EE 541 – Computational Introduction to Deep Learning

Initial Project Proposal

©Zijian Cheng, Wenhui Cui, Rongke Zhang

November 19, 2021

Project Title: PetFinder.my - Pawpularity Contest

Project Team: Zijian Cheng, Wenhui Cui, Rongke Zhang

Competition summary: For this project we propose to create a model to predict the popularity (pawpularity) of shelter pet photos. We will build a model to rate the cuteness of given photos using the raw images and their metadata. The prediction made would be compared with the pet profile's page statistics on PetFinder.my website. The purpose of this program is to guild shelter owners to improve the appeal of the pet profiles and to enhance their photo quality.

Figure 1: sample pet photo

Competition page: https://www.kaggle.com/c/petfinder-pawpularity-score

Competition data: The competition dataset is 1.04 GB GB. The training data consists of image files and a .csv file. The train/ image file consists around 10000 training images in jpg format with their unique pet profile id. Pet images are colored pictures of assorted dimensions and sizes. The train.csv file consists of training set photos' metadata and their pawpularity score. The matadata contains labels of 1(Yes) or 0(No) for 12 features of each training set photo. The testing data has a similar format of the training data, with 6800 pet photos similar to the training set photos and randomly generated metadata similar to the training set metadata. It also includes a sample submission file in the correct format.

Primary References and Codebase: We propose to build on the approach used in

- Zhang, H., Cisse, M., Dauphin, Y.N. and Lopez-Paz, D., 2017. mixup: Beyond empirical risk minimization. arXiv preprint arXiv:1710.09412.
- Torrey, L. and Shavlik, J., 2010. Transfer learning. In Handbook of research on machine learning applications and trends: algorithms, methods, and techniques (pp. 242-264). IGI global.
- Tan, M. and Le, Q., 2019, May. Efficientnet: Rethinking model scaling for convolutional neural networks. In International Conference on Machine Learning (pp. 6105-6114). PMLR.
- Guo, M.H., Liu, Z.N., Mu, T.J. and Hu, S.M., 2021. Beyond self-attention: External attention using two linear layers for visual tasks. arXiv preprint arXiv:2105.02358.
- Github codebases: Beyond Self-attention: External Attention using Two Linear Layers for Visual Tasks, EfficientNet

Architecture Investigation Plan: First, we adopt the *EfficientNet* architecture, which scales all dimensions of depth/width/resolution based on a compound coefficient, as our baseline model. Prior to training, we first apply a data augmentation strategy called *mixup*, which linearly combines training samples from different categories and generate new samples. Additionally, we will utilize Meta data into a neural network architecture. Once all of the necessary data visualization and analysis is complete, we will consider combining the results from both image data and meta data. Based on the baseline models described above, we will apply transfer learning to *EfficientNet* model. Transfer knowledge from other publicly available ImageNet datasets, since image data from different domains might share common low-level visual features. To further boost the model performance, we also plan to apply a novel self-attention mechanism called *External Attention (EA)*. Self-attention updates the feature at each position by computing a weighted sum of features using pair-wise affinities across all positions to capture the long-range dependency within a single sample. EA is based on two external, small shared memories, which can be implemented easily by simply using two cascaded linear layers and two normalization layers. EA has been demonstrated as powerful tools in image classification tasks.

Estimated Compute Needs: On the basis of the size of the data set and the benchmarks in this Lambda Labs Blog, we estimate that one training run for our baseline EfficientNet architecture will take 8 hours on a single NVIDIA P100 GPUs with 16GB memory, which is the GPU resource in the Kaggle instance.

Team Roles: The following is the rough breakdown of roles and responsibilities we plan for our team:

- Wenhui Cui: Baseline model coding and experimenting; External Attention implementation; final report and video production;
- Zijian Cheng: Baseline model coding and testing; transfer learning strategy implementation; final report and video production;

- Rongke Zhang: Data collection and cleaning; data augmentation; baseline model coding; final report and video production;

All team members will work on the final report and video.