Global Journal of Pure and Applied Mathematics.
ISSN 0973-1768 Volume 12, Number 2 (2016), pp. 1397-1402

http://www.ripublication.com

Analysis on Various Search Algorithms

L. Sharmila*[1] and Dr. U. Sakthi[2]

[1]Research Scholar, Faculty of CSE, Department of Computer Science and Engineering, Sathyabama University, Chennai, Tamil Nadu, India.*

[2] Associate Professor, Department of Computer Science and Engineering, St. Joseph's College of Engineering, OMR, Semmencherry, Chennai, (TN), India.

Abstract

Data mining is the process of collecting, searching and analyzing a large amount of data in a database, as to discover pattern. Data mining extract data from data set and convert into understandable format. Matching different events with patterns, dependency relationships are not discriminative to find right matching. Finding optimal mapping can maximize matching score with respect to a pattern. Pattern matching is the act of checking a given sequence of tokens for the presence of the constituents of some pattern. Matching recognize pattern for enhancing the similarity between the events. Multiple events internally stores and maintain each and every characteristics of the event. The existing system pay-as-you-go style matching technique has poor discriminative feature and the technique depend mainly on time. So it will affect 1: n matching for events and when matching events with heterogeneous data it may result in duplication of data. A Generic Pattern matching technique implies to validate similarity between two events. Number of events internally stores and maintains all the characteristics of events. Patterns are compared with all stored patterns within the data set. Pattern matching enhances several events to improve the performance and efficiency of events. This is independent of time, so 1 : n issue can be sorted out. It will also retrieve exact result without any approximate value.

Keywords Pattern, Generic Pattern Matching, Multiple events, Generic Pattern Matching.

Introduction

Data mining is the process of extracting data from larger data sets and convert into the understandable format for further use [1]. Data mining is the analysis step of

"Knowledge discovery in databases" KDD process. The computational process of discovering pattern [2] is based on regular expression, strings and finite automata from large data sets. Data mining automatically search the given data from huge data set and discovering the patterns using analysis technique and also finding hidden information from datasets[3][4]. The fields combines tools from statistics and artificial intelligent with database management to analyze large data sets collection. Data mining is the extraction of useful patterns from data sources, e. g., databases, texts, web, image.

The goal of data mining concepts is extracting related information corresponding to given input patterns [7]. It is also referred to as extraction of implicit, previously unknown and potentially useful information of data. The exploration and analysis by semi-automatic means is done which is, of large quantities of data in order to discover meaningful patterns [5]. Data mining uses information from past data to analyze the outcome of a particular problem or situation that may arise. Data mining interprets its data into real time analysis that can be used in business and organizations [8]. Data mining consists of three stages. They are initial exploration, model building, evolution and deployment. Data mining can be a cause for concern when only selected information, which is not representative of the overall sample group, is used to prove a certain criteria [6].

Data sets are the collection of data which corresponds to the content of a single statistical data matrix or a single database table, a single statistical data matrix is a matrix of data of dimension n-by-p, where n is the number of samples observed, and p is the number of variables measured in all samples[9]. A single database table is a set of data elements using a model of vertical columns and horizontal rows. Pattern matching is the act of checking a given sequence of patterns for the presence of the constituents of some pattern. Patterns are rules for transforming input data. It is used in functional programming [11][12]. Matching checks the similarity between two patterns based on score function analysis. Pattern matching include outputting the locations of a pattern within a token sequence to output some component of the matched pattern, and a substitute the matching pattern with some other token sequence[10]. Regular Expression (RE) is a sequence of characters that forms a search pattern, mainly focus in pattern matching with strings. It is otherwise called as rational expression[10]. RE is a codified method of describing search patterns. Regular expression is a object which describes a pattern of characters. RE is used to perform "pattern matching" and "search and replace". Parsing text or string data files into sections are done, into a database Replacing values in text to clean, reformat, or change content RE modifiers which helps to perform case-insensitive matching, a global matching between patterns, perform multiline matching with multiple events[6][7]. RE is a compact way of describing sets of strings which conform to a pattern, which analysis the sequence of patterns from the already stored database [11]. The given pattern is compared with already stored text in database based on score function analysis. Text search is a technique used for searching a single pattern stored in a document or a collection of document in a single database. File renaming function which modifies name of file [12]. Database queries provide the platform for retrieving data from database. Web Directives specify settings that are used by the

user-control compilers when the compilers process web forms pages and user control files [2][10].

Background Study

Boyer-Moore (BM)

This Boyer-Moore algorithm initializes the input string and individual character is searched in the text. Its effective persevere crossways on a number of sources [5]. The Boyer-Moore algorithm uses information's that are collected during the initializing method to avoid unnecessary sections of the text. Commonly, the algorithm runs quickly as the pattern length increases its performance [10]. The Boyer-Moore algorithm considers a pattern A against text B, a mismatch of text character B [i] = p with the corresponding pattern A [j] is handled as follows: If p is not contained anywhere in A, then shift the pattern A completely. Otherwise, shift A until an occurrence of characteristic A gets aligned with B [i] [13].

DFA and NFA Automaton

Choosing the interchange between DFAs and NFAs, number of systems makes to use NFAs or extensions of NFAs. These systems tend to have an expressive pattern language where negations, Kleen closures, and temporal constraints are included. They are more emotional than regular expressions [8]. These systems are impressive in the direction of quick processing over sequential event streams, where an event is hard, and contains excessive attributes. This drives on a simpler trouble where events do not have excessive attributes, and this allows us to generate simpler algorithms i. e, in accordance with the reputation of events which is supported by initially finding all positive events and then pruning of the results that contain reputation of events in the fault temporal sequencing[4]. Added advantage of this algorithm is that it searches for contradiction in-place. Theoretical derivations must be sufficiently labelled as hypothetical. Hypothetical performance will be calculated in the identical way as real performance. Members must be able to explain in the basis for the theoretical results and the underlying theory that generated them enables a single state of execution [12]. Some transitions depend upon the bit values.

Rabin –Karp (RK)

The Rabin-Karp string searching algorithm calculates the hash value of the input, for each character subsequence of pattern to be compared with given input. Hash value is used to identify match score between the patterns. If the match score are equal, the algorithm will compare the pattern and repeated sequence of input [6]. In this way, there is only one alteration is mode per single text subsequence, and character matching is required by any pre-processing of the pattern. For the pattern that is identical we need some additional possibilities in sequence, let's take a method once again to make among the common characters of the pattern with the concurrent character of the input message [9]. RK algorithm requires getting quicker testing of the idealness frequent model that converts each and every string into a numeric value known as the hash value. i. e, we might have a hash ("AT") =2. RK uses the character

equivalent. Moreover, there are two agendas dissimilar or similar have a longer time for larger substrings.

Given a input string ‘b’, the case of string matching transaction with detecting whether a pattern ‘a’ occurs in ‘b’ and if ‘a’ does occur then recurring way in ‘b’ where ‘a’ occurs. One of the most clear approach is to select ‘a’ with the first element of the string ‘b’ in which to locate ‘a’. The comparison takes place between first elements of ‘a’ matches the first element of ‘b’, after compare the second element of ‘a’ with the second element of ‘b’ [3]. If match found proves similarity ‘a’ is detected first to shift ‘a’ one position to the right and replicate comparison which initiates from the starting element of ‘a’. This is a common randomized algorithm which is used to execute in linear time in repeated states of affairs of practical interest [2].

KMP Algorithm

The all-purpose plan at the rear KMP is bit difficult. Depending on this information and its present state the automaton goes to a future state, distinctively resolved by a set of interior regulations. One of the states is believed as "end”. Each time when we arrive this "end" state we have found a last position of a match. KMP is just an array of "pointers" that indicate the "interior rules" and individual "exterior" pointer to some appendix of that array which represents the "present state". The use of the regular expression (RE) is approximately equal to what we did in order to build the "failure function”. We take the other character from the text and try to "expand" the current partial match. If we fail, we go to the subsequent best partial match of the current partial match almost immediately. Generic Pattern matching follows the way of searching the string say “XYZ” is given it searches first character X the Y and as follows. On another case when two strings are given, then the string with more number of characters will be displayed as a result. i. e., when two strings, XYZ ABCD is given the count of first string is 3 and the count of second string is 4, then the string with more count “ABCD” will be displayed as the result since it has the greater count value compared to the other string.

Advantage

- The advantage of this approach is the ideality of avoiding the duplication of values so that it will be time and storage efficient.
- It also has a competitive value of higher accuracy which gives it an added advantage comparing to other similar algorithms.
- Due to its closed approach this algorithm has a improved efficiency value.

Simulation Settings

Boyer-Moore algorithm is used for selected string patterns only, Average amount of commands executed against the pattern length. Number of times each commands is executed for a certain period of time which decreases the performance of the Rabin-Karp algorithm.

KMP algorithm is used to search the scanning existing character by character. This algorithm is used for best performance. In Fig. 1 metrics is based on the length of regular expression. If length of regular expression is minimum it will be easily computable, but if it is maximum then it will be little bit time consuming. Below graph explains the comparison between time and number of pattern of different tasks. Using other algorithm, when compared to the present algorithm it's very easy to find the pattern. It is easy to search the pattern and easy to maintain the search information. In generic pattern matching, time consumption metrics per task represents the ratio between numbers of patterns and individual tasks. Below graph shows time and accuracy level of different algorithms such as KMP, BM and RK.

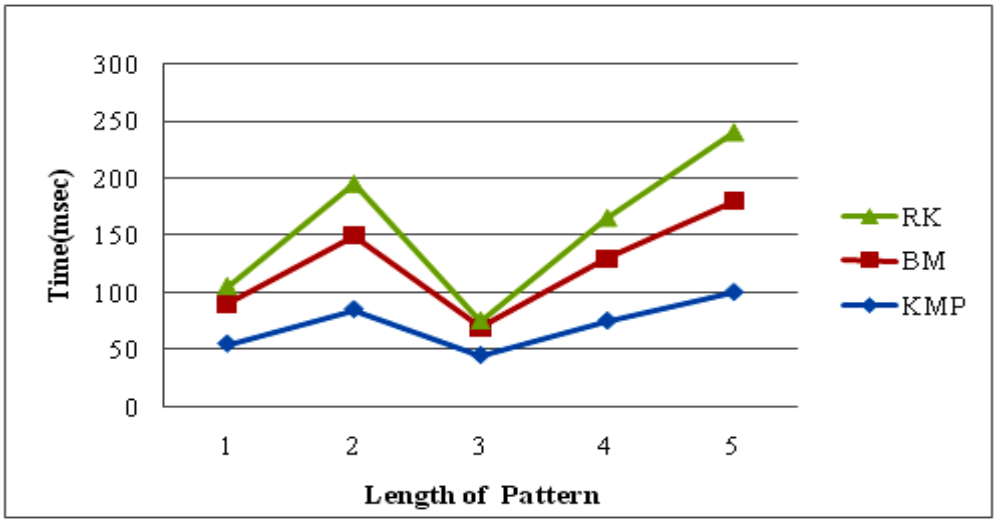

Figure 1: Comparison between Length and Time

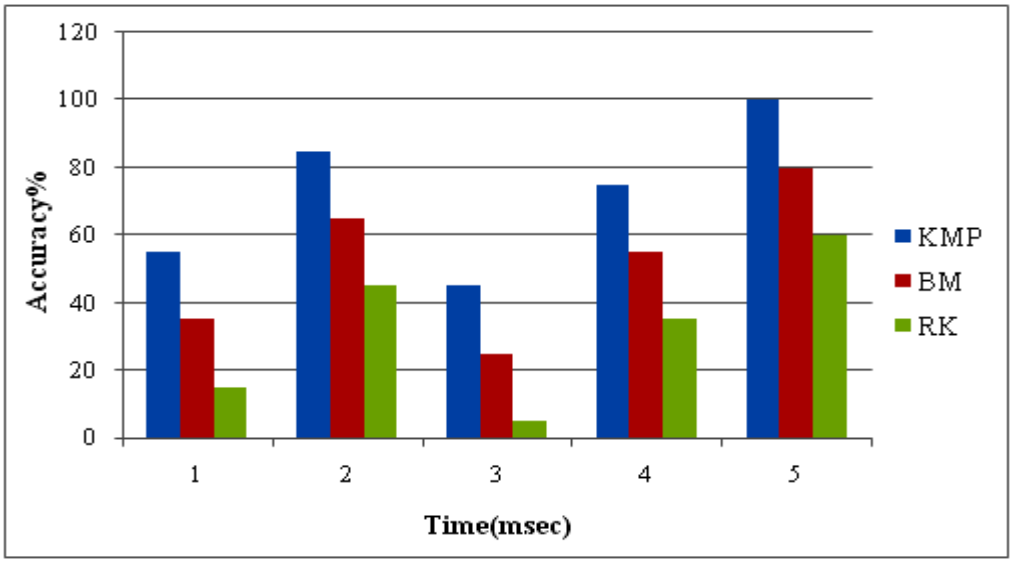

Figure 2: Comparison Between RK, BM and FA

Future Enhancement

In this paper pattern matching techniques are discussed with respect to the accuracy level of multiple events. Since, KMP and Quick sort algorithm is used to provide the exact result and value; it takes more time for generating the resulting event. Thus it is a Time complexity oriented system. As future work, it is interesting to consider the more complicated Time complexity for the events with different granularity in distinct processes. Moreover, we may exploit other attributes of events besides the accuracy and time adjustment.

References

[1] R. Agrawal and R. Srikant, "Mining sequential patterns", In P. S. Yuand A. L. P. Chen, editors, ICDE, pages 3–14. IEEE Computer Society, 1995.

[2] C. Bettini, X. S. Wang, S. Jajodia, and J.-L. Lin, "Discovering frequent event patterns with multiple granularities in time sequences", IEEE Trans. Knowl. Data Eng., 10(2):222–237, 1998.

[3] J. E. Cook and A. L, "Wolf. Event-base detection of concurrency", In SIGSOFT FSE, pages 35–45, 1998.

[4] L. Ding, S. Chen, E. A. Rundensteiner, J. Tatemura, W.-P. Hsiung, and K. S. Candan, "Runtime semantic query optimization for event stream processing", In ICDE, pages 676–685, 2008.

[5] X. Dong, A. Y. Halevy, and J. Madhavan, "Reference reconciliation in complex information spaces", In SIGMOD Conference, pages 85–96, 2005.

[6] L. Gravano, P. G. Ipeirotis, N. Koudas, and D. Srivastava, "Text joins in an rdbms for web data integration", In WWW, pages 90–101, 2003.

[7] G. Jeh and J. Widom, "Simrank: A measure of structural-context similarity", In KDD, pages 538–543, 2002.

[8] J. Kang and J. F. Naughton, "On schema matching with opaque column names and data values", In SIGMOD Conference, pages 205–216, 2003.

[9] D. Luckham, "The power of events: An introduction to complex event processing in distributed enterprise systems", In RuleML, page 3, 2008.

[10] J. Madhavan, P. A. Bernstein, and E. Rahm, "Generic schema matching with cupid", In VLDB, pages 49–58, 2001.

[11] S. Nejati, M. Sabetzadeh, M. Chechik, S. M. Easterbrook, and P. Zave, "Matching and merging of statecharts specifications", In ICSE, pages 54–64, 2007.

[12] T. Pedersen, S. Patwardhan, and J. Michelizzi, "Wordnet: Similarity-measuring the relatedness of concepts", In AAAI, pages 1024–1025, 2004.

[13] E. Rahm and P. A. Bernstein, "A survey of approaches to automatic schema matching", VLDB J., 10(4):334–350, 2001.