**Abstractive summarization:**

Language Model mainly consists of Encoder and Decoder Modules

Encoder accepts input of predefined length and returns the hidden state and cell state

The hiddden state and cell state carries the whole context fed by input . It also forgets some unimportant context present in input which is decided by Forget Gate.

Depending on this required context , output is Encoder is passed to Decoder whose job is to decode the information. Its trained to predict the next word of sequence by considering present word.

The input to decoder is passed along with two special tokens like <START> and <END> which helps to identify the start and end of sequence.

Steps:

* Dataset is preprocessed by lowering text, removing stop words, special characters and and non printable characters as per the requirement.
* Whole text is checked for maximum length of input and maximum length of output such that lengths can be fixed . Other rows having length less than this will be equated by pad\_sequences.
* Whole data is split into tokens using tokenizer and converted to integer sequences to procedd further. Same is done with labelled summaries also.
* Whole dataset split into train and test.
* Sequential Model is built with below layers
  + Input layer
  + Embedding layer
  + LSTM layers which returns hidden state and cell state
  + Decoder Layer
  + Input layer
  + LSTM layers which returns hidden state and cell state
  + Attention Layer to facilitate attention and concatenated with LSTM layers output
  + Dense layer with activation function
  + Model is compiled with appropriate loss and optimizer
* Model is fit using training data by giving number of epochs and callbacks (earlystopping) to avoid overfitting and validation dataset.
* Model cna be played around by different values using GridsearchCV etc to get the best model according to the given metric.