木オートマトン・トランスデューサによる自然言語処理

†林 克彦

NTTコミュニケーション科学基礎研究所
†hayashi.katsuhiko@lab.ntt.co.jp

自然言語処理: 1980-1990年代 (1/5)

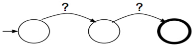
- 1980年代: 木に関連する形式文法の研究 (構文解析手法など)
  - CFG、LFG、HPSG、TAG、Prologなど
- 1990年代: 統計モデル + 文字列(変換)に関連する研究
  - 隠れマルコフモデル (HMM)、有限オートマトン (FSA)、有限トランスデューサ (FST)
  - n-gram言語モデル[eg, Jelinek 90]
  - 品詞タグ付け[eg, Church 88]
  - 統計的機械翻訳[eg, Brown 93]

- 重み付き (w)FSA, wFSTのツールキット (OpenFST) [eg, Mohri et. al. 00]
  - 統計的文字列処理の汎用ツールとして活躍
    1. HMM、CRFなどのモデルをエンコード可能[Kempe 97, Wu et. al. 14]
    2. 合成などの演算を使い複雑な問題が簡単に表せる

自然言語処理: wFSAとwFSTとは? (2/5)

- 塚田元氏のチュートリアル資料参照[塚田 02]

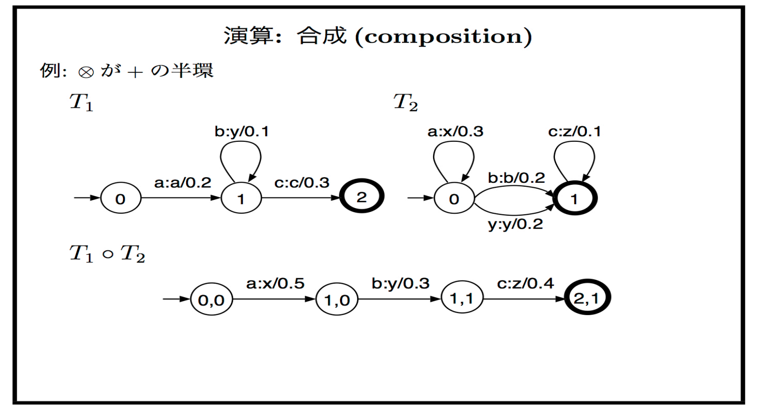
種々の有限状態機械



シンボル	有限状態オートマトン (Finite State Automaton)
シンボル＋遷移確率	確率的 FSA (Probabilistic FSA) 決定的 PFSA ＝ Markov Model 非決定的 PFSA ＝ Hidden Markov Model
入力・出力シンボル	有限状態トランスデューサ (Finite State Transducer)
入力・出力シンボル ＋重み	重み付き FST(Weighted FST)

自然言語処理: wFSTの合成によるモデルの連結 (3/5)

- 塚田元氏のチュートリアル資料参照[塚田 02]



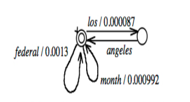
自然言語処理: wFSAとwFSTによる日英翻字[Knight & May 09] (4/5)

- 日英翻字の例
  - マスターズトーナメント ⇒ m a s u t a a z u t o o n a m e n t o ⇒ M AE S T ER Z T ER N UH M EH N T ⇒ Masters Tournament

WFSA A ⇒ English ⇒ WFST B ⇒ English sounds ⇒ WFST C ⇒ Japanese sounds ⇒ WFST D ⇒ Katakana

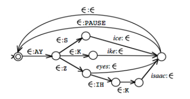
** 一般にNoisy Channelモデルを考えるため逆向き

Eフレーズ言語モデル



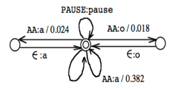
a WFSA A: Produce an English phrase

Eフレーズ ⇒ E読み



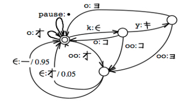
b WFST B: Convert English phrase to English sounds

E読み ⇒ J読み



c WFST C: Convert English sounds to Japanese sounds

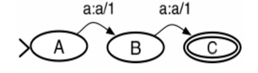
J読み ⇒ カタカナ



d WFST D: Convert Japanese sounds to Katakana

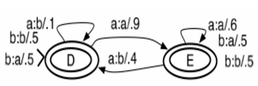
- 複数のwFSTを経由するとき

入力のFSTへの埋め込み I



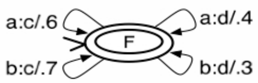
(a) Input string "a a" embedded in an identity WST

モデル1 T_1



(b) first WST in cascade

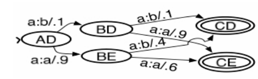
モデル2 T_2



(c) second WST in cascade

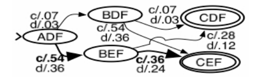
- アプローチ1: バケツリレー式合成 (Bucket brigade)

$I \circ T_1$



(e) Bucket brigade approach: Apply WST (b) to WST (a)

$Proj(I \circ T_1 \circ T_2)$ (出力空間)



(f) Result of offline or bucket application after projection

- アプローチ2: On-the-fly 合成[Pereira & Riley 97, Mohri 00]

(g) Initial on-the-fly stand-in for (f)

(h) On-the-fly stand-in after exploring outgoing edges of state ADF

(i) On-the-fly stand-in after best path has been found

自然言語処理: 2000-2010年代 (5/5)

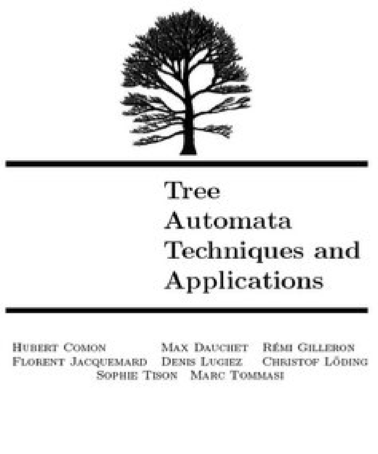
- 2000年代: 統計モデル + 木(変換)に関連する研究
  - 機械翻訳
    [Wu 97, Yamada & Knight 02, Melamed 03, Chiang 05, ...]
  - 文書要約
    [Knight & Marcu 00, ...]
  - 言い換え
    [Pang et al 03, ...]
  - 質問応答
    [Echihabi & Marcu 03, ...]
  - 自然言語文生成
    [Bangalore & Rambow 00, ...]

形式的な枠組み

- 同期文法 (同期文脈自由文法、同期木接合文法など)
- 木トランスデューサ

木オートマトン・トランスデューサ

- TATA[Comon et. al. 08]



- 木オートマトン
  [Thacher 67, Rounds 68]
- 木トランスデューサ
  [Rounds 68, Thatcher 70, Rounds 70]
- 応用:
  - 自然言語処理[Knight 06]
  - XML文書処理[高田 & 関 08]
  - データベース、項書き換え、定理証明 ...

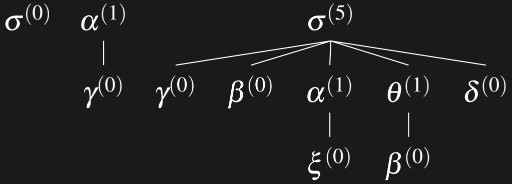
階層化アルファベット (Ranked Alphabet)

- 階層化アルファベット (Ranked Alphabet) (Σ, rk)
  - 記号の有限集合 Σ
  - 記号から正の整数への写像 $rk : \Sigma \to \mathbb{N} \cup \{0\}$
    - ランクnの記号$\sigma^{(n)}$ ($rk(\sigma) = n$, $\sigma \in \Sigma$)
    - ランクnの記号から成る集合 $\Sigma^{(n)} \subseteq \Sigma$

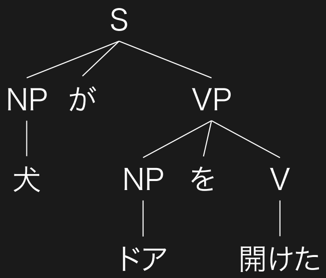
    注意: 場合によって(n)は省き、(Σ, rk)は単にΣと書く

木の定義

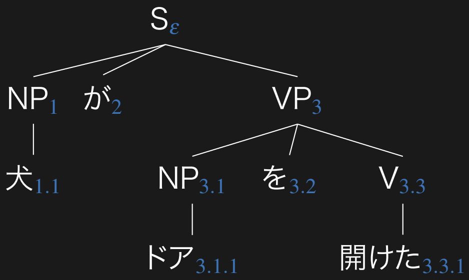
- Σの記号から構成される木の集合 $T_\Sigma(A)$
  - 記号集合A ($A^{(0)}$)の記号は木の葉のみに現れる
- 木の定義
  - 全ての記号$\sigma \in A \cup \Sigma^{(0)}$に対して、$\sigma \in T_\Sigma(A)$
  - 全ての記号$\sigma \in \Sigma^{(k)}$に対して、$\sigma(t_1, \ldots, t_k) \in T_\Sigma(A)$ ($t_1, \ldots, t_k \in T_\Sigma(A)$)
- 例: $\Sigma = \{\sigma^{(0)}, \lambda^{(0)}, \gamma^{(0)}, \xi^{(0)}, \delta^{(0)}, \alpha^{(1)}, \theta^{(1)}, \sigma^{(5)}\}$, $A = \{\beta\}$



木のラベル、部分木、置換

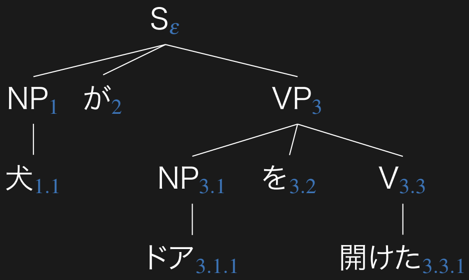


木のラベル、部分木、置換



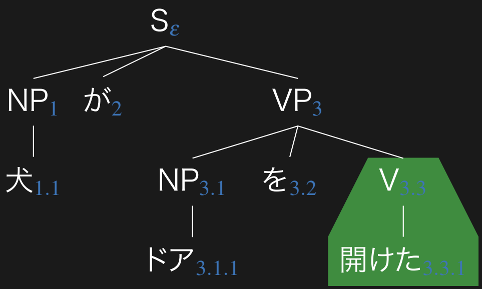
- 位置集合 $pos(t) = \{\varepsilon\} \cup \{i.v \mid 1 \leq i \leq k, v \in pos(t_i)\}$

木のラベル、部分木、置換



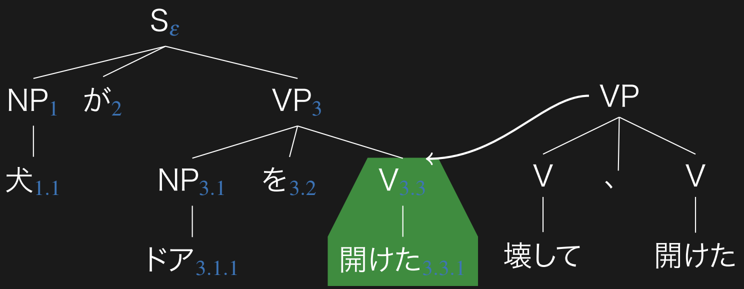
- 位置集合 $pos(t) = \{\varepsilon\} \cup \{i.v \mid 1 \leq i \leq k, v \in pos(t_i)\}$
- 位置 $v \in pos(t)$に対して
  - ラベル $t(v)$: $t(\varepsilon) =$ S、$t(3.3) =$ V

木のラベル、部分木、置換



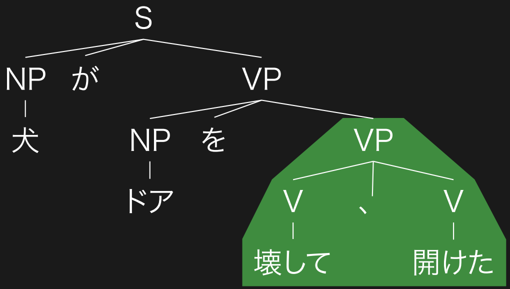
- 位置集合 $pos(t) = \{\varepsilon\} \cup \{i.v \mid 1 \leq i \leq k, v \in pos(t_i)\}$
- 位置 $v \in pos(t)$に対して
  - ラベル $t(v)$: $t(\varepsilon) =$ S、$t(3.3) =$ V
  - 部分木 $t|_v$: $t|_{3.3} =$ (V 開けた)

木のラベル、部分木、置換



- 位置集合 $pos(t) = \{\varepsilon\} \cup \{i.v \mid 1 \leq i \leq k, v \in pos(t_i)\}$
- 位置 $v \in pos(t)$に対して
  - ラベル $t(v)$: $t(\varepsilon) =$ S、$t(3.3) =$ V
  - 部分木 $t|_v$: $t|_{3.3} =$ (V 開けた)
  - 木 sによる置換 $t[s]_v$: t[(VP (V 壊して) 、 (V 開けた))]$_{3.3} =$

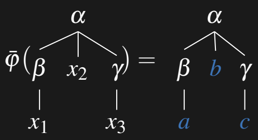
木のラベル、部分木、置換



- 位置集合 $pos(t) = \{\varepsilon\} \cup \{i.v \mid 1 \le i \le k, v \in pos(t_i)\}$
- 位置 $v \in pos(t)$に対して
  - ラベル $t(v)$: $t(\varepsilon) =$ S、$t(3.3) =$ V
  - 部分木 $t|_v$: $t|_{3.3} =$ (V 開けた)
  - 木 sによる置換 $t[s]_v$: $t[$(VP (V 壊して) 、 (V 開けた))$]_{3.3} =$
    (S (NP 犬) が (VP (NP ドア) を (VP (V 壊して) 、 (V 開けた)))

変数、木の代入の定義

- 変数の集合: $X = \{x_1, x_2, \ldots\}$
  - $X_k = \{x_1, \ldots, x_k\}$
- 木の代入 $\varphi : X \to T_\Sigma(X)$
  - 定義域: $\mathrm{DOM}(\varphi) = \{x \in X \mid \varphi(x) \neq x\}$
  - $\mathrm{DOM}(\varphi) = \{x_1, \ldots, x_k\}$、かつ、$\varphi(x_i) = t_i$のとき、$\varphi$を$\{x_1 \leftarrow t_1, \ldots, x_k \leftarrow t_k\}$と書く
  - 拡張代入 $\bar{\varphi} : T_\Sigma(X) \to T_\Sigma(X)$
    - $\bar{\varphi}(\sigma(s_1, \ldots, s_k)) = \sigma(\bar{\varphi}(s_1), \ldots, \bar{\varphi}(s_k))$
  - 代入の例: $\varphi = \{x_1 \leftarrow a, x_2 \leftarrow b, x_3 \leftarrow c\}$



木に対する文法、受理機械、変換機械

- 文法・受理機械:

	文字列	木
文法	正規文法 文脈自由文法	??
受理機械	有限オートマトン プッシュダウンオートマトン	??

- 変換機械:

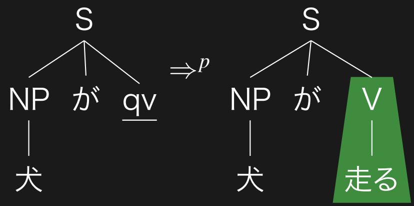
	文字列ペア	木ペア
変換機械	有限トランスデューサ	??

正規木文法 (Regular Tree Grammars)

- 正規木文法 (RTG) $G = (N, \Sigma, P, I)$
  - N: 非終端記号 (状態記号)の集合
  - $I \subseteq N$: 開始となる非終端記号の集合
  - Σ: 終端記号 (木のラベル記号)の集合
  - P: 生成規則の集合
    - 各生成規則 $n \to u \in P$の形をとる ($n \in N$, $u \in T_\Sigma(N)$)

正規木文法 (Regular Tree Grammars)

- 正規木文法 (RTG) $G = (N, \Sigma, P, I)$
  - N: 非終端記号 (状態記号)の集合
  - $I \subseteq N$: 開始となる非終端記号の集合
  - Σ: 終端記号 (木のラベル記号)の集合
  - P: 生成規則の集合
    - 各生成規則 $n \to u \in P$の形をとる ($n \in N$, $u \in T_\Sigma(N)$)
- 導出ステップ $s \Rightarrow_G^p t$ ($s, t \in T_\Sigma(N)$, $p = n \to u \in P$):
  - 規則 pは$n \to u$とする ($n \in N$, $u \in T_\Sigma(N)$)
  - 位置 $v \in pos(s)$に対して、ラベル $s(v) = n$かつ置換 $s[u]_v = t$
  - 例: 規則 $p =$ qv $\to$ (V 走る)



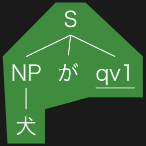
例: RTGによる木の生成過程

- 正規木文法 $G = (N, \Sigma, P, I)$
  - $N = \{$q, qv1, qv2$\}$
  - $\Sigma = \{$S, NP, VP, V,
    犬, が, ドア, を, 開ける$\}$
  - $I = \{$q$\}$
  - 規則集合 P:

例: RTGによる木の生成過程

- 正規木文法 $G = (N, \Sigma, P, I)$
  - $N = \{\underline{\text{q}}, \underline{\text{qv1}}, \underline{\text{qv2}}\}$
  - $\Sigma = \{$S, NP, VP, V, 犬, が, ドア, を, 開ける$\}$
  - $I = \{\underline{\text{q}}\}$
  - 規則集合 P:
    p_1: $\underline{\text{q}} \rightarrow$ (S (NP 犬) が $\underline{\text{qv1}}$)

$\underline{\text{q}} \quad \Rightarrow^{p_1}$

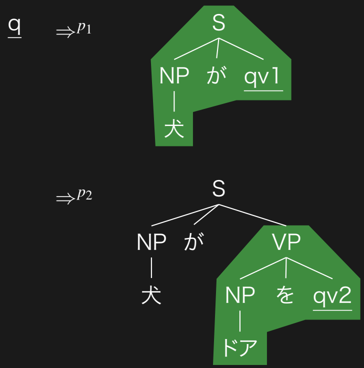


例: RTGによる木の生成過程

- 正規木文法 $G = (N, \Sigma, P, I)$
  - $N = \{$q, qv1, qv2$\}$
  - $\Sigma = \{$S, NP, VP, V, 犬, が, ドア, を, 開ける$\}$
  - $I = \{$q$\}$
  - 規則集合 P:

    p_1: q $\rightarrow$ (S (NP 犬) が qv1)

    p_2: qv1 $\rightarrow$ (VP (NP ドア) を qv2)



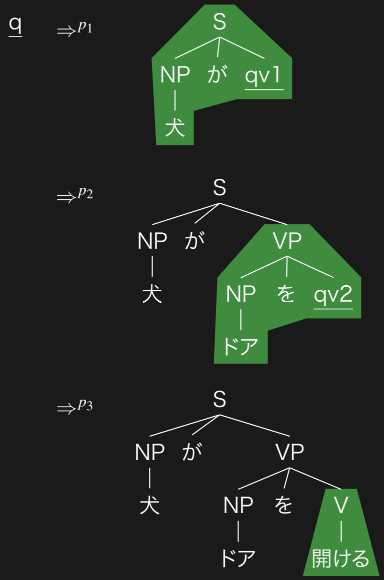
例: RTGによる木の生成過程

- 正規木文法 $G = (N, \Sigma, P, I)$
  - $N = \{\underline{\mathrm{q}}, \underline{\mathrm{qv1}}, \underline{\mathrm{qv2}}\}$
  - $\Sigma = \{$S, NP, VP, V, 犬, が, ドア, を, 開ける$\}$
  - $I = \{\underline{\mathrm{q}}\}$
  - 規則集合 P:

    p_1: $\underline{\mathrm{q}}$ $\rightarrow$ (S (NP 犬) が $\underline{\mathrm{qv1}}$)

    p_2: $\underline{\mathrm{qv1}}$ $\rightarrow$ (VP (NP ドア) を $\underline{\mathrm{qv2}}$)

    p_3: $\underline{\mathrm{qv2}}$ $\rightarrow$ (V 開ける)



例: RTGによる木の生成過程

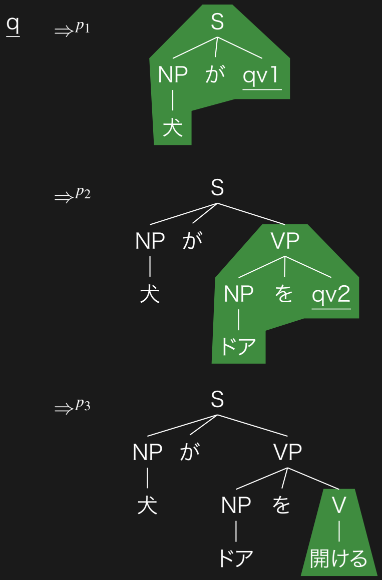
- $\underline{\text{正規木文法 } G = (N, \Sigma, P, I)}$
  - $N = \{\underline{\mathrm{q}}, \underline{\mathrm{qv1}}, \underline{\mathrm{qv2}}\}$
  - $\Sigma = \{$S, NP, VP, V, 犬, が, ドア, を, 開ける$\}$
  - $I = \{\underline{\mathrm{q}}\}$
  - 規則集合 P:

    p_1: $\underline{\mathrm{q}}$ $\rightarrow$ (S (NP 犬) が $\underline{\mathrm{qv1}}$)

    p_2: $\underline{\mathrm{qv1}}$ $\rightarrow$ (VP (NP ドア) を $\underline{\mathrm{qv2}}$)

    p_3: $\underline{\mathrm{qv2}}$ $\rightarrow$ (V 開ける)
- <u>生成可能な木の集合</u>

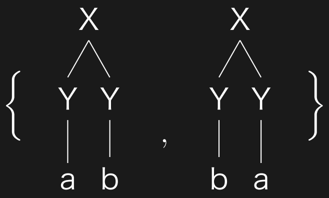
  $L(G) = \{t \in T_\Sigma \mid \underline{\mathrm{q}} \Rightarrow^*_G t, \underline{\mathrm{q}} \in I\}$



文脈自由文法 (CFG)との関係 (1/2)

- 木の生成力
  - あらゆるCFGに対して、その導出木を生成する RTGが存在? ⇒ Yes
  - あらゆるRTGに対して、その生成する木を導出木として持つ CFGが存在? ⇒ No

木の集合



RTG

$\underline{q} \rightarrow$ (X (Y a) (Y b))

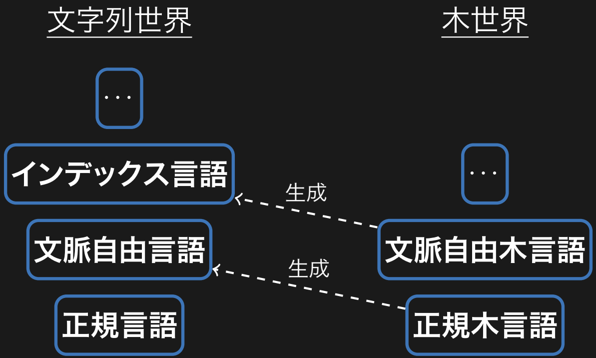
$\underline{q} \rightarrow$ (X (Y b) (Y a))

CFG

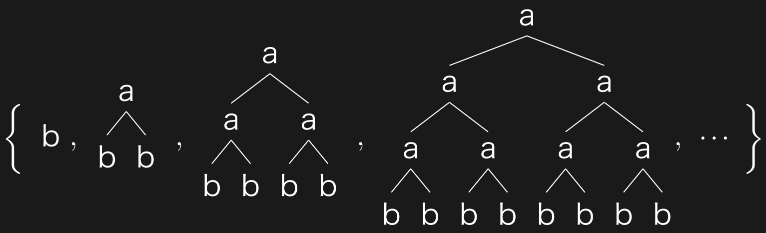
存在しない

文脈自由文法 (CFG)との関係 (2/2)

- 文字列の生成力
  - RTGが生成する木の葉の列を文字列としたとき、
    その集合は文脈自由言語に属する

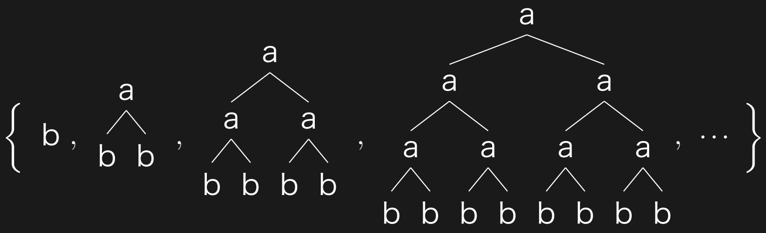


RTGでは表せない木集合



- 左右の部分木の高さが常に同じ木の集合
  - 文字列言語 $\{b^{2^n} \mid n \geq 0\} \neq$ CFL

RTGでは表せない木集合



- <u>左右の部分木の高さが常に同じ木の集合</u>
  - 文字列言語 $\{b^{2^n} \mid n \geq 0\} \neq$ CFL
- <u>文脈自由木文法 (Context-free Tree Grammar)</u>
  - 左辺に変数を持つことができる

$$\underline{q_0} \rightarrow b \qquad \underline{q_0} \rightarrow \underline{q_1}(\underline{q_0}) \qquad \underline{q_1}(x) \rightarrow$$

木に対する文法、受理機械、変換機械

- 文法・受理機械:

	文字列	木
文法	正規文法 文脈自由文法	正規木文法
受理機械	有限オートマトン プッシュダウンオートマトン	??

- 変換機械:

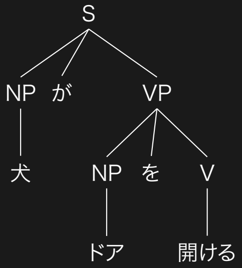
	文字列ペア	木ペア
変換機械	有限トランスデューサ	??

上昇型木オートマトン (Bottom-up FTA)

- 非決定性上昇型木オートマトン (FTA↑) $A = (Q, \Sigma, F, E)$
  [Thacher & Wright 68; Doner 70]
  - Q: 状態集合、$F \subseteq Q$: 終了状態の集合
  - Σ: 入力の階層化アルファベット
  - $E \subseteq \overbrace{Q \times \cdots \times Q}^{k} \times \Sigma^{(k)} \times Q$: 遷移の集合
    - 遷移: $\sigma(q_1, \ldots, q_k) \to q \in E$ ($q, q_1, \ldots, q_k \in Q$、$\sigma \in \Sigma^{(k)}$)
- 決定性:
  - ある記号$\sigma^{(k)} \in \Sigma$と状態系列$q_1, \ldots, q_k$に対して、
    遷移先が一意に決まる
- 実行 (Run) ρ: $pos(t) \to Q$
  - $t(v) = \sigma^{(k)}$となる位置 $v \in pos(t)$に対して、
    $\sigma(\rho(v.1), \ldots, \rho(v.k)) \to \rho(v) \in E$が存在する

例: FTA$\uparrow$による木の受理過程

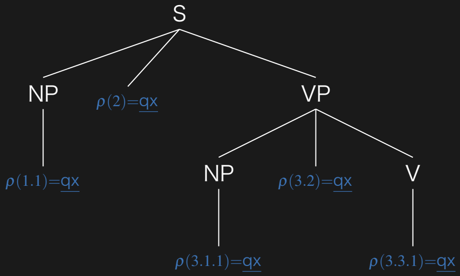
- FTA$\uparrow$ $A = (Q, \Sigma, F, E)$
  - $Q = \{\underline{\mathrm{q}}, \underline{\mathrm{qv}}, \underline{\mathrm{qnp}}, \underline{\mathrm{qvp}}, \underline{\mathrm{qx}}\}$
  - $F = \{\underline{\mathrm{q}}\}$
  - $\Sigma = \{$S, NP, VP, V, 犬, が, ドア, を, 開ける$\}$
  - 遷移集合 E:



例: FTA↑による木の受理過程

- FTA↑ $A = (Q, \Sigma, F, E)$
  - $Q = \{\underline{\text{q}}, \underline{\text{qv}}, \underline{\text{qnp}}, \underline{\text{qvp}}, \underline{\text{qx}}\}$
  - $F = \{\underline{\text{q}}\}$
  - $\Sigma = \{$S, NP, VP, V, 犬, が, ドア, を, 開ける$\}$
  - 遷移集合 E:

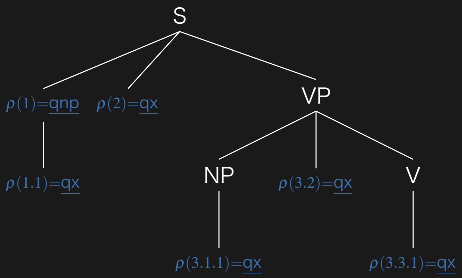
犬 $\to \underline{\text{qx}}$
が $\to \underline{\text{qx}}$
ドア $\to \underline{\text{qx}}$
を $\to \underline{\text{qx}}$
開ける $\to \underline{\text{qx}}$



例: FTA↑による木の受理過程

- FTA↑ $A = (Q, \Sigma, F, E)$
  - $Q = \{\underline{\text{q}}, \underline{\text{qv}}, \underline{\text{qnp}}, \underline{\text{qvp}}, \underline{\text{qx}}\}$
  - $F = \{\underline{\text{q}}\}$
  - $\Sigma = \{$S, NP, VP, V, 犬, が, ドア, を, 開ける$\}$
  - 遷移集合 E:

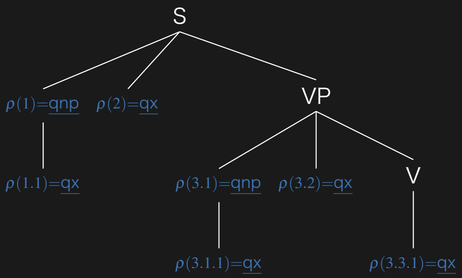
$$
\begin{aligned}
&\text{犬} \rightarrow \underline{\text{qx}}\\
&\text{が} \rightarrow \underline{\text{qx}}\\
&\text{ドア} \rightarrow \underline{\text{qx}}\\
&\text{を} \rightarrow \underline{\text{qx}}\\
&\text{開ける} \rightarrow \underline{\text{qx}}\\
&\text{NP}(\underline{\text{qx}}) \rightarrow \underline{\text{qnp}}
\end{aligned}
$$



例: FTA↑による木の受理過程

- FTA↑ $A = (Q, \Sigma, F, E)$
  - $Q = \{$q, qv, qnp, qvp, qx$\}$
  - $F = \{$q$\}$
  - $\Sigma = \{$S, NP, VP, V, 犬, が, ドア, を, 開ける$\}$
  - 遷移集合 E:

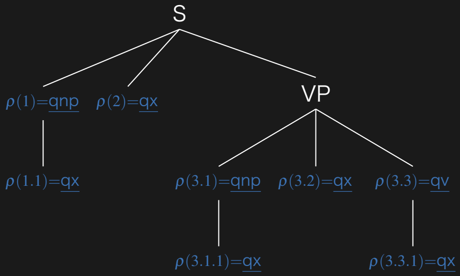
犬 $\rightarrow$ qx
が $\rightarrow$ qx
ドア $\rightarrow$ qx
を $\rightarrow$ qx
開ける $\rightarrow$ qx
NP(qx) $\rightarrow$ qnp



例: FTA↑による木の受理過程

- FTA↑ $A = (Q, \Sigma, F, E)$
  - $Q = \{\underline{\mathrm{q}}, \underline{\mathrm{qv}}, \underline{\mathrm{qnp}}, \underline{\mathrm{qvp}}, \underline{\mathrm{qx}}\}$
  - $F = \{\underline{\mathrm{q}}\}$
  - $\Sigma = \{$S, NP, VP, V, 犬, が, ドア, を, 開ける$\}$
  - 遷移集合 E:

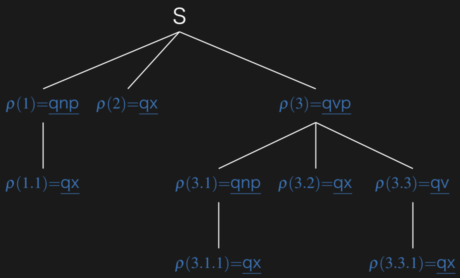
    犬 $\rightarrow \underline{\mathrm{qx}}$
    が $\rightarrow \underline{\mathrm{qx}}$
    ドア $\rightarrow \underline{\mathrm{qx}}$
    を $\rightarrow \underline{\mathrm{qx}}$
    開ける $\rightarrow \underline{\mathrm{qx}}$
    NP($\underline{\mathrm{qx}}$) $\rightarrow \underline{\mathrm{qnp}}$
    V($\underline{\mathrm{qx}}$) $\rightarrow \underline{\mathrm{qv}}$



例: FTA↑による木の受理過程

- FTA↑ $A = (Q, \Sigma, F, E)$
  - $Q = \{$q, qv, qnp, qvp, qx$\}$
  - $F = \{$q$\}$
  - $\Sigma = \{$S, NP, VP, V, 犬, が, ドア, を, 開ける$\}$
  - 遷移集合 E:

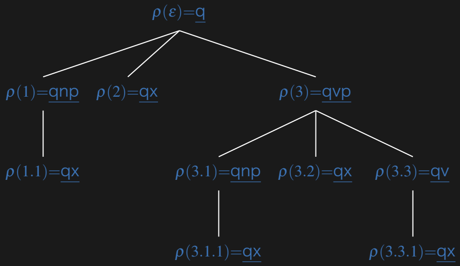
    犬 → qx
    が → qx
    ドア → qx
    を → qx
    開ける → qx
    NP(qx) → qnp
    V(qx) → qv
    VP(qnp,qx,qv) → qvp



例: FTA↑による木の受理過程

- FTA↑ $A = (Q, \Sigma, F, E)$
  - $Q = \{$q, qv, qnp, qvp, qx$\}$
  - $F = \{$q$\}$
  - $\Sigma = \{$S, NP, VP, V, 犬, が, ドア, を, 開ける$\}$
  - 遷移集合 E:

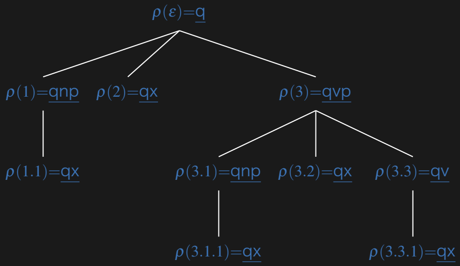
犬 $\rightarrow$ qx
が $\rightarrow$ qx
ドア $\rightarrow$ qx
を $\rightarrow$ qx
開ける $\rightarrow$ qx
NP(qx) $\rightarrow$ qnp
V(qx) $\rightarrow$ qv
VP(qnp,qx,qv) $\rightarrow$ qvp
S(qnp,qx,qvp) $\rightarrow$ q



例: FTA↑による木の受理過程

- FTA↑ $A = (Q, \Sigma, F, E)$
  - $Q = \{$q, qv, qnp, qvp, qx$\}$
  - $F = \{$q$\}$
  - $\Sigma = \{$S, NP, VP, V, 犬, が, ドア, を, 開ける$\}$
  - 遷移集合 E:

    犬 $\rightarrow$ qx
    が $\rightarrow$ qx
    ドア $\rightarrow$ qx
    を $\rightarrow$ qx
    開ける $\rightarrow$ qx
    NP(qx) $\rightarrow$ qnp
    V(qx) $\rightarrow$ qv
    VP(qnp,qx,qv) $\rightarrow$ qvp
    S(qnp,qx,qvp) $\rightarrow$ q

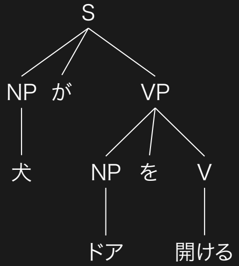


- 受理可能な木の集合
  $L(A) = \{t \in T_\Sigma \mid \rho(\varepsilon) \in F$となる実行$\rho$が存在$\}$

下降型木オートマトン (Top-down FTA)

- 非決定性下降型木オートマトン (FTA↓) $A = (Q, \Sigma, I, E)$
  [Rabin 69; Doner 70]
  - $I \subseteq Q$: 初期状態の集合
  - $E \subseteq Q \times \Sigma^{(k)} \times \overbrace{Q \times \cdots \times Q}^{k}$: 遷移の集合
    - 遷移: $\sigma(q) \to (q_1, \ldots, q_k) \in E$ ($q, q_1, \ldots, q_k \in Q$、$\sigma \in \Sigma^{(k)}$)

- 例: FTA↓による受理過程
  - $I = \{$q$\}$
  - 遷移集合 E:



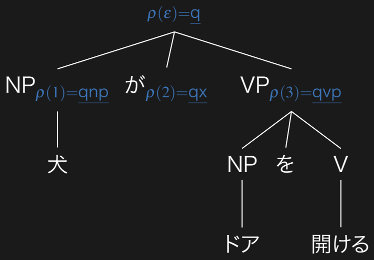
下降型木オートマトン (Top-down FTA)

- 非決定性下降型木オートマトン (FTA↓) $A = (Q, \Sigma, I, E)$

  [Rabin 69; Doner 70]
  - $I \subseteq Q$: 初期状態の集合
  - $E \subseteq Q \times \Sigma^{(k)} \times \overbrace{Q \times \cdots \times Q}^{k}$: 遷移の集合
    - 遷移: $\sigma(q) \to (q_1, \ldots, q_k) \in E$ ($q, q_1, \ldots, q_k \in Q$、$\sigma \in \Sigma^{(k)}$)

- 例: FTA↓による受理過程
  - $I = \{\text{q}\}$
  - 遷移集合 E:

    S(q) $\to$ (qnp,qx,qvp)



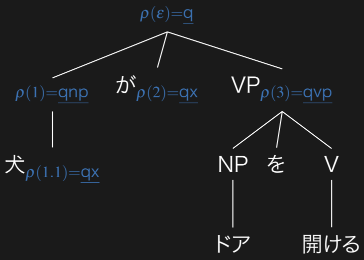
下降型木オートマトン (Top-down FTA)

- 非決定性下降型木オートマトン (FTA↓) $A = (Q, \Sigma, I, E)$

  [Rabin 69; Doner 70]
  - $I \subseteq Q$: 初期状態の集合
  - $E \subseteq Q \times \Sigma^{(k)} \times \overbrace{Q \times \cdots \times Q}^{k}$: 遷移の集合
    - 遷移: $\sigma(q) \to (q_1, \ldots, q_k) \in E$ ($q, q_1, \ldots, q_k \in Q$、$\sigma \in \Sigma^{(k)}$)
- 例: FTA↓による受理過程
  - $I = \{$q$\}$
  - 遷移集合 E:

    S(q) $\to$ (qnp,qx,qvp)

    NP(qnp) $\to$ (qx)

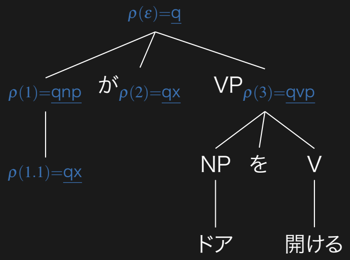


下降型木オートマトン (Top-down FTA)

- 非決定性下降型木オートマトン (FTA↓) $A = (Q, \Sigma, I, E)$
  [Rabin 69; Doner 70]
  - $I \subseteq Q$: 初期状態の集合
  - $E \subseteq Q \times \Sigma^{(k)} \times \overbrace{Q \times \cdots \times Q}^{k}$: 遷移の集合
    - 遷移: $\sigma(q) \rightarrow (q_1, \ldots, q_k) \in E$ ($q, q_1, \ldots, q_k \in Q$、 $\sigma \in \Sigma^{(k)}$)

- 例: FTA↓による受理過程
  - $I = \{$q$\}$
  - 遷移集合 E:

    S(q) $\rightarrow$ (qnp,qx,qvp)
    NP(qnp) $\rightarrow$ (qx)
    犬(qx) $\rightarrow$ accept



下降型木オートマトン (Top-down FTA)

- 非決定性下降型木オートマトン (FTA↓) $A = (Q, \Sigma, I, E)$

  [Rabin 69; Doner 70]
  - $I \subseteq Q$: 初期状態の集合
  - $E \subseteq Q \times \Sigma^{(k)} \times \overbrace{Q \times \cdots \times Q}^{k}$: 遷移の集合
    - 遷移: $\sigma(q) \rightarrow (q_1, \ldots, q_k) \in E$ ($q, q_1, \ldots, q_k \in Q$、 $\sigma \in \Sigma^{(k)}$)

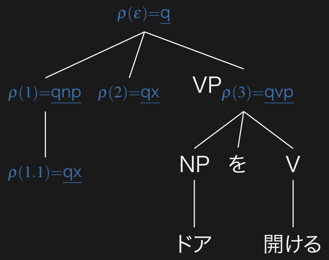
- 例: FTA↓による受理過程
  - $I = \{$q$\}$
  - 遷移集合 E:

    S(q) → (qnp,qx,qvp)

    NP(qnp) → (qx)

    犬(qx) → accept

    が(qx) → accept



下降型木オートマトン (Top-down FTA)

- 非決定性下降型木オートマトン (FTA↓) $A = (Q, \Sigma, I, E)$

  [Rabin 69; Doner 70]
  - $I \subseteq Q$: 初期状態の集合
  - $E \subseteq Q \times \Sigma^{(k)} \times \overbrace{Q \times \cdots \times Q}^{k}$: 遷移の集合
    - 遷移: $\sigma(q) \to (q_1, \ldots, q_k) \in E$ ($q, q_1, \ldots, q_k \in Q$、 $\sigma \in \Sigma^{(k)}$)

- 例: FTA↓による受理過程
  - $I = \{$q$\}$
  - 遷移集合 E:

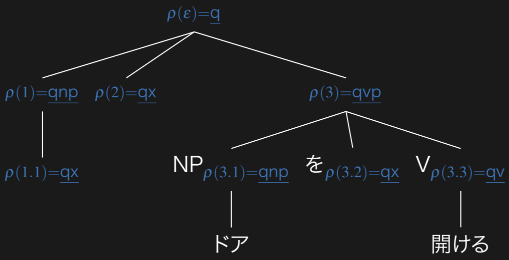
    S(q) $\to$ (qnp,qx,qvp)

    NP(qnp) $\to$ (qx)

    犬(qx) $\to$ accept

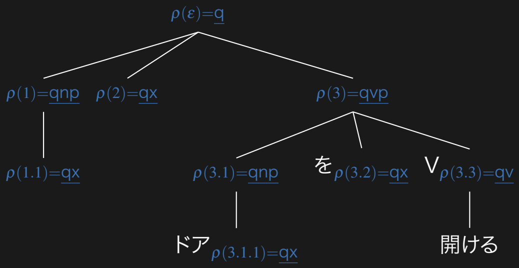
    が(qx) $\to$ accept

    VP(qvp) $\to$ (qnp,qx,qvp)



下降型木オートマトン (Top-down FTA)

- 非決定性下降型木オートマトン (FTA↓) $A = (Q, \Sigma, I, E)$
  [Rabin 69; Doner 70]
  - $I \subseteq Q$: 初期状態の集合
  - $E \subseteq Q \times \Sigma^{(k)} \times \overbrace{Q \times \cdots \times Q}^{k}$: 遷移の集合
    - 遷移: $\sigma(q) \to (q_1, \ldots, q_k) \in E$ $(q, q_1, \ldots, q_k \in Q$、 $\sigma \in \Sigma^{(k)})$
- 例: FTA↓による受理過程
  - $I = \{$q$\}$
  - 遷移集合 E:
    S(q) $\to$ (qnp,qx,qvp)
    NP(qnp) $\to$ (qx)
    犬(qx) $\to$ accept
    が(qx) $\to$ accept
    VP(qvp) $\to$ (qnp,qx,qvp)



下降型木オートマトン (Top-down FTA)

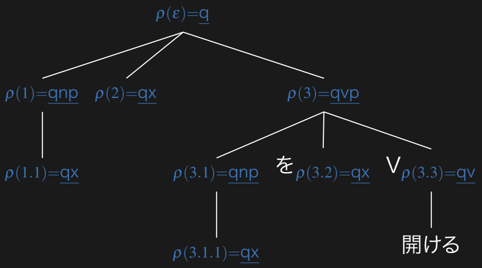
- 非決定性下降型木オートマトン (FTA↓) $A = (Q, \Sigma, I, E)$

[Rabin 69; Doner 70]

  - $I \subseteq Q$: 初期状態の集合
  - $E \subseteq Q \times \Sigma^{(k)} \times \overbrace{Q \times \cdots \times Q}^{k}$: 遷移の集合
    - 遷移: $\sigma(q) \to (q_1, \ldots, q_k) \in E$ ($q, q_1, \ldots, q_k \in Q$、$\sigma \in \Sigma^{(k)}$)

- 例: FTA↓による受理過程
  - $I = \{$q$\}$
  - 遷移集合 E:

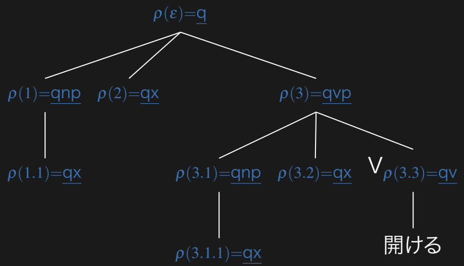
S(q) $\to$ (qnp,qx,qvp)
NP(qnp) $\to$ (qx)
犬(qx) $\to$ accept
が(qx) $\to$ accept
VP(qvp) $\to$ (qnp,qx,qvp)
ドア(qx) $\to$ accept



下降型木オートマトン (Top-down FTA)

- 非決定性下降型木オートマトン (FTA↓) $A = (Q, \Sigma, I, E)$
  [Rabin 69; Doner 70]
  - $I \subseteq Q$: 初期状態の集合
  - $E \subseteq Q \times \Sigma^{(k)} \times \overbrace{Q \times \cdots \times Q}^{k}$: 遷移の集合
    - 遷移: $\sigma(q) \to (q_1, \ldots, q_k) \in E$ ($q, q_1, \ldots, q_k \in Q$、$\sigma \in \Sigma^{(k)}$)
- 例: FTA↓による受理過程
  - $I = \{$q$\}$
  - 遷移集合 E:

    S(q) $\longrightarrow$ (qnp,qx,qvp)
    NP(qnp) $\longrightarrow$ (qx)
    犬(qx) $\longrightarrow$ accept
    が(qx) $\longrightarrow$ accept
    VP(qvp) $\longrightarrow$ (qnp,qx,qvp)
    ドア(qx) $\longrightarrow$ accept
    を(qx) $\longrightarrow$ accept

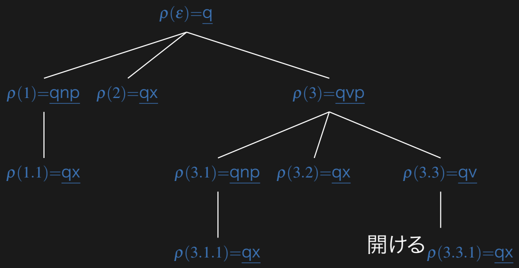


下降型木オートマトン (Top-down FTA)

- 非決定性下降型木オートマトン (FTA↓) $A = (Q, \Sigma, I, E)$
  [Rabin 69; Doner 70]
  - $I \subseteq Q$: 初期状態の集合
  - $E \subseteq Q \times \Sigma^{(k)} \times \overbrace{Q \times \cdots \times Q}^{k}$: 遷移の集合
    - 遷移: $\sigma(q) \to (q_1, \ldots, q_k) \in E$ $(q, q_1, \ldots, q_k \in Q$、$\sigma \in \Sigma^{(k)})$

- 例: FTA↓による受理過程
  - $I = \{$q$\}$
  - 遷移集合 E:

    S(q) → (qnp,qx,qvp)
    NP(qnp) → (qx)
    犬(qx) → accept
    が(qx) → accept
    VP(qvp) → (qnp,qx,qvp)
    ドア(qx) → accept
    を(qx) → accept
    V(qv) → (qx)



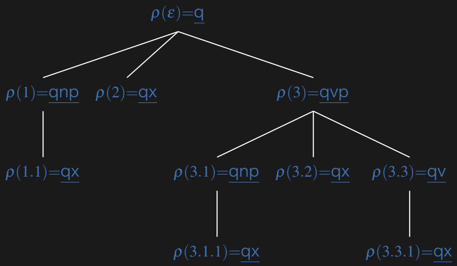
下降型木オートマトン (Top-down FTA)

- 非決定性下降型木オートマトン (FTA↓) $A = (Q, \Sigma, I, E)$

  [Rabin 69; Doner 70]
  - $I \subseteq Q$: 初期状態の集合
  - $E \subseteq Q \times \Sigma^{(k)} \times \overbrace{Q \times \cdots \times Q}^{k}$: 遷移の集合
    - 遷移: $\sigma(q) \to (q_1, \ldots, q_k) \in E$ ($q, q_1, \ldots, q_k \in Q$、$\sigma \in \Sigma^{(k)}$)

- 例: FTA↓による受理過程
  - $I = \{$q$\}$
  - 遷移集合 E:

    S(q) $\longrightarrow$ (qnp,qx,qvp)
    NP(qnp) $\longrightarrow$ (qx)
    犬(qx) $\longrightarrow$ accept
    が(qx) $\longrightarrow$ accept
    VP(qvp) $\longrightarrow$ (qnp,qx,qvp)
    ドア(qx) $\longrightarrow$ accept
    を(qx) $\longrightarrow$ accept
    V(qv) $\longrightarrow$ (qx)
    開けた(qx) $\longrightarrow$ accept



FTA↑とFTA↓の比較

- FTA↑ = FTA↓ (まとめてFTAと呼ぶ)
  - 終了状態の集合 F ⇔ 初期状態の集合 I
  - $\sigma(q_1, \ldots, q_k) \rightarrow q \Leftrightarrow \sigma(q) \rightarrow (q_1, \ldots, q_k)$
- 決定性FTA↑ (Det-FTA↑) ≠ 決定性FTA↓ (Det-FTA↓)

木の集合

Det-FTA↑

X(qa qb) → q

X(qb qa) → q

a → qa

b → qb

Det-FTA↓

存在しない

- Det-FTA↓ ⊊ Det-FTA↑ = FTA↓ = FTA↑

RTGとの関係

- 任意のRTGは標準形を持つ[Alexandrakis & Bozapalidis 87]
  - 標準形のRTG規則 $\underline{q} \to t$
    - $t \in \Sigma^{(0)}$: ランク0の記号
    - $t \in N$: 非終端記号 (状態)
    - $t = \sigma^{(k)}(\underline{q_1}, \ldots, \underline{q_k})$ ($\sigma \in \Sigma$, $\underline{q_1}, \ldots, \underline{q_k} \in N$)
  - FTA↑の規則
    - $\sigma^{(0)} \to \underline{q}$
    - $\varepsilon(\underline{q_1}) \to \underline{q}$
    - $\sigma^{(k)}(\underline{q_1}, \ldots, \underline{q_k}) \to \underline{q}$

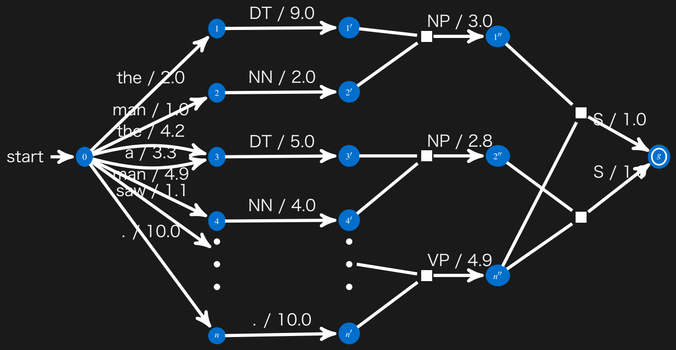
$\Rightarrow$ RTGとFTAは等価

正規木言語の性質 (閉包性、判定問題)

- 文字列[Hopcroft & Ullman 79]、木[Gécseg & Steinby 84]

	文字列		木
	正規言語 (FSA, rexp)	文脈自由言語 (CFG, PDA)	正規木言語 (FTA, RTG)
補集合	Yes	Yes	$\overline{L(A)} = \{t \mid t \notin L(A)\} = L(\overline{A})$
和集合	Yes	No	$L(A_1) \cup L(A_2) = L(A_1 \cup A_2)$
積集合	Yes	No	$L(A_1) \cap L(A_2) = L(A_1 \cap A_2)$
所属判定	Yes	Yes	PTIME
空判定	Yes	Yes	PTIME
等価性判定	Yes	No	Det-FTAの場合 PTIME

重み付きFTA↑



- 重み付き (w)FTA↑ $A = \{Q, \Sigma, F, E, \pi\}$
    - 重み付き遷移 $r = \sigma(q_1, \ldots, q_k) \xrightarrow{w} q \in E$
    - 遷移の重み $\pi(r) = w$
- 構文解析や構文に基づく機械翻訳などの解空間

FTA↑上での演算

- 自然言語処理で役立ちそうな演算等

	重みなし	重み付き	現状 (NLP)
決定化	[Comon et. al. 08]	[May & Knight 06]	現実的に動かない
無曖昧化	[林 & 永田 14]	[林 & 永田 14]	現実的に動かない
最小化	[Comon et. al. 08]	[Maletti 08]	出番がない
積集合	自明?	[May 10]	リスコアリング 制約付き探索
k-best探索	N/A	[Huang & Chiang 05]	識別学習

$\Rightarrow$ 積集合演算とk-best探索について説明

積集合演算 (Intersection)

- Intersection $A_1 = \{Q_1, \Sigma, F_1, E_1, \pi_1\} \cap A_2 = \{Q_2, \Sigma, F_2, E_2, \pi_2\}$

$E = \emptyset$
for all $(q, q') \in Q_1 \times Q_2$ **do**
  for all $\sigma^{(k)} \in \Sigma$ **do**
    for all $\sigma^{(k)}(q_1, \ldots, q_k) \xrightarrow{w_1} q \in E_1$ **do**
      for all $\sigma^{(k)}(q'_1, \ldots, q'_k) \xrightarrow{w_2} q' \in E_2$ **do**
        $r \leftarrow \sigma^{(k)}((q_1, q'_1), \ldots, (q_k, q'_k)) \xrightarrow{w_1 + w_2} (q, q')$
        $E \leftarrow E \cup \{r\}$
        $\pi(r) \leftarrow w_1 + w_2$
return $A = \{Q_1 \times Q_2, \Sigma, F_1 \times F_2, E, \pi\}$

- 計算量 $O(|E_1||E_2|)$
- wFTA↑ A_2: 精巧な木言語モデルや制約を表した木正規表現

wFTA↑上でのk-best導出探索

		ビタビ	最良優先	A*
wFSA	1-best	[Viterbi 67]	[Dijkstra 59]	[Hart et. al. 68]
	k-best	[Eppstein 94]	??	??
wFTA↑	1-best	自明?	[Knuth 77]	[Klein & Manning 03]
	k-best	[Huang & Chiang 05]	[Huang 06]	[Paul & Klein 09]

- 文献[Huang & Chiang 05]ではAlgorithm0、1、2、3が提案されている
  - Algorithm2が実装と効率の観点から最も良く使われている

Huang & Chiang 05のAlgorithm2

- 各状態で累積重み上位k個の仮説を効率的に求める
- 例: 状態q$_0$における累積重み上位3個の仮説を求める ($3 \times 3 + 3 \times 3 = 18$通り)

		q$_2$		
X(q$_1$ q$_2$) $\xrightarrow{0.5}$ q$_0$		0.6	0.4	0.3
	0.9			
q$_1$	0.5			
	0.3			

		q$_4$		
Y(q$_3$ q$_4$) $\xrightarrow{0.3}$ q$_0$		0.8	0.2	0.1
	0.8			
q$_3$	0.4			
	0.2			

priority queue:

3-best:

Huang & Chiang 05のAlgorithm2

- 各状態で累積重み上位k個の仮説を効率的に求める
- 例: 状態q_0における累積重み上位3個の仮説を求める ($3 \times 3 + 3 \times 3 = 18$通り)

$X(q_1\ q_2) \xrightarrow{0.5} q_0$		q_2 0.6	q_2 0.4	q_2 0.3
q_1	0.9	0.27		
	0.5			
	0.3			

$Y(q_3\ q_4) \xrightarrow{0.3} q_0$		q_4 0.8	q_4 0.2	q_4 0.1
q_3	0.8	0.192		
	0.4			
	0.2			

priority queue: 0.27 0.192

3-best:

Huang & Chiang 05のAlgorithm2

- 各状態で累積重み上位k個の仮説を効率的に求める
- 例: 状態$\underline{q_0}$における累積重み上位3個の仮説を求める ($3 \times 3 + 3 \times 3 = 18$通り)

$X(\underline{q_1}\ \underline{q_2}) \xrightarrow{0.5} \underline{q_0}$		0.6	$\underline{q_2}$ 0.4	0.3
	0.9	1-best	0.18	
$\underline{q_1}$	0.5	0.15		
	0.3			

$Y(\underline{q_3}\ \underline{q_4}) \xrightarrow{0.3} \underline{q_0}$		0.8	$\underline{q_4}$ 0.2	0.1
	0.8	0.192		
$\underline{q_3}$	0.4			
	0.2			

priority queue: $\underline{0.192}$ $\underline{0.18}$ $\underline{0.15}$

3-best: $\underline{0.27}$

Huang & Chiang 05のAlgorithm2

- 各状態で累積重み上位k個の仮説を効率的に求める
- 例: 状態q_0における累積重み上位3個の仮説を求める ($3 \times 3 + 3 \times 3 = 18$通り)

$X(q_1\ q_2) \xrightarrow{0.5} q_0$		q_2 0.6	0.4	0.3
	0.9	1-best	0.18	
q_1	0.5	0.15		
	0.3			

$Y(q_3\ q_4) \xrightarrow{0.3} q_0$		q_4 0.8	0.2	0.1
	0.8	2-best	0.048	
q_3	0.4	0.096		
	0.2			

priority queue: 0.18 0.15 0.096 0.048

3-best: 0.27 0.192

Huang & Chiang 05のAlgorithm2

- 各状態で累積重み上位k個の仮説を効率的に求める
- 例: 状態q_0における累積重み上位3個の仮説を求める $(3 \times 3 + 3 \times 3 = 18$通り$)$

			q_2	
$X(q_1\ q_2) \xrightarrow{0.5} q_0$		0.6	0.4	0.3
	0.9	1-best	3-best	
q_1	0.5	0.15		
	0.3			

			q_4	
$Y(q_3\ q_4) \xrightarrow{0.3} q_0$		0.8	0.2	0.1
	0.8	2-best	0.048	
q_3	0.4	0.096		
	0.2			

priority queue: 0.15 0.096 0.048

3-best: 0.27 0.192 0.18

Huang & Chiang 05のAlgorithm2

- 各状態で累積重み上位k個の仮説を効率的に求める
- 例: 状態q_0における累積重み上位3個の仮説を求める ($3 \times 3 + 3 \times 3 = 18$通り)

X(q_1 q_2) $\xrightarrow{0.5}$ q_0		q_2: 0.6	q_2: 0.4	q_2: 0.3
q_1	0.9	1-best	3-best	
	0.5	0.15		
	0.3			

Y(q_3 q_4) $\xrightarrow{0.3}$ q_0		q_4: 0.8	q_4: 0.2	q_4: 0.1
q_3	0.8	2-best	0.048	
	0.4	0.096		
	0.2			

priority queue: 0.15 0.096 0.048

3-best: 0.27 0.192 0.18

- 計算量 $O(|E| + |Q| k \log k)$

木に対する文法、受理機械、変換機械

- 文法・受理機械:

	文字列	木
文法	正規文法 文脈自由文法	正規木文法
受理機械	有限オートマトン プッシュダウンオートマトン	木オートマトン

- 変換機械:

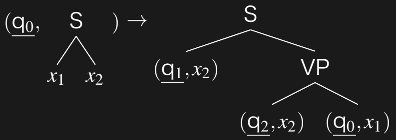
	文字列ペア	木ペア
変換機械	有限トランスデューサ	??

下降型木トランスデューサ (Top-down FTT)

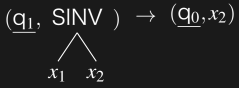
- 下降型木トランスデューサ (FTT↓) $M = (Q, \Sigma, \Delta, I, R)$
  - Q: 状態集合、$I \subseteq Q$: 初期状態の集合
  - Σ: 入力の階層化アルファベット
  - Δ: 出力の階層化アルファベット
  - $R \subseteq (Q \times \Sigma(X)) \times T_\Delta((Q \times X))$: 書き換え規則の集合
    - 規則 $r \in R$: $(q, \sigma^{(k)}(x_1, \ldots, x_k)) \to u$
      ($q \in Q$、$\sigma^{(k)} \in \Sigma$、$x_1, \ldots, x_k \in X$、 $u \in T_\Delta((Q \times X_k))$)
  - 規則の例: $(\underline{\mathrm{q}}, \mathrm{S}(x_1, x_2, x_3)) \to \mathrm{S'}((\underline{\mathrm{qnp}}, x_1)(\underline{\mathrm{qvp}}, x_3))$

FTT↓に対する制約

- <u>FTT↓に対する制約: $(q, \sigma^{(k)}(x_1, \ldots, x_k)) \to u$</u>
  - <u>決定性</u>: 状態$q \in Q$と$\sigma^{(k)} \in \Sigma$に対して、規則が1つ以上存在しない
  - <u>完全性</u>: 状態$q \in Q$と$\sigma^{(k)} \in \Sigma$に対して、規則が1つは存在する
  - <u>線形</u>: 変数$x_1, \ldots, x_k$はたかだか1度しか右辺に現れない
    - <u>複写 (⇔ 線形)の例</u>:

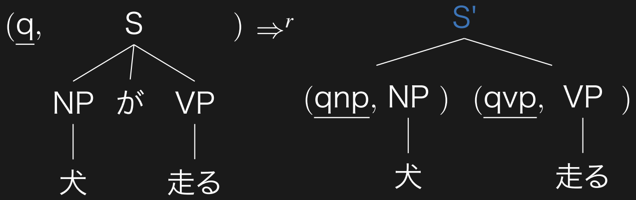


  - <u>非削除</u>: 変数$x_1, \ldots, x_k$は少なくとも1度は右辺に現れる
    - <u>削除 (⇔ 非削除)の例</u>:



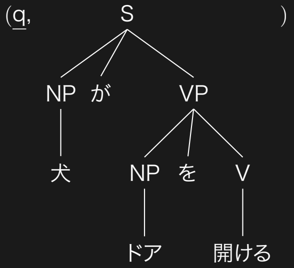
FTT↓の導出ステップ

- 導出ステップ: $s \Rightarrow^r_M t \ (s, t \in T_\Delta((Q \times T_\Sigma)))$:
  - 規則 $r = (q, \sigma(x_1, \ldots, x_k)) \to u$
  - ある位置 $v \in pos(s)$に対して、ラベル $s(v) = (q, \sigma(s_1, \ldots, s_k))$ かつ $s[\bar{\varphi}(u)]_v = t \ (s_1, \ldots, s_k \in T_\Delta((Q \times X)))$
    - 代入: $\varphi = \{(q_1, x_1) \leftarrow (q_1, s_1), \ldots, (q_k, x_k) \leftarrow (q_k, s_k)\}$
  - 例: 規則 $r = (\underline{\mathrm{q}}, \mathrm{S}(x_1, x_2, x_3)) \to \mathrm{S'}((\underline{\mathrm{qnp}}, x_1)(\underline{\mathrm{qvp}}, x_3))$
    代入 $\varphi = \{(\underline{\mathrm{qnp}}, x_1) \leftarrow (\underline{\mathrm{qnp}}, (\mathrm{NP}\ 犬)), (\underline{\mathrm{qvp}}, x_3) \leftarrow (\underline{\mathrm{qvp}}, (\mathrm{VP}\ 走る))\}$



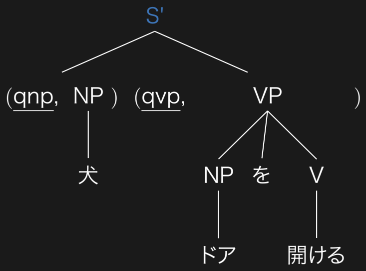
例: FTT↓による木の変換過程

- $\underline{\text{FTT}\downarrow\ M = (Q, \Sigma, \Delta, I, R)}$
  - $Q = \{\underline{\text{q}}, \underline{\text{qnp}}, \underline{\text{qv}}, \underline{\text{qvp}}\}$
  - $I = \{\underline{\text{q}}\}$
  - $\Sigma = \{$S, NP, VP, V, 犬, が, ドア, を, 開ける$\}$
  - $\Delta = \{$S', NP', VP', V', the, dog, opens, door$\}$
  - 規則集合 R:



例: FTT↓による木の変換過程

- FTT↓ $M = (Q, \Sigma, \Delta, I, R)$
  - $Q = \{\underline{\text{q}}, \underline{\text{qnp}}, \underline{\text{qv}}, \underline{\text{qvp}}\}$
  - $I = \{\underline{\text{q}}\}$
  - $\Sigma = \{$S, NP, VP, V, 犬, が, ドア, を, 開ける$\}$
  - $\Delta = \{$S', NP', VP', V', the, dog, opens, door$\}$
  - 規則集合 R:
    $(\underline{\text{q}}, \text{S}(x_1, x_2, x_3)) \to \text{S'}((\underline{\text{qnp}}, x_1)\ (\underline{\text{qvp}}, x_3))$

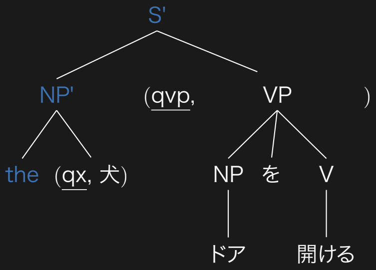


例: FTT↓による木の変換過程

- FTT↓ $M = (Q, \Sigma, \Delta, I, R)$
  - $Q = \{\underline{\text{q}}, \underline{\text{qnp}}, \underline{\text{qv}}, \underline{\text{qvp}}\}$
  - $I = \{\underline{\text{q}}\}$
  - $\Sigma = \{$S, NP, VP, V, 犬, が, ドア, を, 開ける$\}$
  - $\Delta = \{$S', NP', VP', V', the, dog, opens, door$\}$
  - 規則集合 R:

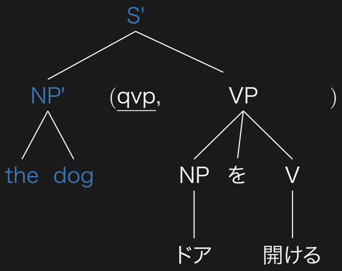
    $(\underline{\text{q}},\ \text{S}(x_1, x_2, x_3)) \rightarrow \text{S'}((\underline{\text{qnp}},\ x_1)\ (\underline{\text{qvp}},\ x_3))$

    $(\underline{\text{qnp}},\ \text{NP}(x_1)) \rightarrow \text{NP'}(\text{the}\ (\underline{\text{qx}},\ x_1))$



例: FTT↓による木の変換過程

- FTT↓ $M = (Q, \Sigma, \Delta, I, R)$
  - $Q = \{\underline{\mathrm{q}}, \underline{\mathrm{qnp}}, \underline{\mathrm{qv}}, \underline{\mathrm{qvp}}\}$
  - $I = \{\underline{\mathrm{q}}\}$
  - $\Sigma = \{$S, NP, VP, V, 犬, が, ドア, を, 開ける$\}$
  - $\Delta = \{$S', NP', VP', V', the, dog, opens, door$\}$
  - 規則集合 R:
    ($\underline{\mathrm{q}}$, S(x_1, x_2, x_3)) $\rightarrow$ S'(($\underline{\mathrm{qnp}}$, x_1) ($\underline{\mathrm{qvp}}$, x_3))
    ($\underline{\mathrm{qnp}}$, NP(x_1)) $\rightarrow$ NP'(the ($\underline{\mathrm{qx}}$, x_1))
    ($\underline{\mathrm{qx}}$, 犬) $\rightarrow$ dog



例: FTT↓による木の変換過程

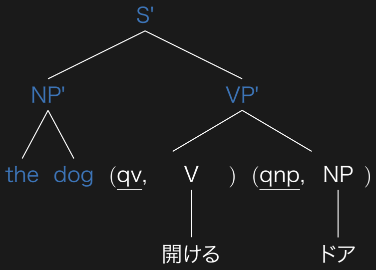
- $\underline{\text{FTT}\downarrow\ M = (Q, \Sigma, \Delta, I, R)}$
  - $Q = \{\underline{\text{q}}, \underline{\text{qnp}}, \underline{\text{qv}}, \underline{\text{qvp}}\}$
  - $I = \{\underline{\text{q}}\}$
  - $\Sigma = \{$S, NP, VP, V, 犬, が, ドア, を, 開ける$\}$
  - $\Delta = \{$S', NP', VP', V', the, dog, opens, door$\}$
  - 規則集合 R:

    $(\underline{\text{q}},\ \text{S}(x_1, x_2, x_3)) \rightarrow \text{S'}((\underline{\text{qnp}},\ x_1)\ (\underline{\text{qvp}},\ x_3))$

    $(\underline{\text{qnp}},\ \text{NP}(x_1)) \rightarrow \text{NP'}(\text{the}\ (\underline{\text{qx}},\ x_1))$

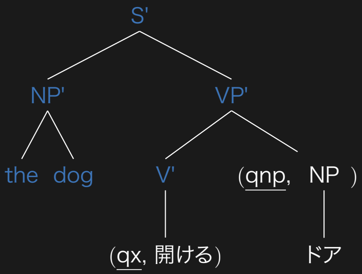
    $(\underline{\text{qx}},\ 犬) \rightarrow \text{dog}$

    $(\underline{\text{qvp}},\ \text{VP}(x_1, x_2, x_3)) \rightarrow \text{VP'}((\underline{\text{qv}},\ x_3)\ (\underline{\text{qnp}},\ x_1))$



例: FTT↓による木の変換過程

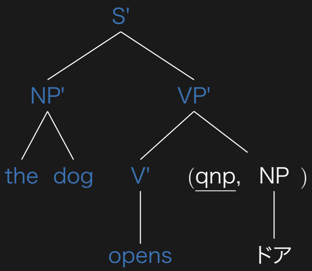
- FTT↓ $M=(Q,\Sigma,\Delta,I,R)$
  - $Q=\{\underline{\text{q}}, \underline{\text{qnp}}, \underline{\text{qv}}, \underline{\text{qvp}}\}$
  - $I=\{\underline{\text{q}}\}$
  - $\Sigma=\{$S, NP, VP, V, 犬, が, ドア, を, 開ける$\}$
  - $\Delta=\{$S', NP', VP', V', the, dog, opens, door$\}$
  - 規則集合 R:
    $(\underline{\text{q}}, \text{S}(x_1,x_2,x_3)) \rightarrow \text{S'}((\underline{\text{qnp}}, x_1)\ (\underline{\text{qvp}}, x_3))$
    $(\underline{\text{qnp}}, \text{NP}(x_1)) \rightarrow \text{NP'}(\text{the}\ (\underline{\text{qx}}, x_1))$
    $(\underline{\text{qx}}, 犬) \rightarrow \text{dog}$
    $(\underline{\text{qvp}}, \text{VP}(x_1,x_2,x_3)) \rightarrow \text{VP'}((\underline{\text{qv}}, x_3)\ (\underline{\text{qnp}}, x_1))$
    $(\underline{\text{qv}}, \text{V}(x_1)) \rightarrow \text{V'}((\underline{\text{qx}}, x_1))$



例: FTT↓による木の変換過程

- FTT↓ $M = (Q, \Sigma, \Delta, I, R)$
  - $Q = \{$q, qnp, qv, qvp$\}$
  - $I = \{$q$\}$
  - $\Sigma = \{$S, NP, VP, V, 犬, が, ドア, を, 開ける$\}$
  - $\Delta = \{$S', NP', VP', V', the, dog, opens, door$\}$
  - 規則集合 R:

    (q, S(x_1, x_2, x_3)) $\rightarrow$ S'((qnp, x_1) (qvp, x_3))
    (qnp, NP(x_1)) $\rightarrow$ NP'(the (qx, x_1))
    (qx, 犬) $\rightarrow$ dog
    (qvp, VP(x_1, x_2, x_3)) $\rightarrow$ VP'((qv, x_3) (qnp, x_1))
    (qv, V(x_1)) $\rightarrow$ V'((qx, x_1))
    (qx, 開ける) $\rightarrow$ opens



例: FTT↓による木の変換過程

- FTT↓ $M = (Q, \Sigma, \Delta, I, R)$
  - $Q = \{$q, qnp, qv, qvp$\}$
  - $I = \{$q$\}$
  - $\Sigma = \{$S, NP, VP, V, 犬, が, ドア, を, 開ける$\}$
  - $\Delta = \{$S', NP', VP', V', the, dog, opens, door$\}$
  - 規則集合 R:

    (q, S(x_1, x_2, x_3)) $\rightarrow$ S'((qnp, x_1) (qvp, x_3))

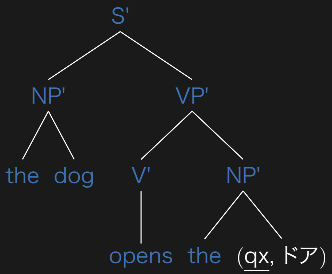
    (qnp, NP(x_1)) $\rightarrow$ NP'(the (qx, x_1))

    (qx, 犬) $\rightarrow$ dog

    (qvp, VP(x_1, x_2, x_3)) $\rightarrow$ VP'((qv, x_3) (qnp, x_1))

    (qv, V(x_1)) $\rightarrow$ V'((qx, x_1))

    (qx, 開ける) $\rightarrow$ opens



例: FTT↓による木の変換過程

- FTT↓ $M = (Q, \Sigma, \Delta, I, R)$
  - $Q = \{\underline{\text{q}}, \underline{\text{qnp}}, \underline{\text{qv}}, \underline{\text{qvp}}\}$
  - $I = \{\underline{\text{q}}\}$
  - $\Sigma = \{$S, NP, VP, V, 犬, が, ドア, を, 開ける$\}$
  - $\Delta = \{$S', NP', VP', V', the, dog, opens, door$\}$
  - 規則集合 R:

    $(\underline{\text{q}}, \text{S}(x_1, x_2, x_3)) \rightarrow \text{S'}((\underline{\text{qnp}}, x_1)\ (\underline{\text{qvp}}, x_3))$

    $(\underline{\text{qnp}}, \text{NP}(x_1)) \rightarrow \text{NP'}(\text{the}\ (\underline{\text{qx}}, x_1))$

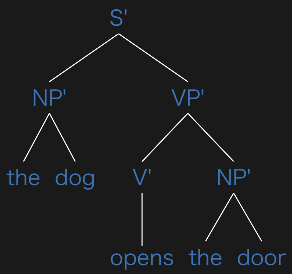
    $(\underline{\text{qx}}, \text{犬}) \rightarrow \text{dog}$

    $(\underline{\text{qvp}}, \text{VP}(x_1, x_2, x_3)) \rightarrow \text{VP'}((\underline{\text{qv}}, x_3)\ (\underline{\text{qnp}}, x_1))$

    $(\underline{\text{qv}}, \text{V}(x_1)) \rightarrow \text{V'}((\underline{\text{qx}}, x_1))$

    $(\underline{\text{qx}}, \text{開ける}) \rightarrow \text{opens}$

    $(\underline{\text{qx}}, \text{ドア}) \rightarrow \text{door}$



例: FTT↓による木の変換過程

- FTT↓ $M = (Q, \Sigma, \Delta, I, R)$
  - $Q = \{\underline{\text{q}}, \underline{\text{qnp}}, \underline{\text{qv}}, \underline{\text{qvp}}\}$
  - $I = \{\underline{\text{q}}\}$
  - $\Sigma = \{$S, NP, VP, V, 犬, が, ドア, を, 開ける$\}$
  - $\Delta = \{$S', NP', VP', V', the, dog, opens, door$\}$
  - 規則集合 R:

    $(\underline{\text{q}}, \text{S}(x_1, x_2, x_3)) \rightarrow \text{S'}((\underline{\text{qnp}}, x_1)\ (\underline{\text{qvp}}, x_3))$

    $(\underline{\text{qnp}}, \text{NP}(x_1)) \rightarrow \text{NP'}(\text{the}\ (\underline{\text{qx}}, x_1))$

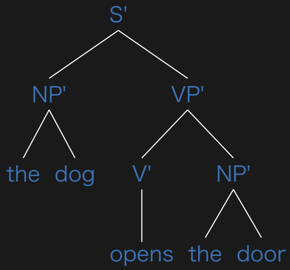
    $(\underline{\text{qx}}, \text{犬}) \rightarrow \text{dog}$

    $(\underline{\text{qvp}}, \text{VP}(x_1, x_2, x_3)) \rightarrow \text{VP'}((\underline{\text{qv}}, x_3)\ (\underline{\text{qnp}}, x_1))$

    $(\underline{\text{qv}}, \text{V}(x_1)) \rightarrow \text{V'}((\underline{\text{qx}}, x_1))$

    $(\underline{\text{qx}}, \text{開ける}) \rightarrow \text{opens}$

    $(\underline{\text{qx}}, \text{ドア}) \rightarrow \text{door}$

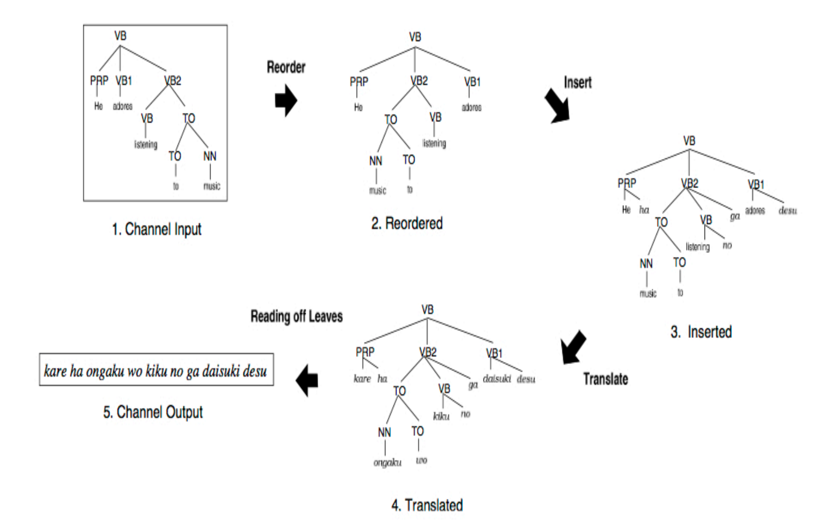


- 変換可能な木対の集合

  $Tr(M) = \{(s, t) \in T_\Sigma \times T_\Delta \mid (q, s) \Rightarrow^*_M t, q \in I\}$

構文木を使った機械翻訳の例 (1/2)

[Yamada & Knight 01]



構文木を使った機械翻訳の例 (2/2)

[Yamada & Knight 01]

original order	reordering	P(reorder)
PRP VB1 VB2	PRP VB1 VB2	0.074
	PRP VB2 VB1	0.723
	VB1 PRP VB2	0.061
	VB1 VB2 PRP	0.037
	VB2 PRP VB1	0.083
	VB2 VB1 PRP	0.021
VB TO	VB TO	0.251
	TO VB	0.749
TO NN	TO NN	0.107
	NN TO	0.893
⋮	⋮	⋮

r−table

parent	TOP	VB	VB	VB	TO	TO	...
node	VB	VB	PRP	TO	TO	NN	...
P(None)	0.735	0.687	0.344	0.709	0.900	0.800	...
P(Left)	0.004	0.061	0.004	0.030	0.003	0.096	...
P(Right)	0.260	0.252	0.652	0.261	0.007	0.104	...

w	P(ins−w)
ha	0.219
ta	0.131
wo	0.099
no	0.094
ni	0.080
te	0.078
ga	0.062
⋮	⋮
desu	0.0007
⋮	⋮

n−table

E	adores		he		i		listening		music		to		...
J	*daisuki*	1.000	*kare*	0.952	*NULL*	0.471	*kiku*	0.333	*ongaku*	0.900	*ni*	0.216	...
			NULL	0.016	*watasi*	0.111	*kii*	0.333	*naru*	0.100	*NULL*	0.204	
			nani	0.005	*kare*	0.055	*mi*	0.333			*to*	0.133	
			da	0.003	*shi*	0.021					*no*	0.046	
			shi	0.003	*nani*	0.020					*wo*	0.038	
			⋮	⋮	⋮	⋮					⋮	⋮	

t−table

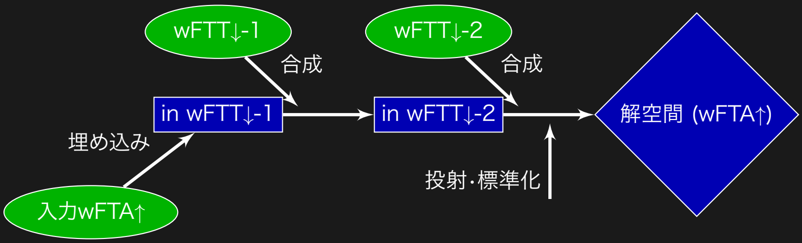
重み付きFTT↓による[Yamada & Knight 01]のモデル化

- 重み付き (w)FTT↓ $M = (Q, \Sigma, \Delta, I, R, \pi)$
  - 重み付き変換規則 $r = (q, \sigma^{(k)}(x_1, \ldots, x_k)) \xrightarrow{w} u$
  - 規則の重み $\pi(r) = w$
- [Yamada & Knight 01]のモデル化

wFTT↓	規則の例
並べ替え	$(\underline{\text{rVB}}, \text{VB}(x_1, x_2, x_3)) \xrightarrow{0.723} \text{VB}((\underline{\text{rRPR}}, x_1), (\underline{\text{rVB1}}, x_3), (\underline{\text{rVB2}}, x_1))$
	$(\underline{\text{rTO}}, \text{TO}(x_1, x_2)) \xrightarrow{0.893} \text{TO}((\underline{\text{rNN}}, x_2), (\underline{\text{rTO}}, x_1))$
	$(\underline{\text{t}}, \text{dog}) \xrightarrow{1.0} \text{dog}$
挿入	$(\underline{\text{iVB}}, \text{NN}(x_1, x_2)) \xrightarrow{0.91} \text{NN}(\text{INS}, (\underline{\text{iNN}}, x_1), (\underline{\text{iNN}}, x_2))$
翻訳	$(\underline{\text{t}}, \text{VB}(x_1, x_2, x_3)) \xrightarrow{1.0} (\text{X}, (\underline{\text{t}}, x_1), (\underline{\text{t}}, x_2), (\underline{\text{t}}, x_3))$
	$(\underline{\text{t}}, \text{dog}) \xrightarrow{0.853}$ 犬
	$(\underline{\text{t}}, \text{INS}) \xrightarrow{0.588}$ が

複数のwFTT↓を経由したデコード問題

- 2つのwFTT↓を経由した前向き段階適用



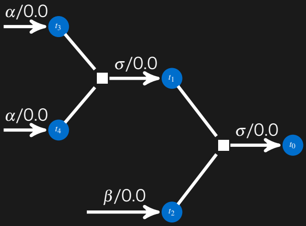
- 埋め込み、合成、値域投射を使って構成できる??

wFTA↑の埋め込み

- wFTA↑ $A = \{Q, \Sigma, F, E, \pi_1\}$の埋め込み

  $R = \emptyset$
  for all $\sigma^{(k)}(q_1, \ldots, q_k) \xrightarrow{w} q \in E$ **do**
  $\quad r \leftarrow (q, \sigma^{(k)}(x_1, \ldots, x_k)) \xrightarrow{w} \sigma^{(k)}((q_1, x_1), \ldots, (q_k, x_k))$
  $\quad R \leftarrow R \cup \{r\}$
  $\quad \pi(r) \leftarrow w$
  return $M = \{Q, \Sigma, \Sigma, F, R, \pi\}$

- 計算量 $O(|E|)$
- 入力木をwFTT↓へ変換する例

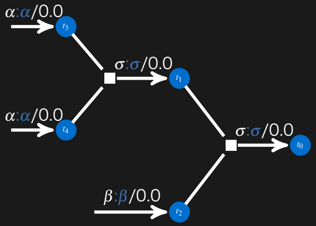


wFTA↑の埋め込み

- wFTA↑ $A = \{Q, \Sigma, F, E, \pi_1\}$の埋め込み

$R = \emptyset$
for all $\sigma^{(k)}(q_1, \ldots, q_k) \xrightarrow{w} q \in E$ **do**
    $r \leftarrow (q, \sigma^{(k)}(x_1, \ldots, x_k)) \xrightarrow{w} \sigma^{(k)}((q_1, x_1), \ldots, (q_k, x_k))$
    $R \leftarrow R \cup \{r\}$
    $\pi(r) \leftarrow w$
return $M = \{Q, \Sigma, \Sigma, F, R, \pi\}$

- 計算量 $O(|E|)$
- 入力木をwFTT↓へ変換する例



wFTT$\downarrow$の値域投射

- wFTT$\downarrow$ $M = \{Q, \Sigma, \Delta, I, R, \pi_1\}$から出力側のwRTGを出力

$P = \emptyset$
for all $q \in Q$ and $x \in X$ **do**
  $\varphi((q, x)) = q$
for all $(q, \sigma^{(k)}(x_1, \ldots, x_k)) \xrightarrow{w} t \in R$ **do**
  $r \leftarrow q \xrightarrow{w} \bar{\varphi}(t)$
  $P \leftarrow P \cup \{r\}$
  $\pi(r) \leftarrow w$
return $G = \{Q, \Delta, P, \pi, I\}$

- 計算量 $O(|R| \max_{r \in R} size(r))$
- wFTT$\downarrow$の規則をwRTGの規則へ変換する例
  - $(q, \sigma^{(k)}(x_1, x_2)) \xrightarrow{\alpha} (\beta((q_1, x_1))\ \beta((q_2, x_2))) \Rightarrow q \xrightarrow{w} \alpha(\beta(q_1)\ \beta(q_2))$
    $\Rightarrow$ wRTGは標準化によってwFTA$\uparrow$へ変換可能

FTT↓の合成と適用演算

	入力	出力	参考文献
合成	FTT↓ M_1, M_2	M_1の入力からM_2の出力を直接出すFTT↓	重みなし [Baker 79] 重み付き [Maletti 06]
前向き適用	FTA↑ A, FTT↓ M	Aが定義する木言語に対するMの値域を表すFTA↑	[May et. al. 10]
後ろ向き適用	FTA↑ A, FTT↓ M	Aが定義する木言語に対するMの定義域を表すFTA↑	[May et. al. 10]

- 文字列の場合
  - 有限トランスデューサは合成に閉じている
  - 適用の結果は正規言語 (有限オートマトン)になる
    - 前 (後ろ)向き適用 ＝ 入力 (出力)FSAの埋め込み + 合成 + 値域 (定義域)投射
    - 前 (後ろ)向き段階適用 ＝ ” + 合成 + ⋯ + 合成 + ”

FTT↓の合成と適用演算

	入力	出力	参考文献
合成	FTT↓ M_1, M_2	M_1の入力からM_2の出力を直接出すFTT↓	重みなし [Baker 79] 重み付き [Maletti 06]
前向き適用	FTA↑ A, FTT↓ M	Aが定義する木言語に対するMの値域を表すFTA↑	[May et. al. 10]
後ろ向き適用	FTA↑ A, FTT↓ M	Aが定義する木言語に対するMの定義域を表すFTA↑	[May et. al. 10]

- 文字列の場合
  - 有限トランスデューサは合成に閉じている
  - 適用の結果は正規言語 (有限オートマトン)になる
    - 前 (後ろ)向き適用 ＝ 入力 (出力)FSAの埋め込み + 合成 + 値域 (定義域)投射
    - 前 (後ろ)向き段階適用 ＝ ” + 合成 + ··· + 合成 + ”
- 木の場合
  - FTT↓の規則は対称性がない (後ろ向きの合成?)
  - 適用で正規性が保存されない、合成が閉じていない場合がある

前向き適用の正規性保存について (1/2)

- 前向きの認識力 (Forward Recognizability)
  - FTA↑をFTT↑の入力としたとき、
    その値域 (出力)は正規性を保存しているか?

制約	重みなしFTT↓	重み付き FTT↓
なし	NO	NO
非削除	NO	NO
線形	YES	??
線形 & 非削除	YES	YES

- 複写機能があると問題がある

前向きの正規性保存について (2/2)

- 正規性が保存されない例

- 複写FTT↓

$$(\underline{\mathsf{q}},\ \begin{array}{c} a \\ | \\ x_1 \end{array}) \quad \to \quad \begin{array}{c} a \\ / \qquad \backslash \\ (\underline{\mathsf{q}}, x_1) \quad (\underline{\mathsf{q}}, x_1) \end{array}$$

$$(\underline{\mathsf{q}}, b) \quad \to \quad b$$

- 単項の木 (文字列)a^*bを表すFTA↑

  - $b \to q$
  - $a(q) \to q$

$$\begin{array}{c} a \\ | \\ a \\ | \\ \vdots \\ | \\ b \end{array}$$

前向きの正規性保存について (2/2)

- 正規性が保存されない例

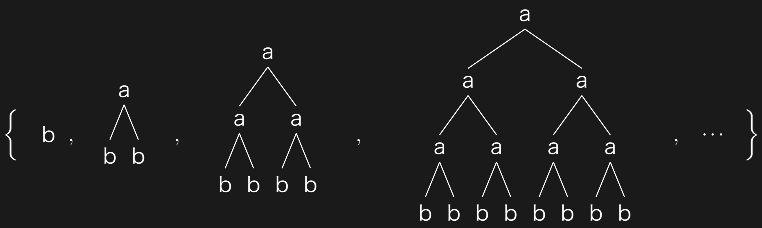
- 複写FTT↓

$$(\underline{\mathsf{q}}, a(x_1)) \to a((\underline{\mathsf{q}}, x_1), (\underline{\mathsf{q}}, x_1))$$

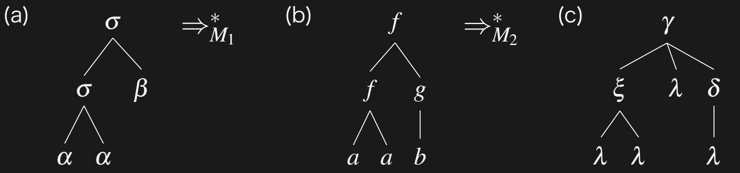
$$(\underline{\mathsf{q}}, b) \to b$$

- 単項の木 (文字列)a^*bを表すFTA↑
  - $b \to q$
  - $a(q) \to q$

- 前向き適用の結果出てくる木言語



FTT↓の合成とは?



- 木(a)から木(c)へと直接変換するFTT↓をM_1とM_2から構築する

FTT↓の合成方法[Baker 79]

- 2つのFTT↓ $M_1 = \{Q, \Sigma, \Gamma, I_1, R_1\}$と$M_2 = \{P, \Gamma, \Delta, I_2, R_2\}$
  - M_1の出力記号集合ΓとM_2の入力記号集合Γは同一

FTT↓の合成方法[Baker 79]

- 2つのFTT↓ $M_1 = \{Q, \Sigma, \Gamma, I_1, R_1\}$と$M_2 = \{P, \Gamma, \Delta, I_2, R_2\}$
  - M_1の出力記号集合ΓとM_2の入力記号集合Γは同一
- FTT↓の合成 $M_1 \circ M_2 = \{P \times Q, \Sigma, \Delta, I_2 \times I_1, R\}$:
  1. M_2の拡張
     - 入力記号の集合 $\Sigma \cup (Q \times X)$
     - 出力記号の集合 $\Delta \cup ((P \times Q) \times X)$
     - 全ての$p \in P$と$q \in Q$に対して、$(p, (q, x_i)) \rightarrow ((p, q), x_i)$を$R_2$に追加

FTT↓の合成方法[Baker 79]

- 2つのFTT↓ $M_1 = \{Q, \Sigma, \Gamma, I_1, R_1\}$と$M_2 = \{P, \Gamma, \Delta, I_2, R_2\}$
  - M_1の出力記号集合ΓとM_2の入力記号集合Γは同一
- FTT↓の合成 $M_1 \circ M_2 = \{P \times Q, \Sigma, \Delta, I_2 \times I_1, R\}$:
  1. M_2の拡張
     - 入力記号の集合 $\Sigma \cup (Q \times X)$
     - 出力記号の集合 $\Delta \cup ((P \times Q) \times X)$
     - 全ての$p \in P$と$q \in Q$に対して、$(p, (q, x_i)) \rightarrow ((p, q), x_i)$を$R_2$に追加
  2. 変換規則の構築
     - $R = \{((p, q), \sigma^{(k)}(x_1, \ldots, x_k)) \rightarrow u \mid$
       for some $(q, \sigma^{(k)}(x_1, \ldots, x_k)) \rightarrow w \in R_1, u \in Tr(p, w)\}$
     - $Tr(p, w) = \{u \in T_\Delta(((P \times Q) \times X)) \mid (p, w) \Rightarrow^*_{M_2} u\}$

例: FTT↓の合成 (1/3)

- $M_1 = \{Q, \Sigma, \Gamma, I_1, R_1\}$
  - $Q = \{\underline{\mathsf{q}_0}, \underline{\mathsf{q}_1}, \underline{\mathsf{q}_2}\}$, $I_1 = \{\underline{\mathsf{q}_0}\}$
  - $\Sigma = \{\sigma^{(2)}, \alpha^{(0)}, \beta^{(0)}\}$
  - $\Gamma = \{f^{(2)}, g^{(1)}, a^{(0)}, b^{(0)}\}$
  - $R_1 = \{(1.1), (1.2), (1.3), (1.4)\}$

(1.1) $(\underline{\mathsf{q}_0}, \sigma(x_1, x_2)) \to f((\underline{\mathsf{q}_1}, x_1), (\underline{\mathsf{q}_2}, x_2))$

(1.2) $(\underline{\mathsf{q}_1}, \sigma(x_1, x_2)) \to f((\underline{\mathsf{q}_2}, x_1), (\underline{\mathsf{q}_2}, x_2))$

(1.3) $(\underline{\mathsf{q}_2}, \alpha) \to a$

(1.4) $(\underline{\mathsf{q}_2}, \beta) \to g(b)$

- $M_2 = \{P, \Gamma, \Delta, I_2, R_2\}$
  - $P = \{\underline{\mathsf{p}_0}, \underline{\mathsf{p}_1}, \underline{\mathsf{p}_2}\}$, $I_2 = \{\underline{\mathsf{p}_0}\}$
  - $\Gamma = \{f^{(2)}, g^{(1)}, a^{(0)}, b^{(0)}\}$
  - $\Delta = \{\gamma^{(3)}, \xi^{(2)}, \delta^{(1)}, \lambda^{(0)}\}$
  - $R_2 = \{(2.1), (2.2), (2.3), (2.4), (2.5)\}$

(2.1) $(\underline{\mathsf{p}_0}, f(x_1, x_2)) \to \gamma((\underline{\mathsf{p}_1}, x_1), \lambda, (\underline{\mathsf{p}_2}, x_2))$

(2.2) $(\underline{\mathsf{p}_1}, f(x_1, x_2)) \to \xi((\underline{\mathsf{p}_2}, x_1), (\underline{\mathsf{p}_2}, x_2))$

(2.3) $(\underline{\mathsf{p}_2}, a) \to \lambda$

(2.4) $(\underline{\mathsf{p}_2}, g(x_1)) \to \delta((\underline{\mathsf{p}_2}, x_1))$

(2.5) $(\underline{\mathsf{p}_2}, b) \to \lambda$

例: FTT↓の合成 (2/3)

- $\underline{M_1 = \{Q, \Sigma, \Gamma, I_1, R_1\}}$
  - $Q = \{\underline{\mathrm{q}_0}, \underline{\mathrm{q}_1}, \underline{\mathrm{q}_2}\}$, $I_1 = \{\underline{\mathrm{q}_0}\}$
  - $\Sigma = \{\sigma^{(2)}, \alpha^{(0)}, \beta^{(0)}\}$
  - $\Gamma = \{f^{(2)}, g^{(1)}, a^{(0)}, b^{(0)}\}$
  - $R_1 = \{(1.1), (1.2), (1.3), (1.4)\}$

- $\underline{M_2 = \{P, \Gamma, \Delta, I_2, R_2\}}$の拡張
  - $\Gamma = \{f^{(2)}, g^{(1)}, a^{(0)}, b^{(0)}\} \cup \{(\underline{\mathrm{q}_1}, x_1), (\underline{\mathrm{q}_2}, x_2), \ldots\}$
  - $\Delta = \{\gamma^{(3)}, \xi^{(2)}, \delta^{(1)}, \lambda^{(0)}\} \cup \{((\underline{\mathrm{p}_1}, \underline{\mathrm{q}_1}), x_1), ((\underline{\mathrm{p}_2}, \underline{\mathrm{q}_2}), x_2), \ldots\}$
  - $R_2 = \{(2.1), \ldots, (2.5)\} \cup \{$
    $(\underline{\mathrm{p}_1}, (\underline{\mathrm{q}_1}, x_1)) \to ((\underline{\mathrm{p}_1}, \underline{\mathrm{q}_1}), x_1) \quad \cdots$ (2.6),
    $(\underline{\mathrm{p}_2}, (\underline{\mathrm{q}_2}, x_2)) \to ((\underline{\mathrm{p}_2}, \underline{\mathrm{q}_2}), x_2) \quad \cdots$ (2.7),
    $\ldots\}$

例: FTT↓の合成 (3/3)

- 合成 $M_1 \circ M_2 = \{P \times Q, \Sigma, \Delta, I_2 \times I_1, R\}$
  - $P \times Q = \{(\underline{\mathsf{p}_0}, \underline{\mathsf{q}_0}), (\underline{\mathsf{p}_1}, \underline{\mathsf{q}_1}), (\underline{\mathsf{p}_2}, \underline{\mathsf{q}_2}), \ldots\}$
  - $I_2 \times I_1 = \{(\underline{\mathsf{p}_0}, \underline{\mathsf{q}_0})\}$
- 変換規則の構築例:

(1.1) $(\underline{\mathsf{q}_0}, \sigma(x_1, x_2)) \rightarrow f((\underline{\mathsf{q}_1}, x_1), (\underline{\mathsf{q}_2}, x_2))$

(2.1) $(\underline{\mathsf{p}_0}, f(x_1, x_2)) \rightarrow \gamma((\underline{\mathsf{p}_1}, x_1), \lambda, (\underline{\mathsf{p}_2}, x_2))$

(2.6) $(\underline{\mathsf{p}_1}, (\underline{\mathsf{q}_1}, x_1)) \rightarrow ((\underline{\mathsf{p}_1}, \underline{\mathsf{q}_1}), x_1)$

(2.7) $(\underline{\mathsf{p}_2}, (\underline{\mathsf{q}_2}, x_2)) \rightarrow ((\underline{\mathsf{p}_2}, \underline{\mathsf{q}_2}), x_2)$

例: FTT↓の合成 (3/3)

- 合成 $M_1 \circ M_2 = \{P \times Q, \Sigma, \Delta, I_2 \times I_1, R\}$
  - $P \times Q = \{(\underline{\mathrm{p}_0}, \underline{\mathrm{q}_0}), (\underline{\mathrm{p}_1}, \underline{\mathrm{q}_1}), (\underline{\mathrm{p}_2}, \underline{\mathrm{q}_2}), \ldots\}$
  - $I_2 \times I_1 = \{(\underline{\mathrm{p}_0}, \underline{\mathrm{q}_0})\}$
- 変換規則の構築例:

(1.1) $\left((\underline{\mathrm{p}_0}, \underline{\mathrm{q}_0}), \sigma(x_1, x_2)\right) \to \gamma\left((\underline{\mathrm{p}_1}, (\underline{\mathrm{q}_1}, x_1)), \lambda, (\underline{\mathrm{p}_2}, (\underline{\mathrm{q}_2}, x_2))\right)$

(2.6) $(\underline{\mathrm{p}_1}, (\underline{\mathrm{q}_1}, x_1)) \quad \to \quad ((\underline{\mathrm{p}_1}, \underline{\mathrm{q}_1}), x_1)$

(2.7) $(\underline{\mathrm{p}_2}, (\underline{\mathrm{q}_2}, x_2)) \quad \to \quad ((\underline{\mathrm{p}_2}, \underline{\mathrm{q}_2}), x_2)$

例: FTT↓の合成 (3/3)

- 合成 $M_1 \circ M_2 = \{P \times Q, \Sigma, \Delta, I_2 \times I_1, R\}$
  - $P \times Q = \{(\underline{\mathsf{p}_0}, \underline{\mathsf{q}_0}), (\underline{\mathsf{p}_1}, \underline{\mathsf{q}_1}), (\underline{\mathsf{p}_2}, \underline{\mathsf{q}_2}), \ldots\}$
  - $I_2 \times I_1 = \{(\underline{\mathsf{p}_0}, \underline{\mathsf{q}_0})\}$
- 変換規則の構築例:

(1.1) $$((\underline{\mathsf{p}_0}, \underline{\mathsf{q}_0}), \sigma(x_1, x_2)) \rightarrow \gamma\big(((\underline{\mathsf{p}_1}, \underline{\mathsf{q}_1}), x_1), \lambda, (\underline{\mathsf{p}_2}, (\underline{\mathsf{q}_2}, x_2))\big)$$

(2.7) $$(\underline{\mathsf{p}_2}, (\underline{\mathsf{q}_2}, x_2)) \rightarrow ((\underline{\mathsf{p}_2}, \underline{\mathsf{q}_2}), x_2)$$

例: FTT↓の合成 (3/3)

- 合成 $M_1 \circ M_2 = \{P \times Q, \Sigma, \Delta, I_2 \times I_1, R\}$
  - $P \times Q = \{(\underline{\mathsf{p}_0}, \underline{\mathsf{q}_0}), (\underline{\mathsf{p}_1}, \underline{\mathsf{q}_1}), (\underline{\mathsf{p}_2}, \underline{\mathsf{q}_2}), \ldots\}$
  - $I_2 \times I_1 = \{(\underline{\mathsf{p}_0}, \underline{\mathsf{q}_0})\}$
- 変換規則の構築例:

(3.1) $((\underline{\mathsf{p}_0}, \underline{\mathsf{q}_0}), \sigma(x_1, x_2)) \to \gamma(((\underline{\mathsf{p}_1}, \underline{\mathsf{q}_1}), x_1), \lambda, ((\underline{\mathsf{p}_2}, \underline{\mathsf{q}_2}), x_2))$

例: FTT↓の合成 (3/3)

- 合成 $M_1 \circ M_2 = \{P \times Q, \Sigma, \Delta, I_2 \times I_1, R\}$
  - $P \times Q = \{(\underline{\mathrm{p}_0}, \underline{\mathrm{q}_0}), (\underline{\mathrm{p}_1}, \underline{\mathrm{q}_1}), (\underline{\mathrm{p}_2}, \underline{\mathrm{q}_2}), \ldots\}$
  - $I_2 \times I_1 = \{(\underline{\mathrm{p}_0}, \underline{\mathrm{q}_0})\}$
- 変換規則の構築例:

(3.1) $((\underline{\mathrm{p}_0}, \underline{\mathrm{q}_0}), \sigma(x_1, x_2)) \to \gamma(((\underline{\mathrm{p}_1}, \underline{\mathrm{q}_1}), x_1), \lambda, ((\underline{\mathrm{p}_2}, \underline{\mathrm{q}_2}), x_2))$

(3.2) $((\underline{\mathrm{p}_1}, \underline{\mathrm{q}_1}), \sigma(x_1, x_2)) \to \xi(((\underline{\mathrm{p}_2}, \underline{\mathrm{q}_2}), x_1), ((\underline{\mathrm{p}_2}, \underline{\mathrm{q}_2}), x_2))$

(3.3) $((\underline{\mathrm{p}_2}, \underline{\mathrm{q}_2}), \alpha) \to \lambda$

(3.4) $((\underline{\mathrm{p}_2}, \underline{\mathrm{q}_2}), \beta) \to \delta(\lambda)$

(3.5) $\ldots$

FTT↓の合成が正しく動作する条件

- 木対集合の合成 $Tr(M_1) \circ Tr(M_2)$:
  - $Tr(M_1) \circ Tr(M_2) = \{(s,u) \in T_\Sigma \times T_\Delta \mid$
    $(s,t) \in Tr(M_1)$ and $(t,u) \in Tr(M_2)\}$
- $Tr(M_1) \circ Tr(M_2) = Tr(M_1 \circ M_2)$が成り立つFTT↓
  - 次の2条件を満たすとき[Engelfriet 75, Baker 79]
    1. M_2が線形、または、M_1が決定性を持つ
    2. M_2が非削除、または、M_1が完全性を持つ

	M_1	M_2
(a)		線形 かつ 非削除
(b)	完全性	線形
(c)	決定性	非削除
(d)	完全性 かつ 決定性	

wFTT↓の合成について

- wFTT↓ Mの意味 ($(s,t) \in T_\Sigma \times T_\Gamma$)
  - $\tau_M(s,t) = \sum_{q \in I} \sum_{(q,s) \Rightarrow_M^{r_1} \cdots \Rightarrow_M^{r_k} t} \sum_{i=1}^{k} \pi(r_i)$
- $\tau_{M_1 \circ M_2}(s,u) = \sum_{t \in T_\Gamma} \{ \tau_{M_1}(s,t) + \tau_{M_2}(t,u) \}$が成り立つ場合

	M_1	M_2	可否	参考文献
(a)	線形 かつ 非削除	線形 かつ 非削除	YES	[Kuich 99]
		線形 かつ 非削除	YES	[Maletti 06]
(b)	完全性	線形	NO	[Maletti 06]
(c)	決定性	非削除	NO	[Maletti 06]
(d)	完全性 かつ 決定性		NO	[Maletti 06]

注意: M_2に削除機能があると問題になる[Lagoutte & Maletti 10]

ここまでの結論

- 重み付き前向き適用の正規性保存
  - 線形 & 非削除 (線形のみの場合は未証明)
- 重み付きの場合の合成が正しく動作する
  - M_2が線形 & 非削除の場合
- 合成に使う全てのwFTT↓が線形 & 非削除の場合のみ
  - 前向き適用 $=$ 埋め込み + 合成 + 値域投射
  - 前向き段階適用 $=$ 埋め込み + 合成 + $\ldots$ + 合成 + 値域投射

木に対する文法、受理機械、変換機械

- 文法・受理機械:

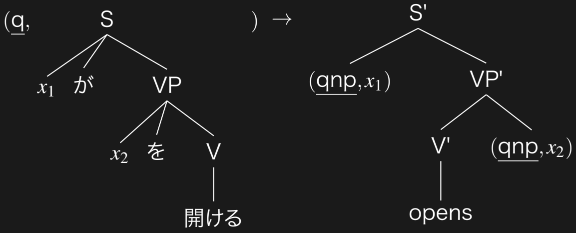
	文字列	木
文法	正規文法 文脈自由文法	正規木文法
受理機械	有限オートマトン プッシュダウンオートマトン	木オートマトン

- 変換機械:

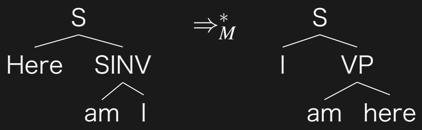
	文字列ペア	木ペア
変換機械	有限トランスデューサ	木トランスデューサ

拡張型FTT$\downarrow_{\text{[Arnold \& Dauchet 76]}}$

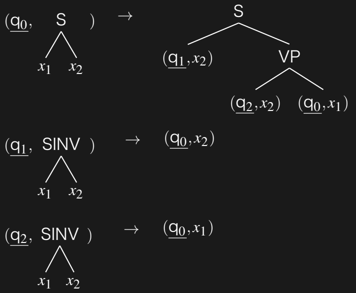
- FTT$\downarrow$の規則集合 $R \subseteq (Q \times \Sigma(X)) \times T_\Delta((Q \times X))$
  - 規則 $r \in R$: $(q, \sigma^{(k)}(x_1, \ldots, x_k)) \to u$
- 拡張型 (x)FTT$\downarrow$の規則集合 $R_{ext} \subseteq (Q \times T_\Sigma(X)) \times T_\Delta((Q \times X))$
  - 左辺に木を持つことができる



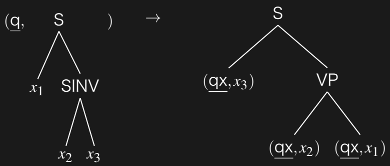
xFTT↓による局所並べ替え



- 通常のFTT↓による並べ替え:



- xFTT↓による並べ替え:



- FTT↓は複写や削除による複雑な規則が必要

xFTT↓の合成について

- xFTT↓同士の合成は正しく動作しない[Maletti 08]
- ただし、合成の第一引数がxFTT↓であれば、Bakerの方法で問題ない

	M_1 wxFTT↓	M_2 wFTT↓
(a)	線形 かつ 非削除	線形 かつ 非削除 線形 かつ 非削除

xFTT↓の前向き適用の正規性保存について

- 前向きの認識力 (Forward Recognizablity)
  - FTA↑をxFTT↑の入力としたとき、
    その値域 (出力)は正規性を保存しているか?

制約	FTT↓	重みなし	重み付き
なし	通常	NO	NO
	拡張型	NO	NO
非削除	通常	NO	NO
	拡張型	NO	NO
線形	通常	YES	??
	拡張型	YES	??
線形 & 非削除	通常	YES	YES
	拡張型	YES	YES

近年の成果のまとめ

- 「埋め込み + 合成 + 値域投射」による前向き適用
    - 線形 & 非削除のwFTT↓はNLPの問題には表現力不足
- 線形 & 非削除の拡張型wFTT↓は高い表現力を持つ
    - 合成について好意的な結論は無い
    - 前向き適用は線形のみの場合でも正規性を保存する

近年の成果のまとめ

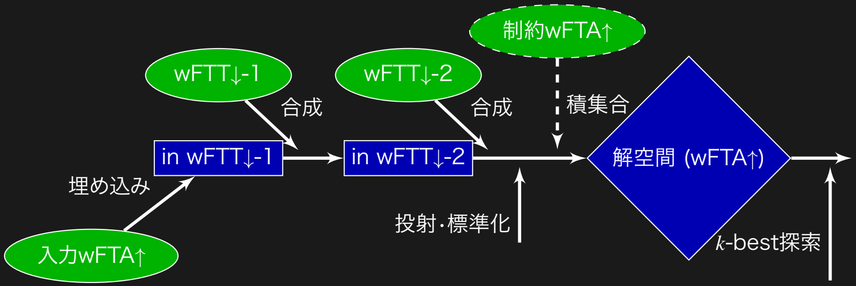
- 「埋め込み + 合成 + 値域投射」による前向き適用
  - 線形 & 非削除のwFTT↓はNLPの問題には表現力不足
- 線形 & 非削除の拡張型wFTT↓は高い表現力を持つ
  - 合成について好意的な結論は無い
  - 前向き適用は線形のみの場合でも正規性を保存する
- wxFTT↓に対して合成を使わない適用演算の開発[May et. al. 10]
  - 線形wxFTT↓を経由した前向き適用が可能
  - 後ろ向き適用についてはより好意的な結論が出ている

まとめ

- 正規木文法
  - 正規木言語、CFGとの関係
- 木オートマトンと木正規言語の性質
  - 上昇型木オートマトン、RTGとの関係
  - 閉包性、判定問題
  - 自然言語処理への応用 (積集合、k-best探索)
- 下降型木トランスデューサ
  - 合成、前向き適用、拡張型
  - 自然言語処理への応用 (前向き段階適用)

まとめ

- 正規木文法
  - 正規木言語、CFGとの関係
- 木オートマトンと木正規言語の性質
  - 上昇型木オートマトン、RTGとの関係
  - 閉包性、判定問題
  - 自然言語処理への応用 (積集合、k-best探索)
- 下降型木トランスデューサ
  - 合成、前向き適用、拡張型
  - 自然言語処理への応用 (前向き段階適用)



ソフトウェア

- 重み付きRTG、重み付き拡張型FTT↓、EM学習機能、各種演算機能、etc.
    - Tiburon: `https://github.com/isi-nlp/tiburon`
- 重み付き拡張型FTT↓、規則抽出アルゴリズム、etc.
    - T3: `http://staffwww.dcs.shef.ac.uk/people/T.Cohn/t3/`