TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI

CHƯƠNG TRÌNH ĐÀO TẠO THẠC SĨ KHOA HỌC

NGÀNH CÔNG NGHỆ THÔNG TIN

TIỂU LUẬN MÔN HỌC

TRÍ TUỆ NHÂN TẠO NÂNG CAO

ỨNG DỤNG MẠNG NƠRON NHÂN TẠO TRONG NHẬN DẠNG CÁC KÝ SỐ TIẾNG VIỆT

Hướng dẫn khoa học: **TS. Nguyễn Đình Thuân**

Học viên: Mai Cường Thọ

Trần Công Cẩn

Huỳnh Quang Đệ

Tháng 03/2010

MỤC LỤC

1- Mở đầu

Ngay khi phát minh ra máy tính, con người đã mơ ước máy tính có thể nói chuyện với mình. Yêu cầu đơn giản nhất là máy có thể xác định được từ ngữ mà chúng ta nói với máy. Đó là mục tiêu của ngành nhận dạng tiếng nói.

Đối với con người, việc nghe, nhất là nghe tiếng mẹ đẻ là một vấn đề khá đơn giản. Còn đối với máy tính, xác định một chuỗi tín hiệu âm thanh là sự phát âm của một từ nào hoàn toàn không đơn giản, khó khăn cũng tương tự như việc học nghe ngoại ngữ của chúng ta.

Lĩnh vực nhận dạng tiếng nói đã được nghiên cứu hơn 4 thập kỷ và hiện nay mới chỉ có một số thành công. Có thể kể đến hệ thống nhận dạng tiếng Anh (ví dụ: phần mềm Via Voice của IBM, hệ thống nhận dạng tiếng nói tích hợp của OfficeXP…). Các hệ thống này hoạt động khá tốt (cho độ chính xác khoảng 90 - 95%) nhưng còn khá xa mới đạt đến mức mơ ước của chúng ta là có một hệ thống có thể nghe chính xác và hiểu hoàn toàn những điều chúng ta nói.

Riêng với tiếng Việt, lĩnh vực nhận dạng tiếng nói còn khá mới mẻ. Chưa hề thấy xuất hiện một phần mềm nhận dạng tiếng Việt hoàn chỉnh trên thị trường. Số công trình nghiên cứu về nhận dạng tiếng nói tiếng Việt được công bố không nhiều, và kết quả còn hạn chế về bộ từ vựng, độ chính xác…. Tiếng Việt có nhiều đặc tính khác với các ngôn ngữ đã được nghiên cứu nhận dạng nhiều như tiếng Anh, tiếng Pháp. Do đó việc nghiên cứu nhận dạng tiếng Việt là rất cần thiết.

Vì những lý do trên, sau khi học xong môn "Trí tuệ nhân tạo tiên tiến", ở mức độ của một tiểu luận môn học, chúng em chọn đề tài ***"Ứng dụng mạng Nơron trong nhận dạng tiếng Việt"*** nhằm nghiên cứu các phương pháp nhận dạng tiếng nói đối với tiếng Việt và thử nghiệm xây dựng một hệ thống nhận dạng cỡ nhỏ để ***nhận dạng việc đọc các số từ "không" đến "chín".***

Mặc dù chúng em đã hết sức cố gắng trong việc nghiên cứu và ứng dụng nhưng do thời gian thực hiện ngắn nên kết quả nghiên cứu còn khá khiêm tốn. Tuy nhiên, đề tài tiểu luận này sẽ làm nền tảng cho chúng em tiếp tục nghiên cứu những đề tài chuyên sâu hơn về nhận dạng tiếng nói nói riêng và ứng ụng trí tuệ nhân tạo vào thực tiễn nói chung.

Chúng em cũng bầy tỏ lòng biết ơn đến Tiến sĩ Nguyễn Đình Thuân đã truyền đạt cho nhóm, cũng như lớp học nhiều kiến thức bổ ích về mạng nơ ron và ứng dụng của nó trong nhận dạng, tạo tiền đề cho chúng em thực hiện tốt bài tập này.

2. Nhận dạng tiếng nói và một số phương pháp nhận dạng phổ biến

2.1. Nhận dạng tiếng nói

Hiểu một cách đơn giản, nhận dạng tiếng nói (speech recognition by machine) là dùng máy tính chuyển đổi tín hiệu ngôn ngữ từ dạng âm thanh thành dạng văn bản. Nói một cách chính xác hơn: ***nhận dạng tiếng nói là phân chia (segmentation) và đính nhãn ngôn ngữ (labeling) cho tín hiệu tiếng nói***.

Nhận dạng tiếng nói có nhiều ứng dụng:

- **Đọc chính tả**. Là ứng dụng được sử dụng nhiều nhất trong các hệ nhận dạng. Thay vì nhập liệu bằng tay thông qua bàn phím, người sử dụng nói với máy qua micro và máy xác định các từ được nói trong đó;

- **Điều khiển - giao tiếp không dây**. Chẳng hạn hệ thống cho phép máy tính nhận lệnh điều khiển bằng giọng nói của con người như: *"chạy chương trình"*, *"tắt máy"*… Một số ưu điểm của việc sử dụng tiếng nói thay cho các thiết bị vào chuẩn như bàn phím, con chuột là: thuận tiện, tốc độ cao, không bị ảnh hưởng của cáp, khoảng cách, không đòi hỏi huấn luyện sử dụng...

- **Điện thoại-liên lạc**. Một số hệ thống (chẳng hạn ở máy điện thoại di động) cho phép người sử dụng đọc tên người trong danh sách thay vì bấm số. Một số hệ thống khác (ở ngân hàng, trung tâm chứng khoán…) thực hiện việc trả lời tự động đối với các các cuộc gọi hỏi về tài khoản…

Tuy nhiên vấn đề nhận dạng tiếng nói gặp rất nhiều khó khăn. Một số khó khăn chủ yếu là:

- Tiếng nói là tín hiệu thay đổi theo thời gian. Mỗi người có một giọng nói, cách phát âm khác nhau... Thậm chí một người phát âm cùng một từ mà mỗi lần khác nhau cũng không giống nhau (chẳng hạn về tốc độ, âm lượng...);
- Các phương pháp nhận dạng hiện tại của máy tính khá "máy móc", còn xa mới đạt đến mức độ tư duy của con người;
- Nhiễu là thành phần luôn gặp trong môi trường hoạt động của các hệ thống nhận dạng và ảnh hưởng rất nhiều đến kết quả nhận dạng.

Do những khó khăn đó, nhận dạng tiếng nói cần tri thức rất nhiều từ các ngành khoa học liên quan như:

- **Xử lý tín hiệu**: tìm hiểu các phương pháp tách các thông tin đặc trưng, ổn định từ tín hiệu tiếng nói, giảm ảnh hưởng của nhiễu và sự thay đổi theo thời gian của tiếng nói;

- **Âm học**: tìm hiểu mối quan hệ giữa tín hiệu tiếng nói vật lý với các cơ chế sinh lý học của việc phát âm và việc nghe của con người;

- **Nhận dạng mẫu**: nghiên cứu các thuật toán để phân lớp, huấn luyện và so sánh các mẫu dữ liệu...;

- **Lý thuyết thông tin**: nghiên cứu các mô hình thống kê, xác suất; các thuật toán tìm kiếm, mã hoá, giải mã, ước lượng các tham số của mô hình…

- **Ngôn ngữ học**: tìm hiểu mối quan hệ giữa ngữ âm và ngữ nghĩa, ngữ pháp, ngữ cảnh của tiếng nói;

- **Tâm-sinh lý học**: tìm hiểu các cơ chế bậc cao của hệ thống nơron của bộ não người trong các hoạt động nghe và nói;

- **Khoa học máy tính**: nghiên cứu các thuật toán, các phương pháp cài đặt và sử dụng hiệu quả các hệ thống nhận dạng trong thực tế.

2.2. Một số phương pháp nhận dạng tiếng nói phổ biến

2.2.1. Phương pháp ngữ âm - âm vị học (acoustic-phonetic approach)

Phương pháp ngữ âm - âm vị học dựa trên lý thuyết âm vị: lý thuyết này khẳng định sự tồn tại hữu hạn và duy nhất các đơn vị ngữ âm cơ bản trong ngôn ngữ nói gọi là âm vị, được phân chia thành: nguyên âm - phụ âm, vô thanh-hữu thanh, âm vang -âm bẹt… Các âm vị có thể xác định bởi tập các đặc trưng trong phổ của tín hiệu tiếng nói theo thời gian.

Đặc trưng quan trọng nhất của âm vị là formant. Đó là các vùng tần số có cộng hưởng cao nhất của tín hiệu. Ngoài ra còn một số đặc trưng khác như âm vực (tần số cơ bản- pitch), âm lượng…

Hệ thống nhận dạng dựa trên phương pháp này sẽ tách các đặc trưng từ tín hiệu tiếng nói và xác định chúng tương ứng với âm vị nào. Sau đó, dựa vào một từ điển phiên âm, máy sẽ xác định chuỗi các âm vị đó có khả năng là phát âm của từ nào nhất.

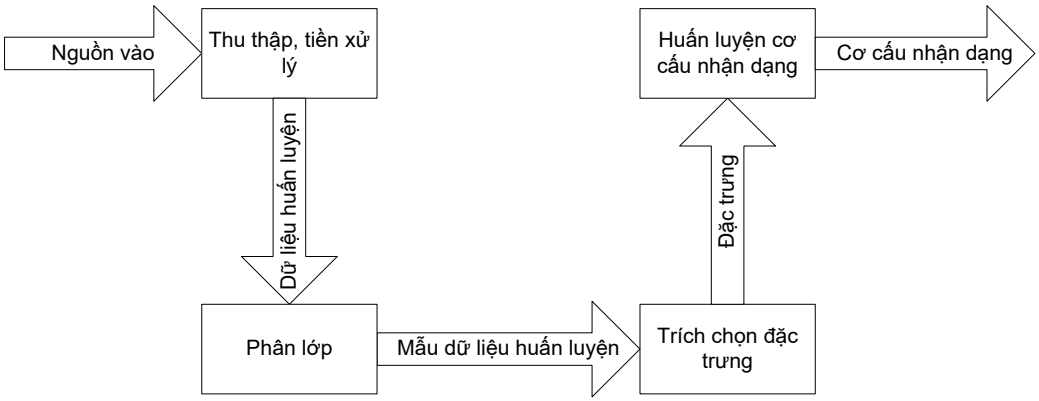
Xét khía cạnh nguyên lý, phương pháp có vẻ rất đơn giản. Tuy nhiên các thử nghiệm trong thực tế cho thấy phương pháp cho kết quả nhận dạng không cao. Nguyên nhân từ những vấn đề sau:

- Phương pháp cần rất nhiều tri thức về ngữ âm học, nhất là các tri thức liên quan đến đặc tính âm học của các âm vị. Mà những tri thức này nhìn chung còn chưa được nghiên cứu đầy đủ;
- Formant chỉ ổn định đối với các nguyên âm, với phụ âm formant rất khó xác định và không ổn định. Hơn nữa việc xác định các formant cho độ chính xác không cao. Đặc biệt khi chịu ảnh hưởng của nhiễu (là vấn đề thường xảy trong thực tế);
- Rất khó phân biệt các âm vị dựa trên phổ, nhất là các phụ âm vô thanh. Có một số phụ âm rất giống nhiễu (ví dụ: /s/, /h/).

2.2.2. Phương pháp nhận dạng mẫu (pattern recognition approach)

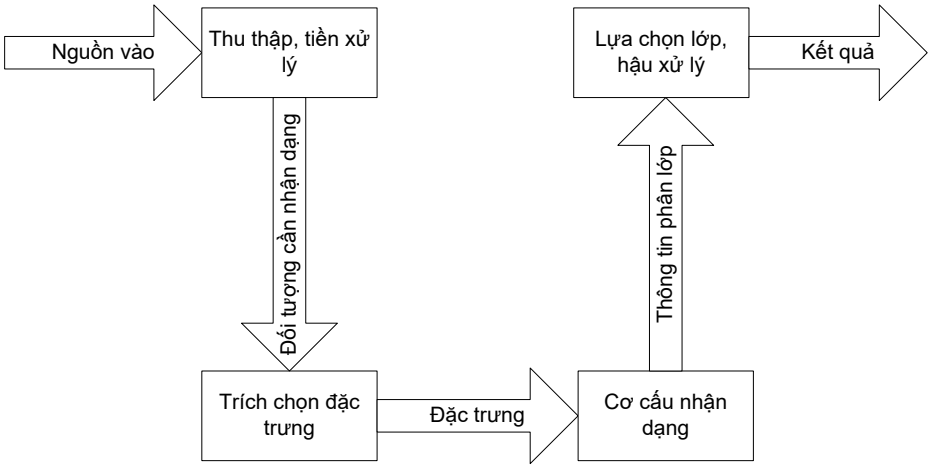
Phương pháp nhận dạng mẫu dựa vào lý thuyết xác suất - thống kê để nhận dạng dựa trên ý tưởng: *so sánh đối tượng cần nhận dạng với các mẫu được thu thập trước đó để tìm mẫu "giống" đối tượng nhất*. Như vậy hệ thống nhận dạng sẽ trải qua 2 giai đoạn:

a) **Giai đoạn huấn luyện**: thực hiện các nhiệm vụ: thu thập mẫu, phân lớp và huấn luyện hệ thống ghi nhớ các mẫu đó:



Hình 2.1: Minh hoạ sơ đồ giai đoạn huấn luyện của phương pháp đối sánh mẫu

b) **Giai đoạn nhận dạng**: nhận vào đối tượng cần nhận dạng, so sánh với các mẫu và đưa ra kết quả là mẫu giống đối tượng nhất:



Hình 2.2: Sơ đồ giai đoạn nhận dạng của phương pháp đối sánh mẫu

Phần lớn các hệ nhận dạng thành công trên thế giới là sử dụng phương pháp này. Phương pháp có những ưu điểm sau:

- Sử dụng đơn giản, dễ hiểu, mang tính toán học cao (lý thuyết xác suất thống kê, lý thuyết máy học, …);
- Ít bị ảnh hưởng của những biến thể về bộ từ vựng, tập đặc trưng, đơn vị nhận dạng, môi trường xung quanh…;
- Cho kết quả cao: Điều này đã được kiểm chứng trong thực tế.

2.2.3. Phương pháp trí tuệ nhân tạo (artifactial intelligence approach)

Phương pháp trí tuệ nhân tạo nghiên cứu cách học nói và học nghe của con người, tìm hiểu các quy luật ngữ âm, ngữ pháp, ngữ nghĩa, ngữ cảnh… và tích hợp chúng bổ sung cho các phương pháp khác để nâng cao kết quả nhận dạng như:

- Có thể thêm các hệ chuyên gia (expert system), các luật logic mờ (fuzzy logic) về ngữ âm, âm vị… vào các hệ nhận dạng tiếng nói dựa trên phương pháp ngữ âm-âm vị học để tăng độ chính xác cho việc xác định các âm vị (vấn đề đã được đề cập là rất khó nếu chỉ sử dụng các thông tin về âm phổ);

- Hay đối với các hệ nhận dạng mẫu, người ta cải tiến bằng cách với mỗi đối tượng cần nhận dạng, hệ thống sẽ chọn ra một số mẫu "giống" đối tượng nhất, sau đó sẽ kiểm chứng tiếp các kết quả đó bằng các luật ngữ pháp, ngữ nghĩa, ngữ cảnh… để xác định mẫu phù hợp nhất.

Hiện nay đang có một phương pháp trí tuệ nhân tạo trong nhận dạng tiếng nói được nghiên cứu rộng rãi là mạng nơron. Tuỳ vào cách sử dụng, mạng nơron có thể coi là mở rộng của phương pháp nhận dạng mẫu hoặc phương pháp ngữ âm-

âm vị học. Do có những đặc tính nổi trội, mạng nơron được hi vọng sẽ tăng cường hiệu quả của các hệ nhận dạng tiếng nói.

Vì vậy, mạng nơron là phương pháp được chúng em nghiên cứu xây dựng hệ nhận dạng trình bày trong tiểu luận này.

3. Nhận dạng tiếng tiếng Việt

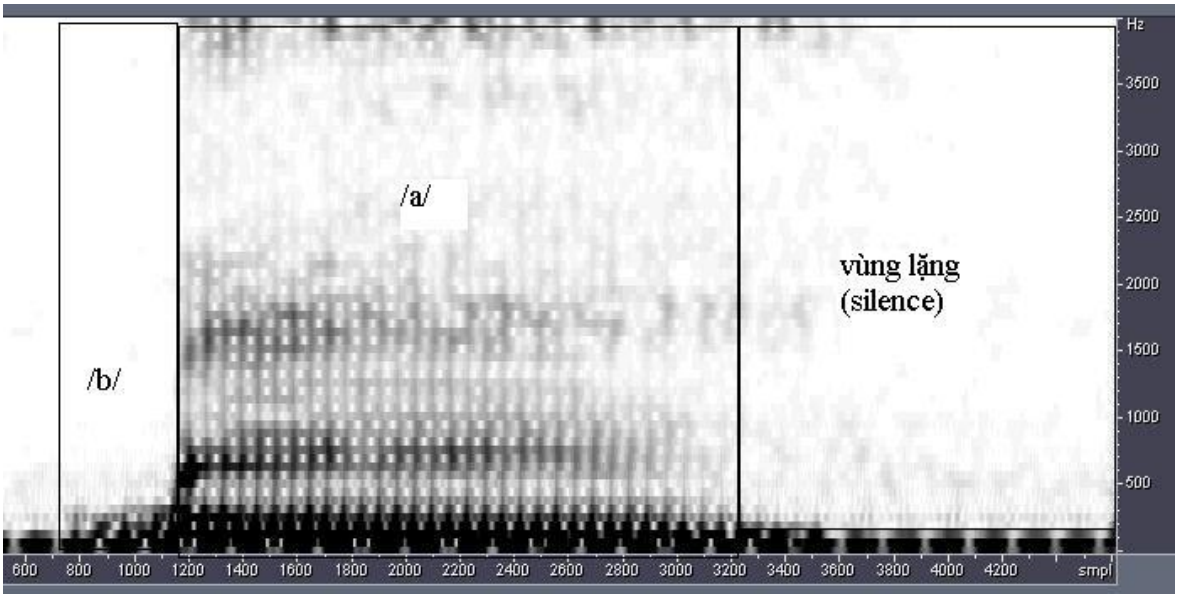
3.1. Một số đặc điểm ngữ âm tiếng Việt

Một đặc điểm dễ thấy là tiếng Việt là ngôn ngữ đơn âm (monosyllable - mỗi từ đơn chỉ có một âm tiết), không biến hình (cách đọc, cách ghi âm không thay đổi trong bất cứ tình huống ngữ pháp nào). Tiếng Việt hoàn toàn khác với các ngôn ngữ Ấn-Âu như tiếng Anh, tiếng Pháp là các ngôn ngữ đa âm, biến hình.

Theo thống kê trong tiếng Việt có khoảng 6000 âm tiết. Nhìn về mặt ghi âm: âm tiết tiếng Việt có cấu tạo chung là: phụ âm-vần. Ví dụ âm *tin* có phụ âm *t*, vần *in*. Phụ âm là một âm vị và âm vị này liên kết rất lỏng lẻo với phần còn lại của âm tiết (hiện tượng nói lái).

Vần trong tiếng Việt lại được cấu tạo từ các âm vị nhỏ hơn, trong đó có một âm vị chính là nguyên âm.

Hình sau là phổ tín hiệu của âm tiết "ba". Chúng ta có thể quan sát và phân biệt rõ miền nhiễu nền, miền phổ của phụ âm b và nguyên âm a (miền đậm hơn là có mật độ năng lượng lớn hơn).

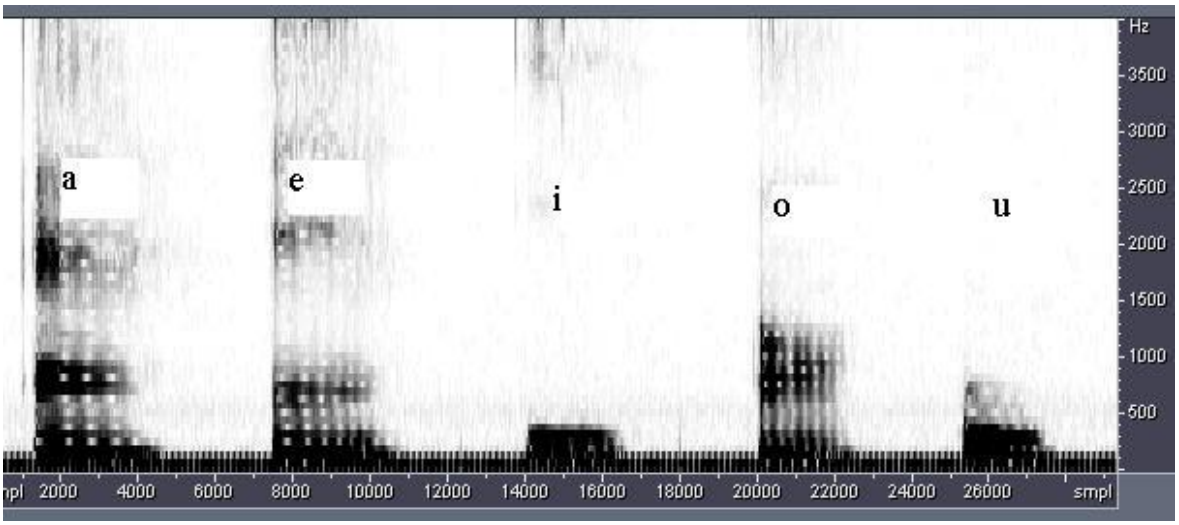


Hình 3.1: Minh hoạ phổ tín hiệu của âm tiết "ba", có miền nhiễu nền (silence), miền tín hiệu của phụ âm /b/ và nguyên âm /a/ (miền

đậm hơn là có mật độ năng lượng lớn hơn).

Quan sát phổ các âm tiết tương tự chúng ta có thể rút ra kết luận: các phụ âm và nguyên âm đều phân biệt với nhau rất rõ qua sự phân bố năng lượng tại các miền tần số, ví dụ: phụ âm ở tần số thấp, năng lượng nhỏ, nguyên âm có năng lượng lớn ở cả vùng tần số cao. Vùng không có tín hiệu tiếng nói (nhiễu nền và khoảng lặng) có năng lượng thấp và chỉ tập trung ở các tần số rất thấp.

Các nguyên âm có tần phổ (spectrum) khác nhau khá rõ. Hình sau minh hoạ sự khác nhau về phổ của 5 nguyên âm cơ bản. Miền đậm là miền có mật độ năng lượng cao.



Hình 3.2: Minh hoạ sự khác nhau về phổ của 5 nguyên âm cơ bản. Miền đậm là miền có mật độ năng lượng cao (vùng có formant).

Xét về mặt ngữ âm-âm vị học thì âm tiết tiếng Việt có lược đồ như sau:

Thanh điệu			
Âm đầu	***Vần***		
	Âm đệm	Âm chính	Âm cuối

Lược đồ cho thấy âm tiết tiếng Việt có cấu trúc rõ ràng, ổn định. Lược đồ còn cho thấy tiếng Việt là ngôn ngữ có thanh điệu. Hệ thống thanh điệu gồm 6 thanh: bằng, huyền, sắc, hỏi, ngã, nặng.

Thanh điệu trong âm tiết là âm vị siêu đoạn tính (thể hiển trên toàn bộ âm tiết). Do đó đặc trưng về thanh điệu thể hiện trong tín hiệu tiếng nói không rõ nét như các thành phần khác của âm tiết.

Sự khác biệt về cách phát âm tiếng Việt rất rõ rệt theo giới, lứa tuổi và đặc biệt là theo vị trí địa lý (giọng miền Bắc, miền Trung và miền Nam khác nhau rất nhiều).

3.2. Những thuận lợi và khó khăn đối với nhận dạng tiếng nói tiếng Việt

3.2.1- Thuận lợi

Những đặc điểm ngữ âm tiếng Việt cho thấy nhận dạng tiếng nói tiếng Việt có một số thuận lợi sau:

- Tiếng Việt là ngôn ngữ đơn âm, số lượng âm tiết không quá lớn. Điều này sẽ giúp hệ nhận dạng xác định ranh giới các âm tiết dễ dàng hơn nhiều. Đối với hệ nhận dạng các ngôn ngữ Ấn-Âu (tiếng Anh, tiếng Pháp...) xác định ranh giới âm tiết (endpoint detection) là vấn đề rất khó và ảnh hưởng lớn đến kết quả nhận dạng;

- Tiếng Việt là ngôn ngữ không biến hình từ. Âm tiết tiếng Việt ổn định, có cấu trúc rõ ràng. Đặc biệt không có 2 âm tiết nào đọc giống nhau mà viết khác nhau. Điều này sẽ dễ dàng cho việc xây dựng các mô hình âm tiết trong nhận dạng; đồng thời việc chuyển từ phiên âm sang từ vựng (lexical decoding) sẽ đơn giản hơn so với các ngôn ngữ Ấn-Âu. Việc chuyển từ phiên âm sang từ vựng cũng là một vấn đề khó khăn trong nhận dạng các ngôn ngữ Ấn-Âu.

3.2.2- Khó khăn

Ngoài những thuận lợi trên, nhận dạng tiếng nói tiếng Việt cũng gặp rất nhiều khó khăn như sau:

- Tiếng Việt là ngôn ngữ có thanh điệu (6 thanh). Thanh điệu là âm vị siêu đoạn tính, đặc trưng về thanh điệu thể hiện trong tín hiệu tiếng nói không rõ nét như các thành phần khác của âm tiết;

- Cách phát âm tiếng Việt thay đổi nhiều theo vị trí địa lý. Giọng địa phương trong tiếng Việt rất đa dạng (mỗi miền có một giọng đặc trưng);

- Hệ thống ngữ pháp, ngữ nghĩa tiếng Việt rất phức tạp, rất khó để áp dụng vào hệ nhận dạng với mục đích tăng hiệu năng nhận dạng. Hệ thống phiên âm cũng chưa thống nhất;

- Các nghiên cứu về nhận dạng tiếng Việt cũng chưa nhiều và ít phổ biến. Đặc biệt khó khăn lớn nhất là hiện nay chưa có một bộ dữ liệu chuẩn cho việc huấn luyện và kiểm tra các hệ thống nhận dạng tiếng Việt.

4. Trích chọn đặc trưng tín hiệu tiếng nói bằng phương pháp MFCC

Quá trình nhận dạng mẫu (cả ở pha huấn luyện hay pha nhận dạng) đều trải qua bước trích chọn đặc trưng (feature extraction). Bước này thực hiện các phân tích tín hiệu tiếng nói nhằm xác định các thông tin quan trọng, đặc trưng, ổn định của tín hiệu tiếng nói, tối thiểu hoá ảnh hưởng của nhiễu; xúc cảm, trạng thái, cách phát âm của người nói; giảm khối lượng dữ liệu cần xử lý...

Mặc dù không mang tính quyết định nhưng giai đoạn trích chọn đặc trưng ảnh hưởng rất lớn đến hiệu năng nhận dạng. Vì vậy việc lựa chọn đặc trưng cho tín hiệu tiếng nói rất quan trọng.

Có nhiều phương pháp trích chọn đặc trưng đã và đang được sử dụng: FBA, MFCC, LPC, PLP.... Mỗi phương pháp có những ưu điểm và nhược điểm riêng. Hiện nay MFCC (*Mel-scale Frequency Cepstral Coefficient, Mel là viết tắ của Melody- âm điệu*) được sử dụng phổ biến nhất. Vì vậy chúng em sử dụng MFCC làm đặc trưng của hệ nhận dạng được trình bày trong tiểu luận này.

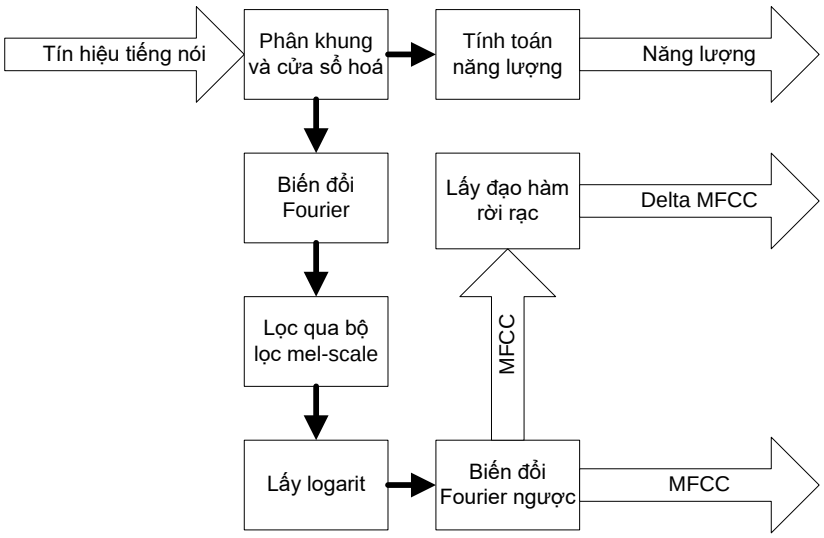
Các hệ nhận dạng tiếng nói thường tách đặc trưng từ tín hiệu bằng cách: chia tín hiệu thành các đoạn độ dài 10-30 ms, mỗi đoạn gọi là một khung (frame). Trong khoảng thời gian ngắn như vậy, phổ của tín hiệu đủ ổn định để tiến hành tách đặc trưng. Mỗi frame sẽ cho đặc trưng là một vector và đặc trưng của toàn bộ tín hiệu sẽ là một dãy vector.

MFCC là phương pháp trích đặc trưng dựa trên đặc điểm cảm thụ tần số âm của tai người: *tuyến tính đối với tần số nhỏ hơn 1kHz và phi tuyến đối với tần số trên 1kHz (theo thang tần số mel - Melody, không phải theo Hz)*.

MFCC là phương pháp tách đặc trưng dựa trên sự cảm thụ của con người nên thường cho kết quả nhận dạng cao nhất. Vì lẽ đó, rất nhiều hệ thống nhận dạng tiếng nói sử dụng MFCC làm đặc trưng.

4.1. Sơ đồ khối của quá trình tính MFCC

Việc tính đặc trưng theo phương pháp MFCC được trình bày ở sơ đồ sau:



Hình 4.1: ***Minh hoạ sơ đồ khối của quá trình trích chọn đặc trưng MFCC***

Quá trình tính toán như sau: đầu tiên tín hiệu tiếng nói được chia thành các frame có độ dài 10-30ms. Mỗi frame sẽ được nhân với một hàm cửa sổ, thường là cửa sổ Hamming sau đó được chuyển sang miền tần số nhờ biến đổi Fourier. Tín hiệu ở miền tần số được lọc qua các bộ lọc mel-scale, lấy logarit rồi biến đổi Fourier ngược (để chuyển sang miền cepstral) sẽ được các hệ số MFCC.

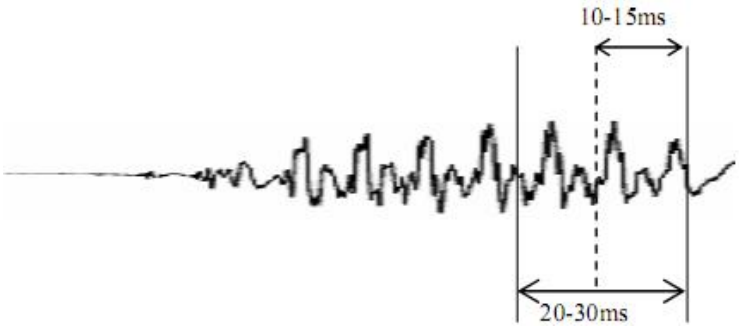
Một số hệ thống có tính thêm năng lượng (cũng lấy logarit) và đặc trưng delta (đạo hàm rời rạc theo thời gian của MFCC) nhằm thêm thông tin cho các pha sau của quá trình nhận dạng.

Các mục sau trình bày từng bước quá trình tính toán đặc trưng MFCC.

4.2. Chia khung và cửa sổ hoá

Lời nói là một tín hiệu không ổn định. Vì vậy, việc phân tích lời nói dùng DFT hay LPC phải được thực hiện trên các đoạn ngắn mà qua các đoạn này tín hiệu lời nói được xem là ổn định. Đây là lý do vì sao chúng ta cần chia tín hiệu đầu vào thành những đoạn con.

Rút trích đặc trưng (feature extraction) thường được thực hiện trên các cửa sổ từ 20 tới 30 ms. Để tránh mất mát thông tin do việc chia nhỏ, các đoạn gần kề thường được chồng lên nhau khoảng 30 tới 50% (khoảng 10 đến 15ms). Nguyên lý này được minh họa trong hình 4.2



Hình 4.2. Minh họa việc chia khung và độ chồng của các khung

Tín hiệu tiếng nói x[n] gồm L mẫu có được sau khi được chia thành các khung độ rộng 10ms (ứng với fs*0.01 mẫu) sẽ được cửa sổ hoá bằng cách nhân tín hiệu với một hàm cửa sổ độ rộng N.

$$x(n) = x_t(n) * w(n); n = 0..N-1$$

Hàm cửa sổ thường được dùng là hàm cửa sổ Hamming:

$$w(n) = 0.54 - 0.46 \times \cos(\frac{2\pi n}{N}) \; ; n = 0..N\text{-}1$$

Mục đích của việc sử dụng hàm cửa sổ là để làm mượt các cạnh của mỗi đoạn, để giảm tính không liên tục hay các thay đổi bất ngờ tại các điểm cuối của đoạn. Các mẫu trong vùng cửa sổ có giá trị khác 0 và các mẫu tại những điểm cuối của cửa sổ là 0.

4.3. Biến đổi Fourier rời rạc

Tín hiệu (của một frame) sau khi nhân với hàm cửa sổ, được chuyển sang miền tần số bằng biến đổi Fourier rời rạc:

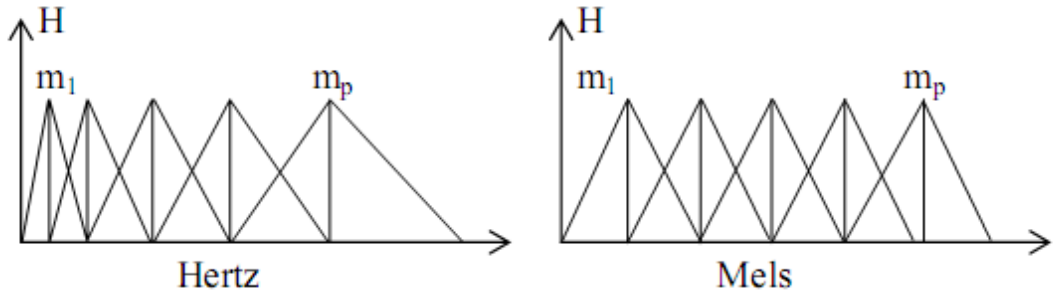
$$X(k) = \sum_{n=0}^{N-1} x(n).e^{-\frac{i2\pi kn}{N}} ; k = 0..N-1$$

4.4. Lọc qua các bộ lọc mel-scale

Phân tích cepstral theo thang đo mel MFCC. Phương pháp được xây dựng dựa trên sự cảm nhận của tai người đối với các dải tần số khác nhau. Với các tần số thấp (dưới 1000 Hz), độ cảm nhận của tai người là tuyến tính. Đối với các tần số cao, độ biến thiên tuân theo hàm logarit

Dải bộ lọc (filter bank) được áp dụng để loại bỏ một số biến đổi trong dải âm thanh. Nó là dải các bộ lọc tần số có dạng hình tam giác và được thiết kế để giữ lại các tần số mong muốn. Một chọn lựa rõ ràng là giữ lại chỉ những tần số mà tai người có thể nghe được.

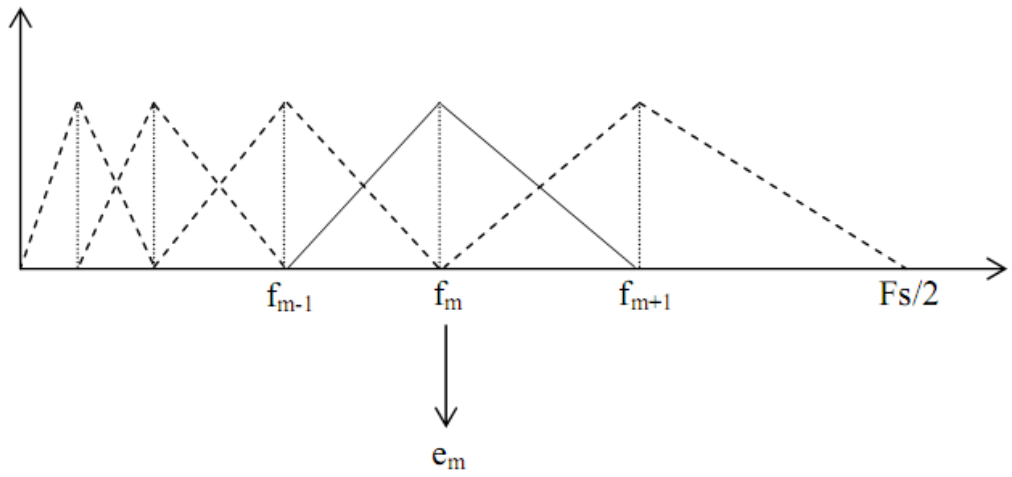
Dải bộ lọc có dạng hình tam giác này được đặt trên trục tần số sao cho tần số trung tâm của mỗi bộ lọc là tuyến tính theo mức mel (melody), và logarit theo mức tần số bình thường. Hơn nữa, các cạnh phải được đặt sao cho trùng với các tần số trung tâm của các bộ lọc lân cận. Chúng ta có thể hình tượng như sau:



Hình 4.4 Minh hoạ các bộ lọc mel-scale tam giác (triangle mel-scale filters)

Bây giờ giả sử chúng ta có dải các bộ lọc như hình 4.4, trong đó f_m là tần số trung tâm của bộ lọc thứ m; Fs là tần số lấy mẫu (sampling rate) và e_m là năng lượng đầu ra của bộ lọc thứ m. Lúc này e_m được tính theo biểu thức sau

$$e_m = \log \sum_{j=1}^{N} (h_m(j) * X(j))$$



Trong đó m=1..M (M là số bộ lọc, và M<<N) N là số mẫu tín hiệu đầu vào và X(j) là cường độ tại tần số j. $h_m(j)$ là bộ lọc thứ m, được định nghĩa:

$$h_m(j) = \begin{cases} 0 & f_j < f_{m-1} \\ \dfrac{f_j - f_{m-1}}{f_m - f_{m-1}} & f_{m-1} \le f_j < f_m \\ \dfrac{f_j - f_{m+1}}{f_m - f_{m+1}} & f_m \le f_j < f_m + 1 \\ 0 & f_j \ge f_{m+1} \end{cases}$$

Tần số mel (m) trung tâm của các bộ lọc được tính theo biểu thức

$$m = 2595*\log_{10}(1 + f/700) \text{ hay}$$

$$m = 1127.01048*\ln(1 + f/700)$$

Sau đó dựa vào mức mel, phân chia phạm vi cho từng bộ lọc

$$\Delta\varphi = (\varphi_{max} - \varphi_{min})/(M + 1) \qquad (*)$$

trong đó, φ_{max} là tần số mel cao nhất trong dải bộ lọc, được tính từ tần số f cao nhất (f_{max}) sử dụng biểu thức (*) bên trên; φ_{min} là tần số mel thấp nhất được tính từ tần số f thấp nhất (f_{min}). Chú ý, fmax thường là ½ của tỉ lệ lấy mẫu (sampling rate: Fs); fmin thường là 0.

Các tần số mel trung tâm được tính toán theo

$$\varphi_c(m) = m.\Delta\varphi \text{ , trong đó, } m = 1..M.$$

Để thu được các tần số trung tâm theo Hertz, chúng ta áp dụng biểu thức:

$$f_m = 700 * (10^{\varphi_c(m)/2595} - 1)$$

4.5. Logarit và biến đổi Fourier ngược

Lấy logarit của tín hiệu ở miền tần số (**spectrum**) rồi biến đổi Fourier ngược sẽ đưa tín hiệu về một miền gọi là **cepstrum** có đơn vị thời gian (thuật ngữ là **cepstrum** đảo ngược của âm đầu tiên trong từ **spectrum:** *spec*trum → *ceps*trum). Biến đổi từ spectrum sang cepstrum là một biến đổi đồng hình (homomorphic). *Biến đổi đồng hình chuyển biểu diễn tín hiệu từ dạng tích về dạng tổng, như vậy cho phép sử dụng các hệ tuyến tính để xử lý các tín hiệu không tuyến tính*. Công thức tính của bước này là:

$$c(n) = \sum_{k=1}^{M} \log(F(k)).\cos\left(\frac{i2\pi n(k-1)}{2M}\right); n = 1..p$$

Chú ý: mặc dù biến đổi từ spectrum sang cepstrum là biến đổi Fourier ngược, tuy nhiên do ta dùng spectrum và cepstrum thực nên chỉ sử dụng biến đổi cosine rời rạc (DCT) để tăng hiệu năng tính toán.

Sau bước này ta được vector cepstral (ở độ đo mel) p thành phần. Thông thường người ta thường nhân thêm vào kết quả một hàm cửa sổ sóng sin (gọi là thủ tục liftering) để giảm bớt ảnh hưởng của các biến đổi đến kết quả.

$$w(n) = 1 + \frac{L}{2}.\sin(\frac{n\pi}{L})$$
$$c(n) = c(n).w(n)$$

4.6. Tính toán năng lượng

Kèm thêm thông tin về năng lượng của tín hiệu sẽ tăng thêm thông tin cho nhận dạng (ví dụ: phân biệt các khoảng chứa tín hiệu âm và khoảng lặng, phân biệt vùng tín hiệu chứa nguyên âm và phụ âm…). Năng lượng của cả frame được tính qua công thức:

$$E = \sum_{n=0}^{N-1} (x(n))^2$$

4.7. Tính toán đặc trưng delta

Đặc trưng delta là đạo hàm bậc nhất (rời rạc) của đặc trưng theo thời gian. Có các đặc trưng delta sẽ tăng thêm thông tin cho nhận dạng (chẳng hạn: xác định các vùng mà phổ tín hiệu ổn định…). Đặc trưng delta được tính theo công thức:

$$C'(t) = \frac{\partial}{\partial t} C(t) = \sum_{u=-T}^{T} u.C(t+u)$$

Trong đó C(t) là cả vector cepstral tại thời điểm t. T là một hằng số chọn trước, thường thì người ta lấy T=3.

5. Mạng Nơron nhân tạo

Bộ não con người, dưới góc độ tính toán có thể coi là một hệ thống xử lý song song lớn và mật độ kết nối cao: phần tử xử lý là các nơ ron là một và kết nối là các dây thần kinh.

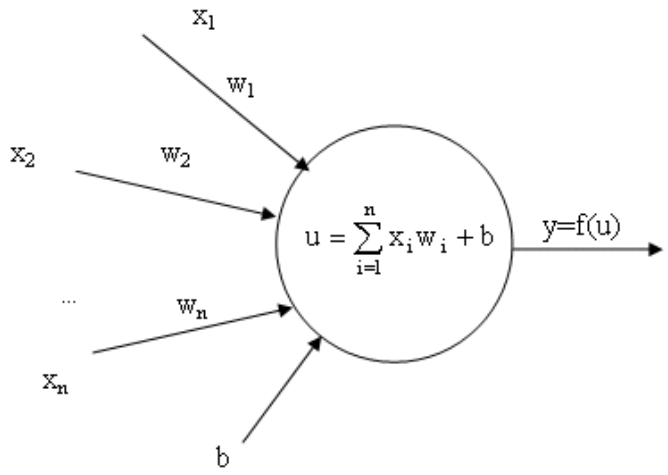
Khả năng tuyệt diệu của bộ não đã gợi nên những ý tưởng về việc mô phỏng chúng trong lĩnh vực tính toán. Và mạng nơ ron nhân tạo (artificial neural network -ANN) là kết quả của những ý tưởng đó.

5.1. Mô hình mạng Nơron nhân tạo

5.1.1. Mô hình một Nơron nhân tạo perceptron

Một nơron perceptron là một phần tử xử lý gồm:

- ***n*** đầu vào x_i, mỗi đầu vào được tham gia vào kết quả đầu ra với trọng số w_i;
- Một giá trị thực ***b*** gọi là ngưỡng (bias);
- Một hàm kích hoạt ***f***;
- Giá trị ra **y**.



Hình 5.1: Minh hoạ mô hình một nơron nhân tạo perceptron

Giá trị ra của perceptron được tính theo quy tắc sau:

$$u = \sum_{i=1}^{n} x_i w_i + b$$
$$y = f(u)$$

Hàm kích hoạt được sử dụng phổ biến là hàm sigmoid (còn gọi là hàm logistic) do tính phi tuyến và khả vi:

$$f(u) = \frac{1}{1+e^{-u}}$$

Ngoài ra còn có một số hàm kích hoạt khác: hàm tang hyperbolic, hàm softmax, ...

Khả năng tính toán của một nơron perceptron khá hạn chế, vì vậy, để cải thiện khả năng tính toán, người ta nối chúng thành mạng. Mô hình mạng đơn giản nhất là mạng perceptron truyền thẳng đa lớp MLP (Multi Layer Perceptron).

5.1.2. Mô hình mạng Nơron nhân tạo MLP (Multi Layer Perceptron)

Mạng nơron MLP n đầu vào, m đầu ra có mô hình như sau:

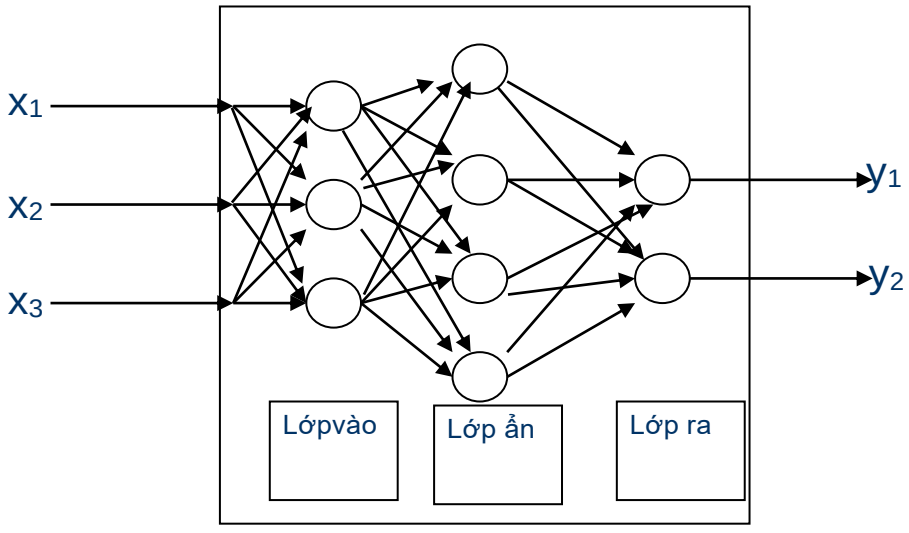
- Các nơron được chia thành các lớp: lớp sau được nối với lớp trước. Lớp đầu tiên là lớp vào (input - nhận đầu vào), lớp cuối cùng là lớp ra (output - cho đầu ra). Giữa lớp vào và lớp ra là các lớp ẩn (hidden). Thông thường chỉ có một lớp ẩn;

- Tất cả các nơron cùng một lớp sử dụng chung một vector đầu vào. Mỗi lớp khi nhận một vector đầu vào sẽ tính đầu ra của mỗi nơron, kết hợp thành một vector và lấy đó làm đầu vào cho lớp sau;

- Mạng MLP nhận đầu vào là một vector n thành phần, lấy đó làm đầu vào của lớp input và tính toán cho đến khi lớp output có đầu ra, lấy đó là đầu ra của mạng: một vector m thành phần;

- Toàn bộ các nơron của toàn mạng sử dụng chung một hàm kích hoạt, thường là hàm logistic.

Ngoài lớp vào và lớp ra, mạng MLP thường có một hay nhiều lớp ẩn. Thông thường người ta chỉ sử dụng một lớp ẩn. Vì vậy đôi khi người ta hay đồng nhất MLP với MLP 3 lớp.



Hình 5.2: Minh hoạ mô hình mạng perceptron 3 lớp (MLP)

Như vậy xét dưới góc độ toán học thì mạng MLP biểu diễn một hàm phi tuyến từ R^n vào R^m. Người ta cũng chứng minh được rằng: ***"một hàm phi tuyến liên tục bất kì có thể xấp xỉ với độ chính xác tuỳ ý bằng mạng MLP"*** (định lý Kolmogorov).

Mạng MLP n×p×m (n đầu vào, m đầu ra, p nơron ẩn) được biểu diễn bằng 2 ma trận trọng số w_1 cỡ n×p, w_2 cỡ p×m và 2 vector ngưỡng b_1 p phần tử, b_2 m phần tử. Lớp input của MLP chỉ có tác dụng nhận đầu vào, hoàn toàn không thực hiện tính toán.

Khi đó tính toán đầu ra y của mạng theo đầu vào x như sau:

$$u = x.w_1 + b_1$$
$$z = f(u)$$
$$v = z.w_2 + b_2$$
$$y = f(v)$$

Ở đây, u, v, z là các vector. Viết z=f(u) có nghĩa là $z_i=f(u_i)$ với mọi i.

Để biểu diễn được một hàm nào đó, mạng MLP cần được huấn luyện.

5.1.3. Huấn luyện mạng Nơron nhân tạo MLP

Để mạng nơron biểu diễn được hàm f, ta cần một bộ dữ liệu gồm N cặp vector (x_i, t_i), trong đó x_i thuộc tập xác định của f và t_i là giá trị của f tại x_i:

$$t_i = f(x_i)$$

Mạng MLP sẽ biểu diễn được hàm f nếu cho đầu vào của mạng là x_i thì đầu ra của mạng là t_i. Thường thì MLP chỉ biểu diễn được xấp xỉ hàm f, do đó ta mong muốn nếu mạng cho đầu ra thực tế là y_i thì y_i càng gần t_i càng tốt.

Như vậy bài toán huấn luyện mạng là cho bộ dữ liệu huấn luyện gồm N cặp vector (x_i, t_i), cần điều chỉnh các trọng số của mạng sao cho tổng sai số của mạng trên bộ dữ liệu là nhỏ nhất:

$$E = \sum_{i=1}^{N} \|t_i - y_i\| \rightarrow \min$$

Trong đó y_i là đầu ra thực tế của mạng ứng với đầu vào x_i.

Thuật toán huấn luyện MLP phổ biến nhất là thuật toán lan truyền ngược lỗi (back-propagation training). Thuật toán có đầu vào là tập mẫu $\{(x_i, t_i)\}$, đầu ra là bộ trọng số của mạng.

Các bước tiến hành huấn luyện như sau:

Bước 1: Khởi tạo trọng số của mạng: w_{ij} được gán các giá trị ngẫu nhiên, nhỏ (nằm trong miền $[-\alpha, \alpha]$).

Bước 2: Với mỗi cặp (x,t) trong bộ dữ liệu huấn luyện:

Giả sử $x = (x_1, ..., x_n)$. Ta thực hiện:

a) Lan truyền x qua mạng để có y (theo công thức tại mục 4.3);

b) Tính sai số e của mạng: e=t-y;

c) Hiệu chỉnh các trọng số liên kết nơron dẫn tới lớp ra w_{ij} từ nơron j tại lớp ẩn tới nơron i tại lớp ra:

$$w_{ij} = w_{ij} + \Delta w_{ij}$$

w_{ij} là trọng số giữa nơron i ở lớp trước và nơron j ở lớp sau.

Δw_{ij} được tính theo công thức sau: $\Delta w_{ij}=\rho\delta_j y_i$, với:

- ρ là hằng số tốc độ học (learning rate),
- y_i là đầu ra của nơron i (nếu i là nơron lớp input thì thay y_i bằng x_i);
- δ_j là sai số tại nơron j.
    - Nếu j là nơron lớp ra (output layer) thì δ_j được tính theo công thức:

$$\delta_j = y_j(1-y_j)(tj-y_j);$$

    - Nếu j là nơron lớp ẩn thì được tính theo công thức:

$$d_j = y_j(1-y_j)\sum_k \delta_k w_{jk}$$

với đó k là các nơron của lớp sau lớp j

Việc đưa mẫu huấn luyện qua mạng, tính toán và cập nhật trọng số được tiến hành với tất cả phần tử trong bộ mẫu (có thể chọn ngẫu nhiên hoặc tuần tự). Quá trình sẽ dừng lại khi sai số trung bình (hoặc tổng sai số) nhỏ hơn một giá trị cho trước hoặc thay đổi không đáng kể (tức là quá trình huấn luyện hội tụ).

5.1.4. Ưu điểm và nhược điểm của mạng nơron nhân tạo

Các nghiên cứu cả về mặt lý thuyết và thực tế cho thấy mạng nơron có những ưu điểm sau:

- Có thể xấp xỉ một hệ phi tuyến động (nonlinear dynamical system) với độ chính xác bất kì;

- Có khả năng miễn nhiễu (robustness) và chịu sai hỏng (fault tolerance) cao. Chẳng hạn mạng có thể nhận các dữ liệu bị sai lệch hoặc không đầy đủ mà vẫn hoạt động được;

- Có khả năng thích ứng: mạng nơron có thể "học" (learn) và "điều chỉnh" (adapt) trong quá trình hoạt động. Đây là điểm đáng chú ý nhất của mạng nơron trong nhận dạng tiếng nói. Đặc điểm này của mạng cho phép ta hi vọng xây dựng được một hệ nhận dạng có thể "học tập" để nâng cao khả năng nhận dạng trong khi hoạt động;

- Có khả năng tổng quát hoá (generalize) tốt và phân lớp (classify) mạng.

Nhưng mạng nơron cũng không phải là công cụ vạn năng cho mọi vấn đề, vì chúng cũng có nhiều nhược điểm:

- Chỉ xử lí được các dữ liệu số. Cần tích hợp thêm nhiều thành phần khác (ví dụ: các hệ mờ, các bộ số hoá...) để có thể xử lí những dữ liệu phi số;

- Hiệu năng của mạng phụ thuộc bộ dữ liệu huấn luyện. Để đảm bảo hiệu năng, mạng cần được huấn luyện với lượng dữ liệu lớn. Quá trình huấn luyện do đó rất dài. Mặt khác nếu bộ dữ liệu được chuẩn bị không tốt thì mạng có khả năng tổng quát hoá thấp;

- Mạng nơron gần như là một "hộp đen" đối với các phân tích. Rất khó xác định được sự phân bố thông tin và xử lí trên các phần tử của mạng;

- Không có một phương pháp chung nào để xác định cấu trúc mạng phù hợp từng bài toán. Nhà nghiên cứu phải tiến hành thử nghiệm hoặc dựa vào kinh nghiệm để xác định;

- Các thuật toán huấn luyện hiện chưa đảm bảo tránh quá trình huấn luyện rơi vào một cực trị địa phương. Hơn nữa sai số huấn luyện giảm không đồng nghĩa với tăng hiệu năng hoạt động của mạng;

- Mạng cấu trúc lớn cài đặt bằng phần mềm trên máy tính hoạt động rất chậm. Việc xây dựng mạng nơron bằng phần cứng vẫn còn đang được nghiên cứu.

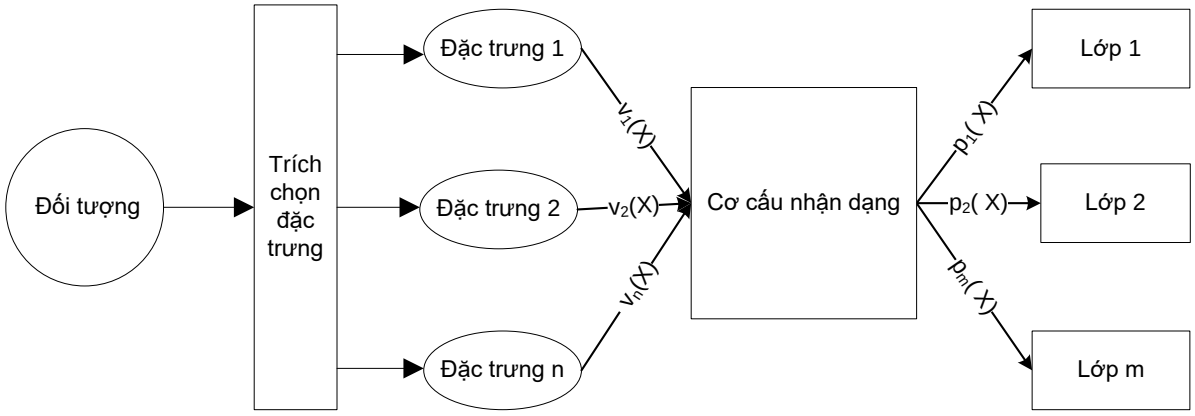
5.2. Sử dụng mạng Nơron nhân tạo trong nhận dạng mẫu

5.2.1. Một phương pháp tiếp cận dựa vào xác suất phân lớp

Bài toán nhận dạng sẽ được giải quyết nếu chúng ta xây dựng được một cơ cấu nhận dạng có:

- Đầu vào là đặc trưng của đối tượng cần nhận dạng;

- Đầu ra là xác suất phân lớp hoặc độ giống (likelihood), độ tương tự (similarity) của đối tượng với những lớp mẫu đã huấn luyện.



Hình 5.3: Minh hoạ mô hình nhận dạng bằng cơ cấu nhận dạng dựa theo xác suất phân lớp

Chúng ta thấy rằng có thể dùng MLP để là một cơ cấu nhận dạng như vậy: Nếu đặc trưng của đối tượng là n số thực và có m lớp mẫu thì ta sẽ xây dựng một MLP n đầu vào, m đầu ra. Đầu vào là các đặc trưng của đối tượng, đầu ra là độ tương tự của đối tượng với mỗi lớp mẫu. MLP sẽ được huấn luyện dựa trên bộ dữ liệu huấn luyện được chuẩn bị trước để tìm mối liên hệ giữa đầu vào và đầu ra (**học** và **tổng quát hoá**). Những vấn đề này sẽ được trình bày kỹ trong những phần tiếp theo.

5.2.2. Nhược điểm của mạng MLP trong nhận dạng tiếng nói

MLP có một số nhược điểm sau khi sử dụng trong nhận dạng tiếng nói:

- Có đầu vào cố định (trong khi tín hiệu tiếng nói là tín hiệu thay đổi theo thời gian: mỗi lần phát âm cho các từ có độ dài thường không bằng nhau);

- Chi phí huấn luyện tốn kém (thời gian, không gian lưu trữ).

Do đó MLP thường chỉ cho kết quả cao trong nhận dạng với bộ từ vựng nhỏ và phân biệt (độ tương tự của các lớp mẫu thấp).

5.2.3. Một số phương pháp tiếp cận khác

Ngoài cách tiếp cận như ở phần 4.2.1, còn có nhiều cách tiếp cận khác đối với nhận dạng tiếng nói bằng mạng nơron:

- Dùng mô hình mạng TDNN (mạng nơron thời gian trễ): là mô hình cải tiến của MLP, có cơ chế để tích hợp thông tin về thời gian (các nơron trễ) khi đưa các mẫu tiếng nói qua mạng. Mô hình này nhằm giải quyết vấn đề về sự phụ thuộc thời gian của tín hiệu tiếng nói;

- Kết hợp MLP và HMM: sử dụng MLP là bộ đo xác suất phát xạ vector quan sát. Cách tiếp cận này kết hợp ưu điểm của cả 2 mô hình.

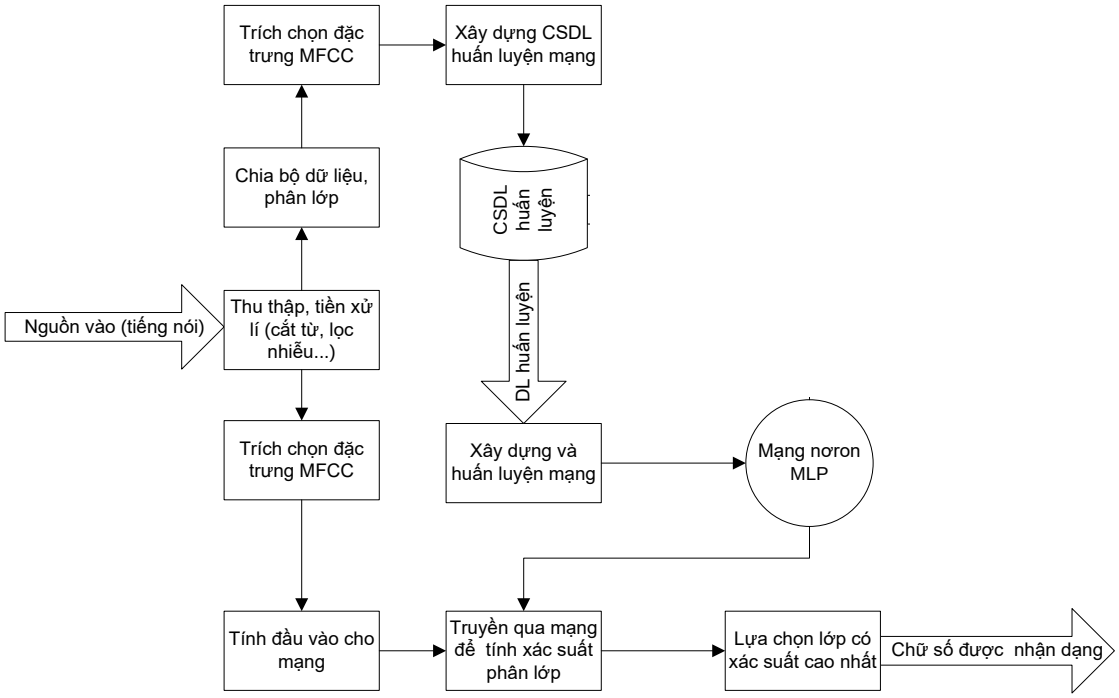
6. Xây dựng hệ nhận dạng chữ số tiếng Việt

6.1. Mô tả chung về hệ thống

Hệ thống nhận dạng được xây dựng trong tiểu luận dựa trên phương pháp nhận dạng mẫu, sử dụng mạng nơron làm cơ cấu nhận dạng như sau:

- Phương pháp: nhận dạng từ đơn (isolate word recognition);
- Input: file wav, mỗi file chỉ chứa một từ. Hoặc ghi âm trực tiếp;
- Output: chữ số được nhận dạng trong file đầu vào;
- Bộ từ vựng: 10 từ đơn âm các số tiếng Việt ("không", "một", "hai"... "chín")

6.2. Sơ đồ khối của hệ thống



Hình 6.1: Minh hoạ sơ đồ khối hệ thống nhận dạng tiếng nói các chữ số tiếng Việt bằng mạng nơron MLP

Các phần tiếp theo mô tả cụ thể từng chức năng của hệ thống.

6.3. Thu thập và tiền xử lí tín hiệu tiếng nói

Thu thập và tiền xử lí tín hiệu tiếng nói ở giai đoạn huấn luyện được thực hiện bằng phương pháp thủ công: sử dụng phần mềm ghi âm, lọc nhiễu và cắt thành các từ riêng rẽ, mỗi từ ghi vào một file (tên file ghi từ tương ứng).

Bộ dữ liệu do chúng em tự xây dựng gồm:

- Mười file wav 16 bit 8kHz, mỗi file là phát âm của một từ: Mười từ là các chữ số tiếng Việt từ **"không"** đến **"chín"**.

Ở giai đoạn nhận dạng, việc thu thập và tiền xử lí (cắt các vùng không chứa tín hiệu tiếng nói) được thực hiện bởi phần chương trình

6.4. Phân chia bộ dữ liệu và phân lớp

Bộ dữ liệu huấn luyện có 10 từ nên chúng tôi chia toàn bộ vector đặc trưng thành 10 lớp, mỗi vector ứng với lớp của từ tương ứng. (Từ "không" thuộc lớp 0, từ "hai" thuộc lớp 2,... từ "chín" thuộc lớp 9).

6.5. Tính đầu vào cho mạng

Vì các file wav có độ dài ngắn khác nhau nên dãy các vector đặc trưng MFCC tương ứng cũng không có cùng số phần tử. Nhưng đầu vào của mạng nơron MLP lại phải cố định. Do đó cần thực hiện một biện pháp đơn giản là *chia dãy đặc trưng thành 5 phần đều nhau, tính trung bình của từng phần được 5 vector rồi ghép lại thành một vector. Kết quả đầu vào của mạng nơron là một vector 60 thành phần.*

Các thao tác về file âm thanh cũng như trích chọn đặc trưng MFCC được cài đặt gọn gàng trong lớp **Wavefile** với các phương thức chính cần cho nhận dạng là

+ toMFCC(x): x là vector đặc trưng cho bài toán nhận dạng này**.**

6.7. Xây dựng, huấn luyện mạng

- Số nơ ron lớp vào: 60 (ứng với 60 hệ số MFCC)

- Số nơ ron lớp ẩn: 100

- Số nơ ron lớp ra: 10 (ứng với 10 số)

- Hàm kích hoạt: Sigmoid

- Hệ số học: 1.2

Để cài đặt mạng, chúng em xây dựng **lớp thư viện MLP** với các phương thức cơ bản như sau:

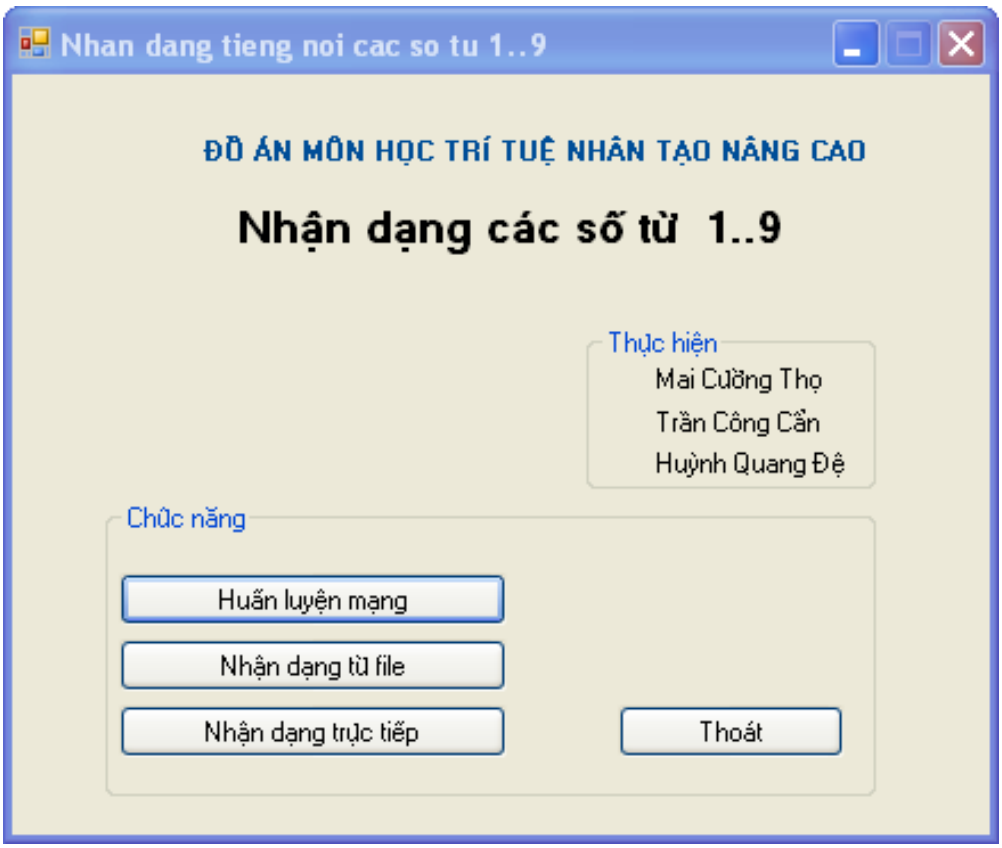
+ netTrain(traindata)

+ netLoad (tên mạng)

+ netSave(tên mạng) và **+ netRun(dữ liệu vào, kết quả)**

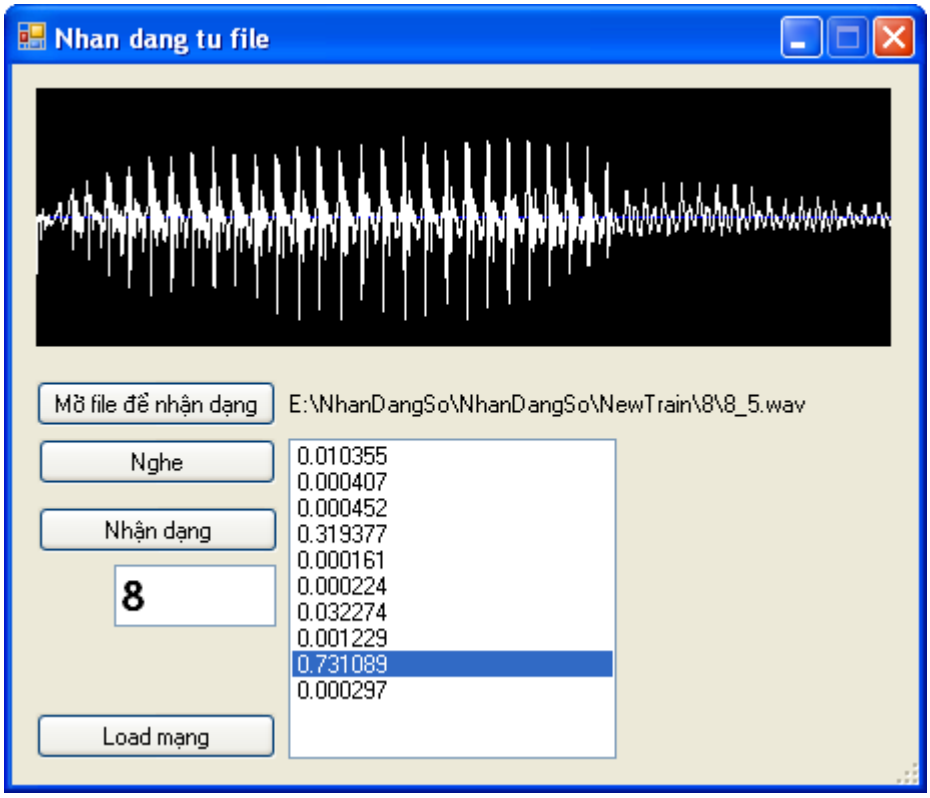
6.8. Giao diện phần mềm demo

+ Giao diện khởi động của phần mềm



Huấn luyện mạng: Chọn số lần lặp và thư mục dữ liệu âm thanh cần luyện. Tên mạng mạc định là so.net, sau khi luyện xong, người dùng cần lưu lại.

Form Nhận dạng: Trước hết cần phải Load mạng vào cho chương trình → sau đó chọn tên file âm thanh cần nhận dạng, → rỗi nhẫn nút nhận dạng. Con số ngay phía dưới nút là con số hệ thống nhận dạng được. Listbox bên cạnh hiển thị xác suất phân lớp theo thứ tự từ lớp 0 đến lớp 9. Có thể nhấn nút nghe để kiểm chứng.



7. Kết luận + Một số hướng mở rộng của tiểu luận

Qua thực hiện đồ án môn học Trí tuệ nhân tạo nâng cao, nhóm đã cài đặt được phần mềm demo việc sử dụng mạng nơ ron nhân tạo trong nhận dạng.

Chương trình thực hiện trích chọn đặc trưng MFCC mà chưa dùng thêm các đặc trưng khác, nên hiệu quả nhận dạng đúng chưa được cao. Trong thời gian tới, nhóm sẽ sử dụng thêm các đặc trưng khác dùng cho nhận dạng để nâng cao hiệu quả.

Các nghiên cứu cho thấy mô hình Markov ẩn (HMM) đang cho kết quả nhận dạng cao nhất. Hướng nghiên cứu mới của đề tài là tìm cách kết hợp mạng nơron và mô hình Markov ẩn nhằm kết hợp ưu điểm của hai mô hình.

Mặt khác, đối với bộ từ vựng nhỏ thì nhận dạng từ đơn (âm tiết) là thích hợp. Tuy nhiên với hệ nhận dạng cỡ lớn, nhất là hệ nhận dạng tiếng Việt hoàn chỉnh (cỡ

6.700 âm tiết) thì chọn đơn vị nhận dạng là âm tiết là không hợp lí lắm. Một hướng nghiên cứu khác của đề tài là nhận dạng đối với đơn vị nhỏ hơn âm tiết là âm vị. Tức là xây dựng các hệ nhận dạng có chức năng:

- Phân biệt được nhiễu nền (khoảng lặng), phụ âm, nguyên âm;
- Nhận dạng phụ âm (phân biệt được các phụ âm khác nhau);
- Nhận dạng nguyên âm (phân biệt được các nguyên âm khác nhau);
- Nhận dạng thanh điệu của âm tiết.

8. Tài liệu tham khảo

Kí hiệu	Tác giả	Tên tài liệu	Ghi chú
[1]	Nguyễn Đình Thuân	***Bài giảng Trí tuệ nhân tạo nâng cao***	
[2]	Neuronale Netze	***Neur Neural Nets, Background Back-propagation***	pdf
[3]	Edmondo Trentin	***Multilayer Perceptron (MLP): the Backpropagation (BP) Algorithm***	pdf
[4]	Trịnh Văn Loan	***Bài giảng Xử lý tiếng nói***	pdf
[5]	Đoàn Thiện Thuật	***Ngữ âm tiếng Việt***	Sách NXB ĐHQG Hà Nội - 2003
[6]	Rabiner L.R, Huang B. H	***Fundamentals of Speech Recognition***	Sách NXB Prentice Hall - 1993