TIPR Assignment - III

TA: Anirban

March 7, 2019

Instructions

- This is a coding assignment. The code has to be written in Python. You can use Python version 2 or 3 to solve all the problems. Please mention the version you have used in the report.
- The assignment deliverable is your python code, results, plots and your report of observations.
- Send the link of your github repo to **tipr-e1313@outlook.com** with the following format for the subject name of email: `TIPR_3_<last 5 digits of serial number>`.
- You are encouraged to discuss among yourselves, but **DO NOT COPY** solutions or code. The consequences will be severe.
- Read the complete assignment carefully, before attempting to solve it.
- Follow the instructions provided for how your code is to be run, the formats for input and output files and the naming conventions. — Detailed Instructions
- The submission deadline is **March 19, 2019**.

Best of luck for the assignment. HAPPY CODING!

Rocket.AI : A start-up by Rocket Singh

Project AI (25 Points)

Once upon a time, Harpreet Singh Bedi, a hopeless, in-confident but utterly honest B.Com graduate with approximately 39 percent marks, started his career as a salesman with a big corporate computer assembly and service company, AYS. The top salesmen at AYS acquire large client contracts through bribery. In such a corrupt company culture, Harpreet's honesty only brings him a demotion and humiliation. In spite of all these, Harpreet remains firm on his belief and forms his own company, Rocket Sales Corporation, from within AYS.

All these happened four years ago from now. Since then, his journey has always been a story of success and in the mean time Harpreet Singh Bedi has become famous as Rocket Singh. Recently, he has got a new client(in fashion domain) where they want to intelligently distinguish the designs provided by their designers in an automated way. And this leads to the beginning of *rocket.ai* - an artifical intelligence start-up. Being based in Bengaluru and knowing about the reputation of IISc students, he seeks out for help from the TIPR-2019 batch! Can you, being a Neural Network expert(by now), build such a system for Mr. Rocket Singh? Give your gray matter a roller-coaster ride!

The task here is simple yet subtle. You need to build an **Apparel Identification (Project AI)** system using convolutional neural network(CNN). You have to design a Multi-Layer CNN from scratch - everything starting from initialization, forward propagation and backward propagation(commonly coined as back-prop) - everything by yourself. No library is allowed. *Just joking.*

This time you are **FREE TO USE ANY LIBRARY** of your choice for any task, only restriction is you should use **tensorflow** to build the CNN.

Write your code in such a way that the architecture of neural network can be plugged in from outside. Your code should be able to take the number of layers, activation function and number of neurons in each layer as external input (format mentioned later). You should use *Acccuracy* and *F1-score* as evaluation metrics for you model.

Task - I (2.5 Points)

Experiment with different number of layers of your neural network and give a plot of accuracy vs. layer count, macro-F1 vs. layer count and micro-F1 vs. layer count.

Task - II (2.5 Points)

Try out various configurations of number of neurons in each layer and find the best one. Provide a suitable plot of the same.

Task - III (2.5 Points)

Fix the number of layers and neurons in each layer and feed the hidden layer with the following activation functions - *sigmoid*, *tanh*, *relu* and *swish*. Evaluate the performance of your model. Do it for different configurations of your neural net. Provide a suitable plot of the results obtained.

Task - IV (2.5 Points)

Try out various initialization techniques and observe the results.

Task-V (5 Points)

Use the following *semi-supervised* approach to generate **embeddings**(vectorial representation of data samples) and evaluate clustering accuracy. Take 10-50% of the entire dataset(train+test) in steps of 10% for training purpose and generate the embeddings (output of the penaltimate layer of your CNN) on the remaining data. Fix embedding-dimension of your choice(which gives best result). To check the definition of clusetring accuracy — Look Here

Task-VI (5 Points)

Plot a **t-SNE** visualization in 2-dimensional space on the obtained embedding in the previous task and observer if they are clustered into distinct groups as expected.

Task-VII (5 Points)

Compare your CNN results with MLP(either use keras or the one you designed in the previous assignment).

Project Jungle : Save The Animals (25 Points)

Rocket Singh is not only a great salesman and entrepreneur but also a huge nature and animal lover. Upon hearing about the accidents that keeps on happening on the roads going through jungles and killing thousands of animals every year by different vehicles, he gets immensely sad and disturbed. To resolve this wildlife conflict, he thinks of building an automated alarm device(talking about IOT, a craze now a days) that will recognise the running vehicles and trigger an alarm on real-time if some animal is coming on its way. So he again, asks for TIPR-2019 students for their help to fulfil his dream to build such device.

A big chunk of this project relies on the capability to recognise the animals and the vehicles accurately. Your job is to design an efficient computer vision system using CNNs that can differentiate between animals and vehicles, and that too extremely fast.

Repeat all the tasks mentioned in the earlier scenario.

Idea to ponder about : This part has nothing to do with the assignment, but if you are someone who loves to build systems in practice, you can actually take it further and think about building such an alarm. Use a small camera attached with a small IOT device and deploy your model into it. And then experiment in real time how it works!

Datasets Description

1. CIFAR-10

The CIFAR-10 dataset consists of 60,000 colour images in 10 classes, with 6000 images per class. There are 50000 training images and 10000 test images. The size of each image is 32×32.

The dataset is divided into five training batches and one test batch, each with 10000 images. The test batch contains exactly 1000 randomly-selected images from each class. The training batches contain the remaining images in random order, but some training batches may contain more images from one class than another. Between them, the training batches contain exactly 5000 images from each class.

For more details about this dataset, visit www.cs.toronto.edu/ kriz/cifar.html.

2. Fashion-MNIST

Fashion-MNIST consists of 60,000 training images and 10,000 test images. It is a MNIST-like fashion product database. The developers believe MNIST has been overused so they created this as a direct replacement for that dataset. Each image is in greyscale and associated with a label from 10 classes. The size of the individual image is 28×28.

For more details about this dataset, visit github.com/zalandoresearch/fashion-mnist.

How to run

Far each of the two datasets mentioned above, there will be hidden test files. The performance of your code will be tested on these test data. I should be able to run the code in the following format:

To test the model :

```
python main.py --test-data <test_data_directory_path> --dataset <dataset_name>
```

To train and then test the CNN :

```
python main.py --train-data <train_data_directory_path> --test-data
<test_data_directory_path> --dataset <dataset_name> --configuration
<layer_config> --activation <activation_func>
```

Assignment Deliverable

- Prepare a report and name it `TIPR_Report_3.pdf`. In your report, you need to briefly describe what you have done, present the results (in a form you think is good) and provide a brief discussion of the results you

obtained. It should contain the answers of the tasks, plots and output data for the given datasets. Please make it concise and to-the-point.

- The name of the plots asked for should follow the following naming convention: `part_i_task_j.png`. For example, the filename for part-one task-I should be `part_1_task_1.png`. Put these plots into the `output_plots/` directory.
- The name of output file for various tasks needs to be `part_i_task_j.txt` or `part_i_task_j.pdf`. Put them into the `output_data/` directory.

Clustering Accuracy

Clustering is an unsupervised method of grouping the data samples into multiple communities or clusters. Use embedding as the features for data samples and then apply KMeans/Spectral Clustering. KMeans/Spectral Clustering just divides the data into different groups. To find the test accuracy we need to assign the clusters with an appropriate label and compare with the ground truth community labels. For finding the test accuracy we use unsupervised clustering accuracy which uses different permutations of the labels and chooses the label ordering which gives best possible accuracy. Mathematically, $Acc(\hat{\mathcal{C}}, \mathcal{C}) = \max_{\mathcal{P}} \frac{\sum_{i=1}^{n} \mathbf{1}(\mathcal{P}(\hat{\mathcal{C}}_i)=\mathcal{C}_i))}{n}$. Here $\mathcal{C}$ is the ground truth labeling of the dataset such that $\mathcal{C}_i$ gives the ground truth label of ith data point. Similarly $\hat{\mathcal{C}}$ is the clustering assignments discovered by some algorithm, and $\mathcal{P}$ is a permutation of the set of labels. We assume $\mathbf{1}$ to be a logical operator which returns 1 when the argument is true, and otherwise returns 0.

Sample Input and Output

The sample format for running the code when I want to just test your neural network is as follows:

```
python main.py --test-data <test_data_directory_path> --dataset Fashion-MNIST
```

The sample format for running the code when I want to both train and test your neural network with a pluggable configuration is as follows:

```
python main.py --train-data <train_data_directory_path> --test-data
<test_data_directory_path> --dataset Fashion-MNIST --configuration [3
5 7]
```

In the previous comman, the value corresponding to the configuration parameter signifies that there will be three hidden layers having 3, 5 and 7 neurons respectively. The input and output layer configurations you will set yourself.

Here is how the sample output would look like in the terminal when I would run the code with test data for, e.g., *Fashion-MNIST* dataset.

```
Test accuracy ::  78.32
Test Macro F1-score ::  47.35
Test Micro F1-score ::  53.23
```