

Forum:[Hack Forums](#)

Thread: [How good is a Neural Network? \[Human vs AI\]](#)

TheBarret - September 20, 2014, 1:14 pm

Ever wondered how good a well trained neural network is?

Let's put it to the test, Human vs Computer, i'll show you a few screenshots without and with an spoiler tag awnser.

You make a guess what letter is in the distorted green box and i'll add a spoiler underneath with the Network match result.

Alright here we go:

Awnsers:

Spoiler ([Click to View](#))

Now ofc this is done on a simple and small scale, but its quit capable of extracting distorted letters and numbers or even images itself with the right dimensions or font.

I hope to inspire you to get more involed with this technique wich btw is considered AI if you might wonder.

Wiki: AI

Quote:In machine learning and related fields, artificial neural networks (ANNs) are computational models inspired by an animal's central nervous systems (in particular the brain) which is capable of machine learning as well as pattern recognition. Artificial neural networks are generally presented as systems of interconnected "neurons" which can compute values from inputs.

Source download: [Link](#)

Odus - September 20, 2014, 1:18 pm

This'll be turned into a captcha solver and up in the marketplace before you know it. I'm loving this AI stuff though, keep it coming!

TheBarret - September 20, 2014, 1:21 pm

(Sep 20, 2014 01:18 PM)Odus Wrote: [CITE](#) This'll be turned into a captcha solver and up in the marketplace before you know it. I'm loving this AI stuff though, keep it coming!

Between you and me? ive already tried all that, works quit well ^^ never released it, now u might go "WUUUT???" exactly that ? makes me giggle when ppl go nuts if u have a perfect solution, its a like swinging a cookie close enough to smell but cannot reach to eat it ? (im rly an asshole i know)

+Haze - September 20, 2014, 1:36 pm

(Sep 20, 2014 01:18 PM)Odus Wrote: [\[CITE\]](#) This'll be turned into a captcha solver and up in the marketplace before you know it. I'm loving this AI stuff though, keep it coming!
I doubt it. I don't think most people would be able to even understand what this done. I'm stumped lol. I guess I need more experience. But yet again another nice release Barret. f0kin hell mate this is actually really good.

Your releases really show what vb.net can be capable of doing. This will help the reputation of vb to to others

TheBarret - September 20, 2014, 3:55 pm

If you start to understand the fundamentals, you can try to add more options like images.

Important remark:

Code Const DEF_CANVAS_SIZE As Integer = &H18

This defines how large the so called visual range is of ur network matrix.

So images of 32x32 or 64x64, then 32 or 64 is your size, this cannot be mixed.

SIM-1847363 - September 20, 2014, 7:12 pm

Dang, i love this types of things, and when you know its from TheBarret its good! Nice work keep it up!

TheBarret - September 21, 2014, 9:51 am

(Sep 20, 2014 07:12 PM)Nettro Wrote: [\[CITE\]](#) Dang, i love this types of things, and when you know its from TheBarret its good! Nice work keep it up!

I never seen someone around here actually make anything like this?

is my demo app not clear enough, i wonder. ?

Dialog - September 21, 2014, 10:10 am

The things you produce are amazing... 99% of the people here (including me) can only dream about pulling of things like these ?

DisguiserNET - September 21, 2014, 1:54 pm

I tried coding something similar but failed most the time on the actual training to make it accurate enough.

Could you teach me more about the amount of test data etc?

TheBarret - September 21, 2014, 2:05 pm

(Sep 21, 2014 01:54 PM)aw3s0m3 Wrote: [\[CITE\]](#) I tried coding something similar but failed most the time on the actual training to make it accurate enough.

Could you teach me more about the amount of test data etc?

the source is included, and for detail i could explain all that but wiki is faster ^^

<http://en.wikipedia.org/wiki/Backpropagation>

Quote:Backpropagation, an abbreviation for "backward propagation of errors", is a common method of training artificial neural networks used in conjunction with an optimization method such as gradient descent. The method calculates the gradient of a loss function with respects to all the weights in the network. The gradient is fed to the optimization method which in turn uses it to update the weights, in an attempt to minimize the loss function.

