JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

DOG BREED IDENTIFICATION MODEL USING ARTIFICIALINTELLIGENCE

[1]Dr. Mrs Fatima M Inamdar 1st
[1]Assistant Professor
[1]Computer Engineering ,
[1]Vishwakarma Institute Of Information Technology , Pune,India

[2]Bhavesh Ingale2nd
[2]Student
[2]Information Technology,
[2]Vishwakarma Institute Of Information Technology , Pune,India

[3]Nidhiy Deshpande3rd
[3]Student
[3]Information Technology,
[3]Vishwakarma Institute Of Information Technology , Pune,India

[4]Shrirang Deshpande 4th
[3]Student
[4]Information Technology,
[4]Vishwakarma Institute Of Information Technology , Pune,India

[5]Rushikesh Gobare5th
[5]Student
[5]Information Technology,
[5]Vishwakarma Institute Of Information Technology , Pune,India

Abstract : Dog Breed Identification is a model used for detecting the breed of the image of the dog provided. It is a classification category project where various images of dogs are classified based on their breeds. There are over 120 breeds ofdogs that can be classified in the project. The images classification is one of the most common applications of deep learning. Images of dogs are mostly used as examples for image classification models, as they are relatively easy for the human eyes to recognize.

The paper can give a model which can classify dog breeds. So, it can be used in various scenarios. A pet care application can be a potential use case of the model. The algorithm can be used to be deployed in an application to classify dog breeds. This model can also be used by pet enthusiasts to know a breed of a certain dog. Various NGOs are specially dedicated to care for the stray and the abandoned dogs. These NGO's can use this model on the daily base since availability of veterinary doctors is limited. An identifier like this could be used in bio-diversity studies, helping scientists save time and resources when conducting studies about the health and abundance of certain species populations. These studies are crucial for assessing the status of ecosystems, and accuracy during these studies is particularly important because of their influence on policy changes. Breed prediction may also help veterinarians treat breed specific ailments for stray, unidentified dogs that need medical care. Ultimately, we found dogs to be the most interesting class to experiment with due to their immense diversity, loving nature, and abundance inphotographs, but we also hope to expand our understanding of the fine-grained classification problem and provide a useful tool for scientists across disciplines.

This work was implemented using tensor flow where a number of dog images with their breeds were provided as data set. The source of the data set was Kaggle. We will use data set from Kaggle. The data set contains 10000+ images with 120 different breeds of dogs. The data set will contain null or non-acceptable values (values which are not present in required format) which have to be removed or treated accordingly. The treatment can be done by checking if the data of a particular category are mean based or median based. And then the subsequent values can be utilized. Then using the modified data set we can train our model for the data set. For this step we will be using TensorFlow. TensorFlow is an end-to-end machine learning solution. Now, we can create a tensor of a specific shape to represent various features of data. The model now needs a definition of various Keras sequential layers. Now the model will be built on these layers. This research work will use test data set for the evaluation of

model. Tensor Board metrics will be used to evaluate our created model. And precautions should be taken that an overfit or underfit is avoided. If the metric shows too high accuracy the model has adapted to the training data set and it is an overfit. If the metric shows too less accuracy the model needs more training and it is an underfit. Then using this model, will test on data set.

Index Terms **– Dog Breed Classification, Image Classification,Artificial Intelligence, Keras.**

I. INTRODUCTION

Computer Vision is a Field of scientific study of algorithm which computer and other systems employ to perform a specific task effectively and autonomously without using clear and detailed instruction from humans. Becauseunder such an algorithm computer can learn and make prediction themselves. Training data is built to make predictions instead of explicitly programmed to perform the task through machine learning algorithms based on the sampledata. The algorithms have been used in a wide variety ofapplications, such as stock prediction, recommendation systems, email filtering, and object classification and computer vision, such as autonomous driving, where it is unfeasible to develop an algorithm of specific instructions to perform the corresponding task.

Artificial Intelligence is showing stellar improvements and is being used now in many fields. It's uses range from predictingan outcome, classifying certain objects or just finding patterns in knowledge and applying knowledge. Few examples of classification in AI can be spam detection and sentiments analysis. Spam detection can filter emails and classify an email as important or spam. This feature Of AI can be very useful. Analysis a sentiment of a person face requires find tuned image recognition and analysis. The recent advancements in the field of AI can make this possible. Dog Breed Identification is another such classification problem. It requires a keen amount of image analysis to predict the breed of a dog based on its image. Dog Breed Identification model has to first identify the dog's features and then based on its inference predict the most likely option. The Model required a dataset of 10000+ images which contained 120 distinct breeds of dogs. The dataset also contained some random background with the image of the dog, making it challenging for the model.

Classification is a systematic arrangement in groups and categories based on its features. Image classification came intoexistence for decreasing the gap between the computer vision and human vision by training the computer with the data.[1]

Fig.1. (a) Malamute (b) Eskimo Dog (c) Husky

II. LITERATURE SURVEY

This paper provides an investigation of the classification model utilizing computer vision. A model can recognize items in an image using computer vision. The research can assist us learn about deep learning, Keras, and other optimizers. We will undertake research to choose the best model and optimizer for our needs. There are no disease-related or high-risk operations included in dog breed identification. As a result, that can also be considered when developing this model. A large amount of data must be studied, as well as how a model learns is carried out. This research falls under supervised learning.So, the model learns while knowing the breeds of dogs and triesto correct itself more and more. The model has very complex number of images to process. As distinguishing between 120 breeds of dogs is challenging in terms of computation cost, that will also consider the option of transfer learning using a pertained model. And then using that model for this research purpose. With these all concerns, can conduct a study on select of various tools for this research.

A. Prerequisites:

1) Keras Sequential layers: Keras is a neural network Application Programming Interface (API) for Python that is tightly integrated with TensorFlow, which is used to build machine learning models. Keras' models offer a simple, user-friendly way to define a neural network, which will then be built for you by TensorFlow. TensorFlow is an open-source set of libraries for creating and working with neural networks,such as those used in Machine Learning (ML) and Deep Learning projects. Keras, on the other hand, is a high-level API that runs on top of TensorFlow. Keras simplifies the implementation of complex neural networks with its easy-to- use framework.The Sequential model API is a way of creating deep learningmodels where an instance of the Sequential class is created, and model layers are created and added to it.

For example, the layers can be defined and passed to theSequential as an array:

```
from keras.models import Sequentialfrom keras.layers import Dense
model = Sequential([Dense(2, input_dim=1), Dense(1)])
```

Layers can also be added piecewise: from keras.models import Sequential

```
from keras.layers import Dense model = Sequential() model.add(Dense(2, input_dim=1))
model.add(Dense(1))
```

Keras sequential class is one of the important classes as part ofthe entire Keras sequential model. This class helps in creating a cluster where a cluster is formed with layers of information or data that flows with top to bottom approach having a lot of layers incorporated with TensorFlow. Keras is a model where most of its features are trained with algorithms that provide a lot of sequence to the model. [5]

Syntax representing the creation. tf.keras.Sequential (layers=No_lyr, name=No_lyr)

The Sequential model API is great for developing deep learning models in most situations, but it also has some limitations. For example, it is not straightforward to definemodels that may have multiple different input sources, produce multiple output destinations or models that re-uselayers.

2) *MobileNet V2*: The model used in this is MOBILENET-V2. Pre-trained MobileNet V2 is used to classify the dogimages. It provides real-time classification capabilities under computing constraints in devices like smartphones. MobileNet-v2 is a convolutional neural network that is 53 layers deep. You can load a pretrained version of the network trained on more than a million images from the ImageNet database. The pretrained network can classify images into 1000 object categories, such as keyboard, mouse, pencil, and many animals. As a result, the network has learned rich feature representations for a wide range of images. The network has an image input size of 224-by-224.

MobileNetV2 is a very effective feature extractor for object detection and segmentation. They can also be trained for various sizes of input images to control inference speed.

MobilenetV2 is a refined version of MobilenetV1. This makes it even more efficient and powerful. The MobileNetV2 models are faster due to the reduced model size and complexity. It is apre-trained model for image classification. Pre-trained models are deep neural networks that are trained using a large images dataset. It is trained using a large images dataset. It enables themodel to effectively learn, therefore, we can expect accurate results. It also simplifies the process of image processing.Image processing helps transform the image dataset into a format that the model can understand to give more accurate results.

MobileNetV2 is lightweight making it have high execution speed and it significantly reduces the number of parameters thus making it less complex. It provides a very efficient mobile-oriented model that can be used as a base for many visual recognition tasks. It can also run-on web browsers sincethe model is lightweight as compared to MobilenetV1. Also, browsers have lower computation power, graphic processing, and storage.

3) *Finding the right optimizer*: Adam is an optimization algorithm that can be used instead of the classical stochastic gradient descent procedure to update network weights iterative based in training data. Adam was presented by Diederik Kingma from OpenAI and Jimmy Ba from the University of Toronto in their 2015 ICLR paper (poster) titled “Adam: A Method for Stochastic Optimization“. I will quote liberally from their paper in this post, unless stated otherwise.

The algorithm is called Adam. It is not an acronym and is not written as “ADAM”.

When introducing the algorithm, the authors list the attractive benefits of using Adam on non-convex optimization problems,as follows:

1. Straightforward to implement.
2. Computationally efficient.
3. Little memory requirements.
4. Invariant to diagonal rescale of the gradients.
5. Well suited for problems that are large in terms ofdata and/or parameters.
6. Appropriate for non-stationary objectives.
7. Appropriate for problems with very noisy/or sparsegradients.
8. Hyper-parameters have intuitive interpretation andtypically require little tuning.

Sebastian Ruder developed a comprehensive review ofmodern gradient descent optimization algorithms titled “Anoverview of gradient descent optimization algorithms”published first as a blog post, then a technical report in 2016. The paper is basically a tour of modern methods. In his sectiontitled “Which optimizer to use?“, he recommends using Adam.Insofar, RMSprop, Adadelta, and Adam are very similaralgorithms that do well in similar circumstances. Its bias-correction helps Adam slightly outperform RMSprop towardsthe end of optimization as gradients become sparser. Insofar,Adam might be the best overall choice.[2]

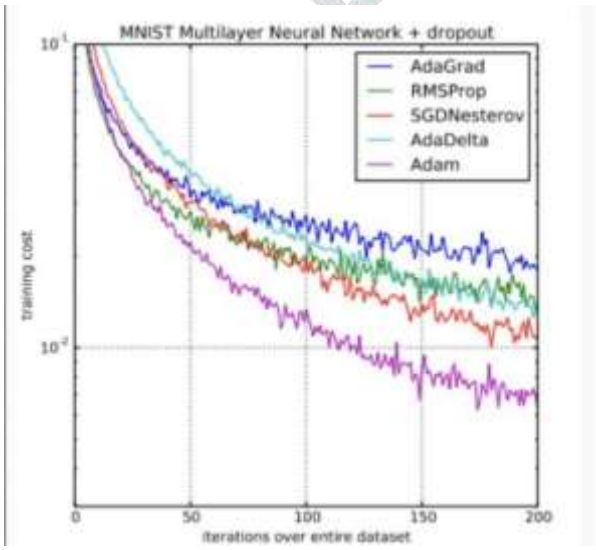

Fig2 : Result

In the Stanford course on deep learning for computer vision titled "CS231n: Convolutional Neural Networks for Visual Recognition" developed by Andrej Karpathy, et al., the Adam algorithm is again suggested as the default optimization method for deep learning applications.

In practice Adam is currently recommended as the default algorithm to use, and often works slightly better thanRMSProp. Adam is being adapted for benchmarks in deep learning papers.

III. IMPLEMENTATION

A. *Data Preprocessing:*

The data preprocessing can often have a significant impact on generalization performance of a supervised MLalgorithm. The Dataset for the given project was a dog breed classification problem from Kaggle. It includes 10000+images of 120 distinct dog breeds. The data set containedimage of a dog along with it's breed. The dataset contains a label.csv file which contains labels of dog breed with theimage name. Now every image there 3 steps to be followed forpreprocessing:

i) Image Resizing: Resizing images is a critical pre- processing step in computer vision. Principally, deep learning models train faster on small images. A larger input imagerequires the neural network to learn from four times as many pixels, and this increase the training time for the architecture. The image has to be resized to a standard size for every image.So, convert the image to (224, 224).

ii) Image Rescaling: When using the image as it is and passing through a Deep Neural Network, the computation of high numeric values may become more complex.

To reduce this, we can normalize the values to range from 0 to

1. In this way, the numbers will be small and the computation becomes easier and faster. As the pixel values range from 0 to 256, apart from 0 the range is 255. So, dividing all the values by 255 will convert it to range from 0 to 1.

iii) Data Batches: We also create a data batch of images of 32 images. We create such batches for two reasons. First being loading only 32 images at a time is less expensive or resource consuming for the model. Secondly, the model won't have to utilize memory for storing error values for all 10000 images. Instead, it stores for 32 images at a time.

B. *Training our Model*:

Creating a model consist of selecting an appropriate model and an optimizer if required for our cause. In our case, we use Keras sequential layers in Mobile net v2 classification model. We use Adam Optimizer. For our output layer we use SoftMaxactivation.

```
MODEL_URL="https://tfhub.dev/google/imagenet/mobilenet
_v2_130_224/classification/4" OUTPUT_SHAPE = 120
model = tf.keras.Sequential([ hub.KerasLayer(MODEL_URL), # Layer 1 (input layer)
 tf.keras.layers.Dense(units=OUTPUT_SHAPE,
                       activation="softmax") # Layer 2 (output layer)
  ])
```

The above code is used to creating our model. It uses mobilenet v2 model. This is a type of transfer learning where our model gets its knowledge from mobilenet v2. The Output Shape is 120 because we have 120 different breeds of dogs.

Softmax is used for output layer in classification networks.[3] The Softmax activation function calculates the relative probabilities. Softmax is a mathematical function that convertsa vector of numbers into a vector of probabilities, where the probabilities of each value are proportional to the relativescale of each value in the vector.

```
model.compile( loss=tf.keras.losses.CategoricalCrossentropy(), optimizer=tf.keras.optimizers.Adam(), metrics=["accuracy"]
  )
```

Computes the crossentropy loss between the labels and predictions. Cross-entropy is a measure of the difference between two probability distributions for a given random variable or set of events.[4] We have studied about Adam Optimizer in previous section and it has been used here.

```
model.build(INPUT_SHAPE)
```

This line actually builds the model with given abovespecifications.

C. *Training the model*

The final training of the model was done using fitting the data to our model.

```
early_stopping=tf.keras.callbacks.EarlyStopping(monitor="val
_accuracy", patience=3)
```

This callback serves the purpose of stopping a model training when it gets the same the validation accuracy for 3 consecutive turns. It is done to not waste computation cost.

```
model.fit(x=train_data,
          epochs=NUM_EPOCHS, validation_data=val_data, validation_freq=1, callbacks=[tensorboard,
      early_stopping])
```

Here, the actually training of our model takes place. We pass the training data and validation data. The number of the epochs that we select are 100. Now the creation and trainingof our model has been completed.

IV. RESULTS

The model gives an accuracy of 0.9986 on validation data set. The model ran for 3 consecutive epochs with the accuracy and received the early stopping keras epoch.

V. CONCLUSION

The model was created using the keras sequential layers on mobilenet v2 classification model. It's accuracy onvalidation data was 0.9986. The accuracy is satisfactory for the purpose of detection of a dog breed. The model can now be extracted and used further in a user interface. It can used byan user based on their needs. The model can be used by an NGO to identify the various dog breeds and their numbers. So, they can keep a track of various breeds of dog. The high accuracy of the model also allows it to be appropriate for veterans.

VI. REFERENCES

[1] M Manoj krishna, M Neelima, M Harshali, *M Venu Gopala Rao Imageclassification using Deep learning,*
[2] Sebastian Ruder B. Simpson, et al *"An overview of gradient descentoptimization algorithms"*
[3] Tomasz Szandała *Review and Comparison of Commonly UsedActivation Functions for Deep Neural Networks*
[4] A Gentle Introduction to Cross-Entropy for Machine Learningby Jason Brownlee on October 21, 2019 in Probability
[5] How to Use the Keras Functional API for Deep Learning by Jason Brownlee on October 27, 2017 in Deep Learning
[6] A Comprehensive Survey of Deep Learning Models Based on KerasFramework Bahzad Taha Chicho1*, Amira Bibo Sallow2
[7] TensorFlow: A system for large-scale machine learning Mart´ın Abadi, Paul Barham et. Al
[8] MobileNetV2: Inverted Residuals and Linear BottlenecksMark Sandler Andrew Howard
[9] Image Classification, Object Detection and Recognition in PythonJason Brownlee et. Al
[10]Keras Convolutional Neural Network Model in Sequential Build of 4and 6 Layer Architectures Praneet Amul Akash Cherukuri, Nguyen Thi Dieu Linh
[11] Mobile net V2 model for Image Classification Ke Dong, Yihan Ruan