有限状態オートマトンによる文の解析と生成

新　田　義　彦

1　はじめに

有限状態オートマトン（Finite State Automaton）が，受理できる文は，正規文法（RG：Regular Grammar）に支配される文である．正規文法は，文脈自由文法（CFG：Context Free Grammar）や文脈依存文法（CSG：Context Sensitive Grammar）と比べると文を記述する能力が弱い．一般に RG は自然言語文の全体を規定（記述）するには能力不足のように言われている．本論文ではこのように役不足と見なされている RG を精いっぱい働かせて，自然言語文を解析あるいは生成する問題を扱う．RG による文の解析や生成を行うプログラムが，有限状態オートマトン（FSA：Finite State Automaton）である．本論文のタイトルでは，正規文法（RG）ではなく有限状態オートマトン（FSA）の方をとった．その理由は，文の解析や生成のメカニズムやアルゴリズムに軸足を置いて議論を展開するからである．

文脈自由文法（CFG）により規定される文の処理（特に受理）を行うプログラムは，プッシュ・ダウン・オートマトン（PDA：Push Down Automaton）と呼ばれる．雑に言えば，FSA に Push Down Stack の能力を追加したオートマトンである．文の受理処理において，文構成文字列を後戻り（つまり行きつ戻りつ）して処理できる分，能力が強化されている．

文脈依存文法（CSG）により規定される文の処理（特に受理）を行うプログラムは，線型有界オートマトン（LBA）Linear Bounded Automaton と呼ばれる．雑に言えば，PDA に「入力文字列の定数倍の長さの作業領域」の利用能力を追加したオートマトンである．文の受理処理において，文字や語句の出現環境を判定する能力が追加されるので，文処理能力は格段に増強される．

このように見てくると，有限状態オートマトン（Finite State Automaton）による文処理能力は，随分と弱体のように感じられるかもしれないが，文字列や語句の並びを，後戻りや環境の見渡しなどのいじましいことをせず，一直線に即決処理するという潔さが際立っている．この潔さは，処理の透明性，直観との相性，文法の簡潔性・透明性などの長所を産む．この特質を筆者は高く評価する．

2　有限状態オートマトンによる文処理の直観的描写

処理対象の文を構成する文字（アルファベット）の集合を Σ で表す．Σ 上の有限状態オートマトン M を５つ組の記号式 M =（K，Σ，δ，q_0，F）で表現する．

ここに，Σ は有限のアルファベット（文字）集合，δ は，Kx Σ から K の中への写像である．q_0 は初期状態であり，$q_0 \in$ K である．F は最終状態の集合であり，F ⊆ K である．K は有限状態オートマトンの状態集合である．K と δ が実質的にオートマトンの制御の機能を担う．たとえば　入力対象の文(つまり文字列）の最初の文字が a であったとしよう．そして，δ（q, a）= p であったとしよう．

この函数式は，次のように解釈できる．有限状態オートマトンMは，状態qにおいて文字aを入力されたとき，入力先を1文字分右にシフトして，Mの状態をpに変更する．

δは元来，Kx ΣからKの中への写像（つまり函数）であったが，Kx Σ^*からKの中への函数に自明な方法で拡張しておく．Σ^*は，Σの要素の有限部分列の集合である．

文x，つまりΣ上の有限文字列x（つまり$x \in \Sigma^*$）は，

$\delta(q_0, x) = p$かつ$p \in F$であるとき，「文xはオートマトンMにより受理された」という．つまりオートマトンMに入力された文xが，Mを最終状態$p \in F$に遷移させたとき，xはMによって受理（俗な言い方では認知）されたと言う．

Mが受理（認知）するすべての文xの集合を，T（M）で表す．すなわち，

$T(M) = \{x \mid \delta(q_0, x) \in F\}$

とおく．有限状態オートマトンMによって受理される文xの集合を，正規言語（regular language）と呼ぶ．

上記が有限状態オートマトンによる，文処理の直観的描写である．処理の仕方により，文解析になったり文生成になったりする．文生成では，状態シフトの際に文字あるいは文字列を出力するようにオートマトンを構成することがよく行われる．

3　正規表現の正確な定義とその内部計算処理

註：本章の骨格は文献「[14] M. Saraki, ed. and Y. Nitta, “Regular Expression and Text Mining（in Japanese）Second Printing”, Akashi-Shoten（2008）312p」の追補Ⅱ「正規表現と有限状態オートマトン（新田　著）」に準拠している．

3.1　パターンマッチング

正規表現の定義にはいる前に，テキスト（文字列）と正規表現の関係，文字列間のパターンマッチングの問題について，いま少し精密に話をまとめておく．

少し一般化することになるが，4文字の漢字熟語を検索しようとするのであれば，

〔亜－熙〕〔亜－熙〕〔亜－熙〕〔亜－熙〕

あるいは

〔亜－熙〕{4}

のように正規表現を使ってパターンマッチング指定ができる．漢字の字種には関係なく，1文字以上任意個数の漢字連続語句を検索パターンに指定したいのであれば，正規表現により

〔亜－熙〕＋

と表記すればよい．

このように正規表現は，検索をしたい（関心のある）語句を，直接的に指定して，あるいは一般化した広い形で指定して，テキスト（つまり長い文字列）の中から探索するために利用できる．文字列のようなストリング・データを対象として，相互の比較照合（摺り合わせ）により一致パターンを検索する方法を，「パターン・マッチング」と言う．正規表現は，“パターン指定を一般化・構造化して便利にする機能”であると解することもできる．

3.2　正規表現におけるメタ記号の意味

パターンマッチングとは，一般的に，正規表現のような構造を持つデータの間の一致判定処理のことを指す. 前節までの議論は文字列パターンマッチングの具体例を中心に論じたが, 以下の部分ではパターンを少し一般化･抽象化して捉え，パターンマッチングのメカニズムを精密に検討してみることにする. 正規表現の抽象化には，形式言語や形式文法（句構造文法）の手法を，パターンマッチングの抽象化にはオートマトンの手法を，それぞれ援用する.

“あるパターン（記号列）を受理するオートマトン”については，すでに述べた.

簡単なパターンの形式的表現を丁寧に追ってみよう.

（1）ac｜xy
（2）a(b｜c)
（3）a(x｜）b
（4）ax*(b｜c)

（1）は，ac または xy という記号パターンを表す.（記号は文字や単語と読み換えてもかまわない. 以下の説明でも同様である.）
（2）は，先頭に a という記号，次に b または c という記号が来るパターンを表す. つまり，ab または ac というパターンを表す.
（3）は，a と b の間に，x があってもなくてもよい，というパターンを表す. つまり，axb または ab というパターンを表す. したがって，“axb｜ab”と表現することもできる. 両者は正規表現として等価である. さらにまた正規表現処理プログラムによっては“a〔x〕?b”のように表すものもある.
（4）は, a と b の間, または a と c の間に, x を“任意個”挟みこんだパターンを表す.「任意個」とは，0 個以上有限個，という意味である. したがって，ab，ac，axb，axc，axxb，axxc，axxxb，axxxc，・・・・・などのパターンを一般化して表現した正規表現であると言える.

上記の正規表現で用いた特別な記号，つまり“a”や“b”，“x”のような対象文字記号以外の，正規表現の構造を定義するために用いた“＊”，“＋”，“｜”，“(”，“)”などの記号のことを,「メタ記号」あるいは「超記号」と呼ぶこともある. 少しだけ超記号の説明をする.

a，b，c,・・・・，x，y，z などのアルファベット：その文字（アルファベット）そのものを表す.
“ab”：文字 a と文字 b の連結（concatenation）を表す.
“a｜b”：a または b を選択（or, alternation）することを表す.
“a ＊”：a の 0 回以上の繰返し（閉包，closure）を表す.
丸カッコのペア“(”と“)”:一塊りの記号列を明記するための透明なバインダのような記号である.

冗長な丸カッコの使用は，透明な効果しか持たぬが，メタ記号の有効範囲を明記（限定）する目的で使われる丸カッコは, 本質的な役割を持つから注意する必要がある. 冗長な丸カッコの使用の例は,“(a)”である. これは“a”と等価である. しかし先ほどの例題（2）や（3）における丸カッコは本質的である.“ab｜c”の意味は元の（2）と違うし,“ax｜b”も（3）とは異なる意味を持つ. ついでに補

足すると，

　a(x｜y)＊

は，a の後に，何も続かないか，または，x または y が任意個続く，というような記号列を定義する正規表現である．つまり，

　a，ax，ay，axy，ayx，axx，ayy，axyxy，ayyyyyyyyyx，axxxxxxyxxy，・・・・

などをすべて代表して表現している正規表現である．

3.3　パターンマッチングとオートマトン

前節で取り上げた4つの正規表現（正規言語）を，パターンマッチングによりテキストから検索するメカニズムについて考えてみる．そのためには，これらの正規表現が表す記号群だけを正確に（弁別的に）認識する装置を構成する必要がある．この装置のことを，「受理オートマトン」と呼ぶ．テキストの先頭文字列から順々に1文字づつ受理オートマトンに入力し，一塊りの文字列が受理されると，この一塊りが所期のパターンとしてマッチングしたことになるから，これを取り出して適当に表示，あるいは蓄積すればよい．そして必要ならば，この一塊りの次の部分から再びオートマトンへの入力を再開して，次のパターンの検出に取り掛かる．オートマトンがパターンマッチングに失敗した場合，つまり正規表現で定義されたパターンが検出できなかった場合は，入力記号の先頭を1つ次にずらせて入力して，再び一致判定作業をやらせることとなる．受理されるか否か調べること，つまり，パターンマッチング検査が再開される．1文字だけずらせただけで，また同じテキスト全体を調べるのは，非効率的であるがこれが原則である．この非効率性を改善すべく，色々な工夫が考案されている．

もう1つだけ注意点を述べる．パターンの多義性がある場合には，どのパターンを優先的に検出するかという問題である．この問題の対処方法はオートマトンの構成方法に依存する．換言すればオートマトンの動作を解読すれば，どのように多義性に対応しているかが直ちに分かる．一般原則を言えば，“最大長優先”かつ“最初検出優先”で，パターンマッチングが行われる，となる．

具体的に言うと，テキストの文字列が，

　aaaaaxxxaaaaaaaaaaaayyyyaaaaaaaaaaaaaaaaaazzzzzz・・・・・

であるとき，

　a ＊

という正規表現でパターンマッチングをすると，まず最初に，出力（マッチングしたと判定されるパターン）は，

　aaaaa

である．正規表現“a ＊”には，Φ（空記号）や a や aa，aaa，aaaa などが含まれるが，“最長一致の原則”により検索されない．aaaaaaaaaaaa や aaaaaaaaaaaaaaaaaa は，“最初検出の原則”により検索されない．これらのパターンは，次の（2回目以降の）検索指定により検索されることとなる．これが一般的なパターンマッチング・アルゴリズムの動作である．

準備が長くなったが，先ほどの例題（1）～（4）の正規表現，

つまり，

　ac｜xy，a(b｜c)，a(x｜)b，ax*(b｜c)

が定義する記号群（アルファベット文字列）を受理するオートマトンを，具体的に構成してみよう．このオートマトンのことを，簡単に，“正規表現（1）～（4）を受理するオートマトン”と呼ぶ.

ここで取り扱うオートマトンは，正確には“有限状態オートマトン（FSA, Finite State Automaton)”と言う．内部状態のバリエーション数が有限個のマシンという意味である．このオートマトンは，

状態を示す丸印：○，

状態の遷移（変化）を示す矢印：→，

状態遷移を引き起こす入力記号（つまり読み込んだ1つの記号）：矢印の上か横に添えて表記される記号，

の3種類の図記号により表現される．特に，Sは初期状態，Fは終了状態を示す．Sから出ている矢印に添えられている記号が，受理すべきパターンの先頭記号，Fに入る矢印に添えられている記号が，受理すべきパターンの末端記号，ということになる.

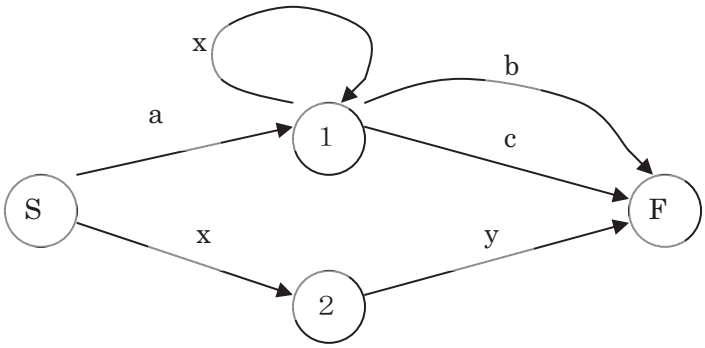


図1：所与の正規表現を受理するFSA（有限状態オートマトン）

上記の有限状態オートマトンが，（1）～（4）で与えた正規表現の表す記号列（所与のパターン）だけを正しく受理することを，オートマトンの状態遷移を追跡しながら観察する．まず，所与のパターンをもう一度検討して冗長性を除去してスッキリさせておこう.

正規表現 ac は，正規表現 a(b｜c)に含まれており，正規表現 a(x｜)b は正規表現 ax*(b｜c)に含まれている．したがって，上記のオートマトンが受理すべき正規表現群は，

xy，a(b｜c)，ax*(b｜c)

という3つの正規表現と等価である．“｜”，“(”，“)”などのメタ記号をなるべく除去したいというのであれば，上記の3ゆ正規表現群は，

xy，ab，ac，ax*b，ax*c

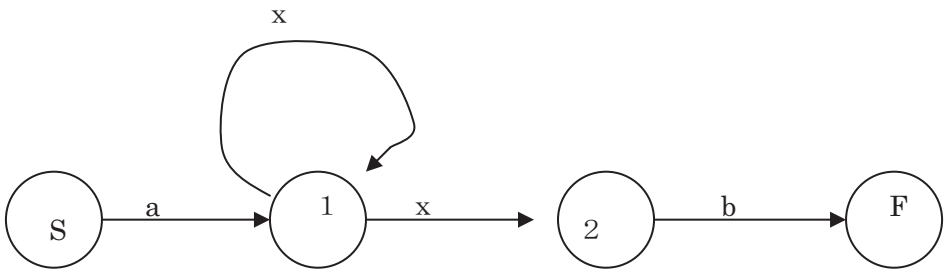
という5つの正規表現群として記述できる.

3.4　決定性オートマトンと非決定性オートマトン

以前の節で取り上げたオートマトンは，「決定性オートマトン」であった．“決定性”オートマトンとは，入力文字に対して，その状態遷移先がただ1つに（ユニークに）定まるようなオートマトンのことを言う．“状態遷移先が1つ定まる”ことには，遷移先が存在せずに行き詰まること，つまり，「オートマトンが現在の状態に遷移するまでに読み込んだ部分文字列は，”受理すべきパターンを構成しない“と

判断すること」も含まれる．

これに対して，入力文字（スキャンした文字）に対して，遷移可能な状態が2つ以上存在する場合が存在するようなオートマトンを，「非決定性オートマトン」と言う．具体例を次に図示する．

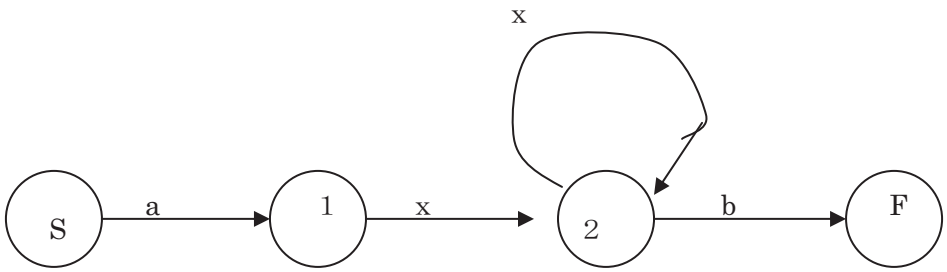


註：状態[1]において，入力文字“x”に対する遷移先が，[1]および[2]のように2つ存在するので，このオートマトンの動作は非決定性である．

図2：非決定性オートマトン

上図に示したオートマトンは，入力文字列“ax・・・”に対して，遷移可能な状態が[1]および[2]のように2箇所存在するので，その動作は決定性ではなく非決定性である．

非決定性オートマトンを決定性オートマトンに変換する方法の概略を述べておく．そのやり方の要諦は，オートマトンへの入力文字列群のバリエーションを，ベクトル値のように考えて，非決定性を発現している状態を細分類するのである．図2の非決定性オートマトンの場合であると，非決定性を発現している状態[1]が細分類の対象となる．x という文字を最初に1つ読み込んだ状態，およびその後の読み込み状態，の2つに細分する．結果的に，図2の非決定性オートマトンを，決定性オートマトンに変換したものは，次の図3のようになる．



註：この決定性オートマトンが定義する（弁別的に受理する）正規表現は，

ax + b

である．

図3：変換により得られた決定性有限状態オートマトン

3.5　正規表現の構文の形式的定義

これまで取り扱った正規表現の形，つまりその構文（統語構造）を数学的形式化により正確に眺めて

みよう．形式化のためには，BNF（Baccus Normal Form，バッカスの標準形，Baccus Nauer Form とも言う）を使う．この BNF は，文脈自由文法と等価である．先ほどのオートマトンの言葉で言えば，BNF は PDA（プッシュダウン・オートマトン，Push-Down Automaton）が受理する形式的言語である（後述）．

（1）Regexp　＝：Term　または　Term Regexp　または　Term｜Regexp
（2）Term　＝：Empty　または　Factor Term
（3）Factor　＝：Primeterm　または　Primeterm ＊
（4）Primeterm ＝：Alphabet　または　（RegExp）
（5）Alphabet　＝：a または b または c または・・・または z

上記の式は，左辺の記号が，右辺の記号（群）により定義される，と読む．左辺記号を右辺記号（群）に次々と置換して行き，終端記号だけからなる記号列に到達したならば，所期の正規表現が 1 つ得れる，と解釈してもよい．

上記の記号群で，

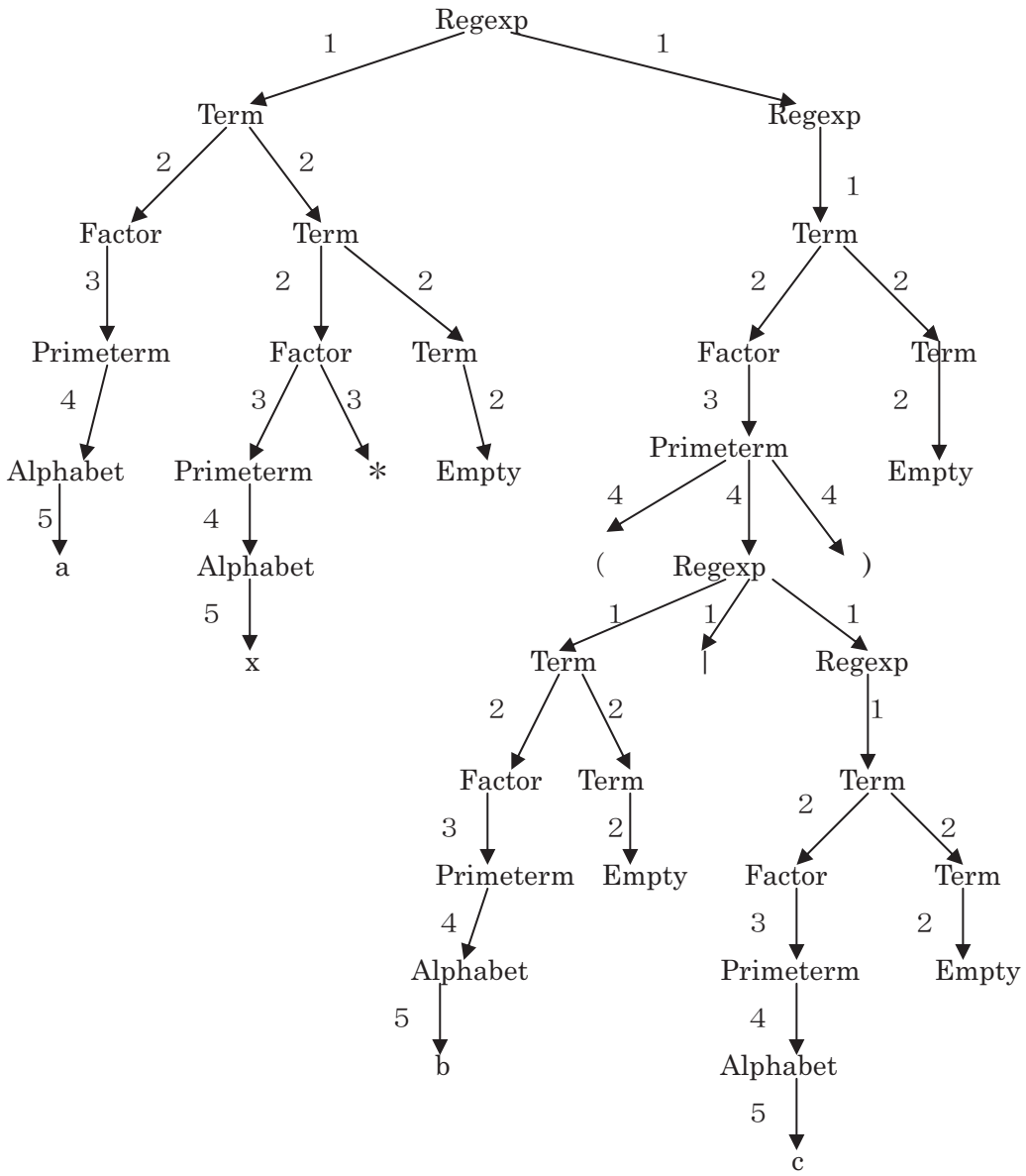
a，b，c，・・・・，z，“｜”，“(”，“)”および“＊”

が，終端記号であり，その他の記号が非終端記号である．非終端記号はさらに書き換えを進めて，終端記号（群）に辿り着く必要がある．終端記号（群）とは，もはやそれ以上書き換える必要のない，終端に行き着いた記号（群）のことである．終端記号群の並びが，何らかの正規表現を表す．ただし，上記の BNF 定義では，正規表現は，アルファベット a，b，c，・・・・，z，“｜”，“(”，“)”および“＊”だけから構成されている．前章でみたように仮名漢字を正規表現の構成要素として加えたいのであれば，Alphabet の定義式の右辺に追加する必要がある．言うまでもなく，BNF による正規表現の定義は非決定性である．RegExp を出発点とする左辺から右辺への書き換えを繰り返して，何らかの終端記号（群）に到着すれば，それが正当な正規表現の 1 つということが保証される．これが，BNF により正規表現を定義するメカニズムである．

1 つだけ具体例を示す．

“ax*(b｜c)”

が，上記の BNF 定義系列（1）～（5）から正しく導出される様子を，樹状図として以下に示した．上から下に向かう矢印が記号の導出を示す．終端記号に到達すると（その終端記号から先の）導出は行われないことは先に述べた．



註：上記の導出樹の枝（下向きの矢印）に付した番号は，上の記号から下の記号（群）を導出する際に適用した BNF 定義式の番号である.

註：逆立ち樹状図式（導出樹，ツリー）の底辺の記号列：

ax*Empty(b Empty | c Empty)Empty

が終端記号列である.

註：上記の終端記号列において Empty 記号を除去すると，下記の導出が確認できる.

ax*(b | c)

図４：BNF 型の正規表現の定義から“ax*(b | c)”を導出する過程

さらに，“a”，“ab”，“xyz”，“a | b ＊”，“r(s | t)pq ＊”，“x | y | z”，“ab(uv)＊ xyz”，などの正規表現が，BNF 定義式から正しく導出でき，構文的に正当な正規表現として保証できることを確認する

作業は，容易であるから記述は省略する．

4　有限状態オートマトンによる文の解析と生成

4.1　バイリンガル・コーパスの具体例

CERST プロジェクト〔S. Ikehara et al. 2002〕の 16 万文例の対訳コーパスの一部を以下に示す．これらの対訳文を解析・生成，つまり一方から他方への変換をする有限状態オートマトンの構成を，本章の実験目標とする．この変換は翻訳と見なすこともできる．有限状態オートマトンは，本質的に記号で表現される状態間の遷移写像（函数）であることをすでに示した．したがって次の節では「状態記号」と「状態間の遷移写像」を示すことにより，有限状態オートマトンを構築する．

AC000004-00　1　110
彼のお母さんがああ若いとは思わなかった　I never expected his mother to be so young.

AC000008-00　1　110
あの建物はああ見えても新しい　That building is still new despite appearance.

AC000016-00　5　110
ああいう人と付き合ってはだめだ　You shouldn't associate with that kind of person.

AC000019-00　3　100
歯医者は女の子に「ほらアーンして」と言った　The dentist said to the little girl, "Open your mouth wide. Say 'Ahh.' "

AC000020-00　4　110
愛と幸せに満ちた家庭にしたいと思います　We hope we will make our home full of love and happiness.

AC000024-00　3　100
資本主義と社会主義は相入れない概念だ　Capitalism is incompatible with socialism.

AC000029-00　5　110
重い病気だというのに彼はあいかわらず酒を飲んでいる　He continues drinking though he is being seriously ill.

AC000032-00　1　100
今年もあいかわりませずお付き合いのほどお願い申し上げます　I am looking forward to our continued friendship this coming year.

AC000043-00　1　110
彼女はにっこり笑ってぼくにあいさつした　She greeted me with a big smile.

AC000044-00　1　110
あの男はあいさつもせずに帰ってしまった　That man left without so much as saying goodbye.

AC000050-00　1　100
彼は愛車を駆って横浜へ行った　He drove his car to Yokohama.

AC000051-00　4　110
これは人々に愛唱されている古い民謡の一つです　This is one of the old folk songs people love to sing.

AC000054-00　5　110
桑田君を嫌うやつが多いけれど，ぼくとは不思議と相性がいい　Though many boys don't like Kuwata, there's a good chemistry between him and me.

AC000056-00　4　110
二人は相性が悪いということで離婚した　They divorced each other on the grounds of incompatibility.

****

AP001655-00　1　100
サーバから印刷作業を開始すると，緑ランプがすぐに点滅し始める.　As soon as a printing job is started from the server, the green light will start to flash.

AQ007544-00　3　111
医者は何も心配する理由はありませんと断言してくれた.　The doctor assured us that there were no grounds for alarm.

AQ056364-00　3　110
腹を立てることは，難しい交渉では逆効果を招きかねない.　Losing one's temper can be counterproductive in delicate negotiation.

AQ093621-00　2　110
彼の目はきょろきょろ部屋をながめ，何かないか物色していた.　His eyes restlessly roamed the room, searching for something.

AQ093631-00　3　110
彼らの視線が背中に突き刺してくるような気がした.　She felt their eyes stab into her back.

AQ093640-00　1　110
煙が目にしみて涙が出た.　The smoke made my eyes water.

AQ093691-00　1　100
彼らの服装が服装なのでイギリス人には給仕と見えた.　The costume gave them the appearance of waiters to English eyes.

AQ093693-00　3　110
それをヨーロッパ人が見たのはそのときが初めてだった.　This was the first time it had been seen by European eyes.
LJ030906: それをヨーロッパ人が見たのはそのときが初めてだった.　LE110688:This was the first time it had been seen by European eyes.

AQ145561-00　5　110
突然ガクンと揺れるのを感じ，やがてすべてのものがガタガタ揺れだした.　I felt a sudden lurch and everything started shaking.

AQ145564-00　1　110
突然ガクンと揺れて機は離陸した.　With a sudden lurch the plane left the ground.

AQ145569-00　2　110
地下鉄がカーブを曲がったとき彼女がよろめいて私にぶつかった.　She lurched into me as the subway went around a bend.

AQ145571-00　1　100
私たちのほうへよろめきながら近づいてきた.　He lurched toward us.

AQ145575-00　3　110
富には確かに人を引きつけるものがある.　There is a certain lure in riches.

4.2　文を解析・生成する有限状態オートマトン

文の解析・生成をする有限状態オートマトンを，状態記号列上の函数（変換）として構築する.

AC000004-00　1　110
LJ098578：彼のお母さんがああ若いとは思わなかった. ⇒　LE045139：I never expected his mother

to be so young.
WJ027379：＜ N1 は ＞ /N2 の /N3 が / あ あ /AJ4 と は /V5.hitei. kako.　⇒　WE027892：(N1 | I) never V5.past N2.poss N3 to be so AJ4.
彼の　⇒　N2 の
お母さんが　⇒　N3 が
若いとは　⇒　AJ4 とは
思わなかった.　⇒　/V5.hitei. kako.
I　⇒　N1 | I
never expected　⇒　never V5.past
his　⇒　N2.poss
mother　⇒　N3
to be so young.　⇒　to be so AJ4.

PJ013130：＜ N1 は ＞ /NP2 が / あ あ /AJ3 と は /V4.hitei. kako.　⇒　PE021860：(N1 | I) never V4. past NP2 to be so AJ3.

AC000008-00　　1　　110
LJ005957：あの建物はああ見えても新しい.　⇒　LE082665：That building is still new despite appearance.
WJ107742：REN1/N2 は / ああ / 見えても /AJ3.　⇒　WE040131：AJ1 N2 be still AJ3 despite appearance.
あの　⇒　REN1
建物は　⇒　N2 は
新しい.　⇒　AJ2.
That　⇒　AJ1
Building　⇒　N2
i　⇒　be
still new　⇒
PJ074990：NP1 は / ああ / 見えても /AJ2.　⇒　PE090162：NP1 be still AJ2 despite appearance.

以下の記述では語句レベルでの変換式は略す.

AC000016-00　　5　　110
LJ004511：ああいう人と付き合ってはだめだ.　⇒　LE123664：You shouldn't associate with that kind of person.
WJ037337：＜ N1 は＞ / ああ / いう /N2 と /V3 ては / だめだ.　⇒　WE033509：(N1 | you) should not V3 that kind of N2.
PJ000639：＜ N1 は＞ /VP2 ては / だめだ.　⇒　PE000161：(N1 | you) should not VP2.

AC000019-00　　3　　100
LJ071861：歯医者は女の子に「ほらアーンして」と言った.　⇒　LE089018：The dentist said to the

little girl, “Open your mouth wide. Say ‘Ahh.’ ”
WJ079726：N1 は /N2 に /「ほら / アーンして」と /V3.kako. ⇒ WE067344：N1 V3.past to N2 , “Open N2.pron. poss mouth wide. Say ‘Ahh.’ ”

AC000020-00　4　110
LJ046033：愛と幸せに満ちた家庭にしたいと思います. ⇒ LE115808：We hope we will make our home full of love and happiness.
WJ024250：＜ N1 は＞ /N2 と /N3 に /V4.kako/N5 に / したいと / ＜ N6 は＞ /V7.teinei. ⇒ WE037287：（N6 | I）V7（N1 | I）will make N1.pron. poss N5 V4 of N2 and N3.
PJ017032：＜ N1 は＞ /VP2.kako/N3 に / したいと / ＜ N4 は＞ /V5.teinei. ⇒ PE029217：（N4 | I）V5（N1 | I）will make N1.pron. poss N3 AJP（VP2）.

AC000024-00　3　100
LJ071815：資本主義と社会主義は相入れない概念だ. ⇒ LE011350：Capitalism is incompatible with socialism.
WJ059255：N1 と /N2 は /AJ3/ 概念だ. ⇒ WE000512：N1 be AJ3 with N2.

AC000029-00　5　110
LJ076000：重い病気だというのに彼はあいかわらず酒を飲んでいる. ⇒ LE018757：He continues drinking though he is being seriously ill.
WJ043741：＜ N1 は＞ / 重い / 病気だという / のに /N1 は / あいかわらず / 酒を / 飲んでいる. ⇒ WE022652：（N1 | I）continue drinking though（N1 | I）. pron be being seriously ill.
PJ007238：＜ N1 は＞ /AJ2/N3.da という / のに /N1 は / あいかわらず /VP4 ている. ⇒ PE018134：（N1 | I）continue VP4.ing though（N1 | I）. pron be being ADV（AJ2）AJ（N3）.

AC000032-00　1　100
LJ063930：今年もあいかわりませずお付き合いのほどお願い申し上げます. ⇒ LE036892：I am looking forward to our continued friendship this coming year.
WJ111210：TIME1 も / あいかわりませず / お付き合いの / ほど / ＜ N2 は＞ / お願い申し上げます. ⇒ WE034096：（N2 | I）be looking forward to our continued friendship N1.

AC000043-00　1　110
LJ113990：彼女はにっこり笑ってぼくにあいさつした. ⇒ LE075948：She greeted me with a big smile.
WJ090456：N1 は / にっこり / 笑って /N2 に /V3.kako. ⇒ WE066558：N1 V3.past N2.obj with a big smile.
PJ055509：N1 は / にっこり /V2 て /VP3.kako. ⇒ PE058691：N1 VP3.past with a big N（V2）.

AC000044-00　1　110

LJ007131：あの男はあいさつもせずに帰ってしまった．⇒　LE083150：That man left without so much as saying goodbye.
WJ112307：あの /N1 は / あいさつも / せずに / 帰ってしまった．⇒　WE110716：That N1 left without so much as saying goodbye.
PJ071860：NP1 は /N2 も / せずに /V3 てしまった．⇒　PE084968：NP1 N（V3）without so much as V（N2）.ing.

AC000050-00　　1　　100
LJ103266：彼は愛車を駆って横浜へ行った．⇒　LE019729：He drove his car to Yokohama.
WJ085482：N1 は /N2 を / 駆って /N3 へ / 行った．⇒　WE084445：N1 drove N1.poss N2 to N3.

AC000051-00　　4　　110
LJ018706：これは人々に愛唱されている古い民謡の一つです．⇒　LE109137：This is one of the old folk songs people love to sing.
WJ079343：N1 は /N2 に /V3.reru. teiru/AJ4/N5 の /N6 です．⇒　WE074897：N1 be N6 of AJ4 N5 N2 V3.
PJ044503：N1 は /N2 に /V3.reru. teiru/NP4 です．⇒　PE001443：N1 be NP4 N2 V3.

AC000054-00　　5　　110
LJ057471：桑田君を嫌うやつが多いけれど，ぼくとは不思議と相性がいい．⇒　LE111108：Though many boys don't like Kuwata, there's a good chemistry between him and me.
WJ102580：N1 を / 嫌う / やつが / 多い（けれど|けど），/N2 とは / 不思議と /N3 が /AJ4．⇒　WE116720：Though many boys do not like N1, there is AJ4 N3 between him and N2.obj.
PJ060277：N1 を / 嫌う / やつが / 多いけれど，/N2 とは / 不思議と /N3 が /AJ4．⇒　PE105871：Though many boys do not like N1, there is AJ4 N3 between him and N2.obj.

AC000056-00　　4　　110
LJ093583：二人は相性が悪いということで離婚した．⇒　LE105829：They divorced each other on the grounds of incompatibility.
WJ095199：N1 は / 相性が / 悪い（という｜と言う）/ ことで /V2.kako．⇒　WE064495：N1 V2.past each other on the grounds of incompatibility.
PJ056911：N1 は / 相性が / 悪いという / ことで /V2.kako．⇒　PE051901：N1 V2.past each other on the grounds of incompatibility.

****

AP001655-00　　1　　100
LJ040460：サーバから印刷作業を開始すると，緑ランプがすぐに点滅し始める．⇒　LE008923：As soon as a printing job is started from the server, the green light will start to flash.

WJ048632：N1 から /N2 を /V3 と，/N4 が / すぐに /V5.kaishi.　⇒　WE045045：As soon as N2 be V3.ed from N1, N4 will start to V5.

AQ007544-00　　3　　111

LJ046732：医者は何も心配する理由はありませんと断言してくれた．⇒　LE089315：The doctor assured us that there were no grounds for alarm.

WJ091578：N1 は / 何も / 心配する /N2 は / ありませんと /V3.tekureru. kako.　⇒　WE067510：N1 V3.past us that there were no N2 for alarm.

PJ055936：N1 は / 何も /V2/N3 は / ありませんと /V4.tekureru. kako.　⇒　PE055117：N1 V4.past us that there were no N3 for N（V2）.

CJ001121：N1 は /CL2.teinei. hitei と /V3.tekureru. kako.　⇒　CE000832：N1 V3.past us that CL2.

AQ056364-00　　3　　110

LJ118620：腹を立てることは，難しい交渉では逆効果を招きかねない．⇒　LE064820：Losing one's temper can be counterproductive in delicate negotiation.

WJ123386：腹を / 立てる / ことは，/#1［難しい］/N2 では / 逆効果を / 招きかねない．⇒　WE060707：Losing one's temper can be counterproductive in #1［delicate］N2.

PJ083205：VP1/ ことは，/NP2 では /VP3 かねない．⇒　PE108835：VP1 can VP3 in NP2.

AQ093621-00　　2　　110

LJ100624：彼の目はきょろきょろ部屋をながめ，何かないか物色していた．⇒　LE034245：His eyes restlessly roamed the room, searching for something.

WJ065510：N1 の /N2 は /ADV3/N4 を / ながめ，/ 何か / ないか /V5.teiru. kako.　⇒　WE093910：N1.poss N2 ADV3 roamed N4, V5.ing for something.

PJ069436：NP1 は /ADV2/N3 を / ながめ，/ 何か / ないか /V4.teiru. kako.　⇒　PE084924：NP1 ADV2 roamed N3, V4.ing for something.

AQ093631-00　　3　　110

LJ110825：彼らの視線が背中に突き刺してくるような気がした．⇒LE075739：She felt their eyes stab into her back.

WJ027345：＜ N1 は＞ /N2 の /N3 が /N4 に /V5.tekuru. suitei/ 気が / した．⇒　WE024269：（N1 | I）felt N2.poss N3 V5 into（N1 | I）. pron. poss N4.

PJ013107：＜ N1 は＞ /NP2 が /VP3.tekuru. suitei/VP4.kako.　⇒PE013609：（N1 | I）VP4.past NP2 VP3.

LJ048791：煙が目にしみて涙が出た．⇒　LE098668：The smoke made my eyes water.

WJ118318：煙が / 目に / しみて / ＜ N1 は＞ / 涙が / 出た．⇒　WE112972：The smoke made（N1 | I）. poss eyes water.

PJ034518：N1 が / 目に / しみて / ＜ N2 は＞ / 涙が / 出た．⇒　PE101873：The V（N1）made（N2 |

I). poss eyes water.

AQ093691-00　　1　　100
LJ110902：彼らの服装が服装なのでイギリス人には給仕と見えた．⇒　LE088453：The costume gave them the appearance of waiters to English eyes.
WJ123158：彼らの /N1 が / 服装なので /N2 には /N3 と / 見えた．⇒　WE085050：N1 gave N2.obj the appearance of N3 to N2 eye.

AQ093693-00　　3　　110
LJ030906：それをヨーロッパ人が見たのはそのときが初めてだった．⇒　LE110688：This was the first time it had been seen by European eyes.
WJ100248：N1 を /N2 が /V3.kako/ のは / そのときが / 初めてだった．⇒　WE116501：This was the first time N1 had been V3.ed by N2 eye.
PJ059680：N1 を /N2 が /V3.kako/ のは / そのときが /NP4.da た．⇒　PE105665：This was NP4 N1 had been V3.ed by N2 eye.

AQ145561-00　　5　　110
LJ093216：突然ガクンと揺れるのを感じ，やがてすべてのものがガタガタ揺れだした．⇒　LE041024：I felt a sudden lurch and everything started shaking.
WJ020434：＜ N1 は＞ /#2［突然］/#3［ガクンと］/ 揺れる / のを /V4, / やがて / すべての / ものが /#5［ガタガタ］/V6.kako.　⇒　WE014201：（N1 | I）V4.past（a | an）#2［sudden］lurch and everything V6.past.
PJ020366：＜ N1 は＞ /VP2/ のを /V3, / やがて /NP4 が /#5［ガタガタ］/V6.kako.　⇒　PE008250：（N1 | I）V3.past NP（VP2）and NP4 V6.past.

AQ145564-00　　1　　110
LJ093215：突然ガクンと揺れて機は離陸した．⇒　bLE121033：With a sudden lurch the plane left the ground.
WJ014925：#1［突然］/#2［ガクンと］/V3 て /N4 は /V5.kako.　⇒　WE123407：With #1［sudden］V3 N4 V5.past.
PJ003993：#1［ADV2］/#3［ガクンと］/V4 て /N5 は /V6.kako.　⇒　PE112939：With V6.

AQ145569-00　　2　　110
LJ088275：地下鉄がカーブを曲がったとき彼女がよろめいて私にぶつかった．⇒　LE077260：She lurched into me as the subway went around a bend.
WJ051547：N1 が /N2 を /V3.kako/ とき /N4 が / よろめいて /N5 に / ぶつかった．⇒　WE103210：N4 lurched into N5.obj as N1 V3.past N2.
PJ031814：N1 が /VP2.kako/ とき /N3 が / よろめいて /N4 に / ぶつかった．⇒　PE082092：N3 lurched into N4.obj as N1 VP2.past.

AQ145571-00　1　100
LJ068065：私たちのほうへよろめきながら近づいてきた．⇒　LE025546：He lurched toward us.
WJ028615：＜N1 は＞/N2 の / ほうへ /V3 ながら / 近づいてきた．⇒　WE013451：（N1｜I）V3.past toward N2.obj.

AQ145575-00　3　110
LJ117474：富には確かに人を引きつけるものがある．⇒　LE102705：There is a certain lure in riches.
WJ062432：N1 には /#2［確かに］/ 人を / 引きつける / ものが / ある．⇒　WE114107：There is（a｜an）#2［certain］lure in N1.
PJ037266:N1 には /#2[ADV3]/VP4/ ものが / ある．⇒　PE103579:There is（a｜an）#2[AJ(ADV3)] NP（VP4）in N1.

5　おわりに

有限状態オートマトン（FSA）の構造と動作の本質を，精密に示した．その基礎の上で文，つまり有限長の文字列を，弁別的に受理するオートマトンを，直観的なグラフと函数式で示した．有限状態オートマトンにより受理される文の集合を「正規言語」と呼ぶ．さらに実際の自然言語文を正規言語の文と見なして，解析・生成するための FSA プログラムおよび正規文法の具体例も示し，その有効性の証左とした．俳句などの断片文を生成する FSA プログラムは次回取り上げることにしたい．

有限状態オートマトン FSA による文の解析・生成を，文の変換（あるいは変形）として解釈しそのように応用プログラムを構成することも可能である．また文の解析の基本動作は，文字列の間のパターン・マッチングとして解釈できることも示した．

従来，能力不足のように言われてきた有限状態オートマトン（FSA）そして正規文法（RG）が，実用上十分な言語解析・生成能力を持つことを，大規模な FSA プログラムの開発により実証することが直近の課題である．

参考文献

[1] L. Bentivogli, and E. Pianta, “Exploiting Parallel Texts in the Creation of Multilingual Semantically Annotated Resources: the MultiSemiCor Corpus”, *Natural Language Engineering*, Vol.11, No.3（2005）pp.247-261

[2] K. Church, I. Dagan, W. Gale, P. Fung, B. Satish and J. HELFMAN, “Aligning Parallel Texts: Do Methods Developed for English French Generalize to Asian Languages?” *Proceedings of the Pacific Asia Conference on Formal and Computational Linguistics*（1993）

[3] S. Ikehara et al. “Semantically Equivalent Language Transformation Method Based on Analogical Thinking Principle”,（in Japanese）, *Journal of Artificial Intelligence Society of Japan*（2002）
Also in: 電子情報通信学会技術研究報告．TL，思考と言語 102（491），7-12，2002-11-29

[4] A. Kinyon, “A Language-Independent Shallow-Parser Compiler”, *Proc. 39th ACL Ann. Meeting*（*European Chapter*）（2001）pp.322-329

[5] ホップクロフト＆ウルマン著（野崎，他訳），言語理論とオートマトン，サイエンス社（刊）（1971）

[6] E. Macklovitch and H. Marie-Louise, “Line ‘em up: Advances in Alignment Tecnology and Their Impact on Translation Support Tools”, *AMTA*（1996）pp145-156

［7］ R. Mihalcea, and M. Simard, “Parallel Texts”. *Natural Language Engineering* Vo.11, No.3（2005）pp.239-246
［8］ J. Munday, *Introducing Translation Studies*, Taylor & Francis Group（2009）
［9］ Y. Nitta, “Idiosyncratic Gap: A Tough Problem to Machine Translation”, *Proc. Comp. Linguistics, COLING’86* ACL（Assoc. Comp. Ling.）（1986）
［10］ Y. Nitta, “Problems of Machine Translation: From a Viewpoint of Logical Semantics”, *Economic Review of Nihon University*. Vol.72, No.2, Nihon University, Tokyo:（2002）pp.23-42
［11］ Y. Nitta, “The Utility and Problem of Insufficient Machine Translation”, *Economic Review of Nihon University*. Vol.80, No.4（2001）pp.1-54
［12］ A. Pim, *Exploring Translation Theories*, Routledge, Taylor & Francis Group（2010）
［13］ M. Saraki and Y. Nitta. “The Semantic Classification of Verb Conjunction in the “Shite” Form”, *Proceedings of Spring IECEI Conference*, IECEI Japan（2005）
［14］ M. Saraki, ed. and Y. Nitta, “Regular Expression and Text Mining（in Japanese）Second Printing”, Akashi-Shoten（2008）312p
［15］ A. G..William and K. W. Church, “A Program for Aligning Sentences in Bilingual Corpora”, *Computational Linguistics*, Vol.19, No.3（1993）pp.75-102
［16］ Hangeveld（Editor）, Simon C. Dik: The Theory of Functional Grammar 1, FGS 20, Mouton de Gruyter（1997）
［17］ Hangeveld（Editor）, Simon C. Dik: The Theory of Functional Grammar 2, FGS 20, Mouton de Gruyter（1997）