INDIAN INSTITUTE OF INFORMATION TECHNOLOGY, ALLAHABAD

PROJECT REPORT

on

CALTECH BIRD DATASET CLASSIFICATION USING CAPSULE NETWORK

Submitted by:

Tejas Ramesh Pawar (IIT2017109)
Kunal Kumar Prasad (IIT2017112)
Gaurav Kumar (IIT2017115)
Mirza Mohd. Aadil Beg (IIT2017145)
Aman Gupta (IWM2017006)

Under the Supervision of:

Dr. Sonali Agarwal

November, 2019

Contents

ABSTRACT

Caltech-UCSD Birds 200 (CUB-200) is a challenging image dataset annotated with 200 bird species. It was created to enable the study of subordinate categorization. Each image is annotated with a bounding box, a rough bird segmentation,and a set of attribute labels. Convolutional neural networks fail in capturing the pose, orientation and view of the images due to the inefficiency of max pooling layer. These limitations are overcome by a novel approach: Capsule Networks. Capsule Networks are made up of layers of capsules representing the instantiation parameters of entities by using the dynamic routing and route by agreement algorithms.

1 INTRODUCTION

Classification of bird species is a difficult problem for both humans and machines to push the limits of visual abilities. While different species of birds share the same basic set of parts, different species of birds that differ dramatically in shape and structure among each other (e.g., consider pelicans versus sparrows). At the same time, even for expert bird watchers (e.g., many sparrow species are visually similar) other pairs of bird species are almost visually indistinguishable. Due to variations in lighting and context and significant variations in posture (e.g., flying birds, swimming birds, and perched birds partially occluded by branches), the intraclass variability is high.

Here, this data is pre-processed before classification. Preprocessing refers to the transformation applied to our data before it is fed to the algorithm for classification. As a technique, we will use preprocessing to convert the raw data into a clean dataset. For our data, we will have to use all three different techniques of preprocessing data. The data will need to be rescale, binarized, and standardized.

Predictive modeling classification is the task of receiving a new observed sample as input and assigning it to one of the predefined categories (y), often known as tags, through the use of a trained model. The role of identification provides the basis for other issues with computer vision such as recognition, localization and segmentation. Given the fact that this function can be viewed as straightforward for humans, a computer-based process is far more challenging; some of the complexities are viewpoint-dependent object variance and the high in-class variability of so multiple object types. Today, engineers and researchers around the world are using convolutional neural networks (CNNs) to solve specific object classification problems, this technique has created remarkable low-test errors on classification tasks across various image types. CNNs also have several disadvantages and limitations, despite their success. At each subsequent layer, CNNs accumulate sets of features; it starts from finding edges, shapes, and finally real objects. Some knowledge about the spatial relationships between these characteristics (perspective, size, orientation) is lost, however. They seem to be easily fooled by images with features in the wrong place (for example, a nose rather than an eye on a human face) or samples of the same images in different orientations. Excessive learning for all possible angles is one way to eliminate this problem, but it normally takes a lot more time and computational resources. In addition, convolutional neural networks may be susceptible to attacks by the white box and the so-called "fast gradient signing method." Recently, a new algorithm called capsule dynamic routing (CapsNet) was proposed to overcome these disadvantages. CNN's is based on the idea of translated replicas of studied feature detectors.In other terms, data can be distributed to other positions regarding properly

trained features collected in one position; this capability has become beneficial for image interpretation. By comparison, CapsNet replaces CNN's scalar-output function detectors with vector-outputs, it also replaces routing-by-agreement with the max-pooling subsampling technique, enabling the replication of learned information across space. Only the first layer of capsules, also known as main capsules, contains groups of convolutional layers in this new CapsNet architecture. Higher-level packets, following traditional CNN principles, cover larger image regions. Nevertheless, unlike regular CNNs, the data about the entity's precise position within the area is retained.

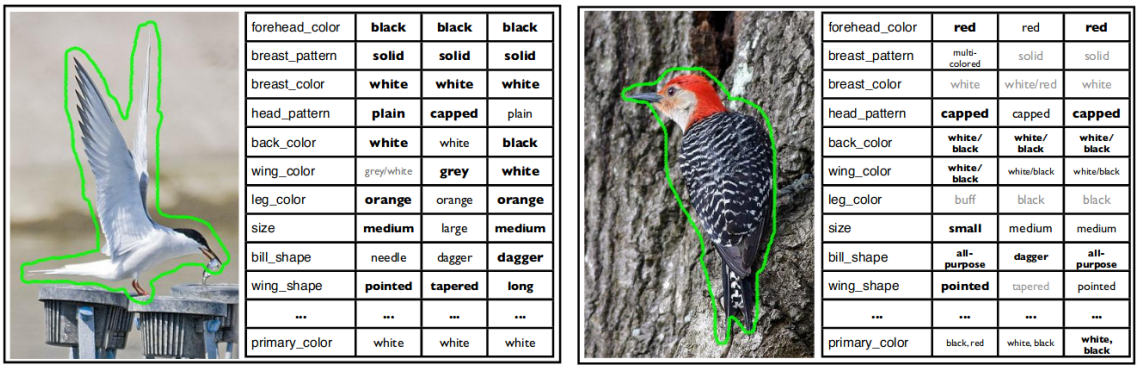

Figure 1.1: Images and annotations from CUB-200. Each example image is shown with a rough outline (segmentation) in green. To the right of each image is a table of attributes

.

2 MOTIVATION

This dataset[5] has fascinated deep learning and data science enthusiasts over the world, endeavouring to push its accuracy consisting of the images. The recent use of Capsule Network on dataset classification has produced a groundbreaking piece of outcome on the contrary to Convolutional Neural Network. The dataset poses numerous challenges starting from preprocessing to the final stages of training and testing.

3 PROBLEM DEFINITION

In this project we are going to train a classification model using the CUB-200-2011 dataset. This dataset contains 200 species of birds,each containing 60 images, and has become a staple for testing new ideas for fine-grained visual classification using Capsule Network.

4 LITERATURE REVIEW

The same dataset has been used and classified using Convolutional Neural Network[6]. The dataset spans over 11,788 images categorized into 200 different bird species, but it does not come with a standard validation set.

Research papers claim that they can get over 80% accuracy on this dataset using only the images, no bounding boxes or parts are needed. Jaderberg et al[1]. claim that they can achieve 82.3% and Krause et al[2]. claim that they can get 84.4% accuracy.

Proposed by Sara Sabour which (and Geoffry Hinton), this model used a Kaggle Dataset on F-MNIST[3]. The drawback of the model was that it took an hour for 28x28

MNIST images which are transferred by 2 pixel which clearly meant that model probably won't be too useful in the immediate future. The dataset had 60,000 images for training and 10,000 images for testing.

Implementation details:

- Keras Implementation of CapsNet in Hinton's paper Dynamic Routing Between Capsules.
- The current version may only work for TensorFlow backend.

The model provided an under fitting validation accuracy of approx 99.5% after 20 epochs. It required about 110 seconds per epoch on a single Nvidia GTX1070 GPU card. Also it provided a test accuracy of 98.6%.

In a paper[4] published on 7th November 2017 by Geoffrey E. Hinton, a capsule is said to be a group of neurons whose activity vector represents the instantiation parameters of a particular entity type such as an object or part of an object. Transformation matrices are used to make predictions of the instantiation parameters of the higher level by the active capsules at a particular level. The mechanism suggests that when multiple layer predictions agree, a higher level capsule becomes active.

The paper shows that the discriminatively trained and multi-layered capsule system used, achieves the best performance on the general MNIST data set. In conclusion, this is considered to be better than a convolutional neural network used on the same data.

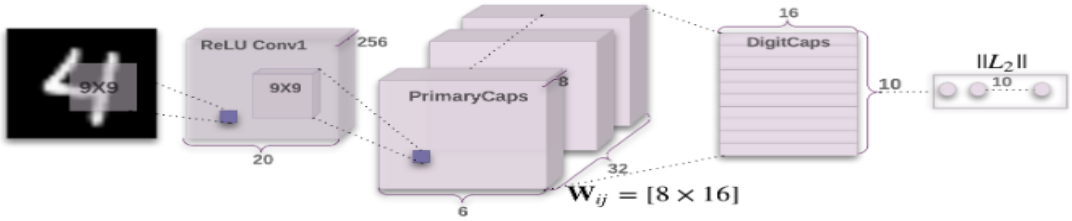

Figure 4.1: Architecture

The network uses three layered architecture. The length of each capsule's activity vector in the DigitCaps layer indicates the presence of each class's instance and is used to calculate the loss of classification.

The article introducing the architecture of CapsNet is 'Dynamic Routing Between Capsules'. A new type of neural network model for the classification of objects is given

in this article. The main advantage is that this model maintains hierarchical spatial relationships; this architecture can learn faster theoretically and use fewer samples per class. The article describes a CapsNet trainable end-to-end model and provides results of modeling. The classification error rate of 0.25% was reached on the MNIST dataset, 5.2% on MultiMNIST. Another recent study enforces the design of CapsNet and explores various effects of model variants, spanning from buffing more capsule layers to shifting hyper parameters. There was also a presentation of a new activation function. Due to computational constraints, they used fewer epochs and ended with an improvement in accuracy of 2.57 percent compared to the MNIST model introduced in the previous paper. In the third related study we found a new type of non-parametric convolutionary architecture.

5 PROPOSED METHODOLOGY

Initially, the literature review on Capsule Network, Dynamic Routing between Capsules, Fashion MNIST classification using capsule networking and our very own except with the title Caltech Bird classification using convolutional neural networks has been done.

The dataset is provided on the official website [5]. It is an image dataset consisting of photos of bird species which are mostly derived from the region of North America. It needed to be pre-processed before we feed it in our model.Mean and standard of the dataset was found for normalization. Images were also cropped to crop out the unuseful part like sky,grass and trees.

In Capsule Network model, images were resized to 128x128 , with the depth of 3(rgb). So, the input shape was (128,128,3).We used the keras implementation of neural network. We set all the layers of capsule network as given below.

The Encoder takes the image input and learns how to represent it as a 16-dimensional vector which contains all the information needed to essentially render the image.

- Conv Layer — Detects features that are later analyzed by the capsules. As proposed in the paper, contains 256 kernels of size 9x9x1.
- Primary(Lower) Capsule Layer — This layer is the lower level capsule layer which I described previously. It contains 32 different capsules and each capsule applies eighth 9x9x256 convolutional kernels to the output of the previous convolutional

layer and produces a 4D vector output.

- Higher Capsule Layer — This layer is the higher level capsule layer which the Primary Capsules would route to(using dynamic routing). This layer outputs 16D vectors that contain all the instantiation parameters required for rebuilding the object.

The decoder takes the 16D vector from the Digit Capsule and learns how to decode the instantiation parameters given into an image of the object it is detecting
The decoder is a really simple feed-forward neural net that is described below.Three Fully Connected (Dense) Layers are used for this purpose.The final output is 200 classes.
The classification was also done on Convolutional Neural Network. Same normalized dataset was used as in Capsule network and same regularisation techniques were used in CNN. The results of both are to be compared.

6 SOFTWARE REQUIREMENTS

- **Python3**
- **NumPy** (version 1.17.3) – pip3 install numpy
- **OpenCV** (cv2 library in Python) – pip3 install cv2
- **TensorFlow** (version 2.0) – pip3 install tensorflow
- **keras** (version 2.2.4) – pip3 install keras==2.2.4
- **matplotlib** (version 3.0.1) – pip3 install matplotlib
- **jupyter notebook** – pip3 install jupyter
- **Tensorflow-gpu** (version 1.15.0) –pip3 install –upgrade tensorflow-gpu

7 HARDWARE REQUIREMENTS

- Personal Computer at least 8GB RAM and core i5 preferable.(for testing the code on small set of dataset)
- HPC Cluster **"Surya"** NVIDIA TESLA V-100(16 GB) GPGPU

8 IMPLEMENTATION PLAN

Objective	Actions	Timeline
Studying the data	1.The dataset provided on the Caltech website is dowloaded and studied. It consists of images of 200 different bird species found across North America, with each one containing around 60 images shot in different environment and angle. Every image is provided with a class name and a unique id, and a bounding box, which is studied and an approach to work on it is chalked out.The data is cleaned i.e all the text files are converted into csv in order to easily fetch the information in coming steps.	Aug 22 - Sep 10 , 2019
About Capsule network	2. As the problem statement suggests to use Capsule Network to classify the data set, CAPSULE NETWORK and all its important subsisdiaries are studied. Various research papers involving this field of research and image classification are being studied, one of them being named 'Dynamic Routing Between Capsules' by Geoffrey Hinton. It is observed that CapsNet is an improvement over Convolutional Neural Networks. Thus the comparison between CNN and CapsNet is established and will be used in the implementation phase to verify the works done.	Sep , 2019
Preprocessing	3. The data includes a large number of images. The images are to be preprocessed before feeding them to the model to be built in future. It was observed that images are not larger that 500 x 500 in step 1, thus every image is resized to 128 x 128 in RGB format. All of this is done with the use of Python libraries like cv2(OpenCV) and numpy(for matrix operations).	Sep, 2019
Implementation	4. Python is the most suited programming language for the work. All the related libraries for modelling are installed. These include Tensorflow(used as backend), keras and matplotlib to plot the results. A three layered Capsule Network is created using Keras implementation and images are split in training and testing phase. The images, first, are split into 200 different labels(as lists in Python). Batch size is taken to be 32 and the model is to be run on 40 epochs.	Oct - Nov, 2019
Testing on a smaller dataset	5. As the dataset is huge and CapsNet is resource-hungry, it is observed that the systems being worked on are not fast enough to produce results on the whole dataset. SO, in order to get the idea, the dataset is reduced to 10% of the original size, which provided the picture. The result clearly meets the expectations and the CapsNet model outperforms the CNN. Snapshots of graphs are taken.	Nov, 2019
Testing on high performance computer	6. In order to run the whole data set, the High Performance system is used. After hours of training, the result is established.	Nov 16 - present, 2019

Figure 8.1: Implemetation Plan

9 RESULTS

9.1 Accuracy

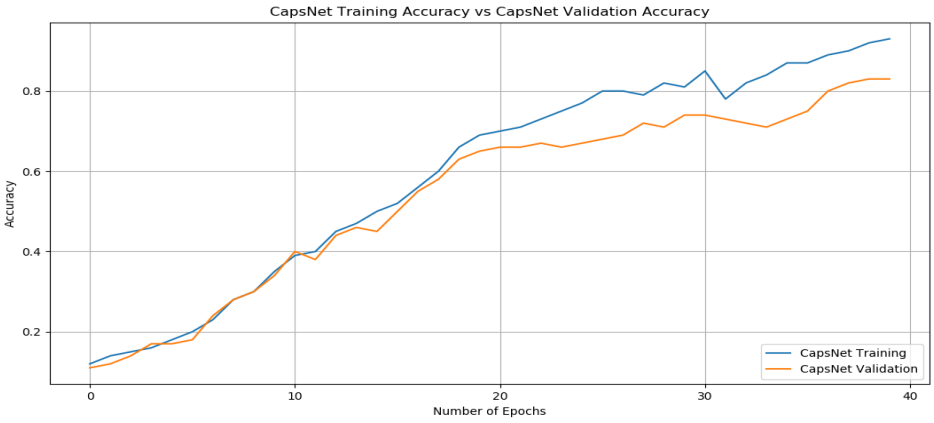

Figure 9.1: CapsNet Training Accuracy vs CapsNet Validation Accuracy

The CapsNet approach has the Training accuracy of 93.95% and the validation accuracy of 85% as shown in the plot above, which stretches itself upto 40 epochs.

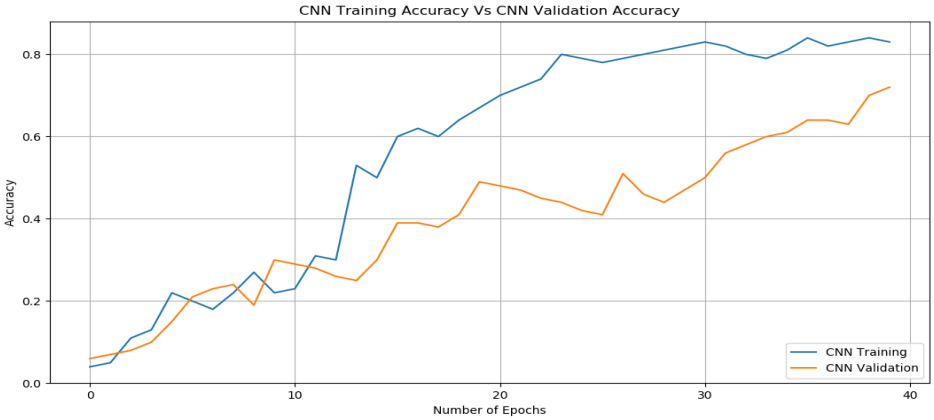

Figure 9.2: CNN Training Accuracy vs CNN Validation Accuracy

The same data set when run on the Convolutional Neural Network gives the training accuracy to be 83% and validation accuracy as 72.1%.

9.2 Comparison

The figure plots the training accuracy of CapsNet and Convolutional Neural Network in the same graph and depicts the difference between them.

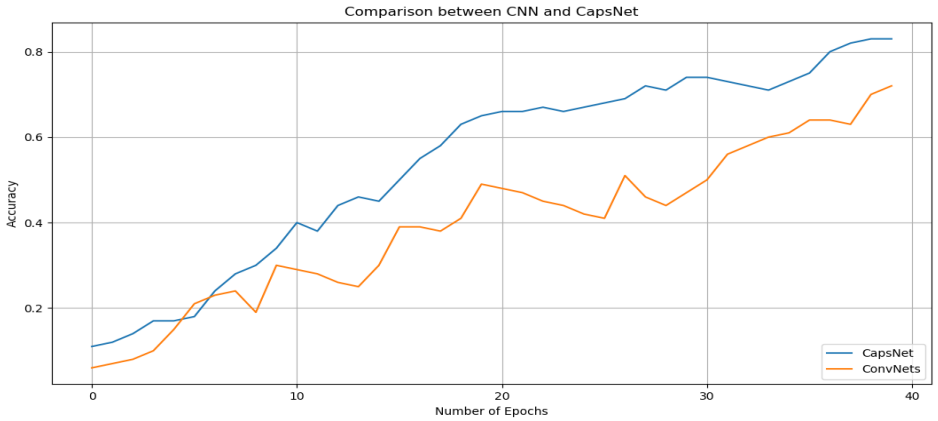

Figure 9.3: Comparison betweeen CNN and CapsNet

The batch size used here is 32.

10 CONCLUSION

CUB-200 has a total of 11788 images allocated over 200 (mostly North American) bird species. The large number of categories should make it an interesting dataset for subordinate categorization. Moreover, since it is annotated with bounding boxes, rough segmentations and attribute labels, it is also ideally suited for benchmarking systems where the users take an active parting the recognition process.

While working with the same dataset under the Capsule Network architecture and CNN architecture,the observation resembles the fact that the accuracy of Capsule Network is significantly higher than the traditional CNN. Also the training and validation loss have been lowered to a much greater extent.

11 REFERENCES

[1] **Max Jadenberg**, *"Spatial Transformer Networks"* : Feb, 2015 [Accessed: Sep, 2019]

[2] **Jonathan Krause**, *"The Unreasonable Effectiveness of Noisy Data for Fine-Grained Recognition"*: Nov, 2015 [Accessed: Sep, 2019]

[3] **Kevin Mader**, [Online] Available: https://www.kaggle.com/kmader/capsulenet-on-fashion-mnist [Accessed: Sep Oct Nov 2019]

[4] **Sara Sabour, Geoffrey E. Hinton**, *"Dynamic Routing Between Capsules"* : Nov, 2017 [Accessed: Sep, 2019]

[5] **"Caltech-UCSD Birds-200-2011"**, [Online] Available: http://www.vision.caltech.edu/visipedia/CUB-200-2011.html [Accessed: Sep Oct 2019]

[6] **Steve Branson, Grant Van Horn**, *"Bird Species Categorization Using Pose Normalized Deep Convolutional Nets"* : June 11, 2014 [Accessed: Sep, 2019]

[7] "Learn_Latex",Overleaf [Online] Available:https://www.overleaf.com/learn/latex/ [Accessed:Nov 20 2019]