**Image Classification using Vegetables**

**Ori Rinaman**

**Bellevue University**

**Business Problem**

Using vision analysis, I am proposing developing a machine learning algorithm to sort through pictures of various vegetables and classify them by their correct name.

**Background/History**

Image classification is the process of categorizing and labeling groups of pixels or vectors within an image based on specific rules. In this specific instance we will be categorizing pictures of different vegetables. Vision classification is starting to become an important part of the data science realm. Machine learning now has the capability to look at images and use predictive analytics to make predictions or classifications based on what is in those images. This project may not entirely sound exciting or interesting, but the concepts applied in this analysis can be used for many other applications.

**Data Explanation (Data Prep)**

For this analysis I will be using the Python coding language to accomplish my task. The data set being used for this project is not coming from a spread sheet, but from images. On Kaggle there was a file download for thousands of images of vegetables. The folder contained pictures of 15 different vegetables that include…

* Beans
* Bitter Gourd
* Bottle Gourd
* Brinjal
* Broccoli
* Cabbage
* Capsicum
* Carrot
* Cauliflower
* Cucumber
* Papaya
* Potato
* Pumpkin
* Radish
* Tomato

**Methods/Analysis**

For the analysis and modeling the Python programming language was used. The analysis starts with the data. I went through the data and divided up the pictures into training, testing, and validation data sets. Inside each of these folders, I created subfolders for each vegetable type. Once this step was completed, I uploaded the zip file to Jupyter Notebook. After the files were uploaded into the notebook it was time to start coding.

To start the code, I uploaded all of the packages needed for the analysis and modeling. Some of the more frequently known packages I used included matplotlib, seaborn, keras, numpy, and tensorflow. Next, I created a block of code to load in the files from the zip file. After brining in the images, I resized the images to make them all the same size and added the labels for each vegetable type. At this point, I was a bit concerned with the capability of my computer to handle all 15 types of vegetables. With some research I did find some good information that allowed me to understand the capability of my computer. I then found it wise to drastically decrease the number of vegetables in the data set. I combed through the pictures and decided on using only four vegetable types (carrot, broccoli, tomato, and pumpkin). By decreasing the number of images in my data set, it drastically sped up the time needed for aggregation and modeling.

At this point, my images have been aggregated and put into their proper grouping for modeling. I did run a quick check to make sure the images were set up correctly. An example of an image aggregated can be seen in Figure 1.

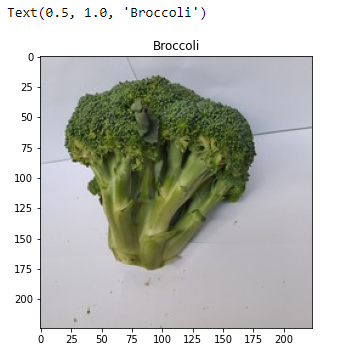


Figure 1. An aggregated picture of broccoli.

Everything to this point in the analysis was correct. Before modeling, I also reshaped the data by normalizing the array of pictures and also added some other data augmentation. Moving forward with the analysis, I started to build the Neural Network (NN). A NN is a system inspired by the biological neural networks that constitute animal brains. It is based on a collection of nodes called artificial neurons, which loosely model the neurons in a biological brain. The NN I used, contained 3 convolutional layers with a dropout layer added after the 3rd maxpool operation to prevent overfitting.

At this point everything was ready to go for modeling. For my modeling I decided to go with 200 epochs. An epoch means training the neural network with all the training data for one cycle. In my NN it will cycle 200 times. This many epochs required a lot of computing power. It took my computer over three hours to complete the training. The wait was worth it because we produced a model that had an accuracy of over 95% when classifying the four variable types. The training and validation accuracy and loss can be seen in Figures 2 and 3.

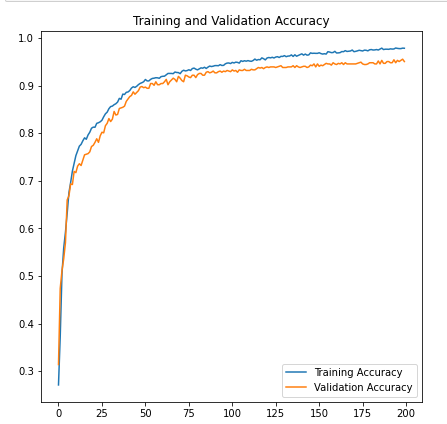
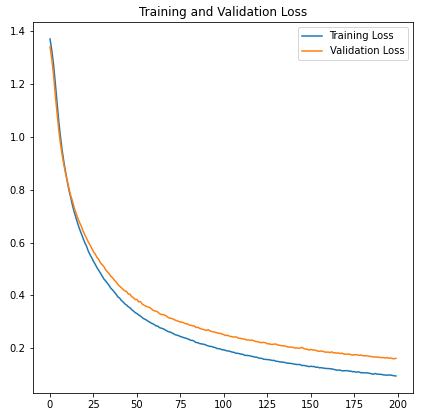
 

Figure 3. The training and validation loss for 200 epochs.

Figure 2. The training and validation accuracy results for 200 epochs.

**Conclusion**

Classifying images is one of the most difficult topics of data science. I was able to classify four vegetable types with approximately 95% accuracy. With the high accuracy I would claim this project to be a success! Now, classifying vegetables does not seem like the most exciting or beneficial project to work on, but the method used to get these results is the most important. This method could be used on MRI images to find diseases, used in CAT scans to find tumors, or even used to find criminals on the streets. The used cases of image classification are limitless and now that I successfully completed a project, the sky is the limit for what I can accomplish.

**Assumptions**

There was only one assumption made for this project, but it had little impact on the final results of the project. I was working off of my laptop for a majority of the project, and assumed that with the large number of pictures with the original data set, that my computer would not be able to handle that many images. Instead of risking my computer and program, I decided to decrease the number of images in the data set and only kept four vegetable types.

**Limitations**

As mentioned already numerous times, one of the main limitations with conducting image classification is the hardware used. My laptop took over two hours to run 200 epochs with the smaller data set. I doubt it would have been able to handle the original data set with 15 classes and a few thousand more images. In the future I would recommend using a more capable computer. This would not only speed up the process, but it would allow the flexibility to handle more images, more classes, and more epochs.

**Challenges**

There were numerous challenges I faced with this project, that were different two my previous two. The first and most obvious challenge was having to use a language that I am not used to using. For my first two projects, I used the R programming language, because I have years of experience using it. For this project I was forced to use Python due to the limited ability of R’s image classification capability. Not only did I have to use a language I am rather new in, but I had to learn methods on a topic that I was a novice at. Before starting this project, I had limited experience with image classification. I had to spend numerous hours researching methods and similar projects to produce good results. Though difficult, I learned many things that I had not known before and my skills as a data scientist have greatly improved since the start of this project.

**Future Uses/Additional Applications**

The topic of my project was classifying images of vegetables. The images used for the project are not as important as the methods. Image classification is used in many fields currently including medical imaging, object identification in satellite images, traffic control systems etc. At my current job, we are actually looking to implement an image system to detect defects in products as it moves down the line. The methods used here have shown me that though difficult, the final outcome can be very rewarding and successful.

**Recommendations**

If I had more resources, I would have liked to have access to a more capable computer system to handle a large amount of image data. I had to drastically reduce the data set size for my laptop to be able to handle the modeling of this project. In the future I would look towards getting a better, stronger computer to complete the task. A better computer would allow me to use many more images and many more classes. This would allow the training set to be larger, hopefully increasing the accuracy of the model. I would also recommend added more categories to the data set to challenge the system and maybe find weak spots in the modeling. The vegetables I chose were very different physically by shape and color. I would be curious to see how adding more vegetables that have similar shape and color would affect the model’s accuracy.

**Implementation Plan**

It would require many hours of work to get an image system up and running successfully. If I was going to implement this, I would like to start by laying out all of the steps needed to complete the project. Once the project was organized, I would decide what I am trying to classify. After I have my goal in place, I would create a plan to collect the images. Once the images are collected, I would divide them up into training and testing data sets. When I get a respectable number of images, I would begin training and testing of the model. After the model has been built, I would run with supervision to make sure it is actually working. When the model has been shown to run successfully and gains the trust of the organization I would deploy. Once it is up and running, I would make sure to check the model and certain intervals to make sure the accuracy is still respectable and add more images to the training and testing data for more training in the future.

**Ethical Assessment**

This project had few ethical risks attached to it, but as always it is important to display the results as is to the audience. Make sure to avoid all biases to the analysis and modeling. Introducing bias can not only affect the results, but could prove costly in the future.

**Citations**

Ahmed, M Israk. “Vegetable Image Dataset.” *Kaggle*, 24 Dec. 2021, https://www.kaggle.com/datasets/misrakahmed/vegetable-image-dataset.

Boesch, Gaudenz. “A Complete Guide to Image Classification in 2022.” *Viso.ai*, 17 Jan. 2022, https://viso.ai/computer-vision/image-classification/#:~:text=Image%20classification%20applications%20are%20used,%2C%20machine%20vision%2C%20and%20more.

“(PDF) DCNN-Based Vegetable Image Classification Using Transfer Learning: A Comparative Study.” *ResearchGate*, https://www.researchgate.net/publication/352846889\_DCNN-Based\_Vegetable\_Image\_Classification\_Using\_Transfer\_Learning\_A\_Comparative\_Study.

“Python: Image Classification Using Keras.” *GeeksforGeeks*, 13 May 2022, https://www.geeksforgeeks.org/python-image-classification-using-keras/.

**Appendix**

**Neural Network:** A collect of connected units or nodes called artificial neurons, which loosely model the neurons in a biological brain. They interpret sensory data through a kind of machine perception, labeling or clustering raw input. The patterns they recognize are numerical contained in vectors, into which all real-world data, be it images, sound, test or time series, must be translated.

**Epochs:** One cycle through the full training dataset.

**Loss:** Quantifies the difference between the expected outcome and the outcome produced by the machine learning model.

**Overfitting:** Occurs when the statistical model fits exactly against it training data. When this happens the algorithm can not perform accurately against unseen data.