



DỰ BÁO THỜI TIẾT ỨNG DỤNG MẠNG NƠN NHÂN TẠO VÀ THUẬT TOÁN BAYES

Nguyễn Quang Hoan¹, Phạm Thị Trang², Hoàng Hồng Công³, Nguyễn Thị Huyền¹

¹ Trường Đại học Sư phạm Kỹ thuật Hưng Yên

² Trường Đại học Thành Đông

³ Trường Cao đẳng nghề Cơ điện và Thủy lợi

Ngày nhận: 10/12/2016

Ngày sửa chữa: 18/01/2017

Ngày xét duyệt: 05/03/2017

Tóm tắt:

Trong bài báo, chúng tôi sử dụng hai thuật toán trong học máy: thuật toán lan truyền ngược và Bayes để cùng dự báo thời tiết tỉnh Hải Dương. Dữ liệu cho bài toán thu thập được theo từng ngày từ Trung tâm Khí tượng Thủy văn từ năm 2013 – 2015 dùng cho pha học (70% tổng dữ liệu) và pha chạy (30%). Một số kết quả thử nghiệm ban đầu được trình bày và đối sánh hai phương pháp khác nhau; 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, thuật toán Bayes.

1. Giới thiệu

Dự báo là sử dụng thông tin hiện có một cách hiệu quả để định hướng các hoạt động tương lai nhằm đạt được mục đích, yêu cầu của một tổ chức nào đó.

Dự báo thời tiết là tiên đoán, ước lượng, có đánh giá sai lệch các sự kiện xảy ra trong tương lai là vấn đề được quan tâm do kết quả dự báo tác động đến đời sống hàng ngày của mỗi chúng ta. Trong dự báo thời tiết, nhiều yếu tố biến đổi khó lường của thiên nhiên nên có độ phức tạp lớn do đó độ chính xác hạn chế nên các phương pháp mới để giải quyết nó rất được chú trọng.

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; khả năng học tốt; khả năng chịu lỗi cao. Nhiều ứng dụng được sử dụng mạng nơ ron nhân tạo trong nhiều bài toán của khoa học. Ứng dụng mạng nơ ron để trong bài toán dự báo thời tiết sẽ làm giảm đi độ tính toán phức tạp [1, 3, 4, 5, 6, 7] và có nhiều hứa hẹn cho độ chính xác cao.

Trên thế giới [7, 8] cũng như ở Việt Nam nhiều nghiên cứu, ứng dụng mạng nơ ron nhân tạo trong dự báo một số tham số khí tượng thủy văn [3, 5, 7]; dự báo nhu cầu dịch vụ viễn thông [1]; dự báo bức xạ [2, 6]; dự báo giá chứng khoán [4] đã được tiến hành. Điều đó khẳng định khả năng của mạng nơ ron trong bài toán dự báo là khả thi và còn phải tiếp tục nghiên cứu, phát triển và ứng dụng.

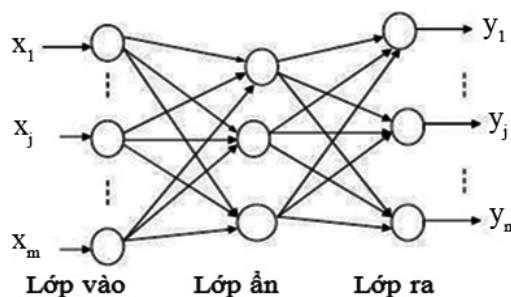
Hiện nay, nhiều phương pháp dự báo được ứng dụng cho nhiều vùng khác nhau nhưng mạng nơ ron và thuật toán Bayes cho dự báo thời tiết cho Hải Dương (một tỉnh đồng bằng, vừa thóc của đồng bằng sông Hồng, gần biển, chịu tác động, ảnh hưởng

nhiều của thời tiết: mưa, bão, lụt...) chưa được thử nghiệm. Đó là mục đích cơ bản của bài báo này.

2. Mạng nơ ron nhân tạo

2.1. Cấu trúc mạng lan truyền ngược

Mạng nơ ron có nhiều cấu trúc, luật học khác nhau [4]. Trong bài báo, chúng tôi chọn mạng nơ ron lan truyền ngược do khả năng hiệu chỉnh sai số đầu ra nhanh (*BackPropagation Neural Networks*) so với một vài luật học khác. Mạng được chọn ba lớp truyền thẳng (Hình 1): lớp vào, lớp ẩn (ở giữa), lớp ra: đủ khả năng nhận biết các mẫu học (đầu vào); hàm tương tác đầu ra liên tục, khả vi, bị chặn (công thức(3))[7].



Hình 1. Sơ đồ cấu trúc mạng nơ ron ba lớp lan truyền ngược

a. Lớp vào (Input Layer)

Đầu vào: x_j ($j = 1, \dots, 4$ cho thực nghiệm, x_1 : nhiệt độ; x_2 : độ ẩm; x_3 : mây; x_4 : mưa).

Lớp vào: chọn số nơ ron ($q = 1, \dots, m$ trong thực nghiệm chọn 3 nơ ron) trong đó q là nơ ron thứ q của lớp vào; v_{qj} là trọng số đầu vào thứ j nối với nơ ron thứ q (tổng số có $q_j = 3 \times 4 = 12$ trọng số trong thực nghiệm).

$$net_q = \sum_{j=1}^m v_{qj}x_j \quad (1)$$

b. Lớp ẩn (Hidden Layer)

Đầu ra số (Digital Output) của lớp vào chính là đầu vào của lớp ẩn, là một hàm của đầu vào tương tự (Analog Output)

$$z_q = f_q(net_q) = f_q\left(\sum_{j=1}^m v_{qj}x_j\right) \quad (2)$$

Hàm tương tác đầu ra của mỗi nơ ron của tất cả các lớp: chọn hàm Sigmoid $f_q(\cdot)$:

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

Chọn $l = 18$ nơ ron cho lớp ẩn. Số này được chọn theo thực nghiệm từ những kết quả tối ưu và giá trị lỗi trong quá trình huấn luyện.

c. Lớp ra (Output Layer): Tổng đầu ra của nơron thứ i (net_i) và đầu ra (bị chặn) y_i có thể xác định theo:

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

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

trong đó i là nơ ron thứ i trong n nơ ron đầu ra được chọn tùy thuộc vào cách mã hoá đặc trưng đầu ra. Trong bối cảnh bài toán đã nêu, số nơ ron lớp ra $n = 3$.

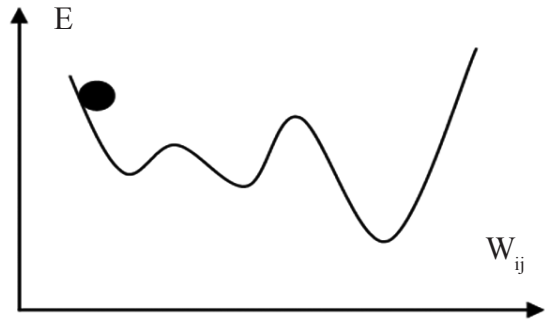
2.2. Pha học: dùng thuật học lan truyền ngược sử dụng phương pháp hạ Gradient

Hoạt động của mạng nơ ron gồm hai giai đoạn, gọi là hai pha: pha học: pha xác định các tham số w_{ij} của mạng. Sau pha học mạng chuyển sang pha hai: (pha mạng đủ điều kiện để chạy) pha chạy thử để kiểm tra và thể hiện khả năng suy diễn của mạng đối với những mẫu lạ (chưa được học).

Mục 2.1 cho ta cấu trúc mạng nơ ron 3 lớp. Để mạng hoạt động được, cần xác định giá trị tất cả các tham số w_{ij} của các lớp. Phương pháp chung để xác định chúng là phương pháp THỬ-SAI-CHỈNH. “THỬ” ở đây có nghĩa là: thử cho ngẫu nhiên các giá trị trọng ban đầu w_{ij} (ngoài ra cần thêm giá trị tốc độ học “ η ” cùng với độ sai số đầu ra E (trong công thức (7)) tối thiểu để chạy được thuật toán học). “SAI” có nghĩa là: theo học có giám sát: là hiệu của đầu ra mong muốn d_i với đầu ra thực tế y_i của từng nơ ron ở mỗi lớp. “CHỈNH” có nghĩa là: điều chỉnh trọng số với giá trị THỬ ban đầu một lượng Δw_{ij} (7).

Có nhiều kỹ thuật xác định Δw_{ij} . Chúng tôi sử dụng kỹ thuật hạ Gradient cho trường hợp này (đạo hàm riêng bậc 1 của một hàm số bậc 2 nhiều biến (6) tức là hàm có dạng parabol có điểm đáy: điểm cực tiểu) là:

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



Hình 2. Minh họa về phương pháp gradient

Theo phương pháp hạ Gradient: tức là tiến trình cập nhật trọng số theo hướng giảm gradient, trọng số liên kết giữa các lớp (ví dụ giữa lớp ẩn và lớp ra) được cập nhật bởi:

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

trong đó: η là hằng số học. Nó chỉ tốc độ cập nhật trọng số từng bước lặp của thuật toán.

Kỹ thuật hạ Gradient có thể được áp dụng để tính trọng số w_{ij} từ lớp đầu vào đến lớp đầu ra gọi là tính trọng số theo kiểu truyền thẳng. Ngược lại, có thể tính trọng số từ lớp ra ngược về lớp đầu (làm cho sai số có xu hướng giảm dần, chống hội tụ nên ưa được sử dụng). Thuật toán tính ngược từ ra tới vào gọi là thuật toán lan truyền ngược (Backpropagation Algorithm: BP).

Tuy nhiên, phương pháp hạ Gradient có hạn chế là có thể tối ưu cục bộ (tức là có thể hội tụ về một đáy. Đáy đó, nói chung chưa hẳn có giá trị sai số nhỏ nhất gọi là tối ưu cục bộ (Hình 2). Để khắc phục tối ưu cục bộ nói trên có thể dùng giải thuật di truyền để xác định giá trị tối ưu toàn cục. Giả sử, chúng ta đã thực hiện được việc đó (chúng tôi đã tiến hành trong bài báo khoa học khác) với tập giá trị trọng có thể đưa về tối ưu toàn cục.

Thuật toán BP có thể tóm tắt như sau:

Xét mạng với q lớp truyền thẳng, $q = 1, 2, \dots, Q$ 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 đầu vào và n nút đầu ra. ${}^q w_{ij}$ biểu thị trọng số liên kết từ ${}^{q-1} y_j$ tới ${}^q y_i$.

- **Đầu vào:** Một tập các cặp huấn luyện $\{(x(k), d(k)) | k = 1, 2, \dots, p\}$.

- **Bước 0 (khởi tạo):** Chọn tốc độ học $\eta = 0.5$ và $E_{min} = 0$ (là giá trị lỗi cực đại chấp nhận được). Khởi tạo các trọng số với giá trị ngẫu nhiên nhỏ; cho bước lặp khởi đầu: $k = 1$.

- **Bước 1 (lập huấn luyện):** Đưa mẫu đầu vào thứ k tới lớp đầu vào:

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

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

$${}^q y_i = f({}^q net_i) = f\left(\sum_j {}^q w_{ij}^{q-1} y_j\right) \quad (9)$$

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:

$$E(k+1) = \frac{1}{2} \sum_{i=1}^n (d_i^{(k)} - {}^q y_i)^2 + E(k) \quad (10)$$

$${}^q \delta_i = (d_i^{(k)} - {}^q y_i) f'({}^q net_i) \quad (11)$$

- Bước 4 (lan truyền ngược lỗi): Lan truyền ngược trở lại để cập nhật các trọng số và tính toán tín hiệu lỗi ${}^{q-1} \delta_i$ cho các lớp trước:

$$\Delta w_{ij}^q = \eta * \delta_i * {}^{q-1} y_j \quad (12)$$

$$\text{và } {}^{new} w_{ij}^q = {}^{old} w_{ij}^q + \Delta w_{ij}^q \quad (13)$$

$${}^{q-1} \delta_i = f'({}^{q-1} net_i) \sum_j {}^q w_{ij}^q \delta_j \quad (14)$$

$$q = Q, Q-1, \dots, 2$$

- Bước 5 (lặp 1 chu kỳ): Kiểm tra toàn bộ tập dữ liệu luyện đã 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): Kiểm tra lỗi toàn thể, nếu $E < E_{min}$ thì kết thúc quá trình huấn luyện, hoặc kiểm tra xem đã đạt đến số chu kỳ (vòng lặp) giới hạn chưa, nếu không $E = 0$, $k = 1$ và bắt đầu một chu kỳ mới bằng việc quay lại bước 1.

2.3. Pha thử nghiệm

Pha thử (Test) thực chất là pha chạy. Pha chạy hoàn toàn sử dụng lại cùng một cấu trúc mạng nơ ron như Hình 1 để kiểm tra khả năng suy diễn của mạng nơ ron.

Quy trình thực hiện trong pha chạy như sau:

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

Bước 2: Tính đầu ra;

Bước 3: Giải mã đầu ra để nhận các tham số (dự báo) thời tiết.

3. Thuật toán Naïve Bayes[9]

Cho trước tập dữ liệu huấn luyện D với mô tả (nhân) của các lớp C_i $i = 1..m$, quá trình phân loại một đối tượng $X = (x_1, x_2, \dots, x_n)$ với mạng Bayes như sau:

- X được phân loại theo C_i nếu và chỉ nếu:

$$P(C_i | X) > P(C_j | X) \text{ với } 1 \leq j \leq m;$$

$$j \neq i \quad P(C_i | X) = \frac{P(X | C_i) P(C_i)}{P(X)} \quad (15)$$

+ Tối đa hóa $P(C_i | X)$ (Chọn C_i nếu $P(C_i | X)$ là giá trị lớn nhất)

$$+ \text{Tối đa hóa } P(C_i | X) P(C_i)$$

$$P(C_1) = P(C_2) = \dots = P(C_m)$$

$$\text{hoặc } P(C_i) = \frac{|C_{i,D}|}{|D|} \dots$$

$$P(C_i | X) = \prod_{i=1}^n P(x_k | C_i) = P(x_1 | C_i) \times \dots \times P(x_n | C_i) \quad (16)$$

- $P(X | C_i)$ được tính với giả định x_k độc lập có điều kiện; $k = 1..n$:

- $P(x_k | C_i)$ được tính như sau:

+ Nếu A_k là thuộc tính rời rạc

$$P(x_k | C_i) = \frac{|\{X' | x'_k = x_k \wedge X' \in C_i\}|}{|C_{i,D}|} \quad (17)$$

+ Nếu A_k là thuộc tính liên tục

$P(x_k | C_i)$ được ước lượng thông qua hàm mật độ:

$$P(x_k | C_i) = g(x_k, \mu_{c1}, \sigma_{c1}) = \frac{1}{\sqrt{2\pi}\sigma_{c1}} e^{-\frac{(x_k - \mu_{c1})^2}{2\sigma_{c1}^2}} \quad (18)$$

Trong đó, μ : giá trị trung bình

$$\mu = \frac{1}{n} \sum_{k=1}^n x_k \quad (19)$$

σ : độ lệch chuẩn, và :

$$\sigma^2 = \frac{1}{n-1} \sum_{k=1}^n (x_k - \mu)^2 \quad (20)$$

Tóm lại, để phân lớp mẫu chưa biết X , ta tính $P(X | C_i) P(C_i)$ cho từng C_i , gán X vào lớp C_i sao cho $P(X | C_i) P(C_i)$ là lớn nhất.

$$\max_{C_i \in C} (P(C_i) \prod_{k=1}^n P(x_k | C_i)) \quad (21)$$

4. Xây dựng hệ thống dự báo thời tiết

4.1. Phân tích bài toán

Trong bài báo, chúng tôi giới hạn 4 tham số chính ảnh hưởng đến thời tiết là: nhiệt độ, độ ẩm, mây và mưa. Mỗi yếu tố có những thuộc tính và độ phụ thuộc khác nhau. Ví dụ tùy thuộc vào từng tháng mà nhiệt độ sẽ có những giá trị khác nhau. Nếu tháng đó rơi vào mùa mưa thì nhiệt độ có thể thấp ($< 25.5^\circ\text{C}$); trung bình ($25.5^\circ\text{C} - 27.0^\circ\text{C}$), ngược lại nhiệt độ cao ($> 27.0^\circ\text{C}$)... Dựa trên các đặc điểm đó ta xây dựng bài toán:

Với bốn yếu tố chính ảnh hưởng đến thời tiết trên, ta có thể phân chia mỗi yếu tố thành các trường hợp cụ thể như sau:

a. *Biến nhiệt độ (kí hiệu ND)*: có các khoảng giá trị: thấp nếu ND từ 0 đến 25.5; trung bình ND từ 25.5 đến 27.0; cao ND từ 27.0 đến 45;

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

c. *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.

d. *Biến lượng mây (MA)*. Ít nếu MA từ 0 đến 20; Nhiều: MA từ 20 đến 30; Rất nhiều nếu lượng mây lớn hơn 30.

4.2. Dữ liệu cho thử nghiệm

Dữ liệu được từ năm 2013 đến năm 2015,

thu thập được theo từng ngày từ Trung tâm Khí tượng Thủy văn của tỉnh Hải dương với 4 tham số chính: nhiệt độ, độ ẩm, mây và mưa với số bản ghi là 1095. Tất cả dữ liệu được chia thành hai phần: 70% dữ liệu dùng để học (800 bản ghi) và 30% dữ liệu để thử (295 bản ghi).

4.3. Kết quả đạt được

Sau khi phân tích hệ thống và tìm hiểu giải thuật, chúng tôi đã xây dựng được chương trình dự báo thời tiết chạy trên máy tính cá nhân trên hệ điều hành Window 7, vi xử lý Intel Core i3 - 4170 với tốc độ 2.7 GHz, bộ nhớ đệm 2MB Cache, RAM 2GB. Chương trình được viết trên ngôn ngữ C#. Chúng tôi thử nghiệm hai phương pháp: dùng thuật toán Bayes (thể hiện kết quả trên Hình 3) mạng nơron lan truyền ngược (kết quả hiện thị trên Hình 4).

Hình 3. Dự báo thời tiết với thuật toán Bayes

Hình 4. Dự báo thời tiết với mạng nơron Lan truyền ngược

4.4. Đánh giá kết quả

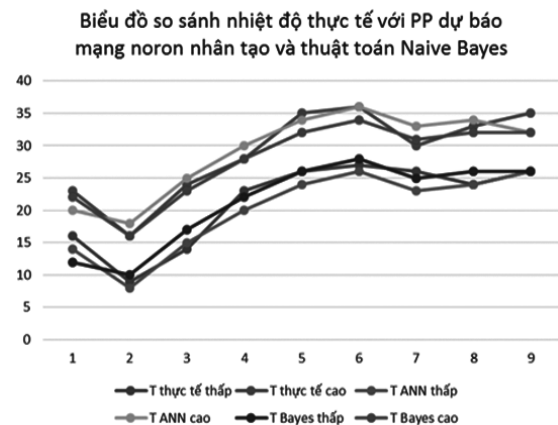
Kết quả dự báo nhiệt độ bằng mạng nơron lan truyền ngược và thuật toán Naïve Bayes thu được từ thử nghiệm một số ngày trong năm 2016.

Trong phạm vi bài báo chúng tôi đánh giá kết quả cho nhiệt độ (về lượng mây, mưa cũng cho những giá trị tương tự).

Trong Hình 5: vùng nhiệt độ, vùng nhiệt độ cao ở trên và vùng nhiệt độ thấp ở dưới. Ở mỗi vùng có 3 đường là nhiệt độ thực tế, nhiệt độ dự báo của mạng nơron, nhiệt độ dự báo của mạng nơron nhân tạo. Trong chương trình những ngày khác cũng có thể được dự báo tùy theo nhu cầu người dùng.

Bảng 1. So sánh kết quả dự báo của mạng nơron lan truyền ngược và và thuật toán Naïve Bayes

TT	Tháng	Kết quả thực tế	Phương pháp ANN	Thuật toán Naïve Bayes
1	02/01/2016	16-23	14-20	12-22
2	02/02/2016	9-16	8-18	10-16
3	05/03/2016	17-24	15-25	17-23
4	06/04/2016	23-28	20-30	22-28
5	06/05/2016	26-35	24-34	26-32
6	06/06/2016	27-36	26-36	28-34
7	06/07/2016	26-30	23-33	25-31
8	16/08/2016	24-33	24-34	26-32
9	16/09/2016	26-35	26-32	26-32



Hình 5. So sánh kết quả dự báo của mạng nơron lan truyền ngược và và thuật toán Bayes

5. Kết luận

Đóng góp của bài báo là thử nghiệm dự báo nhiệt độ tỉnh Hải Dương sử dụng thuật toán lan truyền ngược sai số và thuật toán Bayes cho một vài kết quả khả quan, nhưng độ phức tạp khá lớn. Tuy nhiên, nếu thu thập được thêm dữ liệu đầy đủ như hướng gió, tốc độ gió, áp suất... thì mức độ dự báo sẽ phong phú hơn. Các tham số như mây, mưa cũng hoàn toàn có thể số hóa và cho kết quả như nhiệt độ (trong phạm vi giới hạn của khuôn khổ bài báo và tạp chí, chúng tôi chỉ minh họa một tham số)

Hệ thống có thể dễ dàng cài đặt và sử dụng dựa trên máy tính cá nhân, có tốc độ dự báo nhanh.

Hướng nghiên cứu tiếp theo: Các phương pháp lai khác cho bài toán dự báo thời tiết không chỉ cho tỉnh Hải Dương mà cho các vùng miền khác như: lai giải thuật di truyền-Noron-Mô, học sâu

(Deep Learning) ...sẽ là hướng nghiên cứu tiếp theo. Mặt khác, nhiều tham số dự báo chưa được đề cập trong bài báo với khối lượng tính toán lớn cũng sẽ dự kiến thử nghiệm trên các máy tính lớn.

Tài liệu tham khảo

- [1]. Trần Duy Dũng (2001), *Phát triển phần mềm dự báo nhu cầu dịch vụ cho mạng viễn thông Việt nam sử dụng công nghệ AI*. Mã số:087-2001-TCT-RDP-TH-67, Viện Khoa học Kỹ thuật Bưu điện.
- [2]. Nguyễn Hương Điền, Hoàng Phúc Lâm (2006), *Dự báo tổng lượng bức xạ ngày cho khu vực đồng bằng phía Bắc Việt Nam bằng phương pháp sử dụng mạng thần kinh nhân tạo*, Tạp chí khoa học ĐHQGHN, KHTN&CN.
- [3]. Hồ Thị Minh Hà, Nguyễn Hương 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.
- [4]. Nguyễn Quang Hoan, Hoàng Thị Lan Phương (2006), *Dự báo giá chứng khoán sử dụng công nghệ mạng noron*, Kỷ yếu hội nghị khoa học lần 3 về nghiên cứu, phát triển và ứng dụng công nghệ thông tin và truyền thông. ICT.rda'06 (20-21/5/2006).
- [5]. Hoàng Phúc Lâm (2006), *Nghiên cứu và ứng dụng mạng thần kinh nhân tạo dự báo một số yếu tố khí tượng cho khu vực đồng bằng phía bắc Việt Nam*, Luận văn thạc sĩ khoa học.
- [6]. Hoàng Phúc Lâm, Nguyễn Hương Điền, Công Thanh, Hoàng Thanh Vân (2007), *Sử dụng mạng noron 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).
- [7]. Bin Li (2002), *Spatial Interpolation of Weather Variables using Artificial Neural Network*, Master of Science, University of Georgia, Greece.
- [8]. 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.
- [9]. Nguyễn Thị Thu Hòa (2011), *Ứng dụng phương pháp hồi quy Bayes vào việc dự báo thời tiết*, Luận văn thạc sĩ khoa học.

WEATHER FORECASTING USING ARTIFICIAL NEURAL NETWORK AND BAYES ALGORITHM

Abstract:

In this paper, we use two algorithms in machine learning: Back Propagation Neural Networks and Bayes algorithm for weather forecasting in the Hai Duong province. Data for the forecasting are acquired from the AccWeather on the period time 2013-2015 years (in every day) in which for the learning (include 70% total data) and for the testing (include 30% total data). We received some positive results in the temperature forecasting. The comparison of two forecasting results between Back Propagation and Bayes algorithm is proposed.

Keywords: Artificial Neural Network, Forecast, Back Propagation algorithm, Bayes algorithm.