Train your first neural network [Day 12 of 17]

"Adrian at PyImageSearch" <adrian@pyimagesearch.com>
收件人:navicester@163.com
时　间:2021-2-26 11:00:00
附　件:

Admit it —

You've been dreaming about building your own neural networks ever since you started the crash course, right?

Well, today, you're going to take a BIG leap towards that goal. You're going to build your first neural network using *one of my favorite* Python libraries: Keras.

Keras makes it super easy to get from *idea* to *result* in as few lines of code as possible — and that's what the PyImageSearch blog is *all about.*

One of the big reasons I love sharing solutions like this is that I know firsthand what it's like to work *without* having access to such powerful libraries.

The first time I experimented with neural networks — around 13 years ago — I was trying to do something very similar to what you'll do today.

But back then I had to do it in Java, not Python, which made the entire process significantly harder (and more arduous), and back in those dark days, there were no libraries like Keras available.

Implementing and training that first network took me *days*. You're going to do the *same* work *in less than half an hour.*

Today's neural network is a standard feedforward network, so data only propagates forward, without any cycles or loops. We'll use our neural network to classify a given image as a *dog* or a *cat* — plus you'll start to understand *why* a neural network makes classification mistakes and *what* you can do to improve those results.

Sure, it's a bit basic, but today's lesson is *critical* for what we'll be doing over the next few days, and it won't be long before you're creating your own custom datasets and training networks to recognize your images.

If you're ready to build your first neural network, let's get started!

Adrian Rosebrock
Chief PyImageSearcher
