Lời cảm ơn

Trước hết em xin chân thành cảm ơn thầy giáo Ths. Vũ Mạnh Khánh, là người đã hướng dẫn em rất nhiều trong suốt quá trình tìm hiểu nghiên cứu và hoàn thành khóa luận này từ lý thuyết đến ứng dụng. Sự hướng dẫn của các thầy đã giúp em có thêm được những hiểu biết về xử lý ngôn ngữ tự nhiên và các ứng dụng của nó.

Đồng thời em cũng xin chân thành cảm ơn các thầy cô trong bộ môn công nghệ thông tin cũng như các thầy cô trong trường đã trang bị cho em những kiến thức cơ bản cần thiết để em có thể hoàn thành tốt khóa luận này.

Em xin gửi lời cảm ơn đến các thành viên lớp CT1002, những người bạn đã luôn ở bên cạnh động viên, tạo điều kiện thuận lợi và cùng em tìm hiểu, hoàn thành tốt khóa luận.

Sau cùng, em xin gửi lời cảm ơn đến gia đình, bạn bè đã tạo mọi điều kiện để em xây dựng thành công khóa luận này.

Hải Phòng, ngày…….tháng……năm 2010

Sinh viên

Nguyễn Văn Thành

Mục lục

Article I. MỞ ĐẦU

Xử lý ngôn ngữ tự nhiên (natural language processing - NLP) là một nhánh của trí tuệ nhân tạo tập trung vào các ứng dụng trên ngôn ngữ của con người. Trong trí tuệ nhân tạo thì xử lý ngôn ngữ tự nhiên là một trong những phần khó nhất vì nó liên quan đến việc phải hiểu ý nghĩa ngôn ngữ - công cụ hoàn hảo nhất của tư duy và giao tiếp.

Xử lý ngôn ngữ chính là xử lý thông tin khi đầu vào là "dữ liệu ngôn ngữ" (dữ liệu cần biến đổi), tức dữ liệu "văn bản" hay "tiếng nói". Các dữ liệu liên quan đến ngôn ngữ viết (văn bản) và nói (tiếng nói) đang dần trở nên kiểu dữ liệu chính con người có và lưu trữ dưới dạng điện tử. Đặc điểm chính của các kiểu dữ liệu này là không có cấu trúc hoặc nửa cấu trúc và chúng không thể lưu trữ trong các khuôn dạng cố định như các bảng biểu.

Để máy tính có thể hiểu và thực thi một chương trình được viết bằng ngôn ngữ cấp cao, ta cần phải có một trình biên dịch thực hiện việc chuyển đổi chương trình đó sang chương trình ở dạng ngôn ngữ đích.

Xử lý ngôn ngữ tự nhiên là một lĩnh vực nghiên cứu nhằm giúp cho các hệ thống máy tính hiểu và xử lý được ngôn ngữ con người. Dịch máy là một trong những ứng dụng chính của xử lý ngôn ngữ tự nhiên. Mặc dù dịch máy đã được nghiên cứu và phát triển trong hơn 50 năm qua, song vẫn tồn tại nhiều vấn đề cần nghiên cứu.

Article II. Chương 1 : Giới thiệu về xử lý ngôn ngữ tự nhiên

1.1. Tổng quan

Xử lý ngôn ngữ chính là xử lý thông tin khi đầu vào là "dữ liệu ngôn ngữ" (dữ liệu cần biến đổi), tức dữ liệu "văn bản" hay "tiếng nói". Các dữ liệu liên quan đến ngôn ngữ viết (văn bản) và nói (tiếng nói) đang dần trở nên kiểu dữ liệu chính con người có và lưu trữ dưới dạng điện tử. Đặc điểm chính của các kiểu dữ liệu này là không có cấu trúc hoặc nửa cấu trúc và chúng không thể lưu trữ trong các khuôn dạng cố định như các bảng biểu. Theo đánh giá của công ty Oracle, hiện có đến 80% dữ liệu không cấu trúc trong lượng dữ liệu của loài người đang có [Oracle Text]. Với sự ra đời và phổ biến của Internet, của sách báo điện tử, của máy tính cá nhân, của viễn thông, của thiết bị âm thanh,… người người ai cũng có thể tạo ra dữ liệu văn bản hay tiếng nói. Vấn đề là làm sao ta có thể xử lý chúng, tức chuyển chúng từ các dạng ta chưa hiểu được thành các dạng ta có thể hiểu và giải thích được, tức là ta có thể tìm ra thông tin, tri thức hữu ích cho mình.

Giả sử chúng ta có các câu sau trong các tiếng nước ngoài:

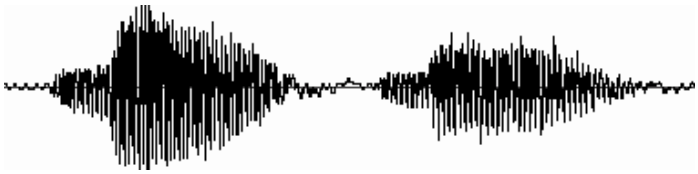
- "We meet here today to talk about Vietnamese language and speech processing."

- "Aujourd'hui nous nous réunissons ici pour discuter le traitement de langue et de parole vietnamienne."

- "Мы встрачаемся здесь сегодня, чтобы говорить о вьетнамском языке и обработке речи."

Nếu có ai đó dịch, hoặc có một chương trình máy tính dịch (biến đổi) chúng ra tiếng Việt, ta sẽ hiểu nghĩa các câu trên đều là: "Hôm nay chúng ta gặp nhau ở đây để bàn về xử lý ngôn ngữ và tiếng nói tiếng Việt.". Nếu các câu này được lưu trữ như các tệp tiếng Anh, Pháp, Nga và Việt như ta nhìn thấy ở trên, ta có các dữ liệu "văn bản". Nếu ai đó đọc các câu này, ghi âm lại, ta có thể chuyển chúng vào

máy tính dưới dạng các tệp các tín hiệu (signal) "tiếng nói". Tín hiệu sóng âm của hai âm tiết tiếng Việt có thể nhìn thấy như sau:



Hình 1.1 : Tín hiệu sóng âm của hai âm tiêt Tiếng Việt

Tuy nhiên, một văn bản thật sự (một bài báo khoa học chẳng hạn) có thể có đến hàng nghìn câu, và ta không phải có một mà hàng triệu văn bản. Web là một nguồn dữ liệu văn bản khổng lồ, và cùng với các thư viện điện tử − khi trong một tương lai gần các sách báo xưa nay và các nguồn âm thanh được chuyển hết vào máy tính (chẳng hạn bằng các chương trình nhận dạng chữ, thu nhập âm thanh, hoặc gõ thẳng vào máy) − sẽ sớm chứa hầu như toàn bộ kiến thức của nhân loại. Vấn đề là làm sao "xử lý" (chuyển đổi) được khối dữ liệu văn bản và tiếng nói khổng lồ này qua dạng khác để mỗi người có được thông tin và tri thức cần thiết từ chúng.

Xử lý ngôn ngữ tự nhiên đã được ứng dụng trong thực tế để giải quyết các bài toán như : nhận dạng chữ viết, nhận dạng tiếng nói, tổng hợp tiếng nói, dịch tự động, tìm kiếm thông tin, tóm tắt văn bản, khai phá dữ liệu và phát hiện tri thức.

Section 2.011.2. Cơ sở khoa học

1.2.1 Một số khái niệm cơ bản

1.2.1.1. Ngôn ngữ tự nhiên

Ngôn ngữ là hệ thống để giao thiệp hay suy luận dùng một cách biểu diễn phép ẩn dụ và một loại ngữ pháp theo logic, mỗi cái đó bao hàm một tiêu chuẩn hay sự thật thuộc lịch sử và siêu việt. Nhiều ngôn ngữ sử dụng điệu bộ, âm thanh, ký hiệu, hay chữ viết, và cố gắng truyền khái niệm, ý nghĩa, và ý nghĩ, nhưng mà nhiều khi những khía cạnh này nằm sát quá, cho nên khó phân biệt nó.

(a) 1.2.1.2. Xử lý ngôn ngữ tự nhiên

Xử lý ngôn ngữ tự nhiên (natural language processing - NLP) là một nhánh của trí tuệ nhân tạo tập trung vào các ứng dụng trên ngôn ngữ của con người. Trong trí tuệ nhân tạo thì xử lý ngôn ngữ tự nhiên là một trong những phần khó nhất vì nó liên quan đến việc phải hiểu ý nghĩa ngôn ngữ - công cụ hoàn hảo nhất của tư duy và giao tiếp.

(b) 1.2.1.3. Trí tuệ nhân tạo

Trí tuệ nhân tạo hay trí thông minh nhân tạo (tiếng Anh: artificial intelligence hay machine intelligence, thường được viết tắt là AI) là trí tuệ được biểu diễn bởi bất cứ một hệ thống nhân tạo nào. Thuật ngữ này thường dùng để nói đến các máy tính có mục đích không nhất định và ngành khoa học nghiên cứu về các lý thuyết và ứng dụng của trí tuệ nhân tạo.

(c) 1.2.1.4. Nhập nhằng

Nhập nhằng trong ngôn ngữ học là hiện tượng thường gặp, trong giao tiếp hàng ngày con người ít để ý đến nó bởi vì họ xử lý tốt hiện tượng này. Nhưng trong các ứng dụng liên quan đến xử lý ngôn ngữ tự nhiên khi phải thao tác với ý nghĩa từ vựng mà điển hình là dịch tự động nhập nhằng trở thành vấn đề nghiêm trọng . Ví dụ trong một câu cần dịch có xuất hiện từ "đường" như trong câu "ra chợ mua cho mẹ ít đường" vấn đề nảy sinh là cần dịch từ này là road hay sugar, con người xác định chúng khá dễ dàng căn cứ vào văn cảnh và các dấu hiệu nhận biết khác nhưng với máy thì không. Một số hiện tượng nhập nhằng: Nhập nhằng ranh giới từ, Nhập nhằng từ đa nghĩa, Nhập nhằng từ đồng âm (đồng tự), Nhập nhằng từ loại.

1.2.1.5. Dịch máy

Dịch máy là một trong những ứng dụng chính của xử lý ngôn ngữ tự nhiên, dùng máy tính để dịch văn bản từ ngôn ngữ này sang ngôn ngữ khác. Mặc dù dịch máy đã được nghiên cứu và phát triển hơn 50 năm qua, xong vẫn tồn tại nhiều vấn đề cần nghiên cứu. Ở Việt Nam, dịch máy đã được nghiên cứu hơn 20 năm, nhưng các sản phẩm dịch máy hiện tại cho chất lượng dịch còn nhiều hạn chế. Hiện nay,

dịch máy được phân chia thành một số phương pháp như: dịch máy trên cơ sở luật, dịch máy thống kê và dịch máy trên cớ sở ví dụ.

1.2.2 Lý thuyết thông tin

(d) 1.2.2.1. Khái niệm

Lý thuyết thông tin nghiên cứu về: Áp dụng các công cụ toán học trong việc lượng hóa dữ liệu cho mục đích lưu trữ và truyền dữ liệu. Độ đo thông tin là Entropy, là số lượng bít trung bình cần thiết để cho việc lưu trữ hay truyền dữ liệu. Đóng vai trò quan trọng trong xử lý thông tin bằng các phương pháp thống kê, đặc biệt trong NLP.

(e) 1.2.2.2. Entropy

Entropy là một độ đo thông tin. Entropy ~ hỗn độn, mờ, trái nghĩa với order...

Đo độ không chắc chắn: Entropy thấp -> Đo độ không chắc chắn thấp; Entropy cao -> Đo độ không chắc chắn cao. Trong vật lý: Entropy giảm khi năng lượng được sử dụng. Ký hiệu p(x) là một phân bố của một biến ngẫu nhiên X. Ω là không gian mẫu của X. Entropy được tính như sau:

$$H(X) = - \sum x \in \Omega \quad p(x) \log 2 p(x).$$

Đơn vị: bits (log10: nats). Kí hiệu: H(X) = Hp(X) = H(p).

(f) 1.2.2.3. Perplexity - Cross Entropy

1. Entropy liên quan thế nào đến hiểu ngôn ngữ?

Liên quan đến sự không chính xác: một vấn đề càng có nhiều thông tin thì Entropy càng thấp. Có nhiều mô hình -> entropy đo chất lượng của các mô hình?

Ví dụ: mô hình mã hóa ký tự với trung bình số bít sử dụng trên mỗi ký tự là 2.5 . Đây là mô hình ngôn ngữ 0-gram, nếu đặt trong sự liên kết của các âm tiết thì chúng ta có thể sinh được mô hình tốt hơn, chẳng hạn cho entropy 1.22 bít trên một ký tự.

2. Perplexity

Entropy của một phân bố p(X) là: Hp(X) thì giá trị 2H được gọi là perplexity perplexity là số lượng mẫu trung bình mà một biến phải lựa chọn. Perlexity càng bé (tức là entropy càng bé) thì mô hình càng tốt <=> số bít dùng để mã hóa thông tin càng bé.

Ví dụ : Cho 8 con ngựa với xác suất lựa chọn như sau:

Ngựa 1: 1/2 ngựa 2: 1/4 ngựa 3: 1/8 ngựa 4: 1/16

Ngựa 5: 1/64 ngựa 2: 1/64 ngựa 3: 1/64 ngựa 4: 1/64

3. Entropy rate

Tính entropy của một dãy các từ trong một ngôn ngữ L

$$H(w1,...,wn) = - \sum W \in L \; p(W1n)\log(W1n)$$

Entropy rate được coi như per-word entropy. Coi một ngôn ngữ như một quá trình ngẫu nhiên sản xuất một dãy các từ. Cần quan tâm đến một dãy vô hạn từ. Entropy rate H(L) được định nghĩa như sau:

$$H(L) = \lim_{n->\infty} \frac{1}{n} H(w_1,..., w_n) = \lim_{n->\infty} -\frac{1}{n} \sum_{\in L} p(w_1,..., w_n) \log p(w_1,..., w_n)$$

4 . Cross Entropy

Cross entropy được sử dụng khi chúng ta không biết phân bố thật p.

Cross-entropy của phân bố m của phân bố thật p được định nghĩa:

$$H(p,m) = \lim_{n->\infty} -\frac{1}{n} \sum_{\in L} p(w_1,..., w_n) \log m(w_1,..., w_n) = \lim_{n->\infty} -\frac{1}{n} \log m(w_1,..., w_n)$$

(theo lý thuyết Shannon-McMillan-Breiman)

5. Cross entropy để so sánh các mô hình : H(p) ≤ H(p,m)

Cross entropy H(p,m) là cận trên của entropy H(p);

Mô hình m càng chính xác thì cross entropy H(p,m) càng gần với entropy H(p);

Độ khác nhau H(p,m) và H(p) đo độ chính xác của mô hình m;

6. Các công thức Cross Entropy

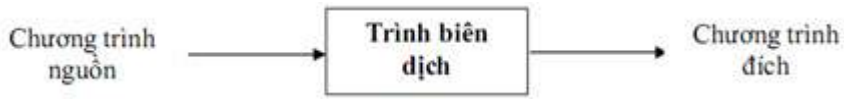
Cross entropy giữa biến X với phân bố xác suất đúng p(x) và một phân bố m được tính như sau:

$$H(X,m) = H(X) + D(p \parallel m) = -\sum_{x} p(x) \log m(x)$$

Chú ý: $D(p||q) = \sum x\ p(x) \log 2\ (p(x)/q(x))$

1.3 Quy trình xử lý ngôn ngữ tự nhiên

Để máy tính có thể hiểu và thực thi một chương trình được viết bằng ngôn ngữ cấp cao, ta cần phải có một trình biên dịch thực hiện việc chuyển đổi chương trình đó sang chương trình ở dạng ngôn ngữ đích. Chương này trình bày một cách tổng quan về cấu trúc của một trình biên dịch và mối liên hệ giữa nó với các thành phần khác - "họ hàng" của nó - như bộ tiền xử lý, bộ tải và soạn thảo liên kết,v.v. Cấu trúc của trình biên dịch được mô tả trong chương là một cấu trúc mức quan niệm bao gồm các giai đoạn: Phân tích từ vựng, Phân tích cú pháp, Phân tích ngữ nghĩa, Sinh mã trung gian, Tối ưu mã và Sinh mã đích. Nói một cách đơn giản, trình biên dịch là một chương trình làm nhiệm vụ đọc một chương trình được viết bằng một ngôn ngữ - ngôn ngữ nguồn (source language) - rồi dịch nó thành một chương trình tương đương ở một ngôn ngữ khác - ngôn ngữ đích (target languague). Một phần quan trọng trong quá trình dịch là ghi nhận lại các lỗi có trong chương trình nguồn để thông báo lại cho người viết chương trình.



Hình 1.2 : Một trình biên dịch

(g) 1.3.1 Phân tích từ vựng (Lexical Analysis)

Trong một trình biên dịch, giai đoạn phân tích từ vựng sẽ đọc chương trình nguồn từ trái sang phải (quét nguyên liệu - scanning) để tách ra thành các thẻ từ (token).

Ví dụ 1.2: Quá trình phân tích từ vựng cho câu lệnh gán position := initial + rate * 60 sẽ tách thành các token như sau:

1. Danh biểu position
2. Ký hiệu phép gán :=
3. Danh biểu initial
4. Ký hiệu phép cộng (+)
5. Danh biểu rate
6. Ký hiệu phép nhân (*)
7. Số 60

Trong quá trình phân tích từ vựng các khoảng trắng (blank) sẽ bị bỏ qua.

(h) 1.3.2 Phân tích cú pháp (Syntax Analysis)

Giai đoạn phân tích cú pháp thực hiện công việc nhóm các thẻ từ của chương trình nguồn thành các ngữ đoạn văn phạm (grammatical phrase), mà sau đó sẽ được trình biên dịch tổng hợp ra thành phẩm. Thông thường, các ngữ đoạn văn phạm này được biểu diễn bằng dạng cây phân tích cú pháp (parse tree) với:

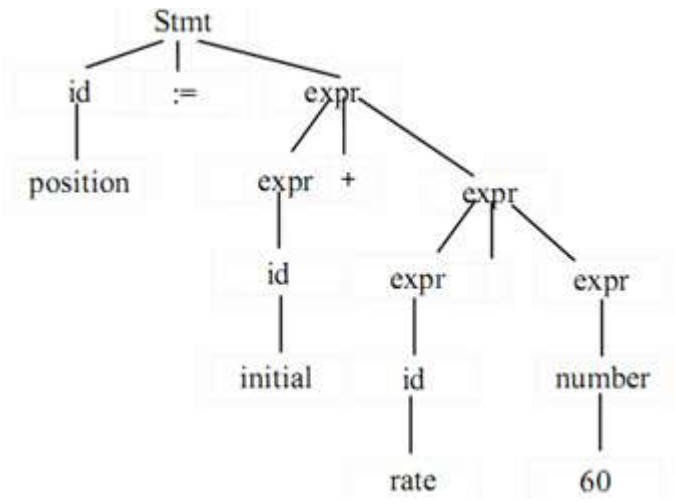
- Ngôn ngữ được đặc tả bởi các luật sinh.
- Phân tích cú pháp dựa vào luật sinh để xây dựng cây phân tích cú pháp.

Ví dụ 1.3: Giả sử ngôn ngữ đặc tả bởi các luật sinh sau:

Stmt → id := expr

expr → expr + expr | expr * expr | id | number

Với câu nhập: position := initial + rate * 60, cây phân tích cú pháp được xây dựng như sau:



Hình 1.3 : Một cây phân tích cú pháp

Cấu trúc phân cấp của một chương trình thường được diễn tả bởi quy luật đệ qui.

Ví dụ 1.4:

1) Danh biểu (identifier) là một biểu thức (expr).

2) Số (number) là một biểu thức.

3) Nếu expr1 và expr2 là các biểu thức thì:

expr1 + expr2

expr1 * expr2

(expr)

4) Cũng là những biểu thức. Câu lệnh (statement) cũng có thể định nghĩa đệ qui :

Nếu id1 là một danh biểu và expr2 là một biểu thức thì id1 := expr2 là một lệnh (stmt).

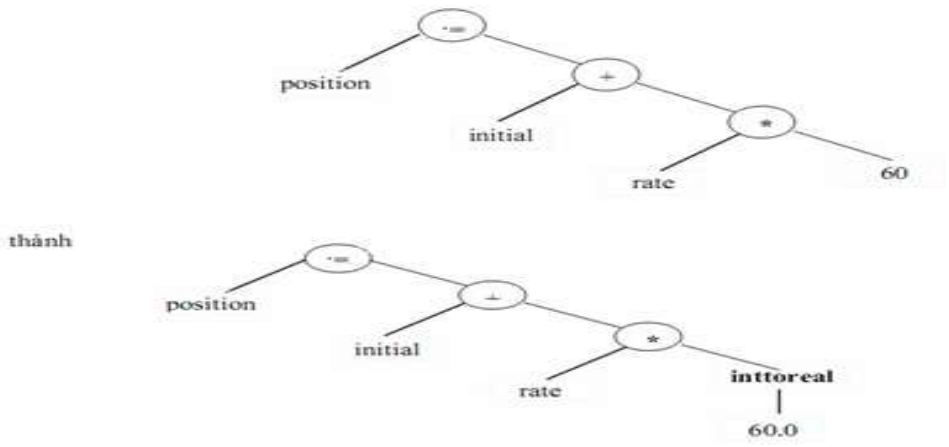
Nếu expr1 là một biểu thức và stmt2 là một lệnh thì while (expr1) do stmt2 và if (expr1) then stmt2: đều là các lệnh. Người ta dùng các qui tắc đệ qui như trên để đặc tả luật sinh (production) cho ngôn ngữ. Sự phân chia giữa quá trình phân tích từ vựng và phân tích cú pháp cũng tuỳ theo công việc thực hiện.

(i) 1.3.3 Phân tích ngữ nghĩa (Semantic Analysis)

Giai đoạn phân tích ngữ nghĩa sẽ thực hiện việc kiểm tra xem chương trình nguồn có chứa lỗi về ngữ nghĩa hay không và tập hợp thông tin về kiểu cho giai đoạn sinh mã về sau. Một phần quan trọng trong giai đoạn phân tích ngữ nghĩa là kiểm tra kiểu (type checking) và ép chuyển đổi kiểu.

Ví dụ 1.5: Trong biểu thức position := initial + rate * 60

Các danh biểu (tên biến) được khai báo là real, 60 là số integer vì vậy trình biên dịch đổi số nguyên 60 thành số thực 60.0



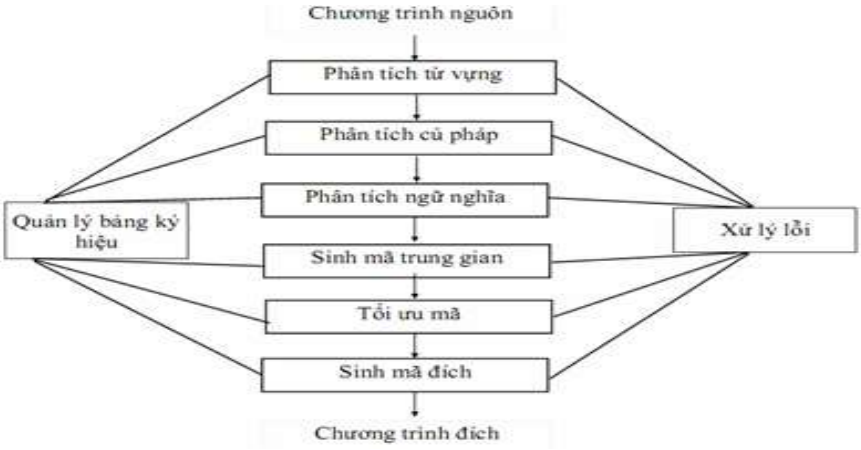
Hình 1. 4 : Chuyển đổi kiểu trên cây phân tích cú pháp

(j) 1.3.4 Các giai đoạn của trình biên dịch

Một trình biên dịch được chia thành các giai đoạn, mỗi giai đoạn chuyển chương trình nguồn từ một dạng biểu diễn này sang một dạng biểu diễn khác.

VÍ DỤ: Một cách phân rã điển hình trình biên dịch được trình bày trong hình

:



Hình 1.5 : Các giai đoạn của một trình biên dịch

Việc quản lý bảng ký hiệu và xử lý lỗi được thực hiện xuyên suốt qua tất cả các giai đoạn. Các giai đoạn mà chúng ta đề cập ở trên là thực hiện theo trình tự logic của một trình biên dịch. Nhưng trong thực tế, cài đặt các hoạt động của nhiều hơn một giai đoạn có thể được nhóm lại với nhau. Thông thường chúng được nhóm thành hai nhóm cơ bản, gọi là: Kỳ đầu (Front end) và kỳ sau (Back end).

1. Kỳ đầu (Front End)

Kỳ đầu bao gồm các giai đoạn hoặc các phần giai đoạn phụ thuộc nhiều vào ngôn ngữ nguồn và hầu như độc lập với máy đích. Thông thường, nó chứa các giai đoạn sau: Phân tích từ vựng, Phân tích cú pháp, Phân tích ngữ nghĩa và Sinh mã trung gian. Một phần của công việc tối ưu hóa mã cũng được thực hiện ở kỳ đầu. Front end cũng bao gồm cả việc xử lý lỗi xuất hiện trong từng giai đoạn.

2. Kỳ sau (Back End)

Kỳ sau bao gồm một số phần nào đó của trình biên dịch phụ thuộc vào máy đích và nói chung các phần này không phụ thuộc vào ngôn ngữ nguồn mà là ngôn ngữ trung gian. Trong kỳ sau, chúng ta gặp một số vấn đề tối ưu hoá mã, phát sinh mã đích cùng với việc xử lý lỗi và các thao tác trên bảng ký hiệu.

1.3.5 Một số thuật toán phân tích cú pháp

(k) 1.3.5.1 Topdown

Phân tích từ trên xuống, từ trái qua phải;

Khi gặp một từ (terminal) thì phân tích nút tiếp theo;

Khi không tương ứng với input word thì quay lui;

(l) 1.3.5.2 Bottom-up

Là một dạng của shift-reduce actions;

Khi gặp vế phải của một luật thì thu gọn thành vế trái;

Khi không phân tích được tiếp thì quay lui;

(m) 1.3.5.3 CYK (Cocke-Younger-Kasami)

Văn phạm dạng chuẩn Chomsky (Chomsky Normal Form);

Các luật thuộc một trong 2 dạng:

A -> B C

A -> a

Ví dụ:

S -> X Y

X -> X A | a | b

Y -> A Y | a

A -> a

Phân tích câu “babaa” -> không sinh ra câu

“baaa” -> sinh ra câu

S, X			
S, X	S,Y		
S, X	S, X, Y	S, X, Y	
X	X, Y, A	X, Y, A	X, Y, A
b	a	a	a

```
S -> X Y
X -> X A | a | b
Y -> A Y | a
A -> a
"baaa"
```

Xác định các đặc điểm sau đây:

1)Sinh ra giá trị một nút như thế nào?

A[i,j] <- ? + ?

2)Lưu lại đường đi như thế nào để sinh lại cây

Tính nhập nhằng: Một A[,] có thể có nhiều tag, mỗi tag lại được dẫn xuất bằng nhiều cách.

3)Tại sao thuật toán CYK lại cần văn phạm dạng chuẩn Chomsky.

Phân tích câu:

“book that flight”

“book the flight through Houston”

$S \rightarrow NP\ VP$	$Det \rightarrow that \mid this \mid a$
$S \rightarrow Aux\ NP\ VP$	$Noun \rightarrow book \mid flight \mid meal \mid money$
$S \rightarrow VP$	$Verb \rightarrow book \mid include \mid prefer$
$NP \rightarrow Pronoun$	$Pronoun \rightarrow I \mid she \mid me$
$NP \rightarrow Proper\text{-}Noun$	$Proper\text{-}Noun \rightarrow Houston \mid TWA$
$NP \rightarrow Det\ Nominal$	$Aux \rightarrow does$
$Nominal \rightarrow Noun$	$Preposition \rightarrow from \mid to \mid on \mid near \mid through$
$Nominal \rightarrow Nominal\ Noun$	
$Nominal \rightarrow Nominal\ PP$	
$VP \rightarrow Verb$	
$VP \rightarrow Verb\ NP$	
$VP \rightarrow Verb\ NP\ PP$	
$VP \rightarrow Verb\ PP$	
$VP \rightarrow VP\ PP$	
$PP \rightarrow Preposition\ NP$	

Figure 13.1 The $\mathscr{L}_1$ miniature English grammar and lexicon.

Chuyển từ văn phạm CFG sang văn phạm dạng chuẩn Chomsky

1) A -> B C D

A -> X D

X -> B C

2) Bỏ luật dạng A -> B

Với mọi B -> α, sinh luật A -> α

$S \rightarrow NP\ VP$	$S \rightarrow NP\ VP$
$S \rightarrow Aux\ NP\ VP$	$S \rightarrow X1\ VP$
	$X1 \rightarrow Aux\ NP$
$S \rightarrow VP$	$S \rightarrow book \mid include \mid prefer$
	$S \rightarrow Verb\ NP$
	$S \rightarrow X2\ PP$
	$S \rightarrow Verb\ PP$
	$S \rightarrow VP\ PP$
$NP \rightarrow Pronoun$	$NP \rightarrow I \mid she \mid me$
$NP \rightarrow Proper\text{-}Noun$	$NP \rightarrow TWA \mid Houston$
$NP \rightarrow Det\ Nominal$	$NP \rightarrow Det\ Nominal$
$Nominal \rightarrow Noun$	$Nominal \rightarrow book \mid flight \mid meal \mid money$
$Nominal \rightarrow Nominal\ Noun$	$Nominal \rightarrow Nominal\ Noun$
$Nominal \rightarrow Nominal\ PP$	$Nominal \rightarrow Nominal\ PP$
$VP \rightarrow Verb$	$VP \rightarrow book \mid include \mid prefer$
$VP \rightarrow Verb\ NP$	$VP \rightarrow Verb\ NP$
$VP \rightarrow Verb\ NP\ PP$	$VP \rightarrow X2\ PP$
	$X2 \rightarrow Verb\ NP$
$VP \rightarrow Verb\ PP$	$VP \rightarrow Verb\ PP$
$VP \rightarrow VP\ PP$	$VP \rightarrow VP\ PP$
$PP \rightarrow Preposition\ NP$	$PP \rightarrow Preposition\ NP$

Figure 13.8 $\mathscr{L}_1$ Grammar and its conversion to CNF. Note that although they aren't shown here all the original lexical entries from $\mathscr{L}_1$ carry over unchanged as well.

S						
	VP					
S						
	VP			PP		
S		NP			NP	
NP	V, VP	Det	N	P	Det	N
she	eats	a	fish	with	a	fork

Hình 1.6. Thử sinh ra một văn phạm tương ứng

1.4 Các ứng dụng của xử lý ngôn ngữ tự nhiên

1. Nhận dạng tiếng nói (speech recognition): Từ sóng tiếng nói, nhận biết và chuyển chúng thành dữ liệu văn bản tương ứng. Giúp thao tác của con người trên các thiết bị nhanh hơn và đơn giản hơn, chẳng hạn thay vì gõ một tài liệu nào đó bạn đọc nó lên và trình soạn thảo sẽ tự ghi nó ra. Đây cũng là bước đầu tiên cần phải thực hiện trong ước mơ thực hiện giao tiếp giữa con người với robot. Nhận dạng tiếng nói có khả năng trợ giúp người khiếm thị rất nhiều.

2. Tổng hợp tiếng nói (speech synthesis): Từ dữ liệu văn bản, phân tích và chuyển thành tiếng người nói. Thay vì phải tự đọc một cuốn sách hay nội dung một trang web, nó tự động đọc cho chúng ta. Giống như nhận dạng tiếng nói, Tổng hợp tiếng nói là sự trợ giúp tốt cho người khiếm thị, nhưng ngược lại nó là bước cuối cùng trong giao tiếp giữa người với robot.

3. Nhận dạng chữ viết (optical character recognition, OCR): Từ một văn bản in trên giấy, nhận biết từng chữ cái và chuyển chúng thành một tệp văn bản trên máy tính. có hai kiểu nhận dạng: Thứ nhất là nhận dạng chữ in như nhận dạng chữ trên sách giáo khoa rồi chuyển nó thành dạng văn bản điện tử như dưới định dạng

doc của Microsoft Word chẳng hạn. Phức tạp hơn là nhận dạng chữ viết tay, có khó khăn bởi vì chữ viết tay không có khuôn dạng rõ ràng thay đổi từ người này sang người khác.Với chương trình nhận dạng chữ viết in có thể chuyển hàng ngàn đầu sách trong thư viện thành văn bản điện tử trong thời gian ngắn. Nhận dạng chữ viết của con người có ứng dụng trong khoa học hình sự và bảo mật thông tin (nhận dạng chữ ký điện tử).

4. Dịch tự động (machine translation): Từ một tệp dữ liệu văn bản trong một ngôn ngữ (tiếng Anh chẳng hạn), máy tính dịch và chuyển thành một tệp văn bản trong một ngôn ngữ khác. Một phần mềm điển hình về tiếng Việt của chương trình này là Evtrans của Softex, dịch tự động từ tiếng Anh sang tiếng Việt và ngược lại, phần mềm từng được trang web vdict.com mua bản quyền, đây cũng là trang đầu tiên đưa ứng dụng này lên mạng. Có hai công ty tham gia vào lĩnh vực này cho ngôn ngữ tiếng Việt là công ty Lạc Việt (công ty phát hành từ điển Lạc Việt) và Google.

5. Tóm tắt văn bản (text summarization): Từ một văn bản dài (mười trang chẳng hạn) máy tóm tắt thành một văn bản ngắn hơn (một trang) với những nội dung cơ bản.

6. Tìm kiếm thông tin (information retrieval): Từ một nguồn rất nhiều tệp văn bản hay tiếng nói, tìm ra những tệp có nội dung liên quan đến một vấn đề (câu hỏi) ta cần biết (hay trả lời). Điển hình của công nghệ này là Google, một hệ tìm kiếm thông tin trên Web, mà hầu như chúng ta đều dùng thường xuyên. Cần nói thêm rằng mặc dù hữu hiệu hàng đầu như vậy, Google mới có khả năng cho chúng ta tìm kiếm câu hỏi dưới dạng các từ khóa (keywords) và luôn "tìm" cho chúng ta rất nhiều tài liệu không liên quan, cũng như rất nhiều tài liệu liên quan đã tồn tại thì Google lại tìm không ra.

7. Trích chọn thông tin (information extraction): Từ một nguồn rất nhiều tệp văn bản hay tiếng nói, tìm ra những đoạn bên trong một số tệp liên quan đến một vấn đề (câu hỏi) ta cần biết hay trả lời. Một hệ trích chọn thông tin có thể "lần" vào

từng trang Web liên quan, phân tích bên trong và trích ra các thông tin cần thiết, nói gọn trong tiếng Anh để phân biệt với tìm kiếm thông tin là "find things but not pages".

8. Phát hiện tri thức và khai phá dữ liệu văn bản (knowledge discovery and text data mining): Từ những nguồn rất nhiều văn bản thậm chí hầu như không có quan hệ với nhau, tìm ra được những tri thức trước đấy chưa ai biết. Đây là một vấn đề rất phức tạp và đang ở giai đoạn đầu của các nghiên cứu trên thế giới.

Có thể phân loại các bài toán

- 1-3 thuộc lĩnh vực xử lý tiếng nói và xử lý ảnh (speech and image processing),
- 4-5 thuộc lĩnh vực xử lý văn bản (text processing),
- 6-8 thuộc lĩnh vực khai phá văn bản và Web (text and Web mining).

Chương 2 : *Ứng dụng xử lý ngôn ngữ tự nhiên trong dịch máy*

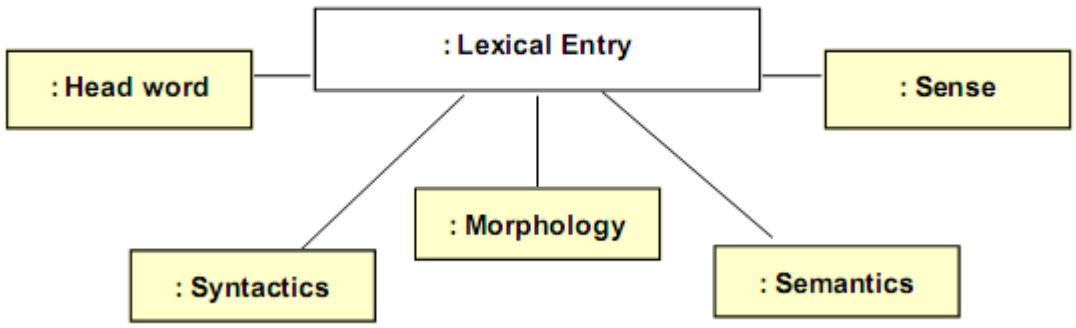
Xử lý ngôn ngữ tự nhiên là một lĩnh vực nghiên cứu nhằm giúp cho các hệ thống máy tính hiểu và xử lý được ngôn ngữ con người. Dịch máy là một trong những ứng dụng chính của xử lý ngôn ngữ tự nhiên. Mặc dù dịch máy đã được nghiên cứu và phát triển hơn 50 năm qua, song vẫn tồn tài nhiều vấn đề cần nghiên cứu. Ở Việt Nam, dịch máy đã được nghiên cứu hơn 20 năm, nhưng các sản phẩm dịch máy hiện tại cho chất lượng dịch còn hạn chế. Hiện nay, dịch máy được phân chia thành một số phương pháp như: dịch máy trên cơ sở luật, dịch máy thống kê và dịch máy trên cơ sở ví dụ. Do những khác biệt về ngữ hệ, khác biệt về văn hóa và thiếu vắng nguồn tài nguyên, nên các phương pháp dịch máy hiện hữu thường gặp trở ngại khi áp dụng vào cặp ngôn ngữ Anh-Việt.

Phương pháp dịch máy trên cơ sở luật cần phải xây dựng hệ thống luật cú pháp, ngữ nghĩa và phải có một từ điển khá đầy đủ thông tin cho các mục từ như ngữ nghĩa, ngữ dụng… Để thực hiện được phương pháp dịch máy trên cơ sở luật, người ta cần nhiều thời gian và tiền bạc nhưng sản phẩm dịch vẫn không đạt độ chính xác như mong đợi. Dịch máy bằng phương pháp thống kê chưa có nhiều nghiên cứu áp dụng cho cặp Anh-Việt. Do sự khác biết khá lớn về cấu trúc cú pháp của câu và nguồn ngữ liệu song ngữ chuẩn, nên ảnh hưởng đến chất lượng đối sánh từ Anh-Việt, mà kết quả của đối sánh từ lại quyết định đến chất lượng dịch. Phương pháp dịch máy trên cơ sở ví dụ truyền thống sử dụng các câu mẫu. Các câu này được lưu trữ trên cơ sở dữ liệu với đầy đủ các thông tin như cây chú giải, các liên kết giữa các thành phàn của hai câu thuộc hai ngôn ngữ. Phương pháp này cũng cần tập luật cú pháp của các ngôn ngữ nguồn để xây dựng cơ sở dữ liệu cho mẫu câu ví dụ. Sự khác biệt từ sẽ được xác định thông qua từ điển phân lớp, câu nhập sẽ được phân tích bằng tập luật cú pháp và xác định cặp cây cú pháp của câu nguồn và cây đích. Dịch máy dựa trên ngữ liệu đang được áp dụng vào nhiều hệ thống dịch tự động trong những năm gần đây, việc lấy đúng được cặp ánh xạ đích và nguồn một cách tự dộng là một yêu cầu thiết yếu cho các phương pháp dịch dựa trên ngữ liệu.

2.1 Xây dựng từ điển Tiếng Việt cho máy tính

Trong xử lí ngôn ngữ tự nhiên (Natural Language Processing), từ điển cho máy tính (Machine Readable Dictionary - MRD) là một dạng tài nguyên thiết yếu cho các bài toán phân tích ngôn ngữ từ đơn giản đến phức tạp. Một kho từ vựng chất lượng tốt phải cung cấp được cho các hệ thống xử lí ngôn ngữ tự nhiên các thông tin ngôn ngữ ở nhiều tầng bậc khác nhau như hình thái, ngữ pháp, ngữ nghĩa, tốt hơn nữa là có thể phục vụ cả các hệ thống xử lí đơn ngữ và đa ngữ.

Một mục từ của từ điển điện tử thường cung cấp tri thức về chính tả, ngữ âm, từ nguyên, cấu tạo từ, khả năng kết hợp, quan hệ ngữ pháp, quan hệ ngữ nghĩa, v.v. (Vũ Xuân Lương, 2002) của từ ngữ. Những tri thức này tuỳ thuộc vào từng ngôn ngữ và tuỳ thuộc vào từng mục đích sử dụng mà có thể có những yêu cầu thể hiện khác nhau. Nhưng nhìn trên tổng thể, một từ điển như vậy phải được xây dựng dựa trên những nét phổ quát cho mọi ngôn ngữ. Mục đích của phần này là đưa ra lí do lựa chọn mô hình biểu diễn thông tin và cách thức biểu diễn thông tin trong từ điển. Các thông tin mô tả được thể hiện trên 3 bình diện: hình thái học, cú pháp học và ngữ nghĩa học.



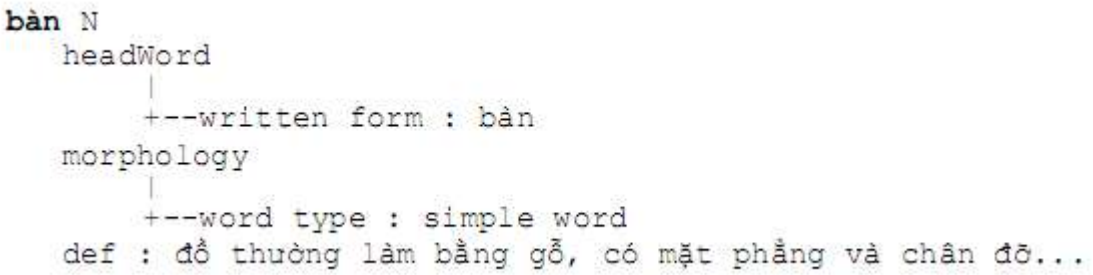
Hình 2.1. Cấu trúc tổng quát của một mục từ

2.1.1 Thông tin hình thái (Morphology)

Từ của tiếng Việt, trong cấu tạo, không có căn tố và phụ tố; trong ngữ nghĩa, không có các ý nghĩa thuộc phạm trù hình thái; trong hoạt động tạo câu, các mối liên hệ ngữ pháp không biểu hiện ở sự biến hình mà biểu hiện bằng trật tự từ. Vì lẽ đó, khi xét về tính hình thái của tiếng Việt, thông thường chỉ xét về vấn đề cấu tạo từ.

Thông tin về cấu tạo từ khi được kết hợp với thông tin cú pháp và ngữ nghĩa sẽ có ích cho các nghiên cứu về tách từ, đoán định đơn vị từ trong văn bản tiếng Việt. Chẳng hạn đoán định cụm từ và từ (*sữa bò và bò sữa, tấm vải và vải tấm, xay máy và máy xay*…), đoán định cơ chế sinh từ láy,v.v. Trong từ điển xây dựng, các dạng cấu tạo từ được chú ý như sau:

- từ đơn: simple word
- từ ghép: composite word
- từ láy: reduplicative word
- từ vay mượn: borrowed word
- từ tắt: abbreviation
- kí hiệu: symbol



Hình 2.2. Thông tin hình thái của "bàn"

Thông tin hình thái được mô tả trong từ điển chỉ mới dừng lại ở mức gán nhãn bậc một cho mỗi đơn vị từ vựng, các thông tin ở mức sâu hơn sẽ được nghiên cứu sau.

2.1.2 Thông tin cú pháp (Syntactics)

Thông tin về từ loại (category)

Các từ thường có chung đặc điểm ngữ pháp và ý nghĩa khái quát, như danh từ, động từ, tính từ, v.v. Mỗi loại từ như vậy phản ánh khả năng kết hợp và chức năng cú pháp khác nhau. Chẳng hạn khi tạo câu, nếu vị ngữ là *danh từ* thì phải dùng *là*, ngược lại nếu vị ngữ là *tính từ* thì không cần *là* (Nguyễn Kim Thản, 1997): *đây là quyển sách; sách này hay quá*. Việc phân định các loại từ là nhằm mục đích tạo câu cho đúng, do vậy việc mô tả chúng là có ý nghĩa. Trong từ điển đề cập đến 14 loại sau:

idPOS	vnPOS	enPOS	symbolPOS
1	Danh từ	Noun	N
2	Động từ	Verb	V
3	Tính từ	Adjective	A
4	Số từ	Numeral	M
5	Định từ	Determiner	D
6	Đại từ	Pronuon	P
7	Phụ từ	Adverb	R
8	Giới từ	Preposition	O
9	Liên từ	Conjunction	C
10	Trợ từ	Auxiliary word	I
11	Cảm từ	Emotivity word	E
12	Yếu tố cấu tạo từ	Component stem	S
13	Từ tắt	Abbreviation	Y
14	Không xác định	Undetermined	U

2.1.3. Thông tin ngữ nghĩa (Semantics)

2.1.3.1. Ràng buộc logic (logical constraint)

Các ngôn ngữ có thể có một hệ thống từ loại ngữ nghĩa căn bản giống nhau. Có hai loại ngữ nghĩa lớn, một loại biểu thị thực thể (thể từ) và một loại biểu thị thuộc tính của thực thể hoặc thuộc tính của thuộc tính (gọi là thuộc từ - mang ý nghĩa trừu tượng). Đại từ và phần lớn danh từ là thể từ, nhưng cũng có nhiều danh từ là thuộc từ (danh từ chỉ tình cảm, màu sắc, hình dáng, v.v.) (Hoàng Phê, 2008). Trong hai loại lớn lại phân chia ra thành các loại nhỏ, trong mỗi loại nhỏ lại được phân chia ra loại nhỏ hơn. Từ điển tổ chức từ loại ngữ nghĩa theo mô hình quan hệ hình cây, gần 100 tiểu loại. Cây ngữ nghĩa này được tham khảo từ dự án TCL (Thai Computational Lexicon) (Charoenporn, 2004) có hơn 60.000 mục từ Thái – Anh, được mô tả trên 3 bình diện: hình thái học, cú pháp học và ngữ nghĩa học, v.v...

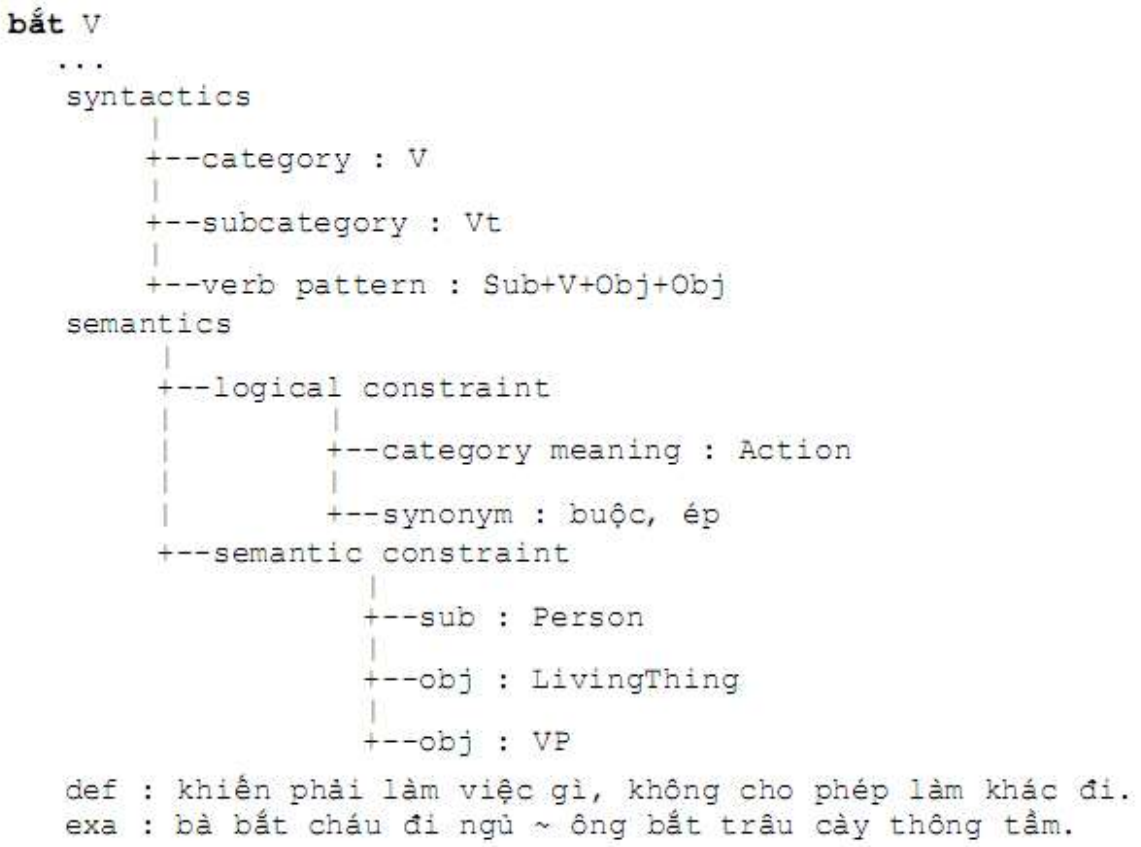
```
SEMANTIC TREE
|
+ Thực thể : Concrete Thing
|   |
|   + Vật hữu sinh : Living Thing
|   |   |
|   |   + Con người    : People
|   |   + Động vật     : Animal
|   |   + Vi sinh vật  : Microorganism
|   |   + Thực vật     : Plant
|   |   ...
|   + Vật vô sinh : Non Living Thing
|   |   |
|   |   + Vật dụng : Artifact
|   |   ...
|   + Vị trí : Location
|   |   ...
+ Trừu tượng : Abstraction
|   |
|   + Lĩnh vực tri thức: Field Of Knowledge
|   + Trạng thái : State
|   + Hoạt động  : Action
|   + Quan hệ    : Relation
|   ...
```

Hình 2.3: Cây ngữ nghĩa trong từ điển

Như vậy, mỗi đơn vị từ vựng trong từ điển ngoài việc được gán nhãn từ loại ngữ pháp (học sinh – Nc) còn được gán thêm một nhãn từ loại ngữ nghĩa (học sinh – Person). Việc làm này giúp cho việc phân loại từ được triệt để hơn, hoặc giúp cho việc phân tích cú pháp được sâu sắc hơn. Cây ngữ nghĩa được chi tiết trong bảng Phụ lục 1.

2.1.3.2. Ràng buộc ngữ nghĩa (semantic constraint)

Trong quá trình tạo câu, ngoài việc câu phải có đầy đủ các thành phần (đúng ngữ pháp) còn đòi hỏi các thành phần câu phải có mối liên kết, ràng buộc ngữ nghĩa lẫn nhau. Chỉ có xác lập được mối liên kết, ràng buộc ngữ nghĩa thì mới nhận ra được câu "xe ăn cơm" là không bình thường.



Hình 2.4 Thông tin ngữ nghĩa của "bắt" đòi hỏi hai bổ ngữ.

Do có vai trò quan trọng trong tiến trình phân tích ngôn ngữ nên các thông tin về semantic constraint và logical constraint thường được sử dụng để tạo ra các bộ luật phân tích cú pháp.

2.2 Khảo sát một số phương pháp gán nhãn từ loại

Gắn nhãn từ loại là việc xác định các chức năng ngữ pháp của từ trong câu. Đây là bước cơ bản trước khi phân tích sâu văn phạm hay các vấn đề xử lý ngôn ngữ phức tạp khác. Thông thường, một từ có thể có nhiều chức năng ngữ pháp, ví dụ: trong câu "con ngựa đá đá con ngựa đá", cùng một từ "đá" nhưng từ thứ nhất và thứ ba giữ chức năng ngữ pháp là danh từ, nhưng từ thứ hai lại là động từ trong câu.

Một số hướng tiếp cận chính trong gắn nhãn từ loại tiếng Anh bao gồm: gắn nhãn dựa trên mô hình Markov ẩn (HMM); các mô hình dựa trên bộ nhớ Daelemans , 1996) ; mô hình dựa trên luật (Transformation Based Learning, Brill, 1995); Maximum Entropy; cây quyết định (Schmid, 1994a); mạng nơ-ron(Schmid, 1994b), v.v. Trong các hướng tiếp cận đó, phương pháp dựa trên học máy được đánh giá rất tốt.

Vấn đề gắn nhãn từ loại tiếng Việt có nhiều khó khăn, ngoài khó khăn về đặc trưng riêng về ngôn ngữ, gắn nhãn từ loại tiếng Việt hiện còn rất thiếu các kho dữ liệu chuẩn như Brown hay Penn Treebank trong tiếng Anh cho quá trình so sánh đánh giá. Sau đây tác giả xin giới thiệu 2 phương pháp gán nhãn từ loại tiếng Việt dựa trên phương pháp học máy thống kê (Maximum Entropy và CRFs) - hướng tiếp cận được đánh giá rất tốt trong tiếng Anh.

2.2.1 Phương pháp Maximum Entropy

Tư tưởng chính của Maximum Entropy là "ngoài việc thỏa mãn một số rang buộc nào đó thì mô hình càng đồng đều càng tốt". Để rõ hơn về vấn đề này, ta hãy cùng xem xét bài toán phân lớp gồm có 4 lớp. Ràng buộc duy nhất mà chúng ta chỉ biết là trung bình 40% các tài liệu chứa từ "professor" thì nằm trong lớp **faculty**. Trực quan cho thấy nếu có một tài liệu chứa từ "professor" chúng ta có thể nói có

40% khả năng tài liệu này thuộc lớp **faculty**, và 20% khả năng cho các khả năng còn lại (thuộc một trong 3 lớp còn lại).

Mặc dù maximum entropy có thể được dùng để ước lượng bất kì một phân phối xác suất nào, chúng ta xem xét khả năng maximum entropy cho việc gán nhãn dữ liệu chuỗi. Nói cách khác, ta tập trung vào việc học ra phân phối điều kiện của chuỗi nhãn tương ứng với chuỗi (xâu) đầu vào cho trước.

Trong maximum entropy, người ta dùng dữ liệu huấn luyện để xác định các ràng buộc trên phân phối điều kiện. Mỗi ràng buộc thể hiện một đặc trưng nào đó của dữ liệu huấn luyện. Mọi hàm thực trên chuỗi đầu vào và chuỗi nhãn có thể được xem như là đặc trưng $f_i(o,s)$. Maximum Entropy cho phép chúng ta giới hạn các phân phối mô hình lý thuyết gần giống nhất các giá trị kì vọng cho các đặc trưng này trong dữ liệu huấn luyện D.Vì thế người ta đã mô hình hóa xác suất P(o | s) như sau (ở đây, o là chuỗi đầu vào và s là chuỗi nhãn đầu ra)

$$P(o \mid s) = \frac{1}{Z(o)} \exp\left(\sum_i \lambda_i f_i(o,s)\right)$$

Ở đây $f_i(o,s)$ là một đặc trưng, λ_i là một tham số cần phải ước lượng và Z(o) là thừa số chuẩn hóa đơn giản nhằm đảm bảo tính đúng đắn của định nghĩa xác suất (tổng xác suất trên toàn bộ không gian bằng 1)

$$Z(o) = \sum_c \exp \sum_c \lambda_i f_i(o,s)$$

Một số phương pháp huấn luyện mô hình từ dữ liệu học bao gồm: IIS (improved iterative scaling), GIS, L-BFGS.

2.2.2 Phương pháp Conditional Random Fields (CRFs)

CRFs là mô hình trạng thái tuyến tính vô hướng (máy trạng thái hữu hạn được huấn luyện có điều kiên) và tuân theo tính chất Markov thứ nhất. CRFs đã được chứng minh rất thành công cho các bài toán gán nhãn cho chuỗi như tách từ, gán nhãn cụm từ, xác định thực thể, gán nhãn cụm danh từ, etc.

Gọi $\mathbf{o} = (\mathbf{o_1}, \mathbf{o_2}, \ldots, \mathbf{o_T})$ là một chuỗi dữ liệu quan sát cần được gán nhãn. Gọi S là tập trạng thái, mỗi trạng thái liên kết với một nhãn $l \in L$. Đặt $\mathbf{s} = (\mathbf{s_1}, \mathbf{s_2}, \ldots, \mathbf{s_T})$ là một chuỗi trạng thái nào đó, CRFs xác định xác suất điều kiện của một chuỗi trạng thái khi biết chuỗi quan sát như sau:

$$p_\theta(\mathbf{s} \mid \mathbf{o}) = \frac{1}{Z(\mathbf{o})} \exp\left[\sum_{t=1}^{T}\sum_{k} \lambda_k f_k(s_{t-1}, s_t, \mathbf{o}, t)\right]. \quad (1)$$

Gọi $Z(o) = \sum_{s'} \exp\left(\sum_{t=1}^{T}\sum_{k} \lambda_k f_k(s'_{t-1}, s'_t, o, t)\right)$ là thừa số chuẩn hóa trên toàn bộ các chuỗi nhãn có thể. f_k xác định một hàm đặc trưng và λ_k là trọng số liên kết với mỗi đặc trưng f_k. Mục đích của việc học máy với CRFs là ước lượng trọng số này. Ở đây, ta có hai đặc trưng f_k: đặc trưng trạng thái (per-state) và đặc trưng chuyển (transition).

$$f_k^{(per-state)}(s_t, \mathbf{o}, t) = \delta(s_t, l) x_k(\mathbf{o}, t). \quad (2)$$

$$f_k^{(transition)}(s_{t-1}, s_t, t) = \delta(s_{t-1}, l)\delta(s_t, l). \quad (3)$$

Ở đây δ là Kronecker-δ. Mỗi đặc trưng trạng thái (2) kết hợp nhãn l của trạng thái hiện tại s_t và một từ ngữ cảnh - một hàm nhị phân $x_k(o,t)$ xác định các ngữ cảnh quan trọng của quan sát **o** tại vị trí *t*. Một đặc trưng chuyển (3) biểu diễn sự phụ thuộc chuỗi bằng cách kết hợp nhãn l' của trạng thái trước s_{t-1} và nhãn l của trạng thái hiện tại s_t.

Người ta thường huấn luyện CRFs bằng cách cực đại hóa hàm likelihood theo dữ liệu huấn luyện sử dụng các kĩ thuật tối ưu như L-BFGS. Việc lập luận (dựa trên mô hình đã học) là tìm ra chuỗi nhãn tương ứng của một chuỗi quan sát đầu vào. Đối với CRFs, người ta thường sử dụng thuật toán quy hoạch động điển hình là Viterbi để thực hiện lập luận với dữ liệu mới.

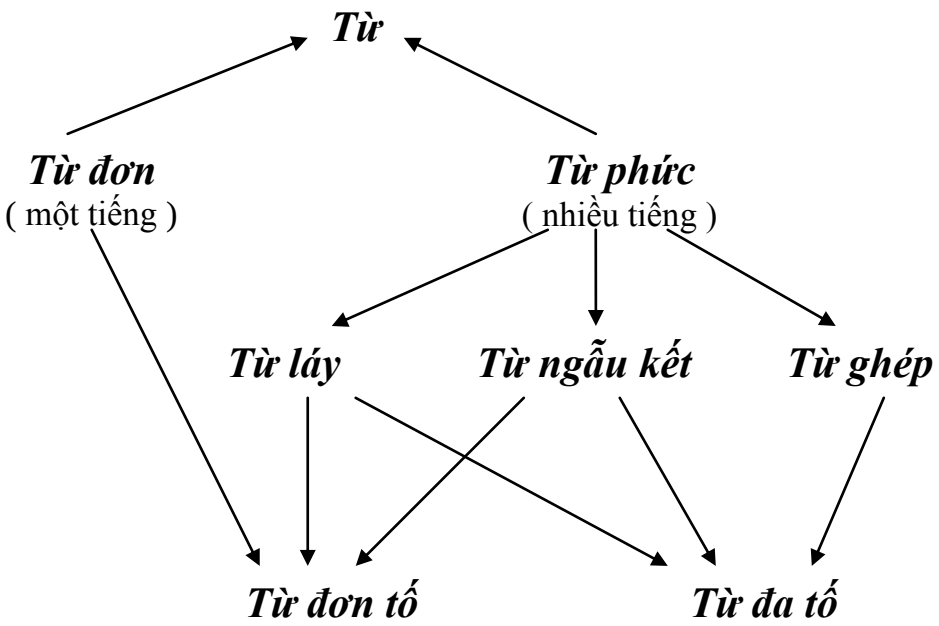
2.3 Ngữ pháp tiếng Việt

2.3.1 Cấu tạo và phân loại từ trong tiếng Việt

Trong tiếng Việt thì âm tiết là một thứ đơn vị ngữ âm học, nó là kết quả sự kết hợp một âm hay nhiều âm với một thanh điệu nào đó theo quy tắc tổ chức của ngữ âm tiếng Việt. Âm tiết hay còn gọi là *tiếng* là đơn vị cơ sở của cấu tạo ngữ pháp ngôn ngữ Việt. *Tiếng* có cấu tạo bằng một âm tiết và tham gia vào hệ thống ngôn ngữ với tư cách một thành tố trong các cơ chế cấu tạo từ (từ đơn, từ láy, từ ghép…).

Ngoài tiếng là đơn vị cơ sở của cấu tạo ngữ pháp Việt Nam, cần phải nhắc đến *từ,* cái có thể dùng làm đơn vị trung tâm của ngữ pháp Việt Nam. Là đơn vị nhỏ nhất mà có nghĩa và hoạt động tự do trong câu, từ chi phối toàn bộ cú pháp tiếng Việt, đảm nhiệm và san sẻ các chức năng cú pháp trong câu và góp phần đưa câu vào các cấu tạo ngôn ngữ lớn hơn câu. Từ có thể được xem xét từ những góc độ khác nhau: từ phía ngữ âm học, từ phía ngữ nghĩa, từ phía ngữ pháp học, từ phía cách sử dụng… Trong số đó, việc xem xét từ từ góc độ ngữ pháp học là xem xét phối hợp mặt ngữ âm và mặt ngữ nghĩa.

Thực trạng của từ tiếng Việt cho ta nhiều cách tiếp cận về cấu tạo ngữ pháp của chúng. Những cách tiếp cận này không bài xích lẫn nhau mà có tác dụng bổ sung cho nhau, giúp bao quát được toàn bộ vốn từ tiếng Việt. Nếu chỉ dừng lại ở một cách tiếp cận nào đó chúng ta sẽ vấp phải hoặc tính sơ lược hoặc sự cưỡng chết hiện thực ngôn ngữ. Nói cách khác, bản thân tính phức tạp của đối tượng nghiên cứu, tính mâu thuẫn nội tại của sự vật đòi hỏi sự phối hợp những cách tiếp cận chung và riêng có phần khác nhau để làm bộc lộ đối tượng đến mức độ cần thiết. Chúng ta có thể hình dung cấu tạo của từ tiếng Việt như trong lược đồ sau:



Lược đồ 2.1 : Phân loại từ trong tiếng Việt dựa trên cấu tạo từ

2.3.2 Cụm từ tiếng Việt

Từ kết hợp với từ một cách có tổ chức và có ý nghĩa làm thành những tổ hợp từ, tức là những kiến trúc lớn hơn từ. Mỗi từ trong tổ hợp từ là một thành tố. Tổ hợp từ có thể là một câu, có thể là một kiến trúc tương đương với câu nhưng chưa thành câu, cũng có thể là một đoạn có nghĩa của câu. Các tổ hợp từ chưa thành câu được gọi chung là *tổ hợp từ tự do*. Về nguyên tắc, tổ hợp từ tự do có thể chứa kết từ ở đầu để chỉ chức vụ ngữ pháp của toàn bộ phần còn lại trong tổ hợp từ này. Những tổ hợp từ có kết từ ở đầu như vậy mang tên là *giới ngữ*. Trái lại, tổ hợp từ tự do không chứa kết từ chỉ chưc vụ ngữ pháp như vậy được gọi là *cụm từ*. Vậy *cụm từ là những kiến trúc gồm hai từ trở lên kết hợp tự do với nhau theo những quan hệ ngữ pháp hiển hiện nhất định và không chứa kết từ ở đầu (để chỉ chức vụ ngữ pháp của kiến trúc này).*

Có thể phân loại các cụm từ trong tiếng Việt theo bảng sau:

Bảng 2.1. Nhãn cụm từ tiếng Việt

Tên	Chú thích
NP	Cụm danh từ
VP	Cụm động từ
ADJP	Cụm tính từ
ADVP	Cụm phó từ
PP	Cụm giới từ
QP	Cụm từ chỉ số lượng

a) Cụm danh từ

Cụm danh từ là tổ hợp từ tự do không có kết từ đứng đầu, có quan hệ chính phụ giữa thành tố chính với thành tố phụ, và thành tố chính là danh từ.

Cấu tạo chung của cụm danh từ gồm có 3 phần : phần trung tâm, phần phụ trước và phần phụ sau. Tại phần trung tâm thường là một danh từ hoặc một ngữ danh từ. Ngữ danh từ gồm một danh từ chỉ loại đứng trước và một danh từ chỉ sự vật hay một động từ, tính từ chỉ hoạt động, trạng thái, tính chất, quan hệ đứng sau, và cả hai cùng gộp lại để chỉ một sự vật. Trong phần phụ trước người ta đã xác định được ba vị trí khác nhau sắp xếp theo một trật tự nhất định. Ở phần phụ sau thường nhận được hai vị trí có trật tự ổn định. Phần phụ trước cụm danh từ thường dùng chỉ yếu tố số lượng của sự vật nêu ở trung tâm, phần phụ sau chủ yếu dùng chỉ yếu tố chất lượng của sự vật nêu ở thành phần trung tâm.

Phần phụ trước(-1,-2,-3)	Phân trung tâm (0)	Phần phụ sau (1,2)

Ví dụ :

tất cả	những	cái	con mèo	đen	ấy
-3	-2	-1	0	1	2

- vị trí 0 là vị trí của danh từ chính
- vị trí -1 là vị trí của từ chỉ xuất **cái**
- vị trí -2 là vị trí của từ chỉ số lượng : *một, hai…; vài, ba, dăm, dăm ba…; mỗi, từng, mọi…; những, các, một…; mấy…*
- vị trí -3 là vị trí của từ chỉ tổng lượng : *tất cả, hết thảy, cả…*
- vị trí 1 là vị trí của từ nêu đặc trưng miêu tả có thể gặp nhiều loại từ khác nhau như: danh từ, động từ, tính từ, số từ, đại từ và thời vị từ, ví dụ: *phòng **tạp chí**, phòng **đọc**, phòng **hẹp**, phòng **14**, phòng **ngoài**, phòng (của) **chúng tôi**, chuyến **trước**…*
- vị trí 2 là vị trí của từ chỉ định : *cái máy **này**, quả táo **kia**…*

b) Cụm động từ

Cụm động từ là tổ hợp từ tự do không có kết từ đứng đầu, có quan hệ chính phụ giữa thành tố chính với thành tố phụ, và thành tố chính là động từ.

Cấu tạo chung của cụm động từ gồm có 3 phần: phần trung tâm, phần phụ trước, phần phụ sau. Ở phần trung tâm có thể gặp một động từ hoặc những tổ hợp gồm nhiều động từ. Các thành tố phụ của cụm động từ có thể chia thành hai loại: thành tố phụ là các phụ từ và thành tố phụ là các thực từ. Thành tố phụ từ chuyên biểu thị mối quan hệ của hành động, trạng thái… nêu ở động từ - thành tố chính với thời gian và biểu thị cái thể trạng của hành động, trạng thái… Thành tố phụ thực từ có tác dụng mở rộng nội dung ý nghĩa của hành động, trạng thái…nêu ở động từ - thành tố chính, cụ thể là cho biết cách thức, môi trường không gian, thời gian, đối tượng chịu tác dụng của động từ làm thành tố chính hay tác động đến động từ làm thành tố chính.

Tại phần phụ trước cụm động từ, tập hợp chủ yếu loại thành tố phụ là phụ từ chỉ mối quan hệ với thời gian, tại phần phụ sau tập hợp chủ yếu các thành tố phụ thực

từ mở rộng nội dung động từ. Như vậy, có thể nói, về cơ bản phần phụ trước của cụm động từ có tác dụng định tính mối quan hệ về thời gian và thể trạng của hành động, trạng thái nêu ở động từ thành tố chính. Phần phụ sau về cơ bản có tác dụng mở rộng nội dung từ vựng của động từ.

c) Cụm tính từ

Cụm tính từ là tổ hợp từ tự do không có kết từ đứng đầu, có quan hệ chính phụ giữa thành tố chính với thành tố phụ, và thành tố chính là tính từ.

Cấu tạo chung của cụm tính từ cũng gồm có 3 phần: phần trung tâm, phần phụ trước, phần phụ sau.

Các thành tố phụ của cụm tính từ gồm có hai loại: thành tố phụ là phụ từ và thành tố phụ là thực từ.

Phần lớn những thành tố phụ là phụ từ xuất hiện ở cụm động từ đồng thời cũng có thể làm thành tố phụ trong cụm tính từ. Cụ thể như: *đã, sẽ, đang, vừa, cũng, đều, mới, vẫn, cứ, cùng*… với tư cách thành tố phụ trước; rồi với tư cách là thành tố phụ sau. Một vài thành tố phụ có tác dụng đánh dấu từ loại động từ, không thể xuất hiện hoặc chỉ xuất hiện với những điều kiện nhất định ở cụm tính từ như: *hãy, đừng,* (thành tố phụ trước), *đã* (thành tố phụ sau).

Article III. *Chương 3:* Chương trình thực nghiệm

3.1 Giới thiệu ngôn ngữ VB 6.0

3.1.1 Giới thiệu

Visual Basic 6.0 (VB) là một ngôn ngữ lập trình hướng đối tượng, trực quan trên môi trường Windows. VB cung cấp một bộ công cụ hoàn chỉnh để đơn giản hóa việc triển khai lập trình ứng dụng, có thể nói đây là cách nhanh và tốt nhất để học và lập trình ứng dụng trên Microsoft Windows.

Phần "Visual- Trực quan" đề cập đến phương pháp được sử dụng để tạo giao diện đồ họa người dùng (GUI - Graphical User Interface). VB có sẵn rất nhiều những bộ phận trực quan gọi là các điều khiển (Controls) mà người lập trình có thể sắp đặt vị trí và quyết định các đặc tính của chúng trên một khung giao diện màn hình, gọi là form. Việc thiết kế các giao diện người dùng ứng dụng trên VB có thể hình dung đơn giản như việc vẽ giao diện trên Word hoặc trên Paint Prush của Windows.

Phần "Basic" đề cập đến ngôn ngữ BASIC (Beginners All-Purpose Symbolic Instruction Code), một ngôn ngữ lập trình đơn giản, dễ học, được viết ra cho các khoa học gia- những người không có thì giờ để học lập trình điện toán sử dụng. Tuy nhiên, ngôn ngữ Basic trong VB đã được cải thiện rất nhiều để phù hợp với phong cách lập trình hiện đại.

3.1.2 Lập trình VB căn bản

Các thành phần liên quan đến lập trình căn bản trên VB.

- Các kiểu dữ liệu trong VB;
- Sử dụng biến và hằng;
- Các cấu trúc lập trình căn bản;
- Kỹ thuật chương trình con;
- Cách thức soạn thảo chương trình;
- Kỹ thuật bắt lỗi và xử lý lỗi trên VB.

3.1.3.1 Kiểu dữ liệu - biến và hằng

a. Kiểu dữ liệu

Cũng như các ngôn ngữ lập trình khác, VB đều hỗ trợ các kiểu dữ liệu cơ bản.

Dưới đây giới thiệu chi tiết về từng kiểu.

Boolean

Kiểu lô gíc, tương tự kiểu Boolean trên Pascal. Kiểu này chiếm 2 byte bộ nhớ; chỉ nhận một trong 2 giá trị là: Yes – No hoặc True – False hoặc đôi khi thể hiện dưới dạng số 0 tương đương với False, True tương ứng với bất kỳ số nào khác 0. Khi lập trình CSDL, kiểu Boolean tương ứng với kiểu Yes/No trong bảng dữ liệu.

Byte

Kiểu số nguyên dương trong phạm vi từ 0..255. Kiểu này chiếm 1 byte bộ nhớ.

Integer

Kiểu nguyên, có giá trị trong khoảng -32768...32767. Kiểu này chiếm 2 bytes bộ nhớ.

Long

Kiểu số nguyên dài, có giá trị trong khoảng 2,147,483,648 .. 2,147,483,647. Kiểu này chiếm 4 bytes bộ nhớ.

Single

Kiểu số thực, có giá trị trong khoảng 1.401298E-45 to 3.402823E38. Chiếm 4 bytes bộ nhớ.

Double

Kiểu số thực có độ lớn hơn kiểu Single, có giá trị trong khoảng 4.94065645841247E-324 to 1.79769313486232E308. Chiếm 8 bytes bộ nhớ.

Currency

Kiểu tiền tệ. Bản chất là kiểu số, độ lớn 8 bytes, có giá trị trong khoảng -922,337,203,685,477.5808 to 922,337,203,685,477.5807. Đặc biệt, kiểu này luôn có ký hiệu tiền tệ đi kèm.

String

Kiểu xâu ký tự. Kiểu này tương ứng với kiểu String trong Pascal, tương ứng với kiểu Text trong VB. Độ lớn tối đa 255 bytes tương đương với khả năng xử lý xâu dài 255 ký tự.

Variant

Variant là kiểu dữ liệu không tường minh. Biến kiểu này có thể nhận bất kỳ một giá trị nào có thể. Ví dụ:

Dim a As Variant

a = 123

a = “Nguyễn Văn Thành”

Hoàn toàn không có lỗi.

Người ta thường khai báo biến kiểu Variant trong những trường hợp phải xử lý biến đó mềm dẻo. Khi thì biến nhận giá trị kiểu này, khi thì nhận giá trị và xử lý theo kiểu dữ liệu khác.

Object

Object là một loại biến kiểu Variant, chiếm dung lượng nhớ 4 bytes, dùng để tham chiếu tới một loại đối tượng (Object) nào đó trong khi lập trình. Tất nhiên muốn khai báo biến Object kiểu nào, phải chắc chắn đối tượng đó đã được đăng ký vào thư viện tham chiếu VB bởi tính năng Project | Reference.

b. Biến

b.1. Biến – khai báo biến

Biến (Variable) là thành phần của một ngôn ngữ lập trình, giúp xử lý dữ liệu một cách linh hoạt và mềm dẻo.

Thông thường trong các ngôn ngữ lập trình, mỗi biến khi tồn tại phải được định kiểu, tức là phải nhận một kiểu dữ liệu xác định. Tuy nhiên trong VB thì không, mỗi biến có thể định kiểu (được khai báo trước khi sử dụng) hoặc không định kiểu (không khai báo vẫn sử dụng được). Trong trường hợp này biến đó sẽ tự nhận kiểu giá trị Variant.

Biến có thể được khai báo bất kỳ ở đâu trong phần viết lệnh của VB. Tất nhiên, biến có hiệu lực như khai báo chỉ bắt đầu từ sau lời khai báo và đảm bảo phạm vi hoạt động như đã qui định. Vì biến trong VB hoạt động rất mềm dẻo, nên có nhiều cách khai báo biến như:

Ví dụ 1: Khai báo biến i kiểu Integer

```
Dim i As Interger
```

Ví dụ 2: Khai báo biến i kiểu Integer, st kiểu String độ dài 15 ký tự

```
Dim i As Integer, st As String*15
```

b.2. Phạm vi biến

Như chúng ta đã biết, mỗi biến sau khi được khai báo nó sẽ nhận một kiểu dữ liệu và có một phạm vi hoạt động, tức là lời khai báo biến chỉ có tác dụng trong những vùng đã được chỉ định; ngoài vùng chỉ định đó biến sẽ không có tác dụng, nếu có tác dụng sẽ theo nghĩa khác (biến cục bộ kiểu Variant chẳng hạn).

Biến cục bộ:

Biến cục bộ được khai báo sau từ khoá Dim, nó chỉ có tác dụng trong một chương trình con, cục bộ trong một form hoặc một module nào đó. Dưới đây sẽ chỉ ra 3 trường hợp biến cục bộ này:

- Trong một chương trình con, nếu nó được khai báo trong chương trình con đó;

- Trong cả một Form, nếu nó được khai báo trong phần Decralations của Form đó;

- Trong cả một Reports, nếu nó được khai báo trong phần Decralations của Report đó;

- Trong cả một Modules, nếu nó được khai báo trong phần Decralations của Modules đó;

* Biến chỉ có tác dụng sau lệnh khai báo Dim

Biến toàn cục:

Biến toàn cục được khai báo sau cụm từ khoá Public, nó có tác dụng trong toàn bộ chương trình (ở bất kỳ chỗ nào có thể viết lệnh). Loại biến này luôn phải được khai báo tại vùng Decralations của một Module nào đó.

Ví dụ:

Public Hoten(45) As String * 45

Trên một dự án VB không được phép khai báo trùng tên biến toàn cục. Tuy nhiên tên biến cục bộ vẫn có thể trùng tên biến toàn cục, trong trường hợp đó VB sẽ ưu tiên sử dụng biến cục bộ trong phạm vi của nó.

c. Hằng

c.1. Khai báo hằng

Hằng (Constan) là đại lượng có giá trị xác định và không bị thay đổi trong bất kỳ hoàn cảnh nào. Tương ứng với từng kiểu dữ liệu, sẽ có những hằng tương ứng.

Khai báo hằng số bởi từ khoá Const. Sau đây là các ví dụ về khai báo các loại hằng:

Ví dụ 1: Hằng a =5 (hằng số)

Const a = 5

Ví dụ 2: Hằng ngày = 24/12/2004 kiểu Date (bao bởi cặp dấu thăng #..#)

Const ngay = #24/12/2004#

Ví dụ 3: Hằng xâu ký tự (bao bởi cặp dấu nháy kép “..”)

Const phongban = "Tài vụ"

Ví dụ 4: Hằng kiểu Lôgíc xác định bởi True hoặc False

Const ok = True

c.2. Phạm vi hằng

Tương tự như biến, hằng cũng có những phạm vi hoạt động của nó. Hằng được khai báo trong thủ tục nào, hoặc cục bộ trong form, report hoặc module nào sẽ chỉ có tác dụng trong phạm vi đó.

Muốn hằng có phạm vi toàn cục, phải được khai báo sau từ khoá Public Const, tại vùng Decralations của một module nào đó như sau:

Public Const a = 12

3.1.3.2 Các cấu trúc lệnh VB

Các cấu trúc lệnh là thành phần cơ bản của mỗi ngôn ngữ lập trình. Thông thường các ngôn ngữ lập trình đều có các cấu trúc lệnh như nhau: lệnh xử lý điều kiện, lệnh lặp biết trước số vòng lặp, lệnh lặp không biết trước số vòng lặp,.. Tuy nhiên cách thể hiện (cú pháp) mỗi cấu trúc lệnh có thể khác nhau tuỳ thuộc vào mỗi ngôn ngữ lập trình. Hơn nữa, mỗi ngôn ngữ cũng có thể có một số điểm khác biệt, đặc trưng trong mỗi cấu trúc lệnh.

Cũng giống như nhiều ngôn ngữ lập trình hiện đại khác, các cấu trúc lệnh trong VB đều tuân thủ các nguyên tắc:

- Có cấu trúc: mỗi cấu trúc lệnh đều có từ khoá bắt đầu và một từ khóa báo hiệu kết thúc;
- Thực hiện tuần tự (loại trừ trường hợp đặc biệt thủ tục Goto <Label>);
- Có khả năng lồng nhau;

a. Cấu trúc IF… END IF

Cấu trúc này thường gọi là lệnh lựa chọn. Tức là nếu một điều kiện nào đó xảy ra sẽ là gì, hoặc trái lại có thể làm gì. Trong VB cú pháp lệnh này như sau:

If <điều kiện> Then
<thủ tục 1>
[Else
<thủ tục 2>]
End If

b. Cấu trúc SELECT CASE .. END SELECT

Đây là một loại của cấu trúc lựa chọn. Thông thường hoàn toàn có thể sử dụng If .. End If để thực hiện các xử lý liên quan đến kiểu cấu trúc này, nhưng trong những trường hợp đặc biệt, cấu trúc Select Case .. End Select thể hiện được sự tiện dụng vượt trội. Trong VB cú pháp lệnh này như sau:

Select Case <biểu thức>
  Case <giá trị 1>
   <thủ tục 1>
  Case <giá trị 2>
   <thủ tục 2>
  ……
  Case <giá trị n>
  <thủ tục n>
 [Case Else
   <thủ tục n+1>]
End Select

Trong đó: <Biểu thức> luôn trả về giá trị kiểu vô hướng đếm được như: số nguyên, xâu ký tự, kiểu lô gíc,..

c. Cấu trúc FOR ... NEXT

For… Next là một cấu trúc lặp biết trước số lần lặp trong VB, tuy nhiên trong những tình huống đặc biệt, vẫn có thể sử dụng cấu trúc này như cấu trúc không biết trước được số lần lặp.

Cú pháp cấu trúc For…Next như sau:

For <biến chạy> = <giá trị 1> To <giá trị 2> [Step <n>]
 <thủ tục>
 [Exit For]
Next

d. Cấu trúc WHILE ... WEND

While … Wend là một cấu trúc lặp không biết trước số lần lặp trong VB. Cú pháp cấu trúc While…Wend như sau (Wend - viết tắt của cụm từ While End):

While <điều kiện>
 <thủ tục>
Wend

3.1.3.3 Các hàm xử lý chuỗi trong Vb6

Space (Num as Long) trả về chuỗi chỉ toàn khoảng trống với số khoảng trống được ấn định bởi tham số Num.

String (Num as Long, character) 'trả về một chuõi (theo dạng variant) gồm các ký tự lặp lại. Ký tự lặp lại là ký tự đầu của biểu thức chuỗi được truyền ở tham số thứ hai của hàm (character). Tham số thứ nhất (Num) xác nhận số lần lặp lại.

Trim (String) ' cắt các khoảng trống ở 2 đầu chuỗi

Len ( ) ' trả về chiều dài của chuỗi bao gồm các khoảng trống và các ký tự.

Mid (string, start as Long, length) ' trích từ tham số 1(string) một chuỗi ở vị trí bắt đầu được xác định bởi tham số 2(start), với số ký tự được qui định bởi tham số 3(length). Nếu bỏ qua tham số length thì hàm Mid sẽ trích đến hết chuỗi **InStr (start, string1, string2, compare)** ' trả về vị trí bắt đầu của một chuỗi con cần tìm trong một chuỗi mẹ. tham số 1(start) xác định vị trí bắt đầu tìm, tham sô 2(string1) là chuỗi mẹ, tham số 3(string2) là chuỗi cần tìm, tham số 4(compare) mặc định là so sánh nhạy ký tự. Khi bỏ qua tham số thứ nhất thì vị trí bắt đầu tìm mặc định là 1 **InStrRev (StringCheck as string, StringMatch as string, Start as Long, Compare)** ' chức năng như InStr nhưng InStrRev hoạt động ngược lại từ cuối chuỗi và cú pháp khác hơn. Cả hai hàm đều là hàm tìm kiếm nhạy ký tự nên cần chú ý chữ thường và chữ HOA. InStrRev thường kết hợp với Mid để tách một tên File khỏi đường dẫn và tên mở rộng.

Left (String, Length as Long)' trích từ đầu một chuỗi của tham số 1(String) với số lượng xác định bởi tham số 2 (Length).

Right (String, Length as Long)' như Left nhưng trích ngược từ cuối chuỗi

Replace (Expression as string, Find as string, Replace as string, start, count, compare) ' tìm trong tham số thứ 1(Expression) một chuỗi xác định bởi tham số 2(Find) và thay thế bằng một chuỗi được đặt ở tham số 3(Replace). Ba tham số còn lại là tùy chọn. Start qui định vị trí bắt đầu tìm chuỗi cần được thay, nếu bỏ qua mặc

định là 1. Count qui định số lần thay thế trong chuỗi, nếu bỏ qua mặc định Replace sẽ tìm và thay thế cho đến hết chuỗi.

StrComp (String1, String2, Compare) ' dùng để so sánh 2 chuỗi. Trị trả về: (String1 < String2) = -1; (String1 = String2) = 0; (String1 > String2) = 1

Like ' so sánh 2 chuỗi cho phép sử dụng biệt ngữ (như dùng ký tự đại diện trong Dos) giá trị trả về = True nếu tương hợp

Chú ý hàm Like mặc định cũng là hàm nhạy ký tự, theo thiết lập Option Compare ở form hoặc module.

Chr(charcode as Long)' Chuyển mã Ascii thành ký tự

Asc(String as String)' trả về mã Ascii của ký tự.

ChrW(charcode)' Chuyển mã Áscii thành ký tự (Hỗ trợ Unicode)

AscW(string)' Chuyển ký tự thành mã Ascii (hỗ trợ Unicode)

Join (SourceArray, Delimiter) ' tạo chuỗi mới từ một mảng chuỗi (SourceArray) với các phần tử được phân định bởi tham số Delimit

Split (Expression as String, Delimiter, Count, Compare) ' tạo mảng chuỗi từ một chuỗi (Expression). Đặt tham số Delimiter để chuyên biệt chỗ ngắt, nếu bỏ qua tham số này mặc định Split sẽ tách tại các khoảng trống của chuỗi. Tham số Count quy định số lần tách. Ba tham số cuối là tùy chọn.

Filter (sourcearray, match [, include [, compare]]) ' Lọc mảng sourcesrray với giá trị lọc là match ; include: Lọc đảo (True hoặc False) ; compare: chỉ rõ kiểu dữ liệu để so sánh trong quá trình lọc, dùng cho tham số compare.

vbUseCompareOption = –1 : Chế độ tùy chọn,

vbBinaryCompare = 0: So sánh nhị phân.

vbTextCompare = 1: So sánh chuỗi.

vbDatabaseCompare = 2: So sánh dữ liệu.

StrReverse(expression as String)' Đảo chuỗi expression

3.2 Chương trình thực nghiệm

3.2.1 Giới thiệu chương trình

Xử lý ngôn ngữ tự nhiên là một lĩnh vực nghiên cứu nhằm giúp cho các hệ thống máy tính hiểu và xử lý được ngôn ngữ con người. Dịch máy là một trong những ứng dụng chính của xử lý ngôn ngữ tự nhiên. Mặc dù dịch máy đã được nghiên cứu và phát triển từ lâu, nhưng hiện nay việc dịch máy áp dụng cho cặp ngôn ngữ Việt-Anh vẫn còn nhiều khó khăn và hạn chế. Do sự khác biệt lớn cả về mặt cú pháp và ngữ nghĩa của cặp ngôn ngữ này. Chương trình mô phỏng từ điển Việt - Anh và ứng dụng trong việc dịch cụm từ Việt - Anh được xây dựng nhằm mục đích :

- **Cung cấp kho từ vựng Việt - Anh có cấu trúc phục vụ cho việc dịch cụm từ Việt - Anh.**
- **Mô phỏng việc dịch cụm từ Việt - Anh.**

Các chức năng chính của chương trình :

- **Tra từ :** Khi người dùng gõ từ cần tra vào ô từ, chương trình sẽ liệt kê các từ có thể là từ người dùng cần tra hiển thị trong ô danh sách từ, người dùng chỉ việc click vào từ muốn tra trong ô danh sách từ. Các thông tin về nghĩa và từ loại sẽ được hiển thị trong ô nghĩa.
- **Thêm từ** : Người dùng có thể thêm một từ mới không có trong từ điển.
- **Sửa từ** : Người dùng có thể sửa nghĩa và từ loại của một từ.
- **Gán nhãn từ :** Người dùng nhập vào một câu, chương trình sẽ tách câu thành các từ và gán cho các nhãn của từ.
- **Dịch cụm từ** : Người dùng nhập vào một cụm từ tiếng Việt, chương trình sẽ dịch cụm từ tương ứng sang tiếng Anh.

3.2.2 Xây dựng chương trình

3.2.2.1 Từ điển Việt - Anh

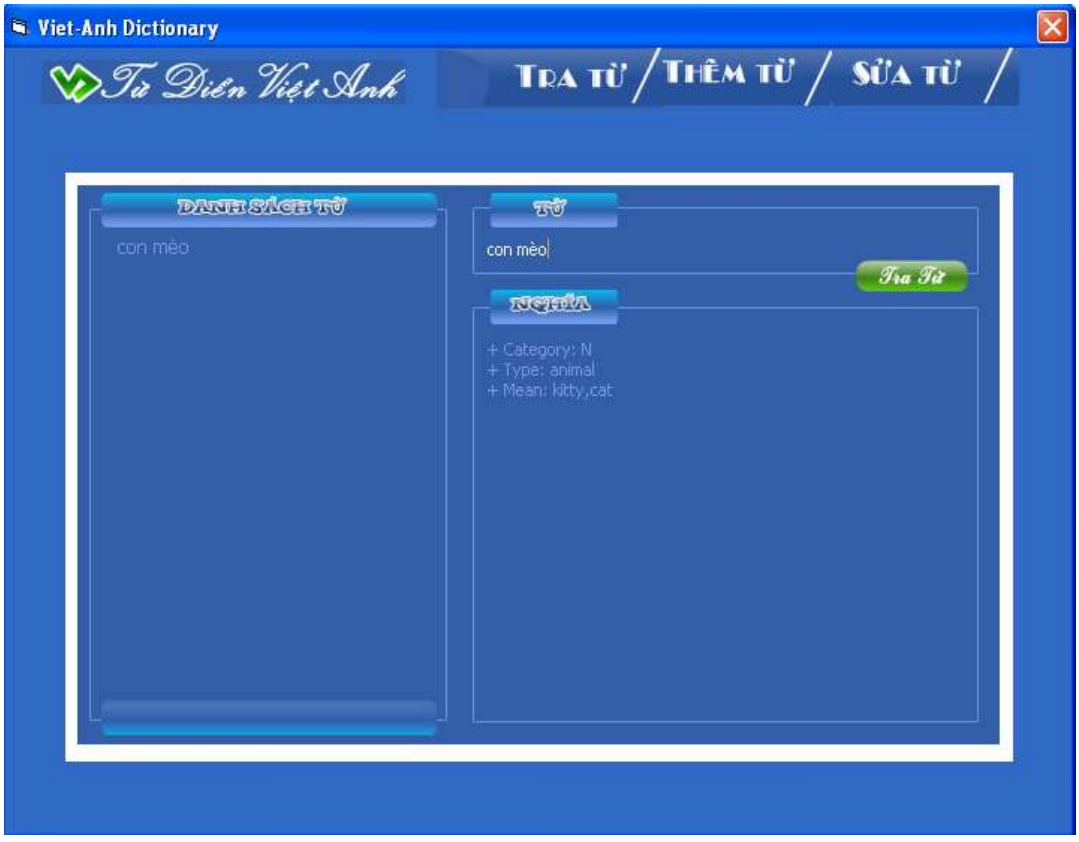
Cấu trúc từ điển Việt - Anh

- Một mục từ trong từ điển gồm các trường : word (dạng viết của từ), category (từ loại của từ), type (phạm trù ngữ nghĩa của từ), sematic (ngữ nghĩa của từ).

- Từ điển được tổ chức thành 27 bảng tương ứng với các chữ cái đầu tiên của từ trong tiếng Việt. Việc chia thành 27 bảng nhằm mục đích : tăng khả năng lưu trữ của từ điển và giúp tốc độ tìm kiếm từ nhanh hơn.

Các chức năng chính của từ điển

a) Tra từ



Hình 3.1 Kết quả tra từ

+) **Input** : từ cần tra nghĩa

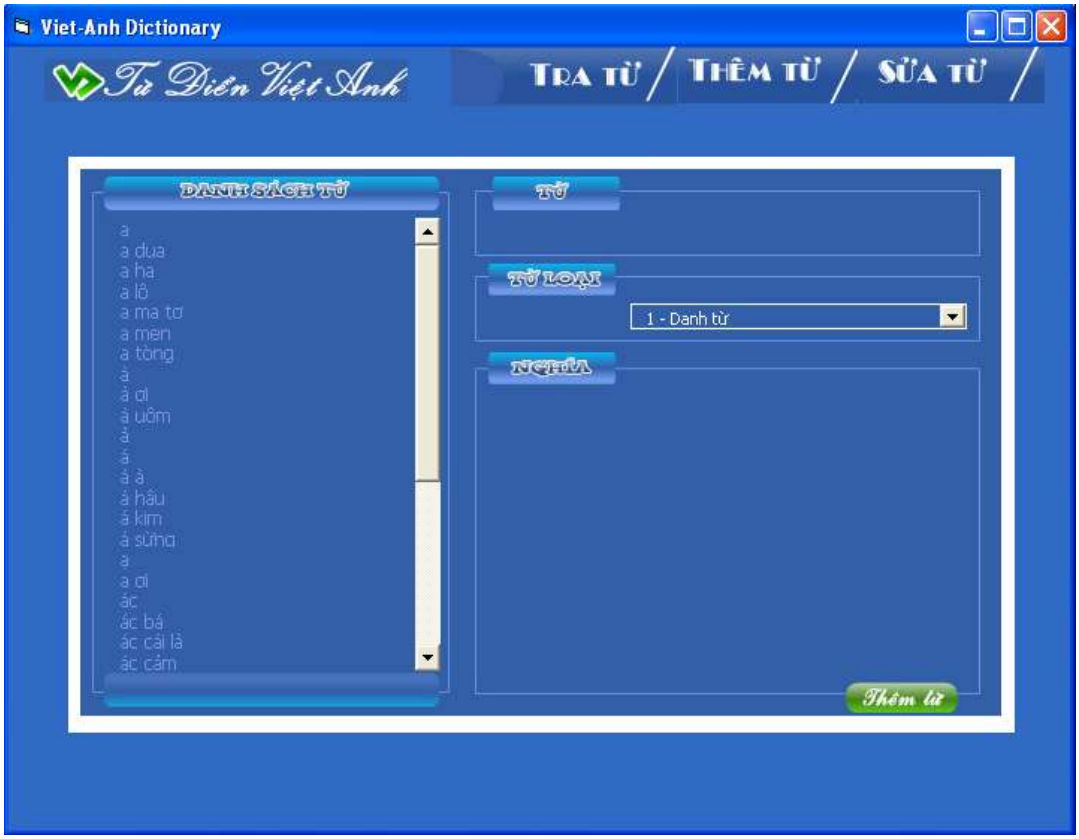
+) **Process** :

```
Private Sub cmd_tim_Click()
txt_nghia1.Text = ""
If txt_tu1.Text = "" Then
MsgBox "Bạn chưa nhập từ", vbInformation, "Warning"
GoTo nhan
End If
Dim tb, kt, sql As String
kt = laykitudau(txt_tu1.Text)
tb = tenbang(kt)
sql = "select * from " + tb + " where word like '" & txt_tu1.Text & "'"
Set rs = New ADODB.Recordset
rs.Open sql, conn
If rs.EOF = False Then
If Not rs.Fields(1).Value = "" Then
txt_nghia1.Text = "+ Category: " + rs.Fields(1).Value
End If
If Not rs.Fields(2).Value = "" Then
txt_nghia1.Text = txt_nghia1.Text + vbCrLf + "+ Type: " +
rs.Fields(2).Value
End If
If Not rs.Fields(3).Value = "" Then
txt_nghia1.Text = txt_nghia1.Text + vbCrLf + "+ Mean: " +
rs.Fields(3).Value
End If
```

Else: MsgBox txt_tu1.Text + vbCrLf + "Không có trong csdl",
vbInformation, "Thông báo"
End If
nhan: txt_tu1.SetFocus
End Sub

+) **Output** : Nghĩa của từ tương ứng trong tiếng Anh

b) Thêm từ



Hình 3.2 Form thêm từ

+) **Input** : từ cần thêm vào từ điển

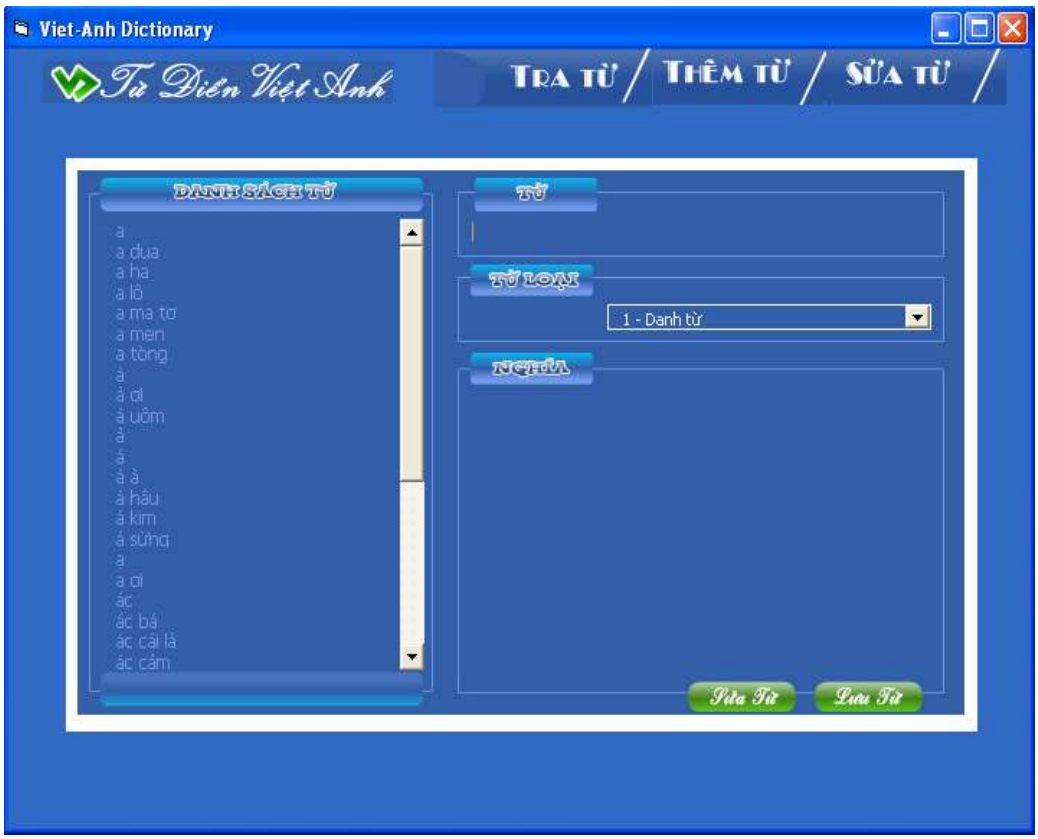
+) **Process** :

```
Private Sub cmd_them_Click()
Dim st, strnew, tuloai As String
st = Left(cmb_loai.Text, 1) ' lay ID cua tuloai
Set rs = New ADODB.Recordset
rs.Open "select * from tb_tuloai where ID like '" + st + "'", conn
If rs.EOF = False Then tuloai = rs.Fields(2)
strnew = Replace(txt_tu2.Text, "'", "", 1, -1, vbTextCompare)
strnew = Trim$(strnew)
If txt_tu2.Text = "" Then loi = 1
GoTo baoloi
ElseIf cmb_loai.Text = "" Then loi = 2
GoTo baoloi
ElseIf txt_nghia2.Text = "" Then loi = 3
GoTo baoloi
End If
Dim tb, kt, sql As String
kt = laykitudau(strnew)
tb = tenbang(kt)
sql = "select * from " + tb + " where word like '" + strnew + "'"
Set rs = New ADODB.Recordset
rs.Open sql, conn
```

```
If rs.EOF = False Then loi = 4
   GoTo baoloi
Else
Set rs = New ADODB.Recordset
   rs.Open sql, conn, adOpenDynamic, adLockOptimistic
   rs.AddNew
   rs.Fields(0).Value = strnew
   rs.Fields(1).Value = tuloai
   rs.Fields(3).Value = Trim$(txt_nghia2.Text)
   rs.Update
   rs.Close
End If
MsgBox "Thông tin đã được lưu trong csdl", vbInformation, "Thông báo"
Exit Sub
baoloi:
If loi = 1 Then MsgBox "Hãy nhập từ cần thêm", vbInformation, "Thông
báo"
If loi = 2 Then MsgBox "Hãy chọn từ loại", vbInformation, "Thông báo"
If loi = 3 Then MsgBox "Hãy nhập nghĩa của từ", vbInformation, "Thông
báo"
If loi = 4 Then MsgBox "Từ " + "[" + strnew + "]" + " đã có trong csdl",
vbInformation, "Thông báo"
End Sub
```

+) **Output** : thông báo thêm từ thành công và hiển thị.

c) Sửa từ



Hình 3.3 Form sửa từ

+) **Input :** từ cần sửa

+) **Process :**

```
Private Sub cmd_luusua_Click()
Dim st, strnew, tuloai As String
st = Left(cmb_loai.Text, 1) ' lay ID cua tuloai
Set rs = New ADODB.Recordset
rs.Open "select * from tb_tuloai where ID like '" + st + "'", conn
If rs.EOF = False Then tuloai = rs.Fields(2)
strnew = Replace(txt_tu2.Text, "'", "", 1, -1, vbTextCompare)
strnew = Trim$(strnew)
```

```
If txt_tu2.Text = "" Then loi = 1
GoTo baoloi
ElseIf cmb_loai.Text = "" Then loi = 2
GoTo baoloi
ElseIf txt_nghia3.Text = "" Then loi = 3
GoTo baoloi
End If
Dim tb, kt, sql As String
kt = laykitudau(strnew)
tb = tenbang(kt)
sql = "select * from " + tb + " where word like '" + strnew + "'"
   Set rs = New ADODB.Recordset
   rs.Open sql, conn, adOpenDynamic, adLockOptimistic
   rs.Fields(0).Value = strnew
   rs.Fields(1).Value = rs.Fields(1).Value + "," + tuloai
   rs.Fields(3).Value = rs.Fields(3) + "," + Trim$(txt_nghia3.Text)
   rs.Update
   rs.Close
MsgBox "Thông tin sửa đã được lưu trong csdl", vbInformation, "Thông
báo"
baoloi:
If loi = 1 Then MsgBox "Hãy nhập từ cần thêm", vbInformation, "Thông
báo"
If loi = 2 Then MsgBox "Hãy chọn từ loại", vbInformation, "Thông báo"
If loi = 3 Then MsgBox "Hãy nhập nghĩa của từ", vbInformation, "Thông
báo"
End Sub
```

+) **Output :** thông báo thông tin được sửa thành công.

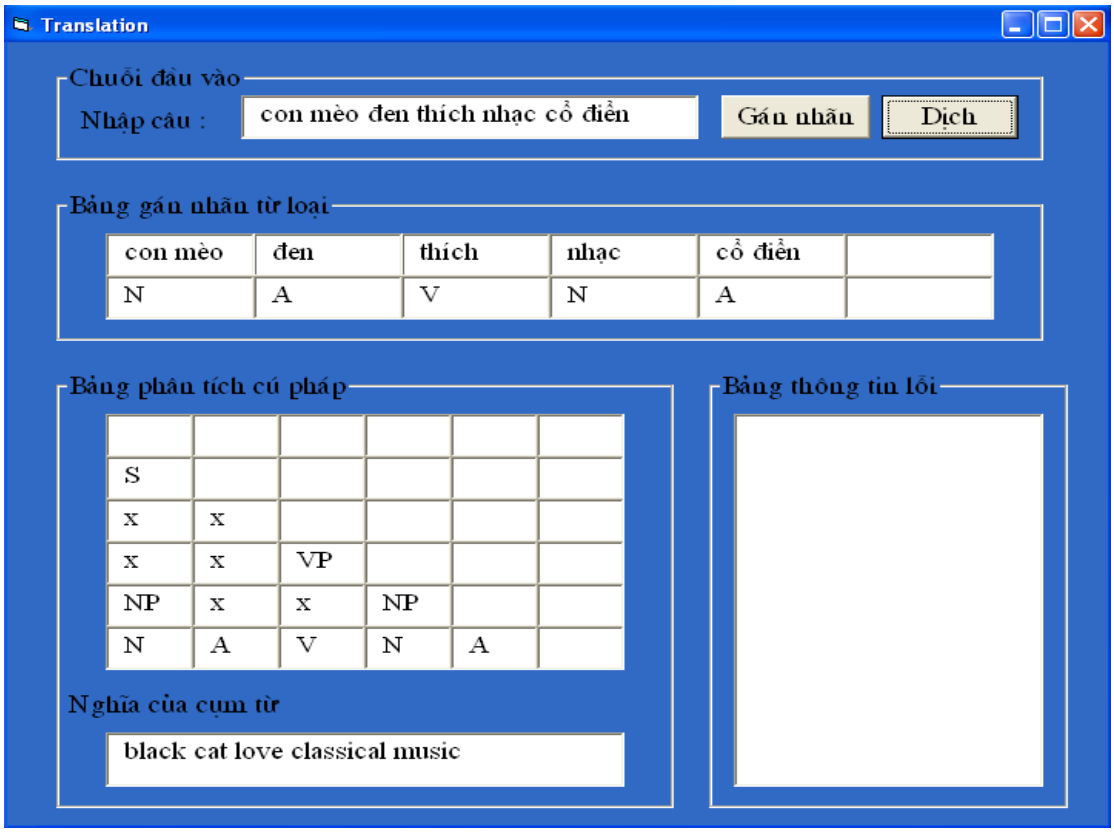
3.2.2.2 Dịch cụm từ Việt - Anh

Các bước thực hiện :

+) Tách từ và gán nhãn từ loại : chương trình tách cụm từ nhập vào thành các từ đơn và từ ghép có nghĩa trong tiếng Việt, sau đó tìm từ loại tương ứng của các từ trong bảng cơ sở dữ liệu và gán nhãn cho các từ đó.

+) Phân tích cú pháp: cụm từ nhập vào sau khi được gán nhãn sẽ được phân tích xem có đúng cú pháp không, có phải là cụm từ đúng trong tiếng Việt không bằng thuật toán CYK (Cocle-Younger-Kasami).

+) Dịch cụm từ: cụm từ sau khi được phân tích cú pháp sẽ được dịch sang tiếng Anh bằng cách tìm các từ tương ứng trong tiếng Anh và sắp xếp theo trật từ cú pháp đúng trong tiếng Việt.



Hình 3.4 Kết quả gán nhãn và dịch cụm từ

a) Tách từ và gán nhãn từ loại

+) Input : cụm từ tiếng Việt

+) Process :

```
Private Sub cmd_nhan_Click()
Dim str() As String
Dim targ(10), tuloai(10) As String
Dim i, k, j, sotu As Integer
If txt_string.Text = "" Then
MsgBox "Bạn chưa nhập từ", vbInformation, "Thông báo"
End If
str = tachtu(txt_string.Text, sotu)
'Lay tu loai tuong ung voi tu trong csdl
Dim tb, kt, sql As String
For i = 0 To sotu - 1
kt = laykitudau(str(i))
tb = tenbang(kt)
sql = "select * from " + tb + " where word like '" & str(i) & "'"
Set rs = New ADODB.Recordset
rs.Open sql, conn
If rs.EOF = False Then
If rs.Fields(1).Value <> "" Then
tuloai(i) = rs.Fields(1).Value
End If
End If
Next
'Hien thi thong tin ve tu và tu loai
For i = 0 To sotu - 1
Select Case i
```

```
Case 0:
txt_0.Text = str(i)
txt_nhan0.Text = tuloai(i)
GoTo nhan
Case 1:
txt_1.Text = str(i)
txt_nhan1.Text = tuloai(i)
GoTo nhan
Case 2:
txt_2.Text = str(i)
txt_nhan2.Text = tuloai(i)
GoTo nhan
Case 3:
txt_3.Text = str(i)
txt_nhan3.Text = tuloai(i)
Case 4:
txt_4.Text = str(i)
txt_nhan4.Text = tuloai(i)
Case 5:
txt_5.Text = str(i)
txt_nhan5.Text = tuloai(i)
GoTo nhan
End Select
nhan:
Next
End Sub
```

+) **Output :** bảng gán nhãn từ loại

b) Dịch cụm từ Việt – Anh

+) Input : cụm từ tiếng Việt

+) Process :

```
Private Sub cmd_dich_Click()
Dim str() As String
Dim targ(10), tuloai(10), tem As String
Dim i, k, j, sotu As Integer
txt_dich.Text = ""
str = tachtu(txt_string.Text, sotu)
'Lay tu và tu loai tuong duong trong csdl
Dim tb, kt, sql As String
k = 0
j = 0
For i = 0 To sotu - 1
kt = laykitudau(str(i))
tb = tenbang(kt)
sql = "select * from " + tb + " where word like '" & str(i) & "'"
Set rs = New ADODB.Recordset
rs.Open sql, conn
If Not rs.EOF = True Then
If rs.Fields(3).Value <> "" Then
targ(k) = rs.Fields(3).Value
k = k + 1
End If
```

```
If rs.Fields(1).Value <> "" Then
tuloai(j) = rs.Fields(1).Value
j = j + 1
End If
Else: MsgBox "Từ " + str(i) + " Không có trong từ điển ", vbInformation,
” Thông báo ”
End If
Next
'Sap xep cau nguon theo trat tu cau dich
j = k - 1
For i = 0 To k - 1
If targ(k - 1) = "i" Or targ(k - 1) = "I" Then
targ(k - 1) = "my"
End If
txt_dich.Text = txt_dich.Text + " " + targ(j)
j = j - 1
Next
End Sub
```

+) Output : cụm từ tiếng Anh tương ứng

3.2.3 Hạn chế và hướng phát triển của đề tài

a) Hạn chế

- Chương trình chỉ mới dịch được cụm từ và những câu đơn giản.
- Chương trình chưa thể xử lý được nhập nhằng trong quá trình dịch.

b) Hướng phát triển

- Xây dựng bộ từ điển tiếng Việt hoàn chỉnh.
- Thống kê đầy đủ tập luật cho ngôn ngữ tiếng Việt.
- Xây dựng hệ thống dịch tự động.

KẾT LUẬN

Trong quá trình nghiên cứu, tìm hiểu và hoàn thành đồ án tốt nghiệp " Tìm hiểu về xử lý ngôn ngữ tự nhiên và máy dịch. Viết chương trình mô phỏng từ điển Việt-Anh ", em đã thu nhận được thêm những kiến thức và em cũng nhận thấy xử lý ngôn ngữ tự nhiên là một lĩnh vực nghiên cứu rộng lớn, còn nhiều điều cần phải khám phá.

Trong đề tài em đã cố gắng tập trung tìm hiểu và nghiên cứu tổng quan về xử lý ngôn ngữ tự nhiên, một số thuật toán phân tích cú pháp và em cũng đã tìm hiểu phương pháp gán nhãn từ loại, các phương pháp dịch máy. Từ đó em đã xây dựng được chương trình mô phỏng dịch cụm từ từ tiếng Việt sang tiếng Anh.

Do thời gian thực hiện đồ án hạn chế nên em mới chỉ tìm hiểu được một số bước trong quá trình xử lý ngôn ngữ tự nhiên và chương trình mô phỏng còn chưa được hoàn thiện như mong muốn. Trong thời gian tới em sẽ cố gắng tiếp tục nghiên cứu và hoàn thiện việc tìm hiểu xử lý ngôn ngữ tự nhiên và chương trình mô phỏng dịch cụm từ Anh-Việt này.

TÀI LIỆU THAM KHẢO

[1] Diệp Quang Ban, Hoàng Văn Thung, *Ngữ pháp tiếng Việt (2 tập)*, Nhà xuất bản Giáo dục, 1999.

[2] Đinh Điền, *Giáo trình xử lý ngôn ngữ tự nhiên*, Đại học Khoa Học Tự Nhiên Tp.HCM, 12/2004.

[3] TS Lê Anh Cường, *Bài giảng Xử lý ngôn ngữ tự nhiên*, Đại học Công Nghệ - ĐHQG Hà Nội, 8/2007.

[4] TS. Nguyễn Chí Hiếu, *Ứng dụng xử lý ngôn ngữ tự nhiên trong dịch máy*, Khoa CNTT - ĐH Công Nghiệp Tp.HCM.

[5] Nguyễn Thị Mình Huyền - Đại học Khoa học Tự nhiên Hà Nội, Vũ Xuân Lương - Trung tâm từ điển học Vietlex, *Nghiên cứu và xây dựng từ điển tiếng Việt cho máy tính* .

[6] Phan Xuân Hiếu - Đại học Tohoku, Nhật Bản, Lê Minh Hoàng - Đại học Sư Phạm Hà Nội, Nguyễn Cẩm Tú - Đại học Công nghệ, ĐHQG Hà Nội, *Gán nhãn từ loại tiếng Việt dựa trên các phương pháp học máy thống kê* .

[7] Nguyễn Thị Minh Huyền, Vũ Xuân Lương, Lê Hồng Phương, *Sử dụng bộ gán nhãn từ loại xác suất QTAG cho văn bản tiếng Việt*, ICT 2003.

PHỤ LỤC 1

Cây ngữ nghĩa

a.CONCRETETHING - VẬT THỂ

a1.LivingThing - Vật hữu sinh

a1a.People - Con người

a1a01.**Person - Người** : ông, bà, cha, mẹ, giáo viên, học sinh, công nhân, binh lính

a1a02.**Organization - Tổ chức** : nhà trường, chi đoàn, chi uỷ, bộ ngoại giao

a1a03.**SupernaturalBeing - Nhân vật siêu nhiên** : phù thuỷ, thần linh, Ngọc Hoàng

a1b.Animal - Động vật

a1b01.**Vertebrate - Động vật có xương sống**

a1b01a.**Warm Blood - Động vật máu nóng**

a1b01a1.Mammal - Thú (ĐV có vú) : chó, sư tử, hổ, báo

a1b01a2.Bird - Chim : gà, vịt, cò, vạc, đại bàng, diều hâu

a1b01b.**Cold Blood - Động vật máu lạnh**

a1b01b1.Fish - Cá : chim, thu, nhụ, đé

a1b01b2.Amphibian - Lưỡng cư : ếch, nhái, ễnh ương

a1b01b3.Reptile - Bò sát : rắn, rết, thằn lằn, cá sấu

a1b02.**Invertebrate - Động vật không xương sống**

a1b02a.**Worm - Sâu** : sâu que, giun, sán

a1b02b.**Insect - Côn trùng** : kiến, ong, ve, bọ xít

a1b02c.**Shellfish - Động vật có vỏ** : tôm, cua, ghẹ

a1b02d.**OtherSeaCreature - Sinh vật biển** : bào ngư, sao biển

a1c.FictionalAnimal - Động vật hư cấu

List: rồng, thuồng luồng, ma cà rồng, ma gà

a1d.Microorganism - Vi sinh vật

List: vi rút, vi khuẩn, vi trùng

a1e.Plant -Thực vật

a1e01.**Tree - Cây cối** : lim, gụ, táu, phi lao, bạch đàn

a1e02.**Bush - Cây bụi** : duối, cúc tần, sim

a1e03.**Vine - Cây thân leo** : tơ hồng, tầm xuân

a1e04.**Herb - Cây thân cỏ** : cỏ tranh, lau, cói

a1e05.**Low Plant - Thực vật bậc thấp** : tảo, nấm

a1e06.**Hight Plant - Thực vật bậc cao** : dương xỉ, rong, rêu

a2.Non-livingThing - Vật vô sinh

a2a.Food - Thức ăn

a2a01.**Grain - Hạt** : đậu, ngô, lạc, vừng, (hạt) điều

a2a02.**Vegetable - Rau củ** : cải, rau muống, mùng tơi, bầu, bí, khoai tây

12a03.**Food - Lương thực** : lúa, ngô, khoai, sắn, kê, vừng, đại mạch

a2a04.**FoodStuff - Thực phẩm** : thịt, lòng, ba chỉ, thăn, mỡ phần, trứng, cá, sữa, bơ

a2a05.**Fruit - Quả** : cam, quýt, mít, bưởi, dừa, lê

a2a06.**Beverage - Đồ uống** : bia, rượu, sinh tố, nước ngọt

a2a07.**Seasoning - Gia vị** : mắm, muối, tiêu, ớt, mù tạt, rau thơm

a2a08.**Dish - Món ăn** : bánh kẹo, cơm, cháo, bún, phở, súp, bánh cuốn, lẩu, canh

a2b.Artifact - Vật dụng

a2b01.**Furniture - Đồ dùng** : bàn, ghế, giường, nồi, dao, kéo, bút, vở, đồng hồ

a2b02.**Tool - Công cụ** : cày, bừa, cuốc, xẻng, cưa, đục, búa, kìm, khoan, quang gánh

a2b03.**Garment - Trang phục** : quần, áo, giày, dép

a2b04.**Ornament - Trang sức** : nhẫn, vòng, lắc, hoa tai

a2b05.**Cosmetic - Mĩ phẩm** : son, phấn, dầu gội, kem

a2b06.**Drug - Thuốc men** : kí ninh, morphine, vitamin, cao hổ cốt, sâm, cam thảo

a2b07.**Plaything - Đồ chơi** : búp bê, bóng bay, cầu tụt, bập bênh

a2b08.**Vehicle - PT giao thông, vận tải** : ô tô, xe máy, xe đạp, máy bay, tàu thuỷ

a2b09.**Equipment - Thiết bị** : acquy, công tơ, băng chuyền, rơ moóc

a2b10.**Accessory - Phụ kiện** : van, vòi nước, ốc, đai, săm, lốp, xích, líp

a2b11.**Machine - Máy móc** : máy nổ, máy bơm, máy kéo, máy xay, máy biến thế

a2b12.**Instrument - Nhạc cụ** : đàn, sáo, trống, kèn, nhị, phèng la

a2b13.**Container - Đồ chứa** : thùng, téc, bồn, bể

a2b14.**Creation - Vật sáng tạo** : tranh ảnh, tượng, tác phẩm, vở kịch, bộ phim

a2b15.**Building - Công trình xây dựng** : nhà cửa, đình chùa, cầu cống, thành quách

a2b16.**Construction - Cấu kiện trong xây dựng** : bê tông, xà, thanh rầm, vì, kèo

a2b17.**Weapon - Vũ khí** : tàu ngầm, xe tăng, đại bác, súng, đạn, bom

a2b18.**OtherThings - Các vật khác** : giàn, hàng rào, thanh, que, nan

a2c.Part - Bộ phận

a2c01.**BodyPart - Bộ phận cơ thể người và động vật**

a2c01a.**Person Part - Bộ phận cơ thể người** : tóc, lông mày, tá tràng, râu ria

a2c01b.**Animal Part - Bộ phận cơ thể động vật** : đuôi, gạc, lòng, mề, vây, vi

a2c02.**Plant Part - Bộ phận của thực vật** : hoa, lá, nụ, cành, nhánh, chồi, mầm

a2c03.**Artifact Part - Bộ phận của đồ vật** : cán, đế, nắp, đít, miệng, gọng, càng

a2d.Substance - Chất

a2d01.**Material - Nguyên liệu** : lụa, gấm, nhung, kaki, giấy, mực, xi măng, hắc in

a2d02.**Solid - Chất rắn** : đá, sắt, cát, kim cương, hồng ngọc, than

a2d03.**Earth - Đất** : đất, bùn, mùn

a2d04.**Liquid - Chất lỏng** : nước, xăng, rượu, bia, cồn, máu, mủ

a2d05.**Gas - Chất khí** : ga, khí hydro, oxy

a2d06.**Addiction - Chất gây nghiện** : heroin, hồng phiến, thuốc phiện, thuốc lá, rượu

a2d07.**Poisonous - Chất độc** : thạch tín

a2d08.**Excrete - Chất thải** : phân, rác

a2e.Natural Object - Tự nhiên

a2e01.**Land - Đất** : ruộng, vườn, đồi, núi, cao nguyên, sa mạc, đảo, quần đảo

a2e02.**BodyOfWater - Vùng có nước** : sông, hồ, đầm, vũng, vịnh

a2e03.**HeavenlyBodies - Vũ trụ** : sao, hành tinh, Mộc Tinh, Đại Hùng

a3.Location - Địa điểm

a3a.**PlaceName - Địa danh** : Hà Nội, Quảng trường Ba Đình, Châu Âu, Đông Phi

a3b.**Region - Vùng miền** : cao nguyên, bình nguyên, trung du, lục địa

a3c.**FictionalPlace - Nơi tưởng tượng** : thiên đàng, địa ngục, miền đất hứa, âm phủ

b.ABSTRACTION - TRỪU TƯỢNG

b1.State - Trạng thái

b1a.**Feeling - Cảm giác** : đói, no, nhức, buốt, cay, xót, rát

b1b.**Physiological - Sinh lí** : thức, ngủ, mơ, mộng, ốm, đau

b1c.**Change - Biến hoá** : hoá, biến, bay hơi, (băng) tan, héo, úa, khô, quăn, teo, tóp

b1d.**Contact - Tiếp xúc** : gần (quần chúng), xa (dân), xa lánh, thân thiện, gần gũi

b1e.**Existence - Tồn tại** : còn (tiền), ra đời (tổ chức), sống, ở, hấp hối, dính, bám

b1f.**Devour - Tiêu biến** : chết, hi sinh, thiệt mạng, tuyệt diệt, tuyệt chủng

b1g.**Consumption - Tiêu huỷ** : long, gãy, đổ, sập, sụp, cháy, rụi, tàn, lụi

b1h.**Motion - Vận động**: (xe) lao (xuống vực), (trăng) nhô (lên cao), chảy

b1i.**Staying - Ngừng nghỉ** : nằm, nghỉ ngơi, dừng, đỗ, ngừng, dứt, tạnh, ngớt, thôi

b1j.**Perception - Nhận thức** : lơ mơ, mang máng, bập bõm, cảm giác, tưởng

b1k.**Possession - Sở hữu** : có (tiền), còn (tiền)

b1l.**Depend - phụ thuộc** : ăn theo, nương nhờ, phụ thuộc, bị, phải, được

b1m.**Surrender - khuất phục** : chịu, chấp nhận, đành

b1n.**Receive - Tiếp nhận** : được, bắt (vải bắt màu),

b1o.**Enjoy - Hưởng thụ** : hưởng, hưởng thụ, hưởng lạc, vui chơi

b1p.**Psychology - Tâm lí** : thẹn, ngượng, xấu hổ, e ấp, run sợ, đau (lòng)

b1q.**Stimulation - Khuyến khích** : vẫy gọi, gọi mời

b1r.**Experience - Trải nghiệm** : sành, trải đời, từng trải

b1s.**Emotion - Cảm súc** : tức giận, sợ sệt, tin tưởng, hoài nghi, thông cảm, yêu thích

b1t.**Desire - Mong muốn** : mong, mong ước, mong ngóng, nguyện ước, ước muốn

b2.Action - Hành động

b2a.**Abandon - Từ bỏ** : bỏ rơi, từ bỏ, chối bỏ, ruồng bỏ, bỏ rơi

b2b.**Care - Chăm sóc** : săn sóc, nuôi dưỡng, bảo dưỡng, bảo hành

b2c.**Act - Tác động** : xô, đẩy, ẩy, thúc, lôi, kéo, bật, tắt

b2d.**Cognitive Act - Nhận thức** : học, hiểu, nghe lời, am hiểu, am tường, quyết định

b2e.**Emotional Act - Xúc cảm** : hôn, yêu đương, ôm ấp, ấp ủ, mong ngóng

b2f.**Communicate - Giao tiếp** : nói chuyện, trò chuyện, gặp gỡ, thảo luận, bàn bạc

b2g.**Contact - Tiếp xúc** : sờ, nắn, nếm, liếm

b2h.**Move - Di chuyển** : đi (ra), chạy (vào), bay, bò, nhảy, lăn

b2i.**Keep - Ngăn giữ** : ách (ai lại), tóm, bắt, đình chỉ, giam cầm

b2j.**Create - Tạo tác** : đẽo, gọt, vót, vẽ, viết, sáng tác, xây dựng, lắp ghép, thiết đặt

b2k.**Change - Biến đổi** : cắt, băm, xé, nấu, luộc

b2l.**PresentAndReceive - Trao nhận** : cho, tặng, gửi, ban, phát, phân phát

b2m.**Order - Gây khiến** : sai, bảo, khiến, bắt, khuyên, nhờ, giúp, hướng dẫn

b2n.**Consume - Tiêu huỷ** : phá (cầu), đốt (nhà), huỷ hoại, tiêu diệt, thanh trừng

b2o.**Oppose - Đối kháng** : chiến đấu, thi đấu, đấu tranh, đấu đá, chống chọi

b2p.**Provoke - Khiêu khích** : chọc giận, chọc tức, trêu ngươi, trêu tức

b2q.**Appropriate - Chiếm đoạt** : cướp, giật, ăn cắp, ăn cướp, chiếm đoạt, xâm lược

b2r.**Negative - Phủ định** : chối, bác bỏ, bài trừ, phủ nhận, phủ định

b2s.**Depend - Chi phối** : a dua, hùa, áp đặt

b2t.**Collect - Tập hợp** : cóp nhặt, gom góp, góp nhặt, góp nhóp

b2u.**PsychologicalReflection - Phản ứng tâm lí** : chữa thẹn, cười trừ, cười trừ, cười ruồi, cười khẩy, cười xoà, cười mũi, nũng nịu, khóc nhè

b2v.**Conclude - Kết luận** : kết án, kết tội, kết luận, khép, thống nhất

b2w.**Recover - Phục hồi** : minh oan, giải oan, tẩy oan, hồi sức, hồi phục

b2x.**Respond - Phản hồi** : đáp, hồi đáp, hồi âm,

b2y.**Punish - Trừng phạt** : trừng phạt, trừng trị, nghiêm trị, đánh đòn, tống cổ, truy tố

b2z.**Advise - Khuyên bảo** : khuyên, khuyên răn, khuyên bảo

b3.Activity - Hoạt động

b3a.**PhysicalActivity - Hoạt động của cơ thể** : thở, ngồi, đứng, nhìn, ngửi, nghe, bài tiết, đái, ỉa

b3b.**PhysiologicalAction - Sinh lí** : giao hợp, giao cấu, đẻ, ấp, nhảy ổ

b3c.**Exchange - Trao đổi** : mua bán, sang nhượng, bàn giao, hợp đồng, hứa hẹn

b3d.**Receive - Tiếp nhận** : hưởng, hưởng lạc, hưởng thụ, kế thừa, ăn, uống, đoc (thông tin), thắng lợi

b3e.**Change - Biến đổi** : (máy) chạy, (máy) nổ, (máy) phát (điện)

b3f.**Contact - Tiếp xúc** : va, đập, đụng, quệt ...

b3g.**Communication - Giao tiếp** : liên lạc, đàm thoại, thông báo

b3h.**Emotion - Cảm xúc** : tức giận, sợ sệt, tin tưởng, hoài nghi, thông cảm, yêu thích

b3i.**Thinking - Tư duy** : suy nghĩ, nhận xét, suy xét, phán đoán

23j.**Motion - Chuyển động** : lung lay, lung linh, chấp chới, dập dờn

23k.**PhysicalAction - Tự nhiên** : giao thoa, thuỷ phân, ăn mòn

23l.**SocialActivity - Xã hội** : tuyên truyền, giáo dục, phát thanh, tổ chức, lãnh đạo

23m.**Motion - Vận động** : co, rụt, gập, duỗi, chảy

23n.**Affect - Ảnh hưởng** : giúp (tiến bộ), khiến (tỉnh giấc)

b4.Phenomenon - Hiện tượng

b4a.**Event - Sự kiện** : cách mạng, khởi nghĩa, chiến tranh

b4b.**PhysiologicalPhenomenon - Sinh lí** : tình dục, dậy thì, phát dục, động đực

b4c.**DiseasePhenomenon - Bệnh lí** : cảm, lao, lậu, ung thư, sẩy thai, băng huyết

b4d.**NaturalPhenomenon - Tự nhiên** : mưa gió, lũ lụt, bão bùng, hạn hán, hải lưu

b4e.**SocialPhenomenon - Xã hội** : trào lưu, khuynh hướng, chế độ, thực dân

b4f.**Cognition - Sự nhận thức** : suy nghĩ, nhận xét, suy xét, phán đoán

(i) b5.AbstractThing - Sự việc trừu tượng

b5a.**Life - Sự sống** : sống, sự sống, đời sống, cuộc sống, cây mầm, giống, con giống

b5b.**ResultOfAction - Kết quả của hoạt động** : thành quả, thắng lợi, sản phẩm

b5c.**SocialAbstractThing - Xã hội** : dân số, dân sinh, hộ khẩu, đơn từ, thư từ

b5d.**CulturalAbstractThing - Văn hoá** : lễ hội, đình đám, ma chay, cưới xin

b5e.**Concept - Khái niệm** : khoa học, công nghệ, toán học, kinh tế, chính trị

b5f.**Sound - Âm thanh** : ầm ầm, ì ầm, đùng đoàng, cọt kẹt, leng keng

b5g.**Colour - Màu sắc** : xanh, đỏ, tím, vàng

b5h.**Smell - Mùi** : thơm, hăng, khai, thối, tanh

b5i.**Taste - Vị** : ngọt, bùi, chua, cay, mặn, chát

b5j.**SportAndRecreation - Thể thao và giải trí** : bóng đá, cờ vua, điền kinh

b5k.**LogicalPlace - Phương hướng** : trên, dưới, trong, đông, tây, thượng nguồn

(ii) b6.Relation - Quan hệ

b6a.**Space - Không gian** : (nhà) gần (trường), sát, liền, kề, bên, cạnh

b6b.**Time - Thời gian**

b6b01.Point Of Time - Thời điểm : khoảnh khắc, thời điểm, lúc, hồi

b6b02.Period - Thời kì, giai đoạn : trước đây, hiện nay, bây giờ, quá khứ, xưa

b6c.**SetOrGroup - Tập hợp hoặc nhóm** : lẻ loi, cô đơn, cô quạnh, đông đúc

b6d.**Comparison - So sánh** : tương đương, hơn, kém

b6e.**Identical - Đồng nhất** : là (giáo viên), làm (công nhân)

b6f.**Negative - Phủ định** : không, chưa, chẳng

b7.Attribute - Thuộc tính

b7a.**Quality - Phẩm chất** : bền, dai, bở, tốt, xấu, trung bình, tuyệt, thông minh

b7b.**Quantity - Số lượng** : nhiều, ít, ngắn, dài, vô khối, vô số, to, nhỏ, dày, mỏng

b7c.**Size - Kích thước** : to, nhỏ, béo, gầy, dày, mỏng, cao, thấp

b7d.**Shape - Hình dạng** : tròn, méo, nhọn, tù, cong, thẳng, vênh, xiên, lệch

b7e.**Characteristic - Đặc tính** : chua, cay, ngọt, mặn, chát, thơm, thối, nóng, lạnh

b7f.**SortOrType - Loại hoặc kiểu** : cũ, mới, hiện đại, mô đen

b7g.**Condition - Điều kiện** : thuận lợi, bất lợi, khách quan, chủ quan

b7h.**Appearance - Xuất hiện** : ló, mọc, lộ, hiện, phơi bày, phanh phui

b7i.**Hidden - Ẩn** : lấp, ẩn, lặn, kín, tối mò

b8.Value - Giá trị

b8a.**Definite - Xác định** : một, hai, tư, các

b8b.**Indefinite - Không xác định** : những, dăm ba, một vài, trăm nghìn

b9.Unit - Đơn vị

b9a.**UnitOfObject - Đơn vị chỉ loại sự vật** : con, cây, cục, cái

b9b.**UnitOfCollection - Đơn vị tập hợp** : toán, bầy, lũ, mớ, khóm, bụi

b9c.**UnitOfMetrical - Đơn vị đo lường** : tấn, tạ, yến, cân, kg, cm, ha

b9d.**UnitOfTime - Đơn vị thời gian** : giờ, phút, giây, ngày, tháng, quý, năm

b9e.**UnitOfCurrency - Đơn vị tiền tệ** : hào, đồng, đô la, bảng, bạt, yên

b9f.**UnitOfFrequency - Đơn vị chỉ tần suất :** lần, lượt, bận

b9g.**UnitOfAction - Đơn vị của hành động :** cú, keo