Reading 1

(1) Title, author and where it was published

- Local Interpretable Model-Agnostic Explanations (LIME): An Introduction
- Authors: Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin
- Published: ArXiv

(2) Summary of the content

This paper introduces a new technique for visualizing the learning behavior of a machine learning model, and it is called Local Interpretable Model-Agnostic Explanation (LIME). The authors first stated a common problem that it is not clear why one could trust the results generated from a black-box model, and that people could attest that the results could be misleading. LIME, by perturbing the input data and output the approximation for the current stage based on the randomly assigned interpretable components, could provide the users with a better understanding of what has happened in the black-box, and it could be applied to multiple complicated tasks including word embeddings, and image classification. Then, the authors provide the readers with two examples – text recognition and image classification – to illustrate the process and the effectiveness of this technique.

(3) Strengths of the paper

The authors use illustrative examples to explain the mechanisms behind LIME in the area of text recognition and image classification, making it more convincing that the technique could actually alleviate the "trust" issue in the machine learning realm. In the example of the tree from image recognition, the authors first interpret the concept of "interpretable components," which are a set of contiguous superpixels that could be turned on or off (colorful or gray) group by group during the training. Therefore, it could add an extra layer of perturbation onto the input data and thus strengthen the model's performance. It then compares the different perturbations applied on the tree from the image and claims how easy it is to fool the model if some areas are turned into graph scale.

Furthermore, the example of text recognition further explains the power of LIME. It maintains that even though the predicted result is correct, the process of learning turns out to be irrelevant. The model has learned that the word "Posting" occurs more commonly for "atheism" topics compared to "Christianity" topics. Yet such a pattern is only due to the setup of the training dataset rather than the learning process itself. Therefore, we cannot conclude that having the word "posting" occurring more times signifies that it is more "atheistic" and "Christian."

(4) Major critiques

I understand the effectiveness of LIME in tasks like image recognition and natural language processing, but I cannot conjecture how it would perform in learning tasks like audio/video processing and other regression analysis. Besides, it seems to me like LIME is a technique that pulls out the "middle" results during the learning process, so technically it does not solve the problem of potential training data leakage nor ensuring the correctness of the learning process.

Reading 2

(1) Title, author and where it was published

- Title: Visualizing and Understanding Convolutional Networks
- Authors: Matthew D. Zeiler, Rob Fergusv
- Published: ArXiv

(2) Summary of the content

This paper introduces a new visualization technique that could give insights into the functions of the intermediate feature layers and operation of the classifier. The paper begins with an overview of the efficacy and current achievements of convolutional neural networks, then it specifies that it uses standard fully supervised convnet models for this paper. The approach is then described as a technique that could map the feature activities in the intermediate layers back to the pixel space, showing what input pattern originally caused a given activation in the feature maps, and the idea is to use a Deconvolutional Network (deconvnet) that could perform the unpooling, rectification, and filtering . The authors then display the actual visualization on a fully trained model across 5 layers to demonstrate the the performance of the deconvnet approach. Later, the authors discuss other benefits that this approach brings, and also clarify for common questions using different image setups. Lastly, this paper concludes with an experiment on the performance of a model refined by deconvnet that outperforms the benchmark from other state-of-the-art models on the ImageNet classification task.

(3) Strengths of the paper

Since it is a paper about visualization on large convolutional neural networks, it includes a wide range of images to support its argument about the effectiveness of this approach. For example, when justifying for the localizability of deconvnet and the possibility that the inversion process did not capture the correct features, the authors conducted an occlusion sensitivity experiment and presented the results with 3 sets of images showing that deconvnet could capture the local features when the original images are systematically covered up by a gray square. For instance, there is an image with the Pomeranian dog and a small tennis ball, and the dog face is the most important feature; based on the visualization from deconvnet on the fifth layer, it is indeed that the model captures the dog face as the most salient attribute. Nevertheless, when the gray square covers the dog's face, it turns out that the tennis ball now becomes the most salient one, and thus the result of classification changes from Pomeranian dog to tennis ball. This experiment justifies that the approach indeed properly located the correct features on the image rather than the surrounding context.

Also, the authors include the experiment on improving Krizhevsky's model architecture based on the visualization generated by the deconvnet. After adjusting for the mix of extremely high and low frequency on some features, the new model that the authors proposed outperformed Krizhevsky and achieved an error that is the best published performance on this dataset.

(4) Major critiques

It seems like there is still lots of manual work needed even with these visualization tools. The researchers still need to perform trial-and-error on searching for the optimal solutions on a complicated task. I wonder if the deconvnet could be used to feed back its unpooled, rectified, and unfiltered results back with the network along with the gradients so that the model could self-tune its parameters and learning behaviors based on this extra information.

Reading 3

(1) Title, author and where it was published

- Title: Understanding LSTM Networks
- Authors: Christopher Olah
- Published: colah's blog

(2) Summary of the content

The author starts this article by introducing the basic concept of the Recurrent Neural Network (RNN). Unlike traditional neural networks, RNN is a chain-like network that has multiple copies of the same single layer network with historical messages passing through. Such an architecture has been widely used in fields such as speech recognition, language modeling, translation, and image captioning.
However, the problem is that in practice, RNNs could not learn information that has "long-term dependencies" on other terms, and therefore the author introduces the LSTM (Long Short Term Memory) network. Followed by a detailed description of the basic mechanism of LSTM accompanied with graphs and examples, the author then discusses potential variations on LSTM, and concludes this paper with a projection on future research topics.

(3) Strengths of the paper

This paper goes in-depth about the working flow of the LSTM network with a step-by-step walkthrough. LSTMs have chain-like structures with four interacting neural networks layers in each repeating module. The cell states keep track of the current information, and there is a gate with sigmoid activation function called "forget gate layer" that controls whether to keep this information or not. The next two layers, a sigmoid and a tanh layer, decide how to update or add new information to the network. The sigmoid layer is a "input gate input" that decides which part of the data to update, and the tanh layers perform the actual transformation. Finally, the last sigmoid layer decides how much information to output. The author illustrates this process with clear graphics, mathematical formulas, and an example from learning a language model, making more sense both visually and intellectually.
Moreover, variations like "peephole connections", coupled forget and input gates, and Gated Recurrent Unit (GRU) adds more simplicity and power to the basic LSTM model. In this part, the author did not go through in depth. He just briefly mentioned these techniques and suggested the readers explore themselves, which fits with the purpose of this blog which is to introduce the basic concepts of LSTM to the general public.

(4) Major critiques

Since it is a descriptive blog introducing LSTM in a simple and understandable way, there is no assumption to be judged about. However, one improvement might be that the author could include more examples regarding the effectiveness and performance of LSTM and how it could outperform other traditional RNNs. Moreover, despite the variations on LSTM, the author could also include some of the disadvantages of LSTM just to provide a more holistic view on this technique.