STEP 1: Preprocessing

* Lowercasing:

All the letters in the sentence are lowercased to ensure uniformity and thereby easier processing. The sentence “The books are on the table” is converted to “the books are on the table”.

* Tokenization:

It refers to splitting an entire sentence into individual words. If the sentence is “The books are on the table” then the sentence is split as ‘the’, ‘books’, ‘are’, ‘on’, ‘the’, ‘table’.

* Special character removal:

Special characters like ‘!’, ‘@’, ‘#’, ‘$’ if present are removed

* Stop word removal:

Stop words are frequently occurring words which include words like ‘is’, ‘the’, ‘are’, ‘of’, ‘in’. These are considered redundant and are hence removed.

* Stemming:

It refers to converting all words to their base or root form. For example, ‘learn’ is a base word for its variants such as ‘learn’, ‘learns’, ‘learning’, and ‘learned’.

* Lemmatization:

It is an advanced form of stemming. As in the case of two sentences “She is right” and “Please turn off the lights”, the lemmatizer will correctly differentiate between the usage of the word ‘right’

STEP 2: Feature extraction

We cannot directly feed our text into an algorithm. We need to process raw data into numerical features that can be processed while preserving information in the original dataset.

* Bag of words:

It is used to preprocess text by keeping a count of frequency of each word in a sentence. Output is in the form of 1 and 0

* TF-TDF:

Term frequency is the total number of times a given word appears per total number of words in the given document. Inverse document frequency measures how much information a word provides. It converts raw data or dataset into vectors and each word has its own vector.

* Word2Vec:

A word is converted to a vector that has some properties which defines the word uniquely. King-man+woman=queen, can be predicted using this.

STEP 3: Representation

* Sequence-to-sequence architecture:

It is a model which is trained using a sequence of input-output pairs. .It uses neural networks. Where input and output are a sequence of tokens. Contains 2 main components: encoder and decoder.

* **Attention mechanism:** The input to the decoder is a single vector that has to store all the information about the context. This becomes a problem with large sequences. Hence the attention mechanism is applied which allows the decoder to look at the input sequence selectively.

STEP 4: Post processing

* Detokenization:

Translated tokens are combined to form coherent sentences.

* Clean-up:

Removes any inconsistencies that might have popped up during translation.