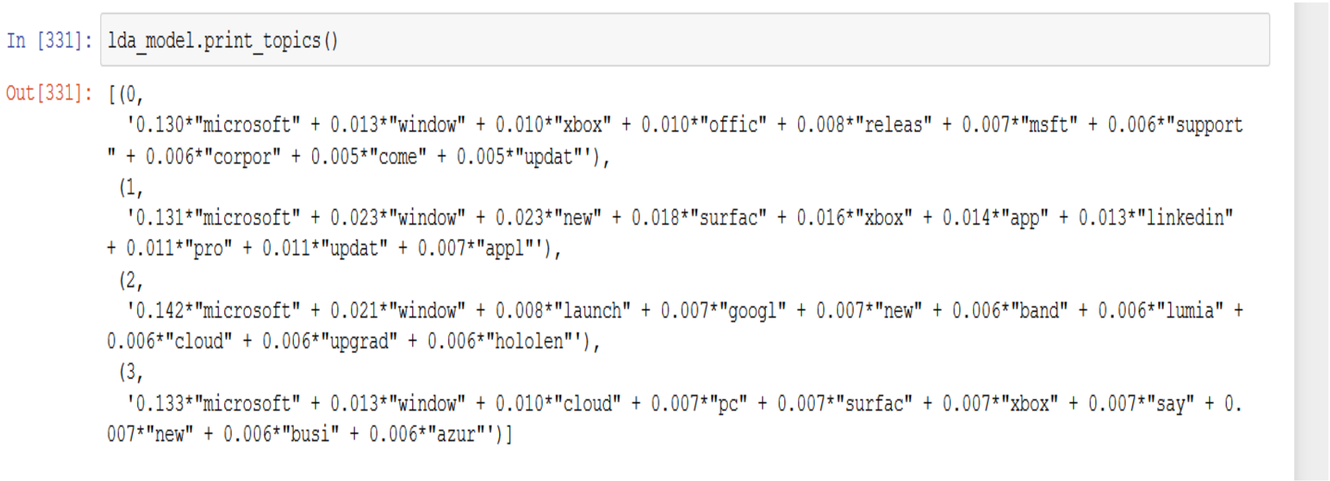
1. Tokenization: Split the text into sentences and the sentences into words. Lowercase the words and remove punctuation.
2. We removed all the stop words.
3. Words are lemmatized—words in third person are changed to the first person and verbs in past and future tenses are changed into the present.
4. Words are stemmed—words are reduced to their root form.

## LDA Model using Bag of Words:

We train our model using **gensim.models.LdaMulticore** and save it lda_model.

For each **Title** related to the topics Microsoft and Palestine, we explore the words occurring in that topic and its relative weights. Also, initially we took 4 topics for the LDA modelling.

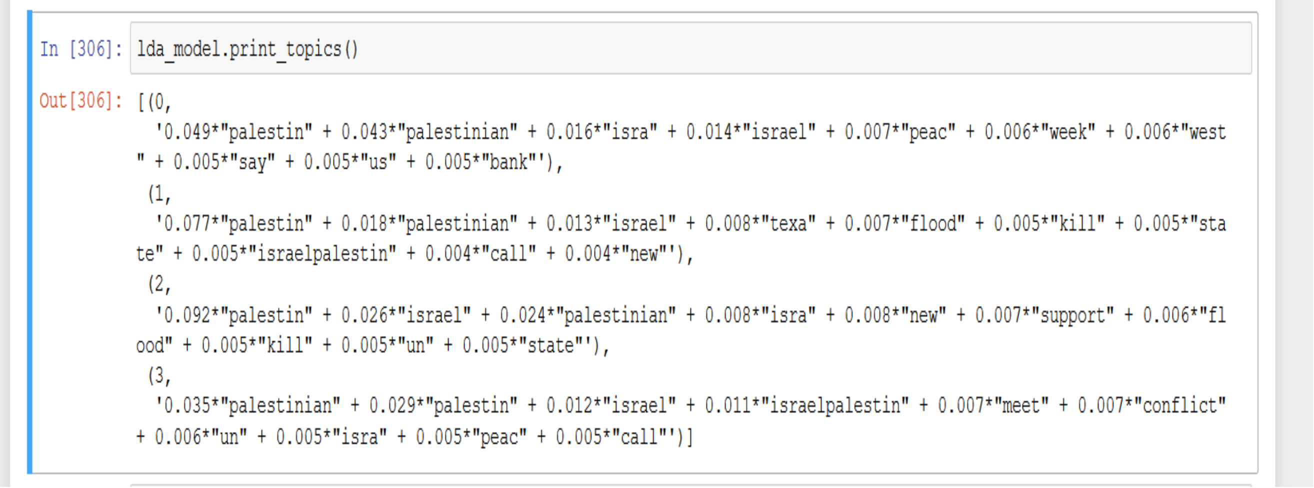
## Microsoft



```
In [331]: lda_model.print_topics()

Out[331]: [(0,
            '0.130*"microsoft" + 0.013*"window" + 0.010*"xbox" + 0.010*"offic" + 0.008*"releas" + 0.007*"msft" + 0.006*"support
          " + 0.006*"corpor" + 0.005*"come" + 0.005*"updat"'),
           (1,
            '0.131*"microsoft" + 0.023*"window" + 0.023*"new" + 0.018*"surfac" + 0.016*"xbox" + 0.014*"app" + 0.013*"linkedin"
          + 0.011*"pro" + 0.011*"updat" + 0.007*"appl"'),
           (2,
            '0.142*"microsoft" + 0.021*"window" + 0.008*"launch" + 0.007*"googl" + 0.007*"new" + 0.006*"band" + 0.006*"lumia" +
          0.006*"cloud" + 0.006*"upgrad" + 0.006*"hololen"'),
           (3,
            '0.133*"microsoft" + 0.013*"window" + 0.010*"cloud" + 0.007*"pc" + 0.007*"surfac" + 0.007*"xbox" + 0.007*"say" + 0.
          007*"new" + 0.006*"busi" + 0.006*"azur"')]
```

### Palestine



```
In [306]: lda_model.print_topics()

Out[306]: [(0,
            '0.049*"palestin" + 0.043*"palestinian" + 0.016*"isra" + 0.014*"israel" + 0.007*"peac" + 0.006*"week" + 0.006*"west
          " + 0.005*"say" + 0.005*"us" + 0.005*"bank"'),
           (1,
            '0.077*"palestin" + 0.018*"palestinian" + 0.013*"israel" + 0.008*"texa" + 0.007*"flood" + 0.005*"kill" + 0.005*"sta
          te" + 0.005*"israelpalestin" + 0.004*"call" + 0.004*"new"'),
           (2,
            '0.092*"palestin" + 0.026*"israel" + 0.024*"palestinian" + 0.008*"isra" + 0.008*"new" + 0.007*"support" + 0.006*"fl
          ood" + 0.005*"kill" + 0.005*"un" + 0.005*"state"'),
           (3,
            '0.035*"palestinian" + 0.029*"palestin" + 0.012*"israel" + 0.011*"israelpalestin" + 0.007*"meet" + 0.007*"conflict"
          + 0.006*"un" + 0.005*"isra" + 0.005*"peac" + 0.005*"call"')]
```

### Inference:

In Microsoft, Topic 0 words includes words like “window”, “Xbox”, “office”, “support”, “corporation” which sounds like the topic is related to the products by Microsoft. Topic 1 includes words like “window”, “app”, “LinkedIn”, “update”, it seems this topic is related to software and applications. Topic 3 includes words like “launch”, “new”, “Lumia”, “cloud”, “upgrade”, it is definitely related to the mobile category. Similarly, Topic 4 includes the words like “pc”,” surface”, “azure”, “Xbox”, “business”, “new” which seems to be their business for new products.

In Palestine, Topic 0 words includes words like “Palestine”, “Israel”, “peace”, “week”, “us” which sounds like the topic is related to the Peace. Topic 1 includes words like “state”, “flood”, “kill”, “israelpalestine”, it seems this topic is related to calamity. Topic 3 includes words like “new”, “support”, “state”, it is definitely related to the support. Similarly, Topic 4 includes the words like “meet”,” conflict”, “peace” which seems to related to war and conflict.

### TF*IDF

TF*IDF is an information retrieval technique that weighs a term’s frequency (TF) and its inverse document frequency (IDF). Each word or term has its respective TF and IDF score. The product of the TF and IDF scores of a term is called the TF*IDF weight of that term.

Put simply, the higher the TF*IDF score (weight), the rarer the term and vice versa.

The TF*IDF algorithm is used to weigh a keyword in any content and assign the importance to that keyword based on the number of times it appears in the document. More importantly, it checks how relevant the keyword is throughout the web, which is referred to as corpus.

For a term t in a document d, the weight Wt,d of term t in document d is given by:

$$Wt,d = TFt,d \log (N/DFt)$$

Where:

TFt,d is the number of occurrences of t in document d.

DFt is the number of documents containing the term t.

N is the total number of documents in the corpus.

**Count Vectorizer**

The Count Vectorizer provides a simple way to both tokenize a collection of text documents and build a vocabulary of known words, but also to encode new documents using that vocabulary.

We can use it as follows:

Create an instance of the CountVectorizer class.

Call the fit() function in order to learn a vocabulary from one or more documents.

Call the transform() function on one or more documents as needed to encode each as a vector.

An encoded vector is returned with a length of the entire vocabulary and an integer count for the number of times each word appeared in the document.

# Chapter 5: Model Comparison

## Model Comparison

The following table shows the R2 Score of Various models created to identify those words which have high popularity, we only used one TF_IDF model because it gave us a low R2 Score.

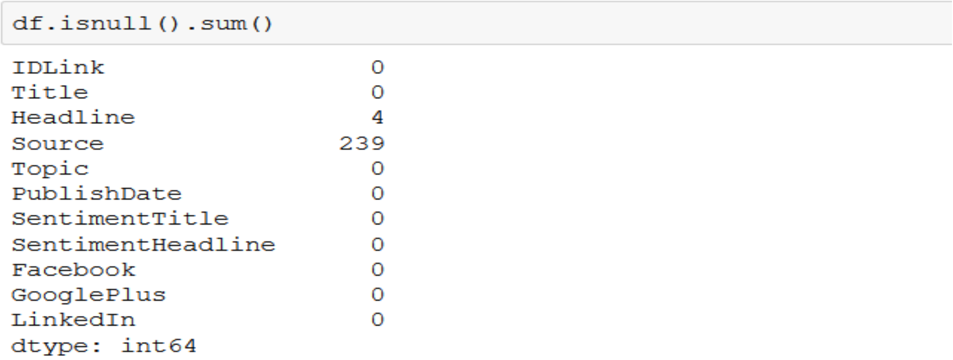
| Topic | Platform | Technique | Model | R2_Value |
|---|---|---|---|---|
| Microsoft | Facebook | TF_IDF Vectorizer | Linear Regression | 0.01 |
| Microsoft | Facebook | Count Vectorizer | Linear Regression | 0.74 |
| Microsoft | Facebook | Count Vectorizer | XGBoost | 0.74 |
| Microsoft | Google Plus | Count Vectorizer | Linear Regression | 0.85 |
| Microsoft | Google Plus | Count Vectorizer | XGBoost | 0.45 |
| Microsoft | LinkedIn | Count Vectorizer | Linear Regression | 0.72 |
| Microsoft | LinkedIn | Count Vectorizer | XGBoost | 0.8 |
| Palestine | Facebook | Count Vectorizer | Linear Regression | 0.95 |
| Palestine | Facebook | Count Vectorizer | XGBoost | 0.66 |
| Palestine | Google Plus | Count Vectorizer | Linear Regression | 0.91 |
| Palestine | Google Plus | Count Vectorizer | XGBoost | 0.56 |
| Palestine | LinkedIn | Count Vectorizer | Linear Regression | 0.99 |
| Palestine | LinkedIn | Count Vectorizer | XGBoost | 0.89 |

## Popular Words in Each Platform

The following table shows the words which caused the most popularity.

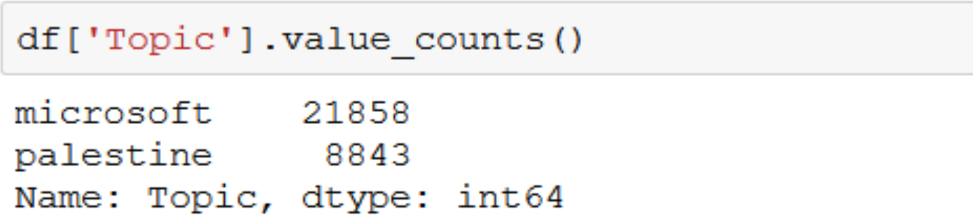
| Topic | Platform | Words |
|---|---|---|
| Microsoft | Facebook | Crossnetwork, Corrupt, Teen |
| Microsoft | Google Plus | Shell, SocialNetwork, Ubuntu |
| Microsoft | LinkedIn | Workfocus, Escrow, Account |
| Palestine | Facebook | React, American, Mastermind |
| Palestine | Google Plus | Uproar, American, React |
| Palestine | LinkedIn | American, uproar, Coastal |

## Appendix:



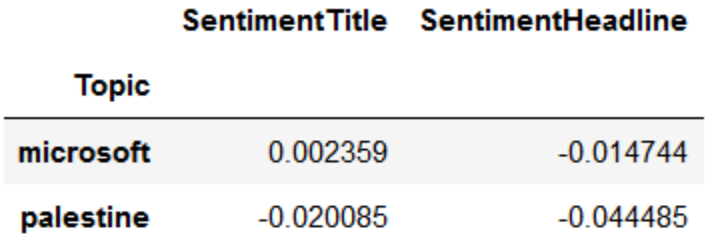
```
df.isnull().sum()
```

```
IDLink                 0
Title                  0
Headline               4
Source               239
Topic                  0
PublishDate            0
SentimentTitle         0
SentimentHeadline      0
Facebook               0
GooglePlus             0
LinkedIn               0
dtype: int64
```

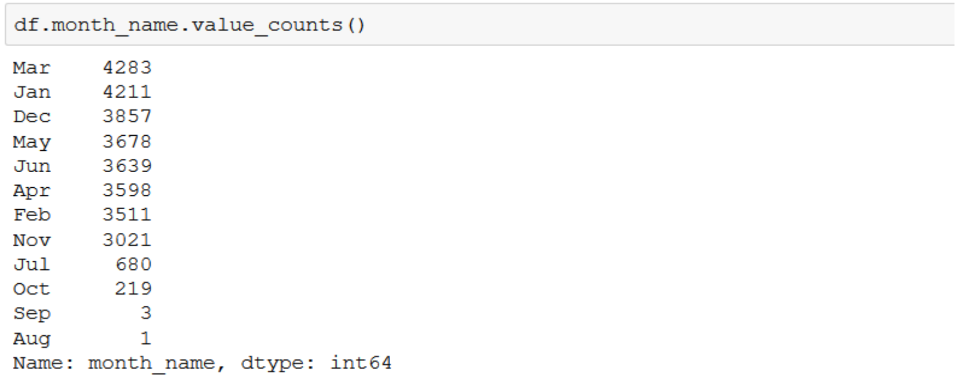


```
df['Topic'].value_counts()
```

```
microsoft     21858
palestine      8843
Name: Topic, dtype: int64
```

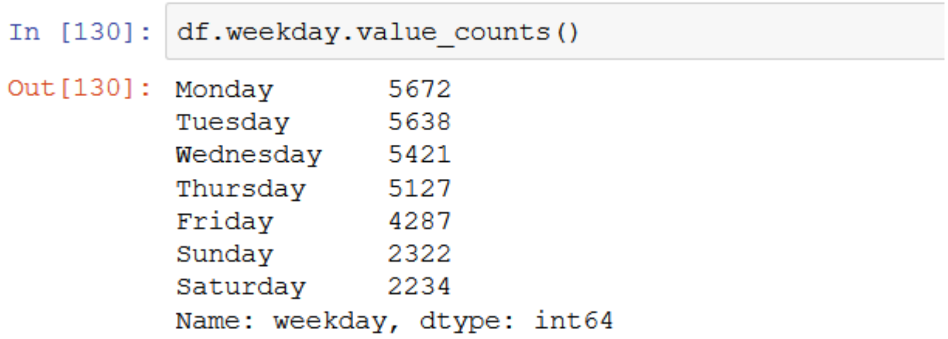


|  | SentimentTitle | SentimentHeadline |
|---|---|---|
| **Topic** |  |  |
| **microsoft** | 0.002359 | -0.014744 |
| **palestine** | -0.020085 | -0.044485 |



```
df.month_name.value_counts()
```

```
Mar    4283
Jan    4211
Dec    3857
May    3678
Jun    3639
Apr    3598
Feb    3511
Nov    3021
Jul     680
Oct     219
Sep       3
Aug       1
Name: month_name, dtype: int64
```



```
In [130]: df.weekday.value_counts()

Out[130]: Monday       5672
          Tuesday      5638
          Wednesday    5421
          Thursday     5127
          Friday       4287
          Sunday       2322
          Saturday     2234
          Name: weekday, dtype: int64
```

articlesg0=combined.loc[((combined.Facebook > 0) & (combined.LinkedIn >0) &(combined.GooglePlus > 0)),:]*#segregating the articles having the popularity greater than zero in all social media platforms*articlesg0.shape*#ARTICLES HAVING THE POPULARITY GREATER THAN ZERO IN ALL SOCIAL MEDIA PLATFORMS*(7820, 17)

articlesg0['Topic'].value_counts()[0]/combined['Topic'].value_counts()[0]*#out of all articles only 32% are having the popularity greater than zero in all the three platforms for Microsoft*0.32419568898448586

articlesg0['Topic'].value_counts()[1]/combined['Topic'].value_counts()[1]*#out of all articles only 8.5% are having the popularity greater than zero in all the three platforms for Palestine*0.08551179272685024

Articlesl0.shape *#articles having the popularity less than zero for the Facebook*(11193, 18)

```
Articlesl0g.shape #articles having the popularity less than zero for the Googleplus(18377, 18)
Articlesl0l.shape #articles having the popularity less than zero for the LinkedIn(17807, 18)
articlesl0g.time_hour.value_counts(). head() #articles releasing at time hour zero
0 2240
```

**References:**

- [1] Nuno Moniz and LuÃ-s Torgo (2018), Multi-Source Social Feedback of Online News Feedsâ€, CoRR
- [2] Susan Li (March 30, 2018). Topic Modelling in Python with NLTK and Gensim [Blog Post]

Retrieved From: https://towardsdatascience.com/topic-modelling-in-python-with-nltk-and-gensim-4ef03213cd21