CS 510, Spring 2018, Second Half of Semester Review
May 8, 2019

The following is not an exhaustive list of what has been covered, and to be very explicit, the Final Exam may cover material discussed in class not mentioned below. That said, hopefully the following will be a helpful as you review the broad topics and techniques we have covered since the Midterm in CS510.

1. There are two complementary ways to understand how a perceptron performs pattern recognition. One is of course the linear algebra underlying the algorithm, and the other is through pictures typically portraying different classes of data on a sheet of paper. The strength of having mastered both interpretations is the ability to relate one to the other. In particular, it's helpful in answering questions such as: "what is the unique solution found by a perceptron that separates two separable classes?"
2. When thinking about perceptrons is as good a time as any to reinforce your ability to explain in geometric terms how a support vector machine might accomplish a classification task solvable using a perceptron.
3. In the late 1960s and early 1970s there was a backlash in terms of enthusiasm for what a perceptron could and could not accomplish. Then, in the 1980s the pendulum swung back in favor of neural networks. As someone now familiar with the most basic aspects of neural networks as they relate to machine learning you're comfortable explaining the principal reasons for both of these historical events.
4. Assigning blame is a nontrivial problem in the context of multilayer neural networks. You should have no difficulty explaining this issue very precisely by comparing and contrasting a neural network node using a simple perceptron style threshold versus a sigmoid function.
5. In the second half of the semester we have used several distinct examples based in TensorFlow in order to better understand the TensorFlow system itself and perhaps more important the behavior of neural nets employed to solve image recognition problems. The first of these examples illustrated explicitly the back propagation training procedure. The best source code to review at the end of the semester is the file `aloni_v01.py` distributed as a starting point for Programming Assignment 4. Through having worked directly with this code you should have a true line-by-line understanding of everything in this example.
6. It is rare in CS510 that the instructor relies heavily on specifics from a single tutorial example on the web. However, the *Handwritten Digit Recognition using Convolutional Neural Networks in Python with Keras* tutorial by Jason Brownlee is excellent and pointers to the code have been included on the CS510 resources page. This is our best in lecture example of a working CNN and you are now comfortable with your understanding of every line of this example. This is particularly useful since the tutorial is relatively short and as such illustrates the convenience, and also information hiding, implicit in the TensorFlow shift to Keras.
7. Stepping up in terms of levels of abstraction, and given what you now understand about training in general and TensorFlow in particular using the backpropagation example, consider carefully your response to someone in a job interview asking you about the way in which back propagation finds the truly best neural network to solve problems such as the NIST hand written digit problem.
8. TensorFlow is first and foremost a dataflow language for performing tensor calculations. Two aspects of the statement require that you be fluent in your ability to explain to other people what you've learned. First, what is a tensor? Second, what is meant by a dataflow engine/machine.
9. It's important that you think hard about what it means to compute with a dataflow engine. As just one way of probing your understanding and that of others, answer this question: is it possible to actually construct a dataflow graph and yet compute nothing?
10. In the lecture discussing the use of neural nets to solve image recognition problems three reasons were given for networks from the 1980s (PDP-style) failing: too many weights, too few training samples, disregard for image geometry. You might have to explain this to someone else someday, be comfortable in your ability to elaborate on each of these points.
11. Repeatedly in the history of computer vision, the right data set with the right amount of ground truth comes along and lurches the entire field forward and in a new direction. The emergence of “ImageNet” beautifully illustrates

this point. It's therefore very good thing that you can describe this data set and how it shifted the landscape of research in computer vision.

12. It is fair to give Yaan LeCun credit for most successfully popularizing and promoting Convolutional Neural Networks. CS510 this year included a full lecture on CNNs including connections to human physiology, low-level features, scaling via image pyramids, and rectified linear units. These are all aspects of modern machine learning applied to image recognition that you can now readily recognize and explain.
13. Thanks to your own team's work on PA4 and presentations by other teams you now have a first-hand intuitive sense for a variety of hyper-parameter tuning issues. These include learning rate, resolution of input, number of hidden units, number of training epochs, and more. You also have first-hand experience with the process of at least attempting to interpret as a scientist the internal representations learned by a network, for example a PDP style network trained to distinguish between animals.
14. Even for relatively simple PDP style networks the intrinsic degrees of freedom for the network may be huge. Having spent weeks now considering different NN and CNN architectures you may developed a rough feel for relative complexity, as expressed by the number of degrees of freedom, in alternative network architectures.
15. Successfully building a learning model, essentially a complex data flow graph, to solve a recognition task is nontrivial. To make the process easier for us human beings, there is an entire second act in the overall TensorFlow story, namely TensorBoard. Being able to answer questions about what can and cannot be done in TensorBoard represents a basic understanding of how programmers use TensorFlow. Perhaps even more important, being able to motivate the existence of TensorBoard in terms of what it can teach about neural network learning is key.
16. Recurrent neural networks are extremely useful for data that comes naturally as a series of recordings over time. Think for example about voice or video. Because we had a guest lecture on this topic, you have a basic understanding of how these networks differ from those used to assign labels to single images.
17. In this semester we discussed repeatedly issues relating to Machine Learning (ML) experiment design. Being precise, the distinction between training and test date repeatedly arises. Your understanding of the distinction between training and test data has mature both from CS510 and hopefully other courses relating to ML. Going forward in your own career using ML, develop as much as possible a sensitivity for the underlying implications associated with trade-offs in different experiment designs with respect to the key training data versus test data distinction.
18. Following somewhat on the previous item, you now understand that for modern applications of ML the risk of overfitting is high. There several aspects of overfitting that you are now comfortable discussing. In particular, at minimum there are two aspects of overfitting. One relates to the machine learning model, a.k.a. the architecture. The other to training and measurement vis-à-vis the tri-partition approach to machine learning exemplified by: training, validation, testing.
19. Developing a dataset for a new domain, such as you did this semester for the animal trap wildlife photographs, is a part of modern machine learning often underplayed in academic courses. As you go forward with your career think about one or two key insights you gained through this process.
20. This semester you were exposed to the work at CSU on Communicating with Computers. While you certainly are not expected to know specifics about internally how the Diana System is constructed, you certainly know can describe the overall goals of the project along with specific details about a possible future involving multi-modal agents with sight.