ISSN 2354-0575

ỨNG DỤNG MẠNG NƠ RON NHÂN TẠO TRONG NHẬN DẠNG VÀ DỰ BÁO

Nguyễn Quang Hoan[1], **Lý Đông Hà**[2], **Ngô Xuân Trang**[3], **Lê Công Hiếu**[4]
1 Trường Đại học Sư phạm Kỹ thuật Hưng Yên
2 Trường Cao đẳng Kinh tế - Kỹ thuật Điện Biên
3 Sở Tài nguyên và Môi trường tỉnh Điện Biên
4 Trường Đại học Văn Lang

Ngày tòa soạn nhận được bài báo: 17/09/2017
Ngày phản biện đánh giá và sửa chữa: 10/11/2017
Ngày bài báo được chấp nhận đăng: 22/11/2017

Tóm tắt:

Bài báo ứng dụng thuật toán lan truyền ngược (BackPropagation: BP) của mạng nơ ron cho dự báo thời tiết tỉnh Điện Biên và nhận dạng biển số xe máy. Dữ liệu cho dự báo được thu thập từ Trung tâm Khí tượng Thủy văn từ năm 2015–10/2017 theo ngày (dùng cho pha học 70%, pha chạy 30% tổng dữ liệu). Dữ liệu cho bài toán nhận dạng biển số xe được lấy mẫu từ 64 tỉnh Việt Nam. Một số kết quả thử nghiệm ban đầu được trình bày; những nghiên cứu, định hướng phát triển trong tương lai được đề xuất.

Từ khóa: *Mạng nơ ron nhân tạo, dự báo thời tiết, thuật toán lan truyền ngược, nhận mẫu ký tự.*

1. Giới thiệu

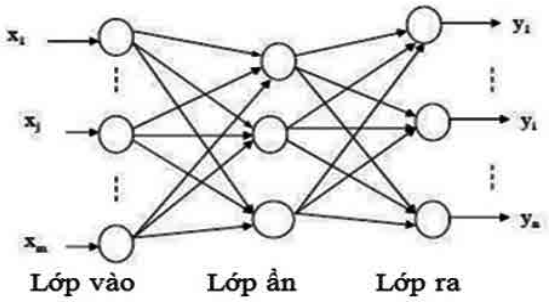
Mạng nơ ron nhân tạo là mô hình phỏng hoạt động của não người, là phương pháp tính toán mềm, chấp nhận thông tin không đầy đủ, chính xác, dễ thích nghi; có khả năng học tốt; khả năng chịu lỗi cao [6]. Rất nhiều ứng dụng mạng nơ ron nhân tạo trong những vấn đề khoa học, công nghệ; nhưng hiệu quả nhất làcác bài toán nhận mẫu và dự báo [1, 2, 3, 4, 5] và sẽ được thử nghiệm ở bài báo này.

2. Mạng noron nhân tạo

2.1. Cấu trúc mạng lan truyền ngược

Mạng nơ ron có nhiều cấu trúc, luật học khác nhau. Chúng tôi chọn mạng nơ ron lan truyền ngược (*Back Propagation Neural Networks*) do khả năng hiệu chỉnh sai số đầu ra nhanh so với một số luật học khác. Mạng được chọn ba lớp truyền thẳng (hình 1): lớp vào (lớp 1), lớp ẩn (lớp 2), lớp ra (lớp 3): đủ khả năng nhận các mẫu học (đầu vào); hàm tương tác ra liên tục, khả vi, bị chặn theo (3) [7]. Hàm tương tác đầu ra nơ ron ỏ các lớp được chọn là hàm Sigmoid $f_q(.)$ như nhau:

$$f_q(net_q) = \frac{1}{1+e^{-net_q}} \quad (3)$$



Hình 1. *Cấu trúc mạng nơron ba lớp*

ở đây, net_i là tổng đầu ra của nơ ron thứ *i*:

$$net_i = \sum_{q=1}^{l} w_{iq} z_q = \sum_{q=1}^{l} w_{iq} f\left(\sum_{j=1}^{m} v_{qj} x_j\right) \quad (4)$$

và đầu ra y_i được xác định:

$$y_i = f(net_i) = f\left(\sum_{q=1}^{l} w_{iq} z_q\right) = f\left(\sum_{q=1}^{l} w_{iq} f\left(\sum_{j=1}^{m} v_{qj} x_j\right)\right) \quad (5)$$

trong đó, *i, q, j* là nơ ron thứ *i, q, j* trong *n, l, m* nơ ron lớp ra, lớp ẩn, lớp vào tương ứng. Hoạt động của mạng nơ ron có thể chia thành hai pha: i) pha học: xác định các tham số w_{ij} của mạng, ii) pha chạy, kiểm tra (thử).

2.2. Pha học của mạng nơ ron ba lớp

Phương pháp học trong mạng nơ ron là THỬ-SAI-CHỈNH. "THỬ": là thử cho các giá trị trọng khởi tạo w_{ij}; "SAI": theo học có giám sát: là tồn tại hiệu của đầu ra mong muốn d_i với đầu ra thực tế y_i của từng nơ ron ở mỗi lớp lớn hơn một giá trị nhỏ cho trước. "CHỈNH": là điều chỉnh trọng số một lượng nhỏ Δw_{ij} (7).

Có nhiều kỹ thuật trong phương pháp THỬ-SAI-CHỈNH để tìm Δw_{ij}, ở đây sử dụng kỹ thuật Gradient (7) với hàm sai số:

$$E = \frac{1}{2}\sum_{i=1}^{n}(d_i - y_i)^2 \quad (6)$$

$$\Delta w_{iq} = -\eta\frac{\partial E}{\partial w_{iq}} = -\eta\left[\frac{\partial E}{\partial y_i}\right]\left[\frac{\partial y_i}{\partial net_i}\right]\left[\frac{\partial net_i}{\partial w_{iq}}\right] \quad (7)$$

trong đó: η là hằng số, chỉ tốc độ học.

Kỹ thuật Gradient được áp dụng để tính trọng số w_{ij} cho mạng nhiều lớp, từ lớp vào đến lớp ra gọi là thuật toán lan truyền ngược.

Xét mạng với q lớp, $q = 1, 2\ldots, Q$; ký hiệu ${}^{q}net_i$ và ${}^{q}y_i$ lần lượt là tổng đầu vào và đầu ra của nơron thứ i trong lớp q.

Mạng có m nút vào, n nút ra. Trọng số ${}^{q}w_{ij}$ biểu thị trọng số liên kết từ đầu ra thứ j lớp q-1: ${}^{q-1}y_j$ tới đầu vào thứ i của lớp q: ${}^{q}x_i$.

Thuật toán BP có thể tóm tắt như sau:

Đầu vào: Tập các cặp véc tơ vào/ra mong muốn $\{(x(k), d(k)|k=1, 2,\ldots, p)\}$; trọng khởi tạo $[w_{ij}(t=0)]$; η=0.5; E_{min}; Khởi đầu: $k = 1$.

Đầu ra: $[w_{ij}]$.

Bước 1 (lặp huấn luyện): Đưa mẫu đầu vào thứ k tới lớp đầu vào:

$$ {}^{1}y_i = x_i^{(k)} \ \forall i \tag{8} $$

Bước 2 (lan truyền tiến): Lan truyền tiến tính tín hiệu qua mạng sử dụng:

$$ {}^{q}y_i = f\left({}^{q}net_i\right) = f\left(\sum_j {}^{q}\mathrm{w}_{\mathrm{ij}}^{q-1} y_i\right) \tag{9} $$

Với mỗi i và q tới khi mọi đầu ra của lớp ra ${}^{Q}y_i$ đều được áp dụng.

Bước 3 (tính toán lỗi): Tính giá trị lỗi E và tín hiệu lỗi ${}^{Q}\delta_i$ cho lớp đầu ra:

$$ E^{(k+1)} = \frac{1}{2}\sum_{i=1}^{n}\left(d_i^{(k)} - {}^{Q}y_i\right)^2 + E \tag{10} $$

$$ {}^{Q}\delta_i = \left(d_i^{(k)} - {}^{Q}y_i\right) f'\left({}^{Q}net_i\right) \tag{11} $$

Bước 4 (lan truyền ngược lỗi): cập nhật các trọng số; tính ${}^{q-1}\delta_i$ cho các lớp trước:

$$ \Delta \mathrm{w}_{\mathrm{ij}}^{q} = \eta * \delta_i * {}^{q-1}y_j \tag{12} $$

và $$ {}^{new}w_{\mathrm{ij}}^{q} = {}^{old}w_{\mathrm{ij}}^{q} + \Delta w_{\mathrm{ij}}^{q} \tag{13} $$

$$ {}^{q-1}\delta_i = f'\left({}^{q-1}net_i\right)\sum_j {}^{q}\mathrm{w}_{ji}\,{}^{q}\delta_i $$
$$ q = Q, Q\text{-}1, \ldots, 2 \tag{14} $$

Bước 5 (lặp 1 chu kỳ): Kiểm tra toàn tập dữ liệu luyện được duyệt. Nếu $k < p$ thì $k = k$+1 (quay lại bước 1 cho lần lặp mới), nếu không thì thực hiện bước 6.

Bước 6 (kiểm tra lỗi toàn thể): nếu $E < E_{min}$ kết thúc huấn luyện, nếu không, quay lại bước 1.

2.3. Pha thử nghiệm của mạng nơ ron

Pha thử là pha chạy. Quy trình thực hiện:

Bước 1: Chuẩn, mã hóa dữ liệu đầu vào;

Bước 2: Tính đầu ra;

Bước 3: Giải mã đầu ra.

3. Thử nghiệm nhận dạng và dự báo

3.1. Nhận dạng biển số xe máy

a) Bài toán và quy trình thực hiện

Để nhận dạng biển số xe máy, đầu tiên bằng cách nào đó chụp ảnh biển số. Ảnh phía sau xeđộ phân giải bài báo thực hiên là 640x480. Ảnh chụp qua tiền xử lý, trích ảnh biển số, tách từng ký tự trên biển số, đưa vào module nhận dạng từng ký tự. Ký tự sau khi nhận dạng sẽ được kết xuất và lưu vào cơ sở dữ liệu, gán mã để ghép các ký tự. Dưới đây mô tả các thuật toán xử lý.

b) Thuật toán trích biển số

Đầu vào: ảnh thu được từ camera.

Đầu ra: ảnh biển số trích được (với h: ngưỡng cao; w: ngưỡng rộng).

```
Tách ảnh mới: đỉnh là top, đáy là bottom=RowImage i=0;

thiết lập giá trị ban đầu; flag=0;

for (i=0;  i<image.Width;i++)
{ P=trung bình biên độ phổ  của Cột pixel thứ i;

//sử dụng thư viện AForge
If  (P>=ngưỡng)

    If(flag=1) width=width+1;

    else

    {

        left=i; flag=1;

    }

else

    if (width>=w)right=i;
    }
Biển sô xe= ảnh mới với cạnh phải là right.
 trái là left

    }
```

```
Function fft(image)
{
 i=0;
 thiet lập giá trị ban dau;

 flag=0;

 for (i=0;  i<image.Height;i++)
 {
     P=trung bình biên độ phổ của hàng pixel thứ  i;
     //sử dụng thư viện AForge
     If  (P>=ngưỡng)
         If  (flag=1)
             height=height+1;
         else
          { top=i;

          flag=1;

     }
else

     if (height>=h)
```

c) Thuật toán tách ký tự

Đầu vào: ảnh biển số đã được tiền xử lý.

Đầu ra: mảng 37 ký tự trên biển số; kích thước ma trận điểm ảnh chọn 20x10.

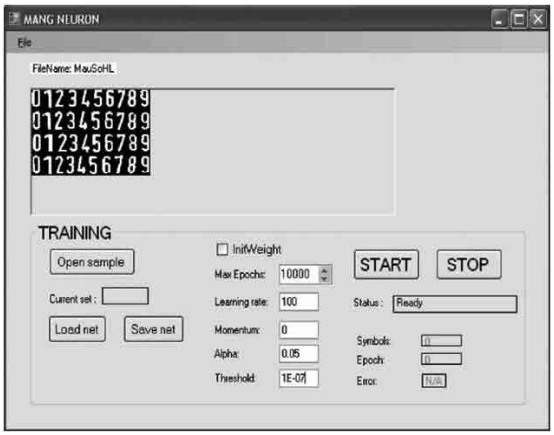
Ảnh nhị phân với giá trị pixel trắng tương ứng là 1, pixel đen có 0. Đây là dữ liệu để đưa vào huấn luyện. Ký tự sau đó chuyển thành mảng 1 chiều 200 phần tử với các giá trị 0 hoặc 1 tương ứng với 200 đầu vào cho mạng nơ ron. Lớp ẩn chọn 100 nút, 10 nút ở lớp ra đối với mạng nhận dạng số và 27 nút đối với mạng nhận dạng chữ. Như vậy, cấu trúc mạng phù hợp với bài toán.

```
Function define pos( image, thres)
{
 i=0;
 thiết lập giá trị ban đầu; c=0;
 flag=0;
 for (i=0; i<image.Width;i++)
 {
      P=trung bình mức xám của cột pixel thứ i;
      If  (P>=ngưỡng)
           If  (flag=1)
                width=width+1;
           else
           {
                left[c]=i; flag=1;
           }
      else
           if (width>=w)
           {
                right[c]=i; c++;
           }
 }
 Tách ảnh ký tự 'c' với canh canh phải là right[c.
 Cạnh trái là left[c],
```

```
Function define_line ( image, thres)
{i=0; c=0;thiet lập giá trị ban đầu; flag=0;
 for (i=0; i<image.Height;i++)
 {
      P=trung bình mức xám của hàng pixel thứ i;
      If  (P>=ngưỡng)
           If  (flag=1)
                height=height+1;
           else
           {
                top[c]=i;
                flag=1;
           }
      else
           if (height>=h)
           {
                bottom[c]=i;
                c++;
           }
 }
 Tách ảnh mới: đỉnh là top, đáy là bottom=  RowImage
}
```

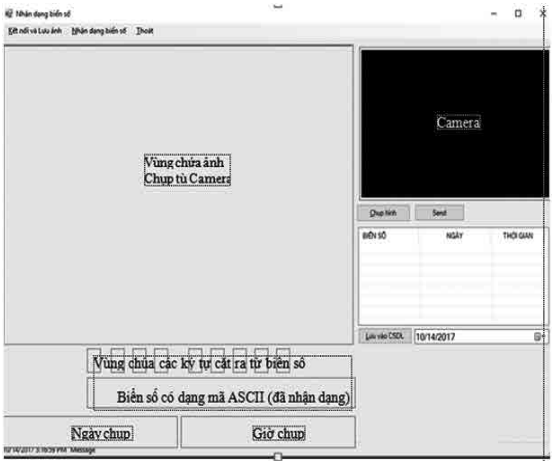
d) Huấn luyện mạng

Huấn luyện mạng hay là học (mục 2.2).



Hình 2. *Giao diện pha học cho nhận dạng*

e) Pha thử nghiệm nhận dạng biển số xe



Hình 3. *Giao diện chương trình nhận dạng*

Bảng 1. *Kết quả nhận dạng biển số xe máy*

Thông số thử nghiệm	IMG_TEST
Số mẫu thử	91
Số trích biển số đúng	79
Số trích biển số sai	12
Tỉ lệ trích biển đúng	88%
Số tách ký tự đúng	71
Số tách ký tự sai	08
Tỉ lệ tách ký tự đúng	90%
Số nhận dạng đúng	59
Số nhận dạng sai	12
Tỉ lệ nhận dạng đúng	84%
Tốc độ xử lý	<2s

Chương trình nhận dạng biển số xe máy (Hình 3) được xử lý trên CPU Core i5 2.66GHz; bộ nhớ RAM 4GB; hệ điều hành Windows 7; ngôn ngữ lập trình Visual Studio 2012. Từ thực nghiệm và Bảng 1 cho thấy, khâu xử lý ảnh trích biển số đúng chưa cao 88% (79/91): phụ thuộc điều kiện và môi trường thu nhận ảnh. Từ 79 biển số tách đúng, tiến hành tách ký tự đạt 90% (71/79).Cuối cùng, khả năng nhận dạng (Hình 3) đạt 84% đối với mạng nơ ron được thiết kế.

3.2. Dự báo thời tiết tỉnh Điện Biên

a. Đặt bài toán, ký hiệu

Cho bộ dữ liệu thời tiết gồm 4 biến số: nhiệt độ, độ ẩm, lượng mây, lượng mưa từ năm 2015 đến ngày 15/10/2017 tại tỉnh Điện Biên. Dự đoán thời tiết từng ngày bất kỳ trong năm được tiến hành khi biết bốn thông số của một ngày trước đó; cụ thể về giá trị:

Biến nhiệt độ (kí hiệu ND), các khoảng giá trị: thấp: ND từ 0 đến 25.5; trung bình: ND từ 25.5

đến 27.0; cao: ND từ 27.0 đến 45.

Biến độ ẩm (kí hiệu ĐA): Thấp: ĐA từ 0 – 80; Cao: ĐA từ 80 – 100.

Biến lượng mưa (kí hiệu M): Không mưa (0); Rải rác: M từ 1 đến 2; Nhỏ: M từ 3 đến 50; lớn: M từ 51 đến 1000 mm.

Biến lượng mây (MA): Ít nếu MA từ 0-20; Nhiều: MA từ 20-30; Rất nhiều MA>30.

b. Dữ liệu dự báo

Dữ liệu theo ngày được thu thập từ Trung tâm Khí tượng Thủy văn 4 tham số với số bản ghi là 1030; 70% (721 bản ghi) dữ liệu dùng để học và 30% (309 bản ghi) để thử.

c. Thuật toán dự báo thời tiết Điện Biên

Các bước thực hiện cho dự báo nhiệt độ:

Bước 1: Chuẩn hóa dữ liệu. Giá trị của nhiệt độ từ 9-40, của độ ẩm từ 59-98, của lượng mưa từ 0-172, của lượng mây từ 0-40. Chuẩn hóa dữ liệu thực hiện trong khoảng [0,1]:

- Tìm $X^{\text{nhiệt độ}}_{\min}$ và $X^{\text{nhiệt độ}}_{\max}$
- Tìm $X^{\text{độ ẩm}}_{\min}$ và $X^{\text{độ ẩm}}_{\max}$
- Tìm $X^{\text{lượng mưa}}_{\min}$ và $X^{\text{lượng mưa}}$
- Tìm $X^{\text{lượng mây}}_{\min}$ và $X^{\text{lượng mây}}_{\max}$

cho từng thành phần ở từng cột

Bước 2: Thiết kế mạng noron (Hình 1): chọn mang 3 lớp (lớp vào có 4 đầu vào; 18 nơ ron lớp ẩn và 3 nơ ron lớp ra). Hằng số học $\eta = 0.5$. Hàm kích hoạt được sử dụng theo (3); khởi tạo giá trị ngẫu nhiên cho trọng số.

Bước 3: Tính lan truyền tiến từ (1)-(5).

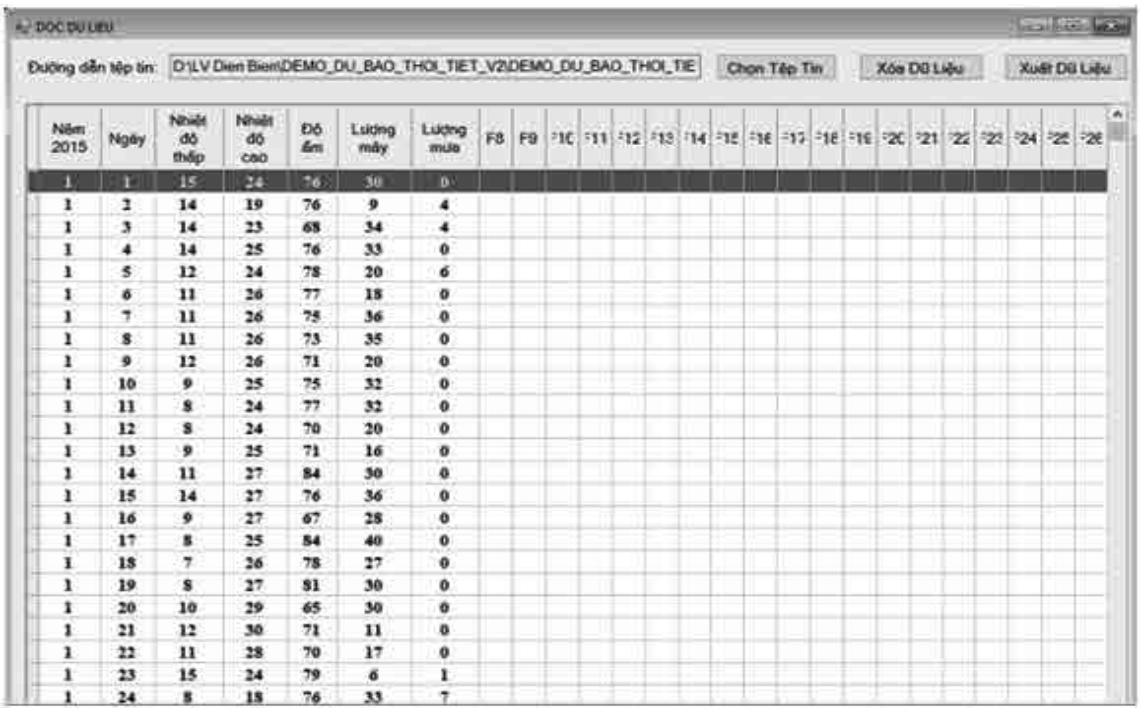
Bước 4: Tính sai số trên đầu ra để quyết định học tiếp? nếu có chuyển đến bước 5.

Bước 5: Tính sai số trên từng nút; cập nhật trọng số và tiếp tục lặp lại bước 3.

d. Các bước thử nghiệm dự báo

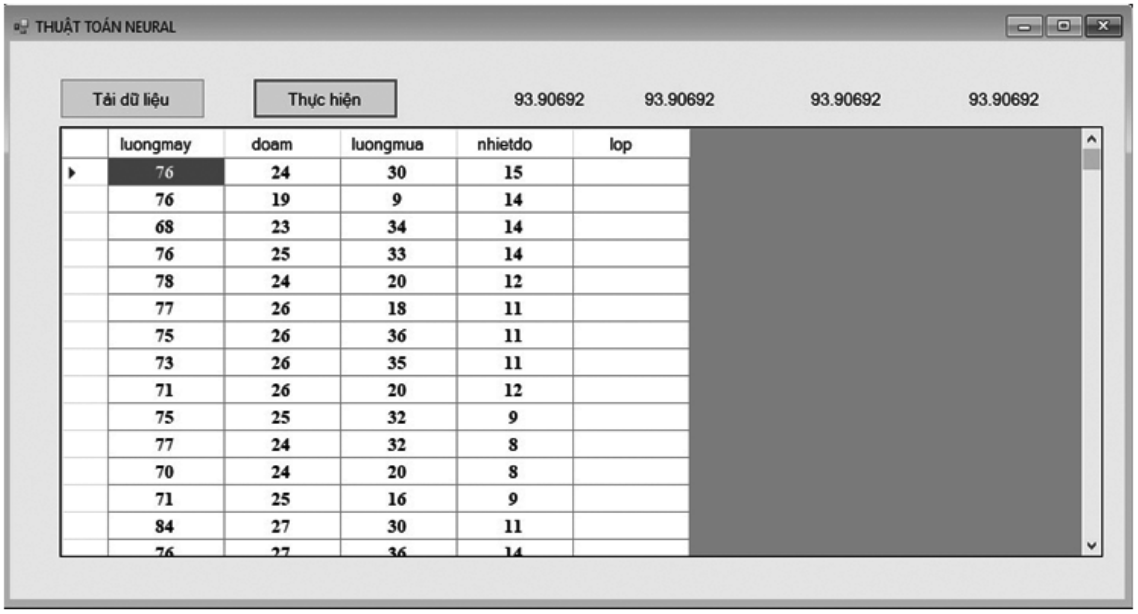
Hình 4. *Giao diện chương trình chính*

Bước 1: Tải dữ liệu mẫu



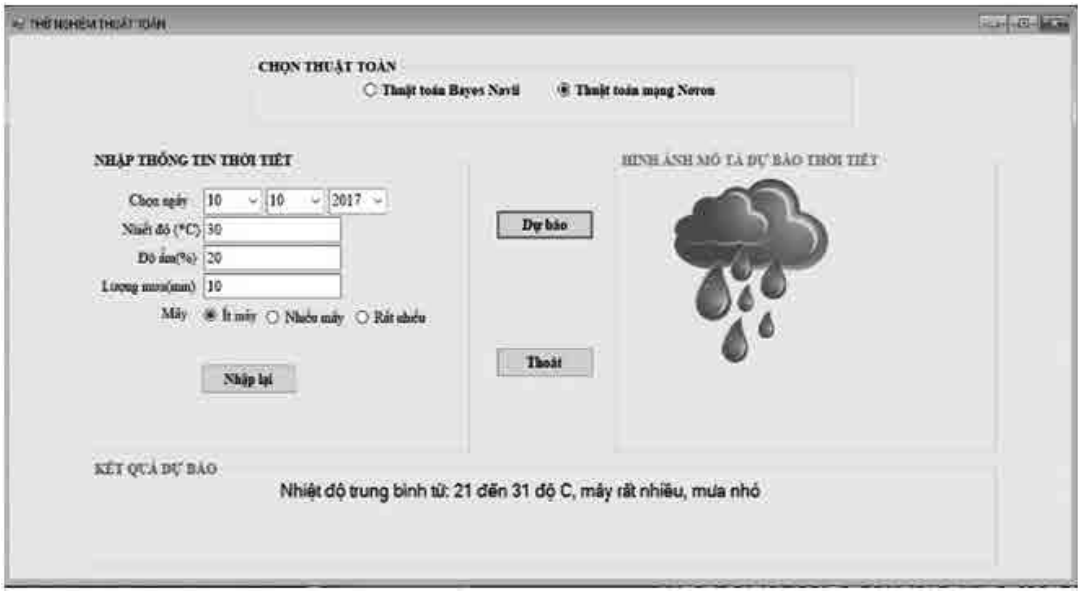
Năm 2015	Ngày	Nhiệt độ thấp	Nhiệt độ cao	Độ ẩm	Lượng mây	Lượng mưa	F8	F9	F10	F11	F12	F13	F14	F15	F16	F17	F18	F19	F20	F21	F22	F23	F24	F25	F26
1	1	15	24	76	30	0																			
1	2	14	19	76	9	4																			
1	3	14	23	68	34	4																			
1	4	14	25	76	33	0																			
1	5	12	24	78	20	6																			
1	6	11	26	77	18	0																			
1	7	11	26	75	36	0																			
1	8	11	26	73	35	0																			
1	9	12	26	71	20	0																			
1	10	9	25	75	32	0																			
1	11	8	24	77	32	0																			
1	12	8	24	70	20	0																			
1	13	9	25	71	16	0																			
1	14	11	27	84	30	0																			
1	15	14	27	76	36	0																			
1	16	9	27	67	28	0																			
1	17	8	25	84	40	0																			
1	18	7	26	78	27	0																			
1	19	8	27	81	30	0																			
1	20	10	29	65	30	0																			
1	21	12	30	71	11	0																			
1	22	11	28	70	17	0																			
1	23	15	24	79	6	1																			
1	24	8	18	76	33	7																			

Bước 2: Huấn luyện mạng (Hình 5)



Hình 5. *Giao diện huấn luyện mạng*

Bước 3: Chọn ngày tháng và dự báo. Kết quả thu được Hình 6.



Hình 6. *Giao diện kết quả dự báo thời tiết*

Bài báo thử các trường hợp sau (Bảng 2):

• Thử nghiệm chỉ tiêu đánh giá trung bình bình phương sai số ***MSE*** **[4]** với số nơ ron lớp ẩn *l=10*, kết quả kém; và *l = 18*: tốt.

• Thử nghiệm dự báo dùng thuật toán Bayes đơn giản. Mạng nơ ron cho kết quả tốt hơn. Điều này có thể giải thích: thuật toán Bayes đã đơn giản hóa với giả thiết giữa các đặc trưng mây, nhiệt độ, độ ẩm, là độc lập do đó có thể làm kém độ chính xác.

• Các đại lượng khác cũng được đánh giá và cho độ chính xác tương tự (trong khuôn khổ bài báo không dẫn bảng số liệu mà chỉ đưa bảng nhiệt độ làm đại diện).

Bảng 2. *Kết quả đánh giá bằng MSE*

Tháng	Số liệu thực		Dự báo	
			Mạng noron	
	T° thấp	T° cao	T° thấp	T° cao
02/01/2017	16	23	15	25
02/02/2017	9	16	9	17
05/03/2017	17	24	16	25
06/04/2017	23	28	22	29
06/05/2017	26	35	25	36
06/06/2017	27	36	26	36
06/07/2017	26	30	24	33
16/08/2017	24	33	24	34
16/09/2017	26	35	26	32
02/01/2017	16	23	15	22
02/02/2017	9	16	8	18
05/03/2017	17	24	16	25
Sai số MSE			***1.11***	***1.78***

4. Kết luận

Bài báo sử dụng các dữ liệu và kết quả chính từ hai luận văn thạc sỹ khóa 2015-2017, Đại học Sư phạm Kỹ thuật Hưng Yên của Ngô Xuân Trang "Nhận dạng biển số xe sử dụng mạng nơron nhân tạo" và"Thử nghiệm dự báo thời tiết tỉnh Điện Biên sử dụng mạng nơ ron nhân tạo" của Lý Đông Hà sử dụng thuật toán lan truyền ngược cho bài toán nhận dang và dự báo. Đóng góp chính của bài báo là: tập hợp, xử lý được dữ liệu, thiết kế mạng nơ ron ba lớp, nhận dạng và thử nghiệm cho nhận dạng biển số xe máy và dự báo thời tiết ở tỉnh Điện Biên. Kết quảdự báo thời tiết của bài báo có khả năng áp dụng cho các tỉnh khác; nhận dạng biển số xe máy cũng có thể mở rộng nhận dạng cho biển số ô tô, hoặc cho các bảng mã kiểu tương tự.

Hướng nghiên cứu tiếp theo:

Một trong những hạn chế của phương pháp học sử dụng luật Gradient là tối ưu chỉ cục bộ hàm lỗi theo công thức (10), có thể không đạt độ chính xác cao. Vì vậy, một trong những hướng phát triển tiếp theo là lai mạng nơ ron với giải thuật di truyền, hoặc sử dụng phương pháp học sâu (Deep Learning)... Mặt khác, dữ liệu thu thập chưa nhiều; nhiều tham số dự báo chưa được đề cập trong bài báo sẽ được bổ sung; việc tính toán cũng sẽ dự kiến thử nghiệm trên các máy tính lớn với tốc độ xử lý nhanh hơn.

Lời cảm ơn

Bài báo này được tài trợ bởi Trung tâm Nghiên cứu Ứng dụng Khoa học và Công nghệ, trường Đại học Sư phạm Kỹ thuật Hưng Yên cho nhóm nghiên cứu "Tính toán mềm", theo Quyết Định số 1417/QĐ-ĐHSPKTHY ngày 06/07/2017.

Tài liệu tham khảo

[1]. Hồ Thị Minh Hà, Nguyễn Hướng Điền (2006), *Thử nghiệm dự báo lượng mưa ngày bằng quy tắc sử dụng mạng thần kinh nhân tạo hiệu chỉnh sản phẩm mô hình số*, Tạp chí Khoa học, Đại học Quốc gia, Hải Dương.

[2]. Nguyễn Quang Hoan, Đoàn Hồng Quang (2014), *Dự báo chỉ số giá chứng khoán bằng mạng Nơ ron-Mờ hồi quy*, Tạp chí Khoa học và Công nghệ, Trường Đại học Sư phạm Kỹ thuật Hưng Yên, Số 1/2014.

[3]. Hoàng Phúc Lâm, Nguyễn Hướng Điền, Công Thanh, Hoàng Thanh Vân (2007), *Sử dụng mạng nơron đa lớp truyền thẳng và mạng truy hồi dự báo tổng lượng bức xạ ngày cho một số trạm ở đồng bằng phía bắc Việt Nam*, Tạp chí Khí tượng Thủy văn, số 10 (559).

[4]. Bin Li (2002), *Spatial Interpolation of Weather Variables using Artificial Neural Network,* Master of Science, University of Georgia, Greece.

[5]. David Silverman, John A. Dracup (2000), *Artificial Neural Networks and Long-Range Precipitation Prediction in California*, Journal of Applied Meteorology, vol 39, pp. 57-66.

[6]. Chi–Teng Lin, C.S. George Lee. (1996), *Neural Fuzzy Systems, A Neuro-Fuzzy Synergism to Intelligent Systems*, Prentice-Hall International, Inc.

[7]. Caruana, Rich; Lawrence, Steve; Giles, Lee. (2000), *Overfiting in Neural Net: Back Propagation*, Conjugate Gradient, and Early Stopping.

ISSN 2354-0575

WEATHER FORECAST AND PATTERN CHARACTER RECOGNITION BY BACKPROPAGATION NEURAL NETWORKS

Abstract:

In this paper, we use BackPropagation Neural Networks to forecast weather of the DienBien province and pattern character recognition on motobicyle codetable. Data for the forecasting are acquired from AccWeather on period time 2015-2017 years for learing (include 70% total data) and testing (include 30% total data). Data for the pattern character recognition on motobicyle codetable are acquired from 64 provinces of Vietnam. The evaluation of two problems using BackPropagation Neural Networkis is proposed.

Keywords: *Artificial Neural Network, Forecast, BackPropagation Algorithm (BP), Pattern Recognition.*