Dog Breed Classification Via Machine Learning

Mel Gerst

For

ITCS 5156

UNCC

2024

Problem Statement & Motivation

- **Problem Statement:** Can a dog breed be visually identified from a picture using machine learning techniques, deep learning and a trained algorithm.

- **Motivation:** My motivation stems from prior work with barcode scanners and the current necessity to scan barcodes to identify products. I foresee a future time when barcodes are outdated, and most items can be identified using visual cues. As as result, I am excited to attempt this project of using a dog picture database and machine learning techniques and see if it can identify the breed of a random dog picture.

- In addition, this project will allow me to deploy the tools we have learned this semester, in addition to a few new technologies.

Reference Papers

Primary Paper Reference

- Dog Breed Identification using ResNet Model
  - Reddy, Y. A. ., Kumar, Y. S. ., M, S. ., & Mana, S. C. . (2023). Dog Breed Identification using ResNet Model.
  - *International Journal on Recent and Innovation Trends in Computing and Communication*, *11*(7s), 64–71. https://doi.org/10.17762/ijritcc.v11i7s.6977

Supporting Reference Papers & Projects

- TechVidVan – Dog's Breed Identification using Deep Learning
  - Website - https://techvidvan.com/tutorials/dog-breed-classification/
- Dog Breed Classification Using Deep Learning
  - Varshney, Akash & Katiyar, Abhay & Singh, Aman & Chauhan, Surendra. (2021). Dog Breed Classification Using Deep Learning. 1-5. 10.1109/CONIT51480.2021.9498338.
- Novel Dataset for Fine-Grained Image Categorization
  - Aditya Khosla and Bangpeng Yao and Nityananda Jayadevaprakash and Li Fei-Fei, Novel Dataset for Fine-Grained Image Categorization, First Workshop on Fine-Grained Visual Categorization, IEEE Conference on Computer Vision and Pattern Recognition,2011,June, Colorado Springs, CO, @inproceedings{khosla_fgvc2011}

Primary Paper Overview

- **Dog Breed Identification using ResNet Model**

  Reddy, Y. A. ., Kumar, Y. S. ., M, S. ., & Mana, S. C. . (2023). *International Journal on Recent and Innovation Trends in Computing and Communication*, *11*(7s), 64–71. https://doi.org/10.17762/ijritcc.v11i7s.6977

- This paper used ResNet50, which is a residual neural network that excels at image classification, to classify dogs. Resnet was used to identify the images and uses transfer learning with convolutional neural network to classify a large database of images. Used in conjunction with TensorFlow, a dataset was created, tested, and trained for the detection of dog breeds. 120 dog breeds are covered in the dataset which includes 20,580 images of dogs. In this reference project, all these images are used and converted into a NumPy array and normalized. Then,100 epochs were used with a batch size of 128 to achieve the best accuracy. Using these tools and process the team was able to achieve an accuracy of 91%.

- For my implementation, and depending on computing power requirements and time requirements, I may choose to focus on a fewer number of breeds. I am targeting at least 80% accuracy for my model. As pointed out in later slides, other reference projects using similar approaches were able to achieve 70% and 85% accuracy results, so I hope to be able to achieve at least 80%.

Summary of the Method

Primary Reference: Dog Breed Identification using ResNet Model
– Reddy, Y. A. ., Kumar, Y. S. ., M, S. ., & Mana, S. C. . (2023)

Top Level Summary

- Project used an existing database (Kaggle / Stanford) of dog pictures to train a neural network using transfer learning techniques.
- Trained the model using these pictures and a defined set of labels.
- The overall model is testedfor accuracy, target 80%+.
- Finally, present some personal dog photos and see if the model can successfully identify the breed.

Main Technologies

1) Python
2) Tensorflow: 2.3.1
3) Opencv
4) Sklearn
5) Numpy
6) Pandas
7) Matplotlib
8) ResNet50V2 - Pretrained Residual Neural Network

Major Project Steps

1) Import necessary libraries and prepare coding environment
2) Download and prep dog photos
3) Preprocess dataset
4) Encode, scale and augment dog pictures
5) Split pictures into training and test sets
6) Build the model
7) Train the model
8) Test the model
9) Predict breed using personal photos
10) Summarize results

Dataset References and Link

- The Stanford Dogs dataset contains images of 120 breeds of dogs from around the world. This dataset has been built using images and annotation from ImageNet for the task of fine-grained image categorization. Contents of this dataset:
  - Number of categories: 120
  - Number of images: 20,580
  - Annotations: Class labels, Bounding boxes

- **Dataset Credits Primary: Aditya Khosla, Nityananda Jayadevaprakash, Bangpeng Yao and Li Fei-Fei. Novel dataset for Fine-Grained Image Categorization. *First Workshop on Fine-Grained Visual Categorization (FGVC), IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011.**
- Dataset Credits Secondary: J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li and L. Fei-Fei, ImageNet: A Large-Scale Hierarchical Image Database. IEEE Computer Vision and Pattern Recognition (CVPR), 2009.
- **Dog's Breed Identification Dataset**
  - https://www.kaggle.com/c/dog-breed-identification
  - Kaggle Citation: @misc{dog-breed-identification, author = {Will Cukierski}, title = {Dog Breed Identification}, publisher = {Kaggle}, year = {2017}, url = {https://kaggle.com/competitions/dog-breed-identification} }

Source Code References and Links

Primary: TechVidVan

- TechVidVan offers courses and support for software developers and a strong focus on machine learning, deep learning and software supporting these areas.
- Through their website they have an excellent project example for dog breed classification, with source code and a link to the original Stanford dataset I planned to use.

- **Dog Breed Classification Project Code**
  - https://drive.google.com/file/d/1M1CbfXTzJwnhviVVGmWVvbpM-pMDelN-/view
- TechVidVan Website Link
  - https://techvidvan.com/tutorials/dog-breed-classification/

- Supporting Source Code References
  - **Deploying a Dog Breed Classification ML Model: An End-to-End Guide, Subin Thomas**
  - https://medium.com/@subin60/deploying-a-dog-breed-classification-ml-model-an-end-to-end-guide-fc7c025e13a2

  - **Dog-Breed-Prediction, Prathamesh Parit**
  - https://github.com/prathameshparit/Dog-Breed-Prediction

Summary of Related Works and Links

1) TechVidVan - https://techvidvan.com/tutorials/dog-breed-classification/

- TechVidVan is a group and website with software and machine learning tools and topics. They have conducted a similar project to the primary paper with good explanation of the process, steps and techniques to be successful.
- Through their website they have an excellent project example for dog breed classification, with source code and a link to the original dataset I had planned to use. Therefore, this is a strong reference as a related work.
- Similar tools were used as the primary paper and appears to have used a similar process. One difference from the primary, which I will likely follow, is to test the model with personal files via a simple local file upload, instead of hosting the system on website.
- For this project, the team reached an accuracy of 80.5% which is above my 80%. It will be interesting to see if I can achieve similar or possibly better results.

2) Dog Breed Classification Using Deep Learning

- https://www.researchgate.net/publication/353693466_Dog_Breed_Classification_Using_Deep_Learning
- Varshney, Akash & Katiyar, Abhay & Singh, Aman & Chauhan, Surendra. (2021). Dog Breed Classification Using Deep Learning. 1-5. 10.1109/CONIT51480.2021.9498338.
- This related work also performed dog breed classification using the same Stanford dog database, so is a good candidate for support on this project.
- In addition, the team used two different CNN tools to perform the modeling. Name, they used Inception V3 neural network and the VGG16 neural network. With these models they were able to achieve top accuracy performance of 85% and 70% respectively.
- This team also showed the increase in accuracy of both models as the number of epochs was increased. Again, it will be interesting to see what performance I can achieve with my implementation.

Plan and Timetable

- My plan and expected time-table to complete this project is outlined below.

Project Milestones – Mel Gerst	Prep Time	Completion Date
Complete project proposal and submit	1 week	Sept 15th
Complete reading all primary and related papers	1 week	Oct 6th
Complete primary coding of the project	3 weeks	Oct 20th
Finalize code, train models, complete experiments and record results	2 weeks	Nov 3rd
Complete and submit video presentation	1 week	Nov 17th
Complete and submit final report	1 week	Nov 24th

THANK YOU