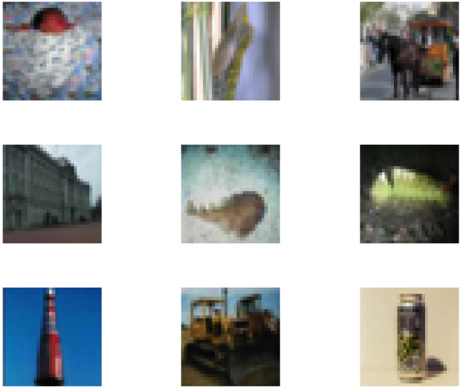
Lecture 5: Deep learning
COSC470

Lech Szymanski
Department of Computer Science, University of Otago

August 7, 2018

CIFAR100 classification task

Labels: aquatic mammals, fish, flowers, food containers, fruit and vegetables, household electrical devices, household furniture, insects, large carnivores, large man-made outdoor things, large natural outdoor scenes, large omnivores and herbivores, medium mammals, non-insect invertebrates, people, reptiles small mammals, trees, vehicles



State of the art accuracy: 75%

CIFAR100 classification task

Labels: aquatic mammals, fish, flowers, food containers, fruit and vegetables, household electrical devices, household furniture, insects, large carnivores, large man-made outdoor things, large natural outdoor scenes, large omnivores and herbivores, medium mammals, non-insect invertebrates, people, reptiles small mammals, trees, vehicles

State of the art accuracy: 75%

In this lecture...

- Evolution of deep learning
- Deep learning models - how do they work?
- Tensorflow - how to build them?

History of neural networks (review)

- McCulloch and Pitt's neuron (40's)
- Rosenblatt's perceptron and the perceptron learning rule (50's)
- Minsky and Papert's perceptron (60's)
- Backpropagation (80's)
- LeCun's Convolutional Neural Network's (ConvNets) (90's) [1]

Sigmoid neural networks (2000's)

- Would call them now fully connected now - every neuron connects to every neruon between the layers
- Not obvious how many layers and neurons to use...(but in hindsight)
  - Not that many layers to play with (more than 5 encountered disappearing gradient problem)
  - Don't have to worry about connectivity between layers (fully connected)
- Capable of working well, but in practice hard to train
- SVM's were just better
- Scaled well with problem size, but not well in architecture (breadth nor width)

Deep belief nets (DBNs) (2006) [2]

- Learning layer by layer
- A stack of Restricted Boltzmann Machines (RBMs) trained in unsupervised way using Hinton's contrastive divergence algorithm
- Supervised fine-tuning (using backpropagation)
- Popularised softmax output training with cross-entropy

Resurgence of ConvNets (2012)

- Kryzhevsky's convolutional neural network wins ImageNet Large-Scale Visual Recognition Challenge (ILSVRC 2012), with top 5 test error rate of 15.4% (beating next best by 10%) [3]
- Use of ReLU activation functions
- Use of GPUs for training

Deep learning explosion (2014 and on)

- Industry adopts neural networks (2014)
  - Google tests a convnet for a natural language processing tasks and beats its best model by 5%
- Everyone else starts using deep learning for everything that has to do with images and speech/language
  - GoogLeNet wins ILSVRC 2014 with error rate of 6.7%
  - VGG net [4] doesn't win ILSVRC 2014 (error rate of 7.3%)...but its filters become standard for transfer learning in image related tasks.
  - Microsoft ResNet wins ILSVRC 2015 with error rate of 3.6%.
- AlphaGo (2016) - computer beats human master at Go on a 19x19 board without *handicaps*.

But not all is well...

- Deep neural network make surprising mistakes [5];
- it's not clear why generalisation is so good when the models are big enough to learn anything;
- it takes skill, experience and intuition to build good deep models, including use of many *tricks*...

Regularisation

Regularisation adds a penalty for the model complexity to the loss function. It often tends to push parameters to go towards zero.
Example: An L2 norm regularisation.

$$\text{Minimize: } \mathcal{L}\big(f(x,\beta), y\big) + \alpha\beta^2$$

Other tricks

- drop-out
- adam optimiser
- batch normalisation
- data augmentation

Designing a ConvNet

This example is based on a ConvNet for CIFAR-10 capable of reaching 86% classification accuracy.

CIFAR10

- Image size: 32x32x3
- Train images: 60000
- Test images: 10000
- Number of classes: 5

Augmentation:

- Rnd crop to 24x24
- Rnd horizontal flip/brightness/contrast (training only)
- Normalise.

ConvNet:

- L1 conv, 5x5x64, str 1, bias 0, stdev 5e-2, ReLU
- L2 maxpool, 3x3x64, str 2
- L3 norm
- L4 conv, 5x5x64, str 1, bias 0.1, stdev 5e-2, ReLU
- L5 maxpool, 3x3x64, str 2
- L6 fc, 384 neurons, bias 0.1, stdev 0.04, wd 0.004, ReLU
- L7 fc, 192 neurons, bias 0.1, stdev 1/192, wd 0.004, ReLU
- L8 fc, 5 neurons, bias 0, softmax

Training a ConvNet

This example is based on a ConvNet for CIFAR-10 capable of reaching 86% classification accuracy.

- Model parameters $\beta = \{\kappa, \omega\}$ consist of conv layer parameters κ and fully connected layer parameters ω
- Stochastic steepest gradient descent optimisation with learning momentum.
- Mini-batch size of 128
- For update at time t gradient $\Delta\beta_t = \{\Delta\kappa_t, \Delta\omega_t\}$ of the loss function $\mathcal{L}\big(f(x,\beta), y\big) + \alpha\omega^2$ with respect to β is computed using backpropagation
- Update: $\kappa_{t+1} = \kappa_t - \mu_t \Delta\kappa_t$
- Update: $\omega_{t+1} = \omega_t - \mu_t(\Delta\omega_t + \alpha\frac{1}{2}\omega_t)$
- $\mu_t = 0.1^{1+(t/136500)}$

Building a convent in Tensorflow

- If you missed my introductory lecture for on Tensorflow for COSC420 last semester, refer to `https://altitude.otago.ac.nz/lechszym/tfintro`.
- Anaconda+PyCharm+Tensorflow

References

[1] Y. LeCun, L. Bottou, Y Bengio, and P. Haffner. *Gradient-Based Learning Applied to Document Recognition* Proceedings of the IEEE, 86(11):2278-2324, 1998.

[2] G. E. Hinton, S. Osindero, and Y-W. Teh. *A Fast Learning Algorithm for Deep Belief Nets* Neural Computation, 18(7):1527–1544, 2006.

[3] A. Krizhevsky, I. Sutskever, and G. E. Hinton. *ImageNet Classification with Deep Convolutional Neural Networks*, Proceedings of the 25th International Conference on Neural Information Processing Systems, 1:1087-1105, 2012.

[4] K. Simonyan, and A. Zisserman. *Very Deep Convolutional Networks for Large-Scale Image Recognition*, CoRR, `http://arxiv.org/abs/1409.1556`,2014.

[5] A. Nguyen, J. Yosinski, J. Clune. *Deep Neural Networks are Easily Fooled: High Confidence Predictions for Unrecognizable Images*. In Computer Vision and Pattern Recognition (CVPR '15), IEEE, 2015.