Web App for Cassava Leaves' Diseases Detection

Anonymous Author(s)
Affiliation
Address
email

1 Introduction

Cassava is one of the most important types of food in many countries in Africa and the Caribbean. Cassava's tuber presents in many African dishes. Beside being cheap, affordable and suitable food for large-scale consumption, cassava is also a rich natural source with multiple nutritional properties. Moreover, Cassava is rich with vitamin A, C and dietary fibre content, they are effective for relieving fever, diarrhea, stomach aches, and also used in wound disinfection. Cassava leaves are also an excellent dietary supplement to reduce salt levels in the body.

There are more than 70 million people in Eastern, Central and Southern Africa depend on cassava as a primary source of food according to the Food and Agriculture Organization (FAO) [1]. But cassava, like many other crops, is vulnerable to viruses and other plant diseases. These diseases can affect cassava yields, cost farmers money, and threaten food security in sub-Saharan Africa. In order to protect the crops from poor production and post harvest losses, we developed a deep learning model to classify the Cassava leaf's disease among four common diseases and deployed the model on the web. What makes it doable and able to be improved over time is that agriculture experts usually have consent on the diagnosis of a specific plant.

2 Challenges

Only agriculture experts can identify a Cassava disease; as different diseases may have similar symptoms, and each leaf can have more than one disease. Furthermore, any model should generalize over different views and angles of images because they will be taken by different farmers with various camera types. Other challenges are different backgrounds in the images (i.e. different farms), image's quality, the time the image was captured[5].

3 Dataset Description

The trained model used in this web application is from the Cassava Disease Classification Competition hosted by Kaggle [3]. The data set consists [5] of 12,595 unlabelled images and 9,436 labelled images of cassava leaves. It is annotated in five categories: **(a)** healthy leaf (316/211 examples of trains/tests) and; the four categories of the diseased leaves are: **(b)** Cassava Mosaic Disease (CMD) (2658/1773 train/test images), **(c)** Cassava Brown Streak Disease (CBSD) (1443/963 train/test images), **(d)** Cassava Bacterial Blight (CBB) (466/311 train/test images), **(e)** and Cassava Green Mite (CGM) (773/516 train/test images).

The data collection was done in different farms in Uganda as part of a crowd-sourcing project by the Artificial Intelligence Laboratory at Makerere University and the National Crops Resources Research Institute (NaCRRI) using smart phones [5] (Note that NaCRRI is the Ugandan government agency responsible for agricultural research in the country [5]). All images collected were manually labelled by NaCRRI experts who assessed the impact and severity of each image. As a result, only disease class annotations (without severity) were used. All the pictures were taken by the same type of phone and since the pictures were taken from different farms so they have different characteristic so there are some data cleaning have applied to it.

3rd Workshop - Black in AI (BAI 2019), Vancouver, Canada.

(a) Healthy (b) CBB (c) CGM (d) CMD (e) CBSD

Figure 1: The five cassava classes include the healthy cassava leaves and the four common diseases. Source [5]

4 Deep Learning Model and Web Application Description

We used the advancement of deep learning in image classification task. This paper aims to highlight the practical usage of successful models in image classification. We experimented with Fastai [2], a high-level deep learning library built on top of Pytorch. The reason for using Fastai is quoted from their documentation: "Fastai simplifies training fast and accurate neural nets using modern best practices".

We trained a simple Convolution Neural Network (CNN) architecture VGG-16 with the following transformation: max. rotate of 180 degree, max. zoom with ratio of 1.1, max. lighting of 0.2, max. warp of 0.2, probability each affine transform and symmetric warp is applied (p_affine) of 0.75, probability that each lighting transform is applied (p_lighting) of 0.7, and zoom crop with scale between 0.5 and 1.5. All transformations are detailed in Fastai documentation.

The training goes in cycles of enhancements. First we used the pre-trained network with ImageNet dataset (i.e. we used transfer learning), training only the last two fully connected layers for 15 epochs and freezing the first few layers from the pre-trained network, then unfreeze the whole network and re-train for 25 epochs, after that we try to find a suitable range of learning rates for all layers via learning rate finder in Fastai and train for another 15 epochs. The idea of learning rate finder is to train the model for few iterations starting with a low learning rate, changing it at each mini-batch until it reaches a high learning rate, recording the loss at each iteration and plotting those losses against the learning rate to find the optimal value before it diverges[1].

We applied another trick to enhance the accuracy of the classification which is progressive resizing; starting with 224*224 size and batch size of 64, then 256*256 with batch size 32 and finally 512*512 with batch size 8. And the model was trained using free GPU on Google Colab. The web application is hosted by Render. With few steps to deploy the Fastai model on Render using a published tutorial by Fastai, we were able to make a public website. We believe that Kaggle is a good benchmark to test our model. We measured the accuracy after the competition has ended and it recorded a score of 0.93021 on the private leader board and 0.91655 on the public leader board which comes in the top five teams.

5 Limitations and Future Work

Pictures were taken with the same type of phone which can constitute a limitation in terms of the model robustness. Another limitation is the computation resources; despite that we experimented with Google Colab, we needed more computational capacity to try different experiments especially after using the unlabelled extra images in the training set. We plan to do enhancements so that it can be used by the farmers themselves, such as:

- Add to the interface the African dialects where the Cassava is planted.
- Display automated feedback for the predicted class (i.e. explain the prediction with one of the deep learning interpretablility methods).
- Extend to a mobile application.
- Use the extra unlabelled images to enhance the model with semi-supervised techniques such as: pseudo-labeling[4] and a trained decoder in an unsupervised settings (e.g. auto-encoder).

[1] https://docs.fast.ai/callbacks.lr_finder.html

References

[1] FAO. Cassava diseases in africa: a major threat to food security, September 2009. URL http://www.fao.org/emergencies/resources/documents/resources-detail/en/c/171103/.

[2] J. Howard. Fastai: Deep learning high-level library, January 2018. URL https://docs.fast.ai/.

[3] Kaggle. Cassava disease classification competition, June 2019. URL https://www.kaggle.com/c/cassava-disease.

[4] D.-H. Lee. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on Challenges in Representation Learning, ICML*, volume 3, page 2, 2013.

[5] E. Mwebaze, T. Gebru, and al. icassava 2019 fine-grained visual categorization challenge, 2019. URL https://drive.google.com/file/d/1GW0Ak_fS0ZMXcy89B7di1xNF1MBIga_4/view.