MIDAS IIIT DELHI TASK - 3

Submitted By - Tanmay Jain , 2nd Year (Btech) , DTU

Submitted To - Avinash Anand Yadav Sir(MIDAS)

About me and my task selection -

Thank you **MIDAS** for giving me the opportunity to perform this task of Product Categorization through text description

This task introduced me to the beautiful world of natural language processing

I had knowledge about **Convolutional Neural Networks** and didn't know much about **NLP** , so I took this task to challenge myself and didn't undertook **task 2 which was regarding CNN's as in the case of research internship we would have to explore something new within a given time frame , so I decided to go with the challenge**

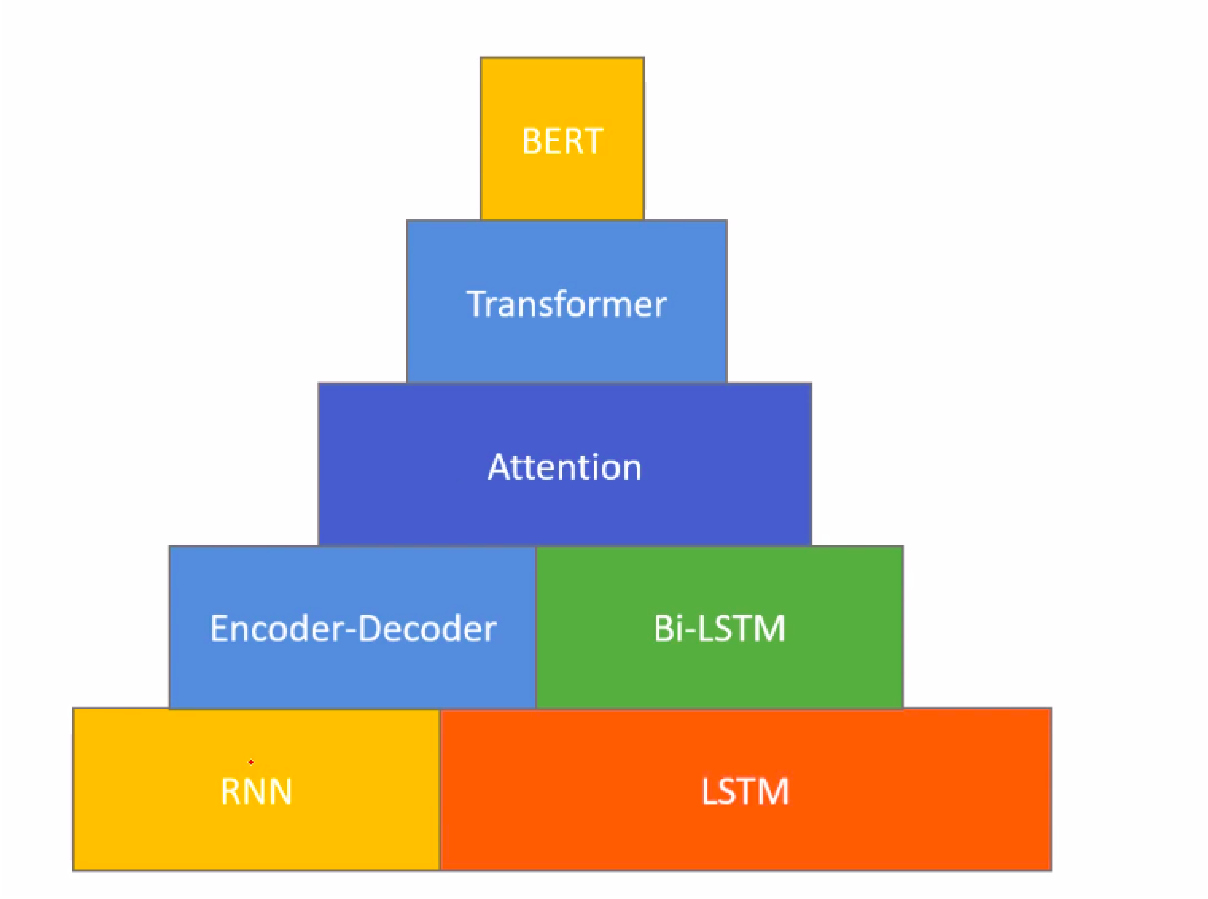
I want to pursue my career in research in field of robotics and deep learning(AI in general)
And want to pursue masters abroad , and there is no better place than MIDAS to start with research for undergraduates

I currently lead a group of 40 people wherein we are developing a **autonomous race car** for **Formula Student Competition** which will be held in **Germany** in year 2022 (I have worked closely with implementations of various path planning and control algorithms in ROS and Lidar data preprocessing which is fed into our SLAM system)

I started with the task on around **5-6 April 2021** due to recommendation by my friend

Being new to field of NLP , I read some blogs about how to get started with NLP

I discovered about a pyramid given below -



I read about RNN and other related architectures , then went to read about the attention and then to transformers and then to Bert

I have also read about the maths behind the transformers architecture and behind the bert architecture , and I enjoyed reading about it

Below I present the detailed report for my task

TASK 3 -

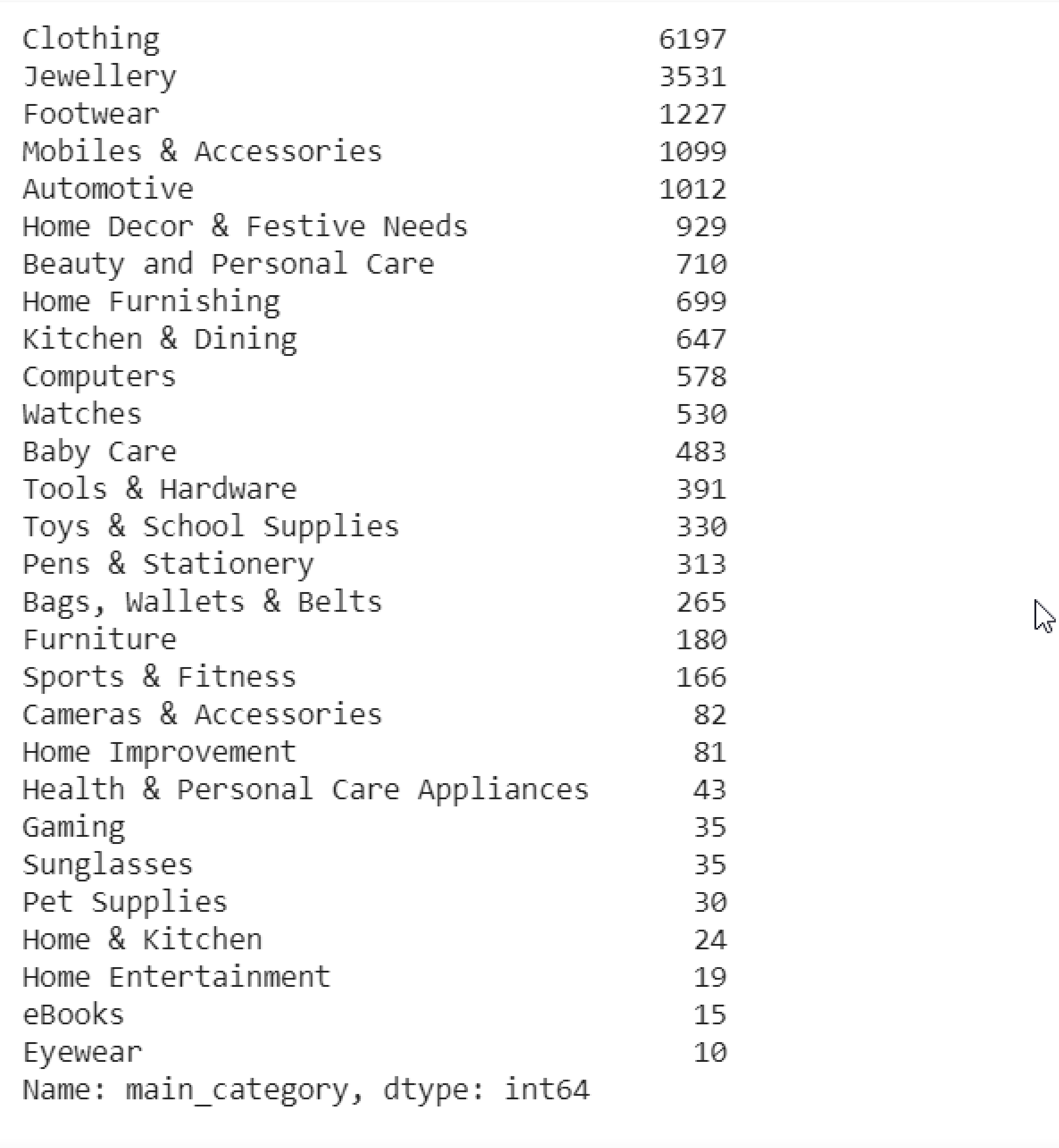
Started with discovering about the dataset , removed the unnecessary which would not help in predicting the product category for the columns like

```
"uniq_id" , "crawl_timestamp" , "product_url" ,"pid" , "retail_price" ,
"discounted_price" , "image" ,"product_specifications",
"is_FK_Advantage_product" , "product_rating" , "overall_rating" ,
"brand" , "product_name"
```

Although product specification could be of use to the model , but I only went away with product description to predict category

Using the product category tree I constructed the a main category column consisting of only first word of the category before first (“<<”)
Viewing the unique categories we got many categories which were occurring only once or twice , so I removed those categories from the dataset

Remaining categories are



```
Clothing                                6197
Jewellery                               3531
Footwear                                1227
Mobiles & Accessories                   1099
Automotive                              1012
Home Decor & Festive Needs               929
Beauty and Personal Care                 710
Home Furnishing                          699
Kitchen & Dining                         647
Computers                                578
Watches                                  530
Baby Care                                483
Tools & Hardware                         391
Toys & School Supplies                   330
Pens & Stationery                        313
Bags, Wallets & Belts                    265
Furniture                                180
Sports & Fitness                         166
Cameras & Accessories                     82
Home Improvement                          81
Health & Personal Care Appliances         43
Gaming                                    35
Sunglasses                                35
Pet Supplies                              30
Home & Kitchen                            24
Home Entertainment                        19
eBooks                                    15
Eyewear                                   10
Name: main_category, dtype: int64
```

We can see we have got an imbalanced dataset with clothing constituting of major portion of the dataset

Next I removed the rows from the dataset where we got empty categories or empty descriptions

After some some processing we got columns with description and main category , now goal is to predict main_category from the product description

I tried various models on this dataset

1. First model I tried was word2vec for getting the embedding of the description text , word2vec is **context free** embedding in which we dont get relation of every word in sentence with every other word in the sentence , now I averaged the embedding of every word in the description and fed the embedding to the neural network for multiclass classification .

Word2Vec with neural network did not give accurate results for me with accuracy of 37 % and F1 score which is 2 times (Precision * Recall) / (Precision + Recall) (closer the F1 score to 1 much better is the model) , the F1 score I got was 0.06 which is pretty low
I also tried **FastText** to get the embeddings but this did not give accurate results too

2. Second model I tried was using cosine similarity , I extracted the embedding of the words in the sentence using word2vec and then averaged those embedding the applied cosine similarity with every category label and one with closest will be the text label will be the class for the text

This algorithm is like KNN wherein instead of using distance criteria we are using cosine similarity
This algorithm is very memory intensive and I wanted to try something state of the art ,

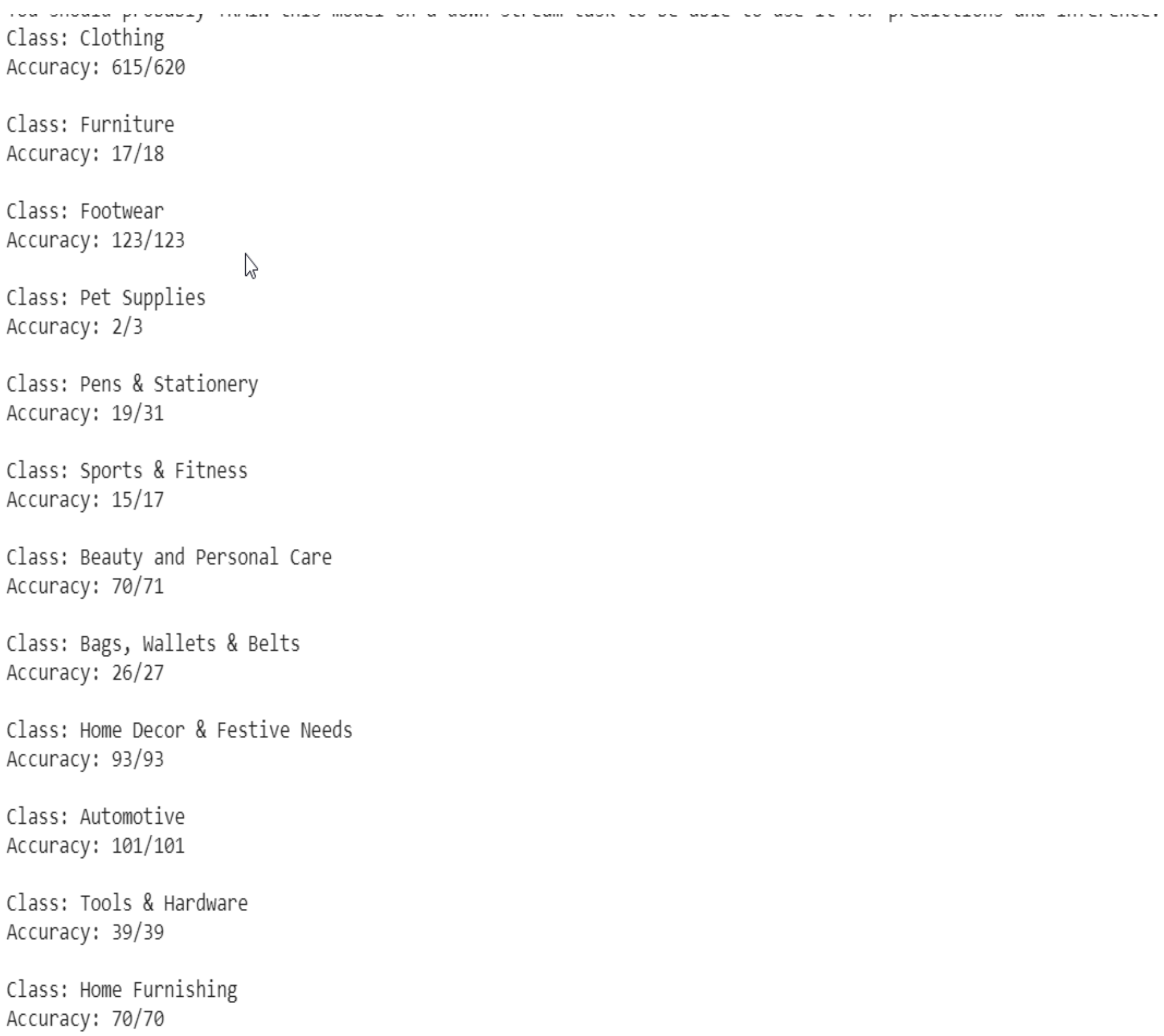
3.Finally the algorithm which I choose was **Bert as a feature extractor and performed sentence classification on in** , I did **transfer learning** on pretrained Bert base model updating the weights for Bert also .

I read the **Attention is all you need** paper which is seminal paper in this field
Bert is based on transformer architecture which only utilizes the encoder part of it
It has around 100 million trainable parameters , pretraining of Bert happens using masked language modelling and next sentence prediction with two special tokens added to every sentence which are <CLS> and <SEP> token
After obtaining the embedding from 12 encoder layers , <CLS> contains the aggregate representation of entire sentence words and this <CLS> representation is fed to Feed Forward Neural Network and this network performs classification on it

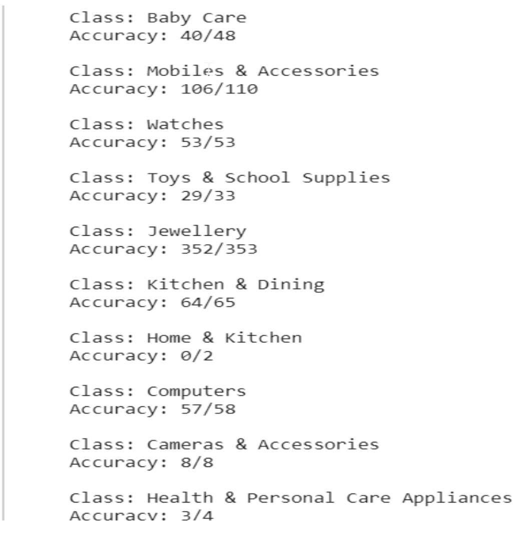
Input embedding for Bert are calculated using the addition of position , token and segment embedding which are then fed to Bert. THe tokenizer we use is Wordpiece tokenizer which also tokenizes to subwords if they are not in vocabulary

After fine tuning Bert for sequence classification task with 4 epochs we reached a F1 score of 0.96 which is very good F1 score

On the validation set the image below shows the accuracy per class



```
You should probably TRAIN this model on a down-stream task to be able to use it for predictions and inference.
Class: Clothing
Accuracy: 615/620

Class: Furniture
Accuracy: 17/18

Class: Footwear
Accuracy: 123/123

Class: Pet Supplies
Accuracy: 2/3

Class: Pens & Stationery
Accuracy: 19/31

Class: Sports & Fitness
Accuracy: 15/17

Class: Beauty and Personal Care
Accuracy: 70/71

Class: Bags, Wallets & Belts
Accuracy: 26/27

Class: Home Decor & Festive Needs
Accuracy: 93/93

Class: Automotive
Accuracy: 101/101

Class: Tools & Hardware
Accuracy: 39/39

Class: Home Furnishing
Accuracy: 70/70
```



```
Class: Baby Care
Accuracy: 40/48

Class: Mobiles & Accessories
Accuracy: 106/110

Class: Watches
Accuracy: 53/53

Class: Toys & School Supplies
Accuracy: 29/33

Class: Jewellery
Accuracy: 352/353

Class: Kitchen & Dining
Accuracy: 64/65

Class: Home & Kitchen
Accuracy: 0/2

Class: Computers
Accuracy: 57/58

Class: Cameras & Accessories
Accuracy: 8/8

Class: Health & Personal Care Appliances
Accuracy: 3/4
```

Above image shows BERT has received pretty good results !!

Conclusion -

I learned a lot during this task about NLP , and I really want to do research at MIDAS

BERT which is currently SOTA gave use pretty good results with pretty good results on the validation set

Other models which I want to try definitely is **XLNet , Roberta and Hierarchical Attention and other context based models** , and bigger the training dataset better would be the accuracy of the model . **Also I could have used other metrics for accuracy of the model like Mathews Coefficient**

Thank you
Tanmay