

CLOSURE PROPERTIES OF REGULAR LANGUAGES USING MYHILL-NERODE THEOREM

¹Shrawan Kr. Pandey, ¹Manish Kr. Gupta, ²Abhinandan Tripathi, ²Ranjeet Singh, ³Chayanika Srivastava

¹Assistant Professor, ²Assistant Professor,

^{1,2} Department of Computer Science & Engineering

^{1,2} Buddha Institute of Technology, GIDA, Gorakhpur, India

Abstract—Closure properties is a very important properties for formal languages used in automata theory. A certain type of language under some operation yield same type of language, then it is said to be language is closed under that operation. Example is that regular language under union, intersection, concatenation and some others is closed. Myhill-Nerode Theorem is a very important theorem for determining whether a given language is regular or not, because it gives necessary and sufficient condition for a given language to be regular language, on the basis of finite number of equivalence classes.

Keywords- DFA, Closure properties, Formal languages, Union, Intersection, Concatenation, Myhill-Nerode theorem, Regular language, Equivalence classes.

I. INTRODUCTION

DFA is a deterministic finite automata machine which recognizes strings of regular languages. It is a pure recognising or validating machine which can only recognize or validate input strings. It does not give any output corresponding to input strings. DFA is defined by 5-tuple definition i.e. $(Q, \Sigma, \delta, q_0, F)$, where:

Q : It is a finite nonempty set of states.

Σ : It is a finite nonempty set of input alphabets.

δ : It is a transition function, which define how transition of state will take place from one state to another state according to mapping rule:

$\delta: Q \times \Sigma \rightarrow Q$

q_0 : It is one of the states used as a start state or initial state for the purpose of to begin recognition of input strings, where $q \in Q$.

F : It is non empty set of final states or accepting states, where F is subset of set of states i.e. Q .

Myhill-Nerode Theorem:

Myhill- Nerode theorem states that Language L is a regular language if and only if the set of equivalence classes of L is finite.

OR

1. Language L can be divided into set of all possible strings into separate (mutually exclusive) classes.
2. If L is a regular language then number of classes created is finite.
3. If number of classes that L has is finite, then L is a regular language.

Equivalence Classes:

Classes: Let $L = (a+b)^*a + (a+b)^*b$

There is two classes-

$C_a = (a+b)^*a$ ending in a.

$C_b = (a+b)^*b$ ending in b.

A binary relation is said to be equivalence relation if and only if it is

1. Reflexive
2. Transitive
3. Symmetric

There is a natural equivalence relation between the strings of finite automata.

Suppose DFAM= (Q, E, δ , q_0 , F) for $x, y, z \in \Sigma^*$, a relation R_m is defined as follows

$xR_my \iff \delta^*(q_0, x) = (q_0, y)$

Reflexive:

$\because \delta^*(q_0, x) = \delta^*(q_0, y) \quad \forall x \in \Sigma^*$

$\therefore xR_mx$

$\therefore R_m$ is reflexive.

Transitive:

$\because \delta^*(q_0, x) = \delta^*(q_0, y)$ and $\delta^*(q_0, y) = \delta^*(q_0, z)$

$\therefore \delta^*(q_0, x) = \delta^*(q_0, z)$

It means that if xR_my and yR_mz then xR_mz .

$\therefore R_m$ is transitive.

Symmetric:

$\because \delta^*(q_0, x) = \delta^*(q_0, y)$

$\therefore \delta^*(q_0, y) = \delta^*(q_0, x)$

$\therefore R_m$ is symmetric.

$\therefore R_m$ is equivalence classes.

The equivalence relation R_m divides the set Σ^* into equivalence classes. The number of equivalence classes is known as index and it is finite always for regular language.

II. STATE OF ART

In existing approach of proving closure properties of regular language used properties of regular expression which is given below:

UNION: Let R and S be two regular expression of equivalent regular language L_R and L_S respectively. $L_R \cup L_S$ will be also regular language because $R+S$ being also regular expression.

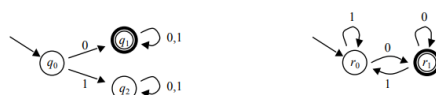
$L_R \cup L_S = \{a \mid a \in L_R \text{ or } a \in L_S \text{ (or both)}\}$

Example: –

$L_R = \{0a \mid a \in \{0,1\}^*\} = \text{strings start with 0}$

$L_S = \{a0 \mid a \in \{0,1\}^*\} = \text{strings end with 0}$

$L_R \cup L_S = \{a \in \{0,1\}^* \mid a \text{ either will starts with 0 or ends with 0 (or both)}\}$ [10].



$L_1 = \{0x \mid x \in \{0,1\}^*\}$

$L_2 = \{x0 \mid x \in \{0,1\}^*\}$

$M_1 = (Q, \Sigma, \delta_1, q_0, F_1)$

$M_2 = (R, \Sigma, \delta_2, r_0, F_2)$

$L_1 = L(M_1)$

$L_2 = L(M_2)$

FIG 1: Union of Two DFAs

Here the concept used basically of set theory. Regular expression is actually algebraic fashion representation of regular language. If we view regular language as a set then, regular language is a set of strings. R is a one set, S is another set, and $R+S$ is actually union of these two set that is actually $R \cup S$.

INTERSECTION: Let R and S be two regular expression of equivalent regular language L_R and L_S respectively. $L_R \cap L_S$ will be also regular language because $R \cap S$ being also regular expression. In this also set concept have used to prove that regular language is closed under intersection operation.

Let " $M_1 = (Q_1, \Sigma, \delta_1, q_1, F_1)$ " and " $M_2 = (Q_2, \Sigma, \delta_2, q_2, F_2)$ " be DFAs recognizing L_1 and L_2 , respectively.

"Idea: Run M_1 and M_2 in parallel on the same input and accept if both M_1 and M_2 accept. Consider $M = (Q, \Sigma, \delta, q_0, F)$ defined as follows".

$$Q = Q_1 \times Q_2$$

$$q_0 = \langle q_1, q_2 \rangle$$

$$\delta(\langle p_1, p_2 \rangle, a) = \langle \delta_1(p_1, a), \delta_2(p_2, a) \rangle$$

$$F = F_1 \times F_2$$

M accepts $L_1 \cap L_2$ What happens if M_1 and M_2 where NFAs? Still works! Set $\delta(\langle p_1, p_2 \rangle, a) = \delta_1(p_1, a) \times \delta_2(p_2, a)$ [9].

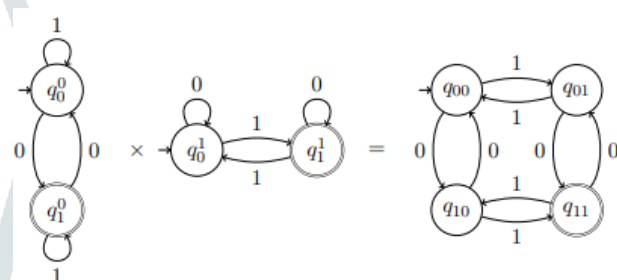


FIG 2: Example of Intersection

In same way there is proof for other closure properties concatenation, difference, complement etc.

III. PROPOSED METHOD

The main motive of to write this paper to acknowledge peoples that we can prove closure properties of regular languages using Myhill-Nerode theorem in more detail understanding of strings of different class of particular regular language.. In my approach using Myhill-Nerode theorem, I have used number of finite classes that have particular regular language in proving closure properties of regular language. In my approach, there is more exploration of strings of regular language. We know that formal languages are set of strings. In case of regular language we can divide language in disjoint subsets of many language that is known as class according to Myhill-Nerode theorem. In proofs, I have used max and min function as a tool to prove closure properties of regular languages. Now we will use these concepts in proving closure properties of regular language.

IV. IMPLEMENTATION

IV.I. CLOSURE UNDER UNION

If L and M are two regular languages, so union of these two regular languages L and M , that is $L \cup M$ will be also regular Language.

Proof: If L and M are regular languages, they have finite number of classes. Suppose L and M have C_L and C_M number of classes respectively. Therefore C_L and C_M are finite according to Myhill-Nerode Theorem because L and M are regular languages.

Let LUM have number of classes is C_U . Value C_U cannot less than greater between C_L and C_M and cannot exceed to $C_L + C_M$.

Since least number of classes in LUM may be C_L when classes in M is subset of classes in L or least number of classes in LUM may be C_M when classes in M is subset of classes in L. Maximum number of classes in LUM may be $C_L + C_M$ when L and M have disjoint set of classes. Therefore following inequalities hold:

$$\max(C_L, C_M) \leq C_U \leq C_L + C_M$$

(Using set theory union concept)

Therefore

C_U have four possible values :

- C_L
- C_M
- $C_L + C_M$
- Any value between C_L or C_M and $C_L + C_M$

All four values are finite.

So number of classes in LUM is finite.

∴ LUM is regular language using Myhill_Nerode theorem concept..

IV.II. CLOSURE UNDER INTERSECTION

If L and M are two regular languages, then intersection of these two regular languages L and M that is $L \cap M$ will be also regular language.

Proof: If L and M are regular languages, so they will have finite number of classes. Suppose number of classes in L and M are C_L and C_M respectively.

Therefore C_L and C_M are finite.

Let $L \cap M$ have number of classes is $C_{\cap}$, Minimum possible number of class in intersection of languages L and M will be 0 when L and M have disjoint set of class in their languages. It means there is no common class. Maximum number of class in $L \cap M$ will be either C_L when classes in L is subset of classes in M or C_M when classes in M is subset of classes in L. Therefore following inequalities hold:

$$0 \leq C_{\cap} \leq \min(C_L, C_M)$$

(Using set theory intersection concept)

With above inequalities it is clear value of $C_{\cap}$ is between 0 and C_L (or C_M), both inclusive, therefore number of classes in $L \cap M$, that is $C_{\cap}$ is finite. Therefore $L \cap M$ is regular language using Myhill –Nerode theorem concept.

IV.III. CLOSURE UNDER CONCATENATION

If L and M are two regular languages, then concatenation of these two regular languages L and M, that is LM will be also regular language.

Proof: If L and M are regular languages, so they will have finite number of classes. Suppose number of classes in L and M are C_L and C_M respectively. Let concatenation of regular languages L and M, which is LM have number of classes C_{LM} .

Since regular languages L and M are set of strings and their concatenated language i.e. LM is similar to Cartesian product of L and M i.e. $L \times M$. $L \times M$ will have number of classes equal to number of class in second language i.e. equal to C_M . This is because regular language formed by concatenation of L and M i.e. LM will inherits properties of second language i.e. of regular language M.

Therefore equalities hold:

$$C_{LM} = C_M$$

Therefore number of classes in concatenated language LM is always equal to second regular language i. e. equal to number of classes in regular language M. Hence number of classes in language LM is finite. Hence concatenated language LM formed by concatenation of two regular languages L and M is always a regular language.

IV.IV. CLOSURE UNDER KLEENE CLOSURE

If L is regular language then language formed by closure operation this language i.e. L^* will be also a regular language.

Let L is regular language having finite number of classes

i.e. C_L . Let number of classes in L^* is C^* .

Proof: Since L^* will have same set of class as in L except to extra class of null string class if L not have. So following inequalities will hold for C^* .

$$C^* \leq C_L + 1$$

$\therefore C_L$ is finite.

$\therefore C^*$ is finite.

And so L^* is regular language.

IV.V. CLOSURE UNDER DIFFERENCE

If L and M are two regular languages, then difference of these two regular languages L and M that is $L-M$ will be also regular language.

Proof: Let L , M and $L-M$ have C_L , C_M , and C_{L-M} number of classes. $L-M$ will have all classes in L excluding common classes in M . If there is no common class between classes of regular languages L and M then language $L-M$ will have classes equal to classes in regular language L .

Therefore following inequalities will satisfy for C_{L-M} .

$$C_L \leq C_{L-M} \leq C_L - C_M$$

(Using set theory concept)

Since value of C_{L-M} is between two finite number, Therefore using concept of Myhill-Nerode theorem difference of two regular languages L and M i.e. $L-M$ is also a regular language.

IV.VI. CLOSURE UNDER COMPLEMENTATION

The complement of a regular language L (w.r.t. an alphabet set Σ) i.e. $\Sigma^* - L$ is also regular language.

Σ^* is always have finite number of classes i.e. always equal to $|\Sigma| + 1$. $\Sigma^* - L$ will have all classes in Σ^* excluding classes in L .

Since Σ^* is regular language, the complement of a regular language is always regular.

Proof: Let Σ have n finite element, let us it is equal to n .

Let Σ^* have number of classes C_{Σ^*} .

$$\therefore C_{\Sigma^*} = n + 1$$

Let $\Sigma^* - L$ have number of classes $C_{\Sigma^* - L}$.

$$\text{Then } C_{\Sigma^* - L} = (n + 1) - C_L$$

(Using set theory concept)

i.e. finite.

(where C_L is number of classes in L) Since $(n + 1) - C_L$ is a finite number being n and C_L finite number. Therefore using Myhill-Nerode theorem concept complement of regular language L is also regular language.

IV.VII. CLOSURE UNDER REVERSAL

If L is regular language then L^R is also regular language.

If number of classes in L is finite, then number of classes in

L^R is also finite because action of reversal of each string in L form set of strings i.e. equal to language L^R .

Therefore L^R is regular.

We can also prove other properties in same way.

V. CONCLUSION

It is very easy to prove closure properties of regular languages using Myhill-Nerode but more importantly is to explore Myhill-Nerode theorem in new dimension i.e. importance of Myhill-Nerode theorem beyond the proving of given language is regular or not. I have tried to explore to use of class in regular language that is middle way between individual string and whole language. Actually I have not ever seen use of class of regular language except to Myhill-Nerode theorem itself, so maybe it can be use in future in other dimension.

REFERENCES

- [1] Henzinger, Tom, Lecture 7: Myhill–Nerode Theorem.
- [2] Hopcroft, John E.; Ullman, Jeffrey D. , “ Introduction to Automata Theory, Languages, and Computation,” Addison-Wesley Publishing, ISBN0- 201-02988-X.
- [3] Nerode, Anil , "Linear Automaton Transformations", Proceedings of the AMS 9, JSTOR 2033204.
- [4] Regan, Kenneth , Notes on the Myhill-Nerode Theorem.
- [5] Manish Kumar Gupta, Amity Journal of Computational Sciences, Vol 1, Issue 1, pp. 22- 24, 2017
- [6] Anish Gupta, Manish K. Gupta, HIVE- Processing Structured Data in HADOOP, International Journal of Scientific & Engineering Research Vol. 8, Issue 6, June 2017
- [7] <https://courses.engr.illinois.edu/cs373/sp2013/Lectures/lec08.pdf>
- [8] <https://homepage.cs.uri.edu/faculty/hamel/courses/2014/spring2014/csc445/lecture-notes/ln445-03.pdf>
- [9] Yu.I. Sorkin, The algebra of automata, Problemi Kibernetiki (1961)
- [10] A.A. Letichevskii, On the synthesis of finite automata, Doklady Akad. Nauk U.S.S.R. No. 2 (1961)
- [11] S. Huzino, On the existence of a Sheffer stroke class in sequential machines, Mem. Pac. Sci. Kyusyu Univ. series A13.
- [12] L.A. Skorniyakov, Nerve systems, uspekhi Matem. Nauk 13 vyp. 3, (81), (1958), 233-234