CS519: Deep Learning

Winter 2016
Course Information

• Instructor: Dr. Fuxin Li
  • KEC 2077, lif@eecs.oregonstate.edu
• TA: Mingbo Ma: mam@oregonstate.edu
• My office hour (tentative): Mon 4-6PM
• Class Webpage:
  http://classes.engr.oregonstate.edu/eecs/winter2016/cs519-006/
• Questions/Discussions – on CANVAS
Prerequisites

• Significant knowledge on machine learning, especially the generics (not specific algorithms)
  • CS 534 or equivalent knowledge
  • Refresher will be provided in the next lecture

• Some knowledge of numerical optimization
  • 1.5 weeks will be devoted to optimization and also deep network optimization
Grading

• Initial quiz (5%) based on participation only
• Assignments (30%)
  • No late assignments
  • No downloading code from the Internet
• Quizzes (3 more quizzes totaling 15%)
  • Based on whether you answer the questions correctly
• Final Project (50%)
  • Final project is to be done with teams not more than 3 participants
  • Grading will be done according to:
    • Initial proposal (10%)
    • Final presentation (40%)
Materials

• No published book on this subject yet

• Book in progress:
  • Quite a lot of contents already

• More readings can be found at:
  • http://deeplearning.net/reading-list/
  • http://colah.github.io/
  • http://karpathy.github.io/
  • https://www.coursera.org/course/neuralnets
Toolboxes

• A plethora of deep learning toolboxes around:
  • Caffe
  • Theano
  • Torch
  • TensorFlow

• Toolbox policy:
  • We do not stick to one toolbox
    • Recommended: Caffe for ConvNets
    • Recommended: Keras for other types of neural networks
  • You are responsible to select the one you are more comfortable with
Outcome

• Understand the concepts of deep learning
• Gain some intuitions on deep networks
• Understand the training of deep learning
• Be able to use at least one deep learning toolbox to design and train a deep network
• Be able to design new algorithms and new architectures
What will be covered

• Basic neural network structure
• Training tricks (SGD, Momentum etc.)
• CNNs
• LSTMs
• Unsupervised neural networks
• Neural reinforcement learning (Dead week)
Final Project

• Groups of no more than 3 persons
• Jointly work on a significant project
  • Must use deep learning
  • CANNOT be just running an already-trained classifier on some images
  • Try to solve a real problem
  • One can elect projects from paper readings
  • I will also suggest some standard projects
  • New neural architectures/changes to current architectures are welcome
• Grading – based on the project merit, execution and presentation
Computing Resources

• Working on it
  • Working on getting some free credits for each
  • As well as getting some departmental help on extra AWS credits

• Most likely AWS: (https://aws.amazon.com/)
  • Contains “images” of toolboxes that you can load up quickly
  • Have a GPU-based resource (Tesla K10 equivalent) for $0.65/hr
  • An estimated 200 hours would be needed for each student

• If you want to buy your own:
  • Website will link you to a good article
  • GTX Titan X, GTX 980 Ti, GTX 980, GTX 960 4GB (sorted descendingly by price)
Approximate schedule (will be on website)

• Week 1 (Jan. 4 - 10)
  • 1. Admin + General Introduction
  • 2. Machine Learning Refresher (linear algorithms, empirical risk minimization, regularization, support vector machines)

• Week 2 (Jan. 11 - 17): Standard neural networks
  • 3. Machine Learning Refresher (unfinished parts) + Optimization Primer #1 (nonconvex optimization, stationary points and saddle points, optima, gradients)
  • 4. Basic Neural Networks with Hidden Layer (backpropagation)

• Week 3 (Jan. 18 - 24): Convolutional Networks
  • 5. Convolutional Neural Networks (mostly in computer vision)
  • 6. Continued CNN, Visualization of CNN

• Week 4 (Jan. 25 - 31): Deciding what project to work on
  • 7. An overview of other neural models
  • 8. Introduction of deep learning toolboxes (Caffe, Keras)

• Week 5 (Feb. 1 - 7): Project proposals
  • 9. Project Proposals
  • 10. Neural Network Optimization 1
Approximate schedule

• Week 6 (Feb. 8 - 14): Neural Network Optimization, lead to temporal models
  • 11. Neural Network Optimization (stochastic mini-batch gradient descent, momentum, dropout, learning rate and weight decay)
  • 12. From CNN to temporal neural models

• Week 7 (Feb. 15 - 21): Temporal Neural Models
  • 13. Temporal Neural Models (RNNs and LSTMs)
  • 14. Continued Temporal Neural Models (LSTMs, GRUs, stacked together with CNNs)

• Week 8 (Feb. 22 - 28): Deep Learning Frontiers
  • 15. Deep Learning in Natural Language Processing (Guest lecture from the Algorithms for Computational Linguistics group)
  • 16. Reading of important recent deep learning papers

• Week 9 (Feb. 29 - Mar. 6): Unsupervised Methods
  • 17. Unsupervised deep learning (autoencoders)
  • 18. Restricted Boltzmann Machines and Deep Belief Networks, convolutional DBN

• Week 10 (Mar. 7 - Mar. 13): Deep Reinforcement Learning
  • 19. Deep reinforcement learning (guest lecture by Alan Fern)
  • 20. Deep reinforcement learning (guest lecture by Alan Fern)

• Week 11 (Mar. 14 - Mar. 18)
  • 21. Project Presentations