

NHẬN DẠNG KÝ TỰ QUANG HỌC BẰNG MẠNG NƠN

OPTICAL CHARACTER RECOGNITION BY NEURAL NETWORK

NGÔ VĂN SỸ

Trường Đại học Bách khoa, Đại học Đà Nẵng

TÓM TẮT

Nhận dạng ký tự quang học là kỹ thuật được sử dụng để chuyển đổi ảnh văn bản sang dạng văn bản có thể chỉnh sửa trong máy tính. Nó được ứng dụng trong công tác quét và lưu trữ các tài liệu cũ, đẩy nhanh việc nhập dữ liệu vào máy với ít lỗi hơn. Bài báo giới thiệu một phương pháp nhận dạng ký tự, đó là kỹ thuật mạng nơron.

ABSTRACT

Optical Character Recognition (OCR) is a technology used to convert scanned images of a text into an editable and searchable text on the computer. It can be used for preserving historical documents and scanning data entry forms in a faster and less error prone manner... This paper introduces a character-identifying method called Neural Network Technology.

1. Đặt vấn đề

Mạng nơron nhân tạo (Artificial Neural Network: ANNs) là sự tái tạo bằng kỹ thuật những chức năng của hệ thần kinh con người với vô số các nơron được liên kết truyền thông với nhau qua mạng. Giống như con người, ANNs được học bởi kinh nghiệm, lưu những kinh nghiệm đó và sử dụng trong những tình huống phù hợp.

Trong kỹ thuật nhận dạng ký tự, mạng nơron tỏ ra ưu thế hơn các phương pháp truyền thống ở chỗ không tốn thời gian cho thủ tục tiền xử lý, làm mảnh ký tự, trích trọt đặc trưng... Mặt khác các phương pháp ra quyết định trong nhận dạng truyền thống được cài đặt tĩnh trong chương trình, khi muốn bổ sung thêm các mẫu học mới phải thiết kế lại chương trình. Trong khi với mạng nơron, chỉ cần cung cấp một tập mẫu vào ra của dữ liệu mới cho pha huấn luyện là có thể bổ sung vào “bộ nhớ mạng” những kiểu dữ liệu mới mà không ảnh hưởng đến cấu trúc chương trình ban đầu.

Trong phạm vi bài báo này sẽ giới thiệu tổng quan về lý thuyết mạng nơron và ứng dụng mạng Perceptron nhiều lớp lan truyền ngược sai số để thiết kế chương trình nhận dạng ký tự quang học.

2. Phương pháp và thuật toán nhận dạng ký tự

2.1. Cơ sở dữ liệu

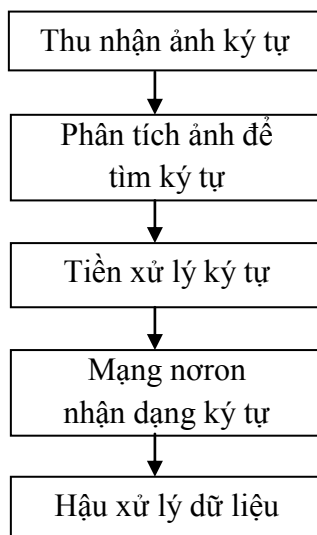
ABCDEFGHIJKLMNOPQRSTUVWXYZ
abcdefghijklmnopqrstuvwxyz
0123456789
~!@#\$%^&*()-+=\{}[]: ';<>, . / ?

*Hình 1: Mẫu các ký tự trong
nhận dạng ký tự quang*

Cơ sở dữ liệu cho bài toán nhận dạng ký tự quang gồm 90 ký tự Latinh với các loại font khác nhau, cùng với giá trị Unicode tương ứng của chúng.

2.2. Phương pháp nhận dạng

Phương pháp nhận dạng ký tự quang bằng mạng nơron bao gồm các bước được mô tả như trong hình 2.



Hình 2: Các bước trong nhận dạng ký tự dùng mạng nơron

2.2.1. Thu nhận ảnh

Ảnh văn bản, tài liệu có thể được thu nhận bằng máy quét scanner, webcam, hoặc các thiết bị thu nhận ảnh thông dụng khác.

2.2.2. Phân tích ảnh để tìm ký tự

Quá trình phân tích ảnh để tìm ký tự bao gồm các bước sau:

- Tách dòng ký tự ra khỏi ảnh ký tự.
- Tách từ riêng biệt ra khỏi dòng ký tự.
- Tách riêng từng ký tự ra khỏi từ.

Thuật toán sử dụng để tách ký tự ra khỏi ảnh văn bản dựa trên đặc tính biên độ về độ sáng của các pixel ảnh.

2.2.3. Tiền xử lý ký tự

Quá trình tiền xử lý ký tự giải quyết vấn đề ánh xạ giá trị pixel ảnh ký tự vào ma trận 10x15 và tuyến tính hóa ma trận thành 150 giá trị đưa vào 150 nơron ở lớp vào của mạng.

2.2.4. Mạng nơron nhận dạng ký tự

Hiện nay, các loại mạng nơron thông dụng gồm có: mạng truyền thẳng (feed-forward), mạng hồi qui (feedback), mạng tự tổ chức (self-organizing). Mạng truyền thẳng feed-forward bao gồm nhiều lớp các đơn vị xử lý phi tuyến (non-linear processing unit). Một vector đầu vào sẽ được đưa vào lớp vào của mạng và sau đó các tính toán được thực hiện lan truyền thẳng từ lớp vào sang các lớp ẩn và kết thúc ở lớp ra. Mạng Perceptron nhiều lớp MLP (MultiLayer Perceptron) là một trong những loại mạng

truyền thẳng điển hình, được sử dụng rộng rãi trong các hệ thống nhận dạng như nhận dạng ký tự quang, chữ viết tay, nhận dạng tiếng nói...

Trong bài báo này, chúng tôi thiết kế chương trình nhận dạng ký tự quang sử dụng mạng MLP có 3 lớp: lớp vào có 150 nút tương ứng với 150 phần tử của vector ma trận pixel, lớp ẩn có 250 nơron và lớp ra có 16 nơron tương ứng với 16 bit nhị phân của giá trị Unicode của các ký tự.

Quá trình huấn luyện mạng:

Quá trình huấn luyện là quá trình học với các tập mẫu (X_s, T_s) , $s = \overline{1, N}$ để điều chỉnh tập trọng số liên kết. Giải thuật huấn luyện được áp dụng trong thiết kế chương trình nhận dạng ký tự là giải thuật lan truyền ngược sai số Back Propagation.

Bước 1: Lan truyền xuôi đầu vào $X_s = \{x_1, x_2, \dots, x_n\}$ qua mạng:

$$* \text{Đầu ra tại nơron } j \text{ của lớp ẩn: } y_j = g\left(\sum_{i=1}^n w_{ji} \cdot x_i - \theta_j\right) \quad (1)$$

* Đầu ra tại nơron k của lớp ra:

$$y_k = g\left(\sum_{j=1}^m y_j \cdot w_{jk}\right) = g\left(\sum_{j=1}^m w_{jk} \cdot g\left(\sum_{i=1}^n w_{ji} \cdot x_i - \theta_j\right) - \theta_k\right) \quad (2)$$

Bước 2: Lan truyền ngược sai số:

So sánh các phần tử của vector đầu ra thực Y_s với các phần tử tương ứng của vector đầu ra mẫu T_s để tính sai lệch: $e_k = t_k - y_k$

Tổng bình phương sai số của mạng ứng với mẫu học (X_s, T_s) :

$$E = \frac{1}{2} \sum_{k=1}^p (y_k - t_k)^2$$

Thông tin về sai số được lan truyền ngược qua mạng để điều chỉnh lại các giá trị trọng số tại vòng lặp thứ l:

* Với liên kết giữa nơron ẩn và nơron ra:

$$\Delta w_{jk} = \eta \cdot \delta_k(l) \cdot y_j(l), \text{ với } \eta \text{ là hệ số học, } y_j \text{ tính theo công thức (1)}$$

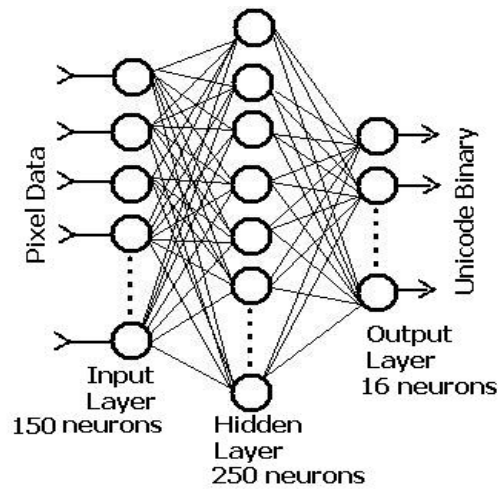
$$\delta_k(l) = e_k(l) \cdot g'_k(y_k(l))$$

$$w_{jk}(l+1) = w_{jk}(l) + \Delta w_{jk} \quad (3)$$

* Với liên kết giữa nơron vào và nơron ẩn:

$$\Delta w_{ij} = \eta \cdot \delta_j(l) \cdot x_i$$

$$\delta_j(l) = g'_j(y_j) \cdot \sum_{k=1}^m \delta_k(l) \cdot w_{jk}(l+1)$$



Hình 3: Mạng MLP trong nhận dạng ký tự quang

$$w_{ij}(l+1) = w_{ij}(l) + \Delta w_{ij} \quad (4)$$

Sau khi hiệu chỉnh trọng số, mẫu X_s tiếp tục được đưa vào mạng lần thứ $(l+1)$ và tiếp tục thuật toán hiệu chỉnh trọng số cho đến khi $E < \varepsilon$ cho trước hoặc số vòng lặp đạt đến mức định trước.

Mẫu tiếp theo được đưa vào mạng và quá trình huấn luyện lặp lại như trên cho đến khi mạng học thuộc tất cả các mẫu. Lưu cấu hình mạng lại để sẵn sàng đưa vào sử dụng.

Hàm kích hoạt sử dụng trong mạng Perceptron nhiều lớp là hàm sigmoid lưỡng cực: $g(s) = \frac{2}{1 + e^{-\alpha s}} - 1$, hoặc hàm sigmoid đơn cực $g(s) = \frac{1}{1 + e^{-\alpha s}}$, đạo hàm của cả hai hàm này đều là: $g'(s) = g(s)(1 - g(s))$.

Sử dụng mạng:

Mạng sau khi được huấn luyện được sử dụng như một bảng tra. Các ảnh ký tự sau bước tiền xử lý được đưa vào đầu vào mạng, đầu ra mạng là vector Y với các phần tử y_k được tính theo công thức (1) và (2). Như vậy đầu ra mạng là giá trị Unicode của ký tự.

2.2.5. Hậu xử lý dữ liệu

Giai đoạn này làm nhiệm vụ chuyển đổi giá trị Unicode sang dạng ký tự tương ứng và sắp xếp lại các ký tự dưới dạng text theo dạng ảnh văn bản ban đầu.

2.3. Kết quả và đánh giá

Thử nghiệm việc huấn luyện mạng với tập mẫu của 3 loại font: Latinh Arial, Latinh Tahoma, Latinh Times Roman.

2.3.1. Kết quả khi thay đổi số vòng lặp (Epoch)

Số ký tự = 90, tốc độ học learning_rate = 150, hệ số góc hàm sigmoid $\alpha = 0.014$

Loại Font	Epoch=300		Epoch=600		Epoch=800	
	Số ký tự sai	% Error	Số ký tự sai	% Error	Số ký tự sai	% Error
Latin Arial	4	4.44	3	3.33	1	1.11
Latin Tahoma	1	1.11	0	0	0	0
Latin Times Roman	0	0	0	0	1	1.11

2.3.2. Kết quả thay đổi tham số learning_rate

Số ký tự = 90, số Epoch = 600, hệ số góc $\alpha = 0.014$

Loại Font	50		100		120	
	Số ký tự sai	% Error	Số ký tự sai	% Error	Số ký tự sai	% Error
Latin Arial	82	91.11	18	20	3	3.33
Latin Tahoma	56	62.22	11	12.22	1	1.11
Latin Times Roman	77	85.56	15	16.67	0	0

2.3.3. Đánh giá ảnh hưởng của sự thay đổi các thông số.

Tăng số vòng lặp tổng quát sẽ tỉ lệ thuận hiệu suất của mạng. Nhưng đến một mức nào đó nó sẽ gây ra hiện tượng học vẹt (over learning), sẽ bỏ qua trạng thái tối ưu.

Kích thước đầu vào cũng là một nhân tố ảnh hưởng trực tiếp đến hiệu suất của mạng. Thực tế số đầu vào càng nhiều mạng yêu cầu được huấn luyện nhiều hơn để có thể nhận bắt lỗi tốt.

Sự thay đổi tham số `learning_rate` cũng ảnh hưởng hiệu suất của mạng đối với số vòng lặp xác định. Giá trị tham số này càng nhỏ thì mạng hiệu chỉnh trọng số càng chậm. Nhưng muốn đạt đến trạng thái tối ưu thì ta phải tăng số vòng lặp nhiều hơn.

3. Kết luận

Với các kết quả thu được trong quá trình thử nghiệm huấn luyện ta chọn các thông số mạng: số vòng lặp `epoch=600`, tốc độ học `learning_rate=150`, hệ số góc hàm sigmoid $\alpha = 0.014$, ngưỡng sai số $\varepsilon = 0.0002$.

TÀI LIỆU THAM KHẢO

- [1] Lương Mạnh Bá, Nguyễn Thanh Thủy (1999), *Nhập môn xử lý ảnh số*, Nhà xuất bản khoa học và kỹ thuật, Hà Nội, tr 153-211.
- [2] Học viện công nghệ Bưu chính Viễn Thông (2007), *Nhập môn Trí tuệ Nhân tạo*, Hà Nội, tr 118-133.
- [3] K.-L. Du, PhD, M.N.S. Swamy, PhD, D.Sc (Eng) (2006), *Neural Networks in a Softcomputing Framework.pdf*, Concordia University, Montreal, Canada.
- [4] CHRISTOPHER M. BISHOP(1995), *Neural Networks for Pattern Recognition.pdf*, CLARENDON PRESS • OXFORD, UK.