Suffix Tree 알고리즘을 이용한 유전자 서열들의 클러스터링과 어노테이팅에 관한연구

이성근, 한상일, 황규석*
부산대학교 화학공학과 공정시스템연구실
(kshwang@pusan.ac.kr*)

The Study on Clustering and Annotating Gene Sequences using Suffix Tree Algorithm

Sung Gun Lee, Sang il Han, Kyu Suk Hwang*
Department of Chemical Engineering, Process Systems Laboratory, Pusan National University
(kshwang@pusan.ac.kr*)

서론

최근 몇 년 동안 컴퓨터를 비롯한 실험장비가 발달함에 따라, 여러 생물에 관련된 genomic data가 급속히 증가하고, data를 빠르고 정확하게 분석할 수 있게 되었다. 이러한 data들을 다루는 방법 중의 하나인 multiple sequence alignment는 세 개 이상의 단백질이나 DNA 서열들을 배열하여서 유사하거나 같은 부분을 찾아낸다. 그러나 기존의 SP-method, CLUSTALW, PILEUP 을 비롯한 multiple sequence alignment 방법들은 pairwise comparison 을 하므로 서열의 개수가 증가할수록 검색 시간이 크게 증가하는 단점이 있다. 따라서 본 연구에서는 탐색 시간을 줄이기 위해 pairwise comparison을 하지 않고 여러 개의 서열들을 동시에 비교하기 위해 Suffix Tree Clustering 알고리즘을 구현하여 multiple sequence alignment에 적용하였다. 또한 클러스터링된 gene cluster들을 annotating 하기 위해서 SwissProt나 Unigene 같은 데이터 베이스를 BLAST를 이용하여 검색하였다. 우리는 gene clustering의 6단계를 제시하였다. 1) Constructing suffix tree, 2)Searching and overlapping common subsequences, 3)Grouping subsequence pairs, 4)Masking cross-matching pairs, 5)Clustering pair groups, 6)annotating gene clusters by BLAST search. 우리의 시스템은 박테리아의 TCA cycle에서 가져온 42개의 gene을 모두 11개의 그룹으로 클러스터링 하였다.

본론

STC(Suffix Tree Clustering)는 string의 공유된 조각을 바탕으로 clusters를 만드는 standard clustering methods 보다 더 빠른 incremental, linear time 알고리즘이고, Web documents를 clustering하는 STC 의 절차는 다음과 같다.

The procedure of STC(Suffix Tree Clustering) for web documents clustering;

-Step 1

Document "Cleaning" (the string of text representing each document is transformed)

-Step2

Identifying Base Clusters (searching for sets of document sharing common phrase)

-Step3

Combining Base Clusters (merging base clusters with a high overlap)

우리는 유전자를 클러스터링 하기위해 STC를 도입하여서, document string을 변환하는 불필요한 Document "Cleaning" 단계는 수행하지 않고, Step2와 Step3를 수행하였다. 그리고 매우 긴 길이를 가진 유전자들의 엇갈리는 공통부분을 없애고 common subsequences가 순차적으로 매치되도록 하기 위해 Step3-Combining Base Clusters를 수정하였고, step 3를 two steps (grouping the common subsequence pairs and clustering the common subsequence pair groups)으로 나누어서 유사한 DNA sequence들이 clustering 될 수 있도록 하였다.

Suffix Tree 알고리즘을 이용해 여러 개의 서열들에서 공통으로 존재하는 subsequences를 찾아내고 위치 정보를 테이블화 하여서 sequences을 클러스터링 하였다. 우리가 만든 프로그램의 흐름도는 Fig. 1 과 같다.

우리는 suffix tree를 이용한 gene clustering 프로그램을 만들기 위해 Perl language 를 사용하였다. Perl 은 Practical Extraction and Report Language의 약어이며 프로그래머 Larry Wall에 의해 만들어졌다. 약어의 의미처럼 문자 data을 추출하고구성하는 데에 강력하고 실용적인 언어이므로 시스템 관리와 world wide web에서 CGI프로그래밍에 주로 사용되다가 근래에 genomic data 처리에도 사용되고 있다. 따라서 Perl 은 gene sequence를 다루고 공통부분을 찾아내는 우리의 목적에 부합하는 언어이다.

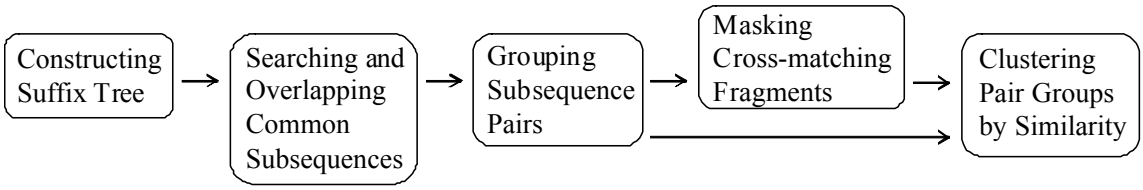

Figure 1. The organization of our gene clustering system

알려지지 않은 DNA나 단백질의 기능을 밝혀내는 것은 중요하다. 그리고 그 기능이 이미 기능이 알려진 gene들을 모아놓은 데이터베이스를 검색하는 것에 의해서 간접적으로 유추할 수 있다. 따라서 우리의 gene 클러스터링 시스템은 BLAST 검색과 결합이 되었다. NCBI 에서 'blastall' 툴을 가지고 와서 GenBank, EMBL, DDBJ, Swissprot databases를 연결하여 검색을 가능하게 하였다. 인터넷을 경유한 데이터베이스의 접속은 불안정하고 느리기 때문에 local BLAST를 우리의 Linux 시스템에 설치를 하였다. Fig. 2는 BLAST 검색 절차를 보여준다.

시스템을 수행한 결과는 사용자에게 세 개의 파일을 제공한다. 파일 'cluster.out'은 gene의 클러스터링된 배열을 보여준다. 파일 'nucleotide_blast.out' 은 DNA 데이터베이스에 대한 BLAST 검색 결과를 보여준다. 파일 ‘protein_blast.out'은 protein 데이터베이스에 대한 BLAST 검색 결과를 보여준다.

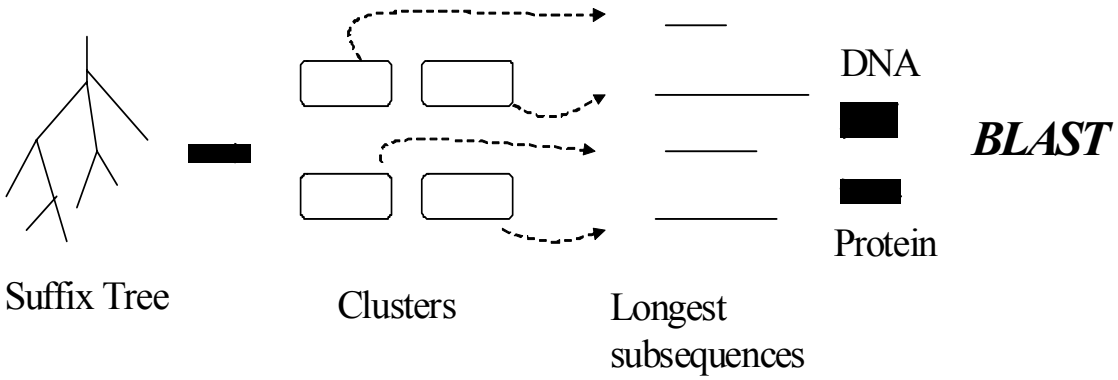

Figure 2. The BLAST search procedure.

결론

본 연구에서는 KEGG에서 TCA cycle의 42개의 gene sequences들을 가지고 와서 우리의 시스템을 평가하기 위해 사용하였다. TCA cycle은 미토콘도리아에서 에너지 생성을 위한 잘 알려진 chemical process 이다. 분석 Intel's Pentium 2.4 GHz processor, 512 RAM, Linux OS에서 수행되었다. 42개의 gene sequences 들로 구성된 query data는 fasta file format 으로 프로그램에 입력된다. 프로그램을 실행하기 위해, 우선 query data file name을 입력하고, 다음 minimum block size(시스템이 인식할 수 있는 최소의 common subsequence 길이)를 입력하고, 마지막으로 BLAST 검색을 수행할 데이터베이스를 선택한다. 여기에서는 minimum block size를 10으로 설정하였다.

그 결과 모두 11개의 clusters 들이 형성되었고 , Fig. 3은 cluster 결과의 일부분을 보여준다.

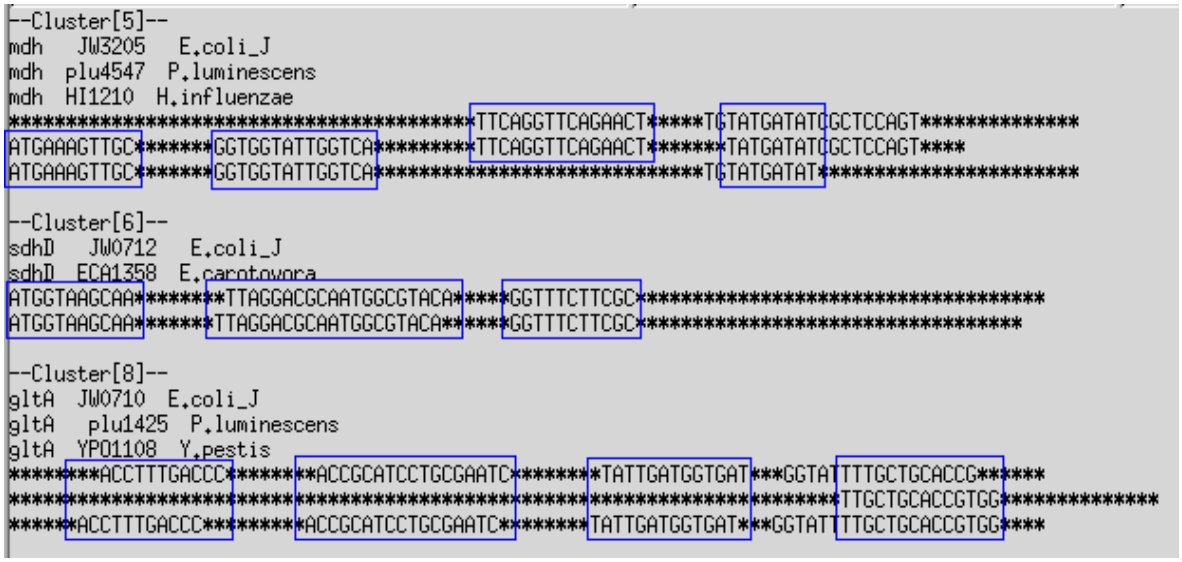

Figure 3. The alignments and clusters.

Fig.3 에서 박스안의 DNA common subsequences 들은 여러 개의 종들 사이에서 서로 보존된 부분을 나타내고, 우리의 시스템에서 cross matching subsequences들을 제거하는 과정을 거쳤기 때문에 공통되는 부분들이 순차적으로 나타나고 있다. 이러한 부분들은 특정한 기능을 나타내는데 필요한 중요한 부분으로 예상 된다.

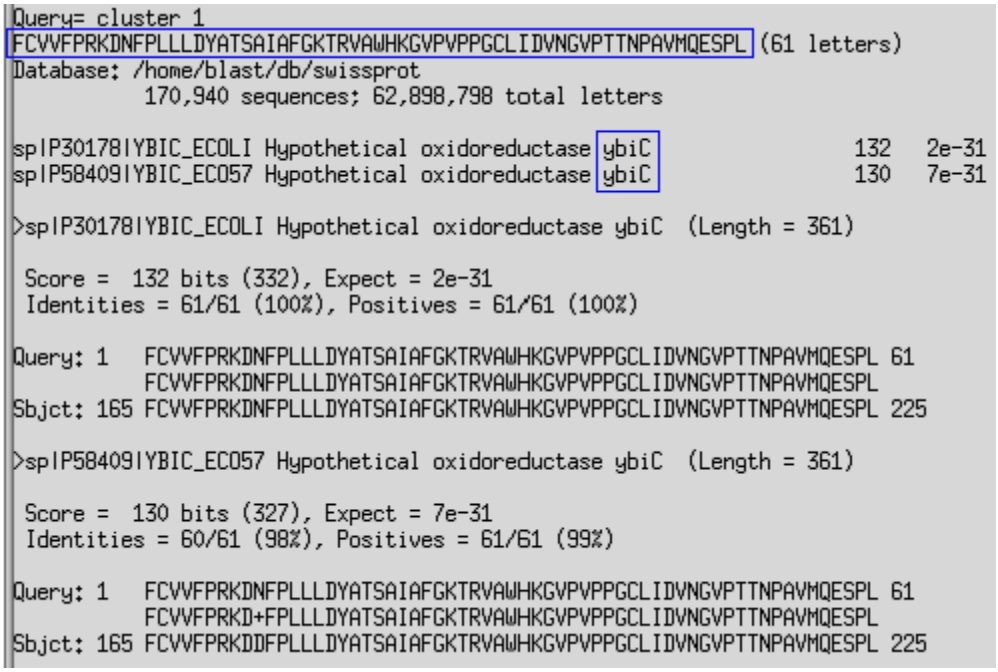

Figure 4. A part of BLAST result.

Fig. 4 는 Cluster 1의 가장 긴 common subsequence 에 대한 단백질 데이터베이스의 BLAST 검색 결과의 일부분을 보여준다. 박스안의 common subsequence는 61의 길이를 가지는 단백질 서열이고 검색 결과 Hypothetical oxidoreductase ybiC 의 기능을 가지는 이미 알려진 서열과 매치가 되었다. 이것으로 우리는 Cluster 1의 서열들이 ybiC의 기능과 관련이 있음을 간접적으로 유추해 볼 수가 있다.

본 연구에서는 선형시간 자료구조 알고리즘은 Suffix Tree 를 이용하여 gene sequence들을 클러스터링하고 클러스터의 기능을 밝혀내고 진화관계를 파악하기 위해서 BLAST 검색을 수행하는 생물학적 도구를 개발하였다. 대량의 자료들을 다루기 위해서 프로그램의 핵심 부분을 low level 언어를 이용하여서 개발을 해야 할 것이고, 정확성을 위해 gap penalty등의 세부적인 옵션을 추가를 해야 할 것이다.

참고자료

1. Altschul, S.F., Gish, W., Miller, W., Myers, E., Lipman, D.J., 1990. Basic Local Alignment Search Tool. J. Mol. Biol. 215, 403-410.
2. Chen, J.Y., Carlis, J.V., 2003. Genomic data modeling. Information Systems, 28, 287.
3. Delcher, A.L., Phillippy, A., Carlton, J., Salzberg, S.L., 2002. Fast algorithms for large-scale genome alignment and comparison. Nucleic Acids Res. 30(11), 2478-2483.
4. Higgins, D.G., Sharp, P.M., 1988. CLUSTAL: A package for performing multiple sequence alignment on a microcomputer. Gene 73, 237-244.
5. McCreight, E., 1976. A space economical suffix tree construction algorithm. Journal of the ACM 23, 262-272.
6. Ostell, J.M., Wheelan, S.J., Kans, J.A., 2001. The NCBI data model. Methods Biochem. Anal. 43, 19.
7. Ukkonen, E., 1995.On-line construction of suffix trees. Algorithmica 14, 249-260.