Deep Neural Structures and Learning

"Deep Learning" systems, typified by deep neural networks, have become essential for most AI tasks, including language understanding, speech and image recognition, statistical machine translation, planning, game playing, and autonomous driving. As a result, expertise in deep learning has become a mandatory prerequisite in many advanced academic settings and in the industrial job market. In this course, we will learn about the basics of deep neural networks and their applications to AI tasks in a variety of domains. Emphasis will be placed on the neural structures (see below) that have resulted in deep learning becoming so successful and ubiquitous for addressing complex real-world AI problems.

Prerequisites: A good understanding of python (including programming, optimization, Jupyter/CoLab, and modules such as NumPy, scikit-learn), background in probability, statistics, linear algebra, calculus, and some background on the basics of machine learning principles.

Learning Outcomes:

By the end of the course, it is expected that students will:

1. Have a better understanding of why deep learning systems have been so successful for addressing AI problems; develop good familiarity with the subject, and apply deep learning to a variety of tasks. They will also be positioned to understand the current literature on the topic and extend their knowledge through further study.
2. Understand the most important deep neural structures and try to gain an intuitive understanding of their unique benefits. This includes: distributed representations, Convolution, GANs, residual layers, ReLU units, algorithmic and data regularization (e.g., incremental/stochastic gradients, dropout, mixup, adversarial training), gating mechanisms (attention, LSTMs, GRUs), autoencoders, end-to-end learning (such as sequence to sequence), GPU availability, mini-batch training and batch normalization, and toolkits and auto-differentiation.
3. Understand the principles of and intuition behind gradient-based optimization and learning for non-linear non-convex optimization.
4. Understand various traditional learning paradigms (e.g., supervised, unsupervised, semi-supervised, active learning, ensembles) as new learning paradigms (e.g., machine teaching, curriculum learning, self-supervised learning, meta-learning, transfer learning) in the context of deep neural systems.
5. Understand the historical context of deep learning, including perceptrons, the xor problem, multi-layered perceptrons, and the apparent periodic nature of neural network popularity.
6. Have developed experience programming deep learning systems on a variety of data modalities. This includes having the experience of debugging such systems.
7. Be aware of state-of-the-art deep learning systems, such as GPT-3. Understand some of the philosophical questions regarding strong AI and societal implications.

Assignments:

Each assignment will be a combination of implementation and training of models and a series of statistical, mathematical, or philosophical questions to answer. There will also be a series of multiple-choice quizzes to answer online. The course will consist of the following four main assignments:

1. Learn to train a simple MLP training and experiment with depth and width of the model and study the intermediate layer representations.
2. Train, study and understand convolutional networks on CIFAR datasets.
3. Train and study fully connected RNN and LSTM models (on time series data).
4. Train and study GAN models and use them for exciting tasks.

Rough Schedule:

Week 1: Class logistics, History of NNs, perceptrons, multi-layered perceptrons, universal approximators, the human brain, Boolean circuit analogies, why depth, distributed representations, abilities, network capacity.

Week 2: Learning, risk minimization, logistic/softmax, function minimization, gradient descent, the mathematics of backpropagation.

Week 3: Convergence issues, Loss Surfaces, Momentum, Batch Size, SGD, Minibatch, second-order methods

Week 4: Optimizers and Regularizers, loss functions, Batch normalization, Dropout, algorithmic and data regularization (e.g., incremental/stochastic gradients, dropout, mixup, adversarial training)

Week 5: Shift invariance Convolutional Neural Networks, Learning in CNNs, transpose Convolution, residual layers

Week 6: Time Series and Recurrent Networks, Stability and Memory, LSTMs, gating mechanisms (attention, GRUs)

Week 7: Loss Functions in RNNs, Sequence Prediction, Connectionist Temporal Classification, Sequence prediction, sequence to sequence methods, end-to-end learning

Week 8: Representations and Autoencoders, Variational Auto Encoders, EM and Variational Bounds, Variational Auto Encoders

Week 9: Generative Adversarial Networks

Week 10: Hopfield Nets and Boltzmann Machines, toolkits and auto-differentiation, review.

Instructor:

Jeffrey A. Bilmes is a professor at the Department of Electrical Engineering at the University of Washington, Seattle, Washington, and an adjunct professor in Computer Science & Engineering and the Department of Linguistics. Prof. Bilmes is the founder of the MELODI (MachinE Learning for Optimization and Data Interpretation) laboratory at UW. Bilmes received his Ph.D. from the Computer Science Division of the Department of Electrical Engineering and Computer Science, University of California in Berkeley, and an MS degree from MIT. Prof. Bilmes pioneered (starting in 2003) the development of submodularity in machine learning and received a best paper awards at ICML 2013, NeurIPS in 2013, and ACM-BCB in 2016, all in this area. In 2014, Prof. Bilmes also received a most influential paper in 25 years award from the International Conference on Supercomputing. Prof. Bilmes is also the founder and CEO of a company involved in making data sets more efficient and informative for all AI and machine learning tasks. Prof. Bilmes is a 2001 NSF Career award winner, a 2002 CRA Digital Government Fellow, a 2008 NAE Gilbreth Lectureship award recipient, and a 2012/2013 ISCA

Distinguished Lecturer. Prof. Bilmes was, along with Andrew Ng, one of the two UAI (Conference on Uncertainty in Artificial Intelligence) program chairs (2009) and then the general chair (2010). He was also a workshop chair (2011) and the tutorials chair (2014) at NIPS/NeurIPS (Neural Information Processing Systems) and is a regular senior technical chair at NeurIPS/NIPS since then. He was an action editor for JMLR (Journal of Machine Learning Research).

Evaluation:

Learning projects and assignments, approximately balanced number of points for each.