Machine Learning Classifier for American Sign Language Alphabet and COVID-19 Symptom Gestures

Terry Leo Wambolt
Computer Science Department
California Polytechnic State University
San Luis Obispo, CA, U.S.A
twambolt@calpoly.edu

Abstract—This project describes the development of creating a machine learning classifier for the American Sign Language Alphabet and COVID-19 symptom gestures. The model was built using Google's TensorFlow library, and Keras on the Google Colab platform. Datasets came from Kaggle and the other was self-made with the help of data augmentation techniques. The result was two models with over 90% validation scores based on the data used to train them, as well as scores around 50% for user submitted images.

Keywords—Machine Learning, TensorFlow, American Sign Language, COVID-19.

I. Introduction

In the midst of the global pandemic brought about by the spread of COVID-19, health experts and governments have taken measures to "flatten the curb". These officials have urged the public to follow social distancing guidelines, implemented shelter in place orders, and required the use of face masks in certain public establishments. These precautions have changed the way people interact with one another, making communication more difficult and arguably riskier. Though these actions will be beneficial in the long term, these can make life more difficult for individuals who already have a more challenging time navigating the world as it is.

Healthcare services are critical during these times especially. However, they have become battlegrounds for dealing with the virus as the volume of people needing treatment has surged dramatically. The previously mentioned precautions pertain to hospitals and other healthcare institutions. In these locations, staff now more than ever are always wearing facemasks in order to prevent themselves from becoming contaminated. Additionally, some institutions have completely cancelled in person meetings in order to follow social distancing guidelines and stay at home orders. This poses a problem for individuals who are deaf or hard of hearing in getting the help they need and communicating effectively with a person without knowledge of sign language.

Making healthcare and other public services available to everyone is highly important, especially those with disabilities that rely on them more. During times of crisis these services not only become more important they also become more difficult to maneuver. More people become reliant on these services all at once and there is less time to address everyone's specific needs. Helping disabled people to access the resources they need independently is necessary as not everyone has someone they can ask for help. It also empowers these individuals as they have the ability to be able to be self-reliant and independent.

I propose to use machine learning principles to train a classifier which healthcare providers would theoretically be able to use in order to help people from this category. The system would be able to recognize signed letters and also common words, phrases and conditions in American Sign Language (ASL) that are related to the symptoms for COVID-19. This system would ideally be used by hospital staff including nurses, doctors and receptionists. With the increased number of doctors visits happening online as well, these models could also be used within telecommunication platforms.

II. Background

A. Design Overview

For this project, it was decided to use Google's machine learning library, TensorFlow. The reasons being that there was an abundance of documentation and tutorials for getting started building a convolutional neural network to classify images. Within TensorFlow, the Keras library was also chosen due to the fact that it was relatively simple to use, and it also had a variety of examples and tutorials [1][2].

Development for this project was primarily done on Google's Colab platform. This is due to the fact that the computing resources needed to train machine learning models is extensive. By using Colab, I was able to select a GPU runtime that made training the models substantially faster than if done on my machine or without the use of a GPU. The Colab notebook for this project is linked here.

B. Dataset Information

When it comes to machine learning projects, a lot of data is necessary in order to accurately train a model. In order for my project to succeed, I required datasets of hand gestures signing in American Sign Language (ASL) as well as gestures for COVID-19 related symptoms.

In acquiring the appropriate ASL alphabet dataset to train my model I first looked at projects which had accomplished this before and tried and use the same data they used to train their model. The first dataset I used was found on GitHub and included about 86,000 different images for all of the letters of the alphabet and numbers 0-9 [3]. This dataset was comprised of hand gestures which had undergone image augmentation to only focus on the hand. It did this by making all of the background pixels black and relevant hand pixels a grey and white gradient as shown in Fig. 1. Ultimately this dataset was scrapped from my project as I couldn't replicate the image augmentation methods used in my images.

In looking for a more appropriate ASL alphabet dataset to train my model on I next came across the Sign Language MNIST database on Kaggle [4]. This dataset was comprised of two csv files, one used of training and the other for testing. The training dataset is comprised of 27,455 rows, each with 784 columns, while the testing dataset was comprised of 7,172 rows and 784 columns. Each row represents a different color image of only the hand and minimal background objects present. The first column in the row serves as the label and the following columns contain the pixel values for that image. There were 24 unique labels present in the dataset as images for "j" and "z" were not included since those signs have a movement component.

For the gesture data related to Covid-19 symptoms, there was not an easily accessible dataset to be imported. I decided to create a dataset from videos of people performing the gestures related to these symptoms. The videos were found on spread the sign in the "Health and Medicine" category [5]. I searched for each symptom and took screenshots of key and unique moments for the particular sign within the video played. A majority of these videos had sign data for additional languages other than English which were similar. I also took screenshots of those videos at points where the unique moment was shared with the English version. This allowed me to create a more diverse set of data since the people were of different backgrounds and it also allowed me to acquire more data. For the 11 different symptoms to be identified in this model, I captured around 20 different screenshots for each gesture. I then proceeded to crop and organize the images into different folders for later processing. Fig. 2 shows an example of the gesture screenshots used in this data set.

At this point I did not have the necessary amount of data for the gesture model, so I implored some image augmentation techniques to create data from my data by introducing variations such as zooming, rotation, tilting and height and width shifting. [6]. These transformations were applied randomly to each of the images in my dataset and were then saved under the appropriate label. In doing so I was able to expand my dataset to contain over 2700 images. The training and testing datasets were then created by randomly partitioning 25% of the entire dataset and using that as the validation set.

For further testing and validation of my models, I included datasets of me performing each of the 24 different ASL alphabet letters as well as the 11 different signs for Covid-19 symptoms.

Figure 1: Image of preliminary ASL dataset

Figure 2: Person signing "cough" gesture

III. Project Discussion

In developing this model, I first followed the tutorials for using TensorFlow to classify images created by Google and other sources. I then built my model in a manner similar to theirs while adding a few other layers mine would be using as well as changing some of the parameters based on performance. Both the models are similar in that they are deep convolutional neural networks that have three intermediate layers that are comprised of a convolutional layer, a max pooling layer, and a dropout layer. As per the tutorials the convolutional and max pooling layers are helpful in finding features within the image. The dropout layer was included in order to account for some of the overfitting I noticed was happening as I would have high validation scores early on in the training and very low validation scores for the data that I provided.

A. ASL Model

For the ASL alphabet model, the images were reshaped to 28 x 28 greyscale images which would then be fed into the model. This was the maximum size the image could be reshaped into in order to be fed into the model based on the fact that each image was made up of 784 pixels.

The model was then trained for 20 epochs in batches of 128 images. When validated against the test dataset, it would produce results ~95% accuracy. This however wasn't as accurate when compared to using the model to predict the images I personally took. The model would score ~45% accuracy for the entire set of my images. Fig. 3 gives an example of some of the results I obtained. This discrepancy could be due to some of the training images having background noise to them, as well as inaccuracies in the weights applied within the model.

Figure 3: Sample prediction results for images I provided

B. Covid-19 Symptom Gesture Model

The Covid-19 gesture images were reshaped to be 40 x 40 greyscale images. Images of this size produced the best validation results across multiple different trials.

The model was trained over the course of 40 epochs, using the entire dataset to train each time, and testing on the previously separated validation set. This model would produce validation results ~93-95%, however it is important to reiterate that overfitting could have very well been a factor. When used to predict the images I took, it scored ~40% on average which I found to be a particularly good score given the dataset used. Fig. 4 shows an example of these results.

Figure 4 Sample prediction results for images I provided

C. Outcomes

Overall, the models were somewhat accurate in their predictions of image data that I personally submitted. For the ASL alphabet gestures, it was previously mentioned that discrepancies could be due to background noise and the model in general. Moving forward I think the model would have been better if the images were larger as fingers are quite small and for hand signs, their position matters. Therefore, there can be very little variation between different signs (i.e. “a” vs “e”).

The Covid-19 symptom model worked fairly well given the dataset that was available. It would have been nice to be able to have a better dataset with much clearer images and not have to rely on data augmentation too much however it produced reasonable results given the circumstances. Ideally video input would have been used to train the data, however I could not find a dataset containing the amount of data necessary to train the model. It is worth mentioning that the people in the training images were all wearing black long sleeve shirts, whereas in my images I was wearing a short sleeve black shirt which could have been the cause of some of the discrepancies.

IV. Conclusion

Future work for this project would include expanding the dataset for the Covid-19 gestures to learn based on video data from a variety of individuals. For both models, further refinement could be made in the form of more layers and different weights. Being my first time working with machine learning, my development was far from perfect and was the result of a substantial amount of trial and error. Additionally, these models could then be serialized and placed into an application which uses them to make the desired predictions in the hospital setting outlined in the introduction.

All and all this project was a good way for me to get a glimpse of the possibilities of machine learning. I think it would have been nice to have taken a more formal course in the subject before approaching this project in order to have a better sense of what tools and approaches I could have taken to improve my model. Nonetheless it was very satisfying to see this work out and I look forward to doing more independent study in this field.

References

[1] TensorFlow. ”Machine Learning Zero to Hero (Google I/O’19),” *Youtube,* May 09, 2019 Available: https://www.youtube.com/watch?v=VwVg9jCtqaU [Accessed: 30-Sep-2020].

[2] Sentdex. ”Deep Learning with Python, TensorFlow, and Keras tutorial,” *Youtube,* Aug 11, 2018 Available: https://www.youtube.com/watch?v=wQ8BIBpya2k [Accessed: 30-Sep-2020].

[3] EvilPort2, “EvilPort2/Sign-Language,” *GitHub*, 03-Apr-2019. [Online]. Available: https://github.com/EvilPort2/SignLanguage/tree/master/gestures [Accessed: 30-Sep-2020].

[4] Tecperson, “Sign Language MNIST,” *Kaggle*, 20-Oct-2017. [Online]. Available: https://www.kaggle.com/datamunge/sign-language-mnist.
[Accessed: 30-Nov-2020].

[5] T. Lydell, “Category: Health & Medicine,” *SpreadTheSign*, 2018. [Online]. Available: https://www.spreadthesign.com/en.us/search/by-category/194/health-medicine/?q=. [Accessed: 30-Nov-2020].

[6] F. Chollet, “The Keras Blog,” *The Keras Blog ATOM*, 05-Jun-2016. [Online]. Available: https://blog.keras.io/building-powerful-image-classification-models-using-very-little-data.html. [Accessed: 30-Nov-2020].