

VIETNAMESE LFG EDITOR AND GRAMMAR CHECKING TOOL

Lê Huy

Phạm Thị Bích Liên

Đào Văn Tuyết

TÓM TẮT

Ngữ pháp chức năng là một lý thuyết và hệ thống phương pháp được xây dựng trên quan điểm coi ngôn ngữ như một phương tiện thực hiện sự giao tiếp giữa người với người. Là phương pháp khảo sát ngôn ngữ gắn với hoạt động hình thức của nó. Trong khi ngữ pháp cổ điển, ngữ pháp cấu trúc thì lại chú ý đến cơ chế hình thức tách ra khỏi hoạt động ngôn ngữ.

Trên thế giới, ngữ pháp chức năng đã được ứng dụng rộng rãi và đã đem lại một số kết quả đáng kể. Nhưng ở Việt Nam, ngữ pháp chức năng hầu như còn xa lạ với nhiều người. Trong bài báo này, chúng tôi muốn giới thiệu về ngữ pháp chức năng, và quy trình xây dựng Vietnamese LFG Editor và Vietnamese Grammar Checking Tool trên XLE Platform. XLE là một platform được phát triển bởi PARC và Grenoble là một bộ công cụ ngôn ngữ học mạnh. Hỗ trợ việc tạo ra, sửa văn phạm LFG và phân tích câu theo hình thái LFG.

Từ khóa: XLE, Ngữ pháp chức năng, Lexical Functional Grammar, Parallel Grammar and Semantics, LFG Editor, grammar checking.

I. TỔNG QUAN

Trong mọi thời đại, vấn đề giao lưu hợp tác giữa các quốc gia là hết sức quan trọng về mặt kinh tế lẫn chính trị. Do đó, vấn đề ngôn ngữ đặc biệt đáng quan tâm. Gần đây, trường đại học Stanford đã khởi động dự án *Parallel Grammar and Semantics Project* với sự tham gia của nhiều nhà nghiên cứu và các viện, trường học trên toàn thế giới. Mục đích của dự án này là tạo ra các cấu trúc ngữ pháp có thể áp dụng cho nhiều ngôn ngữ trên thế giới. Dự án được viết dựa trên nền ngôn ngữ chức năng từ vựng LFG và sử dụng công cụ XLE.

Hiện nay dự án đã phát triển được các ngôn ngữ như: Anh, Pháp, Đức, Trung Quốc, Nhật, Ả rập, ... và ngôn ngữ tiếng Việt cũng đã bước đầu tham gia vào dự án này. Khi dự án thành công, ta có thể ánh xạ được các cấu trúc, từ vựng giữa các ngôn ngữ với nhau,...

Bộ công cụ ngôn ngữ XLE làm việc với các file lfg, chứa tập luật và tập từ vựng dùng để phân tích các câu văn. Cấu trúc các file lfg cũng gần giống với các cấu trúc C và F của LFG, chỉ thay đổi các ký hiệu để dễ dàng biểu diễn dưới dạng file văn bản trong máy tính. Bộ Vietnamese LFG Editor làm công việc tạo ra, chỉnh sửa và kiểm tra cú pháp các file lfg dựa trên nền XLE, với các giao diện tiện cho viết luật lfg một cách nhanh nhất. Bộ Vietnamese Grammar Checking tool kiểm tra ngữ pháp tiếng Việt của một câu văn, một đoạn văn hay là file văn bản.

II. NGỮ PHÁP CHỨC NĂNG TỪ VỰNG LFG

Ngữ pháp chức năng từ vựng (Lexical Functional Grammars - LFG) được giới thiệu lần đầu tiên vào năm 1982 bởi Ronald M.Kaplan và Joan Bresnan. LFG có 2 cấu trúc biểu diễn ngữ pháp là cấu trúc thành phần (constituent structure) gọi tắt là cấu trúc C và cấu trúc chức năng (functional structure) gọi tắt là cấu trúc F.

Cấu trúc C có dạng cây cú pháp dựa trên tập luật ngữ pháp phi ngữ cảnh (context - free grammar) có bổ sung các ghi chú. Cấu trúc F có dạng ma trận chứa các giá trị thuộc tính bao gồm

các thông tin về các chức năng ngữ pháp như chủ ngữ, tân ngữ hay các đặc trưng ngữ pháp như thì (tense) của câu, ...

Hai cấu trúc này tồn tại độc lập với nhau nhưng có thể ánh xạ giữa các thành phần của chúng với nhau. Mỗi quan hệ giữa cấu trúc C và cấu trúc F tương ứng được cho bởi một hàm chiếu φ từ các nốt của cây cấu trúc C đến ma trận thuộc tính của cấu trúc F.

Cấu trúc thành phần C (C - structure)

Đối với một cấu trúc cây, ta có tập nốt N ánh xạ qua hàm λ tới tập nhãn hữu hạn L và hàm M ánh xạ từ nốt tới nốt và ký hiệu để chỉ thứ tự <:

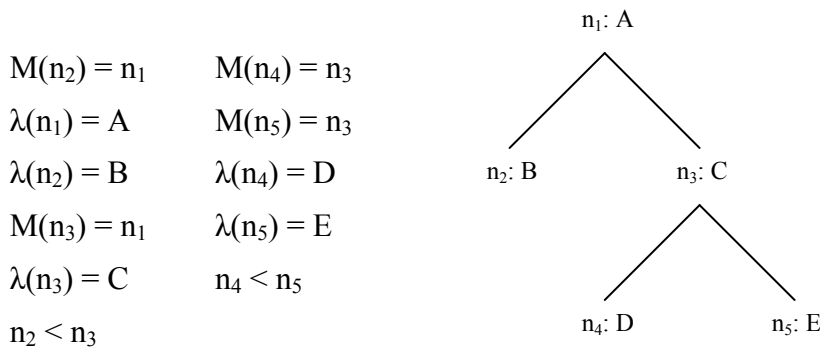
N: tập nốt, L: tập nhãn

M: $N \rightarrow N$

$< \subseteq N \times N$

$\lambda: N \rightarrow L$

Mô tả của một cấu trúc có thể gồm có danh sách định nghĩa các thuộc tính và mối quan hệ. Xét một cấu trúc C dạng cây đơn giản và mô tả đầy đủ của nó:



Chú ý rằng với bất kỳ n_1 và n_2 nếu $M(n_1) < M(n_2)$ thì $n_1 < n_2$.

Cấu trúc chức năng (F - structure)

Cấu trúc F được định nghĩa một cách đệ quy: chúng là các hàm hữu hạn có cấp bậc ánh xạ từ những phần tử trong tập ký hiệu đến các giá trị cũng có thể là các ký hiệu, cấu trúc F con hay dạng ngữ nghĩa như 'robbed <SUBJ, OBJ>' mà ta sẽ xét ở ví dụ trước. Tập các cấu trúc F được phát biểu dưới dạng phương trình đệ quy như sau:

$F = (A \rightarrow {}_tF \cup A \cup S)$, A là tập các ký hiệu nguyên tử, S là tập các dạng ngữ nghĩa.

Xét dạng của cấu trúc F đơn giản như sau:

$$f_1: \left(\begin{array}{cc} q & f_2: \left(\begin{array}{cc} s & t \\ u & v \end{array} \right) \\ w & x \end{array} \right)$$

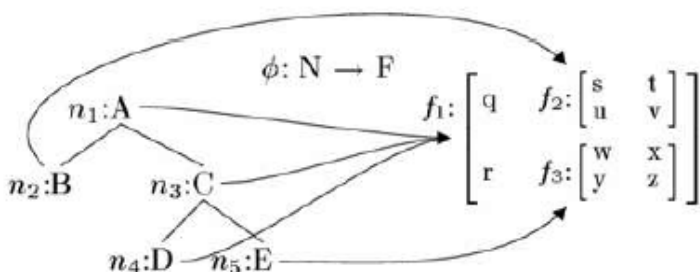
Chú ý rằng hàm f_1 áp dụng cho thuộc tính q có giá trị là f_2 , cho w là x, f_2 áp dụng cho thuộc tính s có giá trị là t, cho u là v.

$(f a) = v$ nếu $\langle a v \rangle$ thuộc f , trong đó f là một cấu trúc F và a là một ký hiệu nguyên tử (atomic symbol). Mô tả cấu trúc F:

$$\begin{aligned} (f_1 \quad q) &= f_2 \\ (f_2 \quad s) &= t \\ (f_2 \quad u) &= v \\ (f_1 \quad w) &= x \end{aligned}$$

Mối quan hệ giữa hai cấu trúc

Mối quan hệ giữa cấu trúc C và cấu trúc F được ánh xạ bởi hàm chiếu ϕ đi từ các nốt của cây cấu trúc C đến các đơn vị không gian cấu trúc F như sơ đồ cây dưới đây:



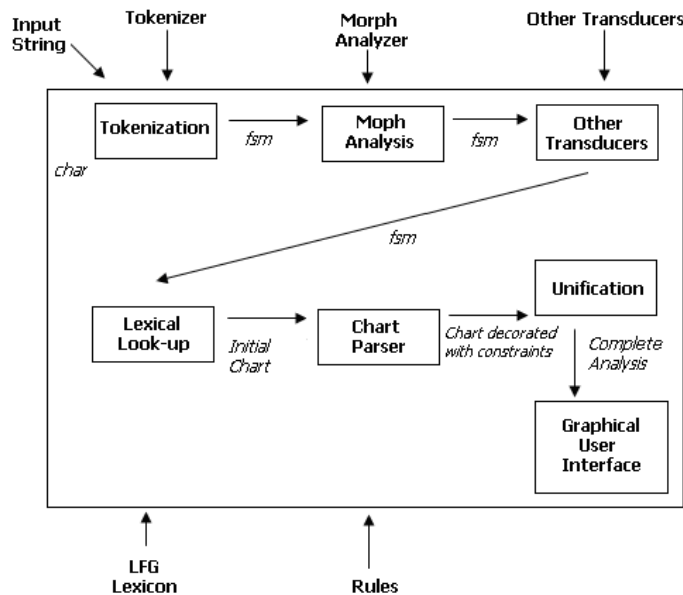
Cấu trúc F trong sơ đồ thỏa mãn điều kiện $(f_1 \quad q) = f_2$: một ràng buộc trong ngôn ngữ mô tả cấu trúc F, mà f_1 và f_2 là những cấu trúc F tương ứng với các nốt n_1, n_2 nên điều kiện này có thể biểu diễn tương đương $(\phi(n_1) \quad q) = \phi(n_2)$. Chú ý rằng n_1 là cha của n_2 ta sẽ có được phương trình $(\phi(M(n_2)) \quad q) = \phi(n_2)$. Thay thế các giá trị hàm f_i trong các miêu tả của cấu trúc F bằng các biểu thức $\phi(n_i)$ và sử dụng mối quan hệ cha con trong cây cấu trúc ta có:

$$\begin{aligned} (\phi(M(n_2)) \quad q) &= \phi(n_2) & M(n_2) &= n_1 \\ (\phi(n_2) \quad s) &= t & (\phi(n_2) \quad u) &= v \\ (\phi(n_5) \quad w) &= x & (\phi(n_5) \quad y) &= z \\ \phi(M(n_3)) &= \phi(n_3) & M(n_3) &= n_1 \\ \phi(M(n_4)) &= \phi(n_4) & M(n_4) &= n_3 \\ (\phi(M(n_2)) \quad r) &= \phi(n_5) & M(n_5) &= n_3 \end{aligned}$$

III. XLE PLATFORM VÀ MÔ PHỎNG CÁCH PHÂN TÍCH CÂU

Mô hình kiến trúc:

XLE lần đầu tiên được viết vào những năm 1990 sử dụng ngôn ngữ C và Tcl/Tk. Đây là một môi trường hỗ trợ việc viết và debug văn phạm LFG chạy trên môi trường Unix. Nó giúp viết các quy tắc về cú pháp một cách dễ dàng, và còn dùng để kiểm tra và chỉnh sửa. Nó cung cấp một giao diện với các bộ phân tích hình thái học trạng thái hữu hạn (finite-state morphological analyzers). Hệ thống có quy trình xử lý dữ liệu đầu vào (các câu hay các chuỗi) rồi sau đó cung cấp thông tin phân tích:



XLE phân tích câu dựa trên các luật ngữ pháp (tập luật - rules), các từ vựng (LFG Lexicon) để phân tích chuỗi nhập vào (Input string) trải qua các giai đoạn tách chuỗi thành các từ (Tokenization), phân tích hình thái (Morph Analyzer), một số phép biến đổi (other transducer), phân tích (Chart Parser), kết quả sẽ được tổng hợp (Unification) để hiển thị ra màn hình giao diện đồ họa người dùng (Graphical User Interface).

Tập luật ngữ pháp và từ vựng được lưu trữ dưới dạng các file văn bản, được nạp vào XLE thông qua các lệnh của chính XLE.

Xây dựng ngữ pháp:

Phần này sẽ giới thiệu tổng quan về các thủ tục xây dựng các luật ngữ pháp, các từ vựng (lexical entries), và các quy tắc hình thái mà một ngôn ngữ cần có để phân tích câu.

Hình thái LFG cho phép người viết ngữ pháp có thể phân biệt các phiên bản khác nhau trên cùng một quy tắc hay một từ vựng bằng cách gán cho chúng những tên khác nhau. Mỗi tên gồm có ba phần: phiên bản, ngôn ngữ và định danh (item- identifier).

Quy ước các ký hiệu tương ứng giữa LFG và XLE:

LFG	XLE	
↑	^	siêu biến cấu trúc F
↓	!	siêu biến cấu trúc F
=	=	dấu bằng
→	->	ràng buộc
{ a b c ... z }	{ a b c ... z }	phân cách (disjunction)
(a)	{ a }	ràng buộc tùy chọn cấu trúc F
symbol_	symbol_	chú thích

Một ngữ pháp tối thiểu phải có một file chứa ngữ pháp và từ vựng. Trước hết hãy xét file toyeng.lfg chứa ngữ pháp và từ vựng, có dạng tổng quát như sau:

```

TOY ENGLISH CONFIG (1.0)
ROOTCAT S.
    
```

FILES .
 LEXENTRIES (TOY ENGLISH).
 RULES (TOY ENGLISH).
 TEMPLATES (TOY ENGLISH).
 GOVERNABLERELATIONS SUBJ OBJ OBJ2 OBL OBL-?+ COMP XCOMP.
 SEMANTICFUNCTIONS ADJUNCT TOPIC.
 NONDISTRIBUTIVES NUM PERS.
 EPSILON e.
 OPTIMALITYORDER NOGOOD.

 TOY ENGLISH RULES (1.0)

 TOY ENGLISH TEMPLATES (1.0)

 TOY ENGLISH LEXICON (1.0)

Mỗi section bắt đầu với tên phiên bản, ngôn ngữ, thành phần và phiên bản của chương trình XLE và kết thúc bởi 4 dấu gạch ngang:

- TOY ENGLISH CONFIG (1.0): TOY là phiên bản của ngữ pháp; ENGLISH là ngôn ngữ; CONFIG hàm ý rằng đây là cấu hình của phiên bản và ngữ pháp; (1.0) là phiên bản của XLE.
- TOY ENGLISH RULES (1.0): tên của phần trong file chứa các luật.
- TOY ENGLISH TEMPLATES (1.0): tên của phần trong file chứa các mẫu.
- TOY ENGLISH LEXICON (1.0): tên file chứa các từ vựng (lexical items).

Cài đặt và chạy chương trình

Sau đây, chương trình sẽ được cài đặt trên hệ điều hành Linux là Fedora Core 6.

Trước hết, phải thiết lập biến môi trường trong Linux bằng cách thêm các dòng lệnh sau vào tập tin khởi động, là file .bashrc:

```
export XLEPATH $xledir
export PATH ${XLEPATH}/bin:$PATH
export LD_LIBRARY_PATH ${XLEPATH}/lib:$LD_LIBRARY_PATH
export DYLD_LIBRARY_PATH ${XLEPATH}/lib:$DYLD_LIBRARY_PATH
```

Trong đó, \$xledir là thư mục chứa XLE, ví dụ:

```
export XLEPATH="/XLE"
export PATH="${XLEPATH}/bin:$PATH"
export LD_LIBRARY_PATH="${XLEPATH}/lib:$LD_LIBRARY_PATH"
export DYLD_LIBRARY_PATH="${XLEPATH}/lib:$DYLD_LIBRARY_PATH"
```

Chọn Save để lưu cấu hình boot. Bây giờ mở ứng dụng Terminal của Linux để gõ các lệnh Shell. Trên màn hình hộp thoại Terminal, gõ lệnh xle để gọi chương trình. Màn hình sẽ hiển thị ra các thông số như phiên bản, tổ chức,...XLE có rất nhiều lệnh, có thể gõ lệnh help để xem các lệnh.

Bây giờ để nạp ngữ pháp từ file, ta dùng lệnh:

```
% create-parser "filename"
```

Quá trình nạp ngữ pháp hoàn tất, dùng lệnh sau để phân tích câu:

```
% parse "sentence"
```

Ví dụ: phân tích câu: “ông già đi nhanh quá.” Với bộ luật sau:

```
DEMO VIETNAMESE CONFIG (1.0)
ROOTCAT S.
FILES .
LEXENTRIES (all all).
RULES (all all).
TEMPLATES (all all).
GOVERNABLERELATIONS ADJ ADJ2 ADV SUBJ OBJ OBJ2 COMP XCOMP OBL OBL-
?+.
SEMANTICFUNCTIONS ADJUNCT TOPIC FOCUS POSS STANDARD.
NONDISTRIBUTIVES NUM PERS CONJ-FORM.
EPSILON e.
OPTIMALITYORDER NOGOOD.
----
DEMO VIETNAMESE RULES (1.0)
S --> NP: (^ SUBJ)=! }
      VP.
VP --> V
      (PE)
      (ADVP: (^ ADV)=!)
      (ADJP: (^ ADJ)=!)
PP --> P.
ADVP --> ADV
      (W).
ADJP --> ADJ
      (DEM).
NP --> (DET)
      (CL)
      N
      (DEM).
----
```

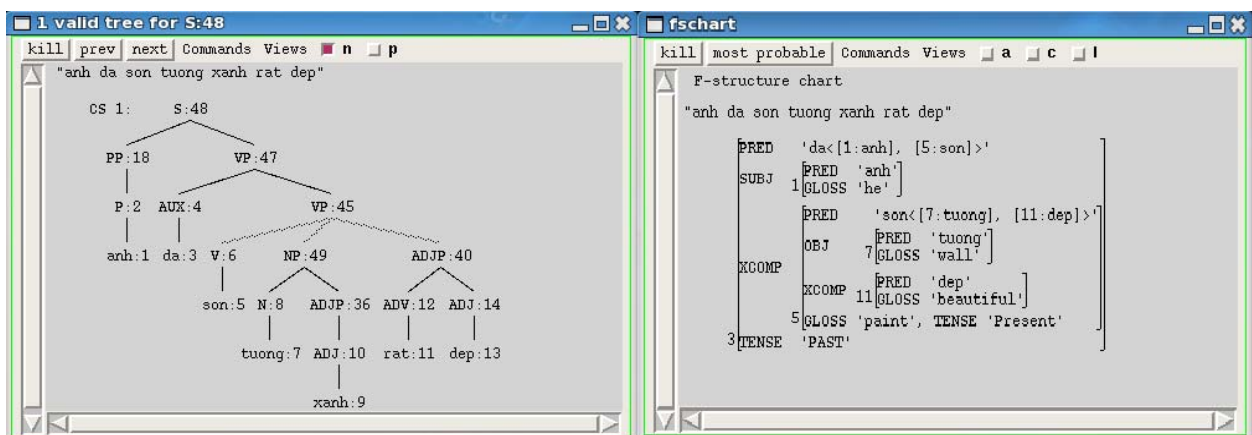
DEMO VIETNAMESE TEMPLATES (1.0)

CN(P) = (^ PRED)= 'P'.
 PN(P) = (^ PRED)= 'P'.
 CV(P) = (^ PRED)= 'P<(^ SUBJ)(^ ADV)>'.
 LAV(P) = (^ PRED)= 'P<(^ SUBJ)(^ ADJ)>'.
 CAVD(P) = (^ PRED)= 'P'.
 CADJ(P) = (^ PRED)= 'P'.
 PER2(P) = (^ PRED)= 'P'.

DEMO VIETNAMESE LEXICON (1.0)

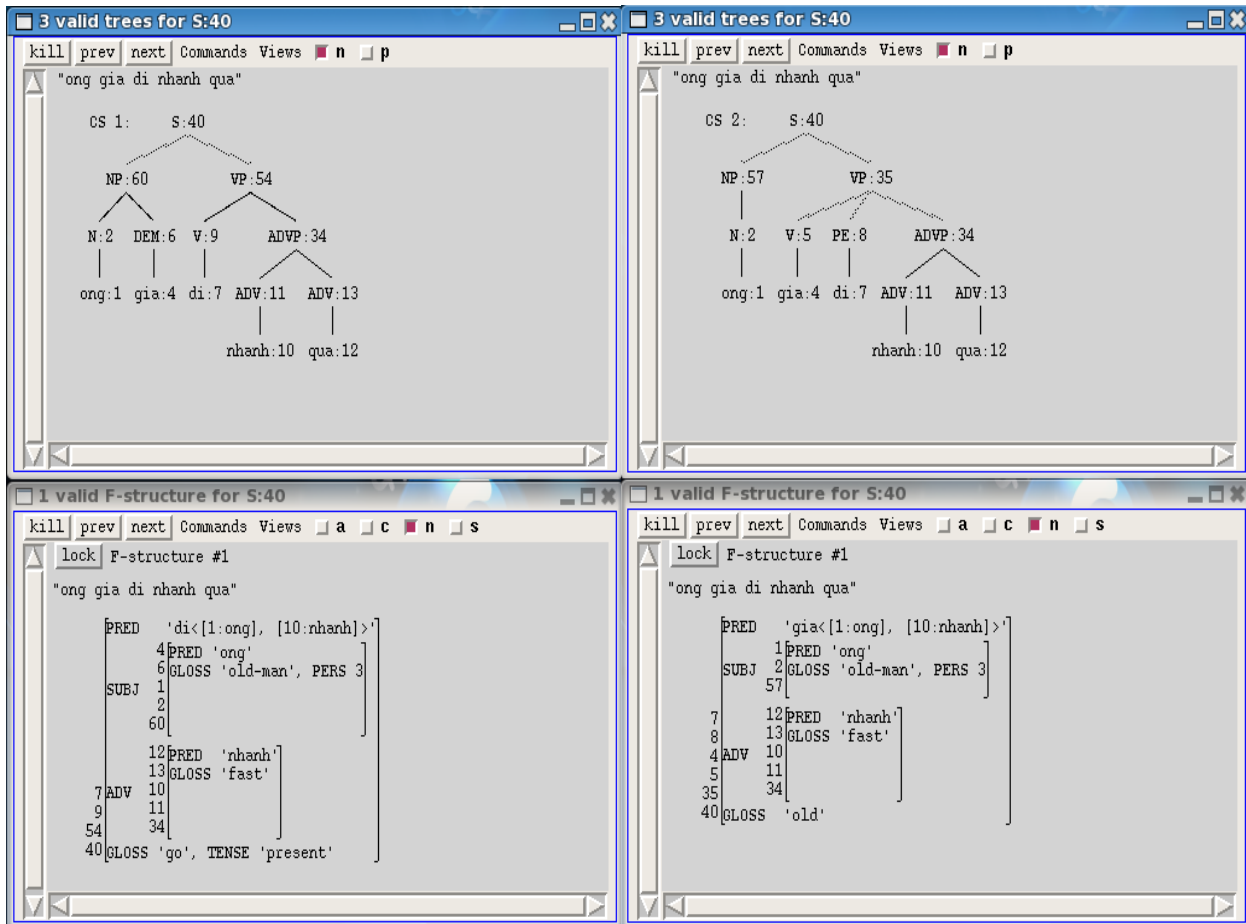
anh N * @(CN anh).
 ong P * @(PER2 ong);
 N * @(CN ong).
 gia DEM *;
 V * @(CV gia).
 di V * @(CV di);
 PE *.
 nhanh ADV * @(CAVD nhanh).
 qua W *.
 Hue N * @(PN HUE).
 that V * @(LAV that).
 dep ADJ * @(CADJ dep).
 co ADJ * @(CADJ co).
 kinh DEM *.

Và kết quả thu được khi phân tích với XLE là:



Kết quả phân tích câu: “anh đã sơn tường xanh rất đẹp”

Và thu được kết quả sau:



Kết quả phân tích câu: “ông già đi nhanh quá”

IV. VIETNAMESE LFG EDITOR, VIETNAMESE GRAMMAR CHECKING TOOL

1. Vietnamese LFG Editor

Trên đây đã xây dựng được tập luật tiếng Việt đơn giản áp dụng vào XLE để phân tích một số câu đơn giản. Việc xây dựng luật đã gặp rất nhiều khó khăn trong việc hình tượng cây luật và xây dựng tập từ vựng. Trong tương lai số luật và tập từ vựng phải xây dựng là rất lớn. Vậy cần thiết phải xây dựng một công cụ hỗ trợ giúp xây dựng, kiểm tra luật LFG và tập từ vựng. Giúp nhanh chóng hoàn thành tập luật LFG tiếng Việt hoàn chỉnh, bước đầu tham gia dự án Parallel Grammar and Semantic.

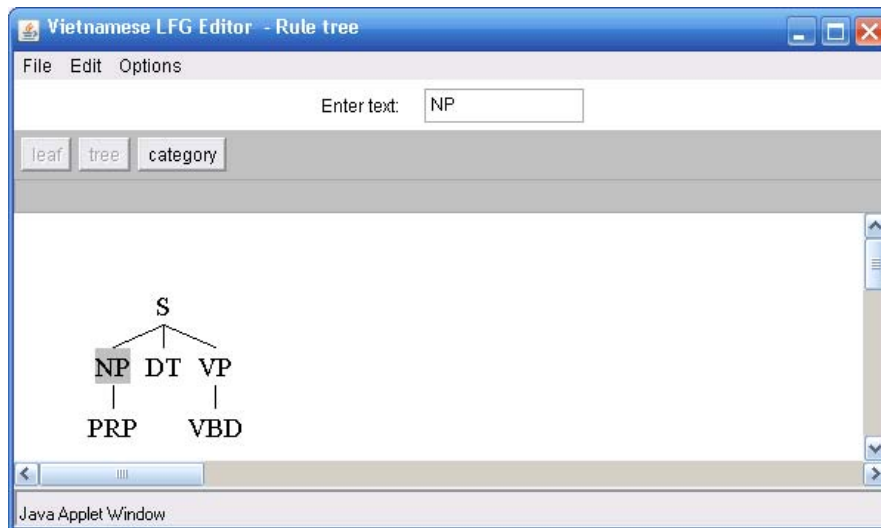
Các chức năng của Vietnamese LFG Editor

- Kiểm tra cú pháp của một file .lfg.
- Biểu diễn cây luật của từng luật trong file lfg. Cho phép chỉnh sửa trên cây và tự động cập nhật vào file.
- Thống kê tập từ vựng, cho thêm từ, xóa từ, sửa từ. Chỉnh sửa cái thuộc tính của một từ vựng:
Word Category Morphcode Schemata;
- Phân tích câu và biểu diễn kết quả bằng cấu trúc C và cấu trúc F để kiểm tra tính đúng đắn của tập luật.

Phương pháp thực hiện

- Sử dụng các interface của XLE để kiểm tra cú pháp của file .lfg .
- Sử dụng kỹ thuật lập trình và đồ họa để biểu diễn, chỉnh sửa luật trong file lfg.
- Sử dụng cơ sở dữ liệu để lưu trữ tập từ vựng và luật trong file lfg.
- Sử dụng các interface của XLE để phân tích câu.

Ví dụ về giao diện chỉnh sửa cây luật:



Giao diện chỉnh sửa cây luật

Tình trạng công việc: Chúng tôi chỉ mới bắt đầu triển khai thiết kế bộ Vietnamese LFG Editor.

2. Vietnamese Grammar Checking Tool

Gần đây, báo chí thường xuyên phê phán tình trạng suy giảm hiểu biết về tiếng Việt của đa số thế hệ trẻ. Hiện tượng viết sai chính tả hoặc sai ngữ pháp thường xuyên xảy ra. Đây là mối quan tâm lo lắng của các nhà ngôn ngữ học cũng như của toàn xã hội. Với tư cách là những người công nghệ thông tin, chúng tôi mong muốn có thể xây dựng và cài đặt những thuật toán để có thể kiểm tra việc sai chính tả và sai ngữ pháp trong các văn bản.

Với sự hỗ trợ của những tập luật trong ngữ pháp chức năng từ vựng cũng như những công cụ trong bộ ngôn ngữ XLE, ta có thể thực hiện điều đó.

Kiểm tra chính tả

Quá trình phân tích câu bắt đầu bằng việc phân tích hình thái học. Những hình thái học này sẽ được khai báo trong mục từ vựng của tập tin .lfg. Vì vậy vấn đề kiểm tra chính tả có thể được giải quyết dễ dàng. Những từ trong quá trình duyệt từ trái qua phải của chuỗi mà không tồn tại trong mục từ vựng đã khai báo được xem như là sai chính tả. Ở đây ta cũng nên lưu ý đến các danh từ riêng cũng có thể bị nhận diện là sai chính tả.

Vậy độ chính xác trong hoạt động của chương trình phụ thuộc vào tập khai báo các mục từ vựng. Tập từ vựng càng hoàn chỉnh, chương trình càng chính xác. Do đó chương trình có thể tận dụng kết quả đã có của những nhà ngôn ngữ học, nghĩa là tài liệu để xây dựng các mục từ vựng

chính là các từ điển tiếng Việt, đồng thời chương trình cũng hỗ trợ giao diện để các nhà ngôn ngữ có thể đóng góp thêm những từ vựng mới.

Kiểm tra ngữ pháp

Trong quá trình hình thành các cấu trúc C và F của một câu, có một số ràng buộc để một câu có thể được xem là đúng chính tả. Đây là một số điều kiện cơ bản:

- *Điều kiện duy nhất (Uniqueness Condition):*

Trong một cấu trúc F cho sẵn, một thành phần đặc biệt chỉ có thể có nhiều nhất một giá trị.

- *Điều kiện để đúng ngữ pháp (Condition on Grammaticality):*

Một chuỗi được xem là đúng ngữ pháp chỉ khi nó có cả hai: một cấu trúc C hợp lệ và một cấu trúc F liên hệ với nó đều vững chắc (*consistent*) và xác định (*determinate*). Trong đó một cấu trúc F được xây dựng từ một chuỗi là giá trị trong giải pháp tối thiểu, đơn nhất của những đặc tả F của **biến**_↓ của nốt gốc của cấu trúc C.

Điều kiện này cần thiết nhưng không hiệu quả đối với ngữ pháp. Vì vậy ta phải thêm một số điều kiện như sau:

Một chuỗi là đúng ngữ pháp nếu và chỉ nếu nó hình thành nên một cấu trúc F hoàn chỉnh (*complete*) và gắn kết (*coherent*).

- Một cấu trúc F là hoàn chỉnh (*complete*) khi và chỉ khi nó và tất cả các cấu trúc F phụ của nó đều hoàn chỉnh cục bộ (*locally complete*).
- Một cấu trúc F là hoàn chỉnh cục bộ (*locally complete*) khi và chỉ khi nó chứa tất cả các hàm ngữ pháp chức năng chi phối mà các vị từ của nó ảnh hưởng.
- Một cấu trúc F là gắn kết (*coherent*) khi và chỉ khi nó và tất cả các cấu trúc F phụ của nó là gắn kết cục bộ (*locally coherent*).
- Một cấu trúc F là gắn kết cục bộ (*locally coherent*) khi và chỉ khi tất cả những hàm ngữ pháp chi phối mà nó chứa đều bị ảnh hưởng bởi một vị từ cục bộ.

V. KẾT LUẬN

Trong bài báo này, chúng tôi đã giới thiệu khái niệm ngữ pháp chức năng từ vựng (LFG). LFG có 2 cấu trúc chính là cấu trúc C và cấu trúc F, có thể chuyển đổi qua lại 2 cấu trúc này dễ dàng. XLE, một platform hỗ trợ phân tích ngữ pháp LFG trên môi trường Unix. Chúng tôi đã tạo ra một bộ luật đơn giản để phân tích một số câu đơn giản của tiếng Việt và đã thu về một số cấu trúc C và cấu trúc F tương ứng. Đồng thời giới thiệu về Vietnamese LFG Editor và Vietnamese Grammar Checking Tool – một bộ công cụ hỗ trợ việc tạo ra, chỉnh sửa luật LFG, kiểm tra ngữ pháp tiếng Việt dựa trên nền XLE.

Đây là những module cơ bản nhóm chuẩn bị để xây dựng một website cho ngữ pháp chức năng tiếng Việt. Đó là một môi trường để những nhà ngôn ngữ có thể cộng thêm các luật ngữ pháp chức năng tiếng Việt góp phần phát triển bộ luật hoàn chỉnh hơn, đối với các nhà tin học có thể góp phần phát triển thêm những giải thuật xử lý trên ngôn ngữ tự nhiên tiếng Việt như tối ưu các thuật toán đang sử dụng, phát triển thêm các thuật toán chuyển đổi, ...

Lời cảm ơn: Chúng tôi xin cảm ơn Viện Cơ học và Tin học Ứng dụng đã tạo mọi điều kiện thuận lợi để nhóm chúng tôi có thể tiếp cận với Parallel Grammar and Semantics Project và sử dụng XLE Platform trong nghiên cứu và xây dựng tập luật văn phạm tiếng Việt. Chúng tôi cũng xin cảm ơn PGS. TS. Đào Thanh Lan, Khoa Ngữ Văn, Đại học quốc gia Hà Nội đã cung cấp cho nhóm một số tài liệu về Ngữ pháp chức năng.

TÀI LIỆU THAM KHẢO

- [1] Formal Issues In Lexical – Functional Grammar của Mary Dalrymple, Ronald M.Kaplan, John T.Maxwell III, Annie Zaenen
- [2] A grammar writer’s Cookbook của Miriam Butt, Tracy Holloway King, Maria – Eugenia Nino & Frederique Segond
- [3] http://www2.parc.com/isl/groups/nlft/xle/doc/xle_toc.html
“XLE Documentation”
- [4] <http://www.computing.dcu.ie/~acahill/tts/>
TreeBank Tool Suite
- [5] Xây dựng tập luật ngữ pháp chức năng tiếng Việt và áp dụng trên nền XLE. Hội Thảo Quốc Gia XI tại Huế 06/2008, Lê Huy, Phạm Thị Bích Liên, Đào Văn Tuyết.