PROJECT PROPOSAL

Pattern Matching Algorithm in Genome Sequence

Submitted by:

- Jagruthi Patibandla, 180001021;
- Shravya Ramasahayam, 180001052.

Submitted to:

Dr Kapil Ahuja,

Associate Professor in CSE.

Introduction:

Unhealthy symptoms or a specific illness in the body is termed as a disease. The disease may be referred to as disabilities, disorders, syndromes, symptoms. Genes are the basic building blocks of heredity. They get passed from parent to child. They hold DNA, the instructions for making proteins. A genetic disease is any disease that is caused by an abnormality in an individual's genome. Some of the genetic disorders are inherited from the parents, while other genetic diseases are caused by mutations in a pre-existing gene or group of genes.

DNA sequences of various diseases are stored in databases for easy retrieval and comparison. If it is found that a particular sequence is a cause for a disease, then the trace of the sequence in the DNA and the number of occurrences of the sequence are searched to define the existence and intensity of the disease. As the DNA is a large database, molecular biologists are increasingly taking the help of computer science algorithms to find DNA patterns in DNA sequences.

Objective:

Complex pattern matching methods are used in locating DNA sequences, fingerprint assessment, soil patterns reporting, and retinal blood vessel assessment. A well-known application of sequence analysis is bioinformatics.

As the DNA is a large database, String and Pattern matching algorithms are used to find out a particular sequence in the given DNA. Pattern matching (PM), is a computerized search operation for finding a given pattern within a database. These algorithms are generally used to specify whether the desired structure is present in the given set of elements or not. This project aims to:

- ✓ To implement these algorithms.
- ✓ To analyse the existing pattern searching algorithms.

Approach:

The pattern-matching algorithm gives either an indication that the pattern does not exist in a text string or the starting index in the text string of a substring matching. There are various pattern-matching algorithms. Here we are to review three pattern matching algorithms. These efficient algorithms can be used to trace the sequence of DNA in a huge gene database.

Naïve String Matching (Brute force) Algorithm:

It is the simplest method. It checks for all character of the main string to the pattern. This algorithm is helpful for smaller texts. It does not need any pre-processing phases. The procedure can be interpreted graphically as sliding a "template" containing the pattern over the text, noting for which shifts all of the characters on the template equal the corresponding characters in the text. It finds all valid shifts using a loop that checks the condition P[1....m]=T[s+1...s+m] for each of the n-m+1 possible values of s.

KMP Algorithm:

The Knuth–Morris–Pratt string-searching algorithm (or KMP algorithm) searches for occurrences of a "word" W within a main "text string" S by employing the observation that when a mismatch occurs, the word itself embodies sufficient information to determine where the next match could begin, thus bypassing re-examination of previously matched characters. Somehow, we should be able to take advantage of this information instead of backing up the pointer over all those known characters

Boyer- Moore Algorithm:

The Boyer–Moore string-search algorithm is an efficient string searching algorithm that is the standard benchmark for practical string-search literature. It was developed by Robert S. Boyer and J Strother Moore in 1977. The key features of the algorithm are to match on the tail of the pattern rather than the head and to skip along with the text in jumps of multiple characters rather than searching every single character in the text.

References:

- https://pdfs.semanticscholar.org/2a08/3422a388c2773668a8808b33207b8cf37241.pdf
- http://ijsetr.com/uploads/625413IJSETR2868-162.pdf
- https://en.wikipedia.org/wiki/String-searching_algorithm