Kevin Chen

6/15/2020

Structuring Machine Learning Projects

Introduction to ML Strategy

* ML strategy is the process of finding the most promising things to try next to improve performance.
* Orthogonalization: the principle of being able to tune a system to achieve one single effect at a time.
  + Example: car has a steering wheel, a brake, and a gas pedal. Each control has a single effect, and thus achieves orthogonalization. Imagine instead that a car has two controls: one control that changes the angle at 3/8 of the rate it changes the speed and another control that change the angle at -11/5 of the rate it changes the speed. In theory, you have full control over the angle/speed of the car, but that is very hard to control since the controls are not orthogonal.
* Criteria for an ML system that works well:
  + Perform well on the training set based on cost function (e.g. achieving human-like performance). If performance is not good, try bigger network, Adam optimization, etc.
  + Perform well on dev set based on cost function. If performance is not good, try regularization, a bigger training set, etc.
  + Perform well on test set based on cost function. If performance is not good, try a bigger dev set, etc.
  + Perform well in real world based on cost function. If performance is not good, then change a dev set, change the cost function, etc.
* Notice how the methods to improve performance for the above criteria only affects that one criteria. Thus, these methods are orthogonal.
  + Early stopping simultaneously affects how well you fit the training set and the dev set, and so this control is not orthogonal.

Setting Up Your Goal

* Single number evaluation metric: you should have a single real number evaluation metric that lets you quickly tell if a change improves your performance.
  + The issue with having multiple real-number evaluation metrics is that you might not be sure which classifier is better. Example: if your evaluation metrics are precision and recall, and Classifier A has 95% precision / 90% recall and Classifier B has 98% precision / 85% recall, it’s not readily obvious which classifier is better. This makes it take longer to pick which classifier is best.
  + Precision and recall are often combined into a single number: score, which is the harmonic mean of precision and recall.
  + A dev set + single real number evaluation metric speeds up tuning of hyperparameters.