Implementing Competitive Learning in a Quantum System

Dan Ventura
fonix corporation
dventura@fonix.com
http://axon.cs.byu.edu/Dan

Abstract

Ideas from quantum computation are applied to the field of neural networks to produce competitive learning in a quantum system. The resulting quantum competitive learner has a prototype storage capacity that is exponentially greater than that of its classical counterpart. Further, empirical results from simulation of the quantum competitive learning system on real-world data sets demonstrate the quantum system's potential for excellent performance.

Introduction

Competitive learning networks perform classification tasks by comparing input patterns with pattern prototypes encoded in the network weights [5] [8]. The basic idea is that the prototype that most closely resembles the input pattern (according to some metric) is determined, and the class associated with that prototype is output as the class of the input pattern (see Figure 1).

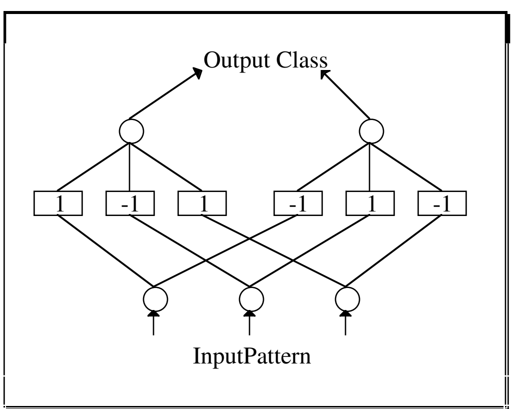

Figure 1. Classical Hamming net storing two patterns

The rapidly developing field of quantum computation has produced some fascinating results that are not possible using classical computational methods. The possibility of applying quantum computational ideas to the field of neural networks is an intriguing one and is beginning to be explored [1] [7] [12]. This paper investigates the possibility of developing a quantum analog of a classical competitive learning network.

The quantum competitive learner developed here makes use of the unique characteristics of quantum mechanical systems to achieve a prototype storage capacity that exponentially exceeds that of classical competitive networks. The result is a quantum competitive learning system that is based on the Hamming distance metric.

The rest of the paper introduces some key concepts from quantum mechanics, briefly discusses some of the well-known quantum algorithms, and then details a quantum version of competitive learning. Preliminary empirical results (obtained through simulation on a classical computer) are presented, and these results demonstrate that a quantum competitive learning system is indeed capable of performance that is impossible with a classical system. Some discussion of the intuition behind the algorithm is provided and directions for further work are discussed.

Quantum Concepts

Quantum computation is based upon physical principles from the theory of quantum mechanics, which is in many ways counterintuitive. Yet it has provided us with perhaps the most accurate physical theory ever devised. The theory is well-established and is covered in its basic form by many textbooks (see for example [3]). Several ideas are briefly reviewed here.

Linear superposition is closely related to the familiar mathematical principle of linear combination of vectors. Quantum systems are described by a wave function ψ that exists in a Hilbert space. The Hilbert space has a set of states, $|\phi_i\rangle$, that form a basis, and the system is described by a quantum state,

$$|\psi\rangle = \sum_i c_i |\phi_i\rangle \quad (1)$$

$|\psi\rangle$ is said to be in a linear superposition of the basis states $|\phi_i\rangle$, and in the general case, the coefficients c_i may be complex. Use is made here of the Dirac bracket notation, where the ket $|\cdot\rangle$ is analogous to a column vector, and the bra $\langle\cdot|$ is analogous to the complex conjugate transpose of the ket.

Coherence and *decoherence* are closely related to the idea of linear superposition. A quantum system is said to be coherent if it is in a linear superposition of its basis states. According to quantum mechanics, if a coherent system interacts in any way with its environment, the superposition is destroyed. This loss of coherence is called decoherence and is governed by the wave function ψ. The coefficients c_i are called probability amplitudes, and $|c_i|^2$ gives the probability of $|\psi\rangle$ collapsing into state $|\phi_i\rangle$ if it decoheres. Note that the wave function ψ describes a real physical system that must collapse to exactly one basis state. Therefore, the probabilities governed by the amplitudes c_i must sum to unity. This necessary constraint is expressed as the unitarity condition

$$\sum_i |c_i|^2 = 1 \tag{2}$$

Consider, for example, a discrete physical variable called spin. The simplest spin system is a two-state system, called a spin-1/2 system, whose basis states are usually represented as $|\uparrow\rangle$ (spin up) and $|\downarrow\rangle$ (spin down). In this system the wave function ψ is a distribution over two values and a coherent state $|\psi\rangle$ is a linear superposition of $|\uparrow\rangle$ and $|\downarrow\rangle$. One such state might be

$$|\psi\rangle = \frac{2}{\sqrt{5}}|\uparrow\rangle + \frac{1}{\sqrt{5}}|\downarrow\rangle \tag{3}$$

As long as the system maintains its quantum coherence it cannot be said to be either spin up or spin down. It is in some sense both at once. When this system decoheres the result is, for example, the $|\uparrow\rangle$ state with probability $(2/\sqrt{5})^2 = 0.8$.

A simple two-state quantum system, such as the spin-1/2 system just introduced, is used as the basic unit of quantum computation. Such a system is referred to as a quantum bit or *qubit*, and renaming the two states $|0\rangle$ and $|1\rangle$, it is easy to see why this is so.

Operators on a Hilbert space describe how one wave function is changed into another. Here they will be denoted by a capital letter with a hat, such as $\hat{A}$, and they may be represented as matrices acting on vectors. Using operators, an eigenvalue equation can be written $\hat{A}|\phi_i\rangle = a_i|\phi_i\rangle$, where a_i is the eigenvalue. The solutions $|\phi_i\rangle$ to such an equation are called eigenstates and can be used to construct the basis of a Hilbert space as discussed above. In the quantum formalism, all properties are represented as operators whose eigenstates are the basis for the Hilbert space associated with that property and whose eigenvalues are the quantum allowed values for that property. Operators in quantum mechanics must be linear and further, operators that describe the time evolution of a state must be unitary so that $\hat{A}^\dagger\hat{A} = \hat{A}\hat{A}^\dagger = \hat{I}$, where $\hat{I}$ is the identity operator, and $\hat{A}^\dagger$ is the complex conjugate transpose of $\hat{A}$.

Interference is a familiar wave phenomenon. Wave peaks that are in phase interfere constructively while those that are out of phase interfere destructively. This phenomenon is common to all kinds of wave mechanics from water waves to optics, and the well-known double slit experiment proves empirically that interference also applies to the probability waves of quantum mechanics.

Quantum Algorithms

The field of quantum computation offers exciting possibilities -- the most important quantum algorithms discovered to date all perform tasks for which there are no classical equivalents. For example, Deutsch's algorithm [2] is designed to solve the problem of identifying whether a binary function is constant (function values are either all 1 or all 0) or balanced (the function takes an equal number of 0 and 1 values). Deutsch's algorithm accomplishes the task in order $O(1)$ time, while classical methods require $O(2^n)$ time. Simon's algorithm [10] is constructed for finding the periodicity in a 2-1 binary function that is guaranteed to possess a periodic element. Here again an exponential speedup is achieved; however, admittedly, both these algorithms have been designed for artificial, somewhat contrived problems as a proof of concept. Grover's algorithm [4], on the other hand, provides a method for searching an unordered quantum database in time $O(\sqrt{2^n})$, compared to the classical lower bound of $O(2^n)$. Here is a real-world problem for which quantum computation provides performance that is classically impossible. Finally, the most well-known and perhaps the most important quantum algorithm discovered so far is Shor's algorithm for prime factorization [9]. This algorithm finds the prime factors of very large numbers in polynomial time, while the best classical algorithms require exponential time.

Quantum Competitive Learning

A quantum algorithm is represented as a series of operators operating on a quantum system beginning from some initial state. Here the quantum system we will consider is a set of $n+q$ qubits, corresponding to binary input patterns of length n and binary patterns of length q that represent corresponding output classes. The system's initial state will be $|\bar{0}\rangle$, the single basis state corresponding to all the qubits being in the $|0\rangle$ state.

There are two major parts to the classical algorithm -- initializing the network weights and pattern classification -- and both can be implemented using quantum operators. First, the information carried in the weights of a classical system is represented in the quantum wave function's coefficients. Therefore, the quantum analog to initializing the network's weights is the time evolution of the quantum wave function such that the basis states with non-zero coefficients correspond to the prototypes to be stored in the system. In other words, given a set $\wp$ of prototype patterns, we require a wave function $|\psi\rangle$ such that

$$|\psi\rangle = \frac{1}{\sqrt{m}} \sum_{p \in \wp} |p\rangle \tag{4}$$

where m is the number of patterns in the set $\wp$. This initialization of a quantum state is a process that is beyond the scope of this paper. However, an efficient quantum algorithm for doing something slightly more general is detailed in [11], and a slight modification presented in [12] accomplishes this specific initialization. Here we represent this entire initialization process as the single operator $\hat{P}$,

$$\hat{P}|\bar{0}\rangle = |\psi\rangle = \frac{1}{\sqrt{m}} \sum_{p \in \wp} |p\rangle \tag{5}$$

Less formally, the $\hat{P}$ operator transforms the single basis state $|\bar{0}\rangle$ state to the desired wave function $|\psi\rangle$ which represents all of the prototypes in the set $\wp$.

The second part of the algorithm, pattern classification, is accomplished with a combination of two operators, which will be represented here as matrices indexed by row and column, corresponding to the system basis states. Given a test pattern z of length n, the first of these operators, $\hat{R}$, is defined as

$$\hat{R} = r_{ij} = \begin{cases} 1 & \text{if } i = j,\ h(z, i^n) \geq n - \alpha \\ -1 & \text{if } h(z, i^n) \geq n - \alpha,\ h(z, j^n) \geq n - \alpha, \\ & h(z, i^n) \geq h(z, j^n) \\ 0 & \text{otherwise} \end{cases} \tag{6}$$

where n is the number of binary inputs, i^n is represents the first n bits of i, h returns the Hamming distance between two binary patterns, and α is a threshold indicating how close potential patterns must be to z in order to compete. The non-zero diagonal entries in the matrix indicate which states can compete within the restrictions of the threshold α. The non-zero off-diagonal elements represent extra inhibition for patterns with lower Hamming distance from z over those with higher Hamming distance from z.

The second operator, $\hat{T}$, is defined as

$$\hat{T} = t_{ij} = \begin{cases} 1 & \text{if } i = j \text{ and } \exists p \begin{pmatrix} p \in \wp,\ p \equiv x \to c, \\ h(x, i^n) = n \end{pmatrix} \\ 0 & \text{otherwise} \end{cases} \tag{7}$$

This operator will have non-zero entries only on the diagonal and only corresponding to prototypes. This operator ensures that some legal prototype is the winner of the competition. Now taking $|\psi\rangle$ obtained from Equation (5), the classification phase of the algorithm can be written

$$|\psi\rangle = \hat{T}\hat{R}|\psi\rangle \xrightarrow{collapse} |p\rangle \tag{8}$$

with the collapse due to observation of the quantum system. Recall that operator precedence is right to left and since the $\hat{R}$ and $\hat{T}$ operators do not commute, their order is important. When the system is observed, it will collapse into one of the basis states corresponding to some prototype $p \equiv x \to c$, whose output class c will be taken as the classification of the input pattern z.

To express the dynamics of prototype competition informally, we can say that competition between prototype patterns occurs through the interference in the wave function induced by the operators and ultimately through decoherence and wave function collapse. The quantum algorithm can be summarized as in Figure 2.

1. $|\psi\rangle = \hat{P}|\bar{0}\rangle$
2. $|\psi\rangle = \hat{T}\hat{R}|\psi\rangle$
3. observe $|\psi\rangle$

Figure 2. Algorithm for Quantum Competitive Learning

At this point, a simple example will help clarify much of the preceding discussion. Suppose our set $\wp$ of prototype patterns is defined as

$$\wp = \begin{Bmatrix} 10 \to 1 \\ 11 \to 0 \end{Bmatrix} \tag{9}$$

so that n = 2, and suppose we would like to classify the input pattern z = 00. Step one of the algorithm, using Equation (5), will result in

$$\hat{P}\left|\bar{0}\right\rangle = |\,\psi\rangle = \frac{1}{\sqrt{2}}|101\rangle + \frac{1}{\sqrt{2}}|110\rangle \tag{10}$$

For convenience, this may be rewritten in vector notation as

$$|\,\psi\rangle = (0,0,0,0,0,1,1,0)^T \tag{11}$$

Where T indicates the vector transpose, and the vector entries correspond to wave function coefficients for the basis states $|000\rangle, |001\rangle \cdots |111\rangle$. Using α = 1, calculating the $\hat{R}$ operator gives

$$\hat{R} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & -1 & 1 & 0 & 0 & 0 & 0 & 0 \\ -1 & -1 & 0 & 1 & 0 & 0 & 0 & 0 \\ -1 & -1 & 0 & 0 & 1 & 0 & 0 & 0 \\ -1 & -1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{12}$$

where rows and columns of the matrix are indexed in the same manner as the vector of Equation (11). Now, the first part of step 2 (Equation (8)) of the algorithm applies $\hat{R}$ to the wave function obtained in Equation (10). The resulting wave function has non-zero coefficients for those basis states representing candidate patterns in the competition.

$$|\,\psi\rangle = \hat{R}|\,\psi\rangle = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & -1 & 1 & 0 & 0 & 0 & 0 & 0 \\ -1 & -1 & 0 & 1 & 0 & 0 & 0 & 0 \\ -1 & -1 & 0 & 0 & 1 & 0 & 0 & 0 \\ -1 & -1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} \tag{13}$$

Which is equivalent to the wave function notation

$$|\,\psi\rangle = |101\rangle \tag{14}$$

In this simple example, there is only one competing pattern, so we are in essence done. However, in general there will be multiple competing patterns, and some of those patterns may be spurious in the sense that they do not occur in the original set $\wp$. The purpose of the $\hat{T}$ operator is to remove these spurious patterns from the competition. Calculating the $\hat{T}$ operator in this case gives

$$\hat{T} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{15}$$

The second half of step 2 of the algorithm applies this operator to the wave function from Equation (13) resulting in

$$|\,\psi\rangle = \hat{T}|\,\psi\rangle = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} \tag{16}$$

which in this case does not change the wave function. In other words, we still have

$$|\,\psi\rangle = |101\rangle \tag{17}$$

The final step of the algorithm, observing the system, will result in the system's collapsing to the basis state $|101\rangle$. Examining the third qubit will give us the output class 1 so that we may now classify the pattern z as $00 \to 1$. Notice that examining the first two qubits gives the prototype chosen as the closest match to z and therefore provides some justification for the chosen classification.

Results

Since quantum system dynamics can be simulated on a classical computer at the cost of an exponential slowdown, a simulation has been developed for the quantum competitive learning system. Several real-world problems of modest size were obtained from the UCI machine learning data repository [6] and were used to test the pattern recall and classification accuracy of the quantum competitive learning system.

Table 1 reports results for simulations run using three real-world data sets. The first column indicates the number of quantum neurons used to store the competing prototype patterns, and the second indicates the number of competing patterns stored by these neurons. The third column indicates the number of output classes associated with the problem. The fourth and fifth columns report accuracies on classifying patterns from a training set and test set respectively. All classification results were obtained using ten-fold cross validation, and because quantum systems are probabilistic in nature, each experiment was repeated ten times with the results averaged to obtain a more accurate representation. Notice that the storage ability (number of prototype patterns) of the quantum competitive learner is vastly superior to that possible classically. It should be noted that training set recall is not perfect for the *LED* and *Hayes* data sets due to the fact that these datasets contain inconsistencies (identical patterns with conflicting output classes). Further improvement in test set accuracy is possible by making use of weighted features rather than the simple Hamming distance metric. For example, using just a simple mapping of the last 2 features in the *Lenses* dataset boosts test set accuracy to 0.88.

Table 1. Classification Accuracy

	#Neurons	#Patterns	#Classes	Train	Test
Hayes:	10	83	3	0.91	0.46
LED:	11	180	10	0.75	0.56
Lenses:	7	23	3	1.00	0.62

Conclusions

Ideas from classical neural network theory are recast in a quantum computational framework, using the language of wave functions and operators. The unique characteristics of quantum systems are utilized to produce a quantum competitive learning network capable of storing exponentially more prototype patterns than is possible classically. This demonstrates that quantum computational ideas can be combined with concepts from the field of neural networks to produce useful and interesting results. Simulations using real-world data show that the quantum competitive learner performs very favorably during pattern recall.

Ongoing work is focused on discovering new operators to improve performance by weighted feature discovery. Future work includes searching for further applications of quantum computation to neural networks and generally further developing the field of quantum computational learning.

References

[1] Behrman, E., J. Niemel, J. Steck and S. Skinner, "A Quantum Dot Neural Network", *Proceedings of the Workshop on Physics of Computation*, pp. 22-24, 1996.

[2] Deutsch, D. and R. Jozsa, "Rapid Solution of Problems by Quantum Computation", *Proceedings of the Royal Society, London A*, v439, pp. 553-558, 1992.

[3] Feynman, Richard, R. Leighton and M. Sands, *The Feynman Lectures on Physics*, v3, Addison-Wesley, 1965.

[4] Grover, Lov, "A Fast Quantum Mechanical Algorithm for Database Search", *Proceedings of the ACM Symposium on the Theory of Computing*, pp. 212-19, 1996.

[5] Lippman, R.P., "An Introduction to Computing with Neural Nets", *IEEE ASSP Magazine*, v4, pp. 4-22, 1987.

[6] Mertz, C. J. and P. M. Murphy, *UCI Repository of Machine Learning Databases*, University of California at Irvine, Department of Information and Computer Science, http://www.ics.uci.edu/~mlearn/MLRepository.html, 1996.

[7] Perus, Mitja, "Neuro-Quantum Parallelism in Brain-Mind and Computers", *Informatica*, v20, pp. 173-183, 1996.

[8] Rumelhart, D.E and D. Zipser, "Feature Discovery by Competitive Learning", *Cognitive Science*, v9, p. 75-112, 1985.

[9] Shor, Peter, "Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer", *SIAM Journal of Computing*, v26, pp. 1484-1509, 1997.

[10] Simon, D., "On the Power of Quantum Computation", *SIAM Journal of Computation*, v26, pp. 1474-83, 1997.

[11] Ventura, Dan and Tony Martinez, "Initializing the Amplitude Distribution of a Quantum State", submitted to *Physical Review A*, May 1999.

[12] Ventura, Dan and Tony Martinez, "Quantum Associative Memory", *Information Sciences*, to appear, 1999.