**Information Retrieval -**

**Course's Final Project**

**Hadas Racah 204840532**

**Shahar Golan 209515162**

**Index**

* Project Summary
* Language Model
* Text Classification
* Text Clustering
* Project Conclusion

Project Summary

IR focuses on retrieving relevant information from large text collections based on user queries. It combines many techniques such as tokenization, stop word removal, stemming, and vector space models that are commonly employed to process and analyze text data. These techniques enable efficient indexing and retrieval of documents matching user queries, forming the backbone of search engines and information retrieval systems.

IR not only involves retrieving relevant information but also encompasses tasks like text classification and clustering. Classification involves categorizing documents into predefined classes or categories, while clustering involves grouping similar documents together without predefined categories. These tasks often involve similar techniques to those used in traditional IR, such as tokenization, stop word removal, stemming, and vector space models.

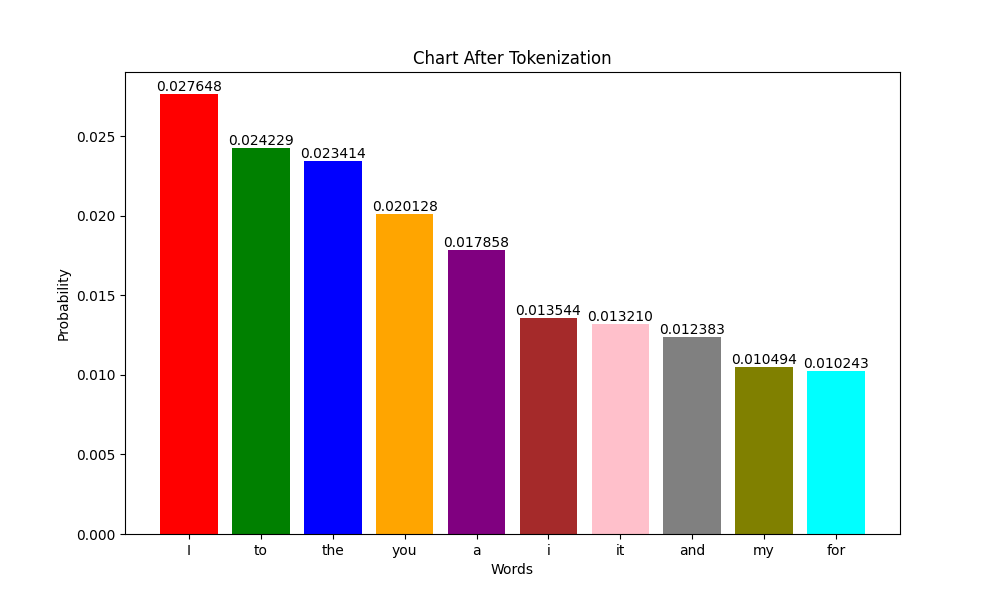
This project aims to get hands-on the common techniques of IR, including language modelling, text classification and text clustering.

This project was based on 4 collections of documents: Search engine fairness, Discrimination discovery, Political bias and Filter bubble.  
Each of this collections was taking a part in one or more tasks.

משימה 1:

במשימה זו נציג 5 ניתוחים שונים על קובץ הנתונים שקיבלנו. נעבור שלב שלב מקבלת הקובץ בצורה גולמית עד לצורה הסופית ממנה ניתן לסווג את הדאטה בצורה הטובה ביותר. ההסברים יהיו בשפה האנגלית מכיוון שהתרגום לעברית קשה מאוד עבור חלק מהמושגים.

**Tokenization**: This step forms the baseline analysis, showcasing the raw frequency of words as they appear in the collection before preforming any text operations and just splitting to tokens.



**Stop words**: This step forms the removal of stop words. It helps to focus on the more meaningful words in the collection.

תמונה שמכילה טקסט, צילום מסך, צבעוני, תרשים

התיאור נוצר באופן אוטומטי

**Case folding**: This step aids to perform normalizing the text data, ensuring that the same words in different cases are treated uniformly.

תמונה שמכילה טקסט, צילום מסך, צבעוני, עלילה

התיאור נוצר באופן אוטומטי

**Stemming**: This step focusing on the core meaning of words by stripping suffixes. It can sometimes be overly simplistic or inaccurate due to the crude nature of most stemming algorithms.

תמונה שמכילה טקסט, צילום מסך, צבעוני, עלילה

התיאור נוצר באופן אוטומטי

**Lemmatization**: This step provides a more refined and accurate approach to reducing words to their base forms.

תמונה שמכילה טקסט, צילום מסך, צבעוני, עלילה

התיאור נוצר באופן אוטומטי

**Conclusions:** There are many interesting things that came up from this part, we chose to talk about 4 of them:

1. The tokens ‘**fairness**’ and ‘**fair**’ are shown in the graphs 2 & 3, shows the importance of this word to the collection.
2. The term ‘**Google**’ appeared in top@6 in the second graph, but after case folding it went to top@8. We assume that it’s happened because the term appears in the texts mainly with case character, in contrast to other words such ‘**fair**’ or ‘**bias**’ that probably appear in both ways.
3. It was magnificent to see the appearance of ‘**use**’ and ‘**user**’ in graph 4, and how the algorithm doesn’t omit one of the terms and recognized the differences between them.
4. The term ‘**data**’ position decrease from graph 3 to graph 4.  
   We realized that it was because ‘**data’** is the only form in which the word appears.

building the language model was truly an interesting and teaching part of the assignment.

Text Classification

In this task, we used the first and the second collections of documents (“Search engine fairness” and “Discrimination discovery”). We Labeled the first collection of documents to be Yes, and the second collection to be No.

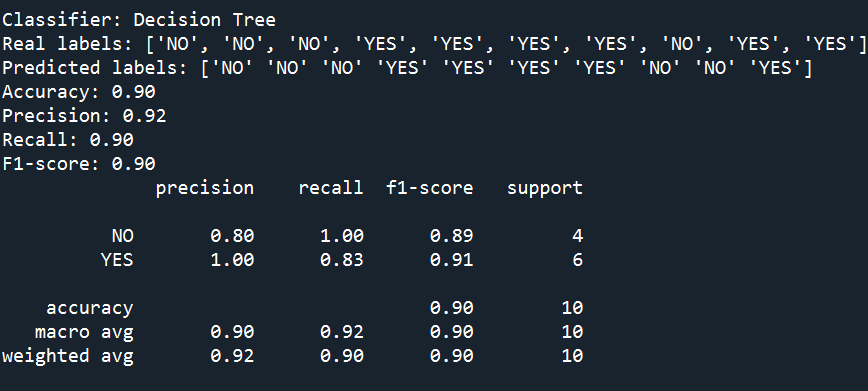
We splited the documents to train set and test set, then we trained 4 different models using cross validation to get the predicted labels of the test set.

By making this part we also could calculate the scores of the predicted label, by using the formulas we learned in class.

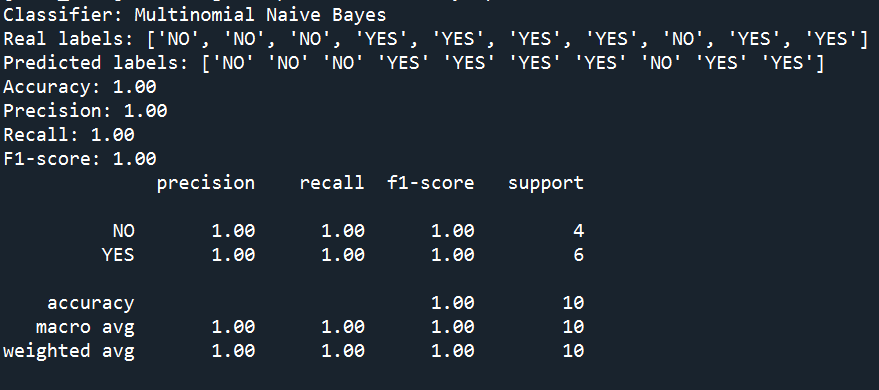
We chose the following 4 classifiers because of their accuracy in text classification:

* Decision Tree
* Naïve base
* Random forest
* SVM

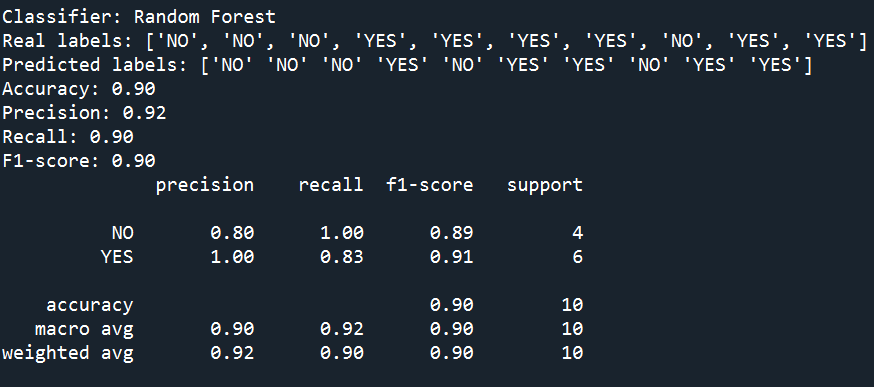
The following images shows each of the classifiers we chose, its target label comparing to the predicted label, and the scores it got.

**Decision Tree:** 

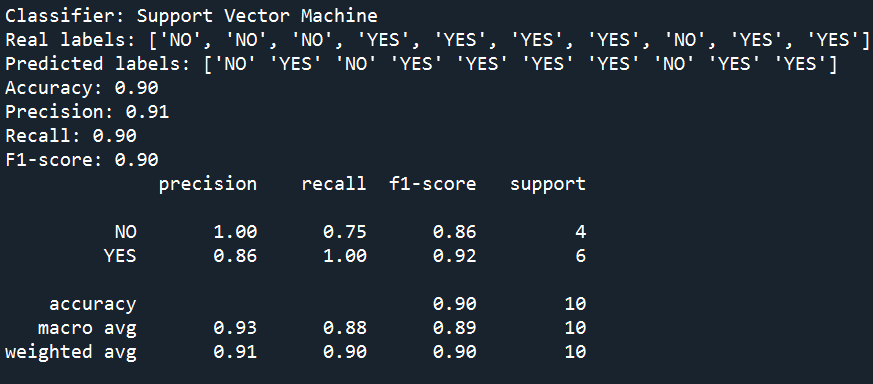
We can see that the accuracy of this classifier is 90%, and that No got perfect recall when Yes got perfect precision.

**Naïve Base**:

We can see that the prediction using Naive Base was 100% successful. In the real world, this probably won't happen.

**Random Forest:**

We can see that the scores of this classifier was also high and like decision tree classifier.

**SVM:**

We can see that the scores of this classifier similar to the other.

**Conclusion:** The 4 chosen classifier got pretty much similar results. We can agree that the results are good and can even improve by using other text operation to clean the text, or use another techniques to improve the results.

We can also see that the classifier of Naïve Base got perfect prediction, that caused the results to be 1 for all of them.   
It can happen for many reasons, such as:

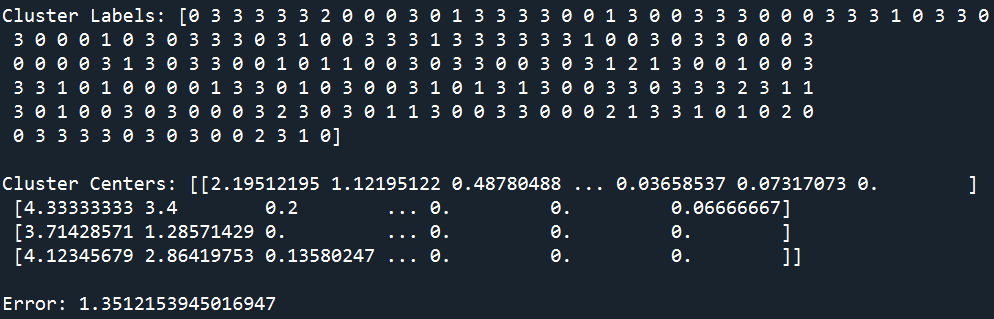
1. Small dataset – when the dataset is not that big it means that the test set is not that big either, and it’s easy for the classifier to predict the labels. Maybe changing the split to bigger percentage can show more real results.
2. Simple problem – it's reasonable to assume that the text was not that complicated, and was arranged well, with just text inside, so the classifiers had simple job to do.

Text Clustering

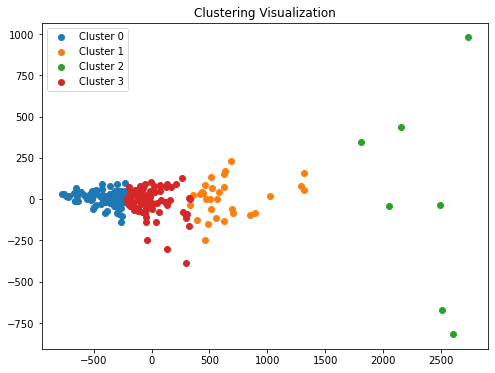
In this task, we used all 4 collections and preformed clustering on them.

We used K-Means classifier with 4 clusters, that shows in the following image as labels 0,1,2,3.

We also printed the centers of the clusters that each of the samples was labeled to.

The last thing that can be seen in the image is the error of the clustering. We used the error calculation based in Davies-Bouldin index that measures the average 'similarity' between each cluster and its most similar one. The similarity is the ratio of within-cluster distances to between-cluster distances. Lower values indicate better clustering.

The following graph shows the samples and their chosen label.



**Conclusions:** In this task, the documents were clustered into four distinct groups, each representing a different cluster. The distribution of documents across the clusters varied, indicating diversity in the dataset.

It's very interesting to see the centers gets their final centroids and the samples that finds their way to the closest cluster.   
When running our code, we also tried to run with 3 clusters instead of 4, just to see the difference of the clustering. It was interesting to see that the samples at the edges was still belonged to the clusters at the edges.

Overall, the clustering analysis successfully organized the documents into meaningful groups based on their textual content. Further investigation and domain-specific knowledge may be necessary to fully interpret the clusters and extract valuable insights from the document collection.

Project Conclusion

This project meant to let us get hands-on some basics of IR process. It included creating language model and preform text operation on the texts, text classification to predict the labels for new documents and text clustering to cluster the documents to 4 different clusters.

This project required us to understand the text in order to understand what problem we are dealing with, required research to understand which models should be run and required us to understand the differences between process to process in order to see the change at each stage.

It was very instructive to see the differences between the stages and to discuss among ourselves the changes and the conclusions that each one got from each stage.