BỘ GIÁO DỤC VÀ ĐÀO TẠO

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ TP. HCM

PHẠM CÔNG THIỆN

NGHIÊN CỨU MẠNG NƠRON NHÂN TẠO VÀ ỨNG DỤNG VÀO TRAO ĐỔI KHÓA BÍ MẬT

LUẬN VĂN THẠC SĨ

Chuyên ngành : Công Nghệ Thông Tin

Mã số ngành: 60480201

TP. Hồ Chí Minh, Tháng 04 Năm 2015

BỘ GIÁO DỤC VÀ ĐÀO TẠO

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ TP. HCM

PHẠM CÔNG THIỆN

NGHIÊN CỨU MẠNG NƠRON NHÂN TẠO VÀ ỨNG DỤNG VÀO TRAO ĐỔI KHÓA BÍ MẬT

LUẬN VĂN THẠC SĨ

Chuyên ngành : Công Nghệ Thông Tin

Mã số ngành: 60480201

CÁN BỘ HƯỚNG DẪN KHOA HỌC: TS LƯ NHẬT VINH

TP. Hồ Chí Minh, Tháng 04 Năm 2015

CÔNG TRÌNH ĐƯỢC HOÀN THÀNH TẠI
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ TP. HCM

Cán bộ hướng dẫn khoa học : TS Lư Nhật Vinh
(Ghi rõ họ, tên, học hàm, học vị và chữ ký)

Luận văn Thạc sĩ được bảo vệ tại Trường Đại học Công nghệ TP. HCM ngày 11 tháng 04 năm 2015

Thành phần Hội đồng đánh giá Luận văn Thạc sĩ gồm:

TT	Họ và tên	Chức danh Hội đồng
1	PGS TS Lê Trọng Vĩnh	Chủ tịch
2	PGS TS Đỗ Phúc	Phản biện 1
3	PGS TS Lê Hoàng Thái	Phản biện 2
4	TS Võ Đình Bảy	Ủy viên
5	TS Lê Tuấn Anh	Ủy viên, Thư ký

Xác nhận của Chủ tịch Hội đồng đánh giá Luận sau khi Luận văn đã được sửa chữa

Chủ tịch Hội đồng đánh giá LV

TRƯỜNG ĐH CÔNG NGHỆ TP. HCM

PHÒNG QLKH – ĐTSĐH

CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM

Độc lập – Tự do – Hạnh phúc

TP.HCM, Ngày 10 tháng 03 năm 2015

NHIỆM VỤ LUẬN VĂN THẠC SĨ

Họ tên học viên: .....PHẠM CÔNG THIỆN.....Giới tính: ..Nam.....

Ngày, tháng, năm sinh: 18-01-1987.....Nơi sinh:..Vĩnh Long ..

Chuyên ngành: .....Công nghệ thông tin.....MSHV:1341860024....

I- Tên đề tài:

NGHIÊN CỨU MẠNG NƠRON NHÂN TẠO VÀ ỨNG DỤNG VÀO TRAO ĐỔI KHÓA BÍ MẬT

II- Nhiệm vụ và nội dung:

- Nghiên cứu và tìm hiểu về mạng nơron nhân tạo. Tìm hiểu về các thuật toán trao đổi khóa, mã hóa , giải mã. Từ đó xây dựng chương trình trao đổi khóa bí mật dựa vào mạng nơron nhân tạo.

+ Xây dựng mạng nơron phù hợp với bài toán

+ Huấn luyện mạng nơron để tạo ra ma trận trọng số tối ưu .

+ Tạo khóa dựa vào ma trận trọng số được đồng bộ.

+ Mã hóa và giải mã dựa vào khóa vừa được tạo.

III- Ngày giao nhiệm vụ: *18/8/2014*

IV- Ngày hoàn thành nhiệm vụ: 10/3/2015

V- Cán bộ hướng dẫn: TS LƯ NHẬT VINH

CÁN BỘ HƯỚNG DẪN
(Họ tên và chữ ký)

KHOA QUẢN LÝ CHUYÊN NGÀNH
(Họ tên và chữ ký)

LỜI CAM ĐOAN

Tôi xin cam đoan đây là công trình nghiên cứu của riêng tôi. Các số liệu, kết quả nêu trong Luận văn là trung thực và chưa từng được ai công bố trong bất kỳ công trình nào khác.

Tôi xin cam đoan rằng mọi sự giúp đỡ cho việc thực hiện Luận văn này đã được cảm ơn và các thông tin trích dẫn trong Luận văn đã được chỉ rõ nguồn gốc.

Học viên thực hiện Luận văn

(Ký và ghi rõ họ tên)

Phạm Công Thiện

LỜI CẢM ƠN

Trước tiên, tôi xin được gửi lời cảm ơn đến Ban Giám Hiệu, toàn thể cán bộ nhân viên, giảng viên trường Đại Học HUTECH, Ban lãnh đạo Phòng Quản Lý Khoa Học và Đào Tạo Sau Đại Học, khoa Công Nghệ Thông Tin đã tạo điều kiện thuận lợi cho chúng tôi học tập và nghiên cứu trong suốt học trình cao học. Xin được gửi lời cảm ơn đến tất cả quý thầy cô đã giảng dạy trong chương trình Đào tạo thạc sĩ chuyên ngành Công nghệ thông tin, khóa 2, lớp 13SCT11 - Trường Đại học Công Nghệ TPHCM, những người đã truyền đạt cho tôi những kiến thức hữu ích để làm cơ sở cho tôi thực hiện tốt luận văn này.

Với lòng kính trọng và biết ơn, tôi xin bày tỏ lời cảm ơn đến TS Lư Nhật Vinh đã tận tình hướng dẫn cho tôi trong thời gian thực hiện luận văn. Mặc dù trong quá trình thực hiện luận văn có giai đoạn không được thuận lợi, nhưng những gì thầy đã hướng dẫn, chỉ bảo đã cho tôi nhiều kinh nghiệm trong thời gian thực hiện luận văn.

Xin gửi lời cảm ơn đến bạn Phạm Duy đã giúp đỡ và tư vấn tôi về phần code cho chương trình trong suốt quá trình tôi thực hiện luận văn.

Sau cùng tôi xin gửi lời biết ơn sâu sắc đến bạn bè, gia đình, các anh chị trong tập thể lớp 13SCT11 đã luôn tạo điều kiện tốt nhất cho tôi trong suốt quá trình học cũng như thực hiện luận văn.

Do thời gian có hạn và kinh nghiệm nghiên cứu khoa học chưa nhiều nên luận văn còn nhiều thiếu sót, rất mong nhận được ý kiến góp ý của Thầy/Cô và các anh chị học viên.

TÓM TẮT

Mật mã cung cấp các dịch vụ cơ bản như là khả năng gửi thông tin giữa các thành viên tham gia, nhưng phải đảm bảo an toàn có thể ngăn chặn người khác đọc nó. Để bảo vệ nội dung chống lại một kẻ tấn công, người gửi mã hóa thông điệp của mình bằng cách sử dụng một thuật toán mã hóa đối xứng hoặc bất đối xứng. Nhưng người nhận cần phải biết được khóa của người gửi để có thể giải mã và đọc được thông điệp đó, vấn đề này thì ta có thể đạt được bằng cách sử dụng một giao thức trao đổi khóa. Diffie-Hellman là giao thức trao đổi khóa được giới thiệu, và là giao thức trao đổi khóa phổ biến. Tuy nhiên giao thức trao đổi khóa Diffie-Hellman không đảm bảo an toàn trong quá trình trao đổi khóa nếu như có kẻ thứ ba can thiệp.

Với những lý do trên tôi chọn đề tài " Nghiên cứu mạng nơron nhân tạo và ứng dụng vào trao đổi khoá bí mật" . Nghiên cứu này sẽ thay thế thuật toán Diffie-Hellman bằng mạng nơron nhân tạo (sử dụng mạng nơron Perceptron. Cách thức và giao thức trao đổi khóa cũng khác so với thuật toán Diffie-Hellman.

Cụ thể:

+ Khóa bí mật được tạo ra bởi việc đồng bộ các trọng số liên kết của mạng nơron thông qua mô hình Tree Parity Machines.

+ Khi tạo ra khóa bí mật thì thông tin sẽ được sử dụng tiêu chuẩn mã hóa tiên tiến để mã hóa và giải mã.

ABSTRACT

Cryptography is the art of mangling information into apparent unintelligibility in a manner allowing a secret method of unmangling. The basic service provided by cryptography is the ability to send information between participants in a way that prevents others from reading it. In order to protect the content against an opponent, sender encrypts histher message using a fast symmetric encryption algorithm. But receiver needs to know sender's key for reading her message, one can achieve this by using a key-exchange protocol. Diffie Hellman key exchange protocol was introduced for key exchange protocol. But Diffie Hellman is prune to man in middle attack so neural cryptography is used for key exchange.

From all reasons above I would like to choose the topic " Research neural networks and applications to serect key exchange ". This research will substitute Diffie-Hellman algorithm by neural network with more secure purpose in the key exchange process. It is also different from Diffie-Hellman algorithm about the way and key exchange protocol.

- Secret key is created by synchronizing the weight of neural network with Tree Parity Machines model.
- When we create secret key, so the information will use Advanced Encryption Standard to encryption and decryption.

MỤC LỤC

DANH MỤC CÁC TỪ VIẾT TẮT

Kí hiệu		
AES	Advanced Encryption Standard	Tiêu chuẩn mã hóa tiên tiến
ANN	Artificial Nơron Network	Mạng nơron nhân tạo
DES	Data Encryption Standard	Tiêu chuẩn mã hoá dữ liệu
GPG	GNU Privacy Guard	
IDEA	International Data Encryption Algorithm	Thuật toán mật mã hóa dữ liệu quốc tế
MLP	Multi Layer Perceptron	Mạng nơron nhiều tầng truyền thẳng
NN	Nơron Network	Mạng nơron
NIST	National Institute of Standards and Technology	Viện tiêu chuẩn và công nghệ
PE	Processing Elements	Các yếu tố xử lý
PGP	Pretty Good Privacy	
RSA	Ron Rivest, Adi Shamir và Len Adleman	Tên của thuật toán lấy từ 3 chữ cái của 3 tác giả Ron Rivest, Adi Shamir và Len Adleman
SSL	Secure Sockets Layer	
TPM	Tree Parity Machines	

DANH MỤC CÁC BẢNG

DANH MỤC CÁC HÌNH

CHƯƠNG I

MỞ ĐẦU

I. Giới thiệu

Ngày nay, không ai có thể phủ nhận vai trò cực kỳ quan trọng của máy tính trong việc nghiên cứu khoa học kỹ thuật cũng như trong đời sống. Máy tính đã làm được những điều kỳ diệu và giải được những điều tưởng chừng như nan giải. Càng ngày càng có nhiều người tự hỏi, liệu máy tính có khả năng suy nghĩ như con người được hay chưa? Chúng ta sẽ không trả lời câu hỏi này, thay vào đó chúng ta sẽ nêu ra sự khác biệt chủ yếu giữa cách làm việc của máy tính và bộ óc của con người.

Một máy tính dù có mạnh đến đâu đi nữa, đều phải làm việc theo một chương trình chính xác đã được hoạch định trước bởi các chuyên gia. Bài toán càng phức tạp thì việc lập trình càng công phu. Trong khi đó con người làm việc bằng cách học tập và rèn luyện, khi làm việc con người có khả năng liên tưởng, kết nối sự việc này với việc khác và quan trọng là họ có thể sáng tạo.

Do có khả năng liên tưởng nên con người có thể dễ dàng làm được nhiều điều mà việc lập trình cho máy tính đòi hỏi rất nhiều công sức. Từ lâu các nhà khoa học đã nhận thấy những ưu điểm của bộ óc con người và tìm cách bắt chước để thực hiện trên những máy tính, tạo cho nó khả năng học tập, phân loại… Các mạng nơron nhân tạo ra đời từ những nỗ lực đó. Nó thật sự được chú ý và nhanh chóng trở thành một hướng nghiên cứu đầy triển vọng trong mục đích xây dựng các máy thông minh tiến gần đến trí tuệ con người.

II. Lý do chọn đề tài

Mật mã cung cấp các dịch vụ cơ bản như là khả năng gửi thông tin giữa các thành viên tham gia, nhưng phải đảm bảo an toàn có thể ngăn chặn người khác đọc nó. Để bảo vệ nội dung chống lại một kẻ tấn công, người gửi mã hóa thông điệp của mình bằng cách sử dụng một thuật toán mã hóa đối xứng hoặc bất đối xứng. Nhưng người nhận cần phải biết được khóa của người gửi để có thể giải mã và đọc được thông điệp đó, vấn đề này thì ta có thể đạt được bằng cách sử dụng một giao thức trao đổi khóa. Diffie-Hellman là giao thức trao đổi khóa được giới thiệu, và là giao thức trao đổi khóa phổ biến. Tuy nhiên giao thức trao đổi khóa Diffie-Hellman không đảm bảo an toàn trong quá trình trao đổi khóa nếu như có kẻ thứ ba cố tình can thiệp. Chúng có thể đọc, hoặc thay đổi nội dung thông tin giữa các thành viên.

Bên cạnh đó với những ưu điểm của mạng nơron nhân tạo, đã ra tạo cho nó khả năng học tập, huấn luyện, phân loại….cùng với sự đa dạng, hỗn loạn giá trị và độ chính xác cao, nên mạng nơron rất thích hợp để vận dụng vào quá trình trao đổi khóa bí mật qua kênh công cộng một cách an toàn.

Với những lý do trên tôi chọn đề tài " Nghiên cứu mạng nơron nhân tạo và ứng dụng vào trao đổi khoá bí mật"

III. Mục tiêu của đề tài

Giao thức trao đổi khóa Diffie-Hellman không đảm bảo an toàn trong quá trình trao đổi khóa nếu như có kẻ thứ ba can thiệp.Vì vậy nghiên cứu này sẽ thay thế thuật toán Diffie-Hellman bằng mạng nơron nhân tạo, với mục đích bảo mật hơn trong quá trình trao đổi khóa. Khóa bí mật được tạo ra bởi việc đồng bộ các trọng số liên kết của mạng nơron thông qua mô hình Tree Parity Machines (TPM). Khi tạo ra khóa bí mật thì thông tin sẽ được sử dụng AES(128bit) để mã hóa và giải mã.

Để đạt được mục tiêu đó tôi đề xuất mô hình mạng nơron Perceptron . Và xây dựng chương trình trao đổi khóa. Cụ thể là:

Mạng nơron Perceptron : Hai máy trao đổi thông số liên tục, đến khi được đồng bộ, có nghĩa là hai ma trận trọng số liên kết của hai máy giống nhau. Từ hai ma trận đó ta tạo ra khóa.

Sau khi hai máy có khóa bí mật thì tôi dùng thuật toán AES để mã hóa và giải mã.

IV. Đối tượng và phạm vi nghiên cứu

Nghiên cứu được tiến hành trên các đối tượng: khái niệm về mật mã, khái niệm trao đổi khóa, khái niệm mạng nơron nhân tạo

Phạm vi nghiên cứu: ứng dụng mạng nơron Perceptron vào trao đổi khóa bí mật.

V. Cấu trúc của luận văn

Nội dung báo cáo gồm những chương sau:

Chương 1 Mở đầu : trình bày lý do chọn đề tài, nêu rõ đối tượng và phạm vi nghiên cứu. Sau cùng là cách tổ chức của luận văn

Chương 2 Tổng quan về mã hóa và trao đổi khóa: nêu lên các khái niệm, giao thức về mã hóa và trao đổi khóa

Chương 3 Tìm hiểu về mạng nơron nhân tạo: nêu lịch sử phát triển, khái niệm nơron và mạng nơron, phân loại và cách huấn luyện mạng nơron

Chương 4 Ứng dụng mạng nơron vào trao đổi khóa bí mật : giới thiệu mô hình Tree Parity Machines, đưa ra thuật toán trao đổi khóa bằng mạng nơron Perceptron

Chương 5 Cài đặt chương trình thử nghiệm: áp dụng thuật toán trao đổi khóa bằng mạng nơron Perceptron. Đưa giao diện chương trình.

CHƯƠNG II

TỔNG QUAN VỀ MÃ HÓA VÀ TRAO ĐỔI KHÓA

I. Mật mã học (Cryptography)

1. Giới thiệu chung:

Mật mã học là ngành khoa học ứng dụng toán học vào việc biến đổi thông tin thành một dạng khác với mục đích che giấu nội dung, ý nghĩa thông tin cần mã hoá. Đây là một ngành quan trọng và có nhiều ứng dụng trong đời sống xã hội. Ngày nay, các ứng dụng mã hóa và bảo mật thông tin đang được sử dụng ngày càng phổ biến hơn trong các lĩnh vực khác nhau trên thế giới, từ các lĩnh vực an ninh, quân sự, quốc phòng…cho đến các lĩnh vực dân sự như thương mại điện tử, ngân hàng…

Cùng với sự phát triển của khoa học máy tính và internet, các nghiên cứu và ứng dụng của khoa học mật mã ngày càng trở nên đa dạng hơn, mở ra nhiều hướng nghiên cứu chuyên sâu vào từng lĩnh vực ứng dụng đặc thù với những đặc trưng riêng.

Ứng dụng của khoa học về mật mã không chỉ đơn thuần là mã hóa và giải mã thông tin mà còn bao gồm nhiều vấn đề khác nhau cần được nghiên cứu và giải quyết: chứng thực nguồn gốc nội dung thông tin (kỹ thuật chữ ký điện tử), chứng nhận tính xác thực về người sở hữu mã khóa (chứng nhận khóa công cộng), các quy trình trao đổi thông tin và thực hiện giao dịch điện tử an toàn trên mạng… Những kết quả nghiên cứu về mật mã cũng đã được đưa vào trong các hệ thống phức tạp hơn, kết hợp với những kỹ thuật khác để đáp ứng các yêu cầu đa dạng của các hệ thống ứng dụng khác nhau trong thực tế, ví dụ như hệ thống bỏ phiếu bầu cử qua mạng, hệ thống đào tạo từ xa, hệ thống quản lý an ninh…

2. Định nghĩa:

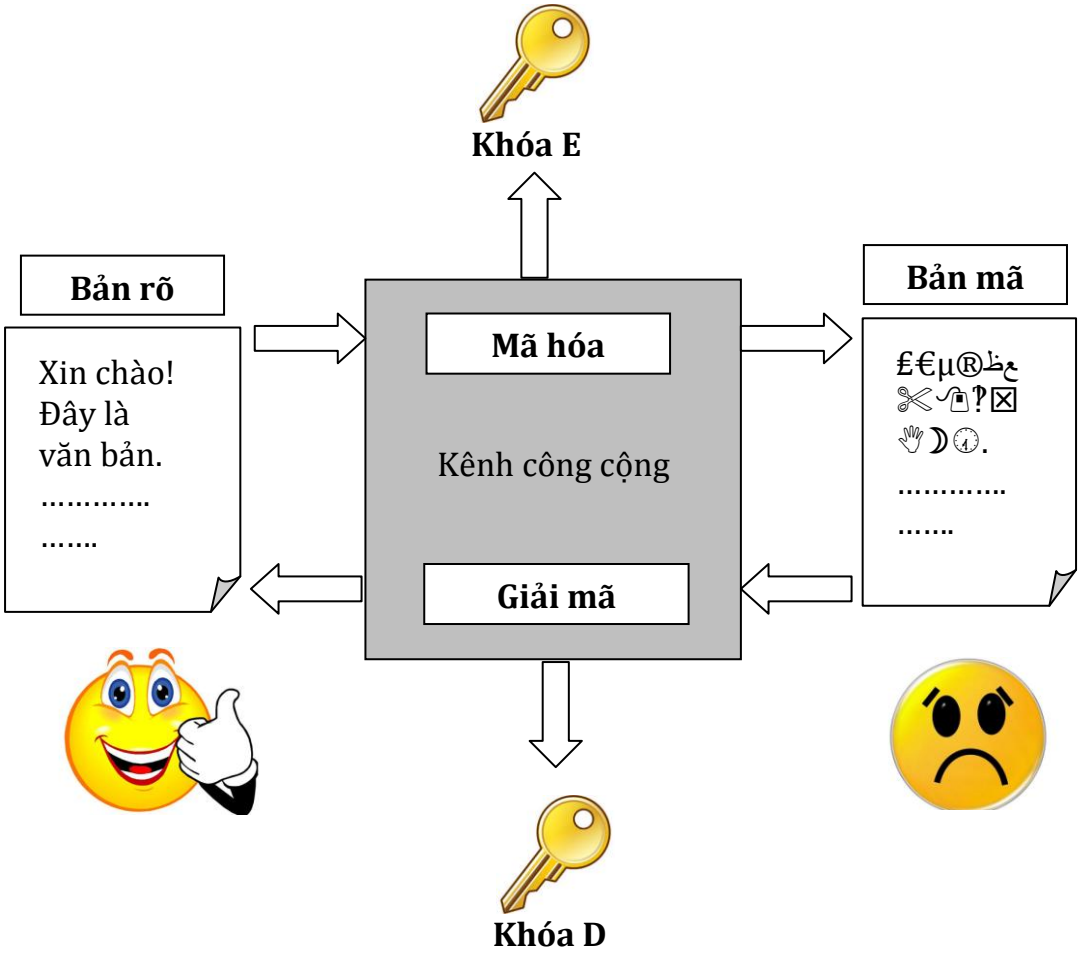
Mật mã học là sự nghiên cứu các phương pháp toán học, liên quan đến một số khía cạnh của thông tin như an toàn, sự toàn vẹn dữ liệu, sự xác nhận tồn tại và sự xác nhận tính nguyên bản của thông tin. [9]

II. Mã hóa

1. Khái niệm mã hóa (Encryption) và giải mã (Decryption):

Mã hóa: là quá trình chuyển thông tin có thể đọc được (gọi là bản rõ) thành thông tin " khó " có thể đọc được theo cách thông thường (gọi là bản mã) đó là một trong những kỹ thuật để bảo mật thông tin.

Giải mã: là quá trình chuyển thông tin ngược lại từ bản mã thành bản rõ. Thuật toán mã hóa hay giải mã là thủ tục để thực hiện mã hóa hay giải mã.



Hình 2.1 Sơ đồ hệ thống mã hóa

Mã hóa đối xứng: Khóa E = Khóa D

Mã hóa bất đối xứng: Khóa E ≠ Khóa D

Khóa mã hóa là giá trị làm cho thuật toán mã hóa thực hiện theo cách riêng biệt và sinh ra bản rõ riêng. Thông thường khóa càng lớn thì bản mã càng an toàn. Phạm vi các giá trị có thể có của khóa được gọi là Không gian khóa.

Hệ mã hóa là tập các thuật toán, các khóa nhằm che giấu thông tin cũng như làm rõ nó.

2. Các kỹ thuật mã hóa:

2.1 *Mã hóa đối xứng* (mã hóa không công khai)

Là lớp thuật toán các mã hóa trong đó việc mã hóa và giải mã đều dùng chung cho 1 khóa (secret key)

2.1.1 Các loại thuật toán khóa đối xứng

Thuật toán đối xứng có thể được chia ra làm hai thể loại, mật mã luồng (stream ciphers) và mật mã khối (block ciphers). Mật mã luồng mã hóa từng bit của thông điệp trong khi mật mã khối gộp một số bit lại và mã hóa chúng như một đơn vị. Cỡ khối được dùng thường là các khối 64 bit. Thuật toán AES được NIST công nhận tháng 12 năm 2001, sử dụng các khối gồm 128 bit.

Các thuật toán đối xứng thường không được sử dụng độc lập. Trong thiết kế của các hệ thống mật mã hiện đại, cả hai thuật toán bất đối xứng và thuật toán đối xứng được sử dụng phối hợp để tận dụng các ưu điểm của cả hai. Những hệ thống sử dụng cả hai thuật toán bao gồm những cái như SSL, PGP và GPG v.v. Các thuật toán khóa bất đối xứng được sử dụng để phân phối khóa mật cho thuật toán đối xứng có tốc độ cao hơn.

2.1.2.Tốc độ

Các thuật toán đối xứng nói chung đòi hỏi công suất tính toán ít hơn các thuật toán khóa bất đối xứng. Trên thực tế, một thuật toán khóa bất đối

xứng có khối lượng tính toán nhiều hơn gấp hằng trăm, hằng ngàn lần một thuật toán khóa đối xứng có chất lượng tương đương.

2.1.3.Hạn chế

Hạn chế của các thuật toán khóa đối xứng bắt nguồn từ yêu cầu về sự phân hưởng khóa bí mật, mỗi bên phải có một bản sao của khóa. Do khả năng các khóa có thể bị phát hiện bởi đối thủ mật mã, chúng thường phải được bảo toàn trong khi phân phối và trong khi dùng. Hậu quả của yêu cầu về việc lựa chọn, phân phối và lưu trữ các khóa một cách không có lỗi, không bị mất mát là một việc làm khó khăn, khó có thể đạt được một cách đáng tin cậy.

Để đảm bảo giao thông liên lạc an toàn cho tất cả mọi người trong một nhóm gồm n người, tổng số lượng khóa cần phải có là $\frac{n(n-1)}{2}$

Hiện nay người ta phổ biến dùng các thuật toán bất đối xứng có tốc độ chậm hơn để phân phối khóa đối xứng khi một phiên giao dịch bắt đầu, sau đó các thuật toán khóa đối xứng tiếp quản phần còn lại . Vấn đề về bảo quản sự phân phối khóa một cách đáng tin cậy cũng tồn tại ở tầng đối xứng, song ở một điểm nào đấy, người ta có thể kiểm soát chúng dễ dàng hơn. Tuy thế, các khóa đối xứng hầu như đều được sinh tạo tại chỗ.

Các thuật toán khóa đối xứng không thể dùng cho mục đích xác thực (authentication) hay mục đích chống thoái thác (non-repudiation) được.

2.2 *Mã hóa bất đối xứng* (Mã hóa công khai)

Là thuật toán trong đó việc mã hóa và giãi mã dùng hai khóa khác nhau là pulic key (khóa công khai) và private key (khóa riêng).

Nếu dùng public key để mã hóa thì private key sẽ dùng để giải mã và ngược lại

2.2.1. An toàn

Về khía cạnh an toàn, các thuật toán mật mã hóa bất đối xứng cũng không khác nhiều với các thuật toán mã hóa đối xứng. Có những thuật toán được dùng rộng rãi, có thuật toán chủ yếu trên lý thuyết; có thuật toán vẫn được xem là an toàn, có thuật toán đã bị phá vỡ… Cũng cần lưu ý là những thuật toán được dùng rộng rãi không phải lúc nào cũng đảm bảo an toàn. Một số thuật toán có những chứng minh về độ an toàn với những tiêu chuẩn khác nhau. Nhiều chứng minh gắn việc phá vỡ thuật toán với những bài toán nổi tiếng vẫn được cho là không có lời giải trong thời gian đa thức. Vì vậy, cũng giống như tất cả các thuật toán mật mã nói chung, các thuật toán mã hóa khóa công khai cần phải được sử dụng một cách thận trọng.

2.2.2. Ứng dụng

Ứng dụng rõ ràng nhất của mật mã hóa khóa công khai là bảo mật: một văn bản được mã hóa bằng khóa công khai của một người sử dụng thì chỉ có thể giải mã với khóa bí mật của người đó.

Các thuật toán tạo chữ ký số khóa công khai có thể dùng để nhận thức. Một người sử dụng có thể mã hóa văn bản với khóa bí mật của mình. Nếu một người khác có thể giải mã với khóa công khai của người gửi thì có thể tin rằng văn bản thực sự xuất phát từ người gắn với khóa công khai đó.

2.2.3. Hạn chế

Tồn tại khả năng một người nào đó có thể tìm ra được khóa bí mật. Không giống với hệ thống mật mã sử dụng một lần (one-time pad) hoặc tương đương, chưa có thuật toán mã hóa khóa bất đối xứng nào được chứng minh là an toàn trước các tấn công dựa trên bản chất toán học của thuật toán. Khả năng một mối quan hệ nào đó giữa 2 khóa hay điểm yếu của thuật toán dẫn tới cho phép giải mã không cần tới khóa hay chỉ cần khóa mã hóa vẫn chưa được loại trừ. An toàn của các thuật toán này đều dựa trên các ước

lượng về khối lượng tính toán để giải các bài toán gắn với chúng. Các ước lượng này lại luôn thay đổi tùy thuộc khả năng của máy tính và các phát hiện toán học mới.

Khả năng bị tấn công dạng kẻ tấn công đứng giữa (man in the middle attack): kẻ tấn công lợi dụng việc phân phối khóa công khai để thay đổi khóa công khai. Sau khi đã giả mạo được khóa công khai, kẻ tấn công đứng ở giữa 2 bên để nhận các gói tin, giải mã rồi lại mã hóa với khóa đúng và gửi đến nơi nhận để tránh bị phát hiện. Dạng tấn công kiểu này có thể phòng ngừa bằng các phương pháp trao đổi khóa an toàn nhằm đảm bảo nhận thực người gửi và toàn vẹn thông tin.

2.2.4. Khối lượng tính toán

Để đạt được độ an toàn tương đương đòi hỏi khối lượng tính toán nhiều hơn đáng kể so với thuật toán mật mã hóa đối xứng. Vì thế trong thực tế hai dạng thuật toán này thường được dùng bổ sung cho nhau để đạt hiệu quả cao. Trong mô hình này, một bên tham gia trao đổi thông tin tạo ra một khóa đối xứng dùng cho phiên giao dịch. Khóa này sẽ được trao đổi an toàn thông qua hệ thống mã hóa khóa bất đối xứng. Sau đó 2 bên trao đổi thông tin bí mật bằng hệ thống mã hóa đối xứng trong suốt phiên giao dịch.

III. Trao đổi khóa

1. Giới thiệu trao đổi khóa Diffie-Hellman:

Trao đổi khóa Diffie-Hellman là một phương pháp trao đổi khóa được phát minh sớm nhất trong mật mã học. Phương pháp trao đổi khóa Diffie-Hellman cho phép hai bên (người, thực thể giao tiếp) thiết lập một khóa bí mật chung để mã hóa dữ liệu sử dụng trên kênh truyền thông không an toàn mà không cần có sự thỏa thuận trước về khóa bí mật giữa hai bên. Khóa bí mật tạo ra sẽ được sử dụng để mã hóa dữ liệu với phương pháp mã hóa khóa đối xứng.

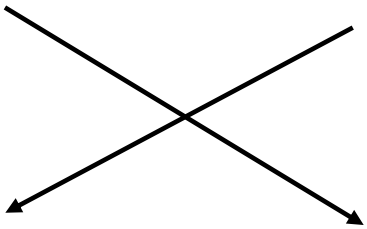
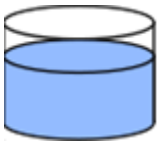
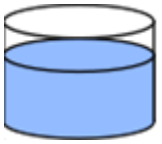
Giao thức này được công bố đầu tiên bởi Whitfield Diffie và Martin Hellman vào năm 1976. Năm 2002, Hellman đề xuất thuật toán nên được gọi là trao đổi khóa Diffie-Hellman-Merkle để ghi nhận sự đóng góp của Ralph Merkle trong phát minh lĩnh vực mật mã hóa khóa công khai (Hellman 2002) [4]

2. Giao thức trao đổi khoá Diffie-Hellman:

Diffie-Hellman thiết lập bí mật chung để sử dụng cho trao đổi dữ liệu an toàn trên một kênh truyền thông công cộng không an toàn. Sơ đồ sau đây minh họa ý tưởng cơ bản của việc trao đổi khóa thông qua ví dụ về màu sơn. Điểm chủ chốt của ý tưởng này là Alice và Bob trao đổi màu sơn bí mật thông qua hỗn hợp sơn.

- Đầu tiên Alice và Bob trộn màu đã biết chung (màu vàng) với màu bí mật riêng của mỗi người.
- Sau đó, mỗi người chuyển hỗn hợp của mình tới người kia thông qua một kênh vận chuyển công cộng.
- Khi nhận được hỗn hợp của người kia, mỗi người sẽ trộn thêm với màu bí mật của riêng mình và nhận được hỗn hợp cuối cùng.

Hỗn hợp sơn cuối cùng là hoàn toàn giống nhau cho cả hai người và chỉ có riêng hai người biết. Mấu chốt ở đây là đối với một người ngoài sẽ rất khó (về mặt tính toán) cho họ để tìm ra được bí mật chung của hai người (nghĩa là hỗn hợp cuối cùng). Alice và Bob sẽ sử dụng bí mật chung này để mã hóa và giải mã dữ liệu truyền trên kênh công cộng. Lưu ý, màu sơn đầu tiên (màu vàng) có thể tùy ý lựa chọn, nhưng được thỏa thuận trước giữa Alice và Bob. Màu sơn này cũng có thể được giả sử là không bí mật đối với người thứ ba mà không làm lộ bí mật chung cuối cùng của Alice và Bob.

Alice		Bob
	**Màu chung**	
+		+
	**Màu riêng**	
=		=
	**Trao đổi công khai** 	
		
+		+
	**Màu riêng**	
=		=
	**Màu chung**	

Bảng 2.1 Bảng trao đổi màu sơn bí mật của Alice và Bob

Giao thức được diễn giải dưới dạng toán học như sau:

Giao thức sử dụng nhóm nhân số nguyên modulo p, trong đó p số nguyên tố, và g là căn nguyên thủy mod p. Trong ví dụ dưới đây, giá trị không bí mật được viết bằng màu xanh, và giá trị bí mật viết bằng màu **đỏ**:

Alice				**Bob**		
Bí mật	Công khai	Tính	Gửi	Tính	Công khai	Bí mật
A	p,g		p,g→			B
A	p,g,A	$g^a \bmod p = A$	A→		p,g	B
A	p,g,A		←B	$g^b \bmod p = B$	p,g,A,B	B
a,s	p,g,A,B	$B^a \bmod p = s$		$A^b \bmod p = s$	p,g,A,B	b,s

Bảng 2.2 Giao thức toán học chia sẻ bí mật giữa Alice và Bob

Alice và Bob thỏa thuận sử dụng chung một số nguyên tố p=23 và căn nguyên thủy g=5.

1. Alice chọn một số nguyên bí mật a=**6**, và gửi cho Bob giá trị $A = g^a \bmod p$
   - $A = 5^6 \bmod 23$
   - $A = 15{,}625 \bmod 23$
   - $A = 8$
2. Bob chọn một số nguyên bí mật b=**15**, và gửi cho Alice giá trị $B = g^b \bmod p$
   - $B = 5^{15} \bmod 23$
   - $B = 30{,}517{,}578{,}125 \bmod 23$
   - $B = 19$

3. Alice tính **s** = B^{a} mod p

- **s** = 19^{6} mod 23
- **s** = **47,045,881** mod 23
- **s** = **2**

4. Bob tính **s** = A^{b} mod p

- **s** = 8^{15} mod 23
- **s** = **35,184,372,088,832** mod 23
- **s** = **2**

5. Như vậy Alice và Bob cùng chia sẻ bí mật chung là số **2** vì 6*15 cũng bằng 15*6.

Cả Alice và Bob đều có được giá trị chung cuối cùng vì $(g^{a})^{b} = (g^{b})^{a}$ mod p. Lưu ý rằng chỉ có a, b và $g^{ab} = g^{ba}$ mod p là được giữ bí mật. Tất cả các giá trị khác như p, g, g^{a} *mod* $p$và g^{b} *mod* p được truyền công khai. Sau khi Alice và Bob tính được bí mật chung, cả hai có thể sử dụng nó làm khóa mã hóa chung chỉ có hai người biết để gửi dữ liệu trên kênh truyền thông mở.

Trong thực tế để giao thức được an toàn, người ta sử dụng giá trị lớn hơn nhiều cho a, b, và p, vì trong ví dụ trên chỉ có tổng cộng 23 kết quả khác nhau cho n mod 23 (do đó kẻ tấn công chỉ cần thử hết 23 trường hợp là tìm ra khóa bí mật). Nếu số nguyên tố p có ít nhất 300 chữ số, còn a và b có ít nhất 100 chữ số, thì ngay cả những máy tính hiện đại nhất hiện nay cũng không thể tìm được a nếu chỉ biết g, p, g^{b} mod p và g^{a} mod p.

Lưu ý, g không cần thiết là một căn nguyên thủy có giá trị lớn. Trong thực tế người ta hay sử dụng các giá trị 2, 3 hoặc 5.

3. Hạn chế:

Trao đổi khóa Diffie-Hellman là dễ bị tấn công bởi một người đứng giữa. Trong cuộc tấn công này, khi Alice truyền các giá trị công khai với Bob thì một đối thủ Eve chặn giá trị gửi đi của Alice và gửi các giá trị riêng của mình đến Bob.Và ngược lại Bob truyền giá trị công khai của mình cho Alice thì Eve thay thế nó với giá trị của mình và gửi nó đến Alice.Do đó Eve và Alice đồng ý về một khóa chia sẻ và Eve và Bob cũng đồng ý trên một khóa chia sẻ khác. Sau khi thực hiện trao đổi này, Eve chỉ đơn giản là giải mã bất kỳ thông điệp được gửi ra bởi Alice hay Bob, và sau đó đọc và có thể thay đổi chúng trước khi tái mã hóa với khóa thích hợp và truyền chúng cho bên kia [5][8].

CHƯƠNG III

TỔNG QUAN VỀ MẠNG NƠRON NHÂN TẠO

I. Lịch sử phát triển mạng nơron

Các nghiên cứu về bộ não con người đã được tiến hành từ hàng nghìn năm nay. Cùng với sự phát triển của khoa học kĩ thuật việc con người bắt đầu nghiên cứu các nơron nhân tạo là hoàn toàn tự nhiên. Sự kiện đầu tiên đánh dấu sự ra đời của mạng nơron nhân tạo diễn ra vào năm 1943 khi nhà thần kinh học Warren McCulloch và nhà toán học Walter Pitts viết bài báo mô tả cách thức các nơron hoạt động. Họ cũng đã tiến hành xây dựng một mạng nơron đơn giản bằng các mạch điện. Các nơron của họ được xem như là các thiết bị nhị phân với ngưỡng cố định. Kết quả của các mô hình này là các hàm logic đơn giản chẳng hạn như " a OR b" hay "a AND b".

Tiếp bước các nghiên cứu này, năm 1949 Donald Hebb cho xuất bản cuốn sách Organization of Behavior. Cuốn sách đã chỉ ra rằng các nơron nhân tạo sẽ trở nên hiệu quả hơn sau mỗi lần chúng được sử dụng.

Những tiến bộ của máy tính đầu những năm 1950 giúp cho việc mô hình hóa các nguyên lý của những lý thuyết liên quan tới cách thức con người suy nghĩ đã trở thành hiện thực. Nathanial Rochester sau nhiều năm làm việc tại các phòng thí nghiệm nghiên cứu của IBM đã có những nỗ lực đầu tiên để mô phỏng một mạng nơron. Trong thời kì này tính toán truyền thống đã đạt được những thành công rực rỡ trong khi đó những nghiên cứu về nơron còn ở giai đoạn sơ khai. Mặc dù vậy những người ủng hộ triết lý "thinking machines" (các máy biết suy nghĩ) vẫn tiếp tục bảo vệ cho lập trường của mình.

Năm 1956 dự án Dartmouth nghiên cứu về trí tuệ nhân tạo (Artificial Intelligence) đã mở ra thời kỳ phát triển mới cả trong lĩnh vực trí tuệ nhân tạo lẫn mạng nơron. Tác động tích cực của nó là thúc đẩy hơn nữa sự quan

tâm của các nhà khoa học về trí tuệ nhân tạo và quá trình xử lý ở mức đơn giản của mạng nơron trong bộ não con người.

Những năm tiếp theo của dự án Dartmouth, John von Neumann đã đề xuất việc mô phỏng các nơron đơn giản bằng cách sử dụng rơle điện áp hoặc đèn chân không. Nhà sinh học chuyên nghiên cứu về nơron Frank Rosenblatt cũng bắt đầu nghiên cứu về Perceptron. Sau thời gian nghiên cứu này Perceptron đã được cài đặt trong phần cứng máy tính và được xem như là mạng nơron lâu đời nhất còn được sử dụng đến ngày nay. Perceptron một tầng rất hữu ích trong việc phân loại một tập các đầu vào có giá trị liên tục vào một trong hai lớp. Perceptron tính tổng có trọng số các đầu vào, rồi trừ tổng này cho một ngưỡng và cho ra một trong hai giá trị mong muốn có thể. Tuy nhiên Perceptron còn rất nhiều hạn chế, những hạn chế này đã được chỉ ra trong cuốn sách về Perceptron của Marvin Minsky và Seymour Papert viết năm 1969.

Năm 1959, Bernard Widrow và Marcian Hoff thuộc trường đại học Stanford đã xây dựng mô hình ADALINE (ADAptive LINear Elements) và MADALINE. (Multiple ADAptive LINear Elements). Các mô hình này sử dụng quy tắc học Least-Mean-Squares (LMS: Tối thiểu bình phương trung bình). MADALINE là mạng nơron đầu tiên được áp dụng để giải quyết một bài toán thực tế. Nó là một bộ lọc thích ứng có khả năng loại bỏ tín hiệu dội lại trên đường dây điện thoại. Ngày nay mạng nơron này vẫn được sử dụng trong các ứng dụng thương mại.

Năm 1974 Paul Werbos đã phát triển và ứng dụng phương pháp học lan truyền ngược (back-propagation). Tuy nhiên phải mất một vài năm thì phương pháp này mới trở nên phổ biến. Các mạng lan truyền ngược được biết đến nhiều nhất và được áp dụng rộng rãi nhất nhất cho đến ngày nay.

Thật không may, những thành công ban đầu này khiến cho con người nghĩ quá lên về khả năng của các mạng nơron. Chính sự cường điệu quá mức

đã có những tác động không tốt đến sự phát triển của khoa học và kỹ thuật thời bấy giờ khi người ta lo sợ rằng đã đến lúc máy móc có thể làm mọi việc của con người. Những lo lắng này khiến người ta bắt đầu phản đối các nghiên cứu về mạng nơron. Thời kì tạm lắng này kéo dài đến năm 1981.

Năm 1982 trong bài báo gửi tới viện khoa học quốc gia, John Hopfield bằng sự phân tích toán học rõ ràng, mạch lạc, ông đã chỉ ra cách thức các mạng nơron làm việc và những công việc chúng có thể thực hiện được. Cống hiến của Hopfield không chỉ ở giá trị của những nghiên cứu khoa học mà còn ở sự thúc đẩy trở lại các nghiên cứu về mạng nơron.

Cũng trong thời gian này, một hội nghị với sự tham gia của Hoa Kỳ và Nhật Bản bàn về việc hợp tác/cạnh tranh trong lĩnh vực mạng nơron đã được tổ chức tại Kyoto, Nhật Bản. Sau hội nghị, Nhật Bản đã công bố những nỗ lực của họ trong việc tạo ra máy tính thế hệ thứ 5. Tiếp nhận điều đó, các tạp chí định kỳ của Hoa Kỳ bày tỏ sự lo lắng rằng nước nhà có thể bị tụt hậu trong lĩnh vực này. Vì thế, ngay sau đó, Hoa Kỳ nhanh chóng huy động quỹ tài trợ cho các nghiên cứu và ứng dụng mạng nơron.

Năm 1985, viện vật lý Hoa Kỳ bắt đầu tổ chức các cuộc họp hàng năm về mạng nơron ứng dụng trong tin học (Neural Networks for Computing).

Ngày nay, không chỉ dừng lại ở mức nghiên cứu lý thuyết, các nghiên cứu ứng dụng mạng nơron để giải quyết các bài toán thực tế được diễn ra ở khắp mọi nơi. Các ứng dụng mạng nơron ra đời ngày càng nhiều và ngày càng hoàn thiện hơn. Điển hình là các ứng dụng: xử lý ngôn ngữ (Language Processing), nhận dạng kí tự (Character Recognition), nhận dạng tiếng nói (Voice Recognition), nhận dạng mẫu (Pattern Recognition), xử lý tín hiệu (Signal Processing), Lọc dữ liệu (Data Filtering),…..

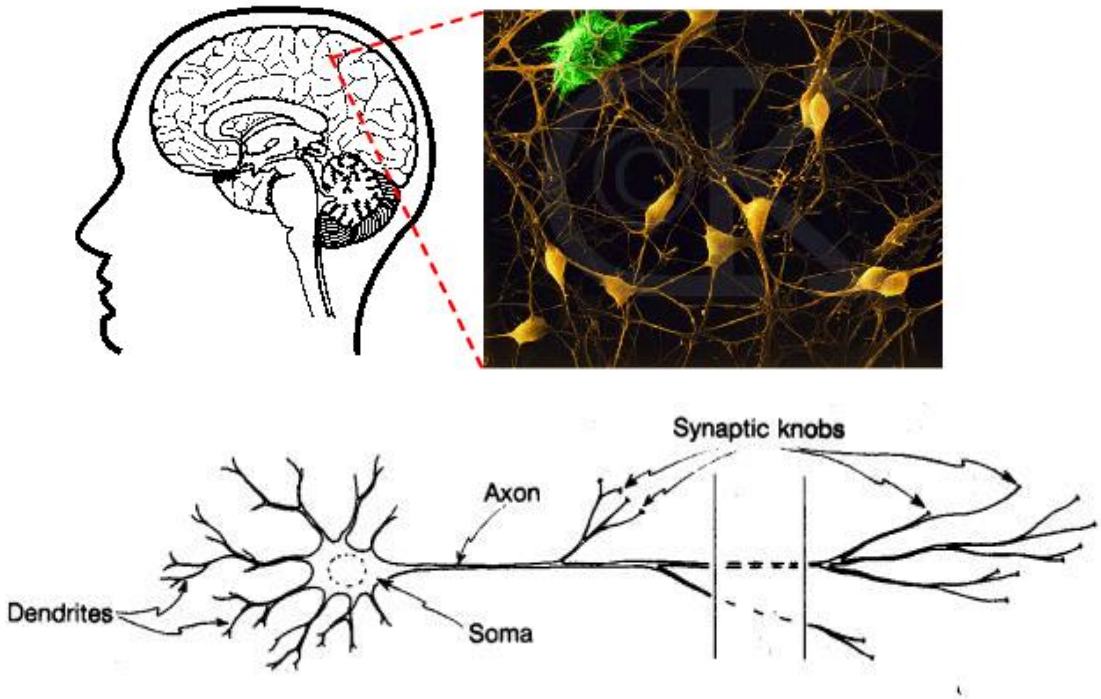
II. Khái niệm về mạng nơron

1. Tìm hiểu về nơron

1.1 *Nơron sinh học*:

Qua quá trình nghiên cứu về bộ não, người ta thấy rằng: bộ não con người bao gồm khoảng 10^{11} nơron tham gia vào khoảng 10^{15} kết nối trên các đường truyền. Mỗi đường truyền này dài khoảng hơn một mét. Các nơron có nhiều đặc điểm chung với các tế bào khác trong cơ thể, ngoài ra chúng còn có những khả năng mà các tế bào khác không có được, đó là khả năng nhận, xử lý và truyền các tín hiệu điện hóa trên các đường mòn nơron, các con đường này tạo nên hệ thống giao tiếp của bộ não.

Nơron sinh học được cấu tạo từ những thành phần chính sau: Soma, Dendrites, Axon như hình 3.1.



Hình 3.1 Mô hình nơron sinh học

(Trích từ 'Giới thiệu mạng nơron' của tác giả Phạm Nguyên Khang)

Soma là thân của nơron .

Các Dendrites là các dây mảnh, dài, gắn liền với soma, chúng truyền dữ liệu (dưới dạng xung điện thế) đến cho soma xử lý. Bên trong soma các dữ liệu đó được tổng hợp lại.

Một loại dây dẫn tín hiệu khác cũng gắn với soma đó là các axon. Khác với dendrites, axon có khả năng phát các xung điện thế, chúng là các dây dẫn tín hiệu đi từ nơron đến các nơi khác. Chỉ khi nào điện thế trong soma vượt quá một giá trị ngưỡng nào đó thì axon mới phát xung điện thế, còn nếu không thì nó ở trạng thái nghỉ.

Axon nối với các Dendrites của một nơron khác qua một mối nối đặc biệt gọi là Synaptic knobs. Khi điện thế của synaptic knobs tăng lên do các xung phát ra từ axon, thì synaptic knobs sẽ tạo ra một số chất hóa học mà các chất này sẽ mở ‘ cửa’ trên dendrites cho các ions truyền qua. Chính dóng ions này làm thay đổi điện thế trên dendrites, tạo ra các sung dữ liệu lan truyền đến các nơron khác.

Có thể tóm tắt hoạt động của một nơron như sau : nơron lấy tổng tất cả các điện thế vào mà nó nhận được, và phát ra một xung điện thế nếu tổng ấy lớn hơn một ngưỡng nào đó. Các nơron nối với nhau bởi các synaptic knobs. Synaptic knobs được gọi là mạnh khi nó cho phép truyền dẫn dễ dàng tín hiệu qua các nơron khác. Ngược lại, một synaptic knobs yếu sẽ truyền dẫn tín hiệu rất khó khăn.

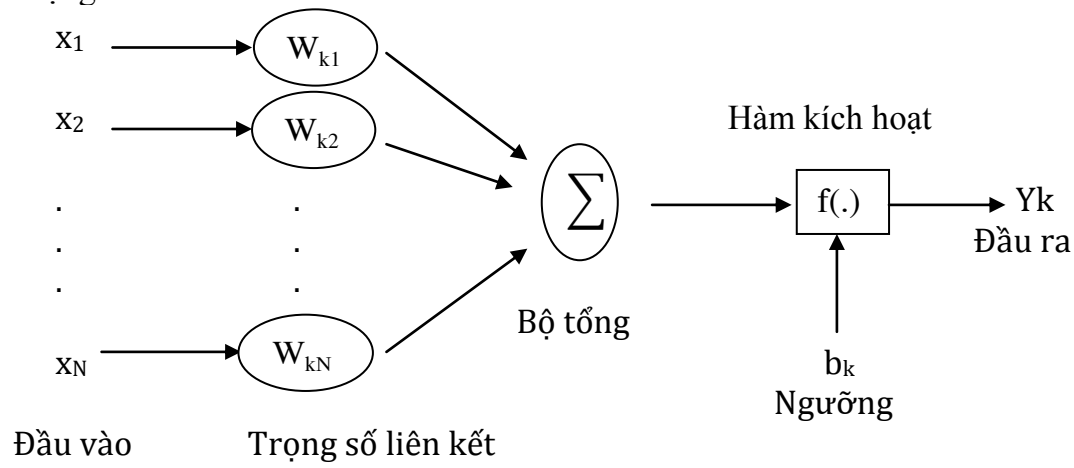
Các synaptic knobs đóng vai trò rất quan trọng trong sự học tập. Khi chúng ta học tập thì hoạt động của các synaptic knobs tăng cường, tạo nên nhiều liên kết mạnh giữa các nơron. Có thể nói người nào học giỏi thì càng có nhiều synaptic knobs và các synaptic knobs ấy mạnh mẽ, nói cách khác thì liên kết giữa các nơron càng nhiều, càng nhạy bén.

Dựa trên những hiểu biết về nơron sinh học, con người xây dựng nơron nhân tạo với hy vọng tạo nên một mô hình có sức mạnh như bộ não.

1.2 *Nơron nhân tạo*:

Nơron nhân tạo là một đơn vị tính toán có nhiều đầu vào và một đầu ra, mỗi đầu vào đến từ một liên kết. Đặc trưng của nơron là một hàm kích hoạt phi tuyến tính chuyển đổi tổ hợp tuyến tính của tất cả các tín hiệu đầu vào thành tín hiệu đầu ra. Hàm kích hoạt này đảm bảo tính chất phi tuyến tính cho tính toán của mạng nơron.

Một nơron là một đơn vị xử lý thông tin và là thành phần cơ bản của một mạng nơron.



Hình 3.2 Mô hình nơron nhân tạo

Các thành phần cơ bản của một nơron nhân tạo bao gồm:

♦ Tập các đầu vào: Là các tín hiệu vào (input signals) của nơron, các tín hiệu này thường được đưa vào dưới dạng một vector N chiều.

♦ Tập các liên kết: Mỗi liên kết được thể hiện bởi một trọng số (gọi là trọng số liên kết). Trọng số liên kết giữa tín hiệu vào thứ j với nơron k thường được kí hiệu là w_{kj}. Thông thường, các trọng số này được khởi tạo một cách ngẫu nhiên ở thời điểm khởi tạo mạng và được cập nhật liên tục trong quá trình huấn luyện.

♦ Bộ tổng (Summing function): Thường dùng để tính tổng của tích các đầu vào với trọng số liên kết của nó.

♦ Ngưỡng (còn gọi là một độ lệch - bias): Ngưỡng này thường được đưa vào như một thành phần của hàm kích hoạt.

♦ Hàm kích hoạt (Transfer function) : Hàm này được dùng để giới hạn phạm vi đầu ra của mỗi nơron. Nó nhận đầu vào là kết quả của hàm tổng và ngưỡng đã cho. Thông thường, phạm vi đầu ra của mỗi nơron được giới hạn trong đoạn [0,1] hoặc [-1, 1]. Các hàm kích hoạt rất đa dạng, có thể là các hàm tuyến tính hoặc phi tuyến. Việc lựa chọn hàm kích hoạt nào là tuỳ thuộc vào từng bài toán và kinh nghiệm của người thiết kế mạng.

♦ Đầu ra: Là tín hiệu đầu ra của một nơron, với mỗi nơron sẽ có tối đa là một đầu ra.

Tên hàm	Công thức
Hardlim	a = 0 với n < 0 a = 1 với n ≥ 0
Hardlims (SIGN)	a = -1 với n < 0 a = 1 với n ≥ 0
Purelin	a = n
Satlin	a = 0 với n < 0 a = n với 0 ≤ n ≤ 1 a = 1 với n > 1
Satlins	a = -1 với n < 0 a = n với 0 ≤ n ≤ 1 a = 1 với n > 1
Tansig	$a = \frac{e^n - e^{-n}}{1 + e^{-n}}$

Poslin	a = 0 với n < 0 a = n với n ≥ 0
Compet	a = 1 với nơron có n lớn nhất a = 0 với các nơron còn lại
Logsig	$a = \frac{1}{1+e^{-n}}$

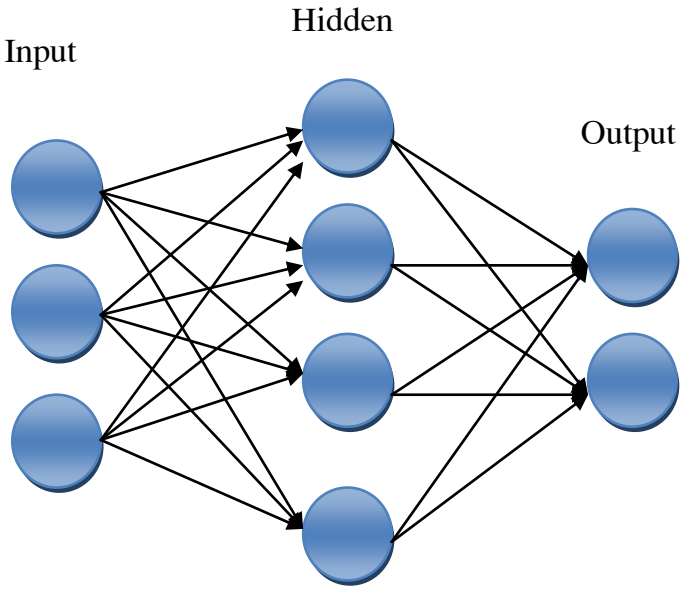
Bảng 3.1 Một số hàm kích hoạt cơ bản trong mạng nơron

2. Mạng nơron nhân tạo

Mạng nơron nhân tạo, Artificial Nơron Network (ANN) gọi tắt là mạng nơron, nơron network, là một mô hình xử lý thông tin phỏng theo cách thức xử lý thông tin của các hệ nơron sinh học. Nó được tạo lên từ một số lượng lớn các phần tử (gọi là phần tử xử lý hay nơron) kết nối với nhau thông qua các liên kết (gọi là trọng số liên kết) làm việc như một thể thống nhất để giải quyết một vấn đề cụ thể nào đó.

Một mạng nơron nhân tạo được cấu hình cho một ứng dụng cụ thể (nhận dạng mẫu, phân loại dữ liệu, ...) thông qua một quá trình huấn luyện từ tập các mẫu. Về bản chất học chính là quá trình hiệu chỉnh trọng số liên kết giữa các nơron.

Là một hệ thống bao gồm nhiều phần tử xử lý đơn giản, giống như nơron thần kinh của não người, hoạt động song song và được nối với nhau bởi các liên kết nơron. Mỗi liên kết kèm theo một trọng số liên kết.



Hình 3.3 Mô hình đơn giản về một ANN

Mô hình một ANN trên gồm 3 lớp: lớp nhập (Input), lớp ẩn (Hidden) lớp xuất (Output). Mỗi nút trong lớp nhập nhận một giá trị của một biến độc lập và chuyển vào lớp mạng.

III. Đặc trưng của mạng nơron

1. Tính phi tuyến:

Một nơron có thể tính toán một cách tuyến tính hay phi tuyến tính. Một mạng nơron, cấu thành bởi sự kết nối các nơron phi tuyến tính thì tự nó sẽ có tính phi tuyến. Hơn nữa, điều đặc biệt là tính phi tuyến này được phân tán trên toàn mạng. Tính phi tuyến là một thuộc tính rất quan trọng, nhất là khi các cơ chế vật lý sinh ra các tín hiệu đầu vào (ví dụ tín hiệu tiếng nói) vốn là phi tuyến.

2. Tính chất tương ứng đầu vào đầu ra:

Mặc dù khái niệm "học" hay "huấn luyện" chưa được bàn đến nhưng để hiểu được mối quan hệ đầu vào – đầu ra của mạng nơron, chúng ta sẽ đề

cập sơ qua về khái niệm này. Một mô hình học phổ biến được gọi là học với một người dạy hay học có giám sát liên quan đến thay đổi các trọng số liên kết của mạng nơron bằng việc áp dụng một tập hợp các mẫu tích lũy hay các ví dụ tích lũy. Mỗi một ví dụ tích lũy bao gồm một tín hiệu đầu vào và một đầu ra mong muốn tương ứng. Mạng nơron nhận một ví dụ lấy một cách ngẫu nhiên từ tập nói trên tại đầu vào của nó, và các trọng số liên kết của mạng được biến đổi sao cho có thể cực tiểu hóa sự sai khác giữa đầu ra mong muốn và đầu ra thực sự của mạng theo một tiêu chuẩn thống kê thích hợp. Sự tích lũy của mạng được lặp lại với nhiều ví dụ trong tập hợp cho tới khi mạng đạt tới mạng trạng thái ổn định mà ở đó không có một sự thay đổi đáng kể nào của các trọng số liên kết. Các ví dụ tích lũy được áp dụng trước có thể được áp dụng lại trong thời gian của phiên tích lũy nhưng theo một thứ tự khác. Như vậy mạng nơron học từ các ví dụ bằng cách xây dựng nên một tương ứng đầu vào-đầu ra cho vấn đề cần giải quyết.

3. Tính chất thích nghi.

Các mạng nơron có một khả năng mặc định là biến đổi các trọng số liên kết tùy theo sự thay đổi của môi trường xung quanh. Đặc biệt, một mạng nơron đã được tích lũy để hoạt động trong một môi trường xác định có thể được tích lũy lại một cách dễ dàng khi có những thay đổi nhỏ của các điều kiện môi trường hoạt động.

4. Tính chất đưa ra lời giải có bằng chứng.

Trong ngữ cảnh phân loại mẫu, một mạng nơron có thể được thiết kế để đưa ra thông tin không chỉ về mẫu được phân loại, mà còn về sự tin cậy của quyết định đã được thực hiện. Thông tin này có thể được sử dụng để loại bỏ các mẫu mơ hồ hay nhập nhằng.

5. Tính chất chấp nhận sai sót.

Một mạng nơron, được cài đặt dưới dạng phần cứng, vốn có khả năng chấp nhận lỗi, hay khả năng tính toán thô, với ý nghĩa là tính năng của nó chỉ thoái hóa khi có những điều kiện hoạt động bất lợi. Ví dụ, nếu một nơron hay các liên kết kết nối của nó bị hỏng, việc nhận dạng lại một mẫu được lưu trữ sẽ suy giảm về chất lượng.

6. Khả năng cài đặt VLSI (Very-large-scale-intergrated).

Bản chất song song đồ sộ của một mạng nơron làm cho nó rất nhanh trong tính toán đối với một số công việc. Đặc tính này cũng tạo ra cho một mạng nơron khả năng phù hợp cho việc cài đặt sử dụng kỹ thuật Very-large-scale-intergrated (VLSI). Kỹ thuật này cho phép xây dựng những mạch cứng tính toán song song quy mô lớn. Chính vì vậy mà ưu điểm nổi bật của VLSI là mang lại những phương tiện hữu hiệu để có thể xử lý được những hành vi có độ phức tạp cao.

7. Tính chất đồng dạng trong phân tích và thiết kế.

Về cơ bản, các mạng nơron có tính chất chung như là các bộ xử lý thông tin. Chúng ta nêu ra điều này với cùng ý nghĩa cho tất cả các lĩnh vực có liên quan tới việc ứng dụng mạng nơron. Đặc tính này thể hiện ở một số điểm như sau:

Các nơron, dưới dạng này hoặc dạng khác, biểu diễn một thành phần chung cho tất cả các mạng nơron.

Tính thống nhất này đem lại khả năng chia sẻ các lý thuyết và các thuật toán học trong nhiều ứng dụng khác nhau của mạng nơron.

Các mạng tổ hợp (modular) có thể được xây dựng thông qua một sự tích hợp các mô hình khác nhau.

IV. Phân loại mạng nơron nhân tạo

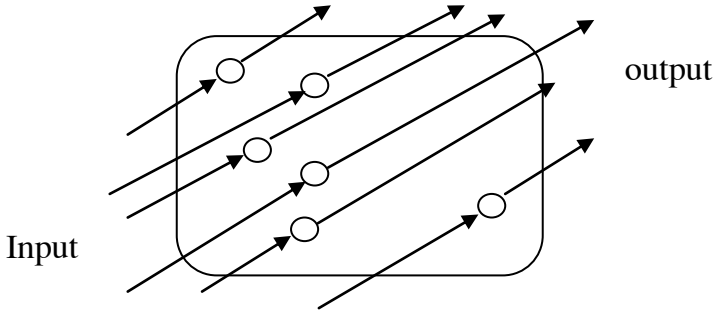
Mặc dù mỗi nơron đơn lẻ có thể thực hiện những chức năng xử lý thông tin nhất định, sức mạnh của tính toán nơron chủ yếu có được nhờ sự kết hợp các nơron trong một kiến trúc thống nhất. Một mạng nơron là một mô hình tính toán được xác định qua các tham số: kiểu nơron (như là các nút nếu ta coi cả mạng nơron là một đồ thị), kiến trúc kết nối (sự tổ chức kết nối giữa các nơron) và thuật toán học (thuật toán dùng để huấn luyện cho mạng).

Về bản chất một mạng nơron có chức năng như là một hàm ánh xạ F: $X \rightarrow Y$, trong đó X là không gian trạng thái đầu vào (input state space) và Y là không gian trạng thái đầu ra (output state space) của mạng. Các mạng chỉ đơn giản là làm nhiệm vụ ánh xạ các vector đầu vào $x \in X$ sang các vector đầu ra $y \in Y$ thông qua "bộ lọc" các trọng số. Tức là $y = F(x) = s(W, x)$, trong đó W là ma trận trọng số liên kết. Hoạt động của mạng thường là các tính toán số thực trên các ma trận.

1. Các kiểu mô hình mạng nơron

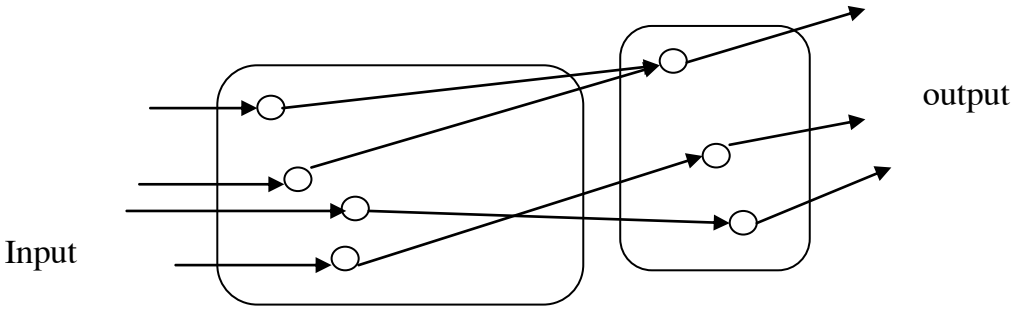
Cách thức kết nối các nơron trong mạng xác định kiến trúc của mạng. Các nơron trong mạng có thể kết nối đầy đủ có nghĩa là mỗi nơron đều được kết nối với tất cả các nơron khác, hoặc kết nối cục bộ chẳng hạn chỉ kết nối giữa các nơron trong các tầng khác nhau. Người ta chia ra hai loại kiến trúc mạng chính:

♦ Tự kết hợp (autoassociative): là mạng có các nơron đầu vào cũng là các nơron đầu ra. Mạng Hopfield là một kiểu mạng tự kết hợp.



Hình 3.4 Mạng tự kết hợp

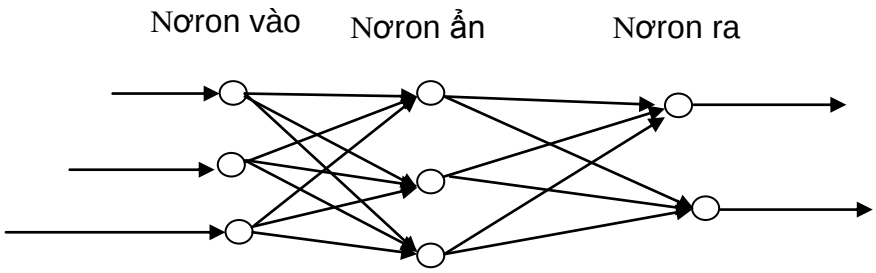
♦ Kết hợp khác kiểu (heteroassociative): là mạng có tập nơron đầu vào và đầu ra riêng biệt. Perceptron, các mạng Perceptron nhiều tầng (MLP: Multi Layer Perceptron), mạng Kohonen, … thuộc loại này.



Hình 3.5 Mạng kết hợp khác kiểu

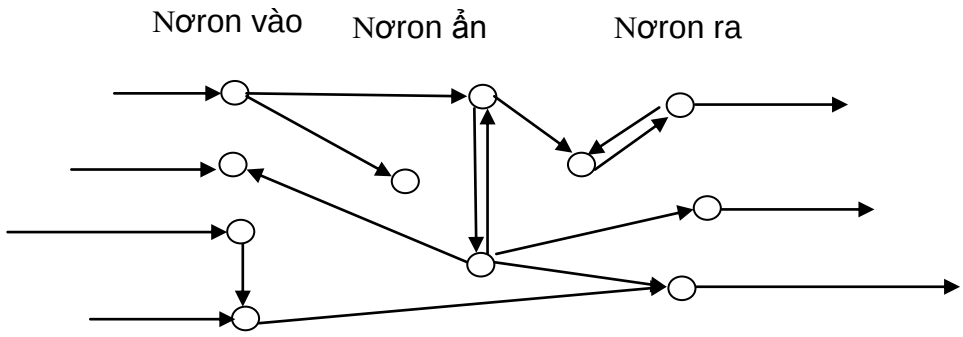
Ngoài ra tùy thuộc vào mạng có các kết nối ngược (feedback connections) từ các nơron đầu ra tới các nơron đầu vào hay không, người ta chia ra làm 2 loại kiến trúc mạng:

♦ Kiến trúc truyền thẳng (feedforward architechture): là kiểu kiến trúc mạng không có các kết nối ngược trở lại từ các nơron đầu ra về các nơron đầu vào; mạng không lưu lại các giá trị output trước và các trạng thái kích hoạt của nơron. Các mạng nơron truyền thẳng cho phép tín hiệu di chuyển theo một đường duy nhất; từ đầu vào tới đầu ra, đầu ra của một tầng bất kì sẽ không ảnh hưởng tới tầng đó. Các mạng kiểu Perceptron là mạng truyền thẳng.



Hình 3.6 Mạng truyền thẳng

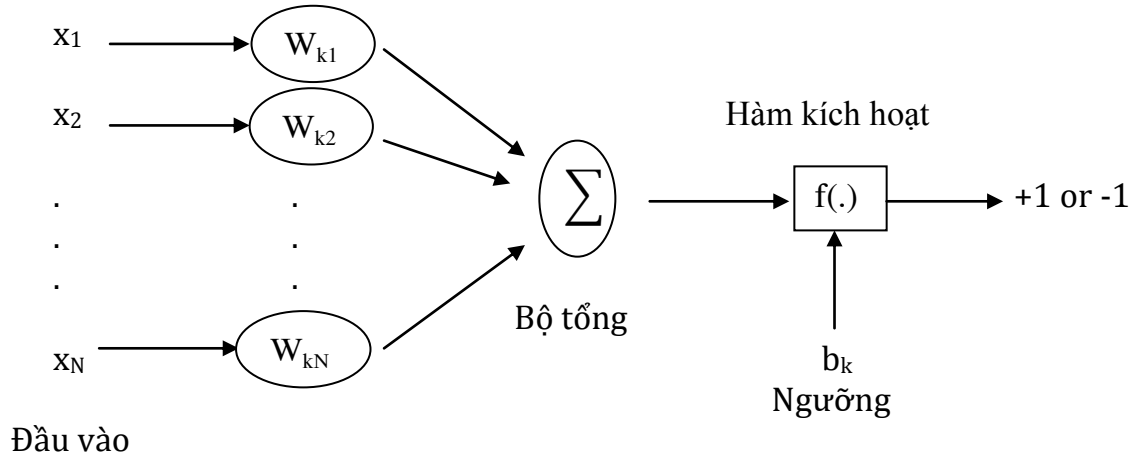
♦ Kiến trúc phản hồi (Feedback architecture): là kiểu kiến trúc mạng có các kết nối từ nơron đầu ra tới nơron đầu vào. Mạng lưu lại các trạng thái trước đó, và trạng thái tiếp theo không chỉ phụ thuộc vào các tín hiệu đầu vào mà còn phụ thuộc vào các trạng thái trước đó của mạng. Mạng Hopfield thuộc loại này.



Hình 3.7 Mạng phản hồi

2. Perceptron

Perceptron là mạng nơron đơn giản nhất, nó chỉ gồm một nơron, nhận đầu vào là vector có các thành phần là các số thực và đầu ra là một trong hai giá trị +1 hoặc -1.



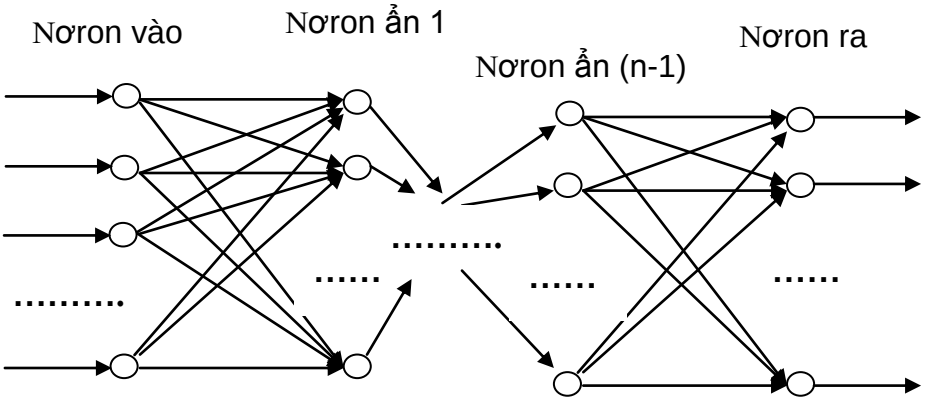
Hình 3.8 Mạng Perceptron

Đầu ra của mạng được xác định như sau: mạng lấy tổng có trọng số các thành phần của vector đầu vào, kết quả này cùng ngưỡng b được đưa vào hàm truyền (Perceptron dùng hàm Hard-limit làm hàm truyền) và kết quả của hàm truyền sẽ là đầu ra của mạng.

Perceptron cho phép phân loại chính xác trong trường hợp dữ liệu có thể phân chia tuyến tính (các mẫu nằm trên hai mặt đối diện của một siêu phẳng). Nó cũng phân loại đúng đầu ra các hàm AND, OR và các hàm có dạng đúng khi n trong m đầu vào của nó đúng ($n \leq m$). Nó không thể phân loại được đầu ra của hàm XOR.

3. Mạng nhiều tầng truyền thẳng (MLP):

Mô hình mạng nơron được sử dụng rộng rãi nhất là mô hình mạng nhiều tầng truyền thẳng (MLP: Multi Layer Perceptron). Một mạng MLP tổng quát là mạng có n ($n \geq 2$) tầng (thông thường tầng đầu vào không được tính đến): trong đó gồm một tầng đầu ra (tầng thứ n) và (n-1) tầng ẩn.



Hình 3.9 Mạng MLP tổng quát

Kiến trúc của một mạng MLP tổng quát có thể mô tả như sau:

♦ Đầu vào là các vector (x1, x2, ..., xp) trong không gian p chiều, đầu ra là các vector (y1, y2, ..., yq) trong không gian q chiều. Đối với các bài toán phân loại, p chính là kích thước của mẫu đầu vào, q chính là số lớp cần phân loại. Xét ví dụ trong bài toán nhận dạng chữ số: với mỗi mẫu ta lưu tọa độ (x,y) của 8 điểm trên chữ số đó, và nhiệm vụ của mạng là phân loại các mẫu này vào một trong 10 lớp tương ứng với 10 chữ số 0, 1, …, 9. Khi đó p là kích thước mẫu và bằng 8 x 2 = 16; q là số lớp và bằng 10.

♦ Mỗi nơron thuộc tầng sau liên kết với tất cả các nơron thuộc tầng liền trước nó.

♦ Đầu ra của nơron tầng trước là đầu vào của nơron thuộc tầng liền sau nó.

Hoạt động của mạng MLP như sau: tại tầng đầu vào các nơron nhận tín hiệu vào xử lý (tính tổng trọng số, gửi tới hàm truyền) rồi cho ra kết quả (là kết quả của hàm truyền); kết quả này sẽ được truyền tới các nơron thuộc tầng ẩn thứ nhất; các nơron tại đây tiếp nhận như là tín hiệu đầu vào, xử lý và gửi kết quả đến tầng ẩn thứ 2;…; quá trình tiếp tục cho đến khi các nơron thuộc tầng ra cho kết quả.

Một số kết quả đã được chứng minh:

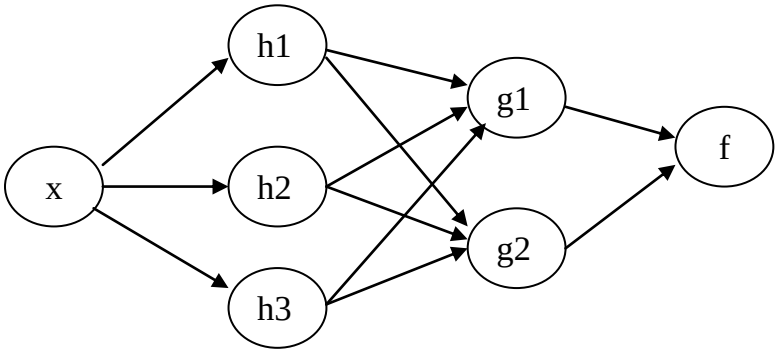
♦ Bất kì một hàm Boolean nào cũng có thể biểu diễn được bởi một mạng MLP 2 tầng trong đó các nơron sử dụng hàm truyền sigmoid.

♦ Tất cả các hàm liên tục đều có thể xấp xỉ bởi một mạng MLP 2 tầng sử dụng hàm truyền sigmoid cho các nơron tầng ẩn và hàm truyền tuyến tính cho các nơron tầng ra với sai số nhỏ tùy ý.

♦ Mọi hàm bất kỳ đều có thể xấp xỉ bởi một mạng MLP 3 tầng sử dụng hàm truyền sigmoid cho các nơron tầng ẩn và hàm truyền tuyến tính cho các nơron tầng ra.

V. Xây dựng mạng nơron

Về cơ bản ta có thể hiểu mạng nơron là một đồ thị có hướng như hình 2.5.1 Trong đó các đỉnh của đồ thị là các nơron và các cạnh của đồ thị là các liên kết giữa các nơron.



Hình 3.10 Sơ đồ đồ thị có hướng đơn giản

Vì vậy để xây dựng một mạng nơron ta xây dựng một đồ thị có hướng: số đỉnh của đồ thị bằng số nơron trong mạng, giá trị của các cạnh chính là trọng số liên kết nơron.

VI. Huấn luyện mạng nơron

Huấn luyện là quá trình thay đổi hành vi của các vật theo một cách nào đó làm cho chúng có thể thực hiện tốt hơn trong tương lai.

Một mạng nơron được huyấn luyện sao cho với một tập các vector đầu vào X, mạng có khả năng tạo ra tập các vector đầu ra mong muốn Y của nó. Tập X được sử dụng cho huấn luyện mạng được gọi là tập huấn luyện (training set). Các phần tử x thuộc X được gọi là các mẫu huấn luyện (training example). Quá trình huấn luyện bản chất là sự thay đổi các trọng số liên kết của mạng. Trong quá trình này, các trọng số của mạng sẽ hội tụ dần tới các giá trị sao cho với mỗi vector đầu vào x từ tập huấn luyện, mạng sẽ cho ra vector đầu ra y như mong muốn

Có ba phương pháp huấn luyện phổ biến là huấn luyện có giám sát , huấn luyện không giám sát và huấn luyện tăng cường.

1. Huấn luyện có giám sát

Là quá trình huấn luyện có sự tham gia giám sát của một "thầy giáo". Cũng giống như việc ta dạy một em nhỏ các chữ cái. Ta đưa ra một chữ "a" và bảo với em đó rằng đây là chữ "a". Việc này được thực hiện trên tất cả các mẫu chữ cái. Sau đó khi kiểm tra ta sẽ đưa ra một chữ cái bất kì (có thể viết hơi khác đi) và hỏi em đó đây là chữ gì?

Với huấn luyện có giám sát, tập mẫu huấn luyện được cho dưới dạng

$D = \{(x,t) \mid (x,t) \in [IR^N \times R^K]\}$ trong đó: $x = (x_1, x_2, ..., x_N)$ là vector đặc trưng N chiều của mẫu huấn luyện và $t = (t_1, t_2, ..., t_K)$ là vector mục tiêu K chiều tương ứng, nhiệm vụ của thuật toán là phải thiết lập được một cách tính toán trên mạng như thế nào đó để sao cho với mỗi vector đặc trưng đầu vào thì sai số giữa giá trị đầu ra thực sự của mạng và giá trị mục tiêu tương ứng là nhỏ nhất. Chẳng hạn mạng có thể học để xấp xỉ một hàm t = f(x) biểu diễn mối quan hệ trên tập các mẫu huấn luyện (x, t).

Như vậy với huấn luyện có giám sát, số lớp cần phân loại đã được biết trước. Nhiệm vụ của thuật toán là phải xác định được một cách thức phân lớp sao cho với mỗi vector đầu vào sẽ được phân loại chính xác vào lớp của nó.

2. Huấn luyện không giám sát

Là việc huấn luyện không cần có bất kỳ một sự giám sát nào.

Trong bài toán huấn luyện không giám sát, tập dữ liệu huấn luyện được cho dưới dạng: D = $\{(x_1, x_2, ..., x_N)\}$, với $(x_1, x_2, ..., x_N)$, là vector đặc trưng của mẫu huấn luyện. Nhiệm vụ của thuật toán là phải phân chia tập dữ liệu D thành các nhóm con, mỗi nhóm chứa các vector đầu vào có đặc trưng giống nhau.

Như vậy với huấn luyện không giám sát, số lớp phân loại chưa được biết trước, và tùy theo tiêu chuẩn đánh giá độ tương tự giữa các mẫu mà ta có thể có các lớp phân loại khác nhau.

3. Huấn luyện tăng cường

Là sự tổ hợp của cả hai mô hình trên. Phương pháp này cụ thể như sau: với vector đầu vào, quan sát vector đầu ra do mạng tính được. Nếu kết quả được xem là "tốt" thì mạng sẽ được thưởng theo nghĩa tăng các trọng số kết nối lên; ngược lại mạng sẽ bị phạt, các trọng số kết nối không thích hợp sẽ được giảm xuống. Do đó huấn luyện tăng cường là huấn luyện theo nhà phê bình (critic), ngược với huấn luyện có giám sát là huấn luyện theo thầy giáo (teacher).

VII. Biểu diễn tri thức cho mạng nơron

Chúng ta có thể đưa ra định nghĩa về tri thức như sau:

Tri thức chính là thông tin được lưu trữ hay các mô hình được con người và máy móc sử dụng để biểu diễn thế giới thực, phán đoán về thế giới và có những đáp ứng phù hợp với thế giới bên ngoài. Tri thức bao gồm các sự kiện và luật.

Các đặc tính cơ bản của biểu diễn tri thức là:

- ✓ Thông tin gì thực sự được biểu diễn.
- ✓ Làm thế nào thông tin được mã hóa một cách vật lý cho việc sử dụng sau này. Trong các ứng dụng thực tế của các máy tính thông minh, có thể nói rằng một giải pháp tốt phụ thuộc vào một biến tri thức tốt. Điều đó cũng đúng với các mạng nơron, một lớp đặc biệt của các máy thông minh. Tuy nhiên, các dạng biểu diễn có thể từ các đầu vào thành các tham số bên trong của mạng là rất đa dạng, và có khuynh hướng là cho việc tìm ra một giải pháp thích hợp nhằm biểu diễn tri thức bằng phương tiện mạng nơron trở nên một sự thách thức về thiết kế.

Ở đây cần nhấn mạnh rằng mạng nơron lưu trữ thông tin về thế giới thực bằng chính bản thân cấu trúc của nó kể cả về mặt hình dạng cũng như giá trị tham số bên trong (có thể thay đổi được để nắm bắt mới). Một nhiệm vụ chính của mạng nơron là học một mô hình của thế giới thực để đạt được một số mục đích xác định cần quan tâm. Tri thức của thế giới bao gồm hai loại thông tin sau:

- ✓ Trạng thái thế giới đã biết, được biểu diễn bởi các sự kiện về những cái đã biết; dạng tri thức này được xem như là các thông tin ban đầu.
- ✓ Các quan sát (đo đạc) về thế giới, thu nhận được thông qua các cảm biến được thiết kế để thăm dò môi trường mà trong đó mạng hoạt động. Nói chung, các quan sát này luôn bị nhiễu và sai lệch do nhiều nguyên nhân khác nhau. Các quan sát thu nhận được như vậy cung cấp một quỹ thông tin, mà từ đó lấy ra các ví dụ được dùng để huấn luyện mạng nơron.

Do cấu trúc một mạng nơron là vô cùng đa dạng, nên để có thể biểu diễn tri thức một cách có hiệu quả, người ta đưa ra bốn quy tắc chung sau:

Quy tắc 1: Các đầu vào tương tự các lớp tương tự cần phải luôn tạo ra những biểu diễn tương tự trong mạng, và như vậy nên được phân lớp thuộc về cùng loại. Trong tiêu chuẩn này, người ta thường sử dụng một

số thước đo để xác định độ "tương tự" giữa các đầu vào (ví dụ khoảng cách euclide).

Quy tắc 2: Các phân tử mà có thể phân ra thành các lớp riêng biệt thì nên có những biểu diễn khác nhau đáng kể trong mạng.

Quy tắc 3: Nếu một đặc trưng nào đó đặc biệt quan trọng thì nên có một số lượng lớn nơron liên quan đến việc biểu diễn đặc trưng này trong mạng. Số lượng lớn các nơron bảo đảm mức độ chính xác cao trong việc thực hiện các quyết định và nâng cao khả năng chịu đựng các nơron hỏng.

Quy tắc 4: Thông tin ban đầu và các tính chất bất biến nên được đưa vào trong thiết kế ban đầu của mạng neural, và như vậy sẽ giảm bớt gánh nặng cho quá trình huấn luyện. Quy tắc 4 đặc biệt quan trọng vì nếu chúng ta áp dụng nó một cách thích hợp sẽ dẫn đến khả năng tạo ra các mạng nơron với một kiến trúc chuyên biệt. Điều này thực sự được quan tâm do một số nguyên nhân sau:

1. Các mạng nơron thị giác và thính giác sinh học được biết là rất chuyên biệt.
2. Một mạng nơron với cấu trúc chuyên biệt thường có một số lượng nhỏ các tham số tự do phù hợp cho việc chỉnh lý hơn là một mạng kết nối đầy đủ. Như vậy mạng nơron chuyên biệt cần một tập hợp dữ liệu nhỏ hơn cho việc tích lũy; nó học sẽ nhanh hơn, và thường có khả năng tổng quát hóa tốt hơn.
3. Tốc độ chuyển thông tin qua một mạng chuyên biệt là nhanh hơn.
4. Giá của việc xây dựng một mạng chuyên biệt sẽ nhỏ hơn do kích thước nhỏ của nó so với mạng kết nối đầy đủ.

VIII. Một số vấn đề của mạng nơron

Khi xây dựng một ứng dụng mạng nơron chúng ta cần quan tâm một số vấn đề sau:

Vấn đề về kiến trúc mạng nơron: nơron nào nối với nơron nào? Đây chính là sự lựa chọn mô hình của mạng nơron. Nó sẽ phụ thuộc vào sự trình bày dữ liệu và ứng dụng. Những mô hình phức tạp quá dẫn đến những vấn đề lựa chọn quá trình huấn luyện hay là việc lựa chọn giải thuật học.

Lựa chọn giải thuật học: ở đây có nhiều sự cân bằng giữa các thuật học. Gần như bất kỳ giải thuật nào sẽ làm tốt với độ chính xác của các siêu tham số cho việc huấn luyện trên tập dữ liệu cố định trước. Tuy nhiên sự lựa chọn và điều hướng của giải thuật cho việc huấn luyện trên các tập dữ liệu này cần thực hiện nhiều thí nghiệm, đó là điều rất quan trọng. Trên một mô hình nếu lựa chọn giải thuật và hàm đánh giá phù hợp thì mạng nơron có thể cho kết quả rất tốt.

Trọng số của các cung nối và ngưỡng thay đổi thường xuyên. Đã có nhiều nghiên cứu về vấn đề này và cũng đã có một số kết quả:

Nếu mạng gây ra lỗi, thì có thể xác định nơron nào gây ra lỗi => điều chỉnh nơron đó.

Với cách tiếp cận này, mạng phải biết rằng nó gây ra lỗi.

Trong thực tế, lỗi chỉ được biết sau một thời gian dài.

Chức năng của một nơron không quá đơn giản như mô hình. Bởi vì mạng nơron hoạt động như một hộp đen.

Một số hướng dẫn khi sử dụng mạng nơron.

Xây dựng mạng khởi tạo (dùng một lớp ẩn có số nơron =1/2 tổng số nơron của lớp nhập và lớp xuất).

Huấn luyện mạng dùng các giải thuật học. Nên thực hiện trên nhiều mạng khác nhau để tránh trường hợp cực tiểu cục bộ.

Nếu máy "Không thuộc bài" => thêm một vài nơron cho tầng ẩn.

Ngược lại nếu máy "Học vẹt" => bớt một vài nơron ra khỏi tầng ẩn.

Khi đã tìm được một kiến trúc mạng tương đối tốt lấy mẫu lại tập dữ liệu và huấn luyện lại để tìm các mạng mới.

IX. Ứng dụng của mạng nơron

Mạng nơron trong một vài năm trở lại đây đã được nhiều người quan tâm và đã áp dụng thành công trong nhiều lĩnh vực khác nhau, như tài chính, y tế, địa chất, vật lý. Thật vậy, bất cứ ở đâu có vấn đề về dự báo, phân loại và điều khiển, mạng nơron đề có thể ứng dụng được. Ví dụ như khả năng nhận dạng mặt người trong các hệ thống quản lý thông tin liên quan đến con người (quản lý nhân sự ở các công sở, doanh nghiệp; quản lý học sinh, sinh viên trong các trường trung học, đại học, cao đẳng…); các ngành khoa học hình sự, tội phạm; khoa học tướng số, tử vi…

Kết hợp chặt chẽ với logic mở, mạng nơron nhân tạo đã tạo nên cuộc cách mạng thực sự trong việc thông minh hóa và vạn năng hóa các bộ điều khiển kỹ thuật cao cho cả hiện nay và trong tương lai. Ví dụ như ứng dụng tự động điều khiển hệ thống lái tàu, hệ thống dự báo sự cố,…

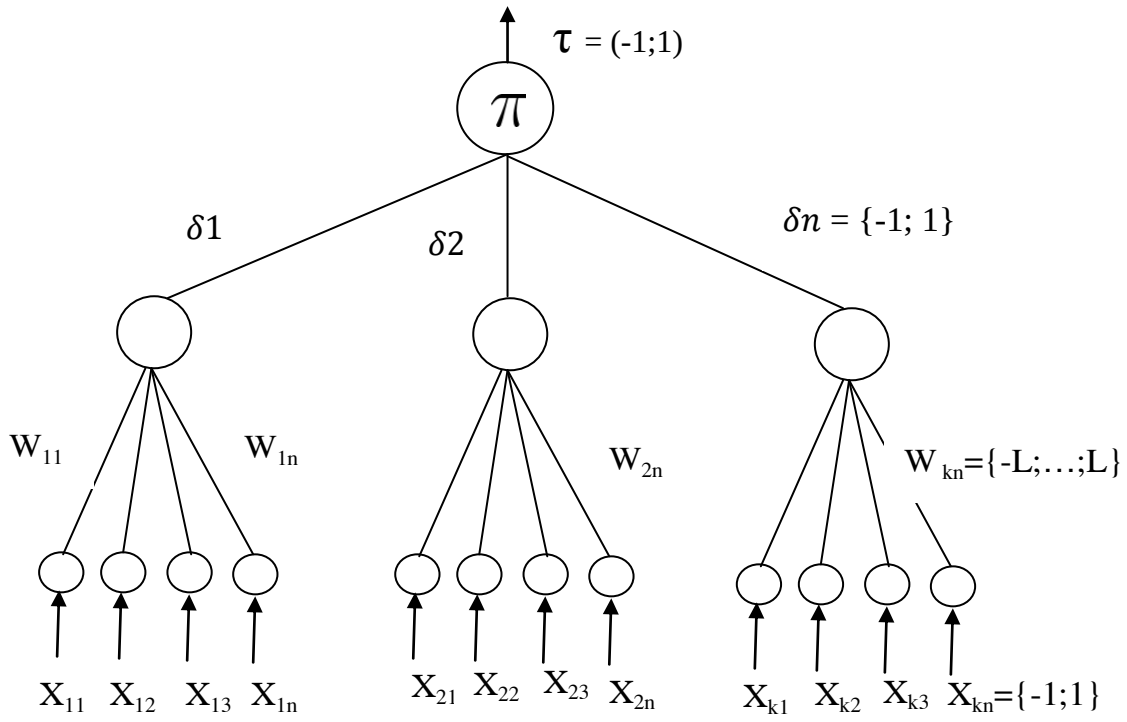
CHƯƠNG IV:

ỨNG DỤNG MẠNG NƠRON VÀO TRAO ĐỔI KHÓA BÍ MẬT

I. Ý tưởng:

Xây dựng một mạng nơron Perceptron mà các phần tử trọng số liên kết được đồng bộ hóa vào trong lớp mạng này, mà trong đó các trọng số liên kết sẽ là các khóa bí mật trong mô hình Tree Parity Machines.

Mô hình TPM (Tree Parity Machines) bao gồm một vector đầu vào X, một lớp ẩn Sigma δ, một trọng số liên kết W giữa các vector đầu vào và lớp ẩn, và một bộ hàm kích hoạt mà đếm các giá trị kết quả là τ. Nó có thể được mô tả bởi ba thông số: K(số lượng tế bào thần kinh ẩn), N (số lượng các tế bào thần kinh kết nối với đầu vào mỗi tế bào thần kinh ẩn) và L (giới hạn giá trị cho trọng số ({-L ... + L}).



Hình 4.1 Mô hình Tree parity machine

Hai máy có cùng một cấu trúc mạng nơron theo mô hình TPM tương tự nhau. Để tính giá trị đầu ra, ta sử dụng một phương pháp đơn giản:

$$\tau = \prod_{i=1}^{K} Sign\left(\sum_{j=1}^{w} W_{ij} X_{ij}\right)$$

Với:

Nơron ẩn: K

Nơron đầu vào : $X_{ij} \in \{-1, 1\}$

Giá trị trọng số liên kết: $W_{ij} \in \{-l, \ldots 0, \ldots +l\}$

$$\text{Sign}(x) = \begin{cases} -1 \text{ nếu } x < 0 \\ 1 \text{ nếu } x \geq 0 \end{cases}$$

Khi nào chúng ta cập nhật giá trị trọng số liên kết? Và bằng cách nào? Chúng ta chỉ cập nhật giá trị trọng số chỉ khi nào giá trị đầu ra bằng nhau. Ở đây có 3 qui tắc huấn luyện khác nhau:

$W^{+}_{i,j} = g(W_{i,j} + X_{i,j}.\tau.\theta(\delta i, \tau).\theta(\tau A, \tau B))$ Hebbian learning rule

$W^{+}_{i,j} = g(W_{i,j} - X_{i,j}.\tau.\theta(\delta i, \tau).\theta(\tau A, \tau B))$ Anti-Hebbian learning rule

$W^{+}_{i,j} = g(W_{i,j} + X_{i,j}.\theta(\delta i, \tau).\theta(\tau A, \tau B))$ Random-walk learning rule

Với:

Theta θ là một chức năng đặc biệt. Theta (a, b) = 0 nếu a ≠ b; ngược lại a=b thì Theta = 1.

g (...) có chức năng giữ trọng số trong phạm vi (-L ; + L)

x là vector đầu vào và w là vector trọng số.

Sau khi hai máy được đồng bộ hóa, thì ma trận trọng số liên kết của chúng là bằng nhau. Ta có thể sử dụng ma trận này để xây dựng một khóa chia sẻ giữa hai máy.

Cách tạo khóa từ ma trận trọng số:

Trên mỗi máy có một chuỗi dãy số ngẫu nhiên A giống nhau.

Kích thước = chiều dài chuỗi A / (L * 2 + 1)

Chiều dài của khóa = (K * N) / Kích thước

Dựa vào ma trận trọng số ta lấy ra vị trí ký tự từ chuỗi A để làm khóa.

```
for (int i = 1; i <= Chiều dài của khóa; i++)
        {
           k = 1;
           for (int j = (i - 1) * Kích thước; j <= i * Kích thước - 1; j++)
              k += w[j] + L;
           key += Chuỗi A [k];
        }
```

Nhận xét:

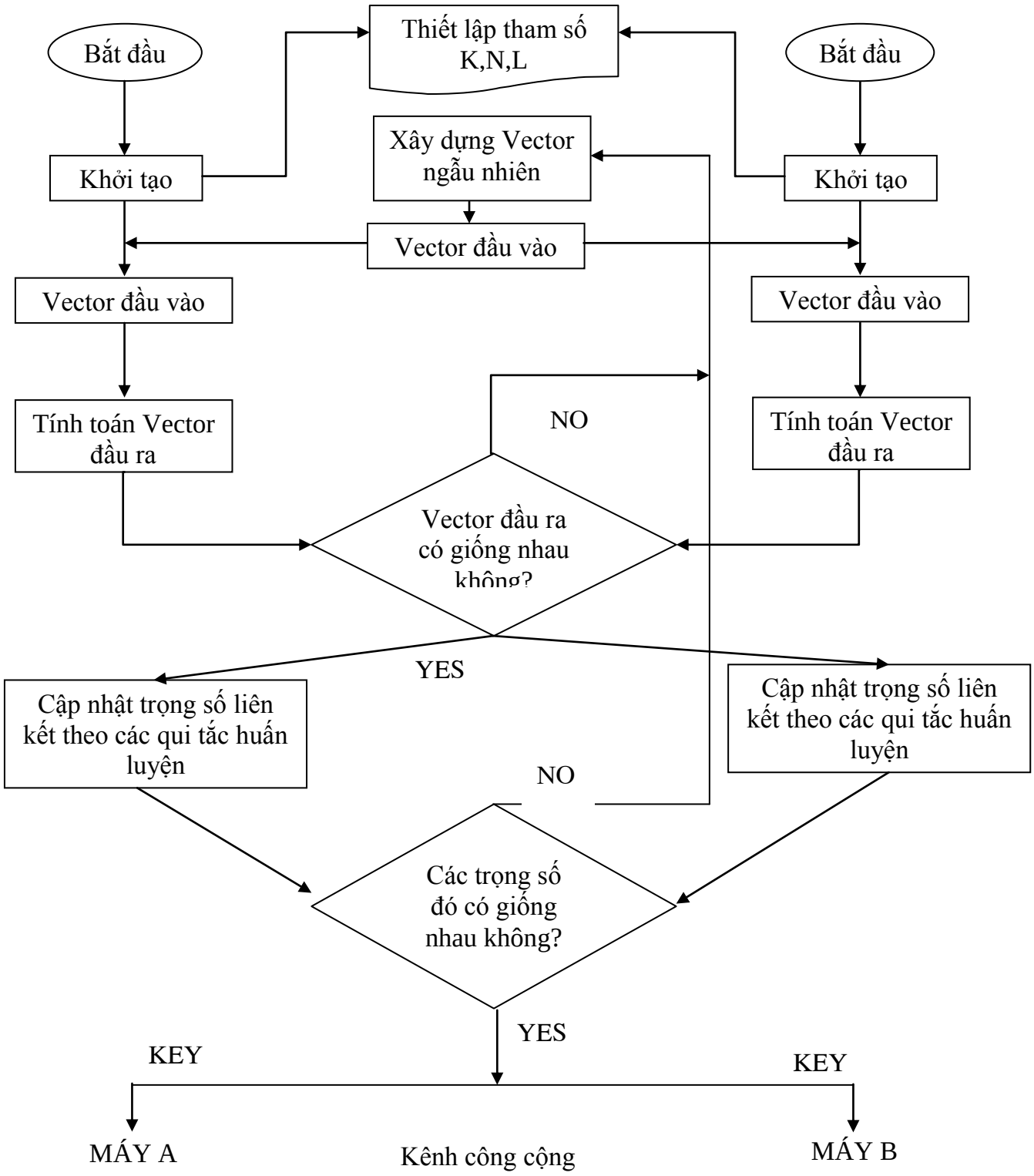
Sau khi các giá trị đầu ra giữa hai máy giống nhau, thì chúng ta mới cập nhật các giá trị trọng số, vì vậy giá trị $\theta(\tau_A , \tau_B)$ trong quy tắc huấn luyện Hebbian là dư. Vì thế tôi đưa ra quy tắc huấn luyện mới dựa trên quy tắc huấn luyện Hebbian để cho phù hợp với bài toán là:

$$W^{+}_{i,j} = g(W_{i,j} + X_{i,j}.\tau.\theta(\delta i , \tau))$$

Trong [11], có rất nhiều thông tin về các cuộc tấn công trên thuật toán này. Trong một kênh công cộng chung thì những kẻ tấn công Eve có thể nghe trộm và bắt được giá trị thông tin giữa bên A và B, nhưng không có cơ hội để thay đổi chúng. Vấn đề ở đây là kẻ tấn công Eve không thể hack được.

Đối với hệ thống mã hóa thông thường, chúng ta có thể cải thiện sự an toàn của các giao thức bằng cách tăng chiều dài của khóa. Trong trường hợp của mã hóa bằng nơron nhân tạo, chúng ta cải thiện bằng cách tăng giá trị L của mạng nơron. Thay đổi tham số này làm tăng chi phí của một cuộc tấn công thành công theo cấp số nhân. Do đó, phá vỡ sự an toàn của trao đổi khóa bằng mạng nơron thuộc độ phức tạp cao.

II. Thuật toán trao đổi khóa bằng mạng nơron Perceptron:



Hình 4.2 Thuật toán trao đổi khóa bằng mạng nơron Perceptron

Ta thực hiện theo 7 bước sau đây để tạo khóa bí mật giữa hai máy dựa trên các mạng nơron Perceptron [7]:

1. Trước hết thiết lập các thông số cho mạng nơron được thực hiện trên cả hai máy A và B:
   K : số lượng các đơn vị lớp ẩn
   N: các đơn vị lớp đầu vào cho từng đơn vị lớp ẩn
   L: phạm vi giới hạn của các giá trị trọng số liên kết trong mạng nơron
2. Các trọng số liên kết được khởi tạo ngẫu nhiên trên mỗi máy.
3. Xây dựng các vectơ đầu vào ngẫu nhiên cho hai máy A và B .
4. Tính toán vectơ đầu ra và trao đổi giữa hai máy A và B.
5. Nếu các vectơ đầu ra của cả hai máy giống nhau có nghĩa là cùng $\tau A = \tau B$ thì tiếp tục bước 6. Nếu các vectơ đầu ra khác nhau thì lặp lại bước 3.
6. Các trọng số liên kết tương ứng trên mỗi máy được cập nhật bằng cách sử dụng quy tắc huấn luyện Hebbian. Nếu các trọng số liên kết giống nhau thì tiếp tục thực hiện bước 7. Nếu chưa đồng bộ thì lặp lại bước 3.
7. Sau khi đồng bộ hoàn chỉnh, trọng số liên kết của mạng nơron giống nhau cho cả hai máy. Và các trọng số này được sử dụng để tạo ra khóa bí mật.

Điều kiện dừng:

Nếu trọng số liên kết của mạng nơron giống nhau cho cả hai máy thì thuật toán dừng.

Nếu sau giá trị số lần lập Max (trong bài toán tôi dùng Max= L^4.N.K) mà trọng số liên kết của mạng nơron không giống nhau thì ta lặp lại bước 2.

CHƯƠNG V

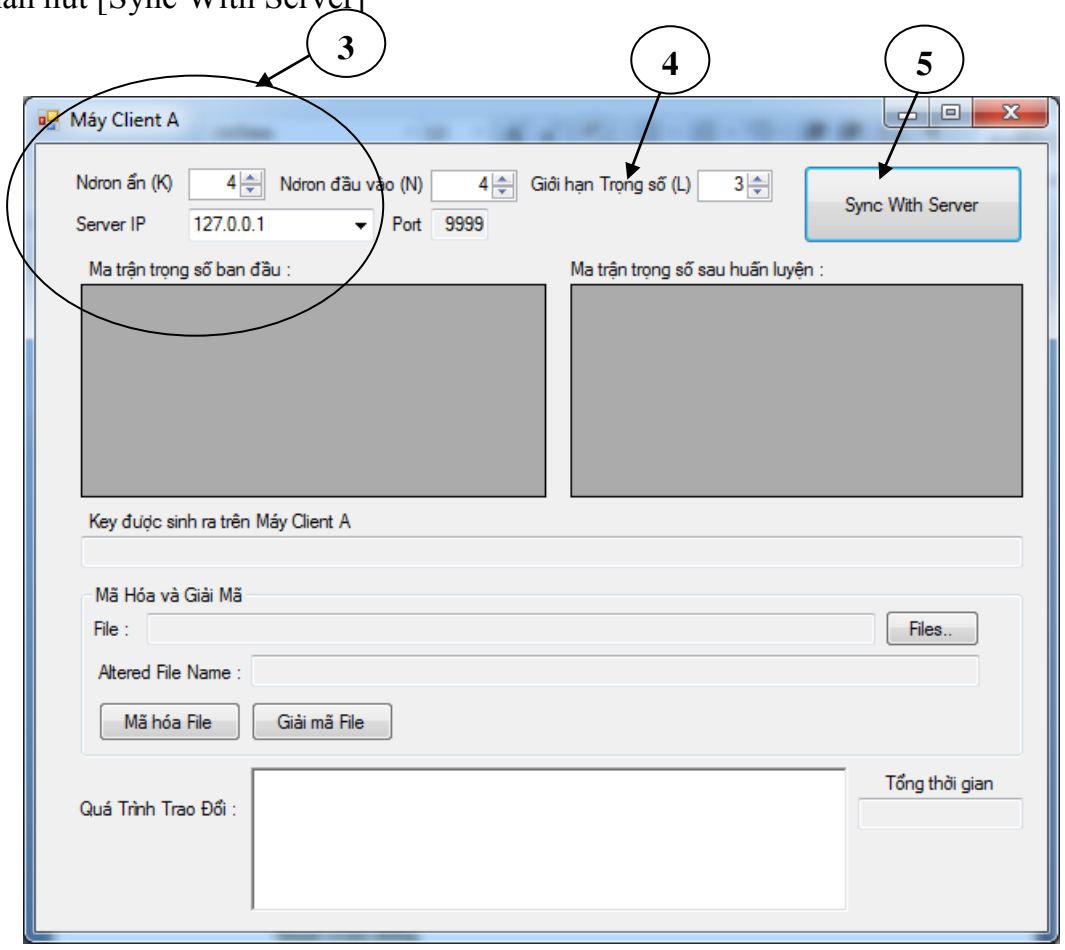
CÀI ĐẶT CHƯƠNG TRÌNH THỬ NGHIỆM

Trong báo cáo này tôi xây dựng mô hình mạng nơron để ứng dụng vào trao đổi khóa bí mật. Sử dụng ngôn ngữ : C# trong bộ Visual studio 2010. Thử nghiệm trên mô hình hệ thống mạng nội bộ tại Trung tâm Viettel Q7

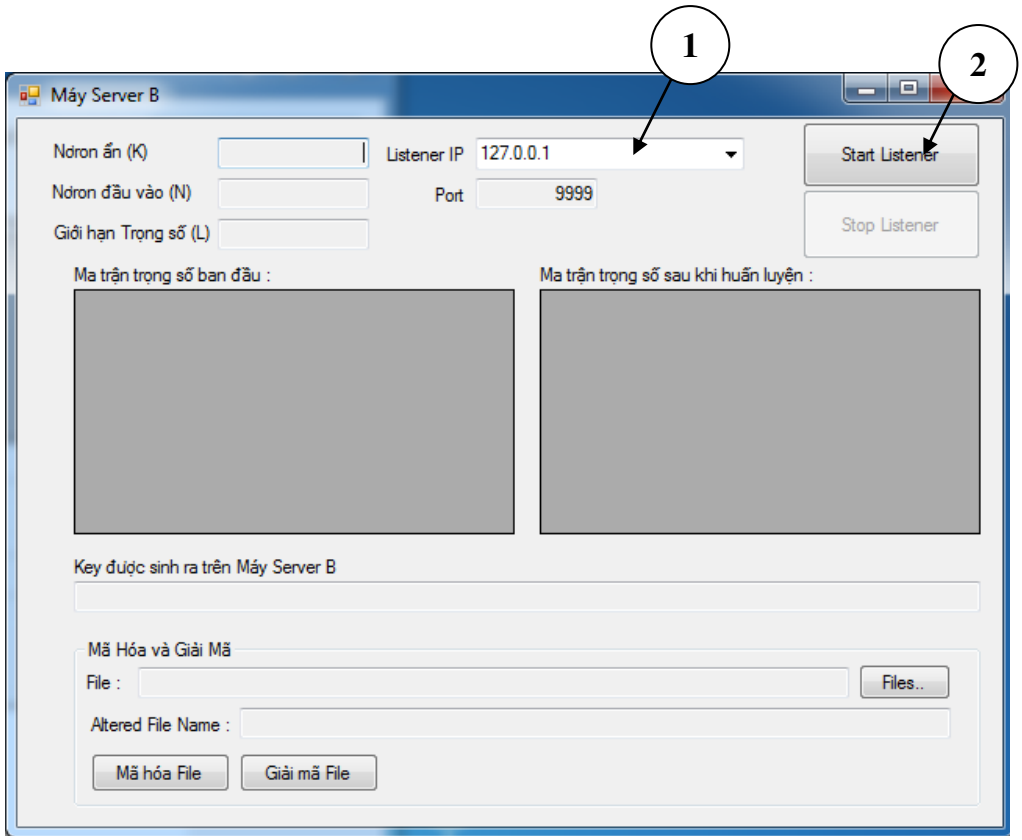
Các bước chạy chương trình:

Trên Server chọn IP và nhấn nút [Start Listener]

Trên Client tạo các giá trị số nguyên: K, L, N. Chọn IP của Server và nhấn nút [Sync With Server]



Hình 5.1 Giao diện chương trình trên máy Client

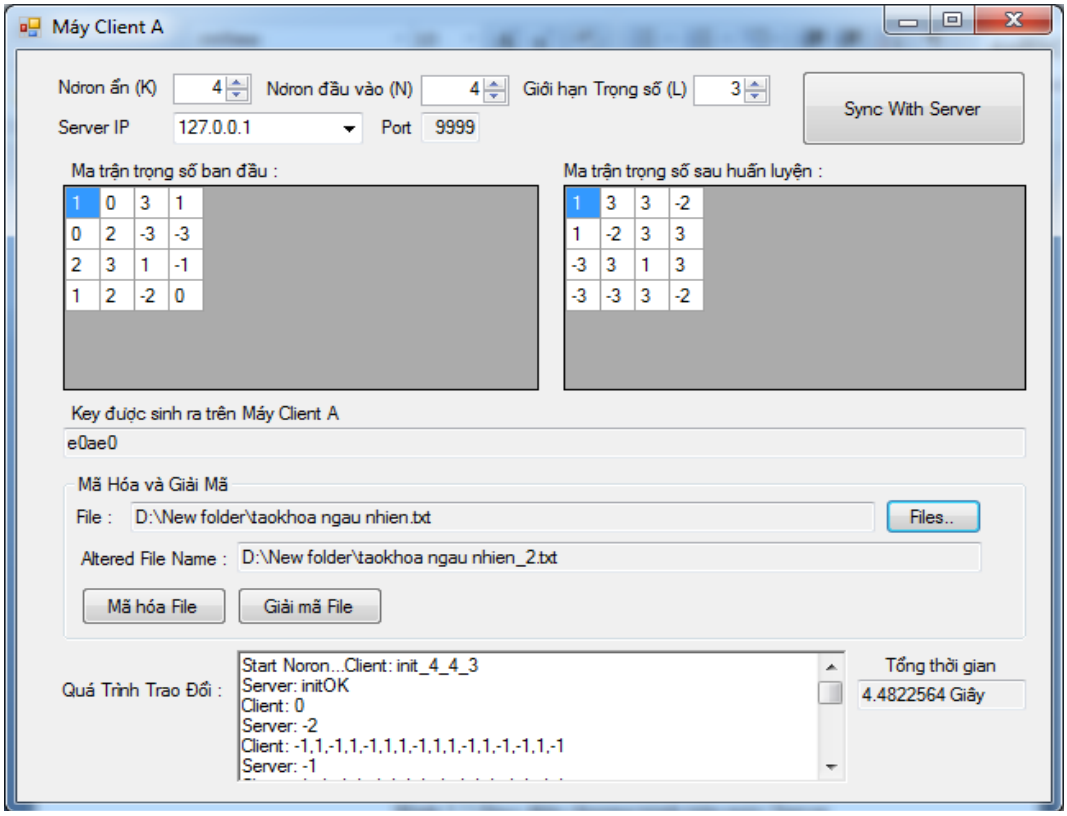


Hình 5.2 Giao diện chương trình trên máy Server

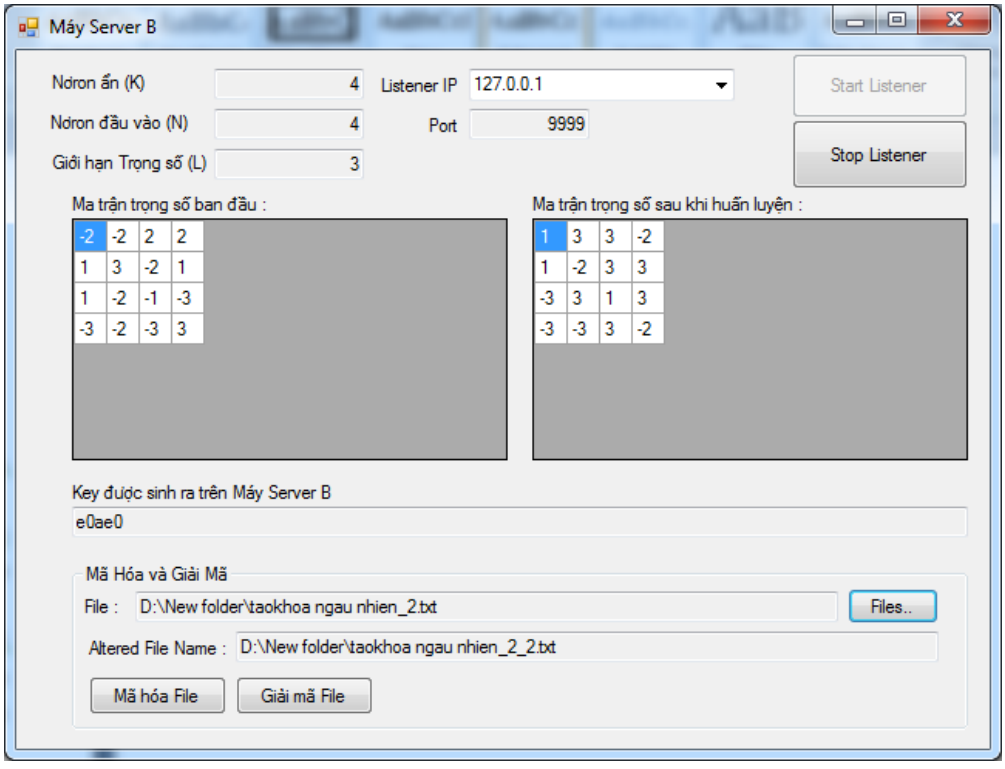
Nếu trọng số liên kết sau huấn luyện của mạng nơron giống nhau cho cả hai máy thì thuật toán dừng.

Nếu sau giá trị số lần lập Max (trong bài toán tôi dùng Max= L^4.N.K) mà trọng số liên kết của mạng nơron không giống nhau thì chương trình tự động tạo ngẫu nhiên lại ma trận trọng số mới trên Client và Server.

Sau khi được đồng bộ thì khóa được tạo ra dựa trên ma trận trọng số sau khi huấn luyện. Với khóa đó tôi dùng thuật toán AES (128bit) Simple Cryptographer - Simple DES/AES Implementation in C# của tác giả Mr.Darcy trên, để mã hóa và giải mã file trên Server và Client. [12]



Hình 5.3 Giao diện chương trình trên máy Client sau huấn luyện



Hình 5.4 Giao diện chương trình trên máy Server sau huấn luyện

KẾT LUẬN

Thuật toán trao đổi khóa bằng mạng nơron Perceptron là một ứng dụng đồng bộ hóa. Cả hai đối tác A và B sử dụng chung một mô hình TPM với cấu trúc tương tự nhau. Các thông số K, L và N là công khai. Mỗi TPM bắt đầu với các vector trọng lựa chọn ngẫu nhiên. Những giá trị trọng số ban đầu được giữ bí mật. Trong suốt quá trình đồng bộ, chỉ có các vector đầu vào và các vector đầu ra được truyền trên các kênh công cộng. Do đó mỗi người tham gia chỉ biết các giá trị nội bộ TPM của riêng mình. Giữ bí mật các thông tin này là điều cần thiết cho sự an toàn của các giao thức trao đổi khóa.

Hướng phát triển:

Trình bày thêm thuật toán trao đổi khóa bằng mô hình mạng nơron nhiều tầng truyền thẳng, mô hình này được cài đặt giống nhau ở cả hai máy A và B, và việc trao đổi giá trị các vectơ đầu vào chỉ xảy ra một lần nên việc tính toán giá trị đầu ra diễn ra rất nhanh chóng. Việc tạo ra các lớp và số lượng nơron ẩn là bảo mật trong chương trình nên kẻ tấn công Eve khó phát hiện ra được mô hình lớp mạng này. Ta có thể thay đổi chương trình đa dạng hơn bằng cách cập nhật thêm lớp ẩn, tăng hoặc giảm số lượng nơron ẩn, thay đổi giá trị của ma trận trọng số. Điều này làm tăng độ phức tạp cho thuật toán và làm giảm khả năng hack được khóa của kẻ tấn công Eve.

Chương trình trong đề tài này sử dụng AES(128bit), cần mở rộng thêm nhiều phương pháp mã hóa khác nhau để tăng độ bảo mật cao hơn.

TÀI LIỆU THAM KHẢO

[1] Dr. Ajit Singh, Aarti nandal CSE, SES, BPSMV India, (2013). " Neural Cryptography for Secret Key Exchange and Encryption with AES", ISSN: 2277 128X, Volume 3, Issue 5, pp. 376-381

[2] Vidushi Sharma, Sachin Rai, Anurag Dev, (2012). " A Comprehensive Study of Artificial Neural Networks", International Journal of Advanced Research in Computer Science and Software Engineering 2 (10), pp.278-284

[3] M.Jogdand1 and Sahana S.Bisalapur2, (2011). " Design of an Efficient Nơron Key Distribution Centre", International Journal of Artificial Intelligence & Applications (IJAIA), Volume 2, No.1, pp. 60–69

[4] Whitfield Diffie and Martin E.Hellman, (1976). " New Directions in Cryptography", the IEEE International Symposium on Information Theory in Ronneby, Sweden, June 21–24

[5] Jean-Francois Raymond and Anton Stiglic, "Security Issues in the Diffie-Hellman Key Agreement Protocol"

[6] Vidushi Sharma, Sachin Rai , Anurag Dev, (2012). "A Comprehensive Study of Artificial Neural Networks", ISSN: 2277 128X, Volume 2, Issue 10

[7] MISS. SAHANA S.BISALAPUR, "Design of an Efficient Neural Key Distribution Centre"

[8] Maryam Ahmed, Baharan Sanjabi, Difo Aldiaz, Amirhossein Rezaei, Habeeb Omotunde, (2012). "Diffie-Hellman and Its Application in Security Protocols", ISSN: 2319-5967, Volume 1, Issue 2, pp. 69 -73

[9] Ths Lê Thụy, "An Toàn Bảo Mật Thông Tin 1", Trường ĐH DL Hải Phòng

[10] Michal Rosen-Zvi, Einat Klein, Ido Kanter, and Wolfgang Kinzel, (2002). "Mutual learning in a tree parity machine and its application to cryptography", ISSN:1063-651X, 4 August 2002; published 30

[11] Neural_cryptography, en.wikipedia.org/wiki/, truy cập vào ngày 18/11/2014

[12] Mr.Darcy, (2007). http://www.codeproject.com/, truy cập vào ngày 10/12/2014