Contents

[Chương 1. Giới thiệu 2](#_Toc357637834)

[1.1 Giới thiệu chung tương tác người máy bằng tiếng nói trên thiết bị di động 2](#_Toc357637835)

[1.1.1 Tương tác người – máy 2](#_Toc357637836)

[1.1.2 Tương tác người – máy bằng tiếng nói 3](#_Toc357637837)

[1.1.3 Tương tác người – máy trên thiết bị di động. 4](#_Toc357637838)

[1.2 Giới thiệu đồ án 4](#_Toc357637839)

[1.3 Kết luận chương 4](#_Toc357637840)

[Chương 2. Hệ thống thư kí ảo VIVA 4](#_Toc357637841)

[2.1 Giới thiệu đề tài tương tác điện thoại di động bằng tiếng nói 4](#_Toc357637842)

[2.1.1 Giới thiệu chung 4](#_Toc357637843)

[2.1.2 Giới thiệu đề tài 7](#_Toc357637844)

[2.1.3 Miền bài toán 8](#_Toc357637845)

[2.1.4 Công cụ và chuẩn bị 9](#_Toc357637846)

[2.2 Thiết kế hệ thống 9](#_Toc357637847)

[2.2.1 Sơ đồ hệ thống 9](#_Toc357637848)

[2.2.2 Mô tả các khối 9](#_Toc357637849)

[2.2.3 Biểu đồ ca sử dụng 11](#_Toc357637850)

[2.2.4 Biểu đồ hoạt động 12](#_Toc357637851)

[2.2.5 Biểu đồ khối 13](#_Toc357637852)

[2.3 Triển khai hệ thống 13](#_Toc357637853)

[2.3.1 Hệ thống nhận dạng tiếng nói 13](#_Toc357637854)

[2.3.2 Hệ thống hiểu ngôn ngữ nói 21](#_Toc357637855)

[Chương 3 Xây dựng hệ thống hiểu ngôn ngữ tự nhiên 26](#_Toc357637856)

[3.1 Lý thuyết automat 26](#_Toc357637857)

[3.1.1 Định nghĩa 26](#_Toc357637858)

[3.1.2 Phân loại và ứng dụng của Automat. 28](#_Toc357637859)

[3.1.3 Otomat hữu hạn đơn định –DFA (Deterministic Finite Automata) 28](#_Toc357637860)

[3.1.4 Otomat hữu hạn không đơn định – NFA (Nondeterministic Finite Automata) 30](#_Toc357637861)

[3.2 Tách từ trong xử lý ngôn ngữ tự nhiên 30](#_Toc357637862)

[3.2.2 Ứng dụng bài toán tách từ 31](#_Toc357637863)

[3.2.3 Một số vấn đề mở rộng 32](#_Toc357637864)

[3.3 Gán nhãn từ loại 32](#_Toc357637865)

[3.3.1 Vấn đề gán nhãn từ loại tiếng Việt 32](#_Toc357637866)

[3.3.2 Bảng nhãn từ loại Tiếng Việt 33](#_Toc357637867)

[3.3.3 Các bước gán nhãn từ loại 34](#_Toc357637868)

[3.3.4 Các trường hợp nhập nhằng 35](#_Toc357637869)

[3.4 Phân tích cú pháp trong xử lý ngôn ngữ tự nhiên 38](#_Toc357637870)

[3.4.1 Biểu diễn cú pháp 38](#_Toc357637871)

[3.4.2 Phương pháp phân tích cú pháp từ trên xuống (Top-Down Parsing) 39](#_Toc357637872)

[3.4.3 Phương pháp phân tích cú pháp từ dưới lên (Bottom – Up Parsing) 40](#_Toc357637873)

[3.4.4 Một số vấn đề trong phân tích cú pháp tiếng Việt 40](#_Toc357637874)

# Chương 1. Giới thiệu

## 1.1 Giới thiệu chung tương tác người máy bằng tiếng nói trên thiết bị di động

### 1.1.1 Tương tác người – máy

* Định nghĩa : Định nghĩa thứ nhất do Backer và Buxton nêu ra năm 1987 :

“Tương tác người – máy là tập các quá trình, đối thoại và các hành động, qua đó con người sử dụng và tương tác với máy tính”

Định nghĩa thứ hai nêu lên cả một quy trình hay vòng đời của tương tác người – máy và cả nghiên cứu các hiện tượng xảy ra trên đó:

“Tương tác người – máy là một lĩnh vực liên quan đến thiết kế, đánh giá và cài đặt hệt thống máy tính tương tác con người sử dụng và nghiên cứu các hiện tượng xảy ra trên đó”

* Các dạng tương tác người máy
  + Giao tiếp dòng lệnh
  + Giao tiếp bảng chọn(menu)
  + Giao tiếp bằng ngôn ngữ tự nhiên
  + Giao tiếp dạng hỏi đáp và truy vấn
  + Giao tiếp dạng form điền

### 1.1.2 Tương tác người – máy bằng tiếng nói

Đây là một loại giao tiếp bằng ngôn ngữ tự nhiên, một trong những dạng tương tác hấp dẫn nhất giữa người dùng và máy tính. Việc hiểu ngôn ngữ tự nhiên là một chủ đề được quan tâm và nghiên cứu của nhiều lĩnh vực. Việc xử lí âm thanh tiếng nói cũng là vấn đề khó khắn. Bên cạnh đó, sự nhập nhằng của ngôn ngữ tự nhiên cũng gây lên khó hiểu cho máy. Sự nhập nhằng của ngôn ngữ tự nhiên thể hiện ở nhiều mức độ: cú pháp,cấu trúc…, làm cho một câu có thể không rõ ràng.



*Giao tiếp bằng ngôn ngữ tự nhiên*

Ngoài ra vẫn tồn tại sự không rõ ràng trong nghĩa của từ. Con người thường dựa vào ngữ cảnh để phân loại các nhập nhằng này. Tuy nhiên điều này với máy thì tương đối khó,chúng ta chỉ có thể sử dụng ngôn ngữ tự nhiên trong một số lĩnh vực cụ thể.

Chúng ta có thể bắt gặp kiểu tương tác này trong một số ngôi nhà thông minh, giao tiếp với một số Robot, hay trên những chiếc SmartPhone hiện nay.

### 1.1.3 Tương tác người – máy trên thiết bị di động.

Với sự phát triển của khoa học công nghệ, các thiết bị di động đã trở nên vô cùng mạnh mẽ và gần gũi với con người. Trong đó có sự vươn lên mạnh mẽ của 2 hệ điều hành Android và IOS, việc điều khiển các thiết bị trở lên rất dễ dàng hơn rất nhiều. Tương tác cơ học đang dần được thay thế. Chúng ta chỉ cần những cái chạm tay nhẹ đơn giản cũng có thể điều khiển được thiết bị di động. Ngoài ra có thể điều khiển bằng giọng nói, thậm trí bằng hình ảnh, hay sóng não – những điều vẫn đang được nghiên cứu rất nhiều và chắc chắn sẽ được ứng dụng trong tương lai.

## 1.2 Giới thiệu đồ án

## 1.3 Kết luận chương

# Chương 2. Hệ thống thư kí ảo VIVA

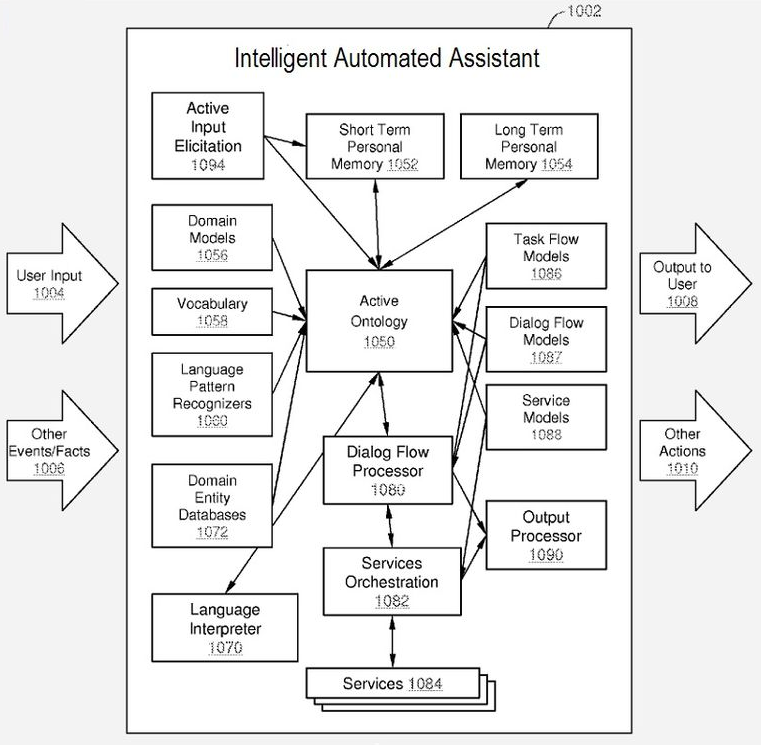
## 2.1 Giới thiệu đề tài tương tác điện thoại di động bằng tiếng nói

### 2.1.1 Giới thiệu chung

Hiện này trên thế giới có rất nhiều công ty tập đoàn lớn đang đầu tư vào lĩnh vực di động. Với xu hướng thời đại, đây có thể nói là một thị trường béo bở cho những người tham gia. Ở thị phần hệ điều hành cho di động, IOS và Android đang cùng nhau vươn lên và gần như không có đối thủ. Dù gần đây Microsoft có sự nỗ lực nhưng Window Phone vẫn chưa thể có được một vị thế xứng tầm để so sánh với 2 đại gia còn lại.

Bên cạnh đó, rất nhiều nhà sản xuất di động hay phần mềm vẫn đang nỗ lực thiết kế, trang bị cho những chiếc điện thoại thông minh nhiều ứng dụng, tiện ích thông minh hơn nữa. Việc điều khiển điện thoại bằng tiếng nói cũng rất hấp dẫn với các nhà phát triển và cũng gây rất nhiều thích thú với người sử dụng. Các đây 3 năm Apple đã bỏ ra 200 triệu đô la để mua lại Siri, và ngay sau đó 1 năm đã đưa ứng dụng Siri vào chiếc Iphone 4s đình đám của mình. Ngay lập tức đã gây được tiếng vang lớn trên thế giới. Ngoài ra còn có Google Talk, và Nuance Talk là những ứng dụng đình đám trong lĩnh vực này. Dưới đây là tổng quan về các hệ thống này:

* Siri
  + Là một ứng dụng trợ lý ảo của Apple đưa ra, hoạt động trên IOS 5.0 trở lên. Ứng dụng sử dụng giao tiếp ngôn ngữ tự nhiên (tiếng anh) để tương tác với người dùng.
  + Chức năng : Giao tiếp thông minh, chào hỏi, trò chuyện, gửi tin nhắn, thực hiện cuộc gọi, xem thông tin, nghe nhạc, đặt hẹn giờ, thông tin giao thông, địa điểm, đường đi,.…
  + Kiến trúc :



* Google Now
* Google Now là hệ thống hỗ trợ thông minh được phát triển bởi Google, cho 2 hệ điều hành Android và IOS(ra sau). Cũng giống như Siri, Google Now chủ yếu sử dụng giao tiếp tiếng nói để tương tác với người dùng. Tuy nhiên cũng như Siri, Google Now vẫn mới chỉ có phiên bản giao tiếp bằng Tiếng Anh.
* Chức năng:

Google Now cũng có rất nhiều chức năng thông minh như trả lời các thông tin thời tiết, giao thông, phim, chứng khoán, các địa điểm, du lịch, đặt báo hiệu, trò chuyện,… và đang dần được cải tiến rất nhiều.

Ngoài ra còn có rất nhiều ứng dụng trợ lý ảo tương tác bằng tiếng nói khác nhưng hiện tại chưa có ứng dụng nào sử dụng tiếng Việt cả.

### 2.1.2 Giới thiệu đề tài

Sau một thời gian nghiên cứu, phân tích, nảy ra ý tưởng, các thầy và trò thuộc Viện nghiên cứu quốc tế Mica đã đề xuất và thực hiện dự án xây dựng hệ thống thư kí ảo tiếng Việt – VIVA (VIetnamese Voice Assistant) với mục tiêu giúp Smartphone :

* Nghe được: nhận dạng tiếng nói (tiếng Việt).
* Hiểu được: hiểu được câu lệnh của người dùng.
* Nói được: phản hồi lại người dùng qua tiếng nói.
* Làm thư ký: giúp người dùng thực hiện các thao tác trên điện thoại mà không cần chạm vào bàn phím hay mèn hình để điều khiển, các tương tác sẽ được yêu cầu, trả lời qua tiếng nói.

Dự án được thực hiện chính thức vào cuối năm 2012 với sự tham gia của 7 thành viên (kỹ sư và sinh viên) dưới sự dẫn dắt của TS Trần Đỗ Đạt, TS Nguyễn Quốc Cường và TS Mạc Đăng Khoa với mục tiêu

* Thực hiện những nghiên cứu khoa học trong bài toán tương tác với điện thoại thông minh bằng tiếng nói.
* Từ những nghiên cứu đạt được, xây dựng ứng dụng ViVa trên điện thoại chạy hệ điều hành Android trong vai trò làm thư ký ảo cho người.

Dự án bao gồm 4 khối nghiên cứu chính:

* Khối nhận dạng tiếng nói: thực hiện bởi sinh viên Nguyễn Thanh Tùng, Nguyễn Hữu Hiển.
* Khối hiểu ngôn ngữ: thực hiện bởi kỹ sư Nguyễn Quang Thắng, sinh viên Phạm Thức Hưng, Nguyễn Văn Hiếu.
* Khối thực thi: sinh viên Nguyễn Duy Bình, Nguyễn Văn Bảo, Nguyễn Văn Hiếu.
* Khối tổng hợp tiếng nói:sinh viên Nguyễn Duy Bình, Nguyễn Văn Bảo.

### 2.1.3 Miền bài toán

Ứng dụng thư ký ảo có khả năng nghe, nói, hiểu, giúp người dùng tương tác với điện thoại bằng tiếng nói.

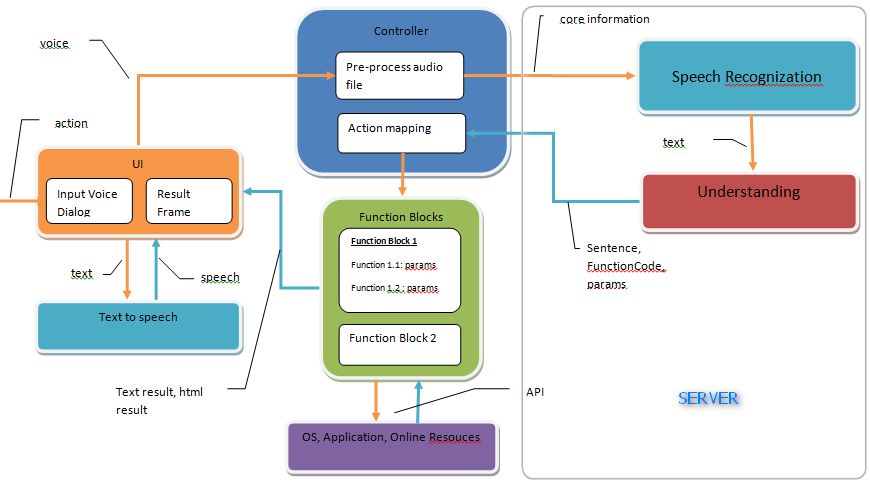
* Giao tiếp đơn giản.
  + Chào hỏi
* Ứng dụng nhắn tin
  + Gửi tin nhắn tới số điện thoại.
  + Gửi tin nhắn tới tên liên lạc trong danh bạ.
  + Thông báo, đọc, trả lời tin nhắn tới.
* Ứng dụng gọi điện:
  + Gọi điện tới số điện thoại.
  + Gọi điện tới tên liên lạc trong danh bạ.
* Ứng dụng Email:
  + Soạn, gửi email.
  + Thông báo, đọc, trả lời email đến.
* Notification.
  + Đọc các thông báo của hệ điều hành.(Nếu ngôn ngữ hệ điều hành đang sử dụng khác tiếng Việt thì qua một bước dịch sang tiếng Việt).
* Truy vấn thông tin: người dùng hỏi, thư ký ảo tìm & trả lời các thông tin.
  + Thông tin thời tiết.
  + Tỉ giá ngoại tệ.
  + Giá vàng
  + Chứng khoán.
* Nguồn thông tin trên điện thoại:
  + Thời gian, ngày tháng.
* Đặt báo thức
* Note công việc
* Nhắc nhở

### 2.1.4 Công cụ và chuẩn bị

* Thiết bị thử nghiệm: SmartPhone hệ điều hành Android 2.2 (API 8) trở lên, máy tính, máy ghi âm,…

## 2.2 Thiết kế hệ thống

### 2.2.1 Sơ đồ hệ thống



## Mô tả các khối

1. Khối UI :

* Đầu vào : Tương tác từ người dùng (bằng tay), Tiếng nói
* Đầu ra : Hình ảnh,Tiếng nói
* Các chức năng :
  + Thu âm và chuyển sang khối nhận dạng
  + Nhận dạng tương tác người dùng
  + Hiển thị thông tin ( hình ảnh, text,…)
  + Gọi khối tổng hợp để phản hồi thông tin bằng tiếng nói.

1. Khối nhận dạng:

* Đầu vào: tiếng nói
* Đầu ra: dữ liệu text.
* Chức năng: nhận dạng giọng nói
  + Trên server, sau đó gửi kết quả về smartphone
  + Nhúng phần trực tiếp trên smartphone

Chú ý: xây dựng cơ sở dữ liệu, vấn đề thu âm trên nhiều loại smartphone ( phải có nhiều loại máy để thu, huấn luyện, các tính chất: f mẫu,…)

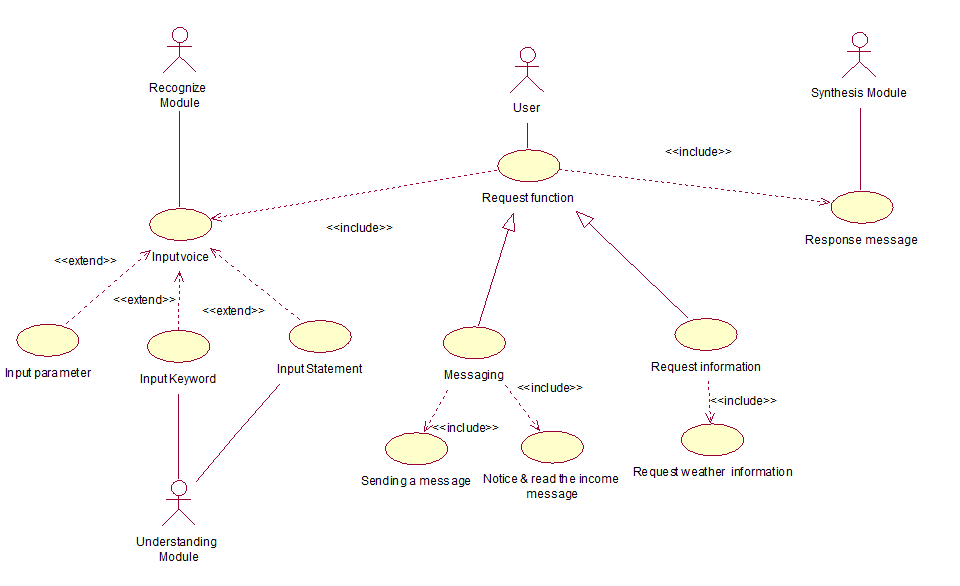
1. Khối hiểu:

* Thực hiện trên Server, sau khối nhận dạng hoặc thực hiện ngay trên smartphone.
* Đầu vào:text
* Đầu ra: Các mã lệnh và các giá trị các tham số
* Chức năng:
  + Tách từ
  + Phân tích văn bản
  + Phân tích ngữ nghĩa ( thống kê: trích chọn sau đó gán nhãn hoặc mạng nơron nhân tạo)
  + Trích xuất mã lệnh, giá trị các tham số.
* Các cách hiểu:
  + Dùng luật
  + Dùng thống kê
  + Lai giữa luật và thống kê

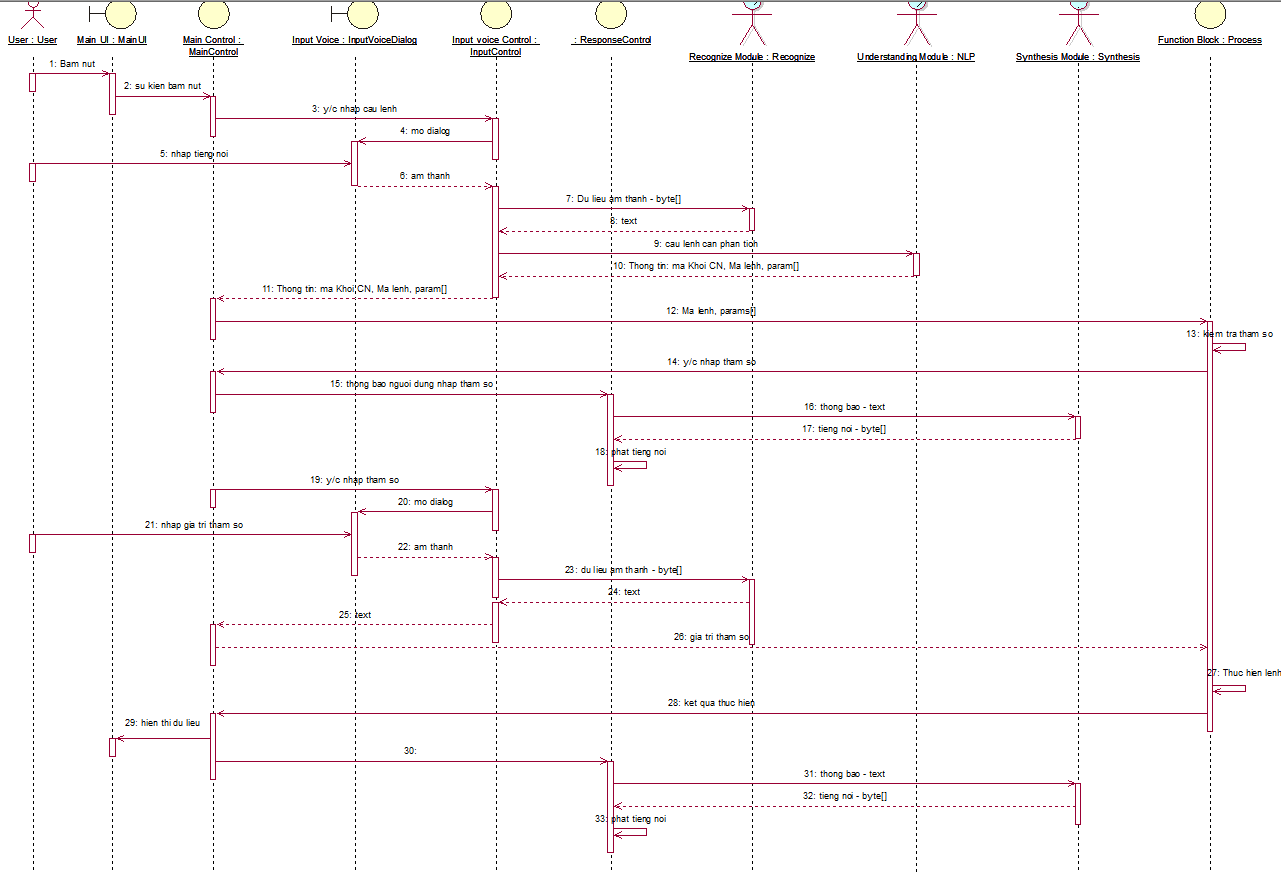
1. Khối thực thi lệnh:

* Đầu vào: kết quả từ khối hiểu.
* Đầu ra:thực thi câu lệnh cùng giá trị các tham số đưa vào, trả lời hoặc hiển thị kết quả ra màn hình cho người dung thông qua khối UI.
* Chức năng: dựa vào mã lệnh để gọi chức năng tương ứng

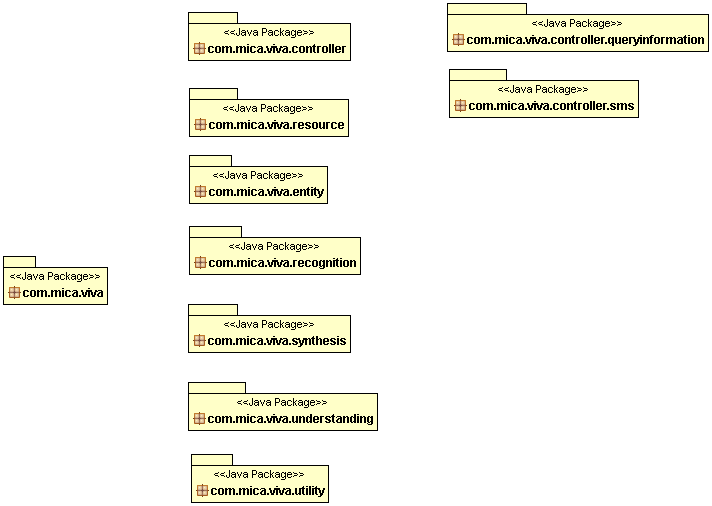
### Biểu đồ ca sử dụng



### Biểu đồ hoạt động



### Biểu đồ khối



## Triển khai hệ thống

### 2.3.1 Hệ thống nhận dạng tiếng nói

a. Chức năng hệ thống

Thu âm giọng nói, nhận dạng, trả về text làm đầu vào của khối hiểu ngôn ngữ.

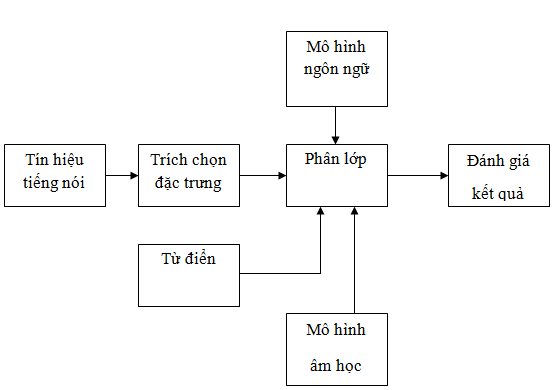
Hệ thống nhận dạng hồm 2 phần chính :

* Ghi âm và tiền xử lý tín hiệu trên SmartPhone, gửi lên Server
* Thực hiện nhận dạng trên Server đưa Text

b. Sơ đồ khối hệ thống nhận dạng tiếng nói

Một hệ thống nhận dạng nói chung thường bao gồm 2 phần, phần huấn luyện ( training phase ) và phần nhận dạng ( recognition phase ). Huấn luyện là quá trình học những mẫu chuẩn được cung cấp bởi những tiếng khác nhau ( từ hoặc âm vị ( phone )), để từ đó hình thành bộ từ vựng của hệ thống. “Nhận dạng” hay còn gọi là decoding là quá trình quyết định xem từ nào đã được đọc căn cứ vào bộ từ vựng đã được huấn luyện.

Tín hiệu tiếng nói sau khi được lấy mẫu được đưa qua khối xử lý tín hiệu để trích chọn đặc trưng phục vụ cho quá trình nhận dạng. Các phương pháp trích chọn đặc trưng thường được sử dụng trong nhận dạng tiếng nói là MFCC ( Mel – scale Frequency Ceptral Coeficients) và PLP (Peceptual Linear Prediction). Sau khi trích chọn đặc trưng, các vector đặc trưng được đưa vào khối nhận dạng, kết hợp với các hiểu biết về đặc trưng ngôn ngữ học và đặc trưng về âm học của tiếng nói, khối nhận dạng sẽ đưa ra kết quả nhận dạng hợp lý nhất. Bộ tham số của mô hình nhận dạng thu được trong quá trình luyện mô hình với một tập dữ liệu train có trước. Nếu tập train mà càng lớn thì khả năng nhận dạng của mô hình sẽ càng cao.

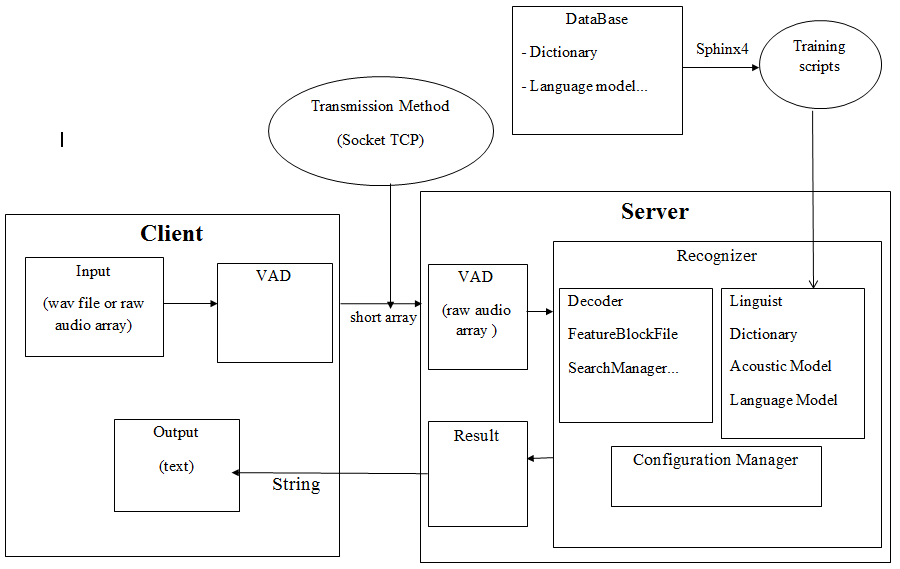


Sơ đồ khối hệ thống nhận dạng tiếng nói

Để thuận tiện cho việc kiểm tra và đánh giá kết quả từ sơ đồ trên ta sẽ chia chương trình nhận dạng tiếng nói thành 3 mô-đun riêng biệt:

* Mô-đun 1: Tiến hành thu tín hiệu tiếng nói từ MIC. Sau đó tiến hành lấy mẫu tín hiệu tiếng nói.
* Mô-đun 2: Trích chọn đặc trưng tín hiệu tiếng nói đã thu ở mô-đun 1 bằng phương pháp MFCC, đồng thời thực hiện ước lượng vector các vector đặc trưng này.
* Mô-đun 3: Sau khi trích chọn đặc trưng, các vector đặc trưng được đưa vào khối nhận dạng bao gồm bộ decoder, mô hình âm hình và mô hình ngôn ngữ.

1. Sơ đồ chung của hệ thống thực hiện



Cách thức thực hiện:

Môi trường truyền là mạng: Internet ( mạng dây, 3G ), LAN ( mạng dây, wifi )

Kiểu kết nối: Multi Client – Server

* Client gửi cho server một mảng kiểu short là tín hiệu âm thanh đã được lấy mẫu
* Server nhận dạng và trả lại kết quả dạng text

Phương thức truyền : Socket TCP

Cách thức hoạt động của cả hệ thống Multi client – server nhận dạng tiếng nói được mô tả như sau :

Sau khi thu âm ở client, client sẽ tự động ngắt thu âm và mở kết nối socket với server, số mẫu tiếng nói ban đầu được ép kiểu từ int thành short và được lưu vào một mảng short và được gửi toàn bộ lên server để tăng tốc độ gửi.

Server khi chạy luôn mở socket để lắng nghe và kết nối khi có một client muốn kết nối. Với mỗi một client đến server tạo một thread riêng để đáp ứng với client đó :

* Nhận mảng short clien gửi lên
* Chuyển mảng short thành mảng int
* VAD mảng int để bỏ đi khoảng lặng và những xung âm thanh ngắn
* Ghi thành file âm thanh định dạng wav làm đầu vào cho khối nhận dạng
* Nhận dạng file âm thanh đó và trả ra kết quả là một string
* Gửi lại string xuống client
* Đóng kết nối với client tương ứng và kết thúc thread

Vì mỗi thread trên server được đặc trưng bởi 1 socket riêng tương ứng với client nên server luôn đảm bảo việc nhận đúng và trả đúng client

Server khi khởi động sẽ tạo song song 3 bộ regconizer tương ứng là 3 bộ decoder, trong mỗi bộ regconizer đó đều có mô đun VAD và mô đun nhận dạng riếng sử dụng Sphinx 4. Ba yêu cầu đầu tiên sẽ lần lượt được phân vào 3 bộ recognizer theo thứ tự 1, 2, 3. Mỗi khi một regconizer nhận yêu cầu từ một client nó sẽ bật cờ báo bận, sau khi nhận dạng xong sẽ bật cờ báo rỗi, với một recognizer sẽ có một biến thời gian đi kèm để đo thời gian rảnh rỗi. Kể từ yêu cầu thứ 4 đến từ client, dữ liệu âm thanh nhận được từ client sẽ được phân vào bộ regconizer đang rảnh rỗi và có thời gian rảnh rỗi lớn nhất.

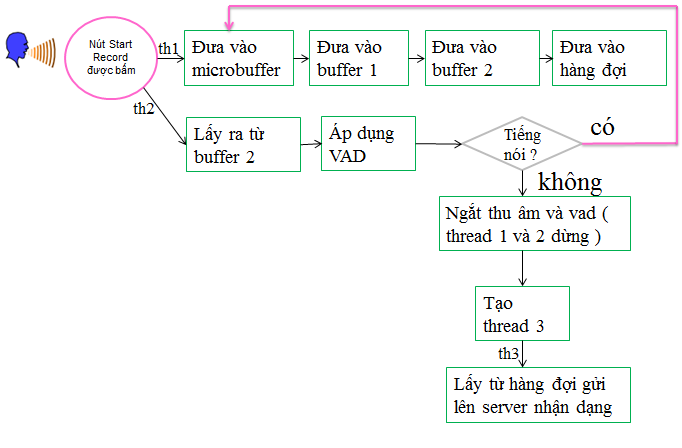
1. Hệ thống thu âm và tiền xử lý trên Client Android

Ứng dụng của chương trình này là vừa thu âm bằng microphone của thiết bị Android ( điện thoại, tab,…), vừa áp dụng thuật toán VAD với luồng âm thanh vào, khi người sử dụng không nói ( tạo ra khoảng lặng ) quá một khoảng thời gian đặt trước xác định ( từ 1,5 đến 2s ) chương trình sẽ tự động ngắt thu âm. Với luồng âm thanh nhận được cho đến lúc ngắt, chương trình sẽ gửi lên server nhận dạng và nhận kết quả là dòng text từ server nhận dạng, sau đó hiện thị lên màn hình. Người sử dụng tiếp tục bấm nút thu âm để nói câu tiếp theo và chờ đợi kết quả nhận dạng hiển thị lên trên màn hình. Ứng dụng được miêu tả trực tiếp như dưới đây:

Đầu vào: âm thanh thu từ micro của điện thoại hoặc tab

Đầu ra: mảng tiếng nói để gửi lên server

Quá trình thực hiện:

 Quá trình thực hiện trên thiết bị Android

* Khi bấm nút thu âm trên màn hình, chương trình tạo 2 thread chạy song song
  + Thread 1
    - Thu âm, đưa tín hiệu âm thanh vào vùng đệm của micro ( microbuffer ),
    - Đưa vào vùng đệm chương trình ( buffer 1 ),
    - Đưa vào buffer 2 để thread 2 có thể lấy ra và áp dụng VAD ( đây là vùng đệm sử dụng cấu trúc tránh tương tranh được java hỗ trợ ( sử dụng cấu trúc Array Blocking Queue với kích thước là 8000, chỉ đưa vào buffer 2 nếu buffer 2 chưa đầy và chỉ lấy ra được khi buffer không trống )
    - Đưa vào hàng đợi để khi ngắt thu âm sẽ gửi lên server nhận dạng
  + Thread 2
    - Lấy ra từ buffer 2 mỗi lần 8000 mẫu
    - Áp dụng thuật toán VAD để tính toán xem ngắt hay không ngắt thu âm
* Khi ngắt thu âm ( tự ngắt tùy thuộc vào kết quả của thread 2 ) chương trình sẽ lấy số mẫu từ hàng đợi lưu sang một mảng kiểu short và gửi lên server nhận dạng sau đó đợi kết quả từ server nhận dạng.
* Khi nhận được kết quả từ server nhận dạng chương trình sẽ hiển thị kết quả trên giao diện của main thread UI
* Sau khi hiển thị kết quả, nút ghi âm lại được enable. Người dùng tiếp tục bấm nút ghi âm để thực hiện lần tiếp theo: nói và chờ đợi kết quả nhận dạng

1. Phần xử lý nhận dạng trên Server(sẽ tham khảo của Hiển-Tùng thêm để sửa lại)

Ứng dụng của chương trình này là đưa vào trước khối nhận dạng trên server một mô đun độc lập để thực hiện VAD giúp bỏ đi khoảng lặng và những xung âm thanh không phải là tiếng nói để kết quả nhận dạng chính xác hơn và tốc độ nhận dạng nhanh hơn. Dưới đây là những đặc trưng của ứng dụng này:

* Mô đun này có thể có hoặc không tùy thuộc vào yêu cầu hay người lập trình
* Độc lập với Sphinx 4 trên server
* Đứng trước module nhận dạng Sphinx 4
* Đầu vào, đầu ra
  + Đầu vào : file wav âm thanh ban đầu
  + Đầu ra: file wav chỉ còn những đoạn tiếng nói được nối liền lại với nhau
  + Thực hiện: Lấy số mẫu từ file wav ban đầu và đưa vào một mảng, áp dụng thuật toán VAD với mảng đó và ghi lại thành file wav mới.

### Hệ thống hiểu ngôn ngữ nói

#### 2.3.2.1 Chức năng

* Đầu vào :

Dữ liệu văn bản text là đầu ra từ hệ thống nhận dạng. Đây là dữ liệu text thuần, chữ viết thường, không dấu chấm phẩy, chưa qua một bước xử lý nào.

* Đầu ra :
* Tên các chức năng yêu cầu từ người dùng có thể hiểu được
* Các tham số kèm theo
* Các thông tin ngoài không thực hiện được(others)
* Chú ý : Có một số trường hợp text nhận dạng không chính xác. Có thể nhận dạng đúng được một số từ, cụm từ liên quan, hệ thống hiểu có thể làm hiểu để ra kết quả như mong đợi.

#### 2.3.2.2 Miền bài toán, phạm vi dữ liệu vào ra và chuẩn quy ước

* Ở đây hệ thống VIVA đang triển khai thực hiện 4 chức năng chính:
* Nhắn tin đi
* Đọc và trả lời tin nhắn tới
* Truy vấn thông tin thời tiết
* Thông tin tỷ giá vàng, tỷ giá ngoại tệ

Vì vậy, hầu hết các câu nói cần hiểu đều nằm trong phạm vi 4 thông tin liên quan trên. Ngoài ra còn một số câu không liên quan thì hệ thống sẽ không thể hiểu được.

* Chuẩn quy ước đầu ra:

Đầu ra của hệ thống hiểu phải gồm đầy đủ các thông tin cần thiết để có thể đưa vào thực thi và trả lời.

Quy ước đầu ra của hệ thống là 1 function:

**public** **class** Function {

**private** String moduleCode = **null**;

**private** Parameter[] paramaters;

**int** number\_of\_paramaters;

}

Gồm 2 tham số chính:

* moduleCode : Đây là thông tin về chức năng sẽ phải thực hiện, có thể nói là chủ đề của câu nói từ người dùng. Các giá trị :
  + SMS\_MESSAGE : Thực hiện chức năng nhắn tin
  + WEATHER : Thực hiện chức năng truy vấn thông tin thời tiết
  + GOLD : Chức năng thông tin giá vàng
  + EXCHANGE : Chức năng thông tin tỷ giá ngoại tệ.
* Paramaters : Đây là mảng các tham số parameter

**public** **class** Parameter {

**private** String key;

**private** String value;

}

Các trường hợp :

* + WEATHER : có thể không có, gồm 1 trong 2, hoặc cả 2 tham số PLACE và TIME

key có các giá trị là PLACE và TIME

value là giá trị của tên tham số key tương ứng.

VD : parameters[] gồm 1 tham số.

{( key = “TIME” ; value = “ ngày mai ” )} hay { ( key = “PLACE” ; value = “ hải phòng ” ) }

parameters[] gồm 2 tham số.

{( key = “TIME” ; value = “thứ hai ” ) ,( key = “PLACE” ; value = “ hà nội” ) }

* + SMS\_MESSAGE: parameters[] gồm 2 tham số RECEIVER và CONTENT

key có các giá trị là RECEIVER và CONTENT

value là giá trị của tên tham số key tương ứng.

VD: parameters[] gồm 2 tham số.

{( key = “” ; value = “ ” ) ,( key = “” ; value = “ ” ) } ;

* + GOLD : Không tham số
  + EXCHANGE : không tham số

2.3.2.3 Thực hiện hiểu sử dụng automat hữu hạn

Để thực hiện quá trình hiểu các thông số như trên, chúng tôi sử dụng automat hữu hạn để biểu diễn và hiểu, lấy thông số, xác định các chức năng.

*Lý thuyết Automat hữu hạn sẽ được trình bày ở chương sau phần 3.1.*

Ta xác định các thành phần của automat :

* Tập các kí hiệu nhập :

Các kí hiệu nhập ở đây không phải là kí tự. Mà ta chuẩn hóa và quy ước, và xác định được trong bài toán là các kí hiệu nhập sau :

**public** **enum** StageType {

*ToContact*, *Tocontent*, *ToMessage*, *ToWeather*, *Time*, *Place*, *ContactName*, *End, Gold, Exchange* , *Other*

*}*

Cách xác định các kí hiệu nhập này được trình bày chi tiết hơn trong phần dưới đây.

* Tập các trạng thái :

**public** **enum** StageType {

*StartStage*, *Message*, *WaitContentNoContact*, *WaitContactHadContent*, *HadContentNoContact*,

*WaitContactNoContent*, *HadContactNoContent*, *WaitContentHadContact*, *CompletedMessage*,

*Weather*, *HadTime*, *HadPlace*, *WeatherHadTime*, *WeatherHadPlace*, *CompletedWeather*, *HadTimeAndPlace, Gold, Exchange*

}

Ý nghĩa các trạng thái :

* *StartStage:*

Trạng thái bắt đầu máy trạng thái

* *Message :*

Trạng thái khi đã xác định được chức năng nhắn tin

* *WaitContentNoContact*

Trạng thái đợi nội dung tin nhắn, khi xác định được đó là phần nội dung tin.

Chưa có thông tin người nhận

* *WaitContactHadContent*

Trạng thái đợi thông tin người nhận (tên hoặc số điện thoại) và đã có nội dung tin nhắn

* *HadContentNoContact*

Trạng thái có nội dung tin nhắn và chưa có thông tin người nhận

* *WaitContactNoContent*

Trạng thái đợi xác định thông tin người nhận mà chưa có nội dung tin

* *HadContactNoContent*

Trạng thái có người nhận và chưa có nội dung tin

* *WaitContentHadContact*

Trạng thái có người nhận và đang đợi nội dung tin sau khi xác định đó là nội dung tin

* *CompletedMessage*

Trạng thái đẩy đủ thông tin của chức năng nhắn tin

* *Weather*

Trạng thái xác định được đó là chức năng hỏi thông tin thời tiết

* *HadTime*

Trạng thái có thông tin thời gian

* *HadPlace*

Trạng thái có thông tin địa điểm

* *WeatherHadTime*

Trạng thái xác định được chức năng thời tiết và thông tin thời gian

* *WeatherHadPlace*

Trạng thái xác định được chức năng thời tiết và thông tin địa điểm

* *HadTimeAndPlace*

Trạng thái xác định được 2 thông tin thời gian và địa điểm, nhưng chưa xác định được đó có phải là chức năng thời tiết hay không

* *CompletedWeather*

Trạng thái đầy đủ thông tin cho chức năng hỏi thông tin thời tiết.

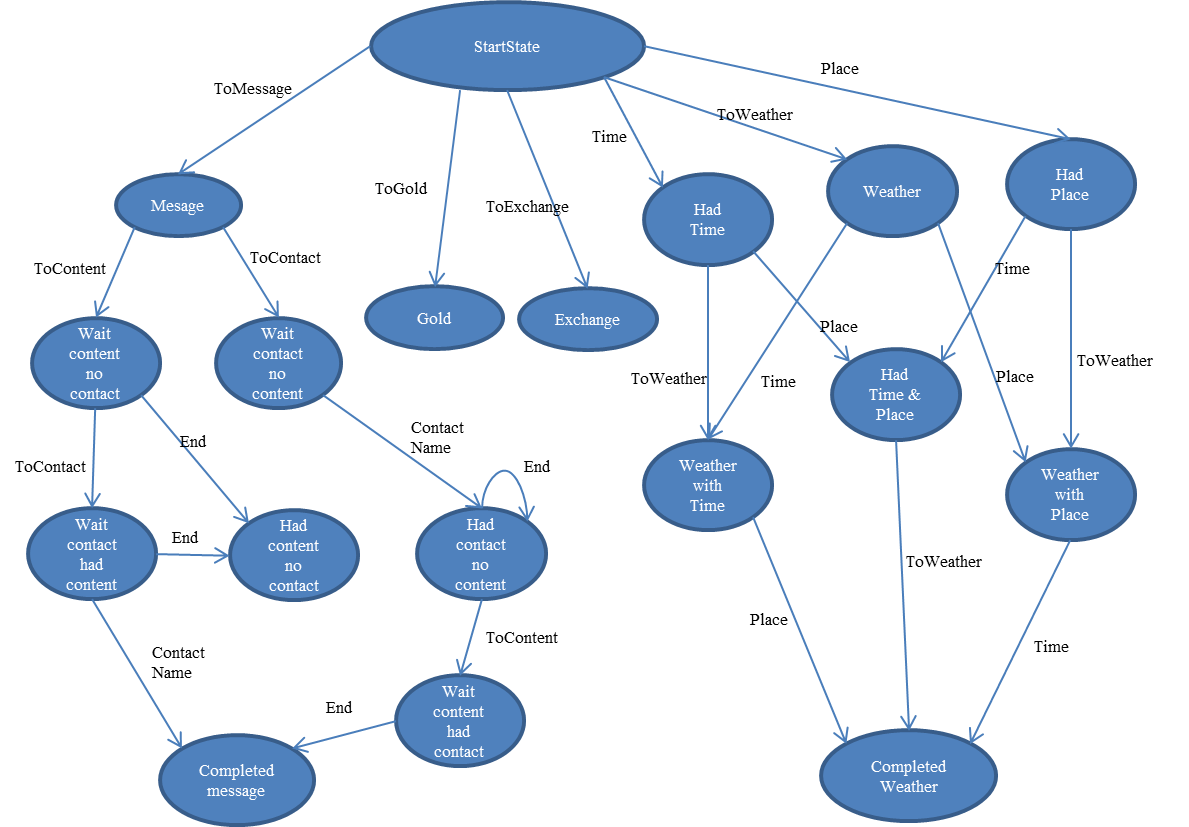
* *Gold:*

Trạng thái xác định được là truy vấn giá vàng

* *Exchange*

Trạng thái xác định được là truy vấn ngoại tệ

Sơ đồ chuyển trạng thái :



#### 2.3.2.4 Các bước thực hiện trong hệ thống

Quá trình hiểu trong hệ thống gồm các bước sau:

* Bước 1 : Tách từ

Sử dụng công cụ có sẵn là bộ tách từ  “*vnTokenizer 4.1.1c (04/08/2010)*” của dự án VLSP.

Tài nguyên (free) : <http://vlsp.vietlp.org:8080/demo/?page=resources>

* Bước 2 : Gán nhãn keyword cho từ

Xác định loại liên quan của từ được tách ra.

Ví dụ: nhắn, nhắn tin 🡪 tomessage ; thời tiết, trời 🡪toweather.

Đây chính là xác định các “ký tự nhập” của automat.

* Bước 3 : Thực hiện dịch automat

Kết quả chạy theo là các tham số Function xác định được.

* Bước 4: Xuất kết quả ra, trả về hàm điều khiển ở hệ thống.

Trong hệ thống hiện tại, gửi về client để xử lý.

#### 2.3.2.4 Đánh giá chất lượng kết quả hệ thống hiểu và đề xuất

* Trong hệ thống demo VIVA 1.0:

Chúng tôi xây dựng 100 câu thường sử dụng, với một số câu nói với nội dung phức tạp, độ chính xác lấy thông tin đạt khoảng 95%.

Các trường hợp hệ thống vẫn chưa giải quyết được là các trường hợp câu nhập nhằng.

VD :

“Gửi tin nhắn tới bảo bảo rằng đang làm gì đấy”

“cho xem thông tin thời tiết gửi tin nhắn này tới hiếu”

“thông tin giá vàng gửi thông tin này tới thầy Đạt”

….

Và còn một số trường hợp khác, thậm trí khó hiểu ngay cả đối với con người.

* Đề xuất

Sử dụng phân tích cú pháp và gán nhãn từ loại và ngữ nghĩa cho từ để nâng cao chất lượng hệ thống.

### Hệ thống tổng hợp tiếng nói

### Module thu thập thông tin

### Module nhắn tin

### Module xử lí trung tâm

### Các dữ liệu cần chuẩn bị

* Dữ liệu thu âm phục vụ nhận dạng

Để thực hiện nhận dạng giọng nói, cần có các dữ liệu mẫu.

Nhóm thực hiện thu âm 50 giờ, với khoảng 2000 câu khác nhau. Khoảng hơn 1000 câu liên quan trực tiếp đến chức năng hệ thống.

Thu âm thực hiện trong 2 tuần, với hơn 20 người khác nhau cả nam và nữ.

Dữ liệu được thu âm là âm thanh tần số 16khz, được mã hóa dạng 16bit

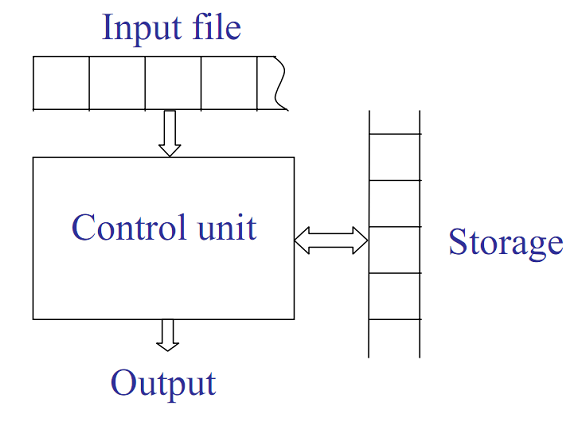
## Kết luận chương

# Chương 3 Xây dựng hệ thống hiểu ngôn ngữ tự nhiên

## 3.1 Lý thuyết automat

### 3.1.1 Định nghĩa

Automat dịch là máy tự động. Là thiết bị có thể thực hiện công việc mà không cần sự can thiệp của con người. Nó hoạt động dựa trên một số nguyên tắc và dựa vào các nguyên tắc này con người lập trình nó hoạt động theo ý muốn của mình. Đây là một mô hình trừu tượng của máy tính số bao gồm các thành phần sau :



* Thiết bị đầu vào (input file) : là nơi mà các chuỗi nhập (input string) được ghi lên và được automat đọc nhưng không thay đổi được nội dung. Nó được chia thành các ô, mỗi ô giữ một ký hiệu.
* Cơ cấu nhập(input mechanism): là bộ phận có thể đọc input file từ trái sang phải, một ký tự tại một thời điểm. Nó cũng có thể dò tìm được điểm kết thúc của chuỗi nhập
* Bộ nhớ tạm(temporary storage) : là thiết bị gồm một số không giới hạn ô nhớ(cell), mỗi ô có thể giữ 1 ký hiệu từ một bảng chữ cái (không cần giống bảng chữ cái nhập). Automat có thể đọc và thay đổi nội dung của các ô nhớ này.
* Đơn vị điều kiển (Control Unit): mỗi automat có một đơn vị điều khiển, cái mà có thể ở trong một trạng thái bất kỳ trong một số hữu hạn các trạng thái nội, và có thể chuyển đổi trạng thái trong một kiểu được định nghĩa sẵn có nào đó.

Một số khái niệm:

* Trạng thái nội: là một trạng thái của đơn vị điều khiển mà nó có thể đạt được
* Trạng thái kế: Là một trạng thái nội của đơn vị điều khiển mà nó sẽ đạt tới ở thời điểm kế tiếp
* Hàm chuyển trạng thái: là hàm đưa ra trạng thái kế của automat dựa trên trạng thái hiện hang, kí hiệu hiện thành được quét và thông tin hiện hành trong bộ nhớ tạm.

Hoạt động của automat:

* Tại một thời điểm bất kỳ đã cho, đơn vị điều khiển đang ở trong một **trạng thái nội**(internal state) nào đó, và cơ cấu nhập là **đang quét**(scanning) một ký hiệu củ thể nào đó trên Input file
* **Trạng thái nội** của đơn vị điều khiển tại thời điểm kế tiếp được xác định bởi trạng thái kế (next state) hay bởi hàm chuyển trạng thái.
* Trong suốt quá trình chuyển trạng thái từ khoảng thời gian này tới khoảng thời gian kế, kết quả có thể được sinh ra và lưu trong bộ nhớ lưu trữ có thể thay đổi.

### 3.1.2 Phân loại và ứng dụng của Automat.

* Dựa vào kết quả xuất ra của automat :
  + Accepter : là automat mà đáp ứng ở ngõ ra của nó được giới hạn trong hai trạng thái đơn giả YES hay NO. YES tương ứng với việc chấp nhận chuỗi, NO tương ứng với việc từ chối, không chấp nhận chuỗi.
  + Transducer : là automat tổng quát hơn, có khả năng sinh ra các chuỗi ký tự ở đầu ra. Máy tính số là một transducer điển hình.
* Một vài ứng dụng:
  + Cung cấp kiến thức neenftangr cho việc xây dựng các ngôn ngữ lập trình, chương trình dịch
  + Ứng dụng vào các lĩnh vực xử lý chuỗi:
    - Các chức năng tìm kiếm, thay thế trong các trình soạn thảo
    - Sửa lỗi chính tả, chú thích từ loại,…
    - Ứng dụng vào lĩnh vực thiết kế số

### 3.1.3 Otomat hữu hạn đơn định –DFA (Deterministic Finite Automata)

- Otomat hữu hạn đơn định gồm một tập hữu hạn các trạng thái và một tập các phép chuyển trạng thái tới trạng thái khác trên các ký hiệu nhập (input symbols) được chọn từ một bộ chữ cái ∑ nào đó. Mỗi ký hiệu nhập có đúng một phép chuyển khỏi mỗi trạng thái. Nó cũng có thể chuyển về chính nó. Có trạng thái bắt đầu và một số trạng thái được thiết kế như trạng thái kết thúc hay trạng thái chấp nhận.

Otomat hữu hạn đơn định gồm 5 thành phần

* Q : là tập hữu hạn các trạng thái
* ∑ : là bộ chữ cái nhập hữu hạn
* δ : Là hàm chuyển ánh xạ từ Q x ∑ 🡪 Q, tức là δ(q,a) là một trạng thái được cho phép bởi phép chuyển từ trạng thái q trên ký hiệu nhập a
* q0 là trạng thái bắt đầu
* F là tập các trạng thái kết thúc

Hoạt động của một DFA:

* Tại thời điểm khởi đầu, nó được giả thiết ở trong **trạng thái khởi đầu q0** với cơ cấu nhập đang ở ký hiệu đầu tiên bên trái của chuỗi nhập
* Mỗi lần di chuyển, cơ cấu nhập tiến về phía phải 1 ký hiệu và lấy ra
* Khi gặp ký hiệu kết thúc, chuỗi được chấp nhận nếu automat đang ở vào 1 trong **các trạng thái chấp nhận được** của nó. Ngược lại thì chuỗi bị từ chối.

Để biểu diễn trực quan cho DFA, dùng đồ thị chuyển trạng thái:

* Các đỉnh biểu diễn trạng thái
* Các cạnh biểu diễn các chuyển trạng thái
* Các nhãn trên các đỉnh là tên các trạng thái
* Các nhãn trên các cạnh là giá trị hiện tại của ký hiệu nhập
* Trạng thái khởi đầu sẽ được biết rằng một mũi tên đi vào không mang nhãn mà không xuất phát từ đỉnh nào.
* Các trạng thái kể thúc được vẽ bằng vòng tròn đôi.

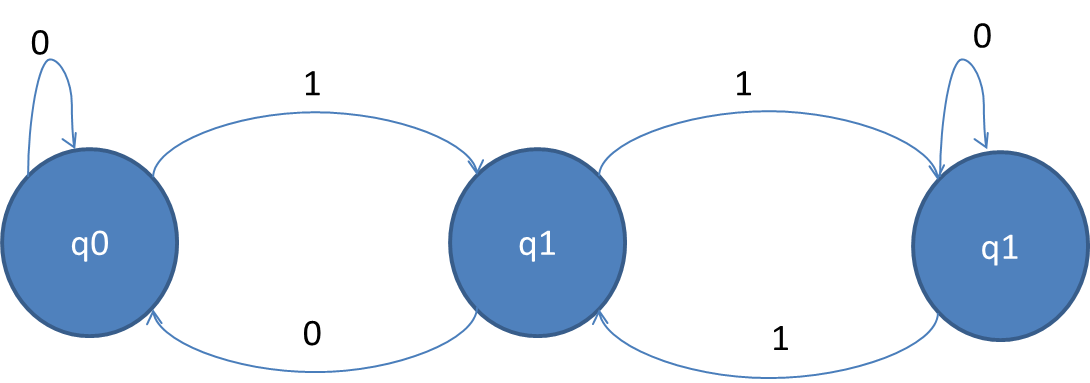
Ví dụ :

Một DFA : M = (Q, ∑, δ, q0, F)

Q = {q0,q1,q2}, ∑ = {0,1}, F{q1}, δ được cho bởi bảng sau :

|  |  |  |
| --- | --- | --- |
| Trạng thái | Kí hiệu nhập | δ |
| q0 | 0 | q0 |
| q0 | 1 | q1 |
| q1 | 0 | q0 |
| q1 | 1 | q2 |
| q2 | 0 | q2 |
| q2 | 1 | q1 |

Ta có đồ thị chuyển trạng thái tương ứng:



Trong hệ thống VIVA, chúng ta sử dụng DFA cho hệ thống hiểu.

### 3.1.4 Otomat hữu hạn không đơn định – NFA (Nondeterministic Finite Automata)

Otomat hữu hạn không đơn định là automat mà tại mỗi thời điểm có một vài khả năng có thể lựa chọn di chuyển. Nó gồm một tập hữu hạn các trạng thái và một tập các phép chuyển trạng thái tới trạng thái khác trên các ký hiệu nhập (input symbols) được chọn từ một bộ chữ cái ∑ nào đó. Mỗi ký hiệu nhập có không, một hoặc nhiều phép chuyển trạng thái khác nhau.

Otomat hữu hạn không đơn định (NFA) cũng gồm 5 thành phần giống như DFA. Trong đó Q, ∑, q0, F được định nghĩa giống DFA còn hàm chuyển trạng thái δ được định nghĩa là:

δ : Q x (∑{}) 🡪 2Q

* Miền giá trị của δ là tập 2Q, vì vậy nó không còn là một phần tử đơn của Q mà là tập con của Q. Ví dụ : δ(q0, a) = { q1, q2}. Tập con này có thể là rỗng 🡪 cấu hình chết.
*  được coi như đối số thứ 2 của δ nên NFA có thể thực hiện dịch chuyển mà không cần có ký tự nhập
* NFA cũng được biểu diễn bằng đồ thị chuyển trạng thái

## 3.2 Tách từ trong xử lý ngôn ngữ tự nhiên

Tách từ là một quá trình xử lý nhằm mục đích xác định ranh giới của các từ trong câu văn, cũng có thể hiểu đơn giản rằng tách từ là quá trình xác định các từ đơn, từ ghép… có trong câu. Đối với xử lý ngôn ngữ tự nhiên, để có thể xác định cấu trúc ngữ pháp của câu, xác định từ loại của một từ trong câu, yêu cầu nhất thiết đặt ra là phải xác định được đâu là từ trong câu. Vấn đề này tưởng chừng đơn giản với con người nhưng đối với máy tính, đây là bài toán rất khó giải quyết.  
Chính vì lý do đó tách từ được xem là bước xử lý quan trọng đối với các hệ thống Xử Lý Ngôn Ngữ Tự Nhiên, đặc biệt là đối với các ngôn ngữ thuộc vùng Đông Á theo loại hình ngôn ngữ đơn lập. Ví dụ: tiếng Trung Quốc, tiếng Nhật, tiếng Thái, và tiếng Việt. Với các ngôn ngữ thuộc loại hình này, ranh giới từ không chỉ đơn giản là những khoảng trắng như trong các ngôn ngữ thuộc loại hình hòa kết như tiếng Anh…, mà có sự liên hệ chặt chẽ giữa các tiếng với nhau, một từ có thể cấu tạo bởi một hoặc nhiều tiếng. Vì vậy đối với các ngôn ngữ thuộc vùng Đông Á, vấn đề của bài toán tách từ là khử được sự nhập nhằng trong ranh giới từ.  
  
3.2.1 Các hướng tiếp cận giải quyết

* Có 3 phương pháp tiếp cận chính trong bài toán tách từ :
  + Tiếp cận dựa vào từ điển cố định.
  + Tiếp cận dựa vào thống kê thuần túy
  + Tiếp cận dựa trên cả hai phương pháp trên.
* Các phương pháp được sử dụng
  + So khớp từ dài nhất (Longest Matching)
  + So khớp cực đại (Maximum Matching)
  + Mô hình Markov ẩn (Hidden Markov Models- HMM)
  + Học dựa trên sự cải biến (Transformation-based Learning – TBL)
  + Chuyển đổi trạng thái trọng số hữu hạn (Weighted Finite State Transducer – WFST)
  + Độ hỗn loạn cực đại (Maximum Entropy – ME)
  + Máy học sử dụng vectơ hỗ trợ (Support Vector Machines)
  + Trường xác xuất có điều kiện (CRFs)
* Tài nguyên hỗ trợ:
  + Từ điển tiếng Việt
  + Ngữ liệu tiếng Việt được tách từ hỗ trợ quá trình huấn luyện

### 3.2.2 Ứng dụng bài toán tách từ

Bài toán tách từ là bài toán cơ bản đầu tiên trong các bài toán đặt ra cho xử lý ngôn ngữ tự nhiên:

* Trong phân tích hình thái (morphological analysis)
  + Phân tích phụ tố
  + Nhận diện tên riêng
  + Nhận diện ranh giới ngữ
* Trong phân tích ngữ pháp (PARSER)
  + Gán nhãn từ loại
  + Gán nhãn ranh giới ngữ
  + Gán nhãn quan hệ cú pháp
* Trong xử lý văn bản
  + Kiểm lỗi chính tả
  + Kiểm lỗi văn phạm
  + Phân loại văn bản
  + Tóm tắt văn bản
  + Hiểu văn bản
  + Khai thác văn bản

### 3.2.3 Một số vấn đề mở rộng

Trên thế giới, các nước như Trung Quốc, Nhật bản, các công cụ tách từ đạt được độ chính xác tới khoảng 99%.

Đối với tiếng Việt, có 1 công cụ tách từ được phát triển trong đề tài VLSP(<http://vlsp.vietlp.org:8080/demo/?page=home>) đạt được độ chính xác 97%.

Ngoài ra, có 1 vấn đề nảy sinh trong khi tách từ là việc xuất hiện các từ mới - các từ không được định nghĩa trong từ điển. Đây là một vấn đề không thể bỏ qua khi ngôn ngữ là luôn luôn thay đổi và sinh ra các từ mới, trong khi từ điển (dành cho xử lý ngôn ngữ tự nhiên) không thể cập nhật hết được.

## 3.3 Gán nhãn từ loại

### 3.3.1 Vấn đề gán nhãn từ loại tiếng Việt

Một trong các vấn đề nền tảng của ngôn ngữ tự nhiên là việc phân loại các từ thành các lớp từ loại dựa theo thực tiễn hoạt động ngôn ngữ. Mỗi từ loại tương ứng với một hình thái và giữ một vai trò ngữ pháp nhất định. Các công cụ chú thích từ loại hay công cụ gán từ loại cho từ có thể thay đổi tuỳ theo quan niệm về đơn vị từ vựng và thông tin ngôn ngữ cần khai thác trong các ứng dụng cụ thể. Mỗi từ trong một ngôn ngữ nói chung có thể gắn với nhiều từ loại, và việc giải thích đúng nghĩa một từ phụ thuộc vào việc nó được xác định đúng từ loại hay không. Công việc gán nhãn từ loại cho một văn bản là xác định từ loại của mỗi từ trong phạm vi văn bản đó. Khi hệ thống văn bản đã được gán nhãn, hay nói cách khác là đã được chú thích từ loại thì nó sẽ được ứng dụng rộng rãi trong các hệ thống tìm kiếm thông tin, trong các ứng dụng tổng hợp tiếng nói, các hệ thống nhận dạng tiếng nói cũng như trong các hệ thống dịch máy.

Đối với các văn bản Việt ngữ, việc gán nhãn từ loại có nhiều khó khăn, đặc biệt là bản thân việc phân loại từ tiếng Việt cho đến nay vẫn là một vấn đề còn nhiều tranh cãi, chưa có một chuẩn mực thống nhất.

Các bộ gán nhãn thường được đánh giá bằng độ chính xác của kết quả: [số từ được gán nhãn đúng] / [tổng số từ trong văn bản]. Các bộ gán nhãn tốt nhất hiện nay có độ chính xác đạt tới 93%.

### 3.3.2 Bảng nhãn từ loại Tiếng Việt

Dưới đây là bảng nhãn từ loại tiếng Việt thường dùng và đang được ứng dụng trong nghiên cứu xử lý ngôn ngữ tự nhiên ở Việt Nam.

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Nhãn | Tên | Ví dụ |
| 1 | N | Danh từ | tiếng, nước, thủ đô, nhân dân, đồ  đạc, cây cối, chim muông |
| 2 | Np | Danh từ riêng | Nguyễn Du, Việt Nam, Hải Phòng,  Trường Đại học Bách khoa Hà Nội, Mộc tinh, Hoả tinh, Phật, Đạo Phật |
| 3 | Nc | Danh từ chỉ loại | con, cái, đứa, bức |
| 4 | Nu | Danh từ đơn vị | mét, cân, giờ, nắm, nhúm, hào, xu,  đồng |
| 5 | V | Động từ | ngủ, ngồi, cười; đọc, viết, đá, đặt;  thích, yêu, ghét, giống, muốn |
| 6 | A | Tính từ | tốt, xấu, đẹp; cao, thấp, rộng |
| 7 | P | Đại từ | tôi, chúng tôi, hắn, nó, y, đại nhân, đại  ca, huynh, đệ |
| 8 | L | Định từ | mỗi, từng, mọi, cái; các, những, mấy |
| 9 | M | Số từ | một, mười, mười ba; dăm, vài, mươi;  nửa, rưỡi |
| 10 | R | Phó từ | đã, sẽ, đang, vừa, mới, từng, xong,  rồi; rất, hơi, khí, quá |
| 11 | E | Giới từ (kết từ chính phụ) | trên, dưới, trong, ngoài; của, trừ,  ngoài, khỏi, ở |
| 12 | C | Liên từ (kết từ đẳng lập) | và, với, cùng, vì vậy, tuy nhiên,  ngược lại |
| 13 | I | Thán từ | ôi, chao, a ha |
| 14 | T | Trợ từ, tình thái từ (tiểu từ) | à, a, á, ạ, ấy, chắc, chăng, cho, chứ |
| 15 | B | Từ tiếng nước ngoài (hay từ  vay mượn) | Internet, email, video, chat |
| 16 | Y | Từ viết tắt | OPEC, WTO, HIV |
| 17 | S | Yếu tố cấu tạo từ | bất, vô, gia, đa |
| 18 | X | Các từ không phân loại  được |  |

### 3.3.3 Các bước gán nhãn từ loại

Việc gán nhãn từ loại thường gồm 3 bước chính như sau:

* Bước 1 : Phân tách xâu kí tự thành chuỗi các từ.

Giai đoạn này có thể đơn giản hay phức tạp tuỳ theo ngôn ngữ và quan niệm về đơn vị từ vựng. Chẳng hạn đối với tiếng Anh hay tiếng Pháp, việc phân tách từ phần lớn là dựa vào các kí hiệu trắng. Tuy nhiên vẫn có những từ ghép hay những cụm từ công cụ gây tranh cãi về cách xử lí. Trong khi đó với tiếng Việt thì dấu trắng càng không phải là dấu hiệu để xác định ranh giới các đơn vị từ vựng do tần số xuất hiện từ ghép rất cao.

* Gán nhãn tiên nghiệm.

Tìm cho mỗi từ tập tất cả các nhãn từ loại mà nó có thể có. Tập nhãn này có thể thu được từ cơ sở dữ liệu từ điển hoặc kho văn bản đã gán nhãn bằng tay. Đối với một từ mới chưa xuất hiện trong cơ sở ngữ liệu thì có thể dùng một nhãn ngầm định hoặc gắn cho nó tập tất cả các nhãn. Trong các ngôn ngữ biến đổi hình thái người ta cũng dựa vào hình thái từ để đoán nhận lớp từ loại tương ứng của từ đang xét.

* Quyết định kết quả gán nhãn,

Đây là giai đoạn loại bỏ nhập nhằng, tức là lựa chọn cho mỗi từ một nhãn phù hợp nhất với ngữ cảnh trong tập nhãn tiên nghiệm. Có nhiều phương pháp để thực hiện việc này, trong đó người ta phân biệt chủ yếu các phương pháp dựa vào quy tắc ngữ pháp và các phương pháp xác suất

### 3.3.4 Các trường hợp nhập nhằng

* Nhập nhằng động từ - kết từ chính phụ:

Trong câu sau “vào” là kết từ chính phụ (E) chứ không phải động từ.

*Vào/E* dịp/N nghỉ/V hè/N rất/R đông/A học sinh/N phổ thông/N theo/V học/V

./.

Hay “lên” trong ví dụ sau:

Rmah Rô/Np khoác/V gùi/N lên/E vai/N ./.

* Nhập nhằng động từ - trợ từ

Bé đang tập đi/V . trông sạch quá đi/T

* Nhập nhằng động từ - danh từ

Hiện tượng nhập nhằng này khá phổ biến. Một số trường hợp là:

* Động từ chỉ hành động như “cày”, “cuốc”, “đục”, v.v. chuyển thành danh từ chỉ đồ vật tương ứng “(cái) cày”, “(cái) cuốc”, “(cái) đục”, v.v.

VD: Tôi mượn cuốc/N để cuốc/V đám đất sau nhà.

* Động từ chỉ hành động như “suy nghĩ”, “đắn đo”, v.v. chuyển thành danh từ chỉ khái niệm hay sự vật trừu tượng “(những) suy nghĩ”, “(những) đắn đo”, v.v.

Vài suy nghĩ/N về Toán học Việt Nam

Hôm nay, đọc lại hồi kí Hồi ức và suy nghĩ/N của Trần Quang Cơ trên tinh thần “ôn cố tri tân”, bạn đọc Thông Luận có thể nhận ra nhiều điều thú vị.

* Nhập nhằng động từ - tính từ

Đồng hồ này chạy/V rất chính xác. Hàng bà ấy dạo này bán không chạy/A .

* Nhập nhằng động từ - phó từ

Nhầm từ “được” là động từ trong trường hợp sau:

Mình/P không/R nuôi/V chúng nó/P được/R ./.

Từ này có độ nhập nhằng cao vì nó vừa có thể là động từ, vừa có thể là tính từ, phó từ, và trợ từ. Trong ví dụ trên “được” là phụ từ vì nó biểu thị điều vừa nói đến (động từ “nuôi”) là có khả năng thực hiện.

* Nhập nhằng danh từ - kết từ chính phụ

Nhà ấy nhiều của/N lắm !

sách của/E thư viện

* Nhập nhằng tính từ - danh từ

Từ chỉ tính chất như “khó khăn”, “gian khổ”, v.v. chuyển thành danh từ chỉ sự vật trừu tượng

“(những) khó khăn”, “(những) gian khổ”, v.v.

Tuy nhiên, sức chống đỡ trước khó khăn/N của Việt Nam đã tốt lên rất nhiều.

Tăng giá sách, khó khăn/N sẽ thêm chồng chất

* Nhập nhằng kết từ đẳng lập – phụ từ

Cháu Hiếu rất bé mà/C rất khỏe . Mẹ đã bảo mà/T !

* Các cụm danh từ dẫn xuất (hình thái dẫn xuất)

Có nhiều cụm danh từ có cấu tạo gồm động từ hoặc tính từ đi sau danh từ chỉ loại. Ví dụ như: “sự nóng hổi”, “vụ hối lộ”, v.v. Các từ này phải được gán nhãn là động từ hay tính từ.

sự/Nc nóng hổi/A; vụ/Nc hối lộ/V

Trường hợp của một số cụm từ như: “những vất vả”, “những thành công”,v.v. Thì ta gán từ loại danh từ cho “vất vả”, “thành công”, v.v.

* Các cụm từ “sau khi”, “trong khi”, v.v.

Các cụm từ này gồm hai từ, một là kết từ chính phụ (chẳng hạn “sau”), từ còn lại là danh từ thời gian (chẳng hạn “khi”).

sau/E khi/N; trong/E khi/N; trước/E khi/N

Ngoài ra còn có: “đến khi”, “tới khi”, “trước lúc”, “trong lúc”, “đến lúc”, “tới lúc”, v.v.; “trong đó” cũng có thể được xếp vào nhóm này (trong/E đó/P).

* Nhóm từ “ra”, “vào”, “lên”, “xuống”

Danh sách : ra, vào, lên, xuống, đến, tới, sang, qua, lại, về

Các từ này có thể được gán nhãn là động từ, phó từ, và kết từ chính phụ (giới từ) tùy thuộc vào ngữ cảnh:

* Khi không có vị từ là thực từ (như “đi”, “hiểu”, “béo”) đứng trước, các từ này có tư cách động từ:

Tôi/P ra/V sân/N. Tôi/P vào/V lớp học/N.

* Khi có vị từ là thực từ đứng trước, các từ này làm phó từ:

đi/V ra/R, bước/V xuống/R, đẩy/V xe/N ra/R, kéo/V xe/N lên/R

* Khi có động từ đằng trước6 và danh từ (thường là chỉ vị trí) đằng sau hoặc khi ở đầu câu, các từ này làm kết từ chính phụ:

đi/V xuống/E Hải Phòng/Np,

đi/V lên/E Kontum/Np,

bê/V bàn/N ra/E vườn/N

* Khi động từ đứng trước có mẫu NP-VP thì các từ này là động từ :

bảo/V nó/P xuống/V đây/P

* Các từ chức năng không có trong từ điển nhưng lại có trong sách ngữ pháp

Kết từ chính phụ (giới từ): “cho đến”, “cho tới”, “để mà”, “để cho”, “cùng với”

Kết từ đẳng lập: “cũng như”

* Các từ (hay tổ hợp từ) chức năng có trong từ điển nhưng lại không được phân loại
* Kết từ chính phụ: “đến nỗi”

khỏe/A đến nỗi/E có thể/V dùng/V hai/M tay/N nắm/V hai/M sừng/N trâu/N ghì/V xuống/E đất/N

* Phó từ: “đúng ra”, “lẽ ra”, “mới đây”, “vừa rồi”, “vừa qua”; “hình như”, “dường như”; “ít nhất”

đúng ra/R ,/, nó/P phải/V bị/V kỷ luật/V

Các từ này có đặc điểm là có thể xuất hiện ở đầu câu hoặc giữa chủ ngữ và vị ngữ.

* Kết từ đẳng lập: “chẳng hạn”; “ngoài ra”; “mặt khác”; “tức”, “tức là”; “nghĩa là”; “ngược lại”; “nói chung”, “nhìn chung”; “nói riêng”; “nói tóm lại”

Mặt khác/C, thông thường các sản phẩm tẩy trang đều chứa thành phần dầu.

* Một số từ khác: “do đó”, “do vậy”, “tuy vậy”, “dẫu vậy”

Dẫu vậy/C, cả hai ông và bất kỳ người nào khác đều không thể thống nhất về những phẩm chất cần có, chính xác của một ông chủ Nhà Trắng.

* Từ bổ nghĩa cho số từ

Các từ: “hơn”, “trên”, “dưới”, “gần”, “khoảng”, v.v.

hơn/A 200/M đại biểu/N, trên/A mười/M suất/N học bổng/N, khoảng/A

50/M bảng/Nu

* Trợ từ “một cách”

Nước/N xả/V Miwon/Np bỗng nhiên/R trong/A một cách/T kỳ lạ/A !/.

Trợ từ này có tác dụng nhấn mạnh. Nó luôn đứng ở trước bộ phận cần được nhấn mạnh

* Mẫu “từ .. sang ..”

Đó/P là/V hải\_trình/N lớn/A nhất/R từ/E tây/N sang/E đông/N

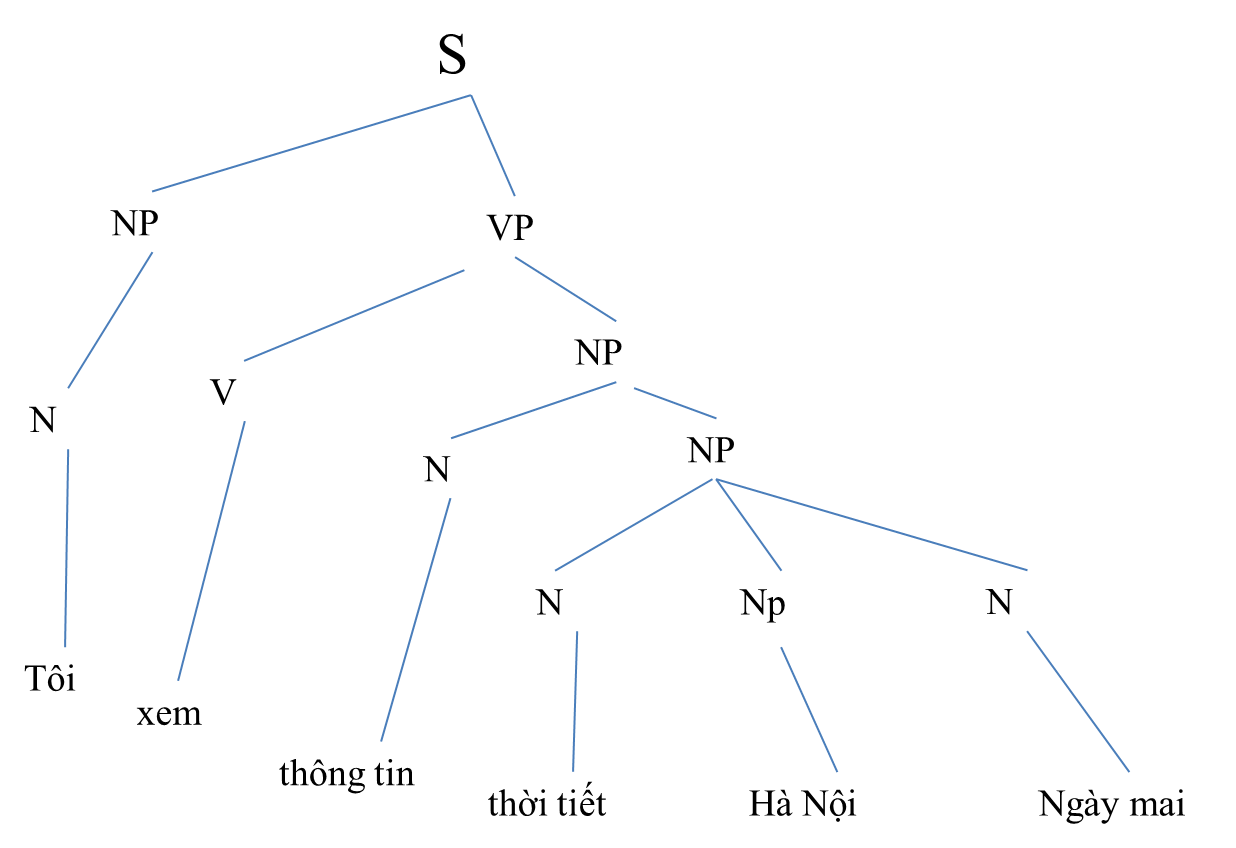
“từ" và “sang” cần được gán nhãn là E. Đây là các thành phần của cụm giới từ chỉ hướng.

## 3.4 Phân tích cú pháp trong xử lý ngôn ngữ tự nhiên

### 3.4.1 Biểu diễn cú pháp

Trong xử lý ngôn ngữ tự nhiên, để đạt được kết quả cao, việc phân tích cú pháp là thực sự rất cần thiết. Để biểu diễn cú pháp 1 câu, người ta thường biểu diễn dưới dạng cây cú pháp. Ví dụ :

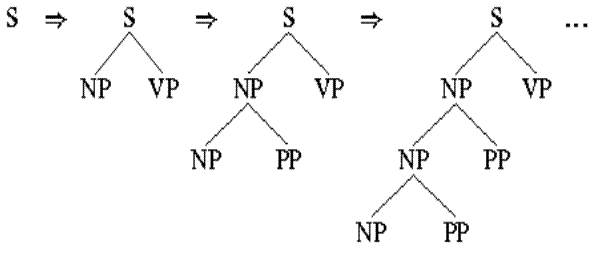
Câu “Tôi xem thời tiết Hà Nội ngày mai” được biểu diễn dưới dạng cây cú pháp:



Việc biểu diễn cú pháp cũng đang là một bài toán rất khó không chỉ với tiếng Việt mà còn với cả hầu hết các ngôn ngữ khác trên thế giới. Ta có 2 phương pháp phân tích cây cú pháp : TOP – DOWN và BOTTOM – UP

### 3.4.2 Phương pháp phân tích cú pháp từ trên xuống (Top-Down Parsing)

Trong phương pháp này, ban đầu chúng ta sẽ đi từ nút gốc của cây. Ở mỗi bước, chúng ta sẽ cố phân tích thành các nhánh con. Từ các nhánh con ta lại phân tích tiếp như vậy. Làm tới khi nào xác định được hết các nút lá.



Có 2 cách thường dùng :

* Phân tích cú pháp đệ qui lùi (Recursive Descent Parsing)

Phân tích cú pháp từ trên xuống có thể được xem như một nỗ lực tìm kiếm một dẫn xuất trái nhất cho chuỗi nhập. Nó cũng có thể xem như một nỗ lực xây dựng cây phân tích cú pháp bắt đầu từ nút gốc và phát sinh dần xuống lá. Một dạng tổng quát của kỹ thuật phân tích từ trên xuống, gọi là **phân tích cú pháp đệ quy lùi**, có thể quay lui để quét lại chuỗi nhập. Tuy nhiên, dạng này thường rất ít gặp. Lý do là với các kết cấu ngôn ngữ lập trình, chúng ta hiếm khi dùng đến nó.

* Bộ phân tích cú pháp dự đoán (Predictive Parser) (Chủ yếu sử dụng trong các chương trình dịch hơn là trong xử lý ngôn ngữ tự nhiên)

Trong nhiều trường hợp, bằng cách viết văn phạm một cách cẩn thận, loại bỏ đệ qui trái ra khỏi văn phạm rồi tạo ra yếu tố trái, chúng ta có thể thu được một văn phạm mà một bộ phân tích cú pháp đệ quy lùi phân tích được, nhưng không cần quay lui, gọi là phân tích cú pháp dự đoán.

### 3.4.3 Phương pháp phân tích cú pháp từ dưới lên (Bottom – Up Parsing)

Trong phương pháp này, ta làm ngược lại với phương pháp Top – Down.

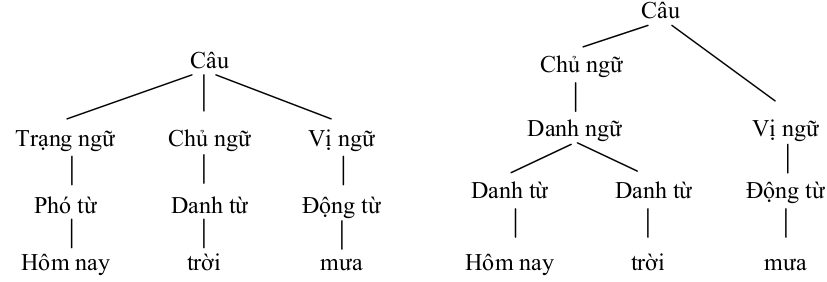
* Đầu tiên chúng ta đi tìm các nút lá.
* Cố gắng đưa chúng vào các nhánh của cây
* Xây dựng cây cú pháp tới mức hoàn chỉnh.

### 3.4.4 Một số vấn đề trong phân tích cú pháp tiếng Việt

#### 3.4.4.1 Hiện tượng nhập nhằng phân tích cú pháp tiếng Việt

Trong phân tích cú pháp tiếng Việt, hiện tượng nhập nhằng xảy ra ở nhiều mức, từ mức từ, từ loại đến mức cú pháp câu. Điều này dẫn đến một câu có thể được phân tích theo nhiều cách khác nhau, trong khi chỉ có một vài cách phân tích trong số đó đúng. Có hai loại nhập nhằng thường xảy ra :

* Nhập nhằng do câu có thể hiểu theo nhiều nghĩa khác nhau dẫn đến các cây cú pháp khác nhau. Trong trường hợp này, mỗi cách hiểu sẽ ứng với một cây cú pháp và các cây cú pháp đó đều được chấp nhận. Ví dụ câu “***Tôi nhìn thấy thầy Nam ở tầng hai***” có thể hiểu theo 2 cách. Cách thứ nhất, khi tôi nhìn thấy thầy Nam thì thầy đang ở tầng hai. Trong trường hợp này, *ở tầng hai* bổ nghĩa cho danh ngữ *thầy Nam*. Cách hiểu thứ hai, khi tôi đứng ở tầng hai thì tôi nhìn thấy thầy Nam. Trong trường hợp này, *ở tầng hai* là bổ ngữ của *tôi nhìn thấy thầy Nam*.
* Nhập nhằng khi câu chỉ có một nghĩa nhưng bộ phân tích cú pháp vẫn tạo ra nhiều cây cú pháp, trong đó chỉ có một cây đúng. Lý do của sự nhập nhằng này là quá trình phân tích cú pháp đã lược bỏ ngữ nghĩa từ/ngữ mà chỉ quan tâm đến nhãn cú pháp của chúng, dẫn đến nhiều luật cú pháp có thể áp dụng để phân tích câu. Ví dụ câu “hôm nay trời mưa” có thể phân tích ra thành 2 cây cú pháp sau:



Hoặc các trường hợp đặc biệt có các từ như “và”. Ví dụ “vấn đề trong phần trước và phần này”. Trong trường hợp này cũng sẽ sinh ra sự nhập nhằng với hai cách hiểu khác nhau :

* “vấn đề phần trước” và “phần này”
* “vấn đề phần trước” và “vấn đề phần này”.

Cách giải quyết tốt nhất cho các sự nhập nhằng này thường sử dụng các mô hình xác xuất thống kê để đánh giá sự xuất hiện các cú pháp.

#### 3.4.4.2 Việc xác định câu đúng cú pháp phụ thuộc vào các từ cụ thể cấu tạo nên câu

Ta còn gặp nhiều trường hợp khác trong tiếng Việt mà việc xác định câu đúng cú pháp hay không phụ thuộc vào từ cụ thể cấu tạo nên câu. Ví dụ “*Tôi ăn*” ít khi được chấp nhận là một câu hoàn chỉnh trong một ngữ cảnh chung. Tính hoàn chỉnh ở đây nhìn từ phía cảm nhận của người nghe, anh ta có cảm thấy thỏa mãn một lượng thông tin hay không. Trong ngữ cảnh chung “*Tôi ăn*” mang một giá trị thông tin nhỏ. Với câu này, nếu ta chỉ dựa trên các từ loại của câu và luật cú pháp câu có thể được hình thành từ một danh từ đứng trước một động từ thì câu trên hoàn toàn đúng ngữ pháp. “*Tôi đang ăn*” dễ được chấp nhận là câu hoàn chỉnh hơn vì trong một ngữ cảnh chung mệnh đề trên mang một giá trị thông tin khá lớn. Với những trường hợp nói trên, chúng ta phải dựa trên tính chất cụ thể của từ giữ vai trò chính trong câu hoặc ngữ để xác định xem câu/ngữ đó có đúng cú pháp hay không.

Trong phân tích cú pháp tiếng Việt, chúng ta còn thấy hiện tượng nhập nhằng do lược bỏ quan hệ từ. Chúng ta có thể nói *bạn tôi, con tôi* mà không nói *con chó tôi, con mèo tôi*. Trong trường hợp này, *bạn tôi, con tôi* cần được coi là các danh ngữ, trong khi *con chó tôi, con mèo tôi* cần được coi là các cụm từ sai ngữ pháp.

Qua đây ta thấy bản thân từ cũng có vai trò quan trọng trong quá trình phân tích cú pháp. Vì vậy cần xây dựng một công cụ phân tích cú pháp cho phép phân tích sâu hơn văn phạm phi ngữ cảnh kết hợp xác suất bằng cách đưa thông tin từ vựng vào văn phạm.

## 3.5 Bổ sung nhãn ngữ nghĩa cho từ.