**Data mining – Final project**

**Presenter**

**Tal Tubul**

**I.D 208835355**

1. **Pre-processing**

For start I uploaded the WhatsApp chat group.

Then I started Extract features from the text, I started by extracting the features (cell 2.a):

* Date – The day and month that the message sanded.
* Time – The hour, minutes and seconds the message sanded.
* Message – The message contact.
* Name – Message sender nickname.

Then I split the date feature into month and day features and changed the time feature into hour feature (cell 2.b).

That's a photo of the first data frame:

|  | **name** | **month** | **day** | **hour** | **message** |
| --- | --- | --- | --- | --- | --- |
| 0 | כו"ח - על כל שאלה תשובה 🎼 | 7 | 11 | 6 | ההודעות והשיחות מוצפנות מקצה לקצה. לאף אחד מחו... |
| 1 | רועי מרלי | 8 | 21 | 17 | סקר:\nחברת \*שידורית\* זכתה במכרז של משרד הרווחה... |
| 2 | יעל גורן (מפילדלפיה) | 10 | 8 | 22 | האם יש אפשרות באפליקציה לשנות את צליל ההתראה מ... |

**Data handling**

Now that I have a data frame, I'll start to clear the data from noises and outliers.

* First, I discovered that some of the values at the name feature has the string: "מצורף" so I removed it (cell 3.b).
* I also discovered that some of the name feature values has the message contact inside it, so I split it and added the message contact to the message feature (cell 3.c).
* The name feature is now ready, and the message feature will be cleared of noises by eliminating messages with no meaning, for example, messages of images, audios and videos was removed (cell 3.d)
* In addition to cleaning the messages from noise, I also removed links and email addresses from the message and only left the link's name to the website or the email address, for example, a link for Facebook like that: https://www.facebook.com/profile.php?id=61552570571876 replaced to be L\_facebook. I added a Risha to the links to create a difference between a link and just a mention of Facebook, the company, or any other link or email address (cell 3.e).

The hour feature values range from 0-23, so I aim to minimize the range to 1-4 by the next separation:

0-6 AM: 1

6-12 AM: 2

12-18 PM: 3

18-23 PM: 4

* I added three columns of normalize numerical features: month, day, hour. This step will help me later to create PCA model and unsupervised learning (cell 3.g).

**Outliers**

* For start I assumed that the chat group has activities users and non-activities users, I grouped the data frame by the unique nicknames at the name feature (cell 4.a).

תמונה שמכילה טקסט, קו, עלילה, תרשים

התיאור נוצר באופן אוטומטיI created a plot bar with the number of messages that each user where send (cell 4.b):

We can see from the plot that most activities user are likely to send more than 10 messages, but non-activities users are more likely to send less than 5 messages.

I removed the non-activities users and created a new data frame (cell 4.c).

* I removed days where only one message was sent, assuming it wasn't responded by any other member of the chat group (cell 4.e).

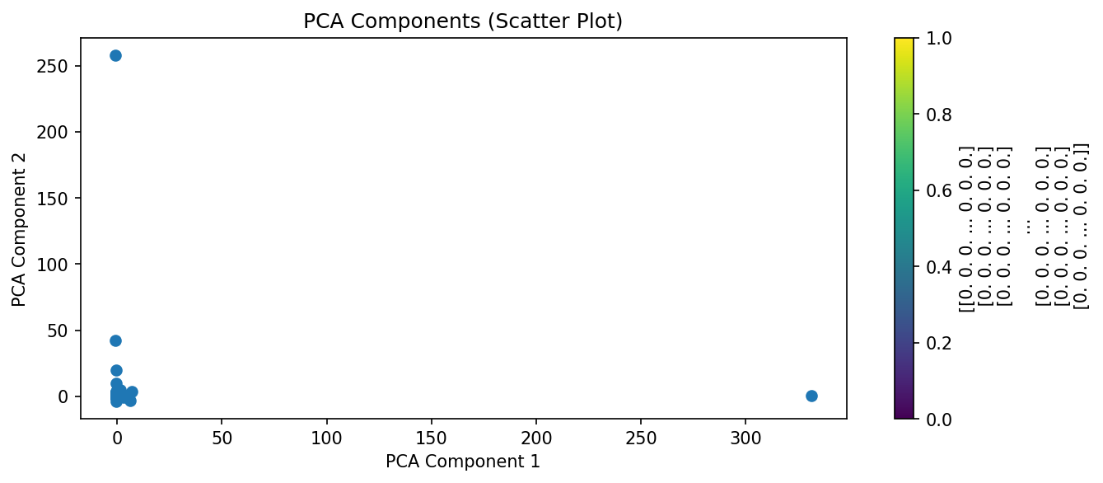
**Bag of words**

* First, I created the function remove\_stopwords() to remove all the stop words from the message contact. I also added many stop words that I found in the text later with the word cloud (cell 5.a).
* Then I created a bag of words and removed from him the stop words. The bag of words is a list of lists with tokens for each message contact, I added the bag of words to be a column at the data frame (cell 5.b).
* I initialized a count vectorizer from the bag of words and normalized it (cell 6.a).

1. **EDA**

**PCA**

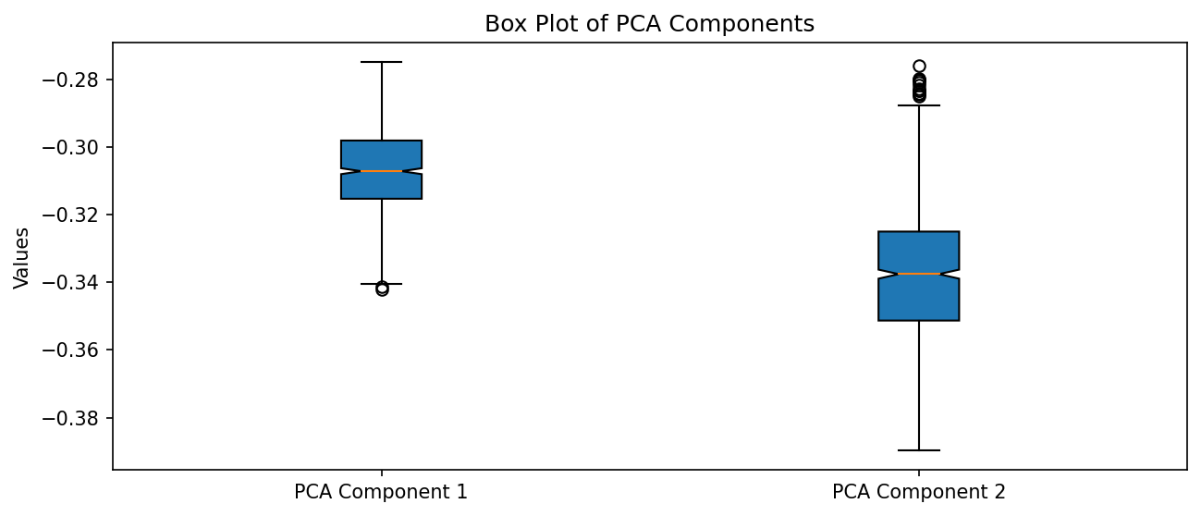
* The PCA model is initialized with two components, the bag of words after vectorization and normalization and the columns:
  + normalized month
  + normalized day
  + normalized hour

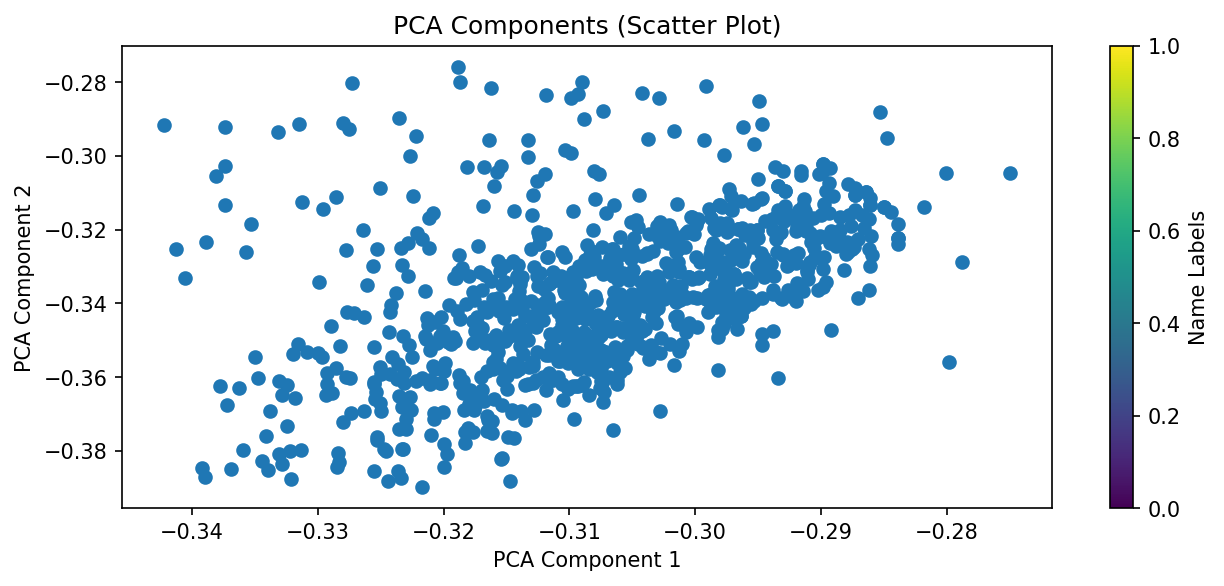
 The PCA model was developed to visually represent the data and identify any outliers (cell 7.a).

* I found that there were more outliers, so I removed them with the IQR method (cell 7.b).

תמונה שמכילה טקסט, צילום מסך, תרשים, קו

התיאור נוצר באופן אוטומטיBefore removing the outliers:

After removing the outliers:

Visualize again the PCA after removing the outliers:

**Word cloud**

****I also visualized the bag of words with word cloud (cell 7.e).

Most common words at the bag of words are:

('העורף', 65)

('שעון', 62)

('שמיעה', 57)

('פיקוד', 48)

('אנשים', 45)

('רטט', 39)

('הרווחה', 36)

('לך', 34)

('אזעקה', 32)

('השעון', 32)

('חכם', 31)

('תקשורת', 30)

('שעונים', 27)

('החזר', 26)

('עידו', 25)

('התראה', 22)

('אפליקציה', 22)

('בזמן', 21)

('עובד', 21)

('האפליקציה', 21)

1. **Unsupervised learning**

**Vectorize**

I initialized again a count vector for the tokenized\_messages columns after I removed the outliers that I founded in the PCA model.

This count vector will be utilized in unsupervised learning models (cell 8.a).

**Correlation matrix**

Before I started with the models, I first made a correlation matrix to remove columns with high correlation.

At the next matrix, I found that there's a high correlation between the columns normalized\_month and normalized\_day (-0.983).

As a result, I removed the normalized\_month column (cell 8.b).

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **normalized\_month** | **normalized\_day** | **normalized\_hour** | **pca\_1** | **pca\_2** |
| normalized\_month | **1.000** | **-0.983** | **0.119** | **-0.971** | **-0.915** |
| normalized\_day | **-0.983** | **1.000** | **-0.118** | **0.972** | **0.915** |
| normalized\_hour | **0.119** | **-0.118** | **1.000** | **-0.250** | **-0.458** |
| pca\_1 | **-0.971** | **0.972** | **-0.250** | **1.000** | **0.975** |
| pca\_2 | **-0.915** | **0.915** | **-0.458** | **0.975** | **1.000** |

**Kmeans++ model**

* **columns**:
  + normalized\_day
  + normalized\_hour
  + normalized\_count\_matrix(the bag of words after vectorization and normalization)
* **Hyper** **parameters**:
  + Init = 'k-means++'
  + n\_clusters = range between 2 to 10.

The best silhouette score was achieved with eight clusters, and the model values were saved in the Kmeans\_dict for future use (cell 8.d).

Cluster Counts:

Cluster Number of Messages

9 113

6 54

4 43

5 38

2 133

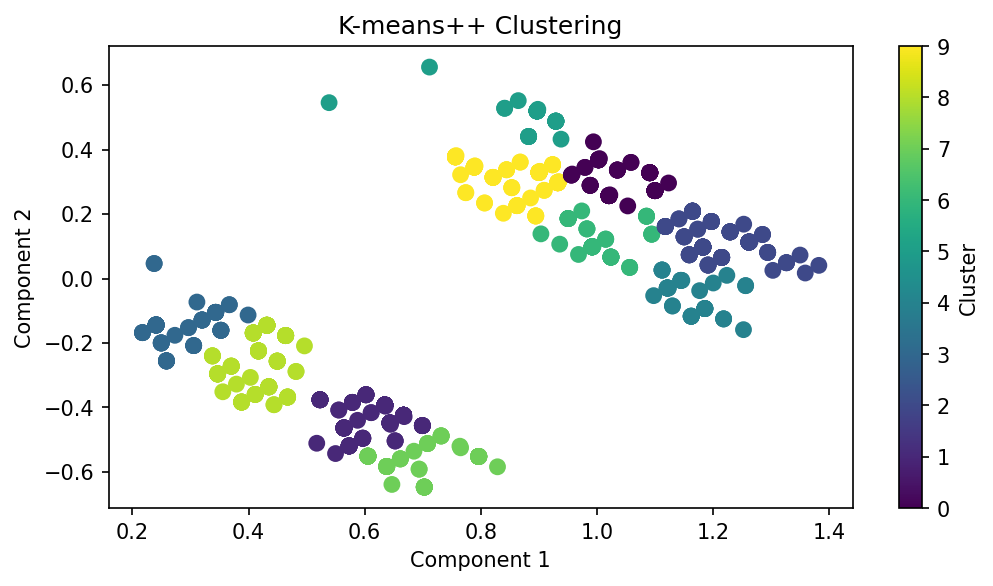
0 94

3 69

8 127

1 213

7 69

Silhouette Score: -0.8324292433387989

explanation about silhouette score:

* The Silhouette Coefficient is calculated using the mean intracluster distance ( a ) and the mean nearest-cluster distance ( b ) for each sample.
* silhouette coefficient = (separation — cohesion) / max(separation, cohesion)
* **Note:** After we get the highest silhouette score, the more clusters that we add will cause the model to enter "overfitting" since the model will provide clusters for each sample or very small groups of samples.

**GaussianMixture model**

**columns**:

* + normalized\_day
  + normalized\_hour
  + normalized\_count\_matrix(the bag of words after vectorization and normalization)

**Hyper** **parameters**:

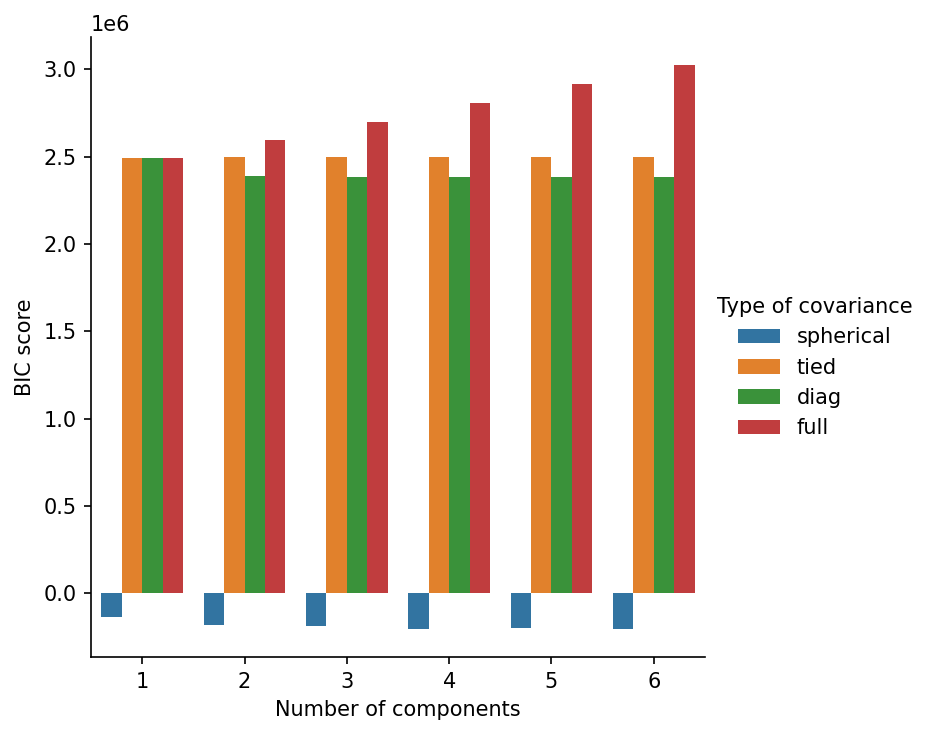
* + Init = 'k-means++'
  + n\_components = range between 2 to 10
  + covariance\_type=["spherical", "tied", "diag", "full"]
* The function gmm\_bic\_score(estimator, X) returns the BIC score of the GaussianMixture model. I saved each model that I tested with its hyperparameters and its score in the grid\_search dictionary (cell 8.f).
* Next step was to Create a table from the results to find the best hyperparameter. The first row in the table represents the best hyperparameter according to the BIC score (cell 8.g).

I found that covariance\_type="spherical" and n\_components=4 are the best hyperparameters(cell 8.h).

|  |  |  |  |
| --- | --- | --- | --- |
| Index | Number of components | Type of covariance | BIC score |
| 3 | 4 | spherical | -203965.707 |
| 5 | 6 | spherical | -203700.550 |
| 4 | 5 | spherical | -197419.492 |
| 2 | 3 | spherical | -186728.482 |
| 1 | 2 | spherical | -179035.028 |
| 0 | 1 | spherical | -133506.335 |
| 14 | 3 | diag | 2383373.992 |
| 15 | 4 | diag | 2383838.475 |
| 16 | 5 | diag | 2384678.941 |
| 17 | 6 | diag | 2385960.244 |

Explanation about the BIC score:

* + BIC = -2 \* loglikelihood + d \* log(N), where N is the sample size of the training set and d is the total number of parameters.
  + The lower BIC score signals a better model.
  + (cell 8.h).

Visualization the hyperparameter BIC scores:

* Next, I initialized the GaussianMixture modelwith the hyperparameters that I founded (cell 8.i):

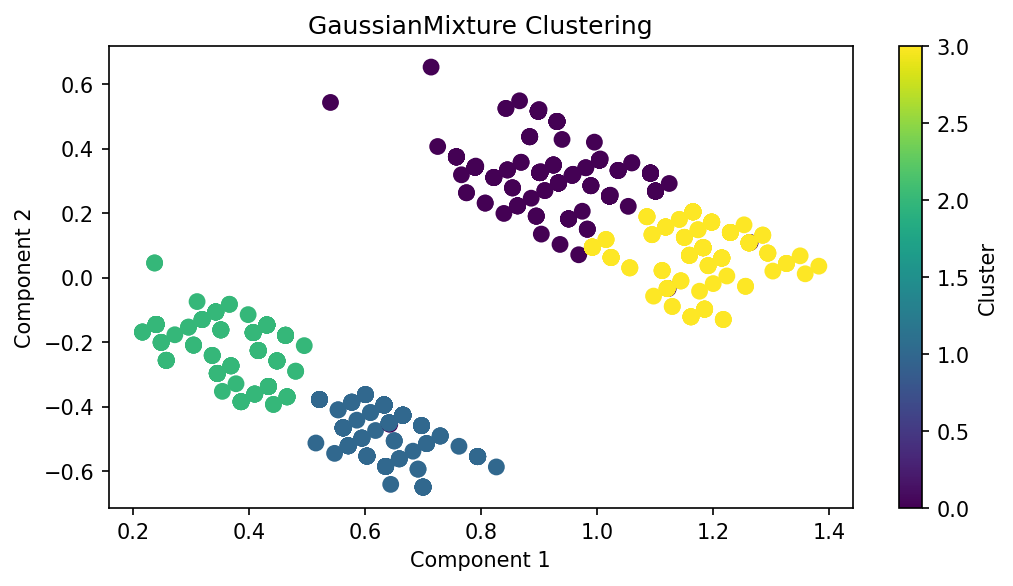
GaussianMixture(covariance\_type='spherical', init\_params='k-means++',

n\_components=4)

**GMM covariances:[0.00181066 0.00028082 0.00040681 0.00043013]**

**GMM weights:[0.27901528 0.29237288 0.2065678 0.22204405]**

Visualization for the GaussianMixture model (cell 8.j):



**Unsupervised learning results**

Now we will print the results from the two unsupervised learning models.

The unsupervised learning process will involve creating a dictionary with tokens and their clusters, then printing the clusters and the number of tokens in each cluster (cell 9.a).

We can conclude that the main subjects that repeat themselves are:

* 1. שעון/ שעון חכם
  2. משרד הרווחה/ משרד העבודה
  3. פיקוד העורף
  4. רטט
  5. אזעקה
  6. חירשים
  7. שיאומי
  8. אפליקציה

The data frame will be expanded with two additional columns: one for analyzing the message's sentiment and the other for indicating if the message is part of the mentioned subject, to determine if there is any correlation between the two columns (cell 9.b).

**5. Sentiment analysis**

The analyze\_sentiment function was initially created to analyze sentiments in Hebrew messages, but the Blobtext library did not fit well. English messages were translated and added to the function. A new column with sentiments for each message was added, labeled (cell 10. a):

* Positive
* Negative
* Natural.

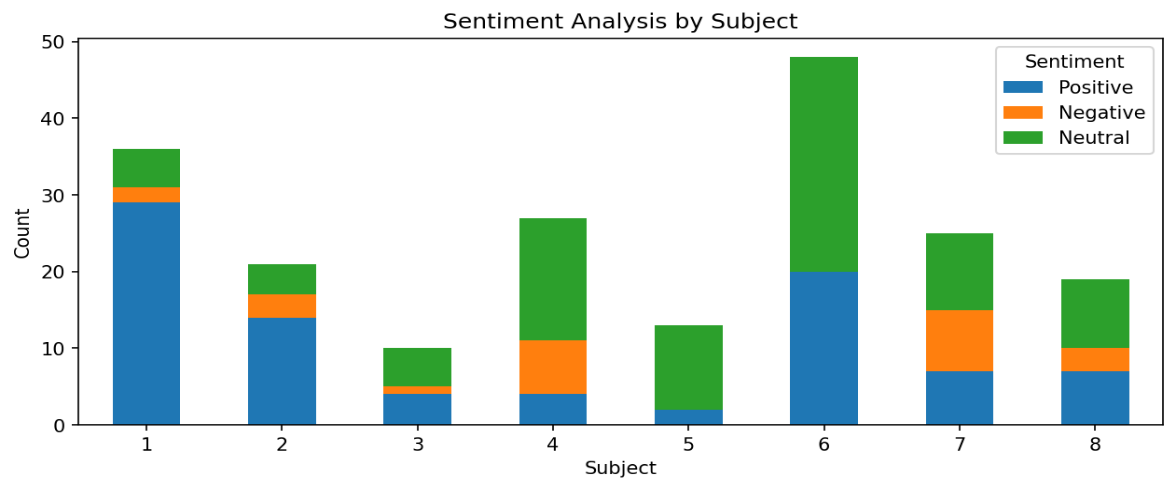
1. **Conclusions**

The next step is to compare the features we discovered and draw conclusions from them.

* At first, I'll compare the main subjects and the sentiment analysis for each message to find out how the people in the group feel and react to each subject (cell 11.a)

|  |  |  |  |
| --- | --- | --- | --- |
| **subject** | **Positive** | **Negative** | **Neutral** |
| **1** | **21** | **1** | **4** |
| **2** | **6** | **3** | **2** |
| **3** | **3** | **1** | **6** |
| **4** | **0** | **0** | **4** |
| **5** | **0** | **0** | **4** |
| **6** | **15** | **0** | **26** |
| **7** | **4** | **8** | **10** |
| **8** | **6** | **1** | **9** |

Visualization of the relationship between the sentiment analysis and the main subjects.



**main subject map**:

1. Smart watches

2. Vibrations

3. Shiomi

4. government office

5. Alert

6. HFC

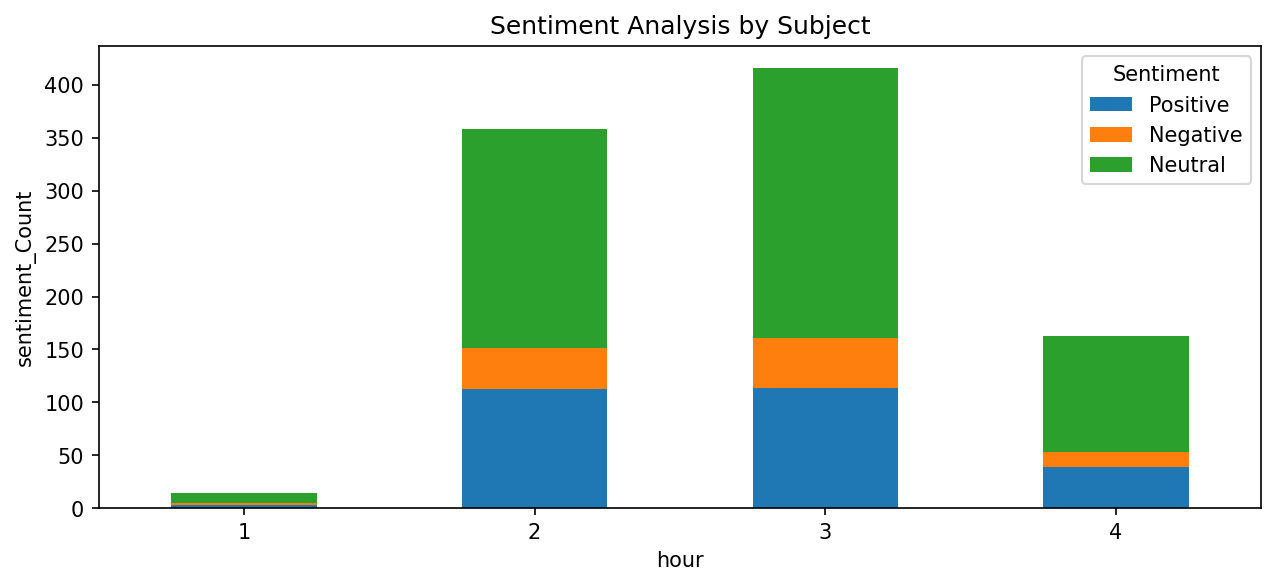
7. deafness

8. Application

From the visualization, we can conclude that the people in the chat group respond positively to the main subjects: smart watches, Vibrations, and the Home Front Command (HFC), and respond negatively to the main subjects: deafness and the government office.

* The next comparison will focus on the hour feature and sentiment analysis to determine if the chat group responds more positively to messages during different hours or negatively (cell 11.c).

|  |  |  |  |
| --- | --- | --- | --- |
| **hour** | **Positive** | **Negative** | **Neutral** |
| **1** | **4** | **0** | **11** |
| **2** | **107** | **33** | **236** |
| **3** | **84** | **37** | **211** |
| **4** | **48** | **18** | **100** |



The visualization indicates no significant difference in the time of day or response rates of the chat group.

* The next comparison will be made between main subjects and day features to determine if the subject was discussed in the long or short term and at which stage (cell 8.d).

תמונה שמכילה תרשים, קו, עלילה

התיאור נוצר באופן אוטומטי**main subject map**:

1. Smart watches

2. Vibrations

3. Shiomi

4. Government office

5. Alert

6. HFC

7. deafness

8. Application

conclusions from the visualization:

* The alerts and the application subjects were discussed for a long-term at the first stage of the war.
* The deafness subject was discussed the most, but for a very short time.
* The smart watch subject was discussed during all the chat group time.
* At the middle stage, the HFC subject was discussed the most.
* At the last stage, the Shiomi subject was discussed not in a wide range but in the long term.
* At the last stage, the office government was discussed the most but for a short-term.