

형태소 분석과 구문분석

2022 11 10

전산언어학과 자연어처리

- 전산언어학

- 전산학적 모델을 사용하여 언어에 대한 이론적인 연구를 하는 분야

- 자연어 처리

- 사람의 언어를 처리하는 컴퓨터 프로그램 제작에 대한 실용적인 연구를 하는 분야
- 언어에 대한 효율적인 처리와 얼마나 많은 유형의 문장을 처리할 수 있는가 하는 것이 중요함

자연어 처리

- NLP (Natural Language Processing)
- HLT (Human Language Technology)
 - 사람의 말을 이해하고 반응하는 것
 - 문장을 이해하고 그에 해당하는 정보를 찾아 주는 것
 - 한 언어에서 다른 언어로 번역 해주는 것
 - 문서의 내용을 요약 해주는 것
 - ...

자연어 처리 예

■ Eliza scripting Example

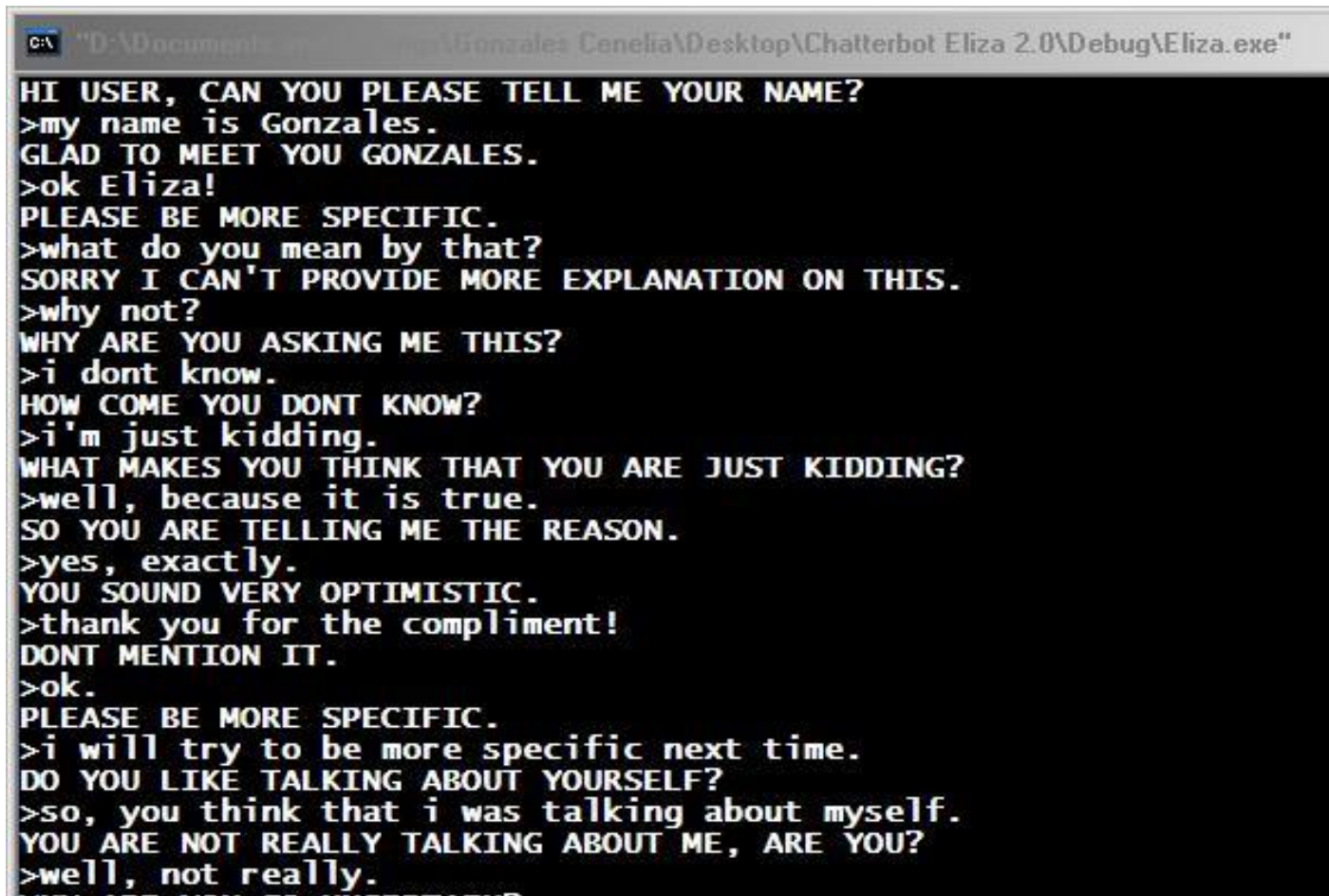
<heading-0> "Rule Heading"
a:0.2 the rule activation level
p:35 *what*keyword*...the pattern priority and word
pattern content/order
r: robot's reply

<work-0>
a:0.5
p:60 Wh *your*job*
r:I'm a full time Verbot

<leasure-2>
a:0.4
p:30 What time * your * job over.
r:I don't get any time off, I always have to be here available for you.

자연어 처리 예

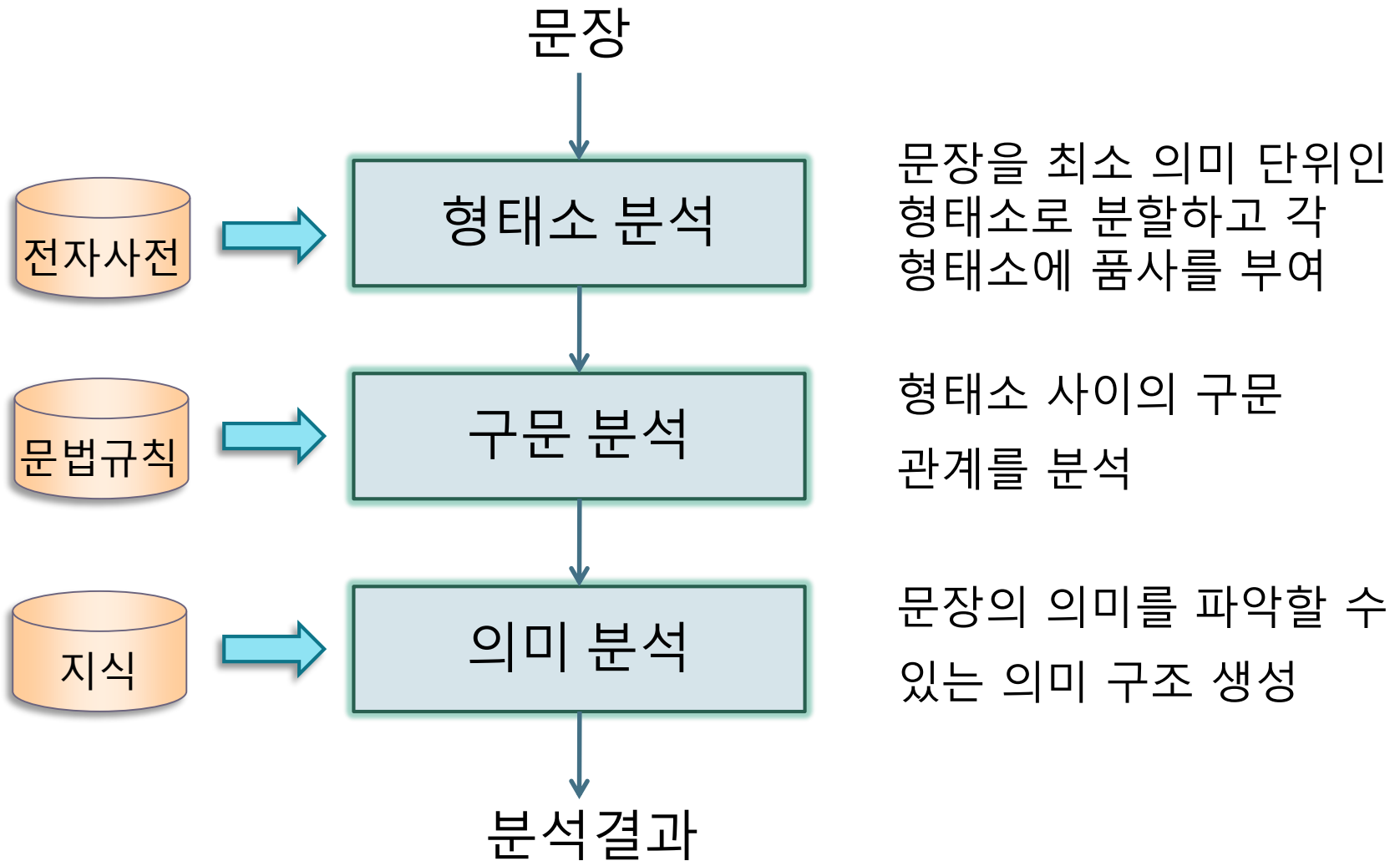
- Eliza 실행 결과



A screenshot of a Windows command prompt window titled "D:\Documents and Settings\Gonzales Cenelia\Desktop\Chatterbot Eliza 2.0\Debug\Eliza.exe". The window shows a text-based conversation between a user and the Eliza chatbot. The user's input is shown in green, and the chatbot's responses are in white. The conversation starts with the chatbot asking for the user's name, followed by a series of questions and answers about the user's name, intentions, and feelings. The chatbot's responses are often generic and repetitive, characteristic of the original Eliza program.

```
C:\ "D:\Documents and Settings\Gonzales Cenelia\Desktop\Chatterbot Eliza 2.0\Debug\Eliza.exe"
HI USER, CAN YOU PLEASE TELL ME YOUR NAME?
>my name is Gonzales.
GLAD TO MEET YOU GONZALES.
>ok Eliza!
PLEASE BE MORE SPECIFIC.
>what do you mean by that?
SORRY I CAN'T PROVIDE MORE EXPLANATION ON THIS.
>why not?
WHY ARE YOU ASKING ME THIS?
>i dont know.
HOW COME YOU DONT KNOW?
>i'm just kidding.
WHAT MAKES YOU THINK THAT YOU ARE JUST KIDDING?
>well, because it is true.
SO YOU ARE TELLING ME THE REASON.
>yes, exactly.
YOU SOUND VERY OPTIMISTIC.
>thank you for the compliment!
DONT MENTION IT.
>ok.
PLEASE BE MORE SPECIFIC.
>i will try to be more specific next time.
DO YOU LIKE TALKING ABOUT YOURSELF?
>so, you think that i was talking about myself.
YOU ARE NOT REALLY TALKING ABOUT ME, ARE YOU?
>well, not really.
WHY ARE YOU SO UNCERTAIN?
```

자연어 처리 과정



형태소 분석

- 문장을 최소 의미 단위인 형태소로 분할하고 각 형태소에 품사를 부여
- “나는 하늘을 나는 새를 보았다”

나는 :

나/NN + 는/JO

나/NP + 는/JO

나/VV + 는/EM

나/VX + 는/EM

날/VV + 는/EM

하늘을 :

하늘/NN + 을/JO

나는 :

나/NN + 는/JO

나/NP + 는/JO

나/VV + 는/EM

나/VX + 는/EM

날/VV + 는/EM

새를 :

새/NN + 를/JO

보았다 :

보/VV + 았/EP + 다/EM

보/VX + 았/EP + 다/EM

형태소 분석

- 형태소 분석을 하기 위해서는 **형태소 사전**이 필요하다.

나 : 나/ NN, 나/ NP, 나/ VV, 나/ VX, 날/ VV

는 : 는/ JO, 는/ EM

다 : 다/ EM

를 : 를/ JO

보 : 보/ VV, 보/ VX

새 : 새/ NN

았 : 았/ EP

을 : 을/ JO

하늘 : 하늘/ NN

... 하늘을 ...

하늘/ NN

을/ JO

하늘을 :

하늘/ NN + 을/ JO

형태소 분석

- 형태소 분석을 하기 위해서는 **접속 정보**도 필요하다.

나 : 나/ NN, 나/ NP, 나/ VV, 나/ VX, 날/ VV

는 : 는/ JO, 는/ EM

다 : 다/ EM

를 : 를/ JO

보 : 보/ VV, 보/ VX

새 : 새/ NN

았 : 았/ EP

을 : 을/ JO

하늘 : 하늘/ NN

... 나는 ...

나/ NN, 나/ NP, 나/ VV, 나/ VX, 날/ VV

는/ JO, 는/ EM

나는 :

나/ NN + 는/ JO

나/ NN + 는/ EM

....

형태소 분석

- 형태소 분석을 하기 위해서는 **접속 정보**도 필요하다.

JO : NN, NP, NU, NX, JO, EM

EM : VV, VX, AJ, AX, SV, EP

EP : VV, VX, AJ, AX, SV

...

... 나는 ...

나/ NN, 나/ NP, 나/ VV, 나/ VX, 날/ VV

는/ JO, 는/ EM

나는 :

나/ NN + 는/ JO

~~나/ NN + 는/ EM~~

....

형태소 사전 : 미등록 형태소 문제

- 미등록 형태소는 필연적으로 존재
 - 언어는 생성/발전/소멸하므로 완전한 사전을 만들 수 없음
- 미등록 형태소 문제
 - 올바른 어절에 대하여 분석 실패 : 차이코프스키에게서
- 미등록 형태소 추정
 - 사전 검색 실패 문자열 = 미등록 형태소 ? / 비형태소 ?
차이코프스키에게서
차이코프스키에게서로서는을
벨기에는
마이크로 프로세서는

형태소 분석의 응용 분야

- 구문 분석의 하위 모듈
- 기계 번역(Machine Translation)
- 정보 검색 (Information Retrieval)
- 문서 요약 (Document Summary)
- 맞춤법 오류 검사 및 교정
- 문자 인식 또는 음성 인식의 후처리
- 용례 및 통계 정보 추출

품사 태깅 (POS tagging)

- 형태론적 중의성

하나의 표층 어절이 두 개 이상의 형태소 분석 결과를 가지는 경우
(예) “나는”

나/NP + 는/JO

날/VV + 는/EM

나/VV + 는/EM

- 품사 태깅 : 형태론적 중의성 해소

(a) 나는 오늘 병원에 가야 합니다.

(b) 하늘을 나는 게 제 꿈입니다.

(c) 삭이 나는 것을 보니 봄이 왔군요.

품사 태깅 (POS tagging)

- “나는 하늘을 나는 새를 보았다”

나는 :

~~나/NN~~

나/NP

~~나/VV~~

~~나/VX~~

날/VV

하늘을 :

하늘/N

나는 :

나/NP + 는/JO

하늘을 :

하늘/NN + 을/JO

나는 :

날/VV + 는/EM

새를 :

새/NN + 를/JO

보았다 :

보/VV + 았/EP + 다/EM

/JO

/JO

/EM

/EM

/EM

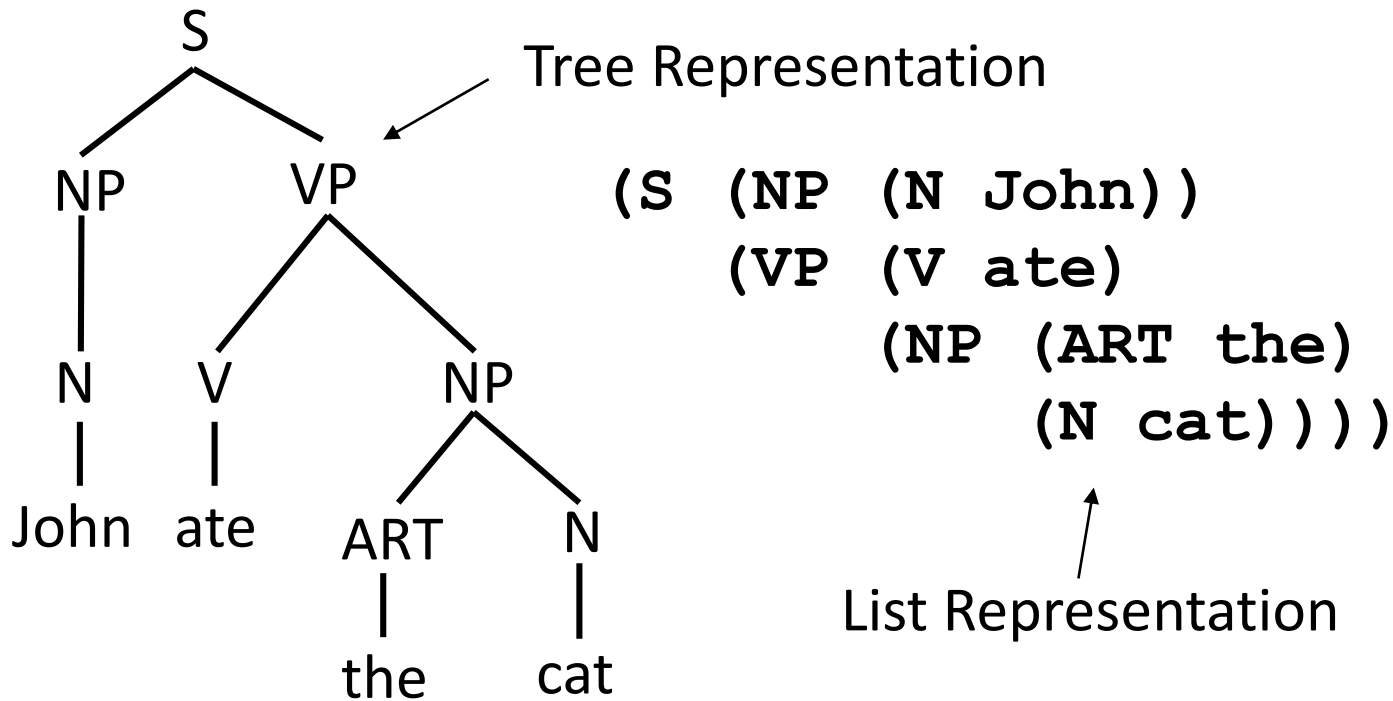
/JO

/EP + 다/EM

~~/EP + 다/EM~~

구문 분석

- 형태소 사이의 구문 관계를 분석
- John ate the cat.



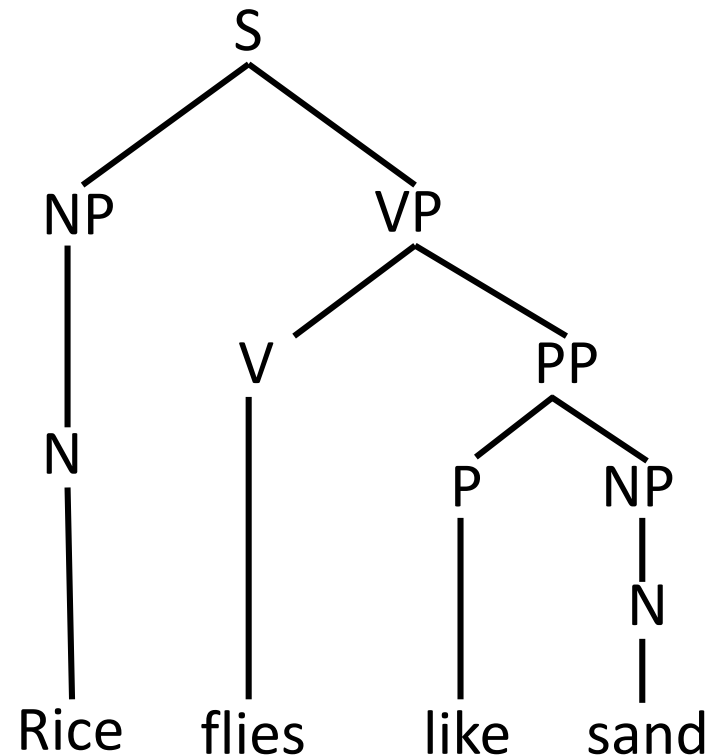
구문 분석

- 구문 분석을 하기 위해서는 문법이 필요하다.

| | |
|-----------------------|-----------------------|
| $S \rightarrow NP VP$ | $N \rightarrow rice$ |
| $VP \rightarrow V NP$ | $N \rightarrow flies$ |
| $VP \rightarrow V PP$ | $N \rightarrow sand$ |
| $NP \rightarrow N$ | $V \rightarrow like$ |
| $NP \rightarrow N N$ | $V \rightarrow flies$ |
| $PP \rightarrow P NP$ | $P \rightarrow like$ |

문법

문법규칙



문법과 파싱

- 문법 (Grammar)

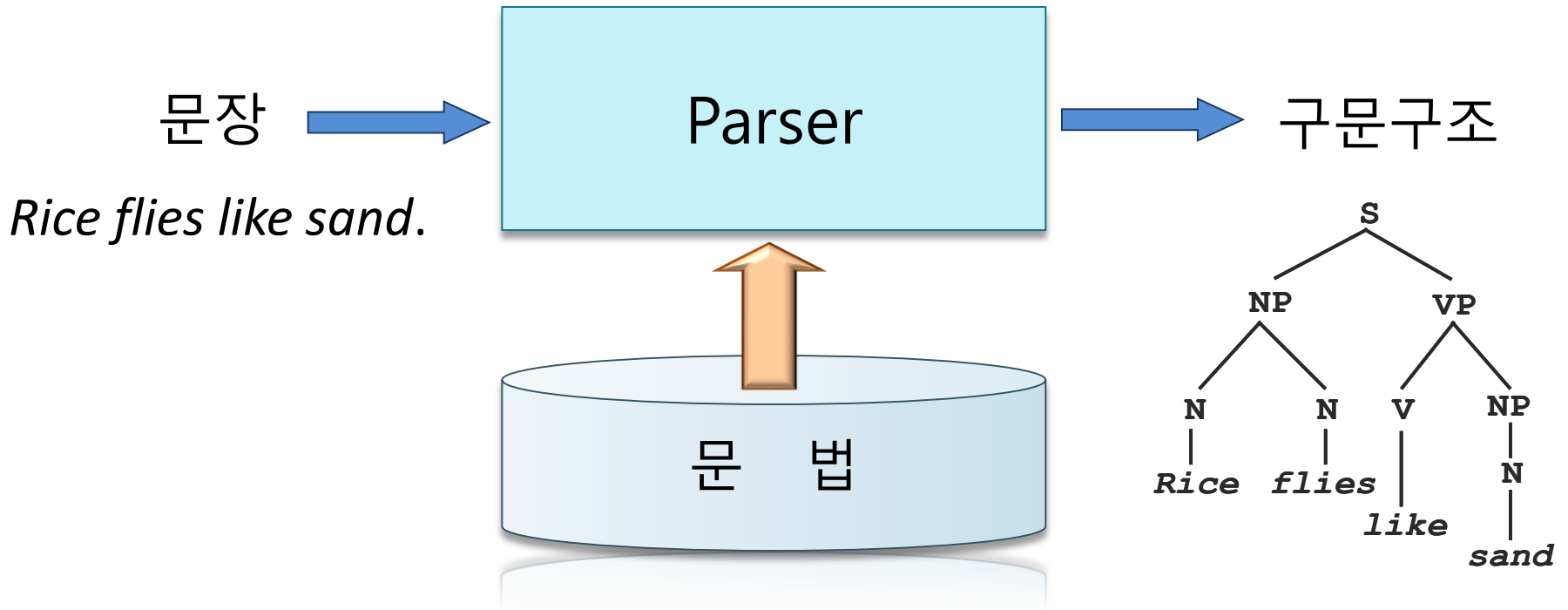
- 언어의 구문 구조를 설명하기 위한 규칙들의 집합
- 문법을 기술하는 다양한 형식들이 있다.
PSG, GPSG, HPSG, LFG, ...

- 파싱 (Parsing)

- 주어진 문법에 따라 문장의 구조를 결정하는 과정을 파싱이라고 한다.
- 파싱을 하는 것과 구문 분석을 하는 것과 같은 의미임.

파서

- 파서(Parser)란 구문 분석을 수행하는 프로그램을 의미한다.



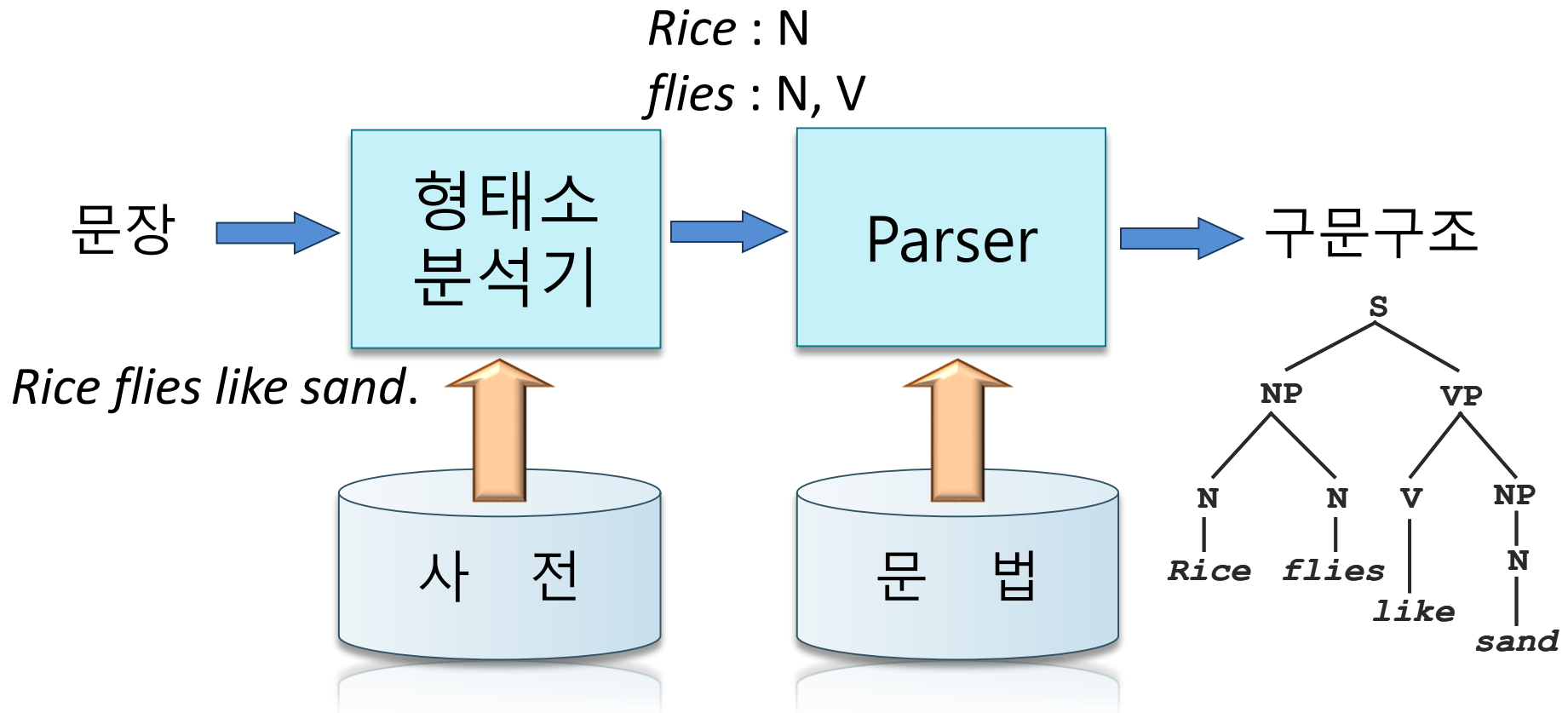
문법

$S \rightarrow NP VP$ $N \rightarrow rice$
 $VP \rightarrow V NP$ $N \rightarrow flies$
 $VP \rightarrow V PP$ $N \rightarrow sand$
 $NP \rightarrow N$ $V \rightarrow like$
 $NP \rightarrow N N$ $V \rightarrow flies$
 $PP \rightarrow P NP$ $P \rightarrow like$

형태소
분석기가
대신함

파싱 과정

- 따라서 일반적인 파싱 과정은 다음과 같다.
- 형태소 분석기란 형태소 분석을 하는 프로그램을 의미한다.

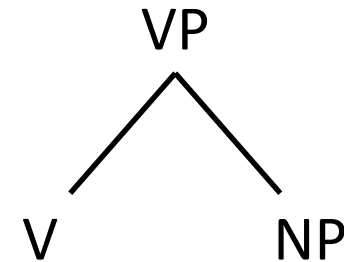


문맥자유문법 (Context Free Grammar)

- context-free rewrite rule들로 이루어짐

VP \rightarrow **V NP**

NP \rightarrow **N**

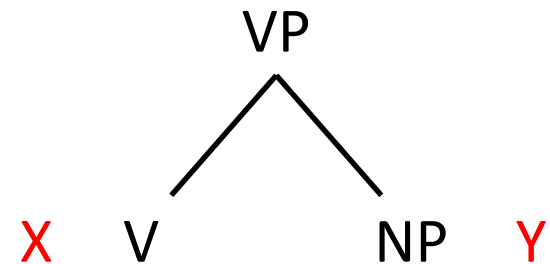


- 효율적인 파싱 알고리즘이 존재함
→ 효율적인 파서 구현 가능
→ 인공 언어의 문법 기술에 사용됨
→ 자연 언어의 문법 기술에 사용됨

- 참고 : context-sensitive grammar

X VP Y \rightarrow **X V NP Y**

Z NP \rightarrow **Z N**



VP \rightarrow **/X/ V NP /Y/** (문맥을 표시하기만 하면 됨)

파싱 방법

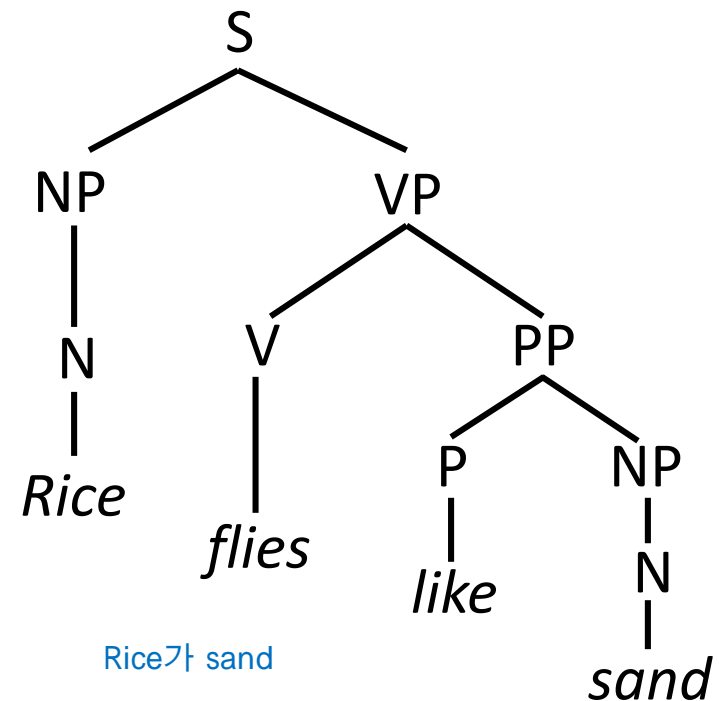
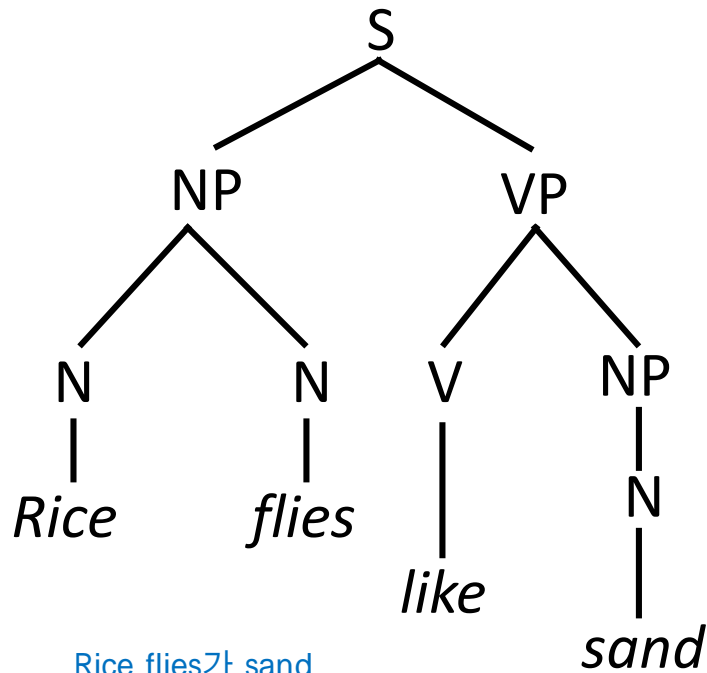
- Top-down parsing
- Bottom-up Parsing
- Chart Parsing 가

구문 구조의 모호성

가

가

- *Rice flies like sand.*



구문 구조의 모호성

- *Time flies like an arrow.*

```
(S (NP (N time))  
   (VP (V flies)  
        (PP (P like)  
             (NP (ART an)  
                  (N arrow))))))
```

```
(S (NP (NP (N time)  
           (N flies))  
   (VP (V like)  
        (NP (ART an)  
             (N arrow)))))
```


구문 구조의 모호성

- *Mary saw the bird with a telescope.*

```
(S (NP (PN Mary))  
   (VP (V saw)  
        (NP (ART the)  
              (N bird))  
        (PP (P with)  
              (NP (ART a)
```

가

```
(S (NP (PN Mary))  
   (VP (V saw)  
        (NP (ART the)  
              (N bird))  
        (PP (P with)  
              (NP (ART a)  
                    (N telescope))))))
```

가

가

통계적 접근 방법

- 말뭉치(corpus) : 대규모 언어 자원
 - Raw Corpus
 - Tagged Corpus
- 말뭉치로부터 얻어진 통계 정보를 자연어 처리에 활용함.
 - **strong** tea (1) heavy (2) powerful (3) strong (4) thick
 - **heavy** smoker (1) heavy (2) powerful (3) strong (4) serious
- 말뭉치로부터 기계 학습을 하고 학습 결과를 자연어 처리에 활용함.
- (말)뭉치 언어학 (corpus linguistics)

품사 태깅 (POS tagging)

- 품사 태깅된 말뭉치를 사용하여 다음과 같은 통계 정보를 얻는다.

$$P(t_i | t_{i-1}) = \frac{C(t_{i-1}, t_i)}{C(t_{i-1})} \quad P(w_i | t_i) = \frac{C(w_i, t_i)}{C(t_i)}$$

- 다음에서 play의 품사를 결정하려 한다고 하자.

[a/AT new/JJ play/NN? VV?]

- 만약 $P(\text{NN} | \text{JJ}) \gg P(\text{VV} | \text{JJ})$ 이라면 play가 NN이라는 결정을 할 수 있을 것이다.
- 만약 $P(\text{play} | \text{VV}) \gg P(\text{play} | \text{NN})$ 이라면 play가 VV라는 결정을 할 수 있을 것이다.

품사 태깅 (POS tagging)

- 문장 전체에 대하여 다음을 만족하는 품사 열을 구한다.

$$\hat{t}_{1,n} = \underset{t_{1,n}}{\operatorname{argmax}} P(t_{1,n} | w_{1,n}) = \underset{t_{1,n}}{\operatorname{argmax}} \prod P(w_i | t_i) P(t_i | t_{i-1})$$

$w_{1,n}$ 은 n 개의 단어로 이루어진 문장을 나타낸다.

$t_{1,n}$ 은 n 개의 단어 각각에 주어진 품사를 나타낸다.

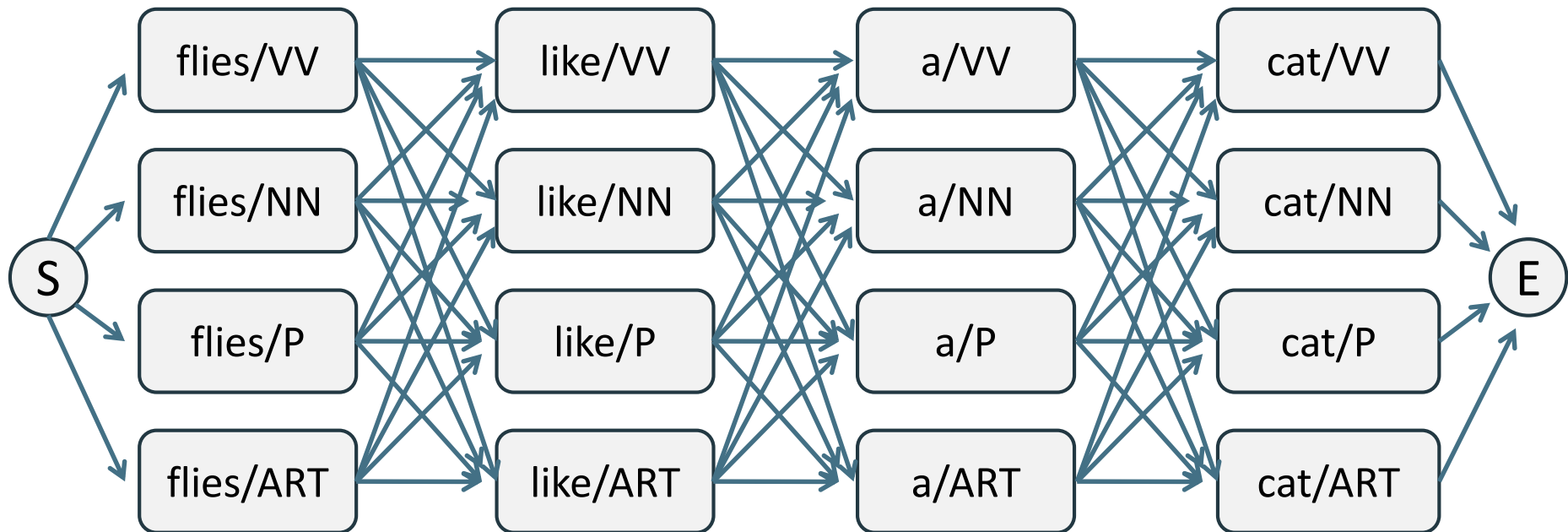
- 다음에서 play의 품사를 결정하려 한다고 하자.

[a/AT new/JJ play/NN? VV?]

- $P(\text{NN} \mid \text{JJ}) \times P(\text{play} \mid \text{NN}) \gg P(\text{VV} \mid \text{JJ}) \times P(\text{play} \mid \text{VV})$ 라면 play가 NN이라는 결정을 한다.

품사 태깅 (POS tagging)

- 태깅 : S에서 E에 이르는 최적 경로를 찾는 문제



- 위와 같은 방법의 정확도는 대략 96-98% 정도이다.