一种高速神经网络HS-K-WTA的研究

朱 红[1], 陈清华[2], 刘国岁[1]

(1. 南京理工大学电光院, 江苏南京 210094; 2. 南京理工大学计算机系, 江苏南京 210094)

摘 要: 本文提出一种新的K-Winners-Take-All神经网络: High-Speed-K-Winners-Take-All(HS-K-WTA). HS-K-WTA是以竞争学习算法为基础. HS-K-WTA能够从任何一个数集中, 识别出K个较大的数, 或较小的数. 本文给出HS-K-WTA算法及算法复杂性的分析结果. 结果显示HS-K-WTA要比Winstrons更好, 更容易硬件实现, 更稳定, 尤其所取的数集中的数较大时.

关键词: 神经网络; 竞争学习算法; 高速算法; 选择K个较大数

中图分类号: TP183 文献标识码: A 文章编号: 0372-2112 (2002) 07-1020-03

A High-Speed Neural Network HS-K-WTA

ZHU Hong[1], CHEN Qing-hua[2], LIU Guo-sui[1]

(1. *College of Electronics and Optics, Nanjing University of Science and Technology, Nanjing, Jiangsu 210094, China;*

2. Dept of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing, Jiangsu 210094, China)

Abstract: A new K-Winners-Take-All neural network: High-Speed-Winners-Take-All(HS-K-WTA) is presented. HS-K-WTA can identify the k larger elements(or k smaller ones) in a data set. The analysis results about HS-k-WTA algorithm and its complexity are given. The results show that the speed, the hardware realization, and the stability of HS-K-WTA are much better than Winstrons, especially for a lot of atoms of a data set.

Key words: neural network; competitive learning algorithm; high speed algorithm; K-WTA; HS-K-WTA

1 引言

Winners-Take-All以及K-Winners-Take-All(K-WTA)神经网络在最近的文章中被广泛地讨论[1~6]. Jui-Cheng Yen等提出的Winstron是一种新的K-WTA[1]. 关于K-WTA的定义由Perfetti[2]中给出. Winstron是一种将SELECTRON[6]一般化的神经网络. Winstron的设计是基于如下假设: 人脑的部分是自组织执行模式识别. 这种自组织当神经元相互竞争时发生. 在竞争中, 竞争力弱的神经元逐渐地退出, 竞争力强的神经元参与进一步竞争. 最终, 最富竞争力的神经元将赢得竞争. 基于以上的思想, Winstron确能选择出k个较大的元素, 或者k个较小元素. Yen等[1]已经提出Winstron以及其阵列结构的实现. 然而Winstron的不足就在于其实现的复杂性.

因此, 我们基于Winstron的思想, 提出一种高速低实现复杂性的K-WTA神经网络, 称之为High-Speed-k-Winners-Take-All(HS-K-WTA).

2 HS-K-WTA的模型

我们采用Yen等在文[1]中给出的定义1-9和注记, 另外的定义和约定如下:

定义10 $x = sign(x) \times \sum_{i=-(n-1)}^{\infty} g(x_{-i}) \times 2^{-i} \quad \forall x \in R$,

$$sign(x) = \begin{cases} 1, & x \geqslant 0 \\ 0, & x < 0 \end{cases}, \quad g(x_{-i}) = \begin{cases} x_{-i} & \text{当 } sign(x) = 1 \text{ 时} \\ \overline{x_{-i}} & \text{当 } sign(x) = 0 \text{ 时} \end{cases}$$

$\overline{x_{-i}}$表示取x_{-i}的反码; $g(x_{-i}) \in B_1$, $i = -(n-1), \dots, 0, 1, 2, \dots$; $B_1 = \{0, 1\}$.

HS-K-WTA算法如下:

步骤1: 设置下列初始值: $t = 0$, $S(0) = N$ 且对于$j \in \mathfrak{R}$ $\Theta(0) = \Phi$, $\Xi(0) = \mathfrak{R}$, $\Gamma(0) = \mathfrak{R}$

$$w_{ij} = \begin{cases} \dfrac{1}{2^{n_0 - t}} & \text{如 } i = j \\ 0 & \text{如 } i \neq j \end{cases}, \quad i, j \in \mathfrak{R}$$

步骤2 设置k值, 提供数集$\tilde{S}$, 这里有$k < |\tilde{S}|$; 如$k = |\tilde{S}|$, 则结束.

步骤3 设$t = t + 1$.

步骤4 在所有的节点之间竞争, 计算$y_j(t)$如下:

$$y_j(t) = f\left(\sum_{i=1}^{N} w_{ij}(t-1) \times x_i + I_j(t)\right) \text{ 对于 } j \in \mathfrak{R}$$

$$f(x) = \begin{cases} 1, & x \geqslant 1 \\ 0, & \text{其他} \end{cases}$$

步骤5 $S(t) = \sum_{j=1}^{N} y_j(t)$; 如$S(t) = k$, 则本算法收敛并停止.

如果$s(t) > k$则设置: $\Gamma(t) = \{j \mid y_j(t) = 1\}$; $\Theta(t) = \Theta(t-1)$

$$w_{ij} = \begin{cases} \dfrac{1}{2^{n_0 - t}}, & \text{如果 } i, j \in \Gamma(t) \\ 0, & \text{其他} \end{cases}$$

$$\Xi(t) = \Gamma(t) - \Theta(t), \quad \Psi(t) = \mathfrak{R} - \Gamma(t);$$

如果$S(t) < k$则设置 $\Theta(t) = \left\{j \mid y_j(t) = 1\right\}$ $\Gamma(t) = \Gamma(t-1) - \Theta(t)$

步骤6 如果$\sum_{i=0}^{t-1} |\Theta(i)| + |\Theta(t)| = k$, 则结束; 在这里设$\sum_{i=0}^{t-1} |\Theta(t)| < k$.

收稿日期: 2000-10-31; 修回日期: 2001-07-18

如果 $\sum_{i=0}^{t-1}|\Theta(i)|+|\Theta(t)|>k$, 则设置 $\Gamma(t)=\Theta(t)$, $k=k-\sum_{i=0}^{t-1}|\Theta(i)|$

$$w_{ij}=\begin{cases}\frac{1}{2^{n_0-t}}, & \text{如果 } i=j, j\in\Gamma(t)\\ 0, & \text{其他}\end{cases},\quad \forall j, I_j(t)=0$$

如果 $t\neq n_0$ 则转到第 3 步继续执行.

步骤 7　(1) 如果 $t=n_0$, 并且 $\forall i$, $i\in\{0,1,2,\cdots,n_0\}$, $S(i)>k$, 则随机地从 $\Gamma(n_0)$ 中选择 k 个数, 就是所要的结果. (2) 如果 $t=n_0$, 并且 $\forall i$, $i\in\{0,1,2,\cdots,n_0\}$, 存在 $S(i)<k$, 并且 $\sum_{i=0}^{n_0}|\Theta(i)|<k$, 则要选择的 k 个数就由两部分组成, 其中一部分是 $\sum_{i=0}^{n_0}|\Theta(i)|$, 再以 $\Gamma(n_0)$ 中随机地选择出 $k-\sum_{i=0}^{n_0}|\Theta(i)|$ 个数, 即为本算法可以给出的解.

步骤 8　如选出的 k 个数中, 其符号位为 0, 则其余的位求反(还原).

从上面的算法知, 无论数集 $\tilde{S}$ 是如何给的, $k\leq|\tilde{S}|$, 则总可以利用以上算法求得 k 个最大数. 如果算法进入步骤 7 第(1)种情况, 则说明 $\tilde{S}$ 集中有多于 k 个相等的数(至少在可表示的范围内). 如果算法进入步骤 7 第(2)种情况, 则说明 $\tilde{S}$ 集中从 $t=0$ 到 $t=n_0$ 止, 尚未选择出 k 个最大数, 但在 $\Gamma(n_0)$ 中则存在一组数, 它们的前 n_0 个位置上的 0, 1 都是相同的, 即这一组数是在可表示范围内认为它们是相等的. 这样从 $\Gamma(n_0)$ 中随机选择的 $k-\sum_{i=0}^{n_0}|\Theta(i)|$ 加上已经选出的 $\sum_{i=0}^{n_0}|\Theta(i)|$ 就是要得到的 k 个数. HS-K-WTA 同样可以选择出从 $\tilde{S}$ 中, 选出 k 个较少的, 如果选择出 $|\tilde{S}|-k$ 个较大的.

HS-K-WTA 算法的基本思想与 Winstron 类似, 都来自于 Coase-Fine 竞争. 在神经网络中的每一节点都是自组织的, 并且相互之间彼此竞争, 在无序竞争中, 那些在现在竞争步骤中失败的哪些节点就从下一个竞争步骤中除去. 基于 $\Gamma(t)$, HS-K-WTA 减少了大量的运算, 不需要在每一个节点上每一次循环计算 $x_i-\frac{\sum x_i}{|\Xi(t)|}$, 而 Winstron 却必须做这样的计算. 至此, 可以说明 HS-K-WTA 要比 Winstron 好.

3　HS-K-WTA 的有关性质

(1) 如果 $y_l(t+1)=1$, 并且 $y_m(t+1)=0$, 对于 $l, m\in\Xi(t)$, 并且 $t=1,2,\cdots,n_0$, 那么 $x_1>x_m$. 设 n_0 就是 $\forall x\in R$, 将 x 表示成二进制(按定义 10) 的最大位长.

证明　因为 $l, m\in\Xi(t)$, 我们可以得到:

$$y_1(t+1)=f\left(\sum_{i=1}^{N}w_{il}(t)\times x_i+I_1(t)\right)=f\left(\frac{1}{2^{n_0-t}}\times x_1\right)$$

根据: $y_l(t+1)=1$ 以及 $f(x)$ 的定义可得:

$\frac{1}{2^{n_0-t}}\times x_l\geq 1\quad\therefore x_l\geq 2^{n_0-t}$; 同样: $y_m(t+1)=f\left(\frac{1}{2^{n_0-t}}\times x_m\right)=0$ 可以得到:

$\frac{1}{2^{n_0-t}}\times x_m\leq 1\quad\therefore x_m\leq 2^{n_0-t}$; $\therefore x_l\geq x_m$　证毕.

(2) HS-K-WTA 算法在有限时间内收敛.

证明: 由定义 10, $\forall x\in R$, x 可表示成$(x_0, x_1, \cdots, x_i, \cdots, x_{n_0-1}, x_{n_0})$, 按 HS-K-WTA 算法, 至多在 $t=n_0$ 次终止. 故 HS-K-WTA 算法在有限时间内收敛. (HS-K-WTA 满足文献[1]中的性质 2~6.)

4　HS-K-WTA 算法复杂性分析结果

在假设表示数的大小的字长给定为 n_0 时, 按定义 10 给定的二进制及符号表示法, 从$\{-R^{n_0}, 0, R^{n_0}\}$中任意选取 N 个数构成数集 $\tilde{S}$, 利用 HS-K-WTA 神经网络选出 k 个较大的数, 所要花的循环次数 t, 在均匀分布假设的条件下, 当 $N\neq k$, t 满足下式:

$$t\leq\frac{1}{4}n_0+\frac{3}{4}\log_2 N-\frac{1}{2}\log_2\left(N-k+\frac{1}{2}\right)-\frac{1}{4}\log_2\left(N-k-\frac{1}{2}\right)+2\frac{3}{4}$$

$$t\geq\frac{1}{4}n_0+\frac{3}{4}\log_2 N-\frac{1}{2}\log_2\left(N-k+\frac{1}{2}\right)-\frac{1}{4}\log_2\left(N-k-\frac{1}{2}\right)+2$$

因此有:

(1) 当 $N-k=1$ 时, 且 n_0, N 固定, HS-K-WTA 有最大循环次数, t 不小于:

$$\frac{1}{4}n_0+\frac{3}{4}\log_2 N-\frac{1}{2}\log_2\frac{3}{2}+2\frac{1}{4}$$

(2) 当 $k=1$ 时, 且 n_0, N 固定, HS-K-WTA 有最小循环次数, t 不大于:

$$\frac{1}{4}n_0+\frac{3}{4}\log_2 N-\frac{1}{2}\log_2\left(N-\frac{1}{2}\right)-\frac{1}{4}\log_2\left(N-\frac{3}{2}\right)+2\frac{3}{4}$$

(3) 当 n_0, k 固定, N 不断增大时, 当 N 大到 $N-k+\frac{1}{2}$ 与 $N-k-\frac{1}{2}$ 与 N 接近时, HS-K-WTA 的循环次数不大于 $\frac{1}{4}n_0+2\frac{3}{4}$.

而 Winstron[1] 在平均分布假设下的循环次数是不超过 $(\log_2 N+\log_2 k+1)$, 但就 $(\log_2 N+\log_2 k+1)$ 而言, 当 N 增大时, 即使 k 不变, 其值也会变大. 再从 Winstron 的硬件实现来看, 当 N 变得较大时, 其复杂性就增加得很多, 相反 HS-K-WTA 的硬件实现更简单, 更快捷, 且稳定性将更好.

5　仿真结果

图 1 到图 4 的 Simulation 线是由程序是在假设 $n_0=32$ 条件下得到仿真结果. 其中 N 是随机选取数集中数的个数, k 是利用 HS-K-WTA 神经网络选出 k 个较大的数. max 线和 min 线是在平均分布的假设下按 HS-K-WTA 算法得到的分析结果的最大和最小循环次数; Winstron 线是按 Winstron 算法得到的分析结果. 在图 1 中, $N=5000$, 对应横坐标 k 的数值是相应的循环次数 t.

图 1 中的仿真结果 Simulation 线在 max 线和 min 线之间.

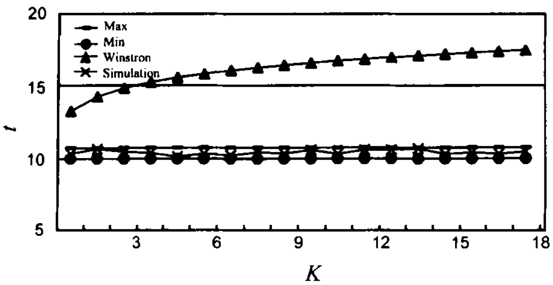

图1 $N=5000$ 仿真结果

仿真结果与分析结果是一致的.

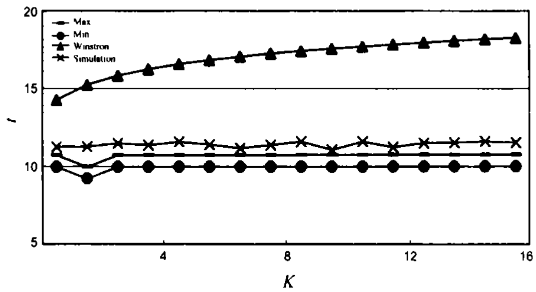

图2 $N=10000$ 仿真结果

图2中的仿真结果Simulation线在max线的上方,说明仿真结果与分析结果是有误差,但误差不大,基本一致.

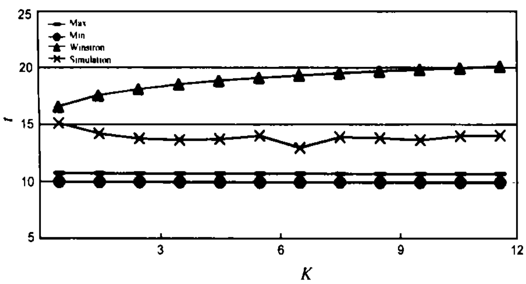

图3 $N=50000$ 仿真结果

从图3,及图4可见,随着 N 的不断增大,仿真结果Simulation线离max线越来越远,说明仿真结果与分析结果是有误差,并且误差不断变大.其中主要原因是在假设 $n_0=32$ 即 n_0 固定不变条件下得到的仿真结果.从图1,图2,图3,及图4可见,无论仿真还是分析的结果,HS-K-WTA比Winstron要好得多.

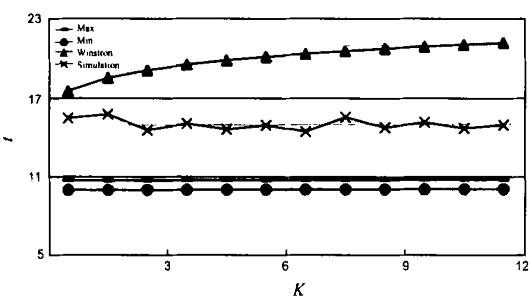

图4 $N=100000$ 仿真结果

6 结论

在文章中,我们提出HS-K-WTA神经网络,我们通过分析证实HS-K-WTA有如下优点:

(1)它能从任何数值中选出 k 个较大的数,只要对算法稍作修改,也可选出 k 个较小的数.

(2)从图1到图4可见,无论仿真还是分析的结果,HS-K-WTA比Winstron要好得多.

(3)HS-K-WTA的硬件实现也比Winstron的硬件实现要简单,且快得多.

最后,我们有理由相信,HS-K-WTA会为模式识别和模式分类,尤其是在高速情况下的模式识别和模式分类提供有效的帮助.

参考文献:

[1] Jui-Cheng Yen, Juinn-In Guo, Hun-Chen Chen. A new K-winners-take-All neural network and its array architecture [J]. IEEE Trans, 1998, NN-9(5):901-912.

[2] R Perfetti. On the robust design K-winners-take-all neural networks [J]. IEEE Trans, 1995, CS-42(1):II55-58.

[3] Jui-Cheng Yen, Juinn-In Guo, Hun-Chen Chen. K-winners-take-all circuit with O(N) complexity [J]. IEEE Trans, 1995, NN-6(6):776-778.

[4] W J Wolfe, D Mathis, C Anderson, J Rothman, M Gottler, G Brady, R Walker, G Duane, and G Alaghband. K-winner networks [J]. IEEE Trans, 1991, NN-2(2):310-315.

[5] Yuen-Hsien Tseng, and Ja-Ling Wu. On a constant-time, low-complexity winners-take-all neural network [J]. IEEE Trans, 1995, Computer-44(4):601-604.

[6] Jui-Cheng Yen, Fu-Juay Chang, Shyang Chang. A new winners-take-all architecture in artificial neural network [J]. IEEE Trans, 1994, NN-5(5):838-843.

作者简介:

朱 红 女,1966年9月生于太原,现为南京理工大学电光学院电子工程技术研究中心博士研究生,研究方向为神经网络,多媒体计算机辅助教学系统,多媒体计算机辅助学习系统.

陈清华 男,1963年10月生于盐城,现为南京理工大学计算机系博士研究生,研究方向为计算机网络及实时分布系统,神经网络,模式识别,先进实时技术.