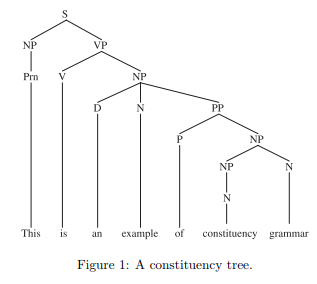
**STANDFORD PARSER**

1. **Khái niệm:**

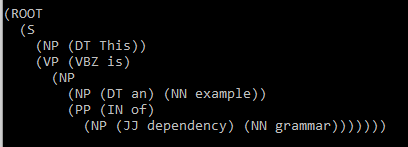
* Phân tích cấu trúc ngữ pháp của câu như những từ nào đi cùng nhau, từ nào là chủ ngữ, từ nào là vị ngữ,…
* Có 2 loại parser:
* Constituency Parser
* Dependency Parser

1. **Constituency Parser**

* Tách câu thành từng phần (cụm từ) sau đó tiếp tục tách thành từng phần nhỏ hơn.



Ví dụ: Với input là câu “This is an example of dependency grammar” thì output là:



Trong đó:

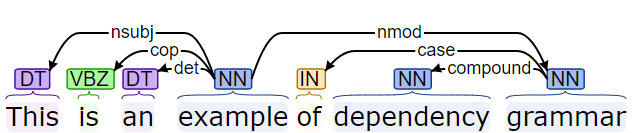
- S là “sentence”, nút gốc của cây

- Các từ “NP”, “VP”, “PP” chỉ loại của cụm từ, trong đó “NP” là cụm danh từ, “VP” là cụm động từ, “PP” là cụm giới từ

- Các từ “DT”, VBZ”, “IN”, “JJ”, “NN” là nhãn của từ trong Penn TreeBank

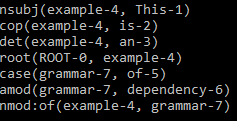
1. **Dependency Parser**

* Là vẽ đường liên hệ giữa các từ



Mũi tên chỉ từ “example” đến “this” nghĩa là từ “example” làm ảnh hưởng đến từ “this”. “nsubj” chỉ loại mối liên hệ giữa từ “example” và “this”.

**Ví dụ:** Với input là câu “This is an example of dependency grammar”, ta có output như sau:



Dòng 1 cho thấy từ “this” làm ảnh hưởng đến “example” và 2 từ này có mối liên hệ là “nsubj”, 4 và 1 là vị trí trong câu của “example” và “this”

1. **Cách thực hiện Parser bằng C#**

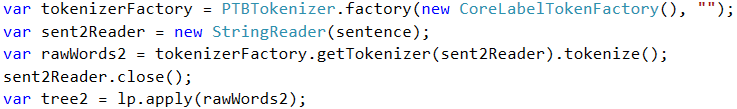
* Đầu tiên tạo đường dẫn đến file model



* Tiếp theo dùng class LexicalizedParser để load model:



* Sau đó xử lí input và đưa input vào model



* Cuối cùng dùng class TreePrint để xuất output. Ta có thể chọn định dạng cấu trúc TreePrint được in ra: “penn”, “oneline”, “latexTree”, “typedDependenciesCollapsed”, “words,typedDependencies”, “rootSymbolOnly” ,…



Ví dụ với 2 định dạng “penn, typedDependenciesCollapsed” như trên ta có output như sau:

