

5.1 문맥자유문법(cfg)과 유도(derivation), 언어(language)

(정의 5.1) 문맥자유(Context-free) 문법(grammar)¹⁾ $G = (N, T, P, S)$ 는

- (1) N^2 은 nonterminal 혹은 variable³⁾이라 불리는 문자(symbol)에 집합이다.
- (2) T 는 terminal 혹은 입력문자라 불리는 문자에 집합이다.
단 $N \cap T = \emptyset$ 이고 $V = N \cup T$ 로 쓰고 V 를 문법의 기본문자라 부르자.
- (3) P 은 (문법) 규칙(rule, production)이라고 부르는 순서쌍 (A, α) 의 집합이다.
규칙 순서쌍 $(A, \alpha) \in P$ 는 $A \rightarrow \alpha \in P$ ⁴⁾ 로 쓰이기도 하고
규칙 좌변은 $A \in N$ ⁵⁾이고 우변은 $\alpha \in (N \cup T)^* = V^*$ 이다.
- (4) $S \in N$ 은 처음(start, axiom)문자라 부르는 특별한 년 터미널이다.

(정의 5.2) 년 터미널 $A \in N$ 를 규칙의 좌변으로 가지는 규칙이, $A \rightarrow \alpha_1, A \rightarrow \alpha_2, \dots, A \rightarrow \alpha_k$ 일 때 $A \rightarrow \alpha_1, A \rightarrow \alpha_2, \dots, A \rightarrow \alpha_k$ 를 A 규칙(A -productions)라고 부르고, 짧게 $A \rightarrow \alpha_1 \mid \alpha_2 \mid \dots \mid \alpha_k$ 로 쓰기도 한다.

(예 5.1) 영어 문법 중 3형식 문장 중 일부를 문법 $G_3 = (N, T, P, \langle \text{문장} \rangle)$ 로 표현해보자.

- $$N = \{ \langle \text{문장} \rangle, S, V, O, \langle \text{관사} \rangle, \langle \text{명사} \rangle, \langle \text{대명사} \rangle, \langle \text{타동사} \rangle, \langle \text{주격대명사} \rangle, \langle \text{목적격대명사} \rangle \}$$
- $$T = \{ \text{the, a, boy, girl, 예쁜이, I, you, he, she, love, loves, me, him, her} \}$$
- $$P = \{ \langle \text{문장} \rangle \rightarrow SVO, \\ S \rightarrow \langle \text{관사} \rangle \langle \text{명사} \rangle \mid \langle \text{주격대명사} \rangle, \\ V \rightarrow \langle \text{타동사} \rangle, \\ O \rightarrow \langle \text{관사} \rangle \langle \text{명사} \rangle \mid \langle \text{목적격대명사} \rangle, \\ \langle \text{관사} \rangle \rightarrow \text{the} \mid \text{a} \mid \epsilon, \\ \langle \text{명사} \rangle \rightarrow \text{boy} \mid \text{girl} \mid \text{예쁜이}, \\ \langle \text{타동사} \rangle \rightarrow \text{love} \mid \text{loves}, \\ \langle \text{주격대명사} \rangle \rightarrow \text{I} \mid \text{you} \mid \text{he} \mid \text{she}, \\ \langle \text{목적격대명사} \rangle \rightarrow \text{me} \mid \text{you} \mid \text{him} \mid \text{her} \}$$

문맥자유 문법규칙의 좌변은 년 터미널 문자(N) 하나이고 우변은 년 터미널(N)이나 터미널(T) 문자 여럿(기본문자열(V^*))이다. 따라서 문법규칙을 사용하면 터미널과 년 터미널 문자들이 여러 개 나타난다. 그 중에 터미널은 문법규칙에 좌변에는 올 수 없음에 유의하라.

처음에는 년 터미널 S 에서 시작하여 문법규칙 P 중에 처음 문자 S 가 좌변인 S 규칙 $S \rightarrow \sigma$ 를

-
- 1) 언어학자이고 철학자이며 전산학에도 큰 영향을 준 N. Chomsky가 1950년대 말 처음으로 시작하였다. 생성문법(generative grammar)이라고 부르기도 한다.
 - 2) 교과서에서는 N 대신 V 를 쓰고 있지만, 우리는 V 를 다른 용도($N \cup T$)로 쓰기 위하여 N 을 쓴다.
 - 3) Syntactic category라고 부르기도 한다.
 - 4) 문법규칙 P 를 $P \subseteq N \times (N \cup T)^*$ 로 정의할 수도 있다.
 - 5) 이것이 문맥자유(context-free)라고 부르는 이유이다.
 - 6) S 규칙이 하나 이상일 수 있으므로 이 유도과정은 nondeterministic하다. σ 는 그리스 문자로 영어 소 문자 s 에 해당한다.

찾아 S 를 그 규칙의 우변 $\sigma \in (N \cup T)^* = V^*$ 로 바꾸고, 바꾼 기본문자열 σ 에 n 터미널 문자 $A \in N$ 이 있으면 다시 A 규칙에서 찾아 그 A 규칙의 우변으로 바꾸는 과정의 연속이 문맥자유 문법의 유도이다. 이 유도는 기본문자열이 n 터미널 문자를 포함하지 않으면 끝난다.

(정의 5.3) 유도(derivation) \Rightarrow 는 기본문자열 V^* 에서 정의된 관계($\Rightarrow \subseteq V^* \times V^*$)이다.

문법 $G = (N, \Sigma, P, S)$ 에서 $\alpha, \gamma \in V^*$, $B \in N$, $B \rightarrow \beta \in P$ 라 하자. 이 때 기본문자열 $\alpha B \gamma$ 가 기본문자열 $\alpha \beta \gamma$ 를 유도한다(derive)하고 $\alpha B \gamma \Rightarrow_G \alpha \beta \gamma$ 로 쓰고 문법 G 가 잘 알려져 있으면 G 를 생략하고 \Rightarrow 로만 쓰기도 한다.

(정의 5.4) 유도 \Rightarrow 의 반복 $\Rightarrow^n (n \geq 0)$ 를 아래와 같이 recursive하게 정의한다.

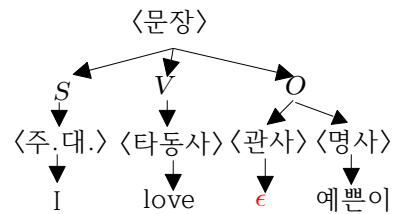
$$\begin{aligned} \Rightarrow^0 &\stackrel{\text{def}}{=} \{id_V\}, \\ \Rightarrow^n &\stackrel{\text{def}}{=} \bigcup_{R \in P} \Rightarrow^{n-1} \circ \Rightarrow \quad n \geq 1. \end{aligned}$$

(정의 5.5) 유도 \Rightarrow 의 반복 합 \Rightarrow^* 를 아래와 같이 정의한다.

$$\Rightarrow^* \stackrel{\text{def}}{=} \bigcup_{i \in N_0} \Rightarrow^i = \Rightarrow^0 \cup \Rightarrow^1 \cup \Rightarrow^2 \cup \dots \quad \text{단 } N_0 \stackrel{\text{def}}{=} \{0, 1, 2, \dots\}.$$

(정의 5.6) 문법 $G = (N, T, P, S)$ 에서 $S \Rightarrow^* \alpha (\alpha \in V^*)$ 이면 기본문자열 α 를 **문장형태 (sentential form)**이라하고, 특히 문장형태 $S \Rightarrow^* x$ 가 입력문자열($x \in T^*$)만으로 이루어져 있을 때 **문장(sentence)**이라 한다.

(예 5.2) $\langle \text{문장} \rangle \Rightarrow SVO \Rightarrow \langle \text{주격대명사} \rangle VO$
 $\Rightarrow \langle \text{대명사} \rangle VO \Rightarrow I VO \Rightarrow I \langle \text{타동사} \rangle O$
 $\Rightarrow I \text{ love} \langle \text{관사} \rangle \langle \text{명사} \rangle \Rightarrow I \text{ love } \epsilon \langle \text{명사} \rangle$
 $= I \text{ love} \langle \text{명사} \rangle \Rightarrow I \text{ love 예쁜이}$



문장 "I love 예쁜이"의 파스(parse) 나무(tree)

(정의 5.7) 문법 $G = (N, T, P, S)$ 의 **문장들**의 집합을 문법의 **언어** $L(G)$ 라 하고, 아래와 같이 정의한다.

$$L(G) = \{x \in T^* \mid S \Rightarrow^* x\}.$$

(정의 5.8) 문맥자유(context-free) 언어(language)

임의의 언어 L 을 만들어내는 **문맥자유문법** G 가 있을 때, $L = L(G)$, 언어 L 을 **문맥자유언어**라 부른다.

(정의 5.9) Regular(정규) Grammar $G = (N, T, P, S)$ 는

- (1) N nonterminal어휘, (2) T 는 terminal어휘는 context-free grammar와 같다.
- (3) P 은 (문법) 규칙(rule, production)은 $P \subseteq N \times (T^* \cdot N \cup T^*)$ 로서 $A, B \in N$ 이고 $x, y \in T^*$ 일 때 순서쌍 (A, xB) 또는 (A, y) 에 모양이다. 규칙 순서쌍 (A, xB) 또는 $(A, y) \in P$ 를 $A \rightarrow xB$ 또는 $A \rightarrow y \in P$ 라 쓴다.

(4) $S \in N$ 은 처음(start, axiom)문자도 cfg와 같다.

문맥자유 문법의 문법규칙은 $P_{cfg} \subseteq N \times (N \cup T)^*$ 이고 정규문법의 문법규칙은 $P_{rg} \subseteq N \times (T^* \cdot N \cup T^*)$ 인데, 문법규칙의 좌변은 년 터미널 N 으로 같고, 우변은 정규문법($T^* \cdot N \cup T^*$)이 문맥자유문법($N \cup T$)의 부분집합이므로 정규문법은 문맥자유문법의 한 종류라고 볼 수 있다.

(예 5.3) Palindrome 문법 $G_{pal} = (\{P\}, \{0, 1\}, P_{pal}, P)$ 은 문맥자유문법이다.

$$P_{pal}: P \rightarrow \epsilon \mid 0 \mid 1 \mid 0P0 \mid 1P1.$$

(정리 5.1) 임의의 finite automaton $A_{fa} = (Q, \Sigma, \delta, q_0, F)$ 와 equivalent한 정규문법 $G_{rg} = (N, T, P, S)$ 을 다음과 같이 만들 수 있다(역도 가능; and vice versa).

$$(1) Q \leftrightarrow N: q \in Q \quad \leftrightarrow A_q \in N$$

$$(2) \Sigma \quad \quad \quad = T.$$

$$(3) \delta, F \leftrightarrow P: p \in \delta(q, x) \quad \leftrightarrow A_q \rightarrow x A_p \in P$$

$$f \in F \quad \quad \quad \rightarrow A_f \rightarrow \epsilon \in P$$

$$f \in \delta(q, x), f \in F \leftarrow A_q \rightarrow x \in P$$

$$(4) q_0 \in Q \quad \quad \quad \leftrightarrow A_{q_0} = S \in N$$

(최종증명) $L(A_{fa}) = L(G_{rg})$.

(증명) $\forall x \in \Sigma^*: p \in \delta^*(q, x)$, iff $A_q \Rightarrow^* x A_p$.

$$(증명1) \forall x \in \Sigma^*: p \in \delta^*(q, x) \Rightarrow A_q \Rightarrow^* x A_p.$$

$$(증명2) \forall x \in \Sigma^*: A_q \Rightarrow^* x A_p \Rightarrow p \in \delta^*(q, x).$$

(최종증명) $\forall x \in \Sigma^*: \delta^*(q_0, x) \in F \Leftrightarrow S \Rightarrow^* x$.

(중요사실) 오토마타의 클래스 \mathbb{M}_{FA} (Finite State automata; 유한상태기계)와 문법의 클래스 \mathbb{G}_{RG} (정규문법; Regular Grammars) 같다.

(사실 5.1) 정규문법은 문맥자유문법의 적절한 하위계급(properly contained class)이다.

(정리 5.2) 문맥자유언어도 정규언어의 적절한 상위계급이다.

5-1 문맥자유문법(Context-Free Grammar)과 정규문법(Regular Grammar)

KAIST 전산학과 최광무

