Fuzzy String Matching with Finite Automata

Armen Kostanyan

Yerevan State University
Yerevan, Armenia
e-mail: armko@ysu.am

ABSTRACT

The string matching problem is one of the widely-known symbolic computation problems having applications in many areas of artificial intelligence. The most famous algorithms solving the string matching problem are the Rabin-Karp's algorithm, finite automata method, Knuth-Morris-Pratt (KMP) algorithm [1, 2]. In this paper we focus on applying the finite automata method to find a fuzzy pattern in a text.

Keywords

String matching with finite automata, fuzzy sets, fuzzy string matching.

1. STRING MATCHING WITH A FINITE AUTOMATON

The classical string matching problem is formulated as follows [1].

We are given a text $T[1..n]$ of length n and a pattern $P[1..m]$ of length m ($n \geq m$). It is assumed that the elements of P and T are characters drawn from a finite alphabet Σ. We say that pattern P *occurs with shift* s in text T if $0 \leq s \leq n-m$ and $T[s+1..s+m]=P[1..m]$ (that is, $T[s+j]=P[j]$ for $1 \leq j \leq m$). If P occurs with shift s in T, then s is said to be a *valid shift*; else it is said to be an *invalid shift*. The ***string-matching problem*** is the problem of finding all valid shifts with which P occurs in T.

The finite automata method is based on the suffix function which is defined as follows.

Given a pattern $P[1..m]$ over an alphabet Σ, the ***suffix function*** for P is defined as a mapping σ: $\Sigma^* \to \{0, 1, \ldots, m\}$ such that

$$\sigma(x)=\max\{\, k \mid 0 \leq k \leq m,\ P_k \text{ is a suffix of } x \,\},$$

where P_k denotes $P[1..k]$.

The pattern $P[1..m]$ derives a finite automaton $M_P=(Q, q_0, F, \Sigma, \delta)$ such that

- $Q=\{0, 1, \ldots, m\}$ is the set of states;
- $q_0=0$ is the initial state;
- $F=\{m\}$ is the one-element set of final states;
- δ: $Q \times \Sigma \to Q$ is the transition function such that $\delta(q, a)=\sigma(P_q a)$ for all $q \in Q$ and $a \in \Sigma$.

Claim. Pattern $P[1..m]$ occurs with shift s in text $T[1..n]$ $\Leftrightarrow$ M_P accepts the string $T[1..s+m]$.

Once the automaton M_P is constructed, all valid shifts of P in text T can be determined in $O(n)$ time. Taking into account that with the use of the ***prefix function*** (which is another string matching function) M_P can be constructed in $O(|\Sigma| \cdot m)$ time, we get the $O(n+|\Sigma| \cdot m)$ total time for string matching with a finite automaton.

2. THE FUZZY STRING MATCHING PROBLEM

Let us generalize the classical string matching problem by formulating the problem of finding a *fuzzy pattern* in a text.

Suppose $(L, \vee, \wedge, 0, 1)$ is a finite lattice with the least element 0 and the greatest element 1. According to [3], a *fuzzy subset* A of a universal set U is defined by a membership function μ_A: $U \to L$ that associates with each element x of U a number $\mu_A(x)$ in L representing the *grade of membership* of x in A. A fuzzy subset A of U can be represented as an additive form

$$A = \sum_{x \in U} x / \mu_A(x).$$

We say that an element x definitely belongs to A, if $\mu_A(x)=1$, and it definitely does not belong to A, if $\mu_A(x)=0$. In contrast, if $0<\mu_A(x)<1$, we say that x belongs to A with degree $\mu_A(x)$.
Let us define a ***fuzzy symbol*** t over the alphabet Σ to be a fuzzy subset of Σ. Given a character $a \in \Sigma$ we say that a matches t with grade $\mu_t(a)$.

Given a set Ξ of fuzzy symbols, we define the ***fuzzy pattern*** $P[1..m]$ to be a sequence of symbols from Ξ of length m. Given a threshold $\lambda \in L$, we say that a pattern $P[1..m]$ λ-occurs in a text $T[1..n]$ with shift s, if $T[s+j]$ matches $P[j]$ with grade at least λ for all j, $1 \leq j \leq m$. We say that s is a λ-*valid shift*, if P λ-occurs in T with shift s. Finally, let us define the ***λ-fuzzy string matching problem*** to be the problem of finding all λ-valid shifts of the fuzzy pattern P in text T.

3. PROCESSING TEXT BY A TRANSITION SYSTEM

Suppose the automaton M_P with transition function δ_P has been constructed for a pattern $P[1..m]$ over the alphabet Ξ. We shall describe the solution to the λ-fuzzy string matching problem in terms of a nondeterministic transition system the states of which are pairs $s=\langle q, \alpha \rangle$, where $0 \leq q \leq m$, α is a sequence of L-values of length q. We interpret the state $s=\langle q, \langle \alpha_1, \ldots, \alpha_q \rangle \rangle$ in the following way: if $T[h+1], \ldots, T[h+q]$ is the sequence of the last q read characters, then

$$\alpha_i = \mu_{P[i]}\big(T[h+i]\big),\ \ 1 \leq i \leq q.$$

More precisely, given $\lambda \in L$, consider the transition system $(S, s_0, \Phi, \Sigma, \Delta)$, where

- $S=\{\, s=\langle q, \alpha \rangle \mid 0 \leq q \leq m, \alpha=\langle \alpha_1, \ldots, \alpha_q \rangle, \alpha_i \in L$ for all $1 \leq i \leq q \,\}$;
- $s_0=\langle 0, \langle\rangle \rangle$ is the start state;
- $\Phi=\{\, s=\langle m, \langle \alpha_1, \ldots, \alpha_m \rangle \rangle \mid \alpha_i \geq \lambda, 1 \leq i \leq m \,\}$ is the set of final states;
- Σ is the text alphabet;
- Δ: $S \times \Sigma \to 2^S$ is the transition function such that

$$\left[q < m, q \xrightarrow{t} (q+1) \in \delta_P\right] \Rightarrow$$

$$\left\langle q, \left\langle \alpha_1, ..., \alpha_q \right\rangle\right\rangle \xrightarrow{a} \left\langle q+1, \left\langle \alpha_1, ..., \alpha_q, \mu_t(a) \right\rangle\right\rangle \in \Delta,$$

$$\left[q \xrightarrow{t} q' \in \delta_P, q' \le q\right] \Rightarrow$$

$$\left\langle q, \left\langle \alpha_1, ..., \alpha_q \right\rangle\right\rangle \xrightarrow{a} \left\langle q', \left\langle \alpha_{q-q'+2}, ..., \alpha_q, \mu_t(a) \right\rangle\right\rangle \in \Delta,$$

for all $\alpha_1, \ldots, \alpha_q \in L$, $a \in \Sigma$;

There are no other transitions in Δ.

Suppose ω: $\Sigma^* \rightarrow 2^S$ is the final-state function for the transition system above such that

$\omega(\varepsilon) = \{ s_0 \}$,

$\omega(xa) = \{ s' \mid$ there exists $s \in \omega(x)$ such that $s' \in \Delta(s, a) \}$.

Theorem. $\omega(T_{s+m}) \cap \Phi \neq \varnothing \Leftrightarrow s$ is a λ-valid shift.

4. EXAMPLE

Let us choose $\Sigma = \{ 1, 2, 3, 4, 5 \}$, $L = \{ 0, 0.25, 0.5, 0.75, 1 \}$ and define **SMALL** and **LARGE** fuzzy symbols as follows:

SMALL= **1/1 + 2/0.75 + 3/0.5 + 4/0.25 + 5/0,**

LARGE= **1/0 + 2/0.25 + 3/0.5 + 4/0.75 + 5/1.**

Assume that Ξ=**{SMALL, LARGE}**, ***P***= **SMALL.LARGE.SMALL** (here "." is used as a separator of symbols from Ξ). **Fig. 1** below presents the diagram of the automaton M_P:

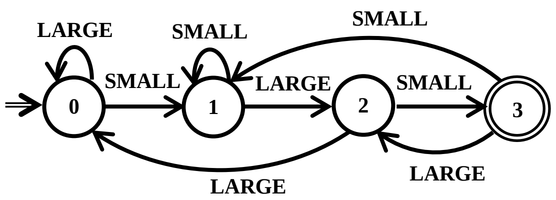

Fig. 1

Assuming that ***T*=32415231**, let us consider the following processing of ***T*** by the transition system ***S***:

$$s_0 = \langle 0, \langle\ \rangle\rangle \xrightarrow{3} s_1 = \langle 0, \langle\ \rangle\rangle$$

$$\xrightarrow{2} s_2 = \langle 1, \langle 0.75 \rangle\rangle$$

$$\xrightarrow{4} s_3 = \langle 2, \langle 0.75, 0.75 \rangle\rangle$$

$$\xrightarrow{1} s_4 = \langle 3, \langle 0.75, 0.75, 1 \rangle\rangle^*$$

$$\xrightarrow{5} s_5 = \langle 2, \langle 1, 1 \rangle\rangle$$

$$\xrightarrow{2} s_6 = \langle 3, \langle 1, 1, 0.75 \rangle\rangle^*$$

$$\xrightarrow{3} s_7 = \langle 2, \langle 0.75, 0.5 \rangle\rangle$$

$$\xrightarrow{1} s_8 = \langle 3, \langle 0.75, 0.5, 1 \rangle\rangle.$$

It follows from this execution of ***T*** that at λ=**0.75** we have two final states (that is, s_4 and s_6) and, respectively, two **0.75**-valid shifts (that is, **1** and **3**). At λ=**0.5** we would have three final states (that is, s_4, s_6 and s_8) and three **0.5**-valid shifts (that is, **1, 3** and **5**).

5. PARTIAL FUZZY STRING MATCHING

The transition system constructed in Section 3 can be restricted in one or another way to construct an approximate algorithm that finds some of the occurrences of a fuzzy pattern in a given text. One of such approximate algorithms is provided below in which the transition of the automaton M_P most suitable for next symbol of the given text is always chosen.

In the description of the algorithm we shall use the *singled-valued* function $\gamma_{qq'}$: $L^q \times \Sigma \rightarrow L^{q'}$ defined for all $0 \le q$, $q' \le m$ in the following way:

$$\gamma_{qq'}(\alpha, a) = \alpha' \Leftrightarrow \langle q', \alpha' \rangle \in \Delta(\langle q, \alpha \rangle, a)$$

for all $\alpha \in L^q$ and $a \in \Sigma$.

Let us denote by $pr_1(s)$ and $pr_2(s)$ the first and second components of the state $s \in S$, respectively. Finally, for $\alpha = \langle \alpha_1, \ldots, \alpha_k \rangle$ denote by $\min(\alpha)$ the least component of α, i. e., $\alpha_1 \wedge \ldots \wedge \alpha_m$.

Algorithm. PARTIAL FUZZY STRING MATCHER.
***Input*:** Fuzzy pattern ***P*[1..*m*]**, text ***T*[1..*n*]**, threshold λ ($0<\lambda\le 1$).
***Method*:**

```
currState=<0,<>>
for i = 1 to n
    maxGrade=0
    for all t∈Ξ
        if μt(T[i])>maxGrade
            maxGrade=μt(T[i])
            sym=t
    currState=<q', γqq'(pr2(currState), T[i])>,
                where q= pr1(currState), q'=δP(q, sym)
    if pr1(currState)==m && min(pr2(currState))≥λ
        Print ("Pattern λ-occurs with shift", i – m)
```

Note, that this algorithm recognizes all **0.75**-valid shifts from the example in Section 4. On the other hand, to recognize the **0.5**-valid shift **5**, the algorithm must perform the **LARGE**-transition among two transitions with the same rate **0.5** while reading the second character **3** from state **3**.

The complexity of the algorithm is $O(n \cdot (|\Xi| + m))$. Considering the $O(m \cdot |\Xi|)$ time needed for the construction of the automaton M_p, we get $O(n \cdot (|\Xi| + m)) + O(m \cdot |\Xi|) = O(n \cdot (|\Xi| + m))$ total time for partial fuzzy string matching.

6. CONCLUSION

The problem of finding occurrences of a fuzzy pattern in a given text with a given accuracy has been considered in this paper. A nondeterministic transition system is constructed to describe the set of all possible ways of processing the pattern reading the text. This transition system is restricted to obtain a $O(n \cdot (|\Xi| + m))$-time algorithm for finding some of the occurrences of a fuzzy pattern in the given text.

7. ACKNOWLEDGEMENT

This work was supported by the RA MES State Committee of Science, in the frames of the research project N 15T-18350.

REFERENCES

[1] T. Cormen, C. Leiserson, R. Rivest, C. Stein, "Introduction to Algorithms", 3rd edition, *The MIT Press*, 2009.

[2] B. Smyth, "Computing Patterns in Strings", *Addison-Wesley UK*, 2003.

[3] L. A. Zadeh, "The Concept of a Linguistic Variable and its Application to Approximate Reasoning-I", *Information Sciences 8*, 199-249, 1975.