JASON ACHONU

MACHINE LEARNING 2 PROJECT

Introduction. An overview of the project and an outline of the shared work.

The aim of this project is to build a Convolution Neural Network to Identify different types of online clothing items such as outwear, footwear, skirts, Bags, Tops and Dresses The model will be evaluated with the following metrics: accuracy, F1 score well as AUC and ROC curves.

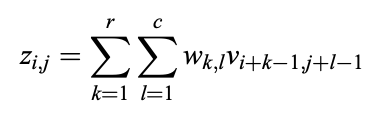
To improve the Convolution Neural Network certain hyperparameters such as Kernel size, number of feature maps and the model optimizer were tuned for different trials and experiments. To fine the best set of parameters for the model.

2. Description of your individual work. Provide some background information on the development of the algorithm and include necessary equations and figures.

The Model chosen to build the Classifier is the Convolution Neural Network. This model was chosen due to the historical performance so far on the model in on the ImageNet Competition from 2012 still date.

Convolutional neural network:

This network is a multilayer feedforward neural network that has two, or three-dimensional inputs. The primary layer for this network is the convolutional layer and the weight functions of this network perform convolution operation on the image, using the convolution kernel. This is done while still preserving the relationship between pixels. To ensure that our code runs correctly the following equation was used to determine feature-map size:



3. Describe the portion of the work that you did on the project in detail. It can be figures, codes, explanation, pre-processing, training, etc.

Data pre-processing

Downloading the Dataset

The dataset was originally complied and uploaded to different cloud storage service by The

University of North Carolina at Chapel Hill. It consists of 404,683 images of clothing from different online stores.

A text file named ‘photo.txt’, was given containing the URL link for each image and a photo ID, also a folder named meta which contain a json for each category was provided. Each json file contain the photo ID of images in that category. I was given the task to provide a script to download each of the images’ using the URL link given by the original authors of the Dataset.

Due to the size/number of images and limited computational resource and storage, a subset of six categories were selected to reduce the number of images. I wrote a script to get the

Using the photo IDs in json files in the Meta folder to randomly select the photo IDs for total of 70,200 images and used the new photo IDs to retrieve the URL link for the photo.txt file.

Den I downloaded the image.

After Downloading, I wrote a script to do a 70%:30% train test split of the images in each of the six categories that would be used for training and test the CNN Model.

I also wrote the script to run the model and calculate the accuracy of the train and test set, plot the ROC, AUC and the confusion matrix,

During training part of the code was subsequently changed such as the number of layers, optimizer, number of input and output channels to tune the network.

4. Results. Describe the results of your experiments, using figures and tables wherever possible. Include all results (including all figures and tables) in the main body of the report, not in appendices. Provide an explanation of each figure and table that you include. Your discussions in this section will be the most important part of the report.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Epochs** | **Batch Size** | **Conv Layers** | **Number of Feature Maps** | **Kernel size** | **Maxpool Layers** | **Accuracy (Train)** | **Accuracy (Test)** | **Time (seconds)** | **Optimizer** |
| 5 | 64 | 5 (Bottleneck) | 32 | 5 | 4 | 65 | 72 | 717 | Adam |
| 5 | 64 | 5 | 32 | 5 | 4 | 67 | 76 | 795 | Adam |
| 10 | 64 | 7 (Bottleneck) | 32 | 5 | 4 | 64 | 71 | 1727 | Adam |
| 10 | 64 | 7 | 32 | 5 | 4 | 70 | 64 | 1654 | Adam |
| 10 | 64 | 5 | 64 | 5 | 4 | 80 | 79 | 1542 | Adam |
|  |  |  |  |  |  |  |  |  |  |

Table 1

This table contains the parameters and the result gotten from each trial in the experiments.

A manual seed was set and then the model ran.

The performance of the training set was less than the performance of the test, hence no overfitting and no need to apply dropout or any regularization techniques required, rather we tried more convolution layers.

In an effort to increase the speed of the model the middle input and output channels for the conv layer was reduced to be more like a bottleneck model from 32 to 16 to compared to it other results that had the same input and output channels (32) across all the conv layers. The bottleneck approach didn’t speed up the model in the 7 convolution

Increasing the convolution layers did not improve the model. But less layers with more output channels (feature maps) improved the accuracy

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 1st | | 2nd | | 3rd | | 4th | | 5th (Best) | |
| Categories | Accuracy | F - Score | Accuracy | F - Score | Accuracy | F - Score | Accuracy | F - Score | Accuracy | F - Score |
| Bags | 74 | 73 | 85 | 78 | 81 | 76 | 76 | 73 | 94 | 80 |
| Dress | 73 | 76 | 78 | 76 | 74 | 75 | 76 | 70 | 78 | 81 |
| Footwear | 86 | 84 | 88 | 88 | 76 | 82 | 85 | 83 | 86 | 90 |
| Outerwear | 65 | 63 | 59 | 65 | 63 | 62 | 58 | 55 | 69 | 72 |
| Skirt | 75 | 77 | 80 | 78 | 81 | 74 | 84 | 73 | 86 | 81 |
| Tops | 60 | 62 | 75 | 68 | 52 | 61 | 40 | 55 | 73 | 73 |

Table 2

This table contains the result for each category in trial in the Experiment.

Across the entire experiment, the Footwear and Bag tops in Accuracy and F1 -Score

5. Summary and conclusions. Summarize the results you obtained, explain what you have learned, and suggest improvements that could be made in the future.

The model performed a bit better than a random guess which is a probability of 1/6 or (16.6%).

however, there are ways this model could be improved going forward. Our best performing model was the one using the SGD optimizer. One way to do this would be to simply gather more data and possessing more processing power. We could also try tuning more hyper-parameter

6. Calculate the percentage of the code that you found or copied from the internet. For example, if you used 50 lines of code from the internet and then you modified 10 of lines and added another 15 lines of your own code, the percentage will be 50−10 50+15 ×100.

download.py

(55 -0) / (55 + 0 ) \* 100 = 100 %

subset\_script.py

(10 - 4)/(10 + 50) \* 100 = 10 %

spilt\_image\_train\_test.py

(10 - 4) / (10 + 45) \* 100 = 10.9 %

Final\_cuda.py

(199 - 38)/(149 + 85) \* 100 = 56 %

All the codes combined

(274 - 46)/ ( 224 + 180) \* 100 = 39 %

7. References.

<https://forums.fast.ai/t/image-normalization-in-pytorch/7534>

<https://stackoverflow.com/questions/47850280/fastest-way-to-compute-image-dataset-channel-wise-mean-and-standard-deviation-in>

<http://acberg.com/papers/wheretobuyit2015iccv.pdf>

<https://github.com/flipkart-incubator/fk-visual-search>

<https://github.com/Airconaaron/blog_post_visualizing_pytorch_cnn/blob/master/Visualizing%20Learned%20Filters%20in%20PyTorch.ipynb>

<https://arxiv.org/pdf/1311.2901.pdf>

<https://stackoverflow.com/questions/47850280/fastest-way-to-compute-image-dataset-channel-wise-mean-and-standard-deviation-in>

<https://github.com/amir-jafari/Deep-Learning/blob/master/Pytorch_/6-Conv_Mnist/Conv_Mnist.py>

<https://discuss.pytorch.org/t/data-augmentation-in-pytorch/7925>

<https://stackoverflow.com/questions/2301789/read-a-file-in-reverse-order-using-python>

<https://pytorch.org/docs/stable/torchvision/transforms.html>

<https://stackoverflow.com/questions/12984426/python-pil-ioerror-image-file-truncated-with-big-images>

<http://www.apsipa.org/proceedings/2017/CONTENTS/papers2017/14DecThursday/Poster%204/TP-P4.14.pdf>

<https://www.quora.com/How-can-I-calculate-the-size-of-output-of-convolutional-layer>

<https://medium.com/@RaghavPrabhu/understanding-of-convolutional-neural-network-cnn-deep-learning-99760835f148>

<https://github.com/pumpikano/street2shop>

<https://stackoverflow.com/questions/123198/how-do-i-copy-a-file-in-python>