Performance Evaluation of Various DNA Pattern Matching Algorithms Using Different Genome Datasets

Article · November 2018

DOI: 10.5281/zenodo.2580651

3 authors:

Mohammad Riyaz Hossen
Bangladesh University of Engineering and Technology
2 PUBLICATIONS **1** CITATION
SEE PROFILE

Md. Shafiul Azam
Pabna University of Science and Technology
20 PUBLICATIONS **120** CITATIONS
SEE PROFILE

Humayan Kabir Rana
Green University of Bangladesh
20 PUBLICATIONS **142** CITATIONS
SEE PROFILE

Performance Evaluation of Various DNA Pattern Matching Algorithms Using Different Genome Datasets

Mohammad Riyaz Hossen[1], Md. Shafiul Azam[2*] and Humayan Kabir Rana[3]

[1,2]Department of Computer Science and Engineering, Pabna University of Science and Technology, Pabna-6600, Bangladesh

[3]Department of Computer Science and Engineering, Green University of Bangladesh, Dhaka -1207, Bangladesh

Email: shahincseru@gmail.com

***Abstract*—Pattern matching algorithm plays a vital role for searching and analyzing patterns in computational biology. The functional and structural relationship among different DNA databases is performed by different algorithms. The similarities among the biological sequences determined by matching algorithms, is a major research area. This paper surveys the performance of Naïve exact matching, Boyer-Moore (BM) and Knuth-Morris-Pratt (KMP) algorithms. The main process of this study is to find the matched DNA sequence from the database using these pattern matching algorithms. This study determines more efficient pattern matching algorithm by analyzing the matching result. Among those three algorithms, Boyer Moore algorithm is faster and efficient for matching large DNA sequence. It has no needless comparison. It works sub linearly on best case. Comparison result also demonstrates that the matching values of natural genomes are higher than virus genomes.**

Keywords: Boyer-Moore, DNA Pattern, DNA Sequence Analysis, Genomes Matching, KMP

I. Introduction

Pattern matching is a vital component of bioinformatics mostly used in the application of computer technology to manage and analyze of biological data (Mukherjee, Dutta, & Chowdhury, 2017; Tun & Swe 2014). DNA pattern matching is a fundamental and upcoming area in computational molecular biology. It is a task of finding subsequences within a long DNA sequence. The problem in pattern discovery is to determine how often a candidate pattern occurs, as well as possibly some information on its frequency distribution across the sequence. In general, a pattern will be a description of a set of strings, each string being a sequence of symbols (Rajesh, Prathima, & Reddy, 2010). We use different pre-procedure and algorithms for matching this pattern of DNA sequence. Many scientist and researcher work uninterrupted for this algorithm evolution. Our analysis is an effort to explore multiple DNA pattern matching algorithms with different genome. We have used working environment Jupiter notebook with anaconda distribution and all pseudo-code implemented with python programming language. The pattern matching can be described as, given a specific sequence of DNA string P generally called pattern searching in large genome sequence T to locate P in T. In this research, a survey is made on the performance of three pattern matching algorithm namely Naïve, BM and KMP with three different type genome database Human hair, Leukemia and HIV. The decision from the experimental result will help to detect any DNA sequence with efficient procedure from long length genome and also be used in future for better bio-informatics research.

II. Review Of Literature

A. Related Works

Different researchers have worked on DNA pattern matching problem in previous. Mukherjee et al. (2017) implemented three pattern matching algorithm for analyzing DNA pattern matching. This paper surveys the performance of Brute-force, Byer-Moor and KMP string matching algorithm to find out a particular pattern in the given DNA sequence more efficiently. Nyo. Me. et al. demonstrated comparison of three pattern matching algorithms. These algorithms are effectively used in matching DNA sequences because of DNA database is very complex and huge and not to retrieve easily (Tun & Swe 2014). Rajesh et al. (2010) was shown that unusual pattern detection in DNA database using KMP algorithm. They use Knuth-Morris-Pratt ("KMP") algorithm avoids this waste of information through matching DNA sequences. Nusaiba Islamet al. showed faster and efficient algorithm for sequence alignment using local and global alignment. Tamal Chakrabartiet al. introduced a parallel dynamic programming approach, to reduce the time of the DNA sequence alignment process. Saman, Rahman, Ahmad, & Tap (2006, May) compares many search process and they provide a minimum cost process in searching for a set of similar DNA sequences. The paper demonstrates the employ of exact string matching. Pavesi, Mauri, & Pesole (2001) have provided a survey of different algorithms and methods for the automatic discovery of patterns in biological sequences. Pandiselvam, Marimuthu, & Lawrance (2014) were studied various kinds of string-matching algorithms with biological sequences such as DNA and Proteins. This paper analyzed that KMP algorithm relatively easier to implement, it requires extra space, Rabin Karp algorithm used to detect the plagiarism, and it requires additional space for matching. The Boyer Moore algorithm is extremely fast for on large sequences,

**Author for Correspondence: shahincseru@gmail.com*

and its best-case running complexity is sub linear (Pandiselvam, Marimuthu, & Lawrance (2014).

B. DNA Pattern Review

Patterns are nonrandom entities that represent a phenomenon within a set of sequences (Adey, 2013). DNA also called deoxyribonucleic acid mostly located in human cell nucleus is the hereditary material in humans and almost all other organisms. Every people of universe have different DNA and almost all of them are identical. We can represent any human cell DNA with sequence of patterns this pattern consists of four nucleic acid. DNA patterns are graphs of DNA sequences. Most DNA molecules consist of two biopolymer strands coiled around each other to form a double helix. DNA contains three components: deoxyribose (a five-carbon sugar), a series of phosphate groups, and four nitrogenous bases, (nitrogen compounds that contain bases). The four bases in DNA are adenine (A), thymine (T), guanine (G), and cytosine (C) (Islam, 2012). DNA sequence analysis is a technology developed by researcher to determine the order of base in DNA. A DNA sequence is our representation of a string of nucleotides contained in a strand of DNA. Example: ATGCGATACAAGTTGTGA represents a string of the nucleotides A, G, C, and T (Islam, 2012). The uses of DNA pattern are fingerprinting or DNA profiling, disease analysis, solving biological problems and bio-informatics challenges.

C. Pattern Matching Algorithms

Pattern matching algorithms performed important task of the pattern discovery process in today's world for finding the structural and functional behavior in proteins and genomes. DNA sequence have a string pattern. String is a finite sequence of characters drawn from alphabet Σ where usually, Σ = {A, C, G, and T}. There have several pattern matching algorithms some of them are shown in table 1.

Table I: Different Algorithms for Pattern Matching.

SL.	Algorithm Name
01	Naïve Exact matching algorithm
02	Brute Force Pattern Matching
03	Boyer–Moore string search algorithm
04	Knuth–Morris–Pratt algorithm
05	Two-way string-matching algorithm
06	Rabin–Karp string search algorithm
07	Smith-Waterman algorithm
08	Needlemanwunsch algorithm
09	Hamming Distance
10	Levenshtein Distance
11	Aho-Corasick algorithm
12	CommentZ- Walter algorithm

III. RESEARCH METHODS

This experiment performed by three different experiments. We have matched three databases one by one with the help of three algorithms. We have got nine different results for three datasets by employing mentioned three algorithms. By analyzing these results, we have measured the performance of different DNA pattern matching algorithms. To compute efficiency of algorithms using different database three main steps are used,

i) Collect input genomes as FASTA file.
ii) Match the pattern by algorithms.
iii) Analyze the algorithms with their characteristics.

The pictorial representation of methodology is shown in fig. 1.

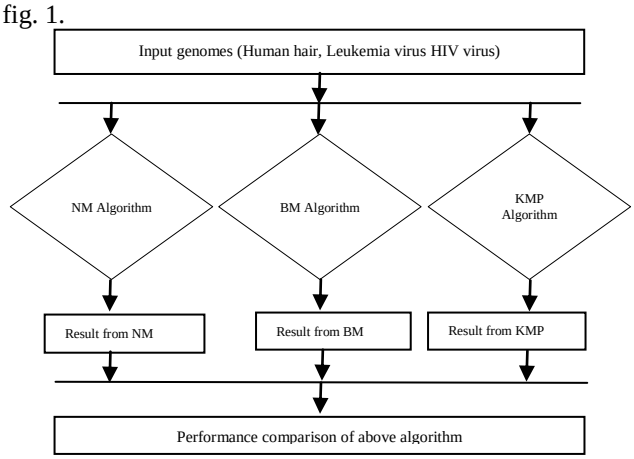

Fig 1: Flow diagram for experiment methodology.

A. Analyzed Algorithms

We have selected three mostly used algorithms for our experiments that are given below:

1) Naive exact matching algorithm (NM)

Exact matching is concerned with finding all the places where a short string occurs in a longer string. The shorter string is called the pattern and we refer to it as P. The longer string is called the text and we refer to it as T. An algorithm is said to be naive when it is simple and straightforward, but does not exhibit a desirable level of efficiency despite finding a correct solution. It does not find an optimal solution to an optimization problem, and better algorithms can be designed and implemented with more careful thought and clever techniques.

2) Boyer–Moore algorithm (BM)

Boyer–Moore was developed by Boyer and Moore (1977) it is an efficient string searching algorithm that is the standard benchmark for practical string search literature (Coursera, 2017). The algorithm scans the characters of the pattern from right to left beginning with the rightmost one. This characteristic allows the algorithm to skip more characters than the other algorithms. In case of a mismatch, it uses two pre-computed functions to shift the window to the right. These two shift functions are called the good-suffix shift and the bad-character shift.

3) Knuth–Morris–Pratt algorithm (KMP)

Knuth–Morris–Pratt string searching algorithm searches for occurrences of a "pattern" P within a main "text string" T 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. The algorithm uses failure function for preprocessing the pattern string P. There is a match, increase the current indices. If not, consult the failure

function the new index in P here needs to continue checking P against T after that its works for whole genome.

Table II: Pseudo-code for analyzed algorithms

Naive exact matching (NM)	Boyer–Moore (BM)	Knuth–Morris–Pratt (KMP)

Function naiveExact(P,T)
occurrences ← 0
for i ←
range(length(t) -
length(p) + 1)
match = True
for j ← range
(length (p)) # loop
over characters
If t[i+j]!=p[j]
match= False
break
If match
occurrence match to i
# all chars matched; 	
record
return occurrences
	BoyerMoore
	(P,T)
	i← m-1
	j←m-1
	repeat
	if
	p[j]=t[i] then
	if
	j=0 then
	return i
	else
	i←i-1
	j←j-1
	else
	i←i+m
	min(j,1+last[t[i]])
	j←m-1
	until i
	>n-1
	return "no
	match"
		KMP_Match(P,T)
		f← failure
		function(p)
		i← 0
		j← 0
		while i <n
		if t[i] =p[j]
		if j=m-1
		return i-j
		else
		i←
		i+1,j← j+1
		else
		if j>0
		j←f[j-1]
		else
		i←
		i+1 [2]

Table III: Comparison of three algorithms

Algorithm name	Preprocessing time	Matching time	Space complexity
Naïve exact matching algorithm	None	Θ(nm)	None
Boyer–Moore algorithm	Θ(m + k)	bestΩ(n/ m),worst O(mn)	Θ(k)
Knuth–Morris–Pratt (KMP)	Θ(m)	Θ(n)u	Θ(n+ m)

B. Database Descriptions

In this study, we have used three datasets include Human hair, Leukemia virus and HIV virus, the short description of these datasets are given in table 4.

Table IV: Used genomes with short description.

Human hair	Leukemia Virus	HIV virus
As hair samples we use Keratin, type II cuticular Hb6 is a protein that in humans is encoded by the KRT86 gene the protein encoded by this gene is a member of the keratin gene family.	Leukemia, is a group of cancers that usually begin in the bone marrow and result in high numbers of abnormal white blood cells. The leukemia virus details is, Kingdom: Viruses; Subgroup: Retrovirida	HIV stands for human immune deficiency virus. It harms your immune system by destroying the white blood cells that fight infection. This puts you at risk for serious infections and certain cancers.

IV. EMPIRICAL RESULT

Genome files are tested by different pattern size with the help of different algorithms. The results are shown in table 5, 6 and figure 2, 3.

Table V: Human Hair genome experimental result

SL.	Genome Name (Keratin, type II hair)	Length	Match % for Read length six(ATGCAT)	Match % for Read length Five(ATGCA)	Match % for Read length Four(ATGC)	Naïve	BM	KMP
1	KRTHB1	350	1.72%	4.28%	8%	best		best
2	KRTHB2	3710	.65%	2.29%	6.03%		best	
3	KRTHB3	2380	.76%	1.68%	4.20%		best	
4	KRTHB4	2450	.49%	2.85%	6.36%		best	
5	KRTHB5	2501	1.91%	4.99%	6.87%		best	
6	KRTHB6	350	1.72%	1.43%	1.14%			best
7	psihHbA	2919	1.02%	4.64%	9.18%		best	
8	psihHbC	2660	.902%	2.44%	6.76%		best	
9	psihHbD	2450	.734%	1.63%	3.92%		best	

Table VI: Leukemia and HIV virus genome experimental result.

SL.	Genome Name leukemia virus &HIV	Length	Match % for Read length six (ATGCAT)	Match % for Read length Five (ATGCA)	Match % for Read length Four (ATGC)	Algorithms decision by time and space complexity		
						Naïve	BM	KMP
1	Friend murine	2100	0%	.71%	2.9%			best
2	Moloney murine	2100	.29%	.98%	2.7%		best	
3	unclassified Murine	2025	0%	.98%	3.0%			best
4	Bovine	2119	0%	.47%	6.0%			best
5	Feline	2148	.56%	2.1%	3.9%		best	
6	Gibbon ape	2030	.30%	.5%	2.2%		best	
7	Abelson m	1484	0%	.34%	3.5%			Best
8	HIV 1	2310	1.6%	2.6%	3.9%		best	
9	HIV 2	2590	.69%	2.9%	4.5%		best	

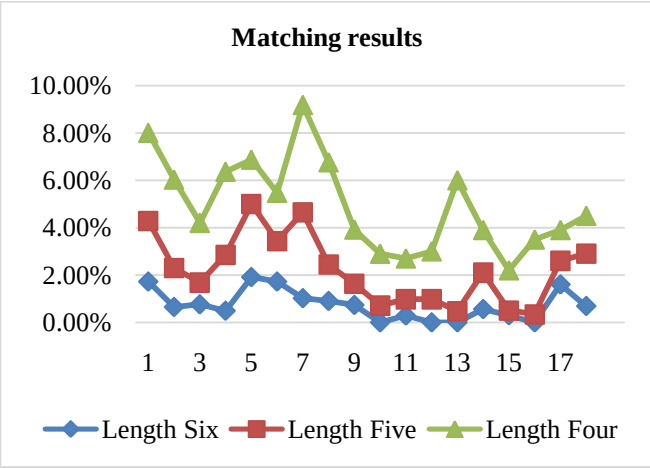

Fig 2: Matching result (%) for different genome datasets.

Fig. 2 shows that when we test the genomes with different size pattern then matching result for short length genome is higher than long length genome. The Performance of three algorithms is shown in fig. 3.

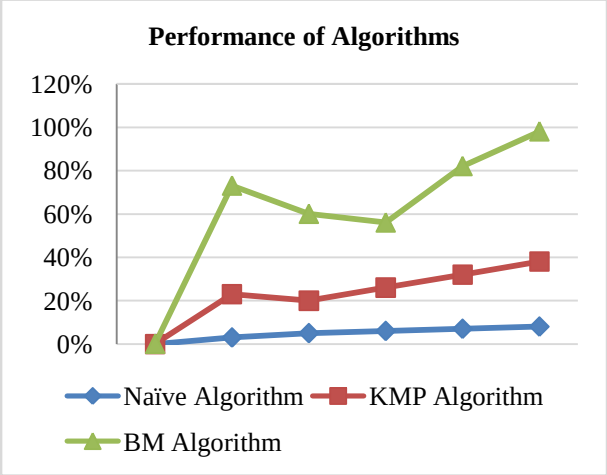

Fig 3: Performance measurement of Algorithms.

A. Performance Evaluation of Algorithm:

From our analysis we have recorded different characteristics, analyzed them and seen that for matching large DNA sequence the faster and efficient algorithm is Boyer Moore algorithm. It has no needless comparison and from complexity analysis we have noticed that its work sub linearly on best case. Pandiselvam et al. (2014) proved it previous. Knuth-Morris-Pratt (KMP) never moves to backwards. By implementation procedure and complexity analysis, we can tell KMP is a simpler algorithm than others. Naïve exact matching is a relatively ancient procedure don't require pre-processing and slow for matching DNA patterns. The Boyer-Moore algorithm is more efficient DNA pattern matching algorithm because it has the characteristics shown in the table 7.

Table VII: Characteristics of BM algorithm

SL.	Attribute	Value/Perception
1	Matching comparisons of BM algorithm	Matches the pattern right to left.
2	It gives the best performance in best case	Complexity O(n/m)
3	Time & space complexity in preprocessing phase	O(m+n)
4	Searching phase complexity	O(mn)
5	BM algorithm works for 3n text character comparisons in the worst case	It searches non-periodic pattern first.

B. Comparison of Matching Result

Analyzing the matching values of genomes, it proves that the normal hair database is more identical than virus database. From calculation the hair is 60% identical, HIV 30% and leukemia virus is only 10%. So it's one of the important decision from our experiment that normal database is more identical then virus database. The comparison shows that for pattern length six the matching value is smaller than length five and four. So, an important implication is that, in long pattern, the matching value is small but exact, in short pattern, the matching value is large but not exact.

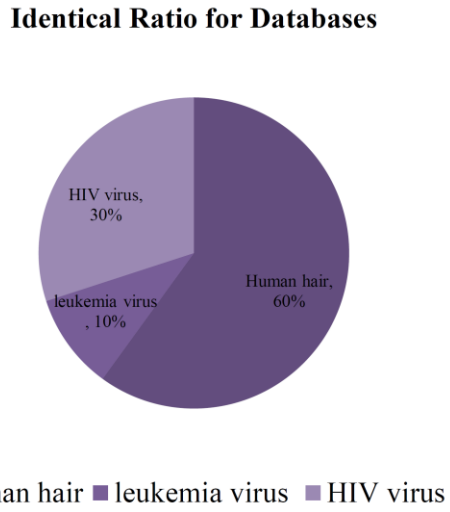

Fig 4: Identical ratio for databases.

After that, the empirical result says Boyer-Moore algorithm is more efficient DNA pattern matching algorithm compared to other's and matching values of natural hair genomes is higher than virus genomes.

V. Discussions And Implications

In this paper various kinds of pattern matching algorithms were analyzed with different genome datasets. Pattern matching is computerized matching from large kind of biological sequences called genomes. DNA databases are huge and consists of large patterns so we need to use pattern matching algorithms. DNA genomes are consists of four nucleobases Adenine, Thiamin, Guanine and C, osine shortly A, T,G, C. We have analyzed three most important DNA pattern matching algorithm. We perform matching process with these databases for analyzing our implemented algorithms. As a result, we get analysis results for both algorithms and databases. From our analysis we have seen that for matching large DNA sequence the faster and efficient algorithm is Boyer Moore algorithm. It has no needless comparison and from complexity analysis, we have noticed that its work sub linearly on best case.

Knuth-Morris-Pratt (KMP) never moves to backwards. By implementation procedure and complexity analysis, we can tell KMP is a simpler algorithm then others. Naïve exact matching is a relatively ancient procedure don't require pre-processing and slow for matching DNA patterns. Our used databases are two types one is natural genome (human hair) another's virus genome (HIV and leukemia). Their CG% is an average 40% to 50% in range. We have used short and long reads DNA pattern. It's proving that for long reads the matching value is exact and accurate. Relatively for short reads the matching values are approximate and cannot identify any specific genome. Matching values for natural genomes are high enough than virus genomes.

VI. Conclusion

This study introduces one of the best pattern-matching algorithms using DNA sequences. This paper gives the most efficient method for determining the DNA pattern matching sequences. We have taken a FASTA file of natural genome and tested randomly by different pattern size. We have analyzed the characteristics of different algorithms. The algorithm mentioned/worked here gives better performance compared with some of the other popular algorithms in case of a number of comparisons and scientific ratio. Based on the experimental work, Boyer-Moore algorithm gives good performance compared to others.

Acknowledgement

First and foremost, praises and thanks to the Allah, the Almighty, for His showers of blessings throughout our research work to complete the research successfully. We would like to show our gratitude to the faculties from the department of Computer Science and Engineering, Pabna University of Science and Technology who provided insight and expertise that greatly assisted the work. We are extending our thanks to the management of Computer Science and Engineering, Pabna University of Science and Technology for their support to do this work. Finally, our thanks go to all the people who have supported me to complete the research work directly or indirectly.

References

Adey, S. P. (2013). GPU Accelerated Pattern Matching Algorithm for DNA Sequences to Detect Cancer using CUDA Dissertation. *College of Engineering, Pune*.

Boyer, R. S., & Moore, J. S. (1977). A fast string searching algorithm. *Communications of the ACM*, *20*(10), 762-772.

Coursera (2017). *Algorithms for DNA Sequencing*. Assessed from: https://www.coursera.org/learn/dna-sequencing/home/welcome, Johns Hopkins University.

Mukherjee A., Dutta G., Chowdhury C. R. (2017). DNA Pattern Matching A Comprehensive Study of Three Pattern Matching Algorithms. *International Journal of Computer Application. XI*(XI).

Pandiselvam, P., Marimuthu, T., & Lawrance, R. (2014). A Comparative Study on String Matching Algorithms of Biological Sequences. In *International Conference on Intelligent Computing* (pp. 1-5).

Pavesi, G., Mauri, G., & Pesole, G. (2001). Methods for pattern discovery in unaligned biological sequences. *Briefings in Bioinformatics*, *2*(4), 417.

Rajesh, S., Prathima, S., & Reddy, L. S. S. (2010). Unusual pattern detection in dna database using kmp algorithm. *International Journal of Computer Applications*, *1*(22).

Saman, M. Y. M., Rahman, M. N. A., Ahmad, A., & Tap, A. O. M. (2006, May). A minimum cost process in searching for a set of similar DNA sequences. In *5th WSEAS International Conference on Telecommunications and Informatics* (pp. 348-353).

Tun, N., & Swe, T. M. M. (2014). Comparison of Three Pattern Matching Algorithms using DNA Sequences. *International Journal of Science, Engineering and Technology Research*, *3*(35)