International Journal of Bioinformatics Research
ISSN: 0975-3087 & E-ISSN: 0975-9115, Volume 4, Issue 2, 2012, pp.-269-272.
Available online at http://www.bioinfo.in/contents.php?id=21

A FAST LOCAL SEARCH ALGORITHM USING HISTOGRAM FEATURES FOR DNA SEQUENCE DATABASE

QIU CHEN[1*], KOJI KOTANI[2], FEIFEI LEE[1] AND TADAHIRO OHMI[1]

[1]New Industry Creation Hatchery Center, Tohoku University, Japan.
[2]Department of Electronics, Graduate School of Engineering, Tohoku University, Japan.
*Corresponding Author: Email- qiu@fff.niche.tohoku.ac.jp

Received: May 06, 2012; Accepted: May 21, 2012

Abstract- DNA sequence search is a very important topic in bioinformatics algorithm development. However, this task usually spends much computational time to search on large DNA sequence database. In this paper, we propose an efficient hierarchical DNA sequence search algorithm to improve the search speed while the accuracy is being kept constant. For a given query DNA sequence, firstly, a fast local search algorithm using histogram features is used as a filtering mechanism before scanning the sequences in the database. An overlapping processing is newly added to improve the robustness of the algorithm. A large number of DNA sequences with low similarity will be excluded for latter searching. The Smith-Waterman algorithm is then applied to each remainder sequences. Experimental results using GenBank sequence data show the proposed algorithm combining histogram information and Smith-Waterman algorithm is more efficient for DNA sequence search.

Key words- Fast search, DNA sequence, Histogram feature, Smith-Waterman algorithm, Local search.

Citation: Qiu Chen, et al. (2012) A Fast Local Search Algorithm Using Histogram Features for DNA Sequence Database. International Journal of Bioinformatics Research, ISSN: 0975-3087 & E-ISSN: 0975-9115, Volume 4, Issue 2, pp.-269-272.

Introduction

The decipherment of 3-billion-base human genome sequence which was called Apollo project of life sciences [1, 2] was finally completed by the international cooperation in April 2003. Since this achievement of human genome project, researchers around the world are now having a very keen competition on clarification of the structure and performance analysis of the protein, genes and protein networks and new gene sequences are clarified every day. The enormous quantity of data has been accumulated in the database like GenBank [9], EMBL [3] and DDBJ [4], etc. Moreover, the volume of data of Genome Database still increases in exponential [10].

Comparison of genome sequences (DNA, mRNA and protein) is the most important task in the life science area. There are 4 types of the DNA nucleotides, namely, A (adenine), C (cytosine), G (guanine) and T (thymine), which are utilized to encode DNA. If gene A and gene B have high homology, it is surmisable that the function of gene A is similar to that of gene B.

Normally, when a new DNA or protein sequence is determined, it would be compared to all known sequences in the annotated databases such as GenBank, EMBL and DDBJ, etc. Because the database is very large, a lot of algorithms are studied and used for the speeding-up of data search. Needleman and Wunsch presented the Needleman-Wunsch algorithm [5], which calculates similarities between sequences by the dynamic programming and Smith-Waterman algorithm is the improved approach [6].

However, it takes much time to retrieve data with these algorithms because they require too many amounts of calculation. Blast [7], FASTA [8] and PatternHunter [11, 12] are three rapid heuristic algorithms are regularly used for searching protein and DNA sequence databases. The idea in these tools is to find subsequences that share some patterns called as filtration techniques. While BLAST and FASTA have improved the retrieving speed with heuristic algorithms, there is a possibility of missing an alignment or giving inaccurate output. Thus, many researches have been trying to improve both the search time and the precision.

We have proposed an efficient algorithm combining histogram features and Smith-Waterman dynamic programming algorithms [6] in order to improve both speed and precision [13]. Histogram features of sequences are firstly used to compare the query sequence with the sequences in database and similarity scores would be obtained. Only the sequences whose similarities exceeded a given threshold are then aligned using exhaustive Smith-Waterman dynamic programming algorithm. The effects have been demonstrated by using GenBank sequence data, which is the NIH genetic sequence database, a collection of all publicly available DNA sequences. For sequences which range of length variation is not very large, the experimental results show the proposed algorithm is very efficient, but the efficiency decreases with variation in sequence length.

In this paper, we propose a local search algorithm in order to improve both efficiency and speed even the sequence length changes largely. An overlapping processing is newly added to improve the robustness. The effects will be demonstrated by using GenBank sequence data.

This paper is organized as follows. At first, we will first introduce the proposed local search algorithm using histogram features for DNA sequences in detail. Experimental results using publicly available GenBank sequence data will be discussed. Finally, conclusions are given.

Proposed algorithm

Searching and comparing a query sequence with the databases with large size of sequences is complicated and requires for more time and spaces complexity when using classical Smith-Waterman algorithm [6] to align two sequences. Therefore, the need of mechanism to discard the unrelated or irrelevant sequences compared to a query is highly demanded. In this paper, we present a new search method for DNA sequence matching in a large size of DNA sequence databases. Histogram features of sequences are firstly used to compare the query sequence with the sequences in database and similarity scores would be obtained. Only the sequences whose similarities exceeded a given threshold are then aligned using exhaustive Smith-Waterman dynamic programming algorithm [6].

The processing steps of our proposed algorithm is shown as Figure 1. When an unknown query base sequence is input, it will be divided into short parts. As shown in figure 3, we newly utilize an overlapping method to divide the sequence into partial sequences. The overlapping step will be discussed in the experimental sectional. It is thought that more robust features can be extracted if order information of the base sequence is added. For each separate partial sequence, it will be divided into small sequence, for instance, ACT and CGG, etc. A small sequence can be considered as a three dimensional vector. This processing overlaps over all the sequence. After that, the histogram feature is calculated. There are only 4 types of DNA bases, so the number of combination of 3-dimensional vector is 64. A reference table with the size of 64 is shown in Figure 2, by which the index number of the 3-dimensional vector is very easy and fast to be determined. The number of vectors with same index number in each separate partial sequence is counted and feature vector histogram is easily generated and it is used as histogram feature of the separate partial sequence.

CCC	CCT	CCG	CCA	CTC	CTT	CTG	CTA
0	1	2	3	4	5	6	7
CGC	CGT	CGG	CGA	CAC	CAT	CAG	CAA
8	9	10	11	12	13	14	15
TCC	TCT	TCG	TCA	TTC	TTT	TTG	TTA
16	17	18	19	20	21	22	23
TGC	TGT	TGG	TGA	TAC	TAT	TAG	TAA
24	25	26	27	28	29	30	31
GCC	GCT	GCG	GCA	GTC	GTT	GTG	GTA
32	33	34	35	36	37	38	39
GGC	GGT	GGG	GGA	GAC	GAT	GAG	GAA
40	41	42	43	44	45	46	47
ACC	ACT	ACG	ACA	ATC	ATT	ATG	ATA
48	49	50	51	52	53	54	55
AGC	AGT	AGG	AGA	AAC	AAT	AAG	AAA
56	57	58	59	60	61	62	63

Fig. 2- Reference table

As the input query base sequence is divided into n partial sequences, the histograms of n parts are generated. On the other hand, the histogram features can also be extracted from the DNA sequence in the database using the same method respectively. In our previous algorithm, the histogram generated from each partial sequence is then compared with the histogram from the same partial sequence in the database by calculating similarities between them. The shortcoming of this approach is, when the difference of sequence length between the input base sequence and that in the database is large, the error of the normalization of histogram cannot be ignored.

In this paper, we propose a local search approach to resolve this problem. As shown in figure 3, when the histograms of n parts of input query base sequence are generated, a search processing will be carried out to get a best matched part in the database for each partial sequence. The similarity between these histograms is used and the best match will be located. Next, the partial sequence is then extended from both sides of it until the corresponding similarity between the partial sequence belonging to input query base sequence and that in the database does not increase any more.

The histogram generated from each extended partial sequence is then compared with the histograms from the corresponding matched partial sequence in the database by calculating similarity (si) between them (as shown in formula (2)). Then the integrated similarities (S) are obtained by averaging as shown in the following formula (1).

$$S = \frac{\sum s_i}{n}, i = 1,...i,...n \tag{1}$$

$$s_i = 1 - \frac{\sum_{j=1}^{64} \left|(freq_j^{in(i)} - freq_j^{db(i)})\right|}{2N} \tag{2}$$

$freq_j^{in(i)}$, $freq_j^{db(i)}$ are the frequencies of 3-dimensional vectors that belong to a separate partial sequence of an input query sequence and that belong to the same separate partial sequence of full length sequences in the database, respectively. N is number of vectors in the separate partial sequence.

Then we compare the integrated similarities (S) with a given threshold (T), only the sequences whose similarities exceeded the given threshold are then aligned using exhaustive Smith-Waterman dynamic programming algorithm [6].

Experiments and Discussions

Data sets

GenBank is the NIH genetic sequence database, an annotated collection of all publicly available DNA sequences. There are approximately 106,533,156,756 bases in 108,431,692 sequence records in the traditional GenBank divisions and 148,165,117,763 bases in 48,443,067 sequence records in the WGS division as of August 2009 [9].

We have downloaded plant sub-database of GenBank DNA database which contain approximately 1,432,314 sequences. From this sub-database, 853,825 DNA sequences with the sequence length within 400-2000 have been selected to be used in the experiments. The performance and reliability of the developed algorithm was evaluated. The query sequences have been chosen randomly from the 853,825 sequences.

We performed all of the experiments on a conventional PC@3.2GHz (2G memory). The algorithm was implemented in ANSI C.

Experimental results

We select 50 results with highest scores among the whole results of the entire DNA sequences which given by the Smith-Waterman algorithm [6] and perform the same search by using histogram information algorithm and calculating the recall and the precision. Recall indicates the proportion of results yielded from histogram information algorithm to the highest 50 scores and precision indicates the proportion of correct scores included in the results from histogram information algorithm. The recall and precision are defined as follows.

$$\text{Recall} = \frac{\text{number of correct - match}}{\text{total number of positives}} \tag{3}$$

$$1 - \text{precision} = \frac{\text{number of false - match}}{\text{total number of matches}} \tag{4}$$

Figure 4 shows the experimental results with variations of overlapping steps, where the partial sequence length is 80. It can be seen the best performance is given at the step 40, where the search domain for the recall of 1.00 is about 0.269% of the whole range 853,825 with the sequence length within 400-2000. The comparison result of required search time for the experiment is shown in Table 1. The time spending of the same search with histogram information algorithm is about 37.4 seconds, which is 0.518% of about 2 hours (7207.8 seconds) of exhaustive search by Smith-Waterman algorithm and is about 2.78 times faster than BLAST algorithm. We can obtain the same results in all cases.

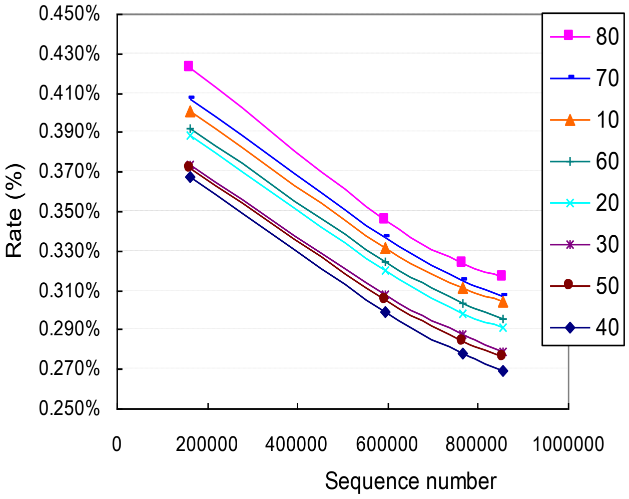

Fig. 4- Experimental results with overlapping step variations

Table 1- Comparison with conventional algorithms and proposed algorithm

Query	Smith Waterman (s)	Proposed Method (s)	Blast (s)
Q1	7527	33	106
Q2	7125	29	101
Q3	6942	43	111
Q4	7296	34	105
Q5	7369	31	103
Q6	7064	57	99
Q7	7198	39	105
Q8	7271	27	97
Q9	7469	48	102
Q10	7357	31	112
Ave.	7207.8	37.4	104

Conclusions

In this paper, we proposed an improved local search algorithm that improves both the speed and the precision of search by combining histogram features and Smith-Waterman dynamic programming algorithms in the fast search of DNA sequences. Experimental results using GenBank sequence data show the proper overlapping step will give more robust resulting and the proposed algorithm is more efficient compared with conventional algorithms for DNA sequence search.

References

[1] Venter J.C., Adams M.D., Myers E.W. (2001) *Science*, 291, 5507, 1304 -1351.

[2] Collins F.S., Morgan M. and Patrinos A. (2003) *Science*, 300, 5617, 286-290.

[3] *http://www.embl.org/.*

[4] *http://www.ddbj.nig.ac.jp/.*

[5] Needlman S.B. and Wunsch C.D. (1970) *Journal of Molecular Biology*, 48, 443-453.

[6] Smith T.F. and Waterman M.S. (1981) *Journal of Molecular Biology*, 47, 195-197.

[7] Altscgul S.F., Gish W., Miller W., Myers E.W. and Lipman D.J. (1990) *Journal of Molecular Biology*, 215, 403-410.

[8] Lipman and Pearson W.R. (1985) *Science*, 227, 1435-1441.

[9] GenBank, *ftp://ftp.ncbi.nih.gov/genbank/.*

[10] *http://www.ncbi.nlm.nih.gov/genbank/genbankstats.html.*

[11] Li M. and Ma B. (2003) *Genome Informatics*, 14, 164-175.

[12] Ma B., Tromp J. and Li M. (2002) *Bioinformatics*, 18(3) 440-445.

[13] Chen Q., Kotani K., Lee F.F. and Ohmi T. (2011) *International Journal of Bioinformatics Research*, 3(1) 161-166.

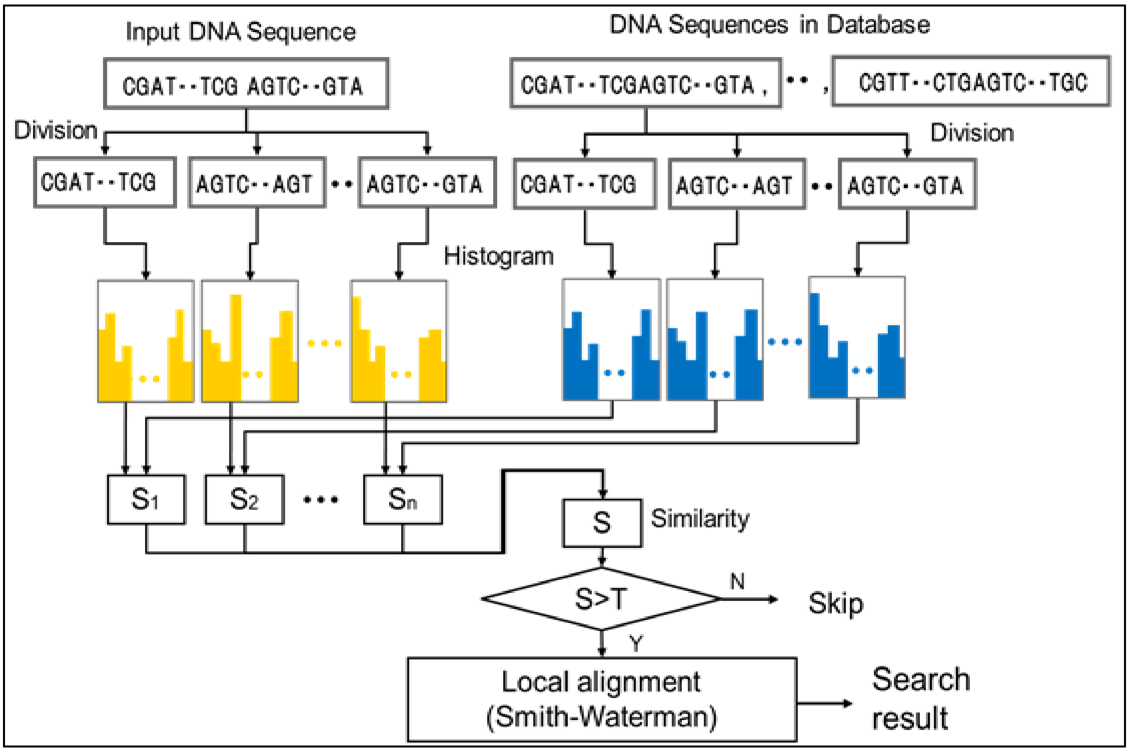

Fig. 1- Processing steps of proposed algorithm

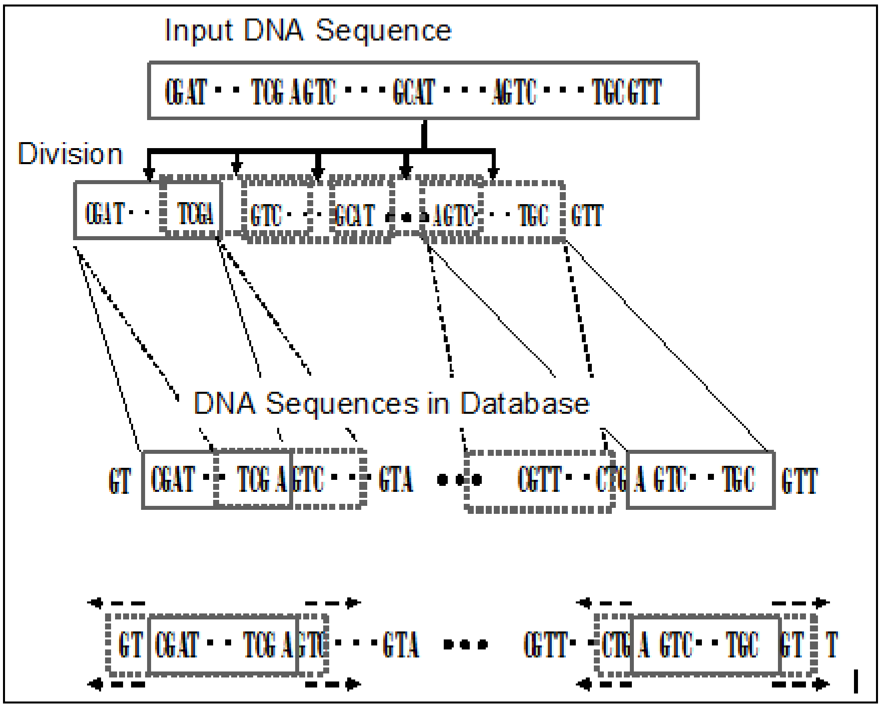
Input DNA Sequence
CGAT · · TCG AGTC · · · GCAT · · · AGTC · · · TGC GTT
Division
CGAT · · TCGA GTC · · GCAT · AGTC · · TGC GTT
DNA Sequences in Database
GT CGAT · · TCG AGTC · · GTA ••• CGTT · · CTG A GTC · · TGC GTT
GT CGAT · · TCG AGTC · · · GTA ••• CGTT · · CTG A GTC · · TGC GT T

Fig. 3-Local search approach