

Basic Recipe for Machine Learning

(SPEECH)

In

(DESCRIPTION)

Text, deep learning dot AI. Setting up your ML application. Basic recipe for machine learning

(SPEECH)

the previous video, you saw how looking at training error and depth error can help you diagnose whether your algorithm has a bias or a variance problem, or maybe both.

It turns out that this information that lets you much more systematically using what they call a basic recipe for machine learning and lets you much more systematically go about improving your algorithms' performance. Let's take a look.

When training a neural network, here's a basic recipe I will use.

After having trained an initial model, I will first ask, does your algorithm have high bias?

And so to try and evaluate if there is high bias, you should look at, really, the training set or the training data performance.

Right. And so, if it does have high bias, does not even fit in the training set that well, some things you could try would be to try pick a network, such as more hidden layers or more hidden units, or you could train it longer.

Maybe run trains longer or try some more advanced optimization algorithms, which we'll talk about later in this course.

Or you can also try, this is kind of a, maybe it work, maybe it won't.

But we'll see later that there are a lot of different neural network architectures and maybe you can find a new network architecture that's better suited for this problem.

Putting this in parentheses because one of those things that, you just have to try.

Maybe you can make it work, maybe not.

Whereas getting a bigger network almost always helps.

And training longer doesn't always help, but it certainly never hurts.

So when training a learning algorithm, I would try these things until I can at least get rid of the bias problems, as in go back after I've tried this and keep doing that until I can fit, at least, fit the training set pretty well.

And usually if you have a big enough network, you should usually be able to fit the training data well so long as it's a problem that is possible for someone to do, alright?

If the image is very blurry, it may be impossible to fit it.

But if at least a human can do well on the task, if you think base error is not too high, then by training a big enough network you should be able to, hopefully, do well, at least on the training set.

To at least fit or overfit the training set.

Once you reduce bias to acceptable amounts then ask, do you have a variance problem?

And

(DESCRIPTION)

High variance?

(SPEECH)

so to evaluate that I would look at dev set performance.

Are you able to generalize from a pretty good training set performance to having a pretty good dev set performance?

And if you have high variance, well, best way to solve a high variance problem is to get more data.

If you can get it this, you know, can only help.

But sometimes you can't get more data.

Or you could try regularization, which we'll talk about in the next video, to try to reduce overfitting.

And then also, again, sometimes you just have to try it.

But if you can find a more appropriate neural network architecture, sometimes that can reduce your variance problem as well, as well as reduce your bias problem. But how to do that?

It's harder to be totally systematic how you do that.

But so I try these things and I kind of keep going back, until hopefully you find something with both low bias and low variance, whereupon you would be done.

(DESCRIPTION)

Flow chart. If there is high bias, go to a bigger training network. Increase the size until bias is low. If there is high variance after that, perform data regularization, then go back to step one, checking bias. Repeat whole process until there is neither high bias nor high variance

(SPEECH)

So a couple of points to notice.

First is that, depending on whether you have high bias or high variance, the set of things you should try could be quite different.

So I'll usually use the training dev set to try to diagnose if you have a bias or variance problem, and then use that to select the appropriate subset of things to try.

So for example, if you actually have a high bias problem, getting more training data is actually not going to help.

Or at least it's not the most efficient thing to do.

So being clear on how much of a bias problem or variance problem or both can help you focus on selecting the most useful things to try.

Second, in the earlier era of machine learning, there used to be a lot of discussion on what is called the bias variance tradeoff.

And the reason for that was that, for a lot of the things you could try, you could increase bias and reduce variance, or reduce bias and increase variance.

But back in the pre-deep learning era, we didn't have many tools, we didn't have as many tools that just reduce bias or that just reduce variance without hurting the other one.

But in the modern deep learning, big data era, so long as you can keep training a bigger network, and so long as you can keep getting more data, which isn't always the case for either of these, but if that's the case, then getting a bigger network almost always just reduces your bias without necessarily hurting your variance, so long as you regularize appropriately.

And getting more data pretty much always reduces your variance and doesn't hurt your bias much.

So what's really happened is that, with these two steps, the ability to train, pick a network, or get more data, we now have tools to drive down bias and just drive down bias, or drive down variance and just drive down variance, without really hurting the other thing that much.

And I think this has been one of the big reasons that deep learning has been so useful for supervised learning, that there's much less of this tradeoff where you have to carefully balance bias and variance, but sometimes you just have more options for reducing bias or reducing variance without necessarily increasing the other one.

And, in fact, [inaudible] you have a well regularized network.

We'll talk about regularization starting from the next video.

Training a bigger network almost never hurts.

And the main cost of training a neural network that's too big is just computational time, so long as you're regularizing.

So I hope this gives you a sense of the basic structure of how to organize your machine learning problem to diagnose bias and variance, and then try to select the right operation for you to make progress on your problem.

One of the things I mentioned several times in the video is regularization, is a very useful technique for reducing variance.

There is a little bit of a bias variance tradeoff when you use regularization.

It might increase the bias a little bit, although often not too much if you have a huge enough network.

But let's dive into more details in the next video so you can better understand how to apply regularization to your neural network.