HW1

This assignment was locked 9 Sep at 23:59.
You are going to repeat this exercise in both TensorFlow (part1) and PyTorch (part2)

Use the provided datafile and complete the following steps. Complete the assignment within a .ipynb notebook. Submit either the .ipynb, .html, or .pdf of the results.

Step 1. Import the data from the given .csv file (diabetes.csv Download diabetes.csv). The first few columns contain the data while the last column is a binary class label of 0 or 1

Step 2. Preprocess the data as you see fit and show basic data visualization for better understanding of data

Step 3. Split into 70% train 30% test

Step 4. Modify the model shown in the demo to take the new data

Step 5. Train the model on the data

Step 6. Make predictions on the test data and compare them to the test labels

Step 7. Calculate the accuracy of your results

Step 8. Hypothesize changes you can make to the model. Type out your hypothesis as comments in the code or in a text cell. (ex. doubling the width/depth of your NN, experiment thoroughly)

Step 9. Test your hypothesis by training your new models

Step 10. Show the results of your changed models by repeating steps 6 and 7. Discuss the differences and explain why.

HW2

This assignment was locked 17 Sep at 23:59.

Use the code provided in demo 02 to complete this assignment.

Step 1. Follow along with the tutorial to gain an understanding of the process

Step 2. In a new .ipynb notebook, reproduce the results utilizing the "QMNIST" dataset

Step 3. Report on the results in terms of prediction accuracy on the train and test datasets

Step 4. Choose one of the proposed modifications below:

- Add another Dense layer of 128 nodes
- Increase the current number of nodes in the layer to 256

Hypothesize how it would change the performance results

Step 5. Modify the model based on the chosen method and train

Step 6. Report on the results of the modified model and if it matches your hypothesis

Step 7. Experiment with different optimizers, loss functions, dropout, and activation functions, and observe the change in performance as you tune these hyperparameters.

Step 8. Show an example of a backpropagation algorithm by hand (one round of forward step and backward step on a smaller network by performing derivatives by hand instead of using coding libraries, you can use the reference video:

https://www.youtube.com/watch?v=0e0z28wAWfg)

Links to an external site.

Homework 03 - Custom CNNs

Refer to the demo notebook shown in the classroom for examples of setting up a custom dataset.

Step 1. Create your custom dataset featuring 3 custom categories of at least 100 images each

Step 2. Split this data between 80% training and 20% test

Step 3. Preprocess the data as you see fit

Step 4. Create a Convolutional Neural Network model to learn about your training set

Step 5. Make predictions on the test data and compare them to the expected categories

Step 6: Use GoogleNet(InceptionNet) and add a LinearLayer on top of it.

Step 7: Train the GoogleNet model and compare the accuracy with the first model.

Assignment 04 - Transfer Learning and Bounding Boxes and YOLOV8

15 pts
Part1: Using available pre-trained models for object detection, conduct inference on a short video (5-10 seconds) of a street scene drawing bounding boxes around detected vehicles.

Step 1. Collect a source video. It may be necessary to divide the video into discrete image frames.

Step 2. Conduct inference on each frame of the video, drawing bounding boxes around detected vehicles.

Step 3. Format the results back into a video.

Use Pytorch.

Upload a .zip file containing your .ipynb notebook containing the code utilized and two video files: before inference (without bounding boxes) and after inference (with bounding boxes)

10 pts + 5 pts

part 2:

EXTRA CREDIT if you do it in Detectron2 - 5 pts

Follow the steps in YOLOV8 and attach a screenshot of object detection

Windows:
https://medium.com/@pat.x.guillen/a-step-by-step-guide-to-running-yolov8-on-windows-122cb586b567

Links to an external site.

Mac: https://pysource.com/2023/03/28/object-detection-with-yolo-v8-on-mac-m1/

Links to an external site.

https://universe.roboflow.com/pxrksuhn/aihub-aizqc/dataset/1#

Use the dataset mentioned above.

Homework 05 - Basic GANs

Watch this tutorial at:
https://drive.google.com/file/d/16uUSVbShp0uNDqIsCitaiZN33pdkkt27/view?usp=share_link

Links to an external site.

Follow the demo tutorial for basic Deep Convolutional GANs: mnist_gan.ipynb **(pytorch**)

Part 1: GAN

Step1. Load the Fashion MNIST dataset and replace the original dataset used in the demo.

Step2. Train the GAN model to produce images in this new domain.

Step3. Attempt to save and display at least 3 samples produced from your GAN model. Describe and discuss any difficulties faced in this process.

Step4. Save the model weights in the folder to be used later.

Step5. Load the model using the saved weights.

Step 6: Re-train the GAN model to see if you can improve.

Step 7: Save the model weights without using checkpoints.

Step 8: Load the model weights without using checkpoints.

Part 2 LSGAN:
Repeat the steps 1-6 with Least Square GAN and compare it with GAN results

Homework 06 - Advanced GANs

This assignment was locked 22 Oct at 23:59.

Part 1 CycleGAN

1) Load the Monet_to_Photo Dataset using **Pytorch.**

monet_to_photo

Links to an external site.

2) Create and train Cycle GAN on this dataset.

3) Show your results(Few converted images).

4) load your **favorite image (or your picture)** apply Cyclegan and upload it with the report.

Part 2

Follow the DCGAN tutorial or write your own, experiment with different parameters or architectures (as you find suitable), document your findings, and visualize the generated results.

DCGAN Tutorial

Links to an external site.

Homework 07 NLP

This assignment was locked 30 Oct at 23:59.

You will be working with basic Word2Vec Problems.

Step 1: Load the Wikipedia GLoVE Word2Vec.

Step 2: Show how similar are these words:

Man and Woman

Chair and Throne

water and baby

Step 3: Using these provide analogies for the following:

_____ is to King as Woman is to Man.

_____ is to Princess as Man is to Woman.

_____ is to a woman as a child is to an adult.

Step 4: Apply Naive-Bayes Classifier on the Spam-Ham dataset shown in the demo.

Homework 08 LSTM Stock Predictions

This assignment was locked 4 Nov at 23:59.

Step 1. Import the data from the given .csv file (Google_Stock_Price_Train.csv

Download Google_Stock_Price_Train.csv

).

Step 2. Preprocess the data as you see fit

Step 3. Split into 80% train 20% prediction

Step 4. Train the model on the data

Step 5. Report on the results of your model

Homework 09 Transformer Based Text Translation

This assignment was locked 24 Nov at 23:59.

Follow the official **Pytorch demo** tutorial for Transformer Translation:

Links to an external site.

(demo in TF: Transformer model for language understanding)

Links to an external site.

Watch this tutorial at:
https://drive.google.com/drive/folders/13CYPly0fhzWT0m-t7p_YtK2xuyLTvlZY?usp=share_link

Links to an external site.

Step1. Run the demo and train a model on the original German-to-English training set.

Step2. Train a **New** model of the same architecture on the opposite training set (English-to-German)

Step3. Insert novel sentences into your English-to-German model. Take the output and feed it to the original German-to-English model. Observe and report qualitatively on the results

Homework 10 Transformer II

Follow the demo and run the demo notebook on your local machine.

Watch this tutorial Video Demo here

Links to an external site.

.

BERT for QnA:

1. Load the SQUAD 2.0 (https://pytorch.org/text/stable/datasets.html#torchtext.datasets.SQuAD2

Links to an external site.

) dataset from pytorch dataset library. take minimum 20 QnA pairs.

2. Display a few raw QnA data samples.

3. Preprocess/Clean dataset as per BERT format.

4. Train the BERT QnA model. Evaluate the model.

5. Perform an Inference and show the predicted vs ground truth answers.

Homework 11 OpenAI API Overview

Write prompts using OPENAI API keys that perform the following task and display the output:

1. Write a chatbot prompt to iteratively create a sequence of chats on one particular topic.

a. Ask the bot to solve one complex math problem.

b. Give a PDF and website document; ask the bot to rewrite and answer questions on the given PDF and website.

c. At the end, ask the bot to summarize your chat.

Homework 12 LangChain

Go through the demo file and run it in your local/Google colab for hands-on:

Part A: Build a code understanding model. Upload your own custom code files to the model and ask questions based on the code file as context.

Part B: Write a chatbot prompt to iteratively create a sequence of chats on one particular custom data.

1. The chatbot should be able to answer the questions based on the text data or multiple documents.

2. The chatbot should save the conversation in the memory.

2. Summarize the chats at the end of the conversation.

You may use other Generative AI APIs, not just limited to OpenAI API