Morzouk Lawal
Morgan State University
Electrical Engineering
morzouklawal@gmail.com

Marzieh Edraki
University of Central Florida
Computer Science
m.edraki@knights.ucf.edu

Dr. GuoJun Qi
University of Central Florida
Computer Science
guojun.qi@ucf.edu

Abstract

Over the years, IoT devices and various social network apps have grown rapidly. These apps are constantly collecting, storing and sharing images and other user-generated contents. These images being collected and shared all at once can become congested and difficult to classify. Therefore, our aim is to assign the most relevant label to user-generated content such as images, using deep learning techniques such as Convolutional Neural Networks (CNN) and Long Short Term Memory (LSTM). We have developed deep learning algorithms to help people rapidly retrieve multi-modal content on their smart device. An example would be a picture of an Airplane flying in the sky. The best possible classification our algorithm will make will either be "Airplane" or "Sky". Another example will be an Image of a train on the railroad, the model will classify this as either a "Train" or a "Railroad".

Approach

The Image classification process involves 3 steps:

Input: The input consist of a total of 13,000 images and 20 different classes. Each class contains 650 images. We used 85% (11,000) of the total number of images as the training set and the other 15% as the test set.

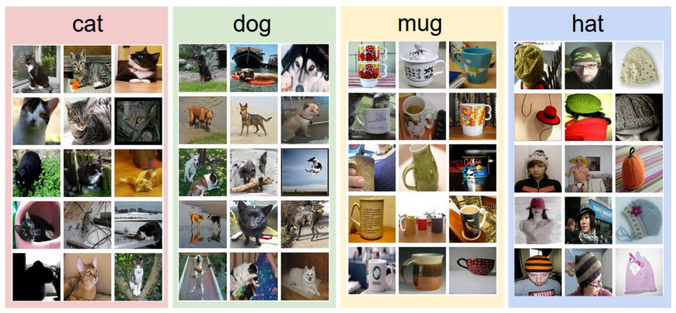

Fig2: An example of a training set with four visual categories. In practice we may have thousands of categories and hundreds of thousands of images for each category.

CNN and LSTM

Convolutional Neural Networks are made up of neurons that have learnable weights and biases. They transform a 3D input volume into a 3D output volume based on the parameters given. We've augmented the convolutional neural network with gating idea from LSTM.

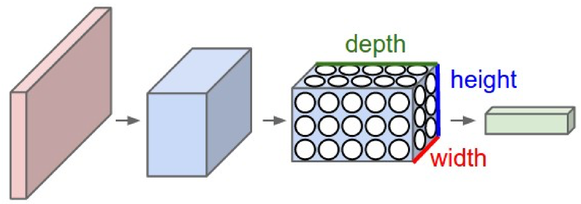

Fig2: A ConvNet arranges its neurons in three dimensions (width, height and depth), as visualized in one of the layers.

LSTMs are a special kind of Recurrent Neural Network(RNN), used to model and predict the behavior of time series and input data. As shown in Fig3 below, the input that goes in the LSTM unit have to pass through an input modulation gate, this gate decides how much of the new information should be entered in the LSTM unit cell. The Forget gates then decides what portion of the previous state of unit cell should be kept and what portion should be forgotten. Then, the output gates modulate the flow of information out of LSTM unit.

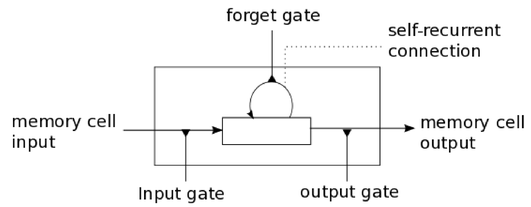

Fig3: A simple LSTM Block showing the input gate, the forget gate and the Output.

Proposed Model

Our model consist of three major types of layers:

1) Gated Convolutional layer: This is a combination of the convolutional neural network and the LSTM idea. The gates in this layer regulate the flow of information among layers and pass relevant information on to next layer.
2) Convolutional Layer: This is consist of CNN and Max pooling. The pooling layer takes a rectangular layer from the convolutional layer and outputs the maximum value for each sub-region.
3) Fully Connected Layer: The fully connected layer is the final layer. It takes all the neurons in the previous layer and connects them together.

Training: The task is to use the training set to learn what every image in each class looks like. We trained the proposed model by back propagating the error of representative cost function down through the network.

We used the negative log likelihood of Softmax output to delegate the cost. The training process is time consuming and using CPUs, this process can take more than a week. Fortunately for us, we had the processing power of today's GPUs and hence, we were able to model in 1 day.

Evaluation: In the end, we evaluate the quality of the classifier by asking it to predict labels for a new set of images that it has never seen before. We test the model using 2000 images, which is about 15% of the total number of images. We then compare the true labels of these images to those predicted by the classifier. Intuitively, we hope the predictions match the true labels.

Results

So far, in our results, we have an accuracy of 23%. We plan on further fine tuning the parameters by cross validation, to get better accuracy. The process of fine tuning is time consuming. We also encountered problems with the RAM of the computer, as the memory was not sufficient enough to perform the training. Therefore, we had to add another convolutional layer before the softmax layer, reducing the size of the feature map by ¼. This also reduced the accuracy of our model. Eventually, we plan on removing this convolutional layer, and thus improving the accuracy of the model.

Future Work

Future work will include:

- **Image Synthesization:** This algorithm can be employed and modified to generate new images using a rough description of the Image.
- **Semantic Segmentation:** Images can further be classified pixel by pixel i.e. semantic segmentation.

Selected References

[1] K.Pathy. "CS231n Convolutional Neural Networks for Visual Recognition." *CS231n Convolutional Neural Networks for Visual Recognition*. Stanford, n.d. Web. 19 July 2016.

[2] S.Hochreiter and J. Schmidhuber. Long Short-term memory *Neural computation*, 9(8):1735-1780,1997.1,4

[3] Krizhevsky, Alex, Ilya Sutskever, and Geoffery E. Hinton. *ImageNet Classification with Deep Convolutional Neural Networks*.. In Advances in neural information processing systems, pages 1097-1105, 2012. 1

Acknowledgement

The support for this work was provided by the National Science Foundation REU program under Award No. 1560302. Any opinions, findings, and conclusions and recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.