Dog Breed Classifier

Machine Learning Nanodegree Project Proposal

Xin Huang

Domain background

According to the study of Internet Center (IDC), 70% of information nowadays is transferred via image or video[1]. Recognizing dogs according to their breed is a well-known problem in ML even for humans. There are hundreds of breeds in existence which are grouped into 10 distinct groups (Group 1-10)[2] according to physical characteristics. This is a multi-class classification problem where we can use supervised machine learning to solve this problem. Convolutional neural network (CNN), is a class of deep neutral network, most commonly applied to analyze visual imagery[3]. In this project, I will design a "Dog breed classifier" based on convolution neural network framework.

Problem statement

The purpose of the project is to build a machine learning model that can be used to process real-world, user-supplied images. Given an image of a dog, the algorithm will identify the dog's breed. If supplied an image of a human, the algorithm will choose an estimate of a dog breed that resembles the human. If neither a dog or a human is detected, the algorithm will return an error message.

Datasets and inputs

For this project, it accepts images as inputs. All the dataset is provided by Udacity. There is a total of 13233 images of humans and 8351 images of dogs. For dog image datasets, including 6680 images for training, 835 images for validation and 836 images for testing. For each training set, it has 133 dog breeds. The dog dataset is imbalanced as it doesn't have a proportionate amount of data for each of the classes. For the dog images, they have different dimensions and they are colored. The human dataset will be used when we use OpenCV's implementation of Haar feature-based cascade classifiers to find human faces. The dog dataset will be used to when we use pre-trained VGG-16 model to detect dogs in images.

Solution statement

For performing this multiclass classification, Convolutional neural network (CNN) is applied to solve the problem. The solution involves four steps. Frist, we use existing algorithm such as OpenCV's implementation of Haar feature-based cascade classifiers to detect human faces. Second, a pretrained VGG-16 model is applied to detect dogs in user-supplied images. Then after the image is identified as dog/human, we can use CNN model to classify dog breeds. This model should perform a test accuracy higher than 10%, which is set by Udacity. Finally, a transfer learning will be used with a ResNet50 model to significantly boost the accuracy of the CNN model. It is required to achieve a test accuracy higher than 60% set by Udacity.

Benchmark

The CNN model created from scratch must have accuracy of at least 10%. The CNN pre-trained model with transfer learning should attain significantly enhanced accuracy. The CNN model created using transfer leaning must have accuracy of 60% and above.

Evaluation metrics

Accuracy will be the main metric used to test both the benchmark model and the solution model. In the equation shown below, TP, FN, FP and TN represent the number of true positives, false negatives, false positives and true negatives, respectively.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

Project design

Step 1: Import dataset and libraries.

Step 2: Detect human faces using OpenCV's implementation of Haar feature based cascade classifiers.

Step 3: Pre-trained VGG-16 model is used to detect dog images from the datasets.

Step 4: Create a CNN to classify dog breeds from scratch, train, validate and test the model.

Step 5: Create a CNN by using pre-trained weights from a ResNet-50 model. Test accuracy should be higher than 60%.

Step 6: Test the solution model with images from the datasets to see whether the model work or not.

References

[1] Jun-e Liu, Feng-Ping An, “Image classification algorithm based on deep learning kernel function”, scientific programming, vol. 2020, article ID 7607612, 14 pages, 220. https://doi.org/10.1155/2020/7607612.

[2] Fci.be.(2019). FCI breeds nomenclature. http://www.fci.be/en/Nomenclature.

[3] Valueva, M.V.; Nagornov, N.N.; Lyakhov, P.A.; Valuev, G.V.; Chervyakov, N.I. (2020). "Application of the residue number system to reduce hardware costs of the convolutional neural network implementation". Mathematics and Computers in Simulation. Elsevier BV. 177: 232–243. doi:10.1016/j.matcom.2020.04.031. ISSN 0378-4754. Convolutional neural networks are a promising tool for solving the problem of pattern recognition.