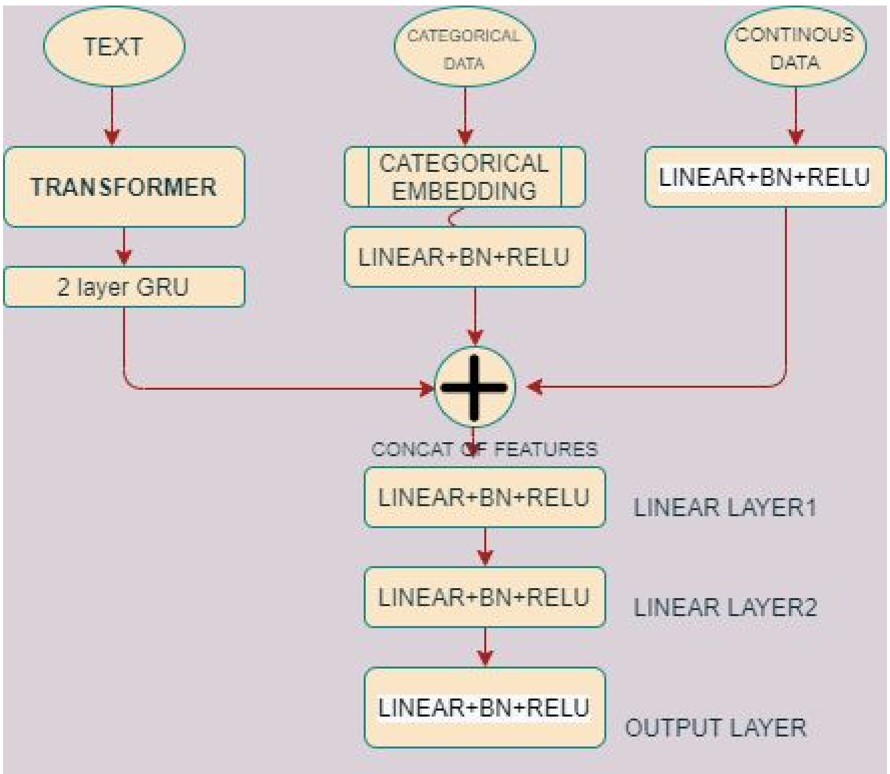
My approach to this problem was to make use of all the relevant data present in the dataset along with the textual content for the classification.

In the website classification dataset contained 3 types of data:
1)Text Data (Website content ,url)
2)Categorical(Website Metadata)
3)Numerical or continuous data(Website Metadata)
Hence to use all the 3 types of the data ,I made my model architecture to handle all these data.

- For textual data i used pretrained transformers as the encoder then followed by Gru layers.
- For categorical data I used an embedding layer to get the proper feature representations from categorical input.
- For continuous data a simple linear layer with BN and Relu was used.

Diagram Showing the structure of the model architecture.




Reasons behind the architecture;

- Pre-Trained transformer Encoder was used to get SOTA representation of the text,for this DistillBert was used along with its pretrained embeddings and tokenizer.
  The reason for using DistillBert inplace of transformer encoders with more parameters like bert to gain computational efficiency.It has 40% less parameters than bert-base-uncased, runs 60% faster while preserving over 95% of Bert's performances as measured on the GLUE language understanding benchmark.[2][3]
  After getting the encoded output from the Transformer Model, a 2 layered GRU was used to  get single sequence output,inplace of any dense or pool layer before classification ,to get a better representational vector that could be passed to the classifier. Here a GRU was used rather LSTM because the GRU has , the performance is on par with LSTM, but computationally more efficient.[4]

- For categorical data-
  Embedding layer was used to generate good representational encodings[1][5]
- After using getting the outputs of a linear layer of continuous all the 3 types of data are concatenated together and then passed through an MLP based classifier to get the final output.

Results-
1.Precision-
 Class 0-0.9883
 Class 1-0.9473
2.Recall-
 Class 0-0.9419
 Class 1- 0.9895

*All results are based on train set.

References-
1.https://www.usfca.edu/data-institute/certificates/fundamentals-deep-learning lesson 2)
2.DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter
arXiv:1910.01108v4
3.https://github.com/huggingface
4.Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling
arXiv:1412.3555v1
5.https://yashuseth.blog/2018/07/22/pytorch-neural-network-for-tabular-data-with-categorical-embeddings/

Raj Aaryaman Patra
rpatra1267@gmail.com
+916370689339