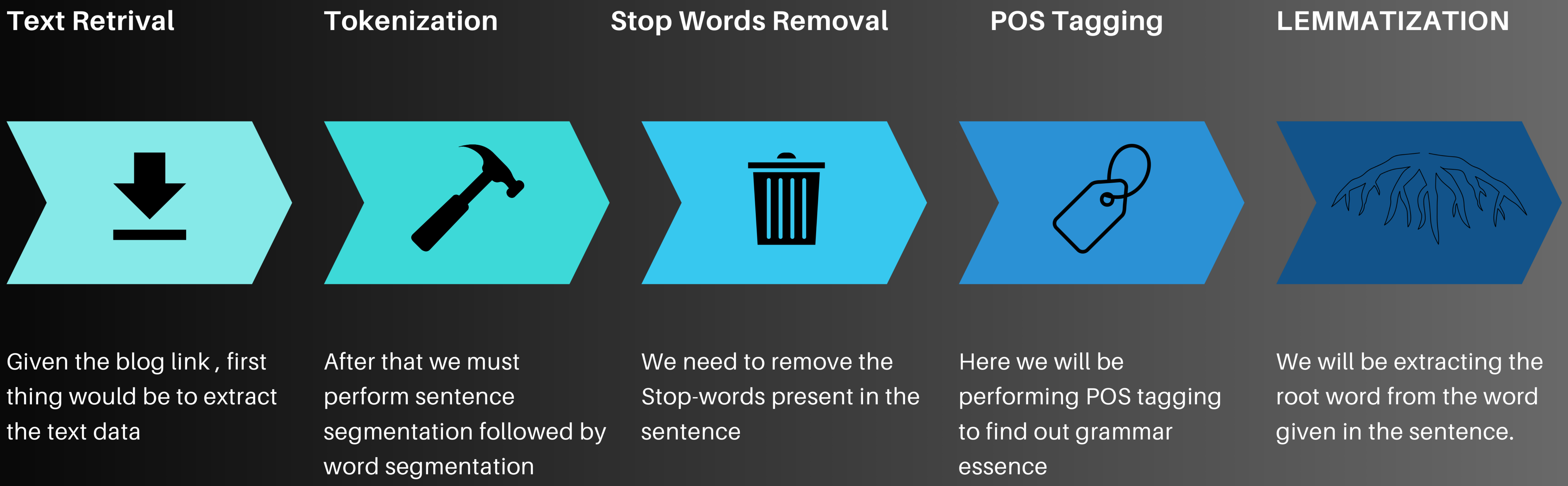
TEXT SUMMARIZATION

Online Blog

Pre-Processing Steps



Text Retrival

Given the blog link , first thing would be to extract the text data

Tokenization

After that we must perform sentence segmentation followed by word segmentation

Stop Words Removal

We need to remove the Stop-words present in the sentence

POS Tagging

Here we will be performing POS tagging to find out grammar essence

LEMMATIZATION

We will be extracting the root word from the word given in the sentence.

Main Model

Similarity Evaluation

After doing the pre-processing steps we find the similarity matrix using BM-25 and original Text Rank Similarity.

Graph Formation

Now we consider each sentence as nodes and then use the similarity matrix to find weights of the edges

Text Rank Algorithm

Now we perform Text Rank Algorithm which will trace out the probability of that node to be able to summerize

Extracting Summary

Now based on 20% rule we find the nodes with highest Probability

Similarity Evaluation

-> Fast-Text
-> Text Similarity

We after doing an ablation study over all the vector representation.
we came to a conclusion that Fast-Text gave us better representation of the meaning.
we also used Text Similarity metric which will give us lexical similarity.

Text - Similarity

$$\mathrm{Sim}\,(\mathrm{S_i},\ \mathrm{S_j}) = \frac{|\{\mathrm{w_k}|\ \mathrm{w_k} \in \mathrm{S_i}\ \&\ \mathrm{w_k} \in \mathrm{S_j}\ \}|}{\log(|\mathrm{S_i}\ |) + \log(|\mathrm{S_j}\ |)}$$

Text Rank Algorithm

we constructed a graph wherein each node is a sentence from the article and each edge has a weight which is equal to the similarity of the two nodes which is given by the similarity metric. Using this Graph structure we perform the Text Rank Algorithm and found the top 20% of the sentences which can summarize the article.

$$PR(p_i; t+1) = \frac{1-d}{N} + d \sum_{p_j \in M(p_i)} \frac{PR(p_j; t)}{L(p_j)}$$