Machine Learning Engineer Nanodegree

Capstone Proposal

Anup Kolte
May 15, 2020

Proposal for Capstone Project

Dog Breed Classification

Domain Background

In Dog Breed Classification, we see the dataset contains images of dogs and humans to which we have to

classify dogs based on their breeds. Then why Humans? the images of the humans are used to see what category of dog breeds will they be classified (for fun purpose) Also, when observed, we categorise such works to the field of Computer Vision and Machine Learning, to which there are various works carried on in the above project.

Recently in 2019, Punyanuch Borwarnginn et al proposed the work on dog breed classifcation using the different approaches to classify them based on their breeds to tackle population control, disease breakout, vaccination control and legal ownnership. He and his co workers used 1. Histogram Oriented Gradient and 2. Convolutional Neural Network using Transfer Learning for the classification purpose. On making a comparative study, they found that the Neural Nets had a better performance compared to the HOG descriptor.

(### Breakthrough Conventional Based Approach for Dog Breed Classification Using CNN with Transfer Learning)

Looking into the work carried above, we see how the dog breed classifcation can be used for determining various information. This can be further expanded to determine information on the 1. Types of Dog Breeds preferably chosen by humans 2. Demographic Needs for the Dogs 3. Behaviour analysis of Dogs at different demographic locations, etc.

Problem Statement

Here the goal is to create a Dog Breed Classifier and build an application for the same. The tasks involved are:

1. Download and Process the Images of the Dogs and Humans 2. Detect the Dogs and Humans using the Pre-Trained models such as Haarcascade and VGG-16 3. Build and train a classifier to calssify dog breeds from scratch 4. Also, train the model using transfer learning with an effieciency to be used for application too. 5. Using the App, predict the breed of the dog and also the category of dog breed the human resembles to.

It is application which can be quite handy to recognise the breeds of unknown dogs for the user and also have fun by creating a resemblance of a dog to the given human images.

Datasets and Input

Here, in the Dog Breed Classification, the dataset contains the images of Dogs and Humans. There are a total of 133 breeds, 8351 images for dogs and 13233 images for humans. Using these images as data, it has to be processed according to our needs and a model has to be designed to train our machine.

The input to the neural net is either an image of a dog or human to which we expect the output as the breed of the dog (for dog input) or tyoe if dog the human resembles to (for a human image input).

Solution Statement

The user end application will be designed which will be useful to identify the dog breeds. To tackle this, we need tp train our model to identiy the dog based on the certain features. The convolutional neural network can be used to train the model and based on the evaluation and performance of the model, an application will be built for the user experience.

Benchmark Model

To tackle such data, it is preferably good to go with neural networks. The extraction of features for an image data is quite tedious. However, on using the convolutional neural networks, the features are easily extracted through it. Neural nets try to find a pattern to recognise features for each category of data and classify them based on the similarity measures.

As observed the input is expected to be a fog image or a human image. The output is the class of breed the dog belongs, and the type of dog the human resembles.

We can benchmark all stages of our project workflow separately as given below:-

- VGG16 model used in transfer learning for dog detection should predict dog in images with high precision. This ensures that our dog-detector is well-trained.
- The custom CNN must have some accuracy to get an intuition of whether the model is working, if working it must be able to output only one dog breed among 133 total dog breeds. This will ensure that our custom model is woking and can be trained on full dataset. Finally it should be able to predict with high Precison and high Recall after all stages of our workflow are working.
- Another creiteria would be like having our OpenCV's Haar cascades classifier to work with high precision.

The benchmark for the model can be referenced to the Kaggle leaderboard for dog breed identification competition. The target for this model is to reach a multiclass loss score less than 0.01, which is in the top 100 of the competition. The other benchmark will be 80% prediction accuracy.

Evaluation Metrics

The evaluation metrics that can be used to evaluate the performance of the machine learning model are:

Accuracy: The ratio of correct predictions to the total size of the data (i.e. (TP+TN)/Data Size)

Confusion matrix:

Recall: The ratio of true positives to the true positive and false negative (i.e. TP/(TP+FN))

Precision: The ratio of true positives to the true positive and false positive (i.e. TP/(TP+FP))

Project Design

The following project can be designed according to the below workflow.

1. Data loading and exploration
2. Data augmentation and processing
3. Detect the Dog and Humans using the given detector algorithms
4. Build a CNN training model from scratch using Deep Learning framework - Pytorch
5. Build a training model using the Transfer Learning

Project would be divided in the following steps:

- Import necessary python packages and libraries required for various tasks. Import and preprocess the data and split it into train, validation and test sets.
- Detect human faces using OpenCV's Haar cascade classifiers.
- Detect dogs using pretrained VGG16 model.
- Create a custom CNN to classify dog breeds.
- Algorithms to combine Dog detector and human detector :
  - If dog is detected, return dog breed prediction
  - If human is detected, return resembling dog breed prediction.
  - If neither is detected, return Error as output.

References

1. Paul Viola and Michael J. Jones. Robust real-time face detection. International Journal of Computer Vision, 57(2):137–154, 2004.

2. Rainer Lienhart and Jochen Maydt. An extended set of haar-like features for rapid object detection. In Image Processing. 2002. Proceedings. 2002 International Conference on, volume 1, pages I–900. IEEE, 2002