

Report on Abstractive summary

Name – Manish Kumar

Student of praxis business school

Contact Info- manish.kumar@praxis.ac.in Mobile No - 7903061900

Abstract:

When I was in my Primary School we got a task like that write summary on given topic then we took the important word and write the summary but our teacher's expectation is something else from me."he wants I not use only those words that are appear in the article but also use those are not in article, on that I am able to do that or not but this time my algorithm able to do this. In this paper I am going to discuss steps to make these types of abstractive summary.

Process

Abstractive summary: The abstractive text summarization algorithms create new phrases and sentences that relay the most useful information from the original text — just like humans do. The abstractive text summarization is used to predict the phrase on given sentence.

Problem Statement: Customer reviews can often be long and descriptive. Analyzing these reviews manually, as you can imagine, is really time-consuming. This is where the brilliance of Natural Language Processing can be applied to generate a summary for long reviews.

Dataset source – Kaggle

Amazon fine food review

Reason behind choosing this dataset review and summary both are given we want to predict new summary from review so we can make comparison between the given summary and generated summary.

Steps follows to reach at accuracy:

1. Load the data
2. Data preprocessing
3. Encoder & decoder
4. Attention layer
5. Build RNN model
6. Check Model accuracy
7. Model for encode & decoder
8. Check final accuracy

Frequently Library Used

Pandas

Numpy

Tensorflow

Keras

NLTK

- 1) Load dataset- this dataset has 578271 observation due to system compatibility we consider only 200000 observation this time.

2) Data preprocessing

In this section mainly we want to clean the data because our data might be in any format so we follow some steps to clean the data these are one by one.

- Used NLTK stopwords to remove
- .Convert everything to lowercase
- 2.Remove HTML tags
- 3.Contraction mapping
- 4.Remove ('s)
- 5.Remove any text inside the parenthesis ()
- 6.Eliminate punctuations and special characters
- 7.Remove stopwords
- 8.Remove short words

Performing basic preprocessing steps is very important before we get to the model building part. Using messy and uncleaned text data is a potentially disastrous move. So in this step, we will drop all the unwanted symbols, characters, etc. from the text that do not affect the objective of our problem.

Encoder-decoder-. **The encoder and decoder blocks are actually multiple identical encoders and decoders stacked on top of each other.** Both the encoder stack and the decoder stack have the same number of units.

The number of encoder and decoder units is a hyperparameter.

Encoder-decoder is nothing but it's only a recurrent neural network.

4) Attention layer: A limitation of the architecture is that it encodes the input sequence to a fixed length internal representation. This imposes limits on the length of input sequences that can be reasonably learned and results in worse performance for very long input sequences.

- The limitation of the encode-decoder architecture and the fixed-length internal representation.
- The attention mechanism to overcome the limitation that allows the network to learn where to pay attention in the input sequence for each item in the output sequence.

After that I build build a RNN model which calculate the predictive summary of given sreview.

