Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2017.DOI

Classification of Alzheimer's Disease in MRI Images using Machine Learning

BAILIE C. DELPIRE[1]

[1]Department of Computer Science, Middle Tennessee State University, Murfreesboro, TN 37132 USA (e-mail: bcd3q@mtmail.mtsu.edu)

ABSTRACT Alzheimer's disease (AD) is a progressive neurodegenerative disorder that has no cure and is currently the seventh leading cause of death in the world. Early detection of Alzheimer's is important for slowing its progress and for studying the disease. MRI scans can be used to detect early signs of Alzheimer's and machine learning models have been shown effective at classifying these images. The more data that can be used for training, and the higher resolution the images, the more accurate machine learning techniques can be. This is why using cloud technologies and analytics engines such as Apache Spark can be useful for creating more accurate models through large-scale data processing.

INDEX TERMS Alzheimer's Disease, Apache Spark, Machine Learning, MRI

I. INTRODUCTION

ALZHEIMER'S disease (AD) is a progressive neurodegenerative disorder that affects memory and thinking skills. It is the most common cause of dementia among older adults and is the seventh leading cause of death globally. Symptoms of Alzheimer's typically appear in later years and worsen with time. Symptoms include memory problems, affected speech and reasoning, and confusion. Those in the severe stages of Alzheimer's cannot communicate or perform simple tasks and are completely dependent on the care of others.

Alzheimer's may begin affecting the brain a decade or more before symptoms first appear. The abnormal buildup of proteins in the brain causes previously healthy neurons to stop functioning, lose connection, and die. This begins in the hippocampus and entorhinal cortex and progresses until brain tissue is widely damaged and has shrunk significantly. There is no cure for Alzheimer's, but current treatments can slow its progress.

Magnetic resonance imaging (MRI) is used to support an Alzheimer's diagnosis and to rule out other possible causes for the symptoms. MRI images can also be used to assess the stages of Alzheimer's disease. Early detection is important so that treatment can begin as soon as possible, which can help to maintain a person's daily function for some time. For this reason, the use of MRI images to detect the early stages of Alzheimer's is researched.

Machine Learning techniques have proven to be effective at identifying between AD-diagnosed patient MRIs and controls. In 2018, Ullah et al [2] used a deep convolutional neural network on 3D MRI data to identify AD-diagnosed scans with an 80% accuracy. In a 2021 study by Uma et al [3], feature extraction was performed on MRI scans using a Gray Level Cooccurrence Matrix and Haralick features, and a Support Vector Machine used these features to classify between AD and control scans with an 84% accuracy. Also in 2021, Chaihtra et al [1] used a DenseNet121 deep learning model to detect AD in MRI scans with 91% accuracy.

II. PROJECT OVERVIEW

A. GOAL

The goal of this project is to compare the efficiency and accuracy of machine learning classification models, locally with Apache Spark and in the cloud with Google Cloud Platform (GCP). The problem for the model to solve is classifying MRI images as demented or non-demented. Future work may expand this to classifying MRI images between the different stages (non-demented, very mildly demented, mildly demented, and moderately demented) of Alzheimer's Disease.

B. DATA

The dataset is obtained from Kaggle. It contains 6,400 images, with 3,200 images labeled as demented, 2,240 labeled as very mildly demented, 896 labeled as mildly demented, and 64 labeled as moderately demented. Each image is a 176 by 208-pixel, 1-channel image. The decision to do binary classification on this dataset stems from the unbalanced nature of the dataset. If split into two classes instead of four, the dataset is balanced but remains large. If we expand the project to a multi-class classification problem, we will have to balance the dataset by removing some of the images, in order to reduce the bias on non-demented labeled images and very mildly demented labeled images.

C. IMAGE PROCESSING

Image processing is the task of turning images into suitable input data. Images are made up of pixels that are represented in a jpeg or png file as raw hexadecimal byte data. This raw data cannot be used directly, therefore we need to load images as pixel data instead by converting the hexadecimal values to decimal values. For 1-channel or gray-scale images, each pixel is represented as one byte, or 8 bits. This means that a pixel can be represented as an integer between 0 and 255, with 0 being completely black and 255 being completely white. These pixel integers can be stored in a structure, such as an array, to be used as input data for machine learning models.

D. APACHE SPARK

Apache Spark is an analytics engine for big data processing. In this work, we will use python, pyspark, and MLlib to load the MRI images, pre-process the images, and train a machine learning model to classify the images. First, the images are loaded in from their respective folders using the image format.

```
dataframe = spark.read.format("image")
   .load([filepath])
```

Then labels are added to the dataframe based on whether they were loaded from the dementia-positive or dementia-negative folders.

```
dataframe = dataframe.withColumn("label",
    lit(0))
```

After combining the dataframes into one dataframe, we use a user-defined function to transform the byte data into pixel data.

```
img2vec = F.udf(lambda x:
    DenseVector(ImageSchema.toNDArray(x)
   .flatten()), VectorUDT())
```

Now the data can be used for training any SparkML model.

E. TENSORFLOW

Tensorflow is a software library for machine learning. It includes its own preprocessing tools. We can directly load the image dataset from a directory that contains folders for each class that will be automatically recognized.

```
dataset = tf.keras.utils
   .image_dataset_from_directory([args])
```

To normalize the data, we can use a Keras Rescaling layer.

```
tf.keras.layers.Rescaling(1./255)
```

This will be included as the first part of any TensorFlow model that is built.

F. GOOGLE CLOUD PLATFORM

GCP's Deep Learning VM is a solution that allows us to use a virtual machine (VM) image containing popular AI frameworks on a Google Compute Engine instance. With this, we can use python and TensorFlow for machine learning because they are part of the pre-packaged image. We can also make use of a number of CPUs and GPUs maintained by Google. With GCP's Deep Learning VM, we can build the same Keras models mentioned above.

G. DATAPROC

Dataproc is Google's cloud service for running Apache Spark and Hadoop clusters. With Dataproc, we can build the same Apache Spark models mentioned above.

H. OVERVIEW

With both Apache Spark and GCP we will test the models with cross-fold validation to get average model accuracies that we can compare. We will also use the same data split, hyperparameters, and number of training epochs so that we can do a fair comparison of the training, validation, and testing times that were taken in each environment. This will yeild the final results for our experiments. A diagram of the project pipeline can be seen in Figure 1.

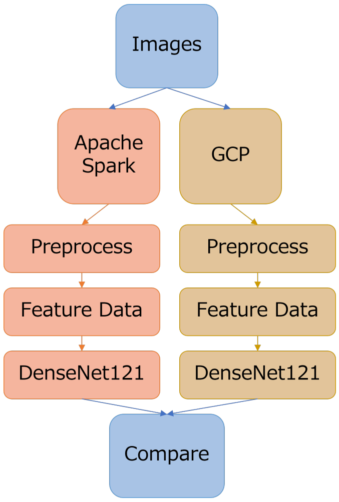

FIGURE 1. Project Overview: Apache Spark and GCP Pipelines

III. PRELIMINARY RESULTS

A. SPARK MLLIB SUPPORT VECTOR MACHINE

With a 70-30% train-test split, a max iteration of 10, and a regularization parameter of 0.1, our pyspark.ml.classification.LinearSVC support vector machine (SVM) model had a testing error of 0.225707 and testing accuracy of 77%. In future work, we will do hyperparameter tuning in an attempt to reduce error. We will also increase the training split.

B. TENSORFLOW CONVOLUTION NEURAL NETWORK

With an 80-20% train-test split, a batch size of 32, the adam optimizer, and the below model architecture, our TensorFlow Convolution Neural Network (CNN) had a model validation accuracy of 94.84% after 10 epochs of training.

```
model = tf.keras.Sequential([
   tf.keras.layers.Rescaling(1./255),
   tf.keras.layers.Conv2D(32, 3,
       activation='relu'),
   tf.keras.layers.MaxPooling2D(),
   tf.keras.layers.Conv2D(32, 3,
       activation='relu'),
   tf.keras.layers.MaxPooling2D(),
   tf.keras.layers.Conv2D(32, 3,
       activation='relu'),
   tf.keras.layers.MaxPooling2D(),
   tf.keras.layers.Flatten(),
   tf.keras.layers.Dense(128,
       activation='relu'),
   tf.keras.layers.Dense(2)
])
```

In future work, we will do more hyperparameter tuning in an attempt to reduce error.

IV. FUTURE WORK

In addition to the future work mentioned above, we will also compare the use of GCP's DataProc for our Spark MLlib SVM model and GCP's Deep Learning VM for our TensorFlow CNN model.

REFERENCES

[1] D. Chaihtra and S. Vijaya Shetty, "Alzheimer's Disease Detection from Brain MRI Data using Deep Learning Techniques," 2021 2nd Global Conference for Advancement in Technology (GCAT), Bangalore, India, 2021, pp. 1-5, doi: 10.1109/GCAT52182.2021.9587756.

[2] H. M. T. Ullah, Z. Onik, R. Islam and D. Nandi, "Alzheimer's Disease and Dementia Detection from 3D Brain MRI Data Using Deep Convolutional Neural Networks," 2018 3rd International Conference for Convergence in Technology (I2CT), Pune, India, 2018, pp. 1-3, doi: 10.1109/I2CT.2018.8529808.

[3] Uma R. K, Sharvari S. S, Umesh M. G and Vinay B. C, "Binary Classification of Alzheimer's disease using MRI images and Support Vector Machine," 2021 IEEE Mysore Sub Section International Conference (MysuruCon), Hassan, India, 2021, pp. 423-426, doi: 10.1109/MysuruCon52639.2021.9641661.