Dataset used:

YouTube channel "الدحيح", all the episodes in the folder, in addition to the annotations included.

Task chosen to preprocess the data for:

Topic classification, based on the episode's title and script.

How the data is read:

Made a dictionary containing 3 lists: one for the filename, the content (script of each episode), and the annotations of each episode.

Cleaning techniques done:

1- Dropped the annotation file named "اميتاب بتشان" because it didn't contain a script, only the annotations for the episode.

2- Cleaned the timestamp beside each line that indicated in which minute in the video this statement was said.

3- Removed [موسيقي] token that was at the beginning of each episode.

4- Normalized the text to standardize variations in letter forms, such as unifying "ه" and "ة" or "ي" and "ى".

5- Removed the diacritics to prevent the words from falling outside the vocabulary when training our machine/deep learning model.

6- Removed the punctuation because they do not add meaningful information for classification.

7- Removed the empty lines.

8- A customized set of stop words was removed. Didn't use the one in NLTK library because it was aggressive, removing words that could change the context, like "ليس". Although this wouldn't matter in the task chosen to be done. It would matter if I was planning to do sentiment analysis, for example.

9- Joined text to be concatenated into a single line instead of being broken by newline characters.

10 - Removed unnecessary spaces in each filename.

11- Removed "الدحيح" at the end of each filename.

Data analysis and insights gained:

1- It was observed that the top 5 most liked episodes are the ones narrating a story, while the episodes with the highest comments are the ones discussing a controversial topic, like "الماسونيه" for example.

2- The most occurring words were "عزيزي" and "ابو حميد" which could be guessed easily since they're considered a signature for el daheeh.

3- The dataset's categories primarily acted as general keywords rather than structured classification labels, so they were unimportant.

4- TF-IDF vs word count: TF-IDF was done to extract numerical representations of episode scripts, enabling analysis of word importance across documents. Also, absolute word counts were computed, and it was observed that words with a high count in a document tend to have a low weight in that same document.

5- The correlation between the likes, views, and comments was high, which is expected.