Project “Sore Vision”

A deep learning based approach to identifying and classifying mouth sores

Rajaram Anantharaman
School of Computing and Engineering
University of Missouri, Kansas City
Rajaram.Anantharaman@mail.umkc.edu

***Abstract*—** The goal of "**Sore Vision**" is to provide an easy way for people to perform self-examinations of mouth sores using a mobile responsive web app. One variety of mouth sore, referred to as the "cold sore" is highly contagious and an infected person can easily pass on the infection to another person just through skin to skin contact. "**Sore Vision**" is implemented as an HTML5 mobile responsive web app that can be accessed through any mobile or standard browser. Sore Vision uses deep learning to train a model and subsequently use the trained model to distinguish a cold sore from a canker sore.

Keywords— deep learning; retrained inception; clarifai; mouth sore;

I. INTRODUCTION

Although mouth sores are a common occurrence in people of all ages including kids in many parts of the world, many people do not clearly understand the risks associated with their condition. While a visit to a dentist or a doctor can help identify the condition accurately, at times this might not be feasible or accessible. The motivation for "Sore vision" is to put a simple tool in the hands of every person to help identify his/her mouth sore accurately so that they can take the necessary precautions to ensure that they are not passing on their infection.

Most people do not realize that mouth sores come in two varieties. One variety of mouth sore, referred to as the "cold sore" is highly contagious while the other variety of mouth sore, referred to as a "canker sore" is luckily not contagious. [Figure-1]

Making it easy for users to access the application is a challenge. Our first approach was to make it a downloadable mobile application. Then it was determined that a mobile responsive web application will better serve the needs of the users. In our project, we are utilizing standard web front end technologies in combination with trained neural network models to classify the input images into the two classes. The goal is to develop models using shallow learning and deep learning technologies and compare the results.

As far as appearance which is important for classification situations such as the one in this project, the two classes have clearly identifiable features that can potentially make it easier to train a classifier. [Figure-2]

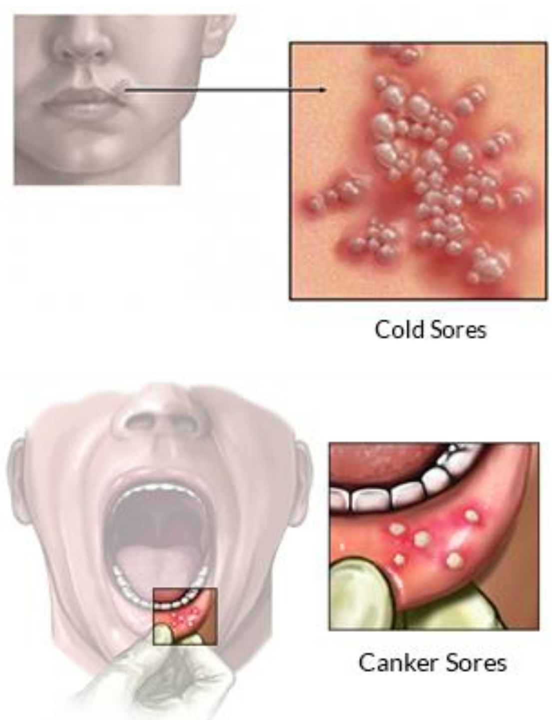

Figure-1 Cold Sore versus Canker Sore.

Canker Sore	Cold Sore
Typically, small open sores with white, yellow or gray center and bright red border. Can be single or in groups.	Red, fluid-filled blisters, typically in groups.

Table-1 Appearance of Canker Sore and Cold Sore

II. RELATED WORK

i-Nside: A company called i-Nside is using Clarifai API to provide an application that enables doctors to effectively diagnose and treat ear problems. Being a endoscope manufacturer, they already had a library of over 100,000 images that they could readily use to build and train their own custom visual recognition model in Clarifai. Clarifai's visual recognition technology powers the software layer in i-Nside's line of endoscopic hardware, enabling the tool to not only take pictures of the ear but also to analyze the results.

While Clarifai's core model can recognize over 11,000 general concepts, ear diseases are not among those core tags. i-Nside had to build a special custom model built for the sole purpose of analyzing ear patterns.

III. PROPOSED WORK

A. Features

The primary features of "Sore Vision" include:

A model that is trained adequately to identify and classify mouth sores. Three separate models will be designed and developed. The First model based on Clarifai API, second based on Spark Machine learning and the third based on Deep learning using Google Tensorflow.

A mobile responsive web application that provides the user with the ability to upload an image and receive feedback right away through the web interface.

B. Use Case:

The person simply takes a picture and presents it to Sore Vision.

Sore Vision sends the picture to the model which will use artificial intelligence to identify, classify and provide immediate feedback with a certain degree of confidence. [Figure-2]

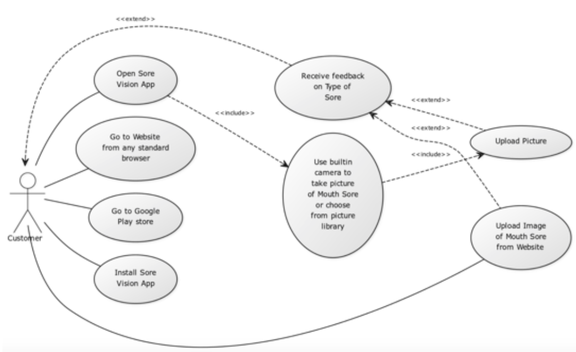

Figure-2 Use Case of the "Sore Vision" app.

C. Sequence Diagram:

The application three major components, the Web Server, the App Server and the Machine Learning Interface. The flow of information is depicted in the Sequence Diagram. [Figure-3]

D. Training Data Source:

Data for training the model will be mainly sourced from http://images.google.com. The goal is to find at least 75 images that provide close shots of cold sores and canker sores. A human dentist will classify the images into the right classes to begin with. Other than this, there is not much of analytical tasks that needs to be performed on the dataset.

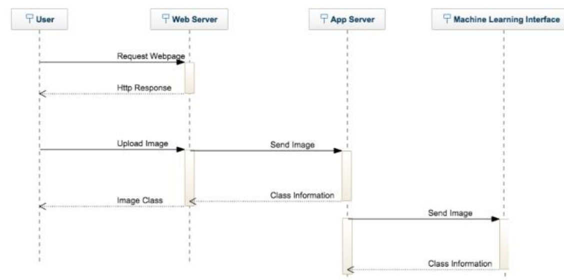

Figure-3 Sequence diagram of the "Sore Vision" app.

E. Algorithms:

The project will be implemented in three distinct phases, each phase focusing on creating a specific training model.

In iteration 1, we will use Clarifai's visual recognition algorithm. Like the i-Ncase situation, oral diseases are not among the core tags for Clarifai. So, we will need to build a special custom model built for the sole purpose of analyzing mouth sore patterns.

In iteration 2, we will use the Random Forest Classifier algorithm to create a model, and use this model to classify images into the two classes.

In iteration 3, we will use the Inception model to classify images into the two classes. However, the Inception model has millions of parameters and can take weeks to fully train. We will utilize a technique called "Transfer learning" to train our model quicker. "Transfer learning" is a technique that takes a fully-trained model for a set of categories like ImageNet, and retrains from the existing weights for new classes. In this iteration, we'll be retraining the final layer from scratch, while leaving all the others untouched.

IV. IMPLEMENTATION AND EVALUATION

A. User Interface

The front end for the application is setup as a mobile responsive website developed using Bootstrap. Bootstrap is one of the most popular HTML, CSS, and JavaScript framework for developing responsive, mobile-first web sites. The binary stream of the image uploaded through the User Interface is converted into its text representation using Base64 encoding and is presented as data to the REST service waiting to receive the image. For security reasons, browsers restrict cross-origin HTTP requests initiated from within scripts. The Cross-Origin Resource Sharing (CORS) mechanism gives web servers cross-domain access controls, which enable secure cross-domain data transfers. The user interface as accessed from a mobile phone browser is shown in Figure 4. There are three versions of the user interface accessing the three different REST endpoints. The REST endpoints will be discussed in the next section.

Figure-4 User interface as seen on a mobile phone.

B. *Web Server:*

The web server has been implemented in two flavors. The first is a simple Java based HTTP server which accepts POST request containing data, which happens to be the textual representation of the binary image. The second flavor is a python based HTTP Server which essentially is performing a similar function as the Java server. The sequence of operations is as below:

1) getRequestBody returns a InputStream for reading the request body.
2) Decode the BASE64 encoded image data.
3) Save the decoded data as a “png” file on the server.
4) Send the png file to the App server for processing
5) Receive the Response from the app server
6) Set the Response headers
7) Send Response stream back to the user interface.

C. *App Server:*

There are two versions of the app server. The first is an app server that sends and reads from Clarifai. The sequence of operations is as below:

1) Connect to Clarifai with Client ID and Client Secret.
2) Get a Token
3) Send the image provided by the web server to the custom model called “Mouth-Sore”
4) Get list of predictions
5) Return the Top 1 predicted class and the prediction confidence back to the web server.

The second version of the app server sends and receives class predictions from a retrained inception model. The sequence of operations is as below.

1) Read in the image data from the web server
2) Loads label file and strip off carriage return
3) Unpersist graph from file
4) Feed the image data as input to the graph and get Top 1 prediction
5) Retrieve class and confidence score from the prediction.

D. *Clarifai Custom Model:*

Custom Training allows us to quickly and easily "teach" Clarifai visual recognition technology to understand any new concept. We created a model with two concepts matching up with the two classes we care about. Then we uploaded 75 images directly sourced from google images. Then we tagged images with the concept that matches the ground truth. Training was done over several iterations manually by tagging the images as shown in the snapshot of Clarifai Application interface. [Figure-5]

E. *Retrained Inception Model:*

we used the Inception model to classify images into the two classes. However, as the Inception model has millions of parameters and can take weeks to fully train, we did not fully retrain the entire model from scratch. Instead, we retrained the final layer from scratch, while leaving all the others untouched. Although this approach is not as good as a full training run, this is quite effective for our application, and we could run in as little as fifteen minutes on a laptop, without requiring a GPU. We utilized the same exact images that we used to train the Clarifai model. This was done so that we can compare the performance of the two models effectively. It goes without saying that none of the sore images were in the original ImageNet classes the full network was trained on.

F. *Results:*

The Test data set consisted of 6 images that were totally new and the models had no prior exposure to these images. Test data was classified by three different means:

1) Human dentist
2) Clarifai API
3) Retrained Inception Model

Clarifai performed the best with an accuracy of 100%, followed by Retrained Inception with an accuracy of 60%. Spark Mllib performed the worst and is not included in the results because of subpar performance.

Since the test set consisted of only 6 images, the accuracy achieved might be overstated. Running the tests with up to 50 images might provide more realistic results. Clarifai performed very well compared to the other methods. Results compared with the ground truth are provided in Table 2.

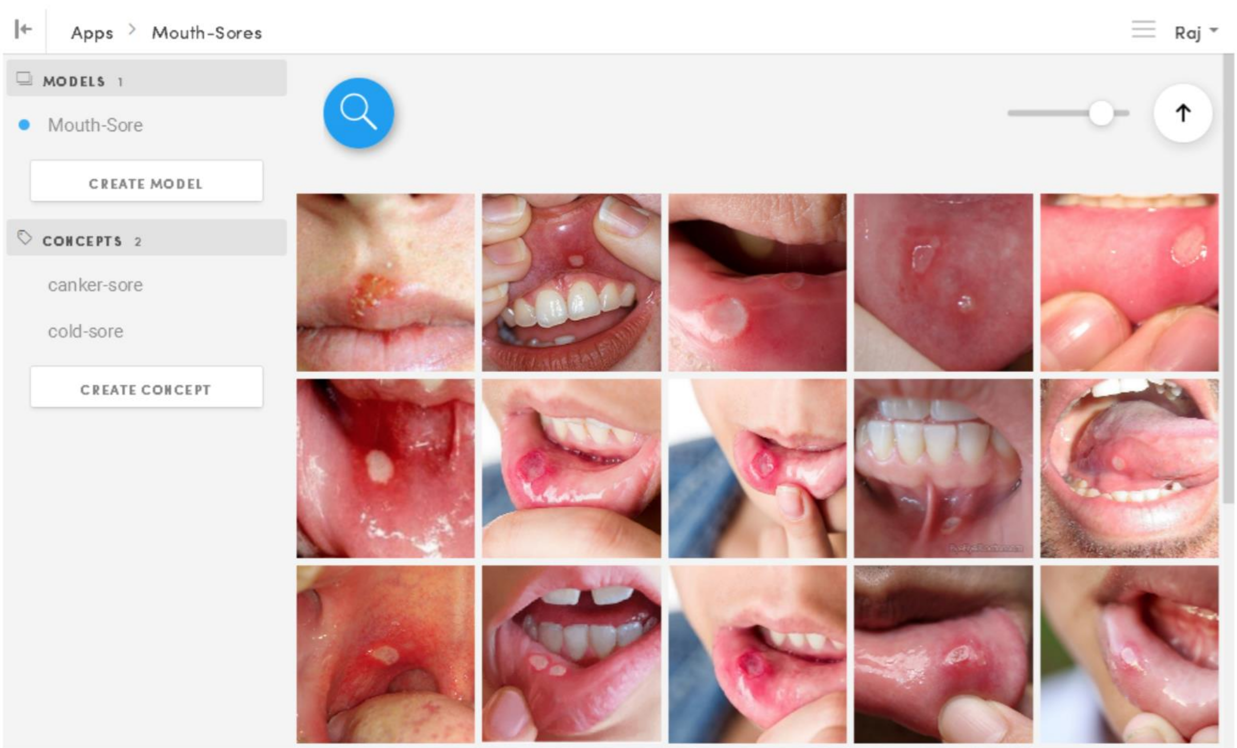

Figure-5 Clarifai Custom "Mouth-Sore" model with concepts and images.

Image	Dentist	Inception	Clarifai
1	Cold – 100%	Cold – 89.88%	Cold – 86.59%
2	Canker – 100%	Cold – 72.73%	Canker – 77.25%
3	Cold – 100%	Cold – 99.37%	Cold – 94.79%
4	Cold – 100%	Canker – 59.01%	Cold – 93.37%
5	Cold – 100%	Cold – 95.52%	Cold – 93.73%
6	Canker – 100%	Canker – 96.22%	Canker – 92.59%

Table 2 – Performance of models as compared to the ground truth

IV. Conclusion

The experiment demonstrated that a direct to consumer tool can be utilized to do some preliminary diagnosis of dental or oral health problems. The dataset used was very small and as a direct result of that, prediction accuracy suffered. More research is necessary in the application of deep learning principles and methods to dentistry and oral diseases.

.

References

[1] Imangaliyev S., van der Veen M.H., Volgenant C.M.C., Keijser B.J.F., Crielaard W., Levin E. (2016) Deep Learning for Classification of Dental Plaque Images. In: Pardalos P., Conca P., Giuffrida G., Nicosia G. (eds) Machine Learning, Optimization, and Big Data. MOD 2016. Lecture Notes in Computer Science, vol 10122. Springer, Cham

[2] Krizhevsky, A., et al.: Imagenet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems, pp. 1097–1105 (2012)

[3] Marsh, P.: Microbial ecology of dental plaque and its significance in health and disease. Adv. Dent. Res. 8(2), 263–271 (1994)

[4] Stutz, D.: Understanding convolutional neural networks (2014)

[5] A. Vashishth, B. Kaushal, and A. Srivastava, “ Caries Detection Technique for Radiographic and Intra Oral Camera Images,” International Journal of Soft Computing and Engineering (IJSCE), pp. 2231-2307, Volume-4, Issue-2, 2014.