Final Project

Amir Sadovnik
COSC 525: Deep Learning (Spring 2021)

1 Overview

The goal of this project is for you to further explore a certain subject that we covered in class or a new subject related to it (based on some previous work). This is an open project in a sense that you will propose the problem you would like to work on and the solution you are planning use. Your main goal should be to explore the characteristics of your problem/solution under the deep learning framework

2 A Few Guidelines

1. Most projects fall into one of these three categories:
   (taken from http://cs229.stanford.edu/projects.html)
   - **Application Project:** This is by far the most common: Pick an application that interests you, and explore how best to apply learning algorithms to solve it.
   - **Algorithmic project:** Pick a problem or family of problems, and develop a new learning algorithm, or a novel variant of an existing algorithm, to solve it.
   - **Theoretical project:** Prove some interesting/non-trivial properties of a new or an existing learning algorithm. (This is often quite difficult, and so very few, if any, projects will be purely theoretical.)
2. Your work should be novel in some respect. That is, it should either address a new domain or further examine certain solutions in a specific domain.
3. Your project will not be judged by how well it succeeded, but rather by how well it was motivated/thought of/tested. Therefore, feel free to try some far fetched ideas.
4. You should not propose a completely new idea. Rather, you should further develop an issue explored in the course or from a paper, or the combination of two issues.

5. Try to find a recent paper which addresses the topic of your choice, and relate your work to it.

6. Try to not introduce multiple novelties - try to focus on one aspect so you can explore it sufficiently.

7. Make sure that the question you bring up in your paper, is answered by your paper.

8. Feel free to integrate things you are already working on into your project.

9. Good previous work can be found at the ICML and NeurIPS conferences.

10. You will be using Google Cloud to run your code. Details will follow.

11. Teams can be 2-3 students.

3 Sample Projects

- Algorithms:
  - *Exploring Active Learning in Medical NLP*
    Exploring ways to select data for training in a more efficient way.
  - *Merging Lottery Tickets*
    Took a method which prunes neural networks and tried to merge them.
- Using GANs to supplement data:
  - *Traffic Sign Recognition: A Test of Small Neural Networks and Expanded Datasets*
    Tried to use GANs to generate traffic signs to improve classification rate
  - *Improving Coronavirus Disease Detection fromLung X-Ray Images Using Generative Adversarial Networks*
    Tried to use GANs to generate x ray images to improve classification rate
  - *Mammography Data Augmentation using conditional DCGAN and Classification of different Image Resolutions*
    Tried to use conditional GANs to generate mammography data to improve classification rate
- Looking at changing input (pre-processing):
  - *Manufacturing Feature Recognition with a 2D Convolutional Neural Network*
    Use a novel 2d feature representation and a use a CNN on that (vs usual 3d).
  - *Classification of COVID-19-Related Pneumonia*
    Mostly tried to add preprocessing + auto-encoder methods before classification.
  - *Measuring the Impact of Unsharp Masking Applied to X-ray Bone Images in Convolutional Neural Networks* Mostly tried to add preprocessing methods before classification

- Applying network on novel application (or a different type of network):
  - *Day-ahead Precipitation Class Prediction via Stacked Denoising LSTM Autoencoders*
    Used a stacked LSTM auto-encoders on a problem they have not been used before.
  - *Extracting model Hamiltonian for a complex quantum material*
    Used autoencoders for physics application.
  - *Deep Learning Based Forecast for Solar Radiation Storms*
    Use a combination of convolutions and LSTM to forecast radiation storms.
  - *Deep Learning in Classification of Power System Outages*
    Introduce new features to previous work and change the design of the CNN/RNN.
  - *Applying Convolutional Neural Networks to Time-Frequency Responses of WiFi Signals*
    Applied CNN's on novel problem.
  - *Application of Convolutional Neural Networks to EGG Signals to Classify Mental State in n-back Task* Applied CNN's on novel problem.
  - *Drone-assisted Parcel Delivery Problem: A Deep Reinforcement Learning Approach*
    Use of deep learning with reinforcement learning on novel problem.
- Generating Images:
  - *Scene Manipulation of Landmark Images Using Deep Neural Style Transfer Networks*
    Devise a simplified style transfer network based on VAE and experimented with it.
  - *Single Image Super Resolution: Exploring*
    Tried to create a very small super resolution network.

Check out this link for more examples of projects: http://cs229.stanford.edu/projects.html

4 Coding(30%)

You are required to write your code in python3. You are allowed to use certain external libraries (i.e. Tensorflow, numpy, matplotlib). However, if you are unsure make sure to check with me in order to get my approval.

In addition, you may use other publicly available code - given that you cite it, and describe the exact changes you made to it.

5 Report (70%)

I will upload a Latex template to Canvas. Your report will need to include the following sections:

1. Abstract(5%)
   Make sure to clearly state the novelty of your work in the abstract.

2. Introduction(5%)
   In this section you should discuss the question you are planning to investigate. Make sure to describe the problem clearly, and how you are planning to solve it. Also, clarify what your new contribution is (as opposed to previous works).

3. Previous Work(10%)
   Discuss the previous work you are using as a starting point/reference. If you are working on a different domain include work that works on the same or similar domain. If you are using a new method, discuss other works which use similar methods. Note that you can use both academic papers in addition to online articles. Make sure to cite all sources.

4. Technical Approach (10%)
   Describe the problem you are working on including the rules, goals, etc. Assume that the reader has not been introduced to this domain problem previously. Describe your network architecture/pre-processing/learning method. If you are examining different options describe them all. In addition, add diagrams to clarify.

5. Dataset and Implementation (10%)
   Describe the details of your dataset and implementation. This should include things like the size of your dataset, training/testing split, network hyper-parameters value, training time, hardware used, etc.

6. Experiments and Results Analysis(20%)
   Present the results which answer your hypothesis questions. These should include different graphs which clarify the answer and examine different aspects of your solution. If not described in the implementation section, make sure to carefully describe how your experiments were conducted (how many runs, how it was initialized, etc. This section is the most important, and will be evaluated both on presentation and correctness.

7. Conclusion(5%)
   What have you achieved in this project? What have you learned?

8. References

9. Appendices:

   - Code Design(5%)
     Describe the general structure of your code including functions, classes, and data structures used. If you are using code written by someone else make sure to cite it and emphasize the parts that you have written yourself.
   - Workload Distribution
     Describe how work was divided between the team members

6 Submission

You are required to submit one zip file with both he report and your code.