Research , paper about BFM: a forward backward string matching algorithm with improved shifting for information retrieval

A**ssignment** Abtract:

Extraction data and gain valuable information from given text is an important part of data mining . Nowadays,applying a string matching algorithm is very necessary As a result of the growing files' sharing over the Internet and the urgent need for a cloud computing .in this paper, we are gonna discuss a new algorithm named Back and Forth Matching algorithm (BFM),which decreases the number of characters comparisons .It makes an improvement than other algorithms that works faster in alignment process , by matching pattern from both forward and backward direction with text. So it simplifies searching of large string to match with pattern.

Introduction :

To find a required text (pattern)in a text decuoment is a simple task.but it may be take long time and huge file size. As a result, a fast pattern matching algorithm is an essential component of the page ranking in search engine, checking syntax ,spelling mistakes and many applications .it also important in bioinformatics, DNA sequences matching, and behavior analysis as well. To search through the text, the pattern is used as a window of length m that moves over the text T starting from it's leftmost character. The target of the algorithm is to match the characters of the window with the text characters in multiple attempts . This succession of attempts and window shifts continues, until the right end of the window reaches the length of the text. This type of window mechanism is known as the sliding window mechanism. The aim of these algorithms are to minimize the character comparisons and to maximize the length of shifts . we've presented a new string-matching algorithm BFM. More importantly, the algorithm does both forward and backward checking and decreases the number of efforts required to match the window with the text during the matching phase.

ASSIGNMENT 2: Related Work ;

The Boyer Moore algorithm is one of the most renowned, efficient and extensively used pattern matching algorithm. It preprocesses the pattern and, in the event of a mismatch, calculates the pattern's maximal changes using two heuristics(the good and bad heuristics). Boyer Moore chooses the better of the two heuristics, thus can slide the pattern by maximum number of characters.

It descripes another way to improve the algorithm by shifting. Another algorithm is called the Quick Search algorithm. it proposed a combination of Quick Search and the Skip Search algorithm. hybrid algorithm was proposed using the idea of Quick-Skip and Boyer–Moore algorithms.

The Knuth–Morris–Pratt (KMP) algorithm, which operates similarly to a naive algorithm, is another highly used algorithm.

instead of checking all the characters after each shift, the algorithm preprocesses P to build a table. Proposes a modern algorithm that combines the Boyer–Moore and KMP ideas.

Beside all these algorithms, a comperatively old string matching technique is Rabin Carp string searching algorithm that can search both single and multiple patterns in a string, that deploys a modified version of the Rabin Carp algorithm using a GPU processor.

ASSIGNMENT3:

The Back and Forth Matching (BFM) method was created with the goal of lowering the total amount of shifts while matching strings by focusing on greater pattern window shifts for each mismatch. A preprocessing table is created as the first phase of the algorithm, storing the points in the text where the first and last characters of the pattern match the text.

Take the same text and pattern we used to explain the KMP and BMH algorithms to show the process of BFM. where the first and last characters of the pattern are matched with the text. If a match is found, *posIndex* stores the position. BFM checks if the second character from the left and the right of the pattern is matched with the text or not . It counts it as a total match if all the characters between the first and last character of the pattern matches . We analysed both BMH and KMP algorithms along our algorithm BFM to compare our method's performance on both files.

The total number of comparisons and shifts required to find a pattern is used as a measure of an algorithm's quality, as it has been in many previous research. These two criteria are being used to evaluate our algorithm. The initial evaluation was performed on the “World192.txt” file in order to locate the pattern “Bangladesh.” This pattern may be found 9 times in the file. From these result it can be said that the number of shifts, comparison and execution time of our algorithm BFM are very much lower than BMH and KMP. This analysis also shows our proposed algorithm is performing better than the other algorithms. The Back and Forth Matching (BFM) is a text preprocessing technique that uses forward and backward matching to match both the first and last characters of the pattern. This preprocessing task improves searching by allowing the algorithm to search only on the indexed places, reducing the amount of character comparisons and shifting. One limitation of this algorithm is if the text and pattern both are small, then BFM’s performance degrades, as there is a preprocessing phase to complete before searching. Nevertheless, we found the performance of BFM in respect to execution time, number of shifts and comparisons to be exceptionally high in contrast to KMP and BMH. Hence, this algorithm would certainly an efficient choice in case of the tasks that require huge amount of text searches.