**What is Sanitization?**

**Jo 250 words ka article araha usko 30-50 words mei transform karna**

Data Compression for e.g. 250 words to 30 to 50 words

**What is Filtration of data?**

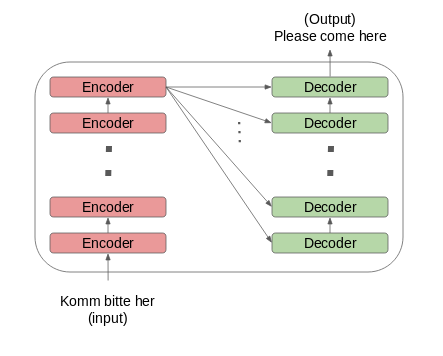
* + Convert Text to lower case
  + Remove Special Characters/Unwanted Characters
  + Correction of any typos 🡪 (TextBlob/PyspellChecker/autocorrect Library)
  + Normalization (such as running, ran into run)
    - Stemming
    - Lemmatization

Stemming uses the stem of the word while lemmatization uses the context in which the word is being used.

* + Tagging (Parts of Speech such as noun, pronoun, verb etc.)
  + Chunking/Chinking: chunking breaks those word. While chinking further break those words (Chinking is not always necessary)

**What is Transformer?**

Works as an encoder and decoder



**NLP working:**

1. Perform Cleansing of Data
   1. Convert Text to lower case
   2. Remove Special Characters/Unwanted Characters
   3. Correction of any typos 🡪 Normalization
      1. Stemming
      2. Lemmatization
   4. Chunking/Chinking
2. The cleanse data would be sent to machine learning/deep learning model -----🡪 (Sending data to Transformer)
3. Tune transformer according to our requirement
4. Consist of 2 phases training and testing. In the training phase we will train the model to generate a summary from a given news article using a dataset of 7000-8000 articles. The testing phase would then consist of our own system producing summarized weather news reports.

**Why are you using Libraries?**

We are using libraries to generate our own model which can perform those tasks required for our system.

1. Typos ---🡪 (TextBlob/PyspellChecker/autocorrect Library)
2. Machine Learning/ Deep Learning ------ > Transformer -----🡪 GPT3, BERT libraries
3. TTS ---🡪 GTTS
4. Web scrapping -----🡪 Urllib, BS4, Selenium
5. Filtration of data------🡪 NLTK, NLG, SpaCy

**Comparison chart (Text to Speech Algorithm)**

3 speech recognition systems were compared which included the Google API, Microsoft API and sphinx 4. The experimental details are as follows:

The audio files were selected from various sources to evaluate the Microsoft API, Google API, and Sphinx-4. According to CMUSphin, Sphinx-4's decoder supports only one of the two specific audio formats (16000 Hz / 8000 Hz). WAV files not supported by all three so they design tool which recognition all audio files in the same format (16000 Hz / 8000 Hz).

**Results:**

According to the results the Sphinx-4 library gave a 37% WER), Google Speech API gave 9% WER and Microsoft Speech API gave 18% WER.

WER here represents Word Error Rate which tells how many words are not transcribed correctly with the reference words

Thus Google API provided a better result compared to the other 2.