BigData

Yusuf Ali Hilooglu

July 2025

Disclaimer

This project is just to test my Apache Spark skills, it doesn't represent the current state of the world for several reasons:

1. The crawl that I'm working on is from 2021.
2. My choices for target substrings and filtering are primitive and not at all modern, should not be trusted for an analysis in a sensitive topic like this.
3. The method of sentiment analysis is really simple. It does not guarantee a definitive result.

Therefore, I want to stress that this project does not aim to target any individual or any country. It just seemed like an interesting topic and I chose it so that I could be more motivated.

Note: ChatGPT was used to get help with SQL queries and for help in order to make the program more efficient.

Idea

In this project, I aimed to perform an sentiment analysis on sentences on the webcrawl WET files that include any of the target substrings regarding Palestine-Israel. First, a lightweight filter is used on the full data to reduce the workload. Then, a more complex filter is used to filter out sentences that don't include any of the target substring. Then, an NLP model is run to analyze each of the sentences. In the end, sentimentally negative and positive sentences are counted for each target substring.

NLP Analysis - External Library

Since working with Scala Spark in a cluster introduces dependency issues by itself, I wanted to find an NLP library that is native to Scala, in order to keep the dependency requirements simple. With this preference in mind, I found the NLP Library of John Snow Labs. This library allowed me to create an object of a pre-trained model which I can provide a sentence as an input and get the sentiment as a output.

Methodology

The final program has 4 main sections that contribute to the end result: Pre-Filtering, Filtering, Sentiment Analysis and Collecting the Results.

Pre-Filtering

In this part, the main objective is to quickly filter out the data that we're certain that we won't use. That is why, I chose to filter out non-Latin sentences. In order to do that, I created a Latin Regex and then tried to fit the first 20 sentences to the regex. This way, a fast filtering is done in O(N) where N is the number of WET files.

Filtering

As a big part of the WET files is already filtered out in the previous part, the data is now small enough to perform a slightly more complex filtering. The goal of this part is to filter out any sentence that do not include at least one target substring. This is done with SQL and the exact code can be found in the provided scala file.

Target substrings are chosen in a way that they represent Palestine or Israel.

Sentiment Analysis

Since the only the relevant data is left, the pre-trained model of John Snow Libraries is used in each sentence.

Collecting the Results

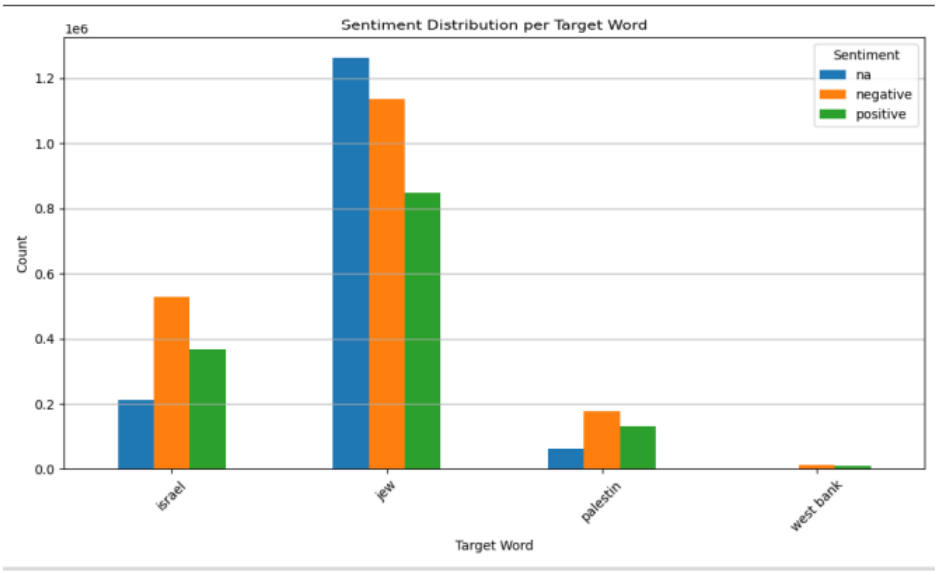
With using pure SQL code, positive and negative occurrences of each target substring is counted and a small table is created where the occurrence count is shown for each sentiment, for each target substring.

Execution of the Code

As the code performed well with smaller data, I decided to run the full code in one segment. Specifically, "1618039594808.94".

Results

After running the code in a full segment, I got the end result of the research. It can be seen that all target substrings occur in more negative sentences than positive sentences. In other words, no matter how frequent the target substring is or no matter if the substring is Palestine-related or Israel-related, it has more negative occurrences than positive occurences. The figure below shows the end result plotted of the research.



Discussion and Limitations

Since this is a sensitive topic, more complex analysis should be done in order to conclude a definitive result. As stated in the Disclaimer section, the analysis that I did is basic and cannot conclude an end result.

Future Work

Based on the efficient filtering that I did, a better sentiment analysis can be done in order to gather better results. Also, using a more recent webcrawl could significantly improve the quality of the end result.