String Parsing Using a Fuzzy Context-Free Pattern

Armen Kostanyan
IT Educational and Research Center
Yerevan State University
Yerevan, Armenia
email: armko@ysu.am

Anna Gevorgyan
American University of Armenia
Yerevan, Armenia
email: anna_gevorgyan4@edu.aua.am

***Abstract*—This paper is devoted to determining the degree of compliance of a given string with a pattern represented as a context-free grammar, the terminal symbols of which are fuzzy properties of the characters of the base alphabet. Parsing is performed by converting the fuzzy context-free pattern into a fuzzy context-free grammar over the base alphabet, after which the fuzzy version of the Cocke-Younger-Kasami (CYK) algorithm is applied to determine the measure by which the given string can be parsed according to the given pattern. The proposed approach makes it possible to classify strings in the basic alphabet based on a fuzzy context-free pattern.**

This work may find application in bioinformatics to classify DNA sequences using fuzzy prototypes described in one way or another. Another application concerns the parsing of natural languages, where fuzzy methods have been used for a long time and give good results.

***Keywords*— Fuzzy context-free grammar, Fuzzy context-free pattern, Fuzzy parsing.**

I. Introduction

The fuzzy context-free grammar, being a direct extension of the ordinary context-free grammar, was introduced to model ambiguity in entities under consideration [1]. In such a grammar, the production rules have a measure of their applicability, which makes the language generated by the grammar fuzzy. Fuzzy grammar should not be confused with probabilistic grammar. It generates a fuzzy language, i.e., a language the strings of which have different degrees of membership, while a probabilistic grammar determines the most probable derivation for a given string belonging to the language [2].

The fuzzy approach to domain analysis has proven itself well in the field of machine learning and bioinformatics. In [3] and in other publications, the concept of fuzzy context-free grammar was used to parse sequences of natural language sentences. In [4], it was proposed to use the concept of fuzzy context-free grammar for the analysis of DNA sequences. It is pointed out that although the DNA of two different individuals of the same species may differ due to mutations occurring over time, the basic similarities remain. In view of the above, the concept of a fuzzy prototype of DNA sequences was developed and its representation was proposed using a fuzzy context-free grammar. As a result, the problem of determining the measure of compliance of a DNA sequence with a prototype turns into the problem of determining the matching degree of a string to a fuzzy context-free language. The latter is then solved by applying a fuzzy version of the CYK algorithm [5], [6].

This paper is in line with the works on fuzzy parsing of a crisp string. In contrast to the traditional approach, where a fuzzy context-free language is defined by a grammar with fuzzy production rules and crisp terminals, we propose to use a crisp context-free grammar over an auxiliary alphabet, the characters of which are interpreted as fuzzy sets over the base alphabet. This approach comes from the research of the authors in the field of fuzzy string matching using a linear fuzzy pattern the symbols of which are replaced by separate characters [7] or sequences of characters [8] of the input string. In [9], the concept of matching a string with a fuzzy linear pattern has been generalized to the case of a pattern represented by a finite automaton with fuzzy properties of alphabetic characters on transitions. The problem of determining the matching degree of a crisp string to the language of such an automaton was solved in [10].

In this paper, the fuzzy pattern is further generalized to context-free grammar with fuzzy properties of alphabetic characters as terminals. The matching problem for such a pattern is solved by converting the latter into a fuzzy context-free grammar in the Chomsky normal form, after which an adapted fuzzy version of the CYK algorithm is applied.

The paper is organized as follows. Section 2 introduces the concept of a fuzzy context-free pattern and formulates the fuzzy parsing problem. Section 3 presents a solution to the fuzzy parsing problem by transforming a given fuzzy pattern into a fuzzy context-free grammar in the Chomsky normal form and applying to it an adapted fuzzy version of the CYK algorithm. Finally, the conclusion summarizes the obtained results.

II. Preliminaries

A. Context-free grammar and Chomsky normal form

Given a finite alphabet Σ, we define a *context-free grammar* (in short, *CF grammar*) as a 4-tuple $G = (N, \Sigma, P, S)$, where

- N is a finite non-empty set of non-terminals,
- Σ is a finite set of terminals, $N \cap \Sigma = \emptyset$,
- $P \subseteq N \times (N \cup \Sigma)^*$ is a finite set of production rules. If $(A, \alpha) \in P$, we say A produces α and write $A \to \alpha$,
- $S \in N$ is a start symbol.

A CF grammar G defines a binary relation "$\Rightarrow$" over the set $(N \cup \Sigma)^*$, called *direct inference* and defined as follows:

$$\alpha \Rightarrow \beta \text{ , iff } \alpha = \alpha_1 A \alpha_2, \beta = \beta_1 \gamma \beta_2, A \to \gamma \in P.$$

https://doi.org/10.51408/csit2023_02

The reflexive and transitive closure of the direct inference relation is called an *inference relation* and is denoted by "$\Rightarrow^*$". The language of the CFG G is defined as

$$L(G) = \{x \in \Sigma^* \mid S \Rightarrow^* x\}.$$

CF grammars G_1 and G_2 are called equivalent if

$$L(G_1) = L(G_2).$$

CF grammar G is represented in the *Chomsky Normal Form* (*CNF*), if each production rule is in one of the following forms:

$$A \to BC \text{ or } A \to a,$$

where A, B, C are non-terminals and a is a terminal. It is found that for every context-free grammar that does not produce an empty string, there exists an equivalent context-free grammar in the Chomsky normal form.

B. Fuzzy sets

Suppose that $(M, \leq, 0, 1, \otimes)$ is a finite linearly ordered set of *measures* with the smallest element 0, the largest element 1, and the monotonic accumulation operation $\otimes$ such that M is a commutative monoid with a zero element 0 and a unit element 1. That is, for all $a, b, c \in M$

$$a \otimes 0 = 0, a \otimes 1 = a, a \leq b \Rightarrow a \otimes c \leq b \otimes c.$$

According to the fuzzy set theory, the *fuzzy subset* X of the universal set U is defined by the membership function $\mu_x : U \to M$ that associates with each element u from U the value $\mu_x(u) \in M$, representing the *degree of belonging* of u to X. A fuzzy subset X from U can be represented in the additive form

$$X = \sum_{u \in U} u/\mu_X(u).$$

We say that an element u certainly belongs to X if $\mu_X(u) = 1$, and it certainly does not belong to X if $\mu_X(u) = 0$. On the contrary, if $0 < \mu_X(u) < 1$, we say that u belongs to X with the degree $\mu_X(u)$. Given a fuzzy subset X of U, we define a *supporting set* for X as a subset of U with no elements that certainly does not belong to X, i.e.,

$$supp(X) = \{u \in U \mid \mu_X(u) > 0\}.$$

C. Fuzzy symbols and fuzzy linear patterns

Given an alphabet Σ of characters, we define a *fuzzy symbol* α over Σ as a fuzzy subset of Σ, i. e., $\alpha : \Sigma \to M$. For the character $c \in \Sigma$ and the fuzzy symbol α, we say that c matches α with degree $\alpha(c)$.

We define a *linear fuzzy pattern* over Σ to be any finite length sequence of fuzzy symbols over Σ. For a linear fuzzy pattern $\omega = \omega_1 \ldots \omega_n$ and a crisp string $x = a_1 \ldots a_n \in \Sigma^*$ of the same lengths, the degree of matching x to ω is defined as

$$\mu_\omega(x) = \mu_{\omega_1}(a_1) \otimes \ldots \otimes \mu_{\omega_n}(a_n) \in M.$$

Example 2.1: Let M be the segment [0, 1] of ordered reals with the accumulation operation defined as multiplication. Assume that $\Sigma = \{1, 2, 3, 4, 5\}$ and define the fuzzy symbols S (small), M (middle) and L (large) as the following fuzzy subsets of Σ:

$$S = 1/1 + 2/0.75 + 3/0.5 + 4/0.25 + 5/0,$$
$$M = 1/0 + 2/0.75 + 3/1 + 4/0.75 + 5/0,$$
$$L = 1/0 + 2/0.25 + 3/0.5 + 4/0.75 + 5/1.$$

Let $x = 24513$, $\omega_1 = SMLSM$, $\omega_2 = MSLSL$. Then

$$\mu_{\omega_1}(x) = \mu_S(2) \otimes \mu_M(4) \otimes \mu_L(5) \otimes \mu_S(1) \otimes \mu_M(3) = {}^3/_4 \cdot {}^3/_4 \cdot 1 \cdot 1 \cdot 1 = {}^9/_{16}.$$

$$\mu_{\omega_2}(x) = \mu_M(2) \otimes \mu_S(4) \otimes \mu_L(5) \otimes \mu_S(1) \otimes \mu_L(3) = {}^3/_4 \cdot {}^1/_4 \cdot 1 \cdot 1 \cdot {}^1/_2 = {}^3/_{32}.\Box$$

D. Fuzzy context-free pattern and the fuzzy parsing problem

We say that a CF grammar Π with a set Ξ of terminals is a *fuzzy context-free pattern* (in short, *fuzzy CF pattern*) over the base alphabet Σ if the elements of Ξ are interpreted as fuzzy symbols over Σ. That is, a mapping $f_\xi : \Sigma \to M$ is defined for all $\xi \in \Xi$.

A fuzzy context-free pattern Π over Σ as a crisp grammar over Ξ defines a crisp CF language $L(\Pi) \subseteq \Xi^*$. On the other hand, it defines a fuzzy language $L'(\Pi) \subseteq \Sigma^*$ such that for all $x \in \Sigma^*$ the degree $\mu_{L'[\Pi]}(x)$ of matching x to $L'(\Pi)$ is defined as

$$\mu_{L'(\Pi)}(x) = \max\{\mu_\omega(x) \mid \text{for all } \omega \in L(\Pi) \text{ such that } |x| = |\omega|\}.$$

Given a fuzzy context-free pattern Π over Σ and a crisp string $x \in \Sigma^*$, we define the Π-*based fuzzy parsing problem of* x as the problem of determining the value $\mu_{L'(\Pi)}(x)$.

Example 2.2: Let $\Sigma = \{1, 2, 3, 4, 5\}$, $\Xi = \{\xi_S, \xi_M, \xi_L\}$, and the elements of Ξ are interpreted on Σ as fuzzy symbols S, M, and L from *Example* 2.1, respectively. Assume that the fuzzy CF pattern Π is defined as the following CF grammar in CNF:

$$A \to AA \mid AB \mid \xi_S,$$
$$B \to BA \mid \xi_L.$$

Let $x = 214$ be the string to be parsed using the pattern Π.

Considering that in $L(\Pi)$ there are the following strings (or, linear fuzzy patterns) of length 3

$$\omega_1 = \xi_S\xi_S\xi_S, \omega_2 = \xi_S\xi_S\xi_L, \omega_3 = \xi_S\xi_L\xi_S, \omega_4 = \xi_S\xi_L\xi_L,$$

we get

$$\mu_{L'(\Pi)}(x) = \max\{\mu_{\omega_1}(x), \mu_{\omega_2}(x), \mu_{\omega_3}(x), \mu_{\omega_4}(x)\} = \max\{{}^3/_4 \cdot 1 \cdot {}^1/_4, {}^3/_4 \cdot 1 \cdot {}^3/_4, 0, 0\} = {}^9/_{16}.\Box$$

E. Fuzzy context-free grammar

To solve the Π-based fuzzy parsing problem of x, we use the *fuzzy context-free grammar* as an intermediate structure, which is defined as follows [6].

For a given set M of measures, the *fuzzy CF grammar* is a CF grammar with the production rules of the form $A \xrightarrow{\mu} \alpha$, where $A \in N$, $\alpha \in (N \cup \Sigma)^*$, $\mu \in M$ and $\mu > 0$.

A fuzzy CF grammar produces a fuzzy binary relation on the set $(N \cup \Sigma)^*$ with the membership degree μ of the pair (α, β) defined as

$$\mu = max\{\nu \mid \alpha = \alpha_1 A \alpha_2, \beta = \beta_1 \gamma \beta_2, A \xrightarrow{\nu} \gamma \in P\}.$$

This relation is called a *fuzzy direct inference* and is denoted as $\alpha \overset{\mu}{\Longrightarrow} \beta$.

Any sequence

$$h = \alpha_0 \overset{\mu_1}{\Longrightarrow} \alpha_1 \overset{\mu_2}{\Longrightarrow} \dots \overset{\mu_{k-1}}{\Longrightarrow} \alpha_{k-1} \overset{\mu_k}{\Longrightarrow} \alpha_k$$

of fuzzy direct inferences such that $k \geq 0, \alpha_0 = \alpha, \alpha_k = \beta$, is called a *fuzzy h-inference* of β from α and is denoted as $\alpha \overset{h}{\Longrightarrow} \beta$. The *measure* of the fuzzy h-inference is defined as

$$\mu(h) = \mu_1 \otimes \dots \otimes \mu_k \in M.$$

A fuzzy CF grammar G over Σ defines a fuzzy language $L(G) \subseteq \Sigma^*$ such that for all $x \in \Sigma^*$ the membership degree $\mu_G(x)$ is defined as

$$\mu_G(x) = max\{\mu(h) \mid \text{for all } h \text{ such that } S \overset{h}{\Longrightarrow} x\}.$$

We say that the fuzzy CF grammar is represented in the CNF if each production rule is in one of the following forms:

$$A \xrightarrow{\mu} BC \text{ or } A \xrightarrow{\mu} a,$$

where A, B, C are non-terminals, a is a terminal, and $\mu \in M$.

Example 2.3: Let M be defined as in *Example* 2.1. Assuming that the set of non-terminals is $\{A, B\}$, the set of terminals is $\{a, b\}$ and the start symbol is A, consider the following production rules:

$$A \xrightarrow{0.3} AA, A \xrightarrow{0.5} AB, A \xrightarrow{1} a,$$
$$B \xrightarrow{0.4} BA, B \xrightarrow{1} b.$$

Let $x = aba$. Consider the following two derivations of x from A:

$$h_1 = A \overset{0.3}{\Longrightarrow} AA \overset{0.5}{\Longrightarrow} ABA \overset{1}{\Longrightarrow} aBA \overset{1}{\Longrightarrow} abA \overset{1}{\Longrightarrow} aba,$$
$$h_2 = A \overset{0.5}{\Longrightarrow} AB \overset{0.4}{\Longrightarrow} ABA \overset{1}{\Longrightarrow} aBA \overset{1}{\Longrightarrow} abA \overset{1}{\Longrightarrow} aba.$$

These two derivations have the following measures: $\mu(h_1) = 0.15$, $\mu(h_2) = 0.2$ (due to commutativity of the accumulation operation, any other derivation of x from A will have one of these two measures). Thus, x is derived from A with measure $max\{0.15, 0.2\} = 0.2$. □

III. Parsing with fuzzy CF pattern

A. Transformation of a fuzzy CF pattern into a fuzzy CF grammar

Let $\Pi = (N, \Xi, P, S)$ be a fuzzy CF pattern in CNF over the basic alphabet Σ. Suppose that for each $\xi \in \Xi$, $supp(f_\xi)$ is a finite subset of M.

Consider a fuzzy CF grammar $G(\Pi) = (N, \Sigma, P', S)$ with a set of production rules defined in the following way:

- If $A \to BC \in P$, then $A \xrightarrow{1} BC \in P'$,
- If $A \to \xi \in P$, then $A \xrightarrow{f_\xi(a)} a \in P'$ for all $a \in supp(f_\xi)$,
- There are no other production rules in P'.

(Note that the *non-terminal* production rules of P' are crisp.)

The obtained set of production rules can then be reduced by skipping the measures of crisp productions and replacing the set $A \xrightarrow{\mu_1} a, \dots, A \xrightarrow{\mu_k} a$ of *terminal* productions differing in measures, with the only terminal production $A \xrightarrow{max\{\mu_1, \dots \mu_k\}} a$ with maximum measure.

Example 3.1: In accordance with this transformation, the fuzzy CF pattern from *Example* 2.2 is transformed into the following fuzzy CF grammar:

$$A \to AA, A \to AB, A \to 1, A \xrightarrow{0.75} 2, A \xrightarrow{0.5} 3, A \xrightarrow{0.25} 4,$$
$$B \to BA, B \xrightarrow{0.25} 2, B \xrightarrow{0.5} 3, B \xrightarrow{0.75} 4, B \to 5. \square$$

The following theorem shows that the fuzzy parsing problem can be solved using the fuzzy grammar $G(\Pi)$.

Theorem: $\mu_{L'(\Pi)}(x) = \mu_{G(\Pi)}(x)$ for all $x \in \Sigma^*$. □

B. Parsing using fuzzy CF grammar

In applied works (for example, [6]), the problem of parsing based on the fuzzy CF grammar is solved using one or another fuzzy version of the CYK algorithm. Taking into account the characteristics of grammar $G(\Pi)$, we can slightly modify the fuzzy version of the CYK algorithm.

Suppose we are given a crisp string $x = a_1 \dots a_n$ in the alphabet Σ and a fuzzy CF grammar G in CNF over the set of terminals Σ with crisp *non-terminal* production rules. For $1 \leq i \leq j \leq n$, denote by X_{ij} a fuzzy subset of non-terminals of G that produce the substring $a_i \dots a_j$ with a non-zero measure of inference. Obviously, the value $\mu_G(x)$ is the degree of membership of S in X_{1n} if S belongs to X_{1n}, and 0, otherwise.

Consider the following inductive way to calculate the sets X_{ij} for all $1 \leq i \leq j \leq n$.

Basis:

$$X_{ii} = \{A/\mu \mid A \xrightarrow{\mu} a_i \text{ is a } \textit{terminal} \text{ production rule}\}.$$

Induction: If $i < j$ and the subsets $X_{i'j'}$ with less than $(j - i)$ difference between the upper and lower indices are already defined, then

$$\begin{aligned} X_{ij} = \{A/\mu \mid \mu = \max\{\mu_1 \otimes \mu_2\} & \text{ for all } \mu_1 \text{ and } \mu_2 \\ & \text{such that there is a production } A \to BC \\ & \text{and an integer } k, i \leq k < j, \text{ such that} \\ & B/\mu_1 \in X_{ik} \text{ and } C/\mu_2 \in X_{k+1,j}\}. \end{aligned}$$

Example 3.2: Consider the fuzzy CF grammar constructed in *Example* 3.1:

$$A \to AA, A \to AB, A \to 1, A \xrightarrow{0.75} 2, A \xrightarrow{0.5} 3, A \xrightarrow{0.25} 4,$$
$$B \to BA, B \xrightarrow{0.25} 2, B \xrightarrow{0.5} 3, B \xrightarrow{0.75} 4, B \to 5.$$

Parsing the string $x = 24513$ using the fuzzy version of the CYK algorithm described above, can be done as follows.

Subsets with index difference 0:

$X_{11} = \{A/_{3/4}, B/_{1/4}\}, X_{22} = \{A/_{1/4}, B/_{3/4}\}, X_{33} = \{B\},$
$X_{44} = \{A\}, X_{55} = \{A/_{1/2}, B/_{1/2}\}.$

Subsets with index difference 1:

$X_{12} = \{A/_{9/16}, B/_{1/16}\}, X_{23} = \{A/_{1/4}\}, X_{34} = \{B\},$
$X_{45} = \{A/_{1/2}\}.$

Subsets with index difference 2:

$$X_{13} = \{A/_{9/16}, B/_{1/16}\}, X_{24} = \{A/_{1/4}\}, X_{35} = \{B/_{1/2}\}.$$

Subsets with index difference 3:

$$X_{14} = \{A/_{9/16}, B/_{1/16}\}, X_{25} = \{A/_{1/8}\}.$$

Subsets with index difference 4:

$$X_{15} = \{A/_{9/32}, B/_{1/32}\}.$$

Thus, $\mu_G(24513) = \mu_{X_{15}}(A) = 9/32$. □

C. Solving the fuzzy parsing problem

Summarizing the above, we get the following algorithm for parsing the string x using the fuzzy pattern Π in CNF.

1) Transform Π into a fuzzy CF grammar $G(\Pi)$.
2) Use the fuzzified CYK algorithm to determine the membership degree $\mu_{G(\Pi)}(x)$.

Example 3.3: Let us consider a crisp string $x = 24513$ and a fuzzy pattern Π with production rules

$$A \to AA \mid AB \mid \xi_S,$$
$$B \to BA \mid \xi_L.$$

Suppose that the terminal symbols ξ_S and ξ_L are interpreted as fuzzy sets S (small) and L (large) over the universal set $\Sigma = \{1, 2, 3, 4, 5\}$, as it was done in *Example 2.1*. It follows from *Example 3.2* that parsing $x = 24513$ according to Π gives a parsing degree $\mu_{L'(\Pi)}(24513) = 9/32$. Note that this degree is obtained based on the inference

$$A \Rightarrow AB \Rightarrow ABB \Rightarrow ABBA \Rightarrow ABBAA \Rightarrow^5$$
$$\Rightarrow^5 \xi_S \xi_L \xi_L \xi_S \xi_S = \omega,$$

and the fact that $\mu_\omega(x) = \mu_S(2) \otimes \mu_L(4) \otimes \mu_L(5) \otimes \mu_S(1) \otimes \mu_S(3) = 3/4 \cdot 3/4 \cdot 1 \cdot 1 \cdot 1/2 = 9/32$.

D. Analysis

The complexity of the proposed algorithm is the sum of the complexities of the transformation step and the fuzzy parsing using the modified CYK algorithm.

Let n be the length of the input string x, p be the number of production rules of the pattern Π, q be the sum of the sizes of the supporting sets of terminal symbols of Π when they are interpreted as fuzzy subsets of Σ. Observe that the transformation step has a complexity $O(p + q)$ and results in a fuzzy CF grammar of size $O(p + q)$. The fuzzy version of the CYK algorithm (like its original version) runs in $O(n^3 s)$ time, where s is the size of the input grammar. Thus, the proposed algorithm has complexity $O(p+q)+O(n^3(p+q)) = O(n^3(p + q))$.

IV. Conclusion

A special approach to fuzzy parsing based on the fuzzy properties of alphabetic characters is proposed in this paper. Unlike the traditional approach, where fuzzy parsing is defined based on a fuzzy context-free grammar, we define fuzzy parsing based on a fuzzy pattern, which is an ordinary context-free grammar, the terminal symbols of which are interpreted as fuzzy subsets of the underlying alphabet. The solution to the considered problem is obtained by converting the fuzzy pattern into a fuzzy context-free grammar followed by applying a fuzzy version of the CYK algorithm to determine the parsing degree of the input string.

The proposed approach can be applied in all areas where fuzzy analysis methods are used. Possible examples include artificial intelligence systems and DNA sequence analysis.

Acknowledgment

This work was supported by the Ministry of Education, Science, Culture and Sports of the Republic of Armenia, project 21T-1B326.

References

[1] H. Alfawareh and S. Jusoh, *Resolving Ambiguous Entity through Context Knowledge and Fuzzy Approach*, International Journal on Computer Science and Engineering (IJCSE), vol. 3, no. 1, 2011.

[2] P. Asveld, *A Bibliography on Fuzzy Automata, Grammars and Languages*, Bulletin of The European Association for Theoretical Computer Science, pp. 187-196, 1995.

[3] S. Kanakaraddi, V Ramaswamy, *Natural Language Parsing using Fuzzy Simple LR (FSLR) Parser*, pp. 1337-1341, 2014.

[4] H. Molina-Lozano, E. Vallejo-Clemente, J. Morett-Sanchez, *DNA Sequence Analysis Using Fuzzy Grammars*, 2008 IEEE International Conference on Fuzzy Systems (FUZZ 2008), pp. 1915-1921, 2008.

[5] P. Asveld, *Fuzzy context-free languages—Part 1: Generalized context-free grammars*, Theoretical Computer Science vol. 347, no. 1-2, pp. 167 – 190, 2005.

[6] P. Asveld, *Fuzzy context-free languages—Part 2: Recognition and parsing algorithms*, Theoretical Computer Science 347, pp. 191 – 213, 2005.

[7] A. Kostanyan, *Fuzzy String Matching using Prefix Table, Transactions of IIAP NAS RA, Mathematical Problems of Computer Science*, vol. 54, pp.116 -121, 2020.

[8] A. Kostanyan, A. Harmandayan, *Mapping a Fuzzy Pattern onto a String*, in Proc. on *2019 IEEE Conference "2019 Computer Science and Information Technologies" (CSIT)*, Yerevan, Armenia, 23-27 Sep., 2019, IEEE Press, pp. 5 – 8, USA 2019.

[9] A. Kostanyan, *Fuzzy String Matching with Finite Automata*, in Proceedings on 2017 IEEE Conference CSIT-2017, Yerevan, Armenia, IEEE Press, pp. 25-2, USA 2018.

[10] A. Kostanyan, *Determining the Degree of Fuzzy Regularity of a String*, Transactions of IIAP NAS RA, Mathematical Problems of Computer Science, vol. 56, pp. 56-64, 2021.