A Machine Accepted Fuzzy Context-Free Languages

------Fuzzy Pushdown Automata

Su Lan

(Dept. of Applied Mathematics, University of Electronic Science and Tecchnology of China)

Abstact: This paper gives the concept of fuzzy pushdown automata, and the relation of a fuzzy context-free grammar and a fuzzy pushdown automaton is provided. i.e . if and only if a fuzzy language is produced by a fuzzy context-free grammar, it can be accepted by a fuzzy pushdown automaton. So fuzzy context-free languages can be recognized automatically.

Key words: fuzzy grammar; fuzzy context-free grammar; fuzzy pushdown automaton; fuzzy pattern recognition; fuzzy context-free language

0 Introduction

Because of the fuzziness existed in amost of pattern recognition problem, the reasonable method is to utilize fuzzy sets to describe and classify pattern recognition problem. For the need of practice problem, in 1969 Lee and Zadeh gave the concept of fuzzy grammars based on fuzzy sets, which brought vigour to pattern recognition. Fuzzy grammars have four types: 0 type; context-sensitive grammar; context-free grammar; finite-state grammar. For the four grammars, the different hardwares (called automata) can be designed. Author of this paper has studied the relation of a fuzzy finite-state grammar and a fuzzy finite-state automaton in (3). A fuzzy finite-state automaton is an excellent acceptor to a fuzzy finite-state language. However, it is unable to accept fuzzy context-free languages which can be accepted by fuzzy pushdown automata defined in this paper.

1 Basic Conception

Definition 1.1. 1) A Fuzzy Grammar FG is four-tuple $FG=(V_N, V_T, P, S)$, where V_N is a finite set of nonterminals, V_T is a finite set of terminals, $V_N \cap V_T = \emptyset$, S is the starting symbol, $S \in V_N$, P is a finite set of fuzzy productions of the forms: $\alpha \xrightarrow{\mu} \beta$, $\alpha, \beta \in (V_N \cup V_T)^*$, μ is called the production membership.

Note: Fuzzy grammars have four types. If the element in P is the form: $A \xrightarrow{\mu} \beta$, where $A \in V_N$, $\beta \in (V_N \cup V_T)^*$, $\mu \in (0, 1)$, then FG is called a fuzzy context-free grammar.

2) The rules of a fuzzy language L(FG) produced by a fuzzy grammar FG: If $\alpha \xrightarrow{\mu} \beta$, then $\forall \delta, \gamma \in (V_N \cup V_T)^*$, we have $\gamma\alpha\delta \xrightarrow{\mu} \gamma\beta\delta$, and $\gamma\beta\delta$ is called a derivation of $\gamma\alpha\delta$.

i. e. $L(FG)=\{(x, \mu(x)) \mid x \in V_T^*, S \xRightarrow{*} x, \mu(x)=\bigvee_{S\xRightarrow{*}\alpha}(\mu_1 \wedge \mu_2 \wedge \cdots \wedge \mu_n)\}$

Actually, a fuuzy pushdown automaton is a fuzzy finite-state automaton adding a pushdown storage, where a pushdown storage is a "first-in-later-out" storehouse.

Definition1.2: 1) A fuzzy pushdown automaton is an eight-tuple $A_{fp}=(Q, \Sigma, \Gamma, \delta, q_0, Z_0, F, f)$, where Q is a finite set of states, Σ is a finite set of input symbols, Γ is a finite set of pushdown symbols (i. e. storehouse symbols), F is a finite set of final states, $F \subset Q$, δ is a mapping from $Q \times (\Sigma \cup \{\lambda\}) \times \Gamma$ to finite set of $Q \times \Gamma^* \times (0, 1)$, Z_0 is initial storehouse symbol, q_0 is initial state. For the three-tuple of a state, a input symbol or null string λ and a storehouse symbol, every element in $\delta(q_0, a, Z)$ has a membership from f; 2) A fuzzy pushdown automaton accepts fuzzy context-free languages in two ways:

First (in final states way): $F \neq \emptyset$, when A_{fp} scans all symbols in x, it stops in a final state in F, we call A_{fp} accepts x, and $\mu(x)=\bigvee_{i=1}^{m_x}\mu_i(x)$

where A_{fp} has m_x paths to accept x, $\mu_i(x)$ is membership of the ith path.
i.e. $L(A_{fp}) = \left\{ (x, \mu(x)) \;\middle|\; x \in \Sigma^*, \mu(x) = \bigvee_{i=1}^{m_x} \mu_i(x) \right.$, when A_{fp} scans all symbols in x, it stops a final state in F $\left. \right\}$

Second (in null storehouse way): when A_{fp} scans x, it stops in a null storehouse, we call A_{fp} accepts x

i.e. $L_\lambda(A_{fp}) = \left\{ (x, \mu(x)) \;\middle|\; x \in \Sigma^*, \mu_\lambda(x) = \bigvee_{i=1}^{m_x} \mu_{i\lambda}(x) \right.$, when A_{fp} scans x, it stops in a null storehouse $\left. \right\}$

In this way, we suppose $F = \emptyset$

2 The Relation of a Fuzzy Context-free Grammar and a Fuzzy Pushdown Automaton

In this way, we suppose A_{fp} accepts input symbols in null storehouse way and $F = \emptyset$.

Theorem: A fuzzy language produced by a fuzzy context-free grammar can be accepted by a fuzzy pushdown automaton in null storehouse way.

i.e: If G_f is a fuzzy context-free grammar, a fuzzy pushdown automaton A_{fp} can be produced such that $L(G_f) = L_\lambda(A_{fp})$

Proof: Suppose $G_f = (V_N, V_T, P, S)$, we produce A_{fp} as follows:

$A_{fp} = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F, f)$, where $Q = \{q_0\}$, $\Sigma = V_T$, $\Gamma = V_N \cup V_T$, $Z_0 = S$, $F = \emptyset$,

δ and f can be obtained as follows:

i) If $A \xrightarrow{\mu} \alpha \in P$, then (q_0, α, μ) is in $\delta(q_0, \lambda, A)$. i.e. The right symbol of a production can replace the first nonterminal in storehouse, and the membership is μ.

ii) $\forall\, a \in \Sigma$, $\delta(q_0, a, a) = \{(q_0, \lambda, 1)\}$. i.e. a is ejected out of storehouse, and the membership is 1.

Furthermore, we prove $L_\lambda(A_{fp}) = L(G_f)$ as follows:

$\forall (x, \mu(x)) \in L(G_f)$, $x=a_1 a_2 \ldots a_r$, then there exist fuzzy productions in P such that $S \xRightarrow{\mu_1} a_1 \xi_1 b_1 \xRightarrow{\mu_2} a_1 a_2 \xi_2 b_2 \xRightarrow{\mu_3} \ldots \xRightarrow{\mu_{r-1}} a_1 a_2 \ldots a_{r-1} \xi_{r-1} b_{r-1} \xRightarrow{\mu_r} a_1 a_2 \ldots a_r = x$, and $\mu(x)=\mu_1 \wedge \mu_2 \wedge \cdots \wedge \mu_r$, where $a_i \in V_T$, $b_i, \xi_i \in (V_N \cup V_T)^*$, $i=1,2,\ldots,r$

From rule i) and ii) we know:

$\delta(q_0, \lambda, s)=\{(q_0, a_1 \xi_1 b_1, \mu_1)\}$, the string in storehouse is $a_1 \xi_1 b_1$;

$\delta(q_0, a_1, a_1)=\{(q_0, \lambda, 1)\}$, a_1 is ejected out of storehouse, and the string in storehouse becomes $\xi_1 b_1$;

$\delta(q_0, \lambda, \xi_1 b_1)=\{(q_0, a_2 \xi_2 b_2, \mu_2)\}$, the string in storehouse becomes $a_2 \xi_2 b_2$;

$\delta(q_0, a_2, a_2)=\{(q_0, \lambda, 1)\}$, a_2 is ejected out of storehouse, the string in storehouse becomes $\xi_2 b_2$;

....

$\delta(q_0, \lambda, \xi_{r-2} b_{r-2})=\{(q_0, a_{r-1} \xi_{r-1} b_{r-1}, \mu_{r-1})\}$, the string in storehouse becomes $a_{r-1} \xi_{r-1} b_{r-1}$;

$\delta(q_0, a_{r-1}, a_{r-1})=\{(q_0, \lambda, 1)\}$, a_{r-1} is ejected out of storehouse, the string in storehouse becomes $\xi_{r-1} b_{r-1}$;

$\delta(q_0, \lambda, \xi_{r-1} b_{r-1})=\{(q_0, a_r, \mu_r)\}$, the string in storehouse becomes a_r ;

$\delta(q_0, a_r, a_r)=\{(q_0, \lambda, 1)\}$, a_r is ejected out of storehouse, and storehouse becomes null.

Where $\mu(a_1 a_2 \ldots a_r)=\mu_1 \wedge 1 \wedge \mu_2 \wedge 1 \cdots \wedge \mu_r \wedge 1 = \mu_1 \wedge \mu_2 \wedge \cdots \wedge \mu_r$

So $(x, \mu(x)) \in L_\lambda(A_{fp})$

So $L(G_f) \subseteq L_\lambda(A_{fp})$

Similarly, we can prove $L(G_f) \supseteq L_\lambda(A_{fp})$

So $L(G_f)=L_\lambda(A_{fp})$.

3 Example

Here, the example shows that a fuzzy context-free grammar how to produce a fuzzy pushdown automaton

Example: Give a fuzzy context-free grammar $G_f=(V_N, V_T, P, S)$, where $V_N = \{S, X, Y\}$, $V_T = \{a, b\}$

P: $S \xrightarrow{0.5} aSbS$ $\quad$ $S \xrightarrow{0.6} bXb$ $\quad$ $X \xrightarrow{0.3} XbXb$

$X \xrightarrow{0.53} aba$ $\quad$ $X \xrightarrow{0.7} aSbYb$

$Y \xrightarrow{0.42} abX$ $\quad$ $Y \xrightarrow{0.8} YabY$

We produce A_{fp} as follows:

$A_{fp} = (\{q_0\}, \Sigma, \Gamma, \delta, q_0, Z_0, \aleph, f)$, where $\Sigma = \{a, b\}$, $\Gamma = \{a, b, S, X, Y\}$, $Z_0 = S$.

From rule i) we have $\delta(q_0, \lambda, S) = \{(q_0, aSbS, 0.5), (q, bXb, 0.6)\}$

$\delta(q_0, \lambda, X) = \{(q_0, aba, 0.53), (q, aSbYb, 0.7), (q, XbXb, 0.3)\}$

$\delta(q_0, \lambda, Y) = \{(q, abX, 0.42), (q, YabY, 0.8)\}$.

From rule ii): $\delta(q_0, a, a) = \delta(q_0, b, b) = \{(q_0, \lambda, 1)\}$

Now let a input string x=babab. From G_f x can be derived as follows: $S \xrightarrow{0.6} bXb \xrightarrow{0.53} babab$, and $\mu(x) = 0.6 \wedge 0.53$, so $(babab, 0.53) \in L(G_f)$

From A_{fp} x can be derived as follows:

$\delta(q_0, \lambda, S) = \{(q_0, aSb, 0.5), (q_0, bXb, 0.6)\}$, we select the narrow part. i.e. when A_{fp} scans the first symbol b in x, the string in storehouse becomes bXb:

$\delta(q_0, b, b) = \{(q_0, \lambda, 1)\}$, b is ejected out of storehouse, and the string in storehouse becomes Xb;

$\delta(q_0, \lambda, X) = \{(q_0, aba, 0.53), (q, aSbYb, 0.7), (q, XbXb, 0.3)\}$, we select the narrow part, then the string in storehouse becomes abab;

$\delta(q_0, a, a) = \{(q_0, \lambda, 1)\}$, a is ejected out of storehouse, the string in storehouse becomes bab;

$\delta(q_0, b, b) = \{(q_0, \lambda, 1)\}$, b is ejected out of storehouse, the string in storehouse becomes ab;

$\delta(q_0, a, a) = \{(q_0, \lambda, 1)\}$, a is ejected out of storehouse, the string in storehouse becomes b;

$\delta(q_1, b, b) = \{(q_0, \lambda, 1)\}$, b is ejected out of storehouse, the string in storehouse becomes null.

Where $\mu(x) = 0.6 \wedge 0.53 \wedge 1 \wedge 1 \wedge 1 \wedge 1 = 0.53$

so $(babab, 0.53) \in L_\lambda(A_{fp})$.

REFERENCE

1. J.E.Hopcraft and J.D.Ullman. Normal Languages and Their Relation to Automaton. Addison-Wesley, Reading, Massachusetts. 1969.
2. Shinichi Tamura and Kokichi Tanaka. Learning of Fuzzy Formal Languages. IEEE Trans. on Syst. Man and Cybernetics. Jan. 98-102, 1973.
3. Su Lan. The Relation of a Fuzzy 3 Type Grammar and a Fuzzy Finite-State Automaton. Applied Mathematics. 1989 (1), 111-112
4. Luo Yao-guang and Shen Li-dong. Pattern Recognition. People's Post and Telecommunications Press. 1989.