MidTerm Report: Bully Images Recognition

Haotian Deng and Qingbo Lai

[1] School of Computing, Clemson University
[2] hdeng@clemson.edu and qingbol@clemson.edu

1 Introduction

Detecting Images with bully Actions is still a challenging problem. In this project, we implement two deep learning models, one is based on Convolutional Neural Network (CNN) models[6] which written by ourselves and modified VGG16 module[5]. Our model are very practical modeles. We use the data which given by CPSC-8810 Deep Learning course and some nonbullying action images from the internet, we also augment our dataset by several ways. we trained our own model and adjusted many parameters and architectures in order to get better accuracy and got good results, but we still have lots of work should to be done in the rest of the semester.

In the report, section 2 talks about our project setting and models, section 3 illustrate our experiment on the datasets, section 4 describe the future work and plans of final project, section 5 is a conclusion.

2 Project Description

The project's goal is develop deep learning models to detect images with bullying actions. In our whole project design, we plan to combination Attentional Pooling[2] and Contextual Action Recognition with R*CNN technologies[3], we also plan to use face or post detection modules to increase accuracy rate and mean AP. However, in the mid term we do not have enough time to do very complicated work, so we worked on classifying and recognize the bullying images' kinds. We chose to make two deep learning models which one is totally written by ourselves with simple CNN network, the other one is revise based on VGG16[5] models. Simple CNN model can make us learn so many details about deep learning network and easy to implement, after implement the simple CNN model we learn and revise VGG16 which is very famous model and with good performance. In this projectour models can be well classified from 10 types of bully and nonbully images, and can also predict the type of single image. Our models also trained, revised and fixed parameters by ourselves. More details about model like the specific accuracy and performance, parameters in Section 3.

2.1 Model Architecture

Simple model have a not complex architecture, but we do it by ourselves and the effect is good. We totally have 11 layers. First is one input layer, then we have three groups of layers which are combinations of convolutional layer and max pooling. After that, we set a flatten layer in orde to transfer Multi-dimensional input to one-dimensional, we use this layer as a transition from convolutional layer to fully connected layer. So we have fully connected layer after flatten layerwe have softmax layer in the end.

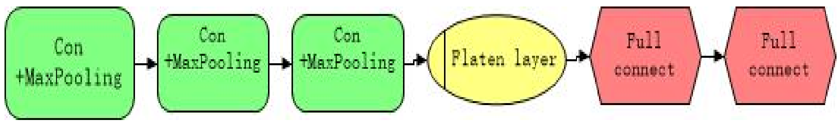

Fig. 1. Simple CNN model architecture

The following table is parameter of Simple CNN network:

Table 1. Parameters of Simple Network

Parameters	Values
Con1	3*3*32
Con2	3*3*32
Con3	3*3*64
fc1-2	128

We also developed a deep learning network based on VGG16[5], our VGG16 have 13 convolutional layers, 5 pooling layers, 1 flaten layers and three fully connected layers. the size of layers are illustrated on table.

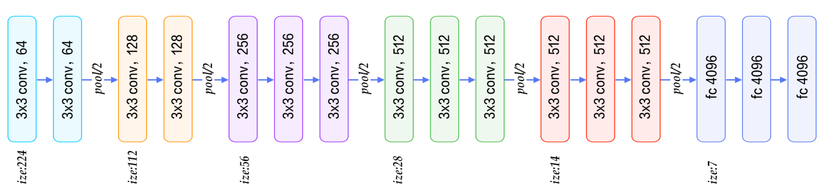

Fig. 2. VGG16 model architecture[5]

The following table is parameter of VGG16 network:

Table 2. Parameters of VGG16

Parameters	Values
Con1-2	3*3*64
Con3-4	3*3*128
Con5-7	3*3*256
Con8-10	3*3*512
Con11-13	3*3*512
Con14-16	4096

3 Experimantal setup

Our enviroments are Python3.6 and Tensorflow basic framework 1.12all the codes implemented on Palmetto Cluster[4] "https://palmetto.clemson.edu/palmetto/" which provided by Clemson University. Our bully picture dataset provided by class and nonbully dataset[1], we download from website. The totally dataset image amount are 2387.

3.1 Parameters setting

We test our model so many times and fix the parameters, the following tables are our model parameter setting include Hyperparameters and parameter of networks.

Table 3. Hyperparameters

Parameters	Values
Learning rate	0.0001
iteration times	20000 s
batch size	64
validation size	0.2 dataset
Image size	128

3.2 Results

We design two Api (script) to develop two functions of our models, we test on the dataset which provided by class.

Our model can predict single image, For example

Fig. 3. Predict image

Our model can classify bully images, For example:

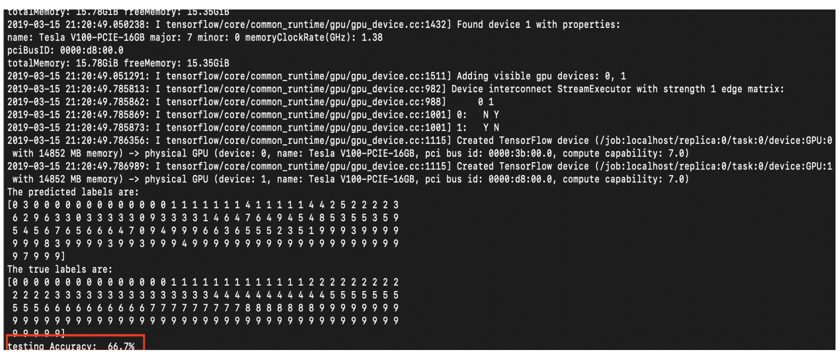

Fig. 4. Predict image

We can find our models successfully predict the image which category is gossiping and can classify images to ten categories like laughing, pullinghair, quarrel, slapping, punching, stabbing, gossiping, strangle, isolation or nonbullying.

4 Future Work

We still have lots of work to do in the future. First, we should make some attentional pooling in to model

Bully action recognition in still or static images is a challenging problem for a variety of reasons. Recognizing bully actions of people in images, we want to identity of the objects surrounding them and the way they interact with those objects and the scene are vital cues, using all surrounding information to recognize, we also plan to add attentional pooling technologies and detecting models.

5 Conclusion

We have put a lot of effort into this project. From the very beginning, we have blurred the concept of deep learning. Now we can write our own model, adjust the hyperparameters, and complete the project completely. In this, we have learned a lot and improved the ability to solve problems. However, our accuracy rate is not particularly high, we still have a lot of learning and improvement space in the next half of the semester.

Bibliography

[1] Actions, S. . (2015). Stanford 40 actions. In *Stanford 40 Actions*.

[2] Girdhar, R. and Ramanan, D. (2017). Attentional pooling for action recognition. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R., editors, *Advances in Neural Information Processing Systems 30*, pages 34–45. Curran Associates, Inc.

[3] Gkioxari, G., Girshick, R., and Malik, J. (2015). Contextual action recognition with r*cnn. In *The IEEE International Conference on Computer Vision (ICCV)*.

[4] palmetto (2015). palmetto. In *https://www.palmetto.clemson.edu/palmetto/*.

[5] Qassim, H., Verma, A., and Feinzimer, D. (2018). Compressed residual-vgg16 cnn model for big data places image recognition. In *2018 IEEE 8th Annual Computing and Communication Workshop and Conference (CCWC)*, pages 169–175.

[6] Sharif Razavian, A., Azizpour, H., Sullivan, J., and Carlsson, S. (2014). Cnn features off-the-shelf: An astounding baseline for recognition. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*.