

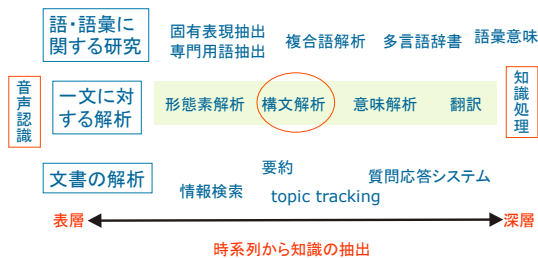
言語解析理論

講師 竹内孔一

本日の内容

- 文の繰り返し構造を扱う理論
 - 形式的な言語の扱い方
 - 文脈自由(型)文法
 - 文法クラスの位置付け
 - チョムスキー階層
 - もう一つの数学モデル
 - オートマトン(automaton)
- 生成できる言語の違い

言語研究の相関



言語の理論

- 形式言語
 - 形式的な言語の側面を捉える
 - 言葉の意味は捨象して形式に注目した語の性質
- 入れ子の関係 (句)
- 句構造文法
 - 例) 私は その本を 読んだ
 - 彼の姉は この本を 読んでました
 - 刑事は 彼がその犯人であることを 知った
 - (主語)(助詞) (目的語)(助詞) (述語)

文脈自由文法

- 形式化
 - 入れ子構造を捉える

非終端記号の集合 V_N

終端記号の集合 V_T

生成規則の集合 P

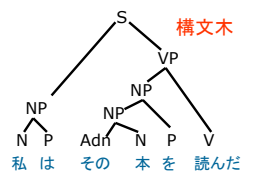
初期記号 σ

$G = \langle V_N, V_T, P, \sigma \rangle$

$V_N = \{S, NP, VP, P, Adn, V\}$

$V_T = \{\text{私, は, その, 本, を, 読んだ}\}$

$\sigma = S$



$P = \{S \rightarrow NP VP, NP \rightarrow N P, NP \rightarrow NP P, NP \rightarrow Adn N, VP \rightarrow NP V, N \rightarrow \text{私}, N \rightarrow \text{本}, \dots, P \rightarrow \text{を}\}$

練習3

- 次の文の構文木を完成させ生成規則を記述せよ
- 例文) 「彼はフランスに行った」
- ただし $V_t = \{\text{彼, は, フランス, に, 行った}\}$ とする
- V_n, σ, P は各自決めること

文脈自由文法

- 理論的な特徴と限界

- 文法の記述に制限がある

P について $A \rightarrow \beta \quad A \in V_N \quad \beta \in (V_N \cup V_T)^+$

- 非終端記号がどこでも書き換え可能 $\rightarrow$ 文脈自由
 - (参考) NP $\rightarrow$ N P などすべてこの範囲

生成規則の形を制限 $\rightarrow$ 生成する言語が異なる

4種類の規則に分類 Chomsky 1959
 $\rightarrow$ 各文法の限界を示した
 チョムスキー階層

チョムスキー階層

- 生成規則による言語の能力の違い

文法	<=等価=>	オートマトン	能力大
0型言語 0型文法		チューリングマシン (汎用計算機)	
$\alpha \rightarrow \beta$			
1型言語 文脈依存文法		線形拘束オートマトン	
$\alpha \rightarrow \beta \quad \alpha \leq \beta $			
2型言語 文脈自由文法		プッシュダウンオートマトン	
$A \rightarrow \beta \quad A \in V_N \quad \beta \in V^+$			
3型言語 正規文法		有限オートマトン	能力小
$A \rightarrow aB \quad A, B \in V_N$ $A \rightarrow a \quad a \in V_T$			

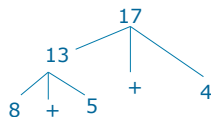
問題

- 8+5+4 といった和の計算のための文法は
 チョムスキー階層のどれにあたるか?

- 文脈自由文法

- $V_t = \{8, 5, 4, +\}$

- $V_n = \{13 \rightarrow 8 + 5, 17 \rightarrow 13 + 4\}$



計算の本質は書き換え

能力の違い

- 以下の2つを取り上げて説明

文法と生成言語について説明する

- 正規文法 regular grammar

- 有限オートマトン

finite automaton (fa)

- 文脈自由文法 context free grammar

- プッシュダウンオートマトン

push down automaton (pda)

正規文法

- 最も制約された文法 (regular grammar)

- 生成規則

$A \rightarrow aB \quad A \rightarrow a \quad A, B \in V_N \quad a \in V_T$

- ポイント a (文字) を必ず含む

- 規則と生成する文字列との関係

例) 「平成□□年」を受け付ける正規文法

$V_N = \{S, Y, A\}$

$V_T = \{\text{平成}, \text{年}, 1, 2, \dots, 27\}$

$\sigma = S$ 単語ベース

$P = \{S \rightarrow \text{平成 } Y, \\ Y \rightarrow 1 A, \\ Y \rightarrow 2 A, \\ \vdots \\ Y \rightarrow 27 A, \\ A \rightarrow \text{年}\}$

練習4

- 正規文法

- 「平成□□年」を受け付ける正規文法を記述せよ
 ただし文字ベースとする

$V_T = \{\text{平}, \text{成}, \text{年}, 0, 1, 2, \dots, 9\}$

正規表現との対応は?

- 正規表現を正規文法で表すと

例) $a^* = \{\epsilon, a, aa, aaa, aaaa, \dots\}$

この場合正規文法は $A \rightarrow \epsilon, A \rightarrow aA$

下記の場合は??

$a^+ = \{a, aa, aaa, aaaa, \dots\}$

$(a+b)^+ = \{a, b, aa, aaa, bb, bbb, ab, abb, \dots\}$

13

過剰生成について

- 受理する目的で作成

- それ以外のものも受け付けることが多い

例) 「平成□□年」を受理する正規文法で記述する

$G = (V_N, V_T, P, \sigma)$

$V_T = \{\text{平, 成, 年, 0, 1, 2, \dots, 9}\}$

$V_N = \{S\}$

$\sigma = S$

$P = \{S \rightarrow \text{平 } S, S \rightarrow \text{成 } S,$

$S \rightarrow 0 S, S \rightarrow 1 S,$

$S \rightarrow 2 S, \dots, S \rightarrow 9 S,$

$S \rightarrow \text{年}\}$

文法Gが生成(受理)する
言語L(G)の一例として

平成05年, 平成119年... ← 過剰生成



有限オートマトン

- 特徴
 - 内部の状態と外部からの入力で次の状態を決める機械
- 種類
 - 決定性有限オートマトン
(deterministic finite automaton (fa, dfa))
有限オートマトンというと普通はこちら
 - 非決定性有限オートマトン
(non-deterministic finite automaton (nfa))
→ nfa は fa に書き下すことができる
- 重要な関係
 - 正規文法と等価な言語生成能力

有限オートマトン

- 定義

状態集合を $K = \{q_0, q_1, q_2, \dots, q_n\}$ とすると

- 状態の集合 K

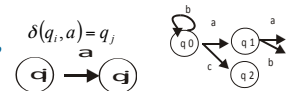
- 入力文字の集合 S

- 状態遷移関数の集合 P

- 初期状態 q_0

- 最終状態の集合 F

$M = \langle K, S, P, q_0, F \rangle$ 文字 a を受け付けて状態 q_i から q_j に遷移する
→ ある手順での決まった規則を書くのに適している

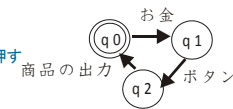


- 応用例

- 自動販売機のモデル

お金を入れる → ボタンを押す

→ 商品がでる



有限オートマトン

- 例題
 - 「平成□□年」を受理するオートマトンを作成せよ

解答例

- 単語ベースの場合

$S = \{\text{平成}, 1, 2, 3, \dots, 27, \text{年}\}$

正規文法

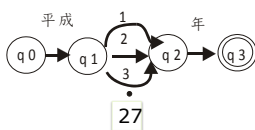
$P = \{S \rightarrow \text{平成 } Y,$

$Y \rightarrow 1 A,$

$Y \rightarrow 2 A,$

$Y \rightarrow 27 A,$

$A \rightarrow \text{年}\}$



最終状態q3に行かなければ不受理

練習5

- 有限オートマトン

- 文字ベース $S = \{\text{平, 成, 0, 1, 2, \dots, 9, 年}\}$ として

「平成□□年」を受理するオートマトンを作成せよ

(過剰生成しないモデルを目指して)