Honors Project(Stage 1)

Akilesh B, CS13B1042

Indian Institute of Technology, Hyderabad

November 19, 2016

Diabetic Retinopathy (DR)

- Diabetic Retinopathy (DR) is a diabetic complication that affects the eyes with the potential to cause severe vision loss and blindness.
- It is regarded as the leading cause of blindness in the working-age population.
- In this work, we investigate the performance of Very Deep Networks for the binary classification of fundus images provided by EyePACS as part of Kaggle's DR detection challenge.

Our work

- We investigate two broad categories of very deep networks in this work:
    1. A standard CNN (DeepNet) with many layers.
    2. Residual Networks (ResNets).
- In ResNets, there are shortcut connections allowing very smooth forward and backward propagation (bypassing the vanishing gradient problem, which restricts the depth of standard CNNs).
- We further study regularization using a recent approach of adding gradient noise.

Regularization using Gradient Noise

- The addition of gradient noise has recently been shown as a good regularizer that helps the network find better local minima, which is important in very deep networks.
- As proposed by Neelakantan et al., we add time-dependent Gaussian noise to the gradient g at every training step t: $g_t \leftarrow g_t + N(0, \sigma_t^2)$ where $\sigma_t^2 = \frac{\eta}{(1+t)^\gamma}$.
- In our case, we found after substantial experimentation that the values $\eta = 0.01$ and $\gamma = 0.55$ worked particularly well.

Dataset

- Fundus images were obtained as part of EyePACS, which was made available through Kaggle's DR detection challenge.
- A rating of 0 corresponds to a healthy patient, while ratings 1, 2, 3 and 4 indicate mild, moderate, severe and proliferative DR, respectively.
- In our work, we focused on binary classification with classes *Healthy* (class 0) and *DR* (class 1), considering solving this seemingly simpler problem can have a direct impact on translation into real-world application by itself.

Handling Class Imbalance

- The frequencies of ratings 0 to 4 in the dataset are 73.48%, 6.95%, 15.07%, 2.49% and 2.02%, respectively.
- In our binary classification setting, the class composition turns out to be: 80.43% healthy and 19.57% diseased.
- While using the mean squared loss objective function, this results in a strong incentive to predict 0 for all the images.
- An effective approach to address this imbalance using sampling methods is to oversample the minority class with a factor that is inversely proportional to the frequency of the class, which we call *replication factor*.
- The replication factor for Class 1 is 4.1, in our case.

Handling Class Imbalance contd ...

- While this approach can be effective, considering the fact that the test distribution is imbalanced too, the specificity can decrease if the replication factor is used always during training.
- Hence, we adaptively decrease the replication factor, r, over time by setting $r_t = \rho r_0 + (1 - \rho^t) r_f$, where $r_0 = 4.1$ from the data, and r_f was chosen to be 2 based on empirical studies.
- $\rho \in (0, 1)$ is a constant factor which controls the rate of decrease of the factors.
- In our experiments, we found $\rho = 0.975$ to work reasonably well.

Performance Metrics

$$Sensitivity = \frac{TP}{TP + FN}$$

$$Specificity = \frac{TN}{TN + FP}$$

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN}$$

where TP, FP, TN and FN are true positives, false positives, true negatives and false negatives, respectively.

Data Augmentation

- Deep networks need large amounts of data for learning a good classifier, and data augmentation using image transformations (such as mirroring, translation, jitters, etc.) are commonly used to bloat the training data.
- We employed two kinds of augmentations in this work:
  1. *Geometric Transformations (e.g. zooming, translation, rotation, mirroring, shearing)*.
  2. *Color Transformations:* These augmentations simulate varying intensities and colors of the illumination setting.

Results

- We trained all networks using Stochastic Gradient Descent with Nesterov momentum on mini-batches of sizes 128 and 64 for DeepNet and ResNets respectively.
- The momentum parameter was $\mu = 0.88$. We trained for a total of 500 epochs with a fixed learning rate schedule, starting with a learning rate of 0.006 and decreasing by a factor of 10 after 200 epochs and, again, after 400 epochs.
- Orthogonal initialization was used for all layers. L_2 regularization, penalizing all weights of the network but not the biases, was added to the objective function, weighted with a factor of 0.005.
- Implementation of the above architectures was done using Theano and Lasagne libraries.
- The training was done on a high-end Tesla K-80 GPU, with 12GB RAM.

Results . . .

Table: DeepNet performance on 512 $\times$ 512 images

Method	Sensitivity	Specificity	Accuracy
State-of-the-art CNN	89	86	88
DeepNet	91	89	90
DeepNet*	94	91	93

Results . . .

Table: ResNets on 128×128 images

Method	Sensitivity	Specificity	Accuracy
ResNet 18 layer	88	87	87
ResNet 18 layer*	90	89	90
ResNet 34 layer	90	88	89.5
ResNet 34 layer*	92	91	91
ResNet 50 layer	91	90	90
ResNet 50 layer*	93	91	92.5
ResNet 101 layer	94	92	92.5
ResNet 101 layer*	96	94	95

Results . . .

Table: DeepNet vs ResNet on 128×128 images

Method	Sensitivity	Specificity	Accuracy
DeepNet*	90	88	89
ResNet 101 layer*	96	94	95

Remarks

- Accepted and presented at IBM I-CARE 2016.
- Also, presented during the poster session of 2nd CSE day, IITH, Oct. 2016.

Student Teacher Networks

- A shallow student network is trained from a deep teacher network (Used in deep model compression).
- Build on the idea of teacher-student learning algorithm (considering a pretrained teacher such as AlexNet or VGGNet, this will also reduce training time complexity.)
- Our objective is to characterize the intelligence of student network.
- That is, given a teacher network, provide the best student network (how deep? what filter size for convolution? etc.) which matches closely to the performance of teacher network.

Questions?

Thank you!