Finite Automata Application in String Identification

Kuldeep Vayadande[1], Harsh Gujar[1], Vivek Ingle[1], Shubham Deokate[1], Yash Chungade[1], Karan Gawali[1]

[1] Vishwakarma Institute of Technology, Pune, India

Abstract. Automata Theory is very useful in programming languages. It can have various applications. One of its applications is for the evaluation of regular expressions. In this paper, the discussion of pattern matching is done. The given input will be searched for the pattern. This proposed model will give the information about the content of the given input string. This concept needs a complicated programming model. Many techniques are present for the process of pattern matching. Finite Automata can be used in the pattern matching process to identify the patterns. It can also be used for making memory efficient by minimizing the number of states, minimize the number of transitions.

Keywords: Pattern matching machine, finite automata, C++.

1 Introduction

Method of taking input and performing some required operations at the center to provide the output based on a set of rules or algorithms is called Computation [2]. Automata principle is the study of abstract machines and automata. It also deals with computational problems and their solutions. It's miles a concept in theoretical PC technological know-how.

A finite state machine abbreviated as FSM, can be used for representing any language . A language is what basically consists of a set of strings. Finite automata are used in diverse fields in solving complex problems, the use of algorithms and different techniques [1]. Finite automata requires input as a string from enter tape hence the enter tape is split into cells and each cellular can have one enter image [2].

Finite automata is basically divided into two subtypes, deterministic finite automata (DFA) and nondeterministic which is also known as finite type of automata(NFA) [5]. Non-deterministic finite automaton is nothing but a finite set with one start state and a set of accepting states. It allows 0, 1 or more transitions from a state for the same input symbol. A DFA consists of a finite set of states and the finite set of some input tapes. This allows you to switch while having an input symbol other than the same input symbol.

Pattern Matching checks a given sequence of tokens/strings for the presence of some pattern. Basically input string is compared with a predefined stored pattern[6]. This paper discusses the

same concept. It tells what the input string contains. Whether it contains a number, character or a symbol. This paper discusses finite automata construction based on programming. This machine is created by using C++ programming language.

2 Literature Review

Below are some of the papers which have been referred while doing research on the topic :

Jiwei Xue, Ygagao Li and author of eighty Nan [1] use Finite Automata with a basic schooling which is one of the key to organizing a lifelong studying software. The corresponding paper describes a finite type automata theory, and to ensure records protection, prioritizes -automaton restrained to reveal and filter out text statistics entered using well suited or asynchronous communique gear furnished by the network have a look at.

Ms.J.Nirmala , Mrs. V.Rajathi authors [4] used numerous finite automata set of rules, size and time complexity is reduced the use of various techniques. Programming languages are used for building numerous types of finite automata and to accept binary input strings. This paper intended to observe distinctive processes of finite automata creation.

Writer Robert L. [3] Constable used generally described automata styles which might be reflected in laptop behavior, programming language structures and device session policies. Systematic take a look at of those patterns has mounted laptop concept, to offer thoughts, techniques and paradigm for a waysachieving and hidden consequences in many components of pc modem theory.

Mikael Pettersson in this [6] paper introduces a new integration set of rules to suit the time period term pattern of purposeful languages. previous algorithms may additionally produce replica code, as well as undesirable or whole bias exams for sure pattern combinations, specifically if the pattern column incorporates a aggregate of developers and variations. This mentioned algorithm, stimulated by the finite automata concept, addresses these issues and solve them to some degree them to a few degree.

Bofivoj Melichar in this [9] paper evaluation of man or woman unit is almost a sequence so it's miles feasible to resolve it using a confined automata. infinite stop automaton is designed to fit cables with k. It shows, that "flexible gadget" and "shift and based" algorithms mimic this countless automata. The corresponding DFA has the shape of O (mTM), where m is duration of pattern and the quantity ok is the distinction. The complex time of the supported algorithms together with the determining automata is O (n), wherein n is textual content period.

Ramanpreet Singh and Ali A. Ghorbanin [5] of the textual content mine, vector website online and bag of word fashions are poor students of identify acquisition as they are organized in phrase order and repetition, which could be very important in understanding the which means of file shape and other - event, very important in know-how the that means of the textual content.

Krishna Kishore Thota, R. JebersonRetna Raj -a refractory novelgerex model [8] bear in mind a tour robotic designed to enhance the show with near-up operation. the important thing idea of the walk automaton has modified the characteristic inside the distribution of various tour numbers and a short time later we implemented it to the modified Deterministic Finite Automaton, referred to as a tour robotic. The reputable scale of the navigation robotic manner, which rewrites the random set of steps in the shifting machine .We also established a strategy for crossing the go with the flow-duration circulation with the target that deception may be reduced to a positive degree.
Vennila Santhanam, on this paper evaluation of ordinary expression is performed.

3 Methodology

The proposed system will be used to match the pattern of the string. For this purpose we have used a finite state machine. This machine will read input and these characters from input will be passed through various finite states.

The machine is limited to taking input of only numbers and words. A total of four states have been created for this purpose. The need of states is that it enables reading characters and transitioning to new states.Initially we have a list of finite states. When the input character is read based on the output of that state, then the transition is limited to the new state. The reading of input character transition to new state is done continuously till the input character no longer corresponds to the output. When this is achieved it can be said that pattern matching is done.

In short we can say that the finite machine which we have proposed is able to identify the type of character present in the input string. No specific character is identified, only its type is identified as to whether it is number or word. It can identify negative numbers also. In simple words it can be said that the output of our program is whether the input is a number or character.

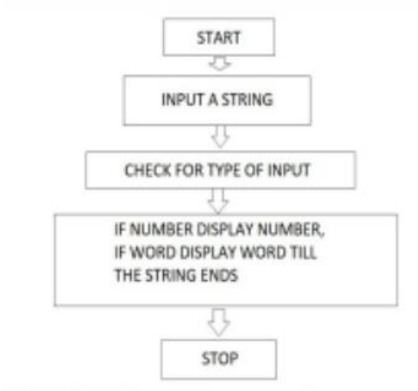


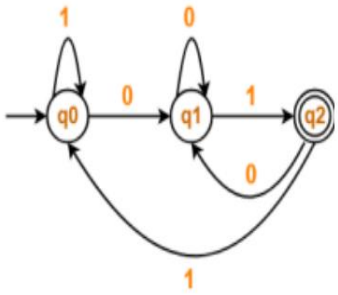
Fig 1. Flowchart

3.1 Algorithm

Below is the algorithm used for doing pattern matching:

- List of finite states is created.
- Input is read.
- Based on the output of these firstly created states, transition to a new set is done.
- A continuous reading of input characters is done and transition to new states takes place.
- When input character no longer corresponds to any of the corresponding output, then it can be said matching is done.
- Once it is done, the result will get displayed according to the input.

Let us understand DFA with an example. The below diagram is of a DFA which consists of three states(q0, q1, q2). The input string will be accepted by the DFA then only when it passes through two states (q0,q1) and reaches the final state (q2). It will pass the q0 state when the character of the input string is 0 and 1. Similar is for q1 and q2. At last if the input string reaches to final state q2, we can say that the string is accepted.



DFA

Fig 2. DFA

3.2 Graph

Below is the graph of time complexity of a novel finite automata:

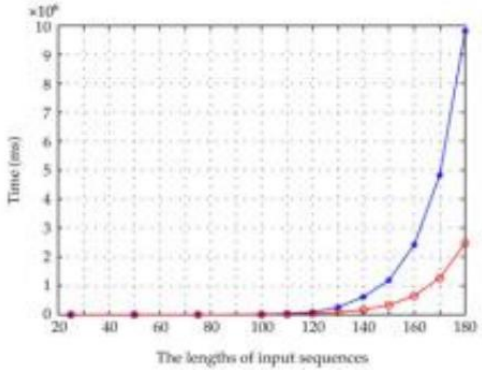


Fig 3. Time complexity graph[19]

4 Types of Automata

An automaton is a machine that scans the input unit and either accepts or rejects it. The input unit is accepted when the automaton reaches its final state (accept) after we "read it". Thread reading is done using one mark at a time. Then use the transformation function, the determination of what the next state will be, is done. If the automata is not in the final receiving state, the character unit is rejected or accepted.

Finite automata can be divided into two subgroups, DFA or NFA. The DFA determines, which means that from one region to another it is different. In NFA, unlimited automata, conversions can be made from a single region to several different regions by "reading" only one symbol. Automata are usually represented by a target graph where the arrows represent the function of the change.

Finite automata is used in a variety of fields for complex problem solving problem using different algorithms and techniques. There are two common algorithms used to match the pattern:

- KMP (Knuth-Morris-Pratt)
- BM (Boyer-Moore)

Both algorithms use the same methods. I the complexity of the algorithms takes the time of the line: O (m + n),where 'm' the length of the cord, and n its length file.

There are some different finite automata approaches:
- Algorithmic and problematic approach construction
- Finite automata construction based on programming
- Finite automata applications in various fields
- Approaches based on performances

Finite Automata is a state-of-the-art machine which uses a set of symbols as it inserts and changes their shape based on those symbols. Finite automata can also serve as a standard speech sensor. When a standard speech unit is provided as an input to finite automata, it transforms its status into a limited automaton, and also changes its real state of each. When the thread is fed successfully to an automaton and when the automata reaches its final state, it is considered valid as a sign language token. A finite automaton mathematical model consists of:

i . Complete set of conditions (Q)
ii. Complete set of input markers (Σ)
iii. Initial condition (q0)
iv. Final set (qf)
v. Change function (δ)

The transition function (δ) locations a map of the completed state (Q) to a confined a fixed of symbols for input (Σ), $Q \times \Sigma \rightarrow Q$.

Let's see an instance of a confined automata construction:

- Allow L (r) to be the standard language conceded by finite automata (FA).
- Countries: The FA areas are depicted as circles. The names of the regions are written in circles
- Initial condition: The initial state refers to the state in which the automata begins. The first shape is indicated by an arrow.
- Medium conditions: All Central Provinces have at least two arrows; Showing one another.

- Final state: The automaton should be in this position if the input unit is correctly transmitted. The final state of automata is symbolized by two circles. It can have odd number of arrows which are pointing at it. Odd = even + 1 has more bizarre arrows than the equation, i.e. odd = even + 1.
- Transformation: When the requested character is found in the input, the transition from one mode to another occurs. The automata can either advance to the next place or stay in the same location during conversion. A directional arrow indicates movement from one place to another, with the arrow pointing to the destination. An arrow leading from the position to the automaton is drawn if it remains in the same place.

4.1 Benefits of using Finite Automata:

- Familiarity: the use of a preferred FSM set machine permits for brief know-how via following a logical collection of occasions.

- Speedy improvement: the use of FSM because the first assignment template manner that the conditions under which it operates are already predetermined, properly locked, and sequential. All this is wanted is the details of the challenge.

- Prediction: the use of FSM, the tool can switch to a limited variety of regions. As a result, any asset that makes use of the same FSM will show the equal behavior as something else.

- Reliability: there is one condition that works at anyone time, which greatly reduces the chance of unexpected errors or sudden conduct within the gadget. In actual-world situations,mistakes are much less possibly to occur, particularly throughout operation. A common example for lots is the engine that fails to show off after pressing the stop button. the usage of FSM, if the system is not in excellent circumstance, it ought to not take place that the engine just pops up or remains grew to become on incorrectly.

- Safety: it's miles plenty less difficult to manipulate the output of a coincidence safety device during operation. via preventing country alternate, for instance, initiation, it is easier to mix sports and prevent surprising conduct.

- Overall performance: What occurs while a person presses a forestall at some point of the begin cycle? have to we be capable of press this button? It changed into easily reduce and changed using FSM. when the system starts off evolved, the subsequent kingdom does now not stop. Or, if important, status can be a priority at any time inside the system; therefore, it can be established as a dominant nation for others. in preference to cautiously adjusting the operating situations of hundreds of traces of code, with FSM, you clearly exchange what ‘forestall’ is described as within the device.

- Tracking: OEE is a beneficial metrics for corporations, big and small, to see how the device works. FSMs can without difficulty be used to impeach successes, screw ups, or spoilage in a system.

4.2 Finite Automata limitations

i. The FA can only list restricted inputs.

ii. No confined automaton can come across and hit upon a fixed binary unit of O's & 1's equal equivalent.

iii. Series of the unit over "(and)" and feature stability brackets.

iv. The input tape is study-most effective and the reminiscence handiest it has, say to mention.

v. It can simplest have twine patterns.

vi. Head motion in a single route most effective.

vii. Few algorithms really check whether sure characters are same or now not. No arithmetic operation is performed.

5 Future Scope

Finite automata are not only important in automata theory research and formal language, but also a very important explanatory tool provided by all employers present at all multiple nodes for improve analysis speed and time. This creates a rich metadata index of the chorus. The detail deposit in the document model can be used to perform a variety of text analysis tasks, such as searching for logical titles in a specific set of documents.

This field has the potential to be used for many other functions such as quiz, word order, word-based structure, abbreviation and keyword extraction. Researching these extensions will be exciting in the future. The proposed system discusses such an application where the input is sent to a limited automated pattern matching automata. Matching pattern using finite automata is very good.

6 Result and Conclusion

This paper uses the Finite State Automata model to monitor the input provided. It tells the story of what is in the string of a given unit. Whether there is a letter, number, symbol or words available. In addition it can also be used to test text based on user text. Test results show that this method works.

Below is a image of an output which we got after giving a input string as "Hello 123 -456 world"-

```
Found a word
Found a number
Found a number
Found a word
Program ended with with exit code: 0
```

Fig 4. Image of Output

This method can also be used for applications to detect the occurrence of large numbers of keywords in a text character unit. Improved time can also be used to improve system efficiency. In addition, in order to avoid distortion, an error can be detected. Therefore, text analysis and

classification techniques are used to reduce error rate and improve performance of the proposed system.

References

[1] Robert L. Constable 1980 “The Role of Finite Automata in the Development of Modern Computing Theory” Computer Science Department Cornell University, Ithaca, NY U.S.A

[2] Nirmala J, Rajathi V 1980 ”SURVEY ON FINITE AUTOMATA CONSTRUCTION ” The International journal of analytical and experimental modal analysis

[3] Singh R, Ghorbani A, Swathi Y, Sundareswarar P 12-14 June 2017 “EfficientPMM: Finite Automata Based Efficient Pattern Matching Machine ”,International Conference on Computational Science, ICCS

[4] Pettersson M, 1992 “A Term Pattern-Match Compiler Inspired by Finite Automata Theory ” Department of Computer Science, Linköping University, Sweden

[5] Robert L 2019 “The Role of Finite Automata in the Development of Modern Computing Theory” Moore Schoolof Electrical Engineering University of Pennsylvania Philadelphia, Pennsyfvanio

[6] Kishore K, Jeberson R ,2019 “An Efficient Regular Expression Pattern Matching Using Stride Finite Automata” International Journal of Engineering and Advanced Technology (IJEAT)

[7] Melichar B 1995 “Approximate String Matching by Finite Automata ” Department of Computer Science and Engineering, Faculty of Electrical Engineering, Czech Technical University

[8] Kuldeep V, Pokarne R, Phaldesai M, Bhuruk T, Patil T, and Kumar P 2022 "SIMULATION OF CONWAY’S GAME OF LIFE USING CELLULAR AUTOMATA." International Research Journal of Engineering and Technology (IRJET) 9, no. 01 (2022): 2395-0056

[9] Kuldeep V, Mandhana M, Paralkar K, Pawal D, Deshpande S, and Sonkusale V 2022 "Pattern Matching in File System." International Journal of Computer Applications 975: 8887

[10] Kuldeep V, Bhavar N, Chauhan S, Kulkarni S, Thorat A, and Annapure Y 2022 “Spell Checker Model for String Comparison in Automata” EasyChair No. 7375.

[11] Kuldeep V. 2022 "Simulating Derivations of Context-Free Grammar"

[12] Kuldeep V, Bhavar N, Chauhan S, Kulkarni S, Thorat A, and Annapure Y 2022 Spell Checker Model for String Comparison in Automata EasyChair No. 7375