Quantum neural network

M.V.Altaisky

Joint Institute for Nuclear Research, Dubna, 141980, RUSSIA;

Space Research Institute RAS, Profsoyuznaya 84/32, Moscow, 117997, RUSSIA

Abstract

It is suggested that a quantum neural network (QNN), a type of artificial neural network, can be built using the principles of quantum information processing. The input and output qubits in the QNN can be implemented by optical modes with different polarization, the weights of the QNN can be implemented by optical beam splitters and phase shifters.

Since it was first proposed by Feynman [1], that quantum mechanics might be more powerful computationally than a classical Turing machine, we have heard a lot of quantum computational networks [3], quantum cellular automata [2], but only a little about quantum neural networks [4]. The possible reason for the omni-penetrating ideas of quantum information processing (QIP) to avoid the field of artificial neural networks (ANN), is the presence of a nonlinear activation function in any ANN. For very similar reason, a need for nonlinear couplings between optical modes was the main obstacle for building a scalable optical QIP system.

It was shown recently [5], that quantum computation on optical modes using only beam splitters, phase shifters, photon sources and photo detectors is possible. Accepting the ideas of [5], we just assume the existence of a qubit

$$|x\rangle = \alpha|0\rangle + \beta|1\rangle, \tag{1}$$

where $|\alpha|^2 + |\beta|^2 = 1$, with the states $|0\rangle$ and $|1\rangle$ are understood as different polarization states of light.

Let us consider a *perceptron*, *i.e.* the system with n input channels $x_1, \ldots, x_n$ and one output channel y. The output of a classical perceptron [6] is

$$y = f(\sum_{j=1}^{n} w_j x_j), \qquad (2)$$

where $f(\cdot)$ is the perceptron activation function and w_j are the weights tuning during learning process.

The perceptron learning algorithm works as follows:

1. The weights w_j are initialized to small random numbers.

2. A *pattern* vector $(x_1, \ldots, x_n)$ is presented to the perceptron and the output y generated according to the rule (2)

3. The weights are updated according to the rule

$$w_j(t+1) = w_j(t) + \eta(d - y)x_j, \qquad (3)$$

   where t is discreet time, d is the desired output provided for teaching and $0 < \eta < 1$ is the step size.

It will be hardly possible to construct an exact analog of the nonlinear activation function f, like sigmoid and other functions of common use in neural networks, but we will show that the leaning rule of the type (3) is possible for a quantum system too.

Let us consider a quantum system with n inputs $|x_1\rangle, \ldots, |x_n\rangle$ of the form (1), and the output $|y\rangle$ derived by the rule

$$|y\rangle = \hat{F} \sum_{j=1}^{n} \hat{w}_j |x_j\rangle, \qquad (4)$$

where $\hat{w}_j$ become 2×2 matrices acting on the basis $(|0\rangle, |1\rangle)$, combined of phase shifters $e^{i\theta}$ and beam splitters, and possibly light attenuators, (cf. eq.1 from [5]); $\hat{F}$ is an unknown operator that can be implemented by the network of quantum gates.

Let us consider the simplistic case with $\hat{F} = 1$ being the identity operator. The output of the quantum perceptron at the time t will be

$$|y(t)\rangle = \sum_{j=1}^{n} \hat{w}_j(t)|x_j\rangle. \qquad (5)$$

In analogy with classical case (3), let us provide a learning rule

$$\hat{w}_j(t+1) = \hat{w}_j(t) + \eta(|d\rangle - |y(t)\rangle)\langle x_j|, \tag{6}$$

where $|d\rangle$ is the desired output.

It is easy to show now, that the learning rule (6) drives the quantum perceptron into desired state $|d\rangle$ used for teachning. In fact , using the rule (6) and taking the module-square difference of the real and desired outputs, we yield

$$\||d\rangle - |y(t+1)\rangle\|^2 = \||d\rangle - \sum_{j=1}^{n} \hat{w}_j(t+1)|x_j\rangle\|^2 = (1 - n\eta)^2\||d\rangle - |y(t)\rangle\|^2 \tag{7}$$

For small η ($0 < \eta < 1/n$) and normalized input states $\langle x_j|x_j\rangle = 1$ the result of iteration converges to the desired state $|d\rangle$. The whole network can be then composed from the primitive elements usinng the standard rules of ANN architecture.

The learning rule (6) may cause questions, for it does not observe the unitarity in general (in contrust, say, to the ideas proposed in [4]), and thus not only phase rotations but also light attenuation may take place. This a point for future consideration: formally we can observe the unitarity by substituting the learning rule (6) by one acting on unit circle basis $e^{im\theta}$, but the model with attenuation of certain weight in QNN seems more reasonable for learn-ability of the network, and more perspective for simmulation on classical computer.

The idea, this note was inspired by, was to suggest a neural network model, which takes into account the phase of the signal, rather than only the amplitude, as existing ANN. The idea seems to be very close no real biological neural nets, where the neurons are sensitive to the phase of the signal rather than the amplitude alone. In the later case, the matrix weights $\hat{w}_j$ could be understood as complex impedances that attenuate the signal and change its phase.

<center>***</center>

References

[1] Feynman, R.P. *Simulating physics with computers*, Int. J.Theor.Phys. **21**(1982)467-488.

[2] Feynman, R.P. *Quantum Mechanical Computers*, Found. Phys. **16**(1986)507-531.

[3] Deutch, D. *Quantum computational networks*, Proc. Roy. Soc. Lond. **A439**(1992)553-558.

[4] Vlasov, A.Yu. *Quantum computations and images recognition*, quant-ph/9703010; *Analogues quantum computers for data analysis* quant-ph/9802028.

[5] E.Knill, R.Laflamme and G.J.Milburn. *A scheme for efficient quantum computation with linear optics*, Nature **409**(2001)46-57.

[6] M.Minsky and P.Papert, *Perceptrons: An Introduction to Computational Geometry*, MIT Press, Cambridge, Mass. 1969.