8 形式言語論 （1） –語と文法–

形式言語の研究は, 計算機科学の根幹をなす. 現在の計算機科学の基礎は, その上の自然言語の文法を数学的に定式化するところから出発した. 計算機科学での大きなテーマは「計算機の本質的な能力とその限界とはなにか」という疑問への回答を与えることである. ここからは, これらの数学的な基礎となる部分を学ぼう.

• 8-1 : アルファベットと言語

文字と文字列は, 計算機科学において基本的な概念である. 文字列は, あるアルファベット上で定義される. そのアルファベットは状況に応じて様々なものを取りうる. きちんと定義しよう.

定義 8.1. 空でない有限集合を **アルファベット** と呼ぶ. アルファベット Σ の要素を **文字** といい, 文字を並べた有限列を **文字列** と呼ぶ. Σ 上の文字列全体からなる集合を Σ^* で書く. 文字列 $w \in \Sigma^*$ に含まれる文字の数をその **長さ** と呼び, これを $\ell(w)$ で表す. 長さが 0 の文字列を **空列** と呼び, 特別な記号 ε で表す. 空列ではない文字列全体を $\Sigma^+ := \Sigma^* \setminus \{\varepsilon\}$ で表す.

ここでは, アルファベットの文字をタイプライタフォントで表すことにする.

例 8-1 (1) アルファベット $\Sigma = \{\texttt{a},\texttt{b},\texttt{c},\ldots,\texttt{y},\texttt{z}\}$ とする. このとき, $w = \texttt{risansuugaku}$ は Σ 上の文字列であり, $\ell(w) = 12$ である.
(2) $\Sigma = \{\texttt{0},\texttt{1}\}$ であるとき, $\Sigma^* = \{\varepsilon, \texttt{0},\texttt{1},\texttt{00},\texttt{01},\texttt{11},\texttt{000},\ldots\}$ となり, $\Sigma^+ = \{\texttt{0},\texttt{1},\texttt{00},\texttt{01},\texttt{11},\texttt{000},\ldots\}$ である.

アルファベット Σ 上の **言語** とは, Σ^* の部分集合のことをいう. 形式言語論における興味は, 与えられたアルファベット上の「言語」の表現方法である. より具体的には

- 言語 $\mathcal{L}$ に属する文字列はどのように特徴付けられるか.
- 言語 $\mathcal{L}$ を特徴づけるような表現は, ある意味「有限個」のルールだけで決めることができるか.

そこで, 以降ではある規則に従って得られる言語がどのようなものがあるかを調べて行くことにしよう.

• 8-2 : 0 型文法

定義 8.2. **0 型文法** （以後, 単に **文法**）とは, 以下の要素からなる 4 つ組 $G = (V_N, V_T, P, S)$ である.

- V_N は有限集合であり, その元を **変数** と呼ぶ.
- V_T は V_N と共通部分を持たない有限集合であり, その元を **終端記号** と呼ぶ.
- P は「$\alpha \to \beta$」 $(\alpha \in (V_N \cup V_T)^+,\ \beta \in (V_N \cup V_T)^*)$ という形の式からなる有限集合であり, P の元を **生成規則** と呼ぶ.
- $S \in V_N$ であり, **開始記号** と呼ぶ.

次に文法が決まったときに生成する言語を定義しよう.
$G = (V_N, V_T, P, S)$ を文法とし, $V = V_N \cup V_T$ とおく. 生成規則 “$\alpha \to \beta$” $\in P$ であったとする. このとき, $u, v \in V^*$ に対して $u\alpha v$ は $u\beta v$ に **移る** といい, 記号で

$$u\alpha v \underset{G}{\Longrightarrow} u\beta v$$

とかく. この状況を, 生成規則 $\alpha \to \beta$ が文字列 $u\alpha v$ に適用されて $u\beta v$ を得た, という. $u, v \in V^*$ に対して,

担当：茨城大学 情報科学領域　宮本 賢伍

$u = v$ であるか, ある $u_1, u_2, \ldots, u_n \in V^*$ が存在して

$$u \underset{G}{\Longrightarrow} u_1 \underset{G}{\Longrightarrow} u_2 \underset{G}{\Longrightarrow} \cdots \underset{G}{\Longrightarrow} u_n \underset{G}{\Longrightarrow} v$$

であるならば, u は v を **導出する** といい, 記号で $u \overset{*}{\underset{G}{\Longrightarrow}} v$ と表す. つまり, u から出発して 0 回以上 P に属する生成規則を適用することで v が得られることをいう. こうして, 言語 $\mathcal{L}(G)$ を

$$\mathcal{L}(G) := \{w \in V^* \mid S \overset{*}{\underset{G}{\Longrightarrow}} w\}$$

と定義し, これを **G が生成する言語** と呼ぶ.

例 8-2　文法 $G = (V_N, V_T, P, S)$ を以下で定義する:

$$V_N = \{S\}, \quad V_T = \{\mathtt{0},\mathtt{1}\}, \quad P = \{S \to \mathtt{0}S\mathtt{1},\ S \to \mathtt{01}\}$$

このとき, S が変数で 0,1 が終端記号である. 生成規則 P の左辺は S であり, $S \to \mathtt{0}S\mathtt{1}$ を適用すれば, S の個数は 1 つのまま変わらず, $S \to \mathtt{01}$ を適用すれば, S がなくなる. 従って, 生成規則は $S \to \mathtt{0}S\mathtt{1}$ を何度か適用した後に $S \to \mathtt{01}$ を適用する他ない. 開始記号 S に生成規則 $S \to \mathtt{0}S\mathtt{1}$ を $n-1$ 回適用すると,

$$S \underset{G}{\Longrightarrow} \mathtt{0}S\mathtt{1} \underset{G}{\Longrightarrow} \mathtt{00}S\mathtt{11} \underset{G}{\Longrightarrow} \mathtt{000}S\mathtt{111} \underset{G}{\Longrightarrow} \cdots \underset{G}{\Longrightarrow} \overbrace{0\cdots0}^{n-1} S \overbrace{1\cdots1}^{n-1}$$

これの最後に生成規則 $S \to \mathtt{01}$ を適用すると $\overbrace{0\cdots0}^{n}\overbrace{1\cdots1}^{n}$ となる. この文字列を $\mathtt{0}^n\mathtt{1}^n$ と表すことにすれば, $\mathcal{L}(G) = \{\mathtt{0}^n\mathtt{1}^n \mid n \geq 1\}$ であることがわかった.

レポート 8-1　次の文法 $G = (V_N, V_T, P, S)$ について, $\mathtt{a}^n\mathtt{b}^n\mathtt{c}^n \in \mathcal{L}(G)$ であることを示せ. ただし, $V_N = \{S, \mathtt{B}, \mathtt{C}\}$, $V_T = \{\mathtt{a},\mathtt{b},\mathtt{c}\}$ であり, P は次の 6 つの生成規則からなる.

(i) $S \to \mathtt{a}S\mathtt{BC}$　(ii) $S \to \mathtt{aBC}$　(iii) $\mathtt{CB} \to \mathtt{BC}$
(iv) $\mathtt{aB} \to \mathtt{ab}$　(v) $\mathtt{bB} \to \mathtt{bb}$　(vi) $\mathtt{bC} \to \mathtt{bc}$
(vii) $\mathtt{cC} \to \mathtt{cc}$

また, $\mathcal{L}(G) = \{\mathtt{a}^n\mathtt{b}^n\mathtt{c}^n \mid n \geq 1\}$ となる理由を考えよ. つまり, $\mathtt{a}^n\mathtt{b}^n\mathtt{c}^n$ 以外の形はありえない理由を考えよ.

• 8-3 : 基本文法

0 型文法の生成規則にある種の制限を加える事によって新たな文法を定義することができる. ここで定義する文法は合わせて基本文法と呼ばれる.

定義 8.3. 文法 $G = (V_N, V_T, P, S)$ を考える.

(1) G の任意の生成規則 $\alpha \to \beta$ について, $\ell(\alpha) \leq \ell(\beta)$ が成り立つとき, G を **単調文法** と呼ぶ. 単調文法によって生成される言語 $\mathcal{L}(G)$ を **第 1 型言語** と呼ぶ.
(2) G の任意の生成規則 $\alpha \to \beta$ について, α に現れる変数は 1 つであり, β は空列ではないとき, G を **文脈自由文法** と呼ぶ. 文脈自由文法によって生成される言語 $\mathcal{L}(G)$ を **文脈自由言語** と呼ぶ.
(3) G の任意の生成規則が変数 A,B と終端記号 a を用いて $\mathtt{A} \to \mathtt{aB}$ もしくは $\mathtt{A} \to \mathtt{a}$ の形であるとき, G を **正規文法** と呼ぶ. 正規文法によって生成される言語 $\mathcal{L}(G)$ を **正規言語** と呼ぶ.

例 8-3　(1) **例 8-2** で定義した文法 G は単調文法である.
(2) 次で定義する文法 $G = (V_N, V_T, P, S)$ は文脈自由文法である. ここで, $V_N = \{S, \mathtt{A}, \mathtt{B}\}$, $V_T = \{\mathtt{a},\mathtt{b}\}$ であり, P は次の 8 つの生成規則からなる.

(i) $S \to \mathtt{aB}$　(ii) $S \to \mathtt{bA}$　(iii) $\mathtt{A} \to \mathtt{a}$　(iv) $\mathtt{A} \to \mathtt{a}S$
(v) $\mathtt{A} \to \mathtt{bAA}$　(vi) $\mathtt{B} \to \mathtt{b}$　(vii) $\mathtt{B} \to \mathtt{bS}$　(viii) $\mathtt{B} \to \mathtt{aBB}$

(3) 次で定義する文法 $G = (V_N, V_T, P, S)$ は正規文法である．ここで, $V_N = \{S, \mathtt{A}, \mathtt{B}\}$, $V_T = \{\mathtt{a}, \mathtt{b}\}$ であり, P は次の 9 つの生成規則からなる．

(i) $S \to \mathtt{a}$　(ii) $S \to \mathtt{aA}$　(iii) $S \to \mathtt{bB}$　(iv) $\mathtt{A} \to \mathtt{aA}$
(v) $\mathtt{A} \to \mathtt{a}S$　(vi) $\mathtt{A} \to \mathtt{bB}$　(vii) $\mathtt{B} \to \mathtt{bB}$　(viii) $\mathtt{B} \to \mathtt{b}$
(ix) $\mathtt{B} \to \mathtt{a}$

命題 8.4. 単調文法 $G = (V_N, V_T, P, S)$ に対して, 次の条件を満たす単調文法 G が存在する．

- $\mathcal{L}(G) = \mathcal{L}(G')$
- G' の開始記号 S' は, G' のどの生成規則の右辺にも現れない．

この主張は, 文脈依存文法を文脈自由文法, 正規文法にしても成り立つ．

証明. まず, 記号 S' は $V_N \cap V_T$ に属していない新たな記号とする．文法 $G' = (V'_N, V'_T, P', S')$ を次のように定義しよう．変数の集合は $V'_N := V_N \cup \{S'\}$, 終端記号の集合は $V'_T := V_T$ として, 生成規則の集合 P' は

$$P' = P \cup \{S' \longrightarrow \alpha \mid (S \to \alpha) \in P\}$$

で定義する．このとき, $\mathcal{L}(G) = \mathcal{L}(G')$ が成り立つ．これを示そう．

$\mathcal{L}(G) \subset \mathcal{L}(G')$ であることを示す．$w \in \mathcal{L}(G)$ をとると $S \overset{*}{\underset{G}{\Longrightarrow}} w$ とできる．このとき, 一番初めに適用する規則が $(S \to \alpha) \in P$ であるとすると $S \underset{G}{\Longrightarrow} \alpha \overset{*}{\underset{G}{\Longrightarrow}} w$ となる．すると, P' の作り方から, $(S' \to \alpha) \in P'$ だから $S' \underset{G'}{\Longrightarrow} \alpha \overset{*}{\underset{G}{\Longrightarrow}} w$ とかける．このとき, 後半では P に属する生成規則を用いており, これらは全て P' に属するから $S' \underset{G'}{\Longrightarrow} \alpha \overset{*}{\underset{G'}{\Longrightarrow}}$ となる．よって, $w \in \mathcal{L}(G')$ である．

逆の包含 $\mathcal{L}(G') \subset \mathcal{L}(G)$ を示す．$w \in \mathcal{L}(G')$ をとれば, $S' \overset{*}{\underset{G'}{\Longrightarrow}} w$ とできる．このとき, 一番初めに適用する規則が $(S' \to \alpha) \in P'$ であるとすると, $S \underset{G'}{\Longrightarrow} \alpha \overset{*}{\underset{G'}{\Longrightarrow}} w$ とかける．このとき, P' の作り方からある生成規則 $(S \to \alpha) \in P$ が存在して

$$S \underset{G}{\Longrightarrow} \alpha \overset{*}{\underset{G'}{\Longrightarrow}} w$$

とかける．ここで, $\alpha \overset{*}{\underset{G'}{\Longrightarrow}} w$ をみれば, α の中には S' を含まず, P' に属する生成規則の右辺には S' は現れないから $\alpha \overset{*}{\underset{G'}{\Longrightarrow}} w$ のどの文にも S' は現れない．従って, $\alpha \overset{*}{\underset{G'}{\Longrightarrow}} w$ のなかに現れる生成規則は全て P に含まれる．従って, これは $S \overset{*}{\underset{G}{\Longrightarrow}} w$ と書き換わるので, $w \in \mathcal{L}(G)$ となる．

この証明は, 文法の定義に依存しないから, 文脈依存文法を文脈自由文法, 正規文法にしても成り立つ．　□

• 8-4 : 空文の生成

文法 G が単調文法, 文脈自由文法, 正規文法のいずれであっても, その定義から空文 ε は生成できない．ここでは, 空文も生成できるように, これらの文法を拡張しよう．

定義 8.5. 文法 $G = (V_N, V_T, P, S)$ が以下の条件を満たすとする：

- 生成規則 $\alpha \to \beta$ について, β が空文でないなら $\ell(\alpha) \leq \ell(\beta)$ が成り立つ．
- P には生成規則 $S \to \varepsilon$ を含まれる．
- P の右辺に開始記号 S は現れない．

このとき, G を **文脈依存文法** と呼ぶ．

命題 8.6. G を文法とすると次が成り立つ.

(1) G を単調文法とすれば, $\mathcal{L}(G) \cup \{\varepsilon\}$ を生成するような文脈依存文法 G' が存在する.
(2) G を文脈依存文法とすれば, $\mathcal{L}(G) \setminus \{\varepsilon\}$ を生成するような単調文法 G' が存在する.

証明. (1) G を単調文法とすれば, **命題 8.4** より単調文法 $G'' = (V_N'', V_T'', P'', S'')$ であって, $\mathcal{L}(G) = \mathcal{L}(G'')$ かつ開始記号 S'' が P'' にあるどの生成規則の右辺にも現れないようにできる. これに対して, $G' = (V_N', V_T', P', S')$ を

$$V_N' = V_N'', \quad V_T' = V_T'', \quad P' = P'' \cup \{S' \to \varepsilon\}, \quad S' = S''$$

と定める.

G'' では, S'' が生成規則の右辺に現れないのだから, G' で空文を生成するには, 一番初めに $S' \to \varepsilon$ を適用する他ない. そうではないとき, 用いられる生成規則は全て P'' に属しており, $\mathcal{L}(G) = \mathcal{L}(G'')$ だから, $\mathcal{L}(G') = \mathcal{L}(G'') \cup \{\varepsilon\} = \mathcal{L}(G) \cup \{\varepsilon\}$ である.

(2) G を文脈依存文法とする. 定義より, ε を生成するには一番初めに $S \to \varepsilon$ を用いるしかない. ここで, $G' = (V_N', V_T', P', S')$ を

$$V_N' = V_N, \quad V_T' = V_T, \quad P' = P \setminus \{S \to \varepsilon\}, \quad S' = S$$

とおくと, G' は単調文法であり, $\mathcal{L}(G) \setminus \{\varepsilon\}$ である. □

注意. 文脈自由文法や正規文法でも, ε を生成できない. 上記と同様の手続きを踏むことで, 文脈自由文法や正規文法でも ε を生成できるように, その定義を拡張することができる. 今後, 空文の生成に関して**命題 8-6** で得られる文法も **文脈依存文法**, **文脈自由文法**, **正規文法** と呼ぶことにする.

担当：茨城大学 情報科学領域　宮本 賢伍