순차 패턴 마이닝 기법을 이용한 단백질 서열 분류

정광호*°, 김진수*, 최성용*, 한승진**, 이정현***
인하대학교 컴퓨터 · 정보공학과*, 경인여자대학 컴퓨터 정보기술 학부**, 인하대학교 컴퓨터공학부***
{khjung*°, kjspace*, sychoi*, softman**}@nlsun.inha.ac.kr, jhlee@inha.ac.kr***

Classification of Protein Sequence Using Sequential Pattern Mining

Kwang-Ho Jung*°, Jin-Su Kim*, Seong-Yong Choi*, Seung-Jin Han**, Jung-Hyun Lee***
Dept. of Computer Science & Information Engineering, Inha University*,
School of Computer Information Technology, Kyung in Women`s College**,
School of Computer Science & Engineering, Inha University***

요 약

기존의 생물정보학 연구는 전체 서열들의 매칭을 통한 상동성 연구에 중점을 두고 진행되어 왔다. 최근에 서열 데이터베이스의 급격한 증가와 게놈 정보가 축적됨에 따라 서열로부터 다양한 정보를 얻기 위해 서열 데이터 분석에 마이닝 기법을 접목시키고자 하는 다양한 기술들이 제안되고 있다. 단백질과 DNA의 서열 비교는 생물정보학의 기본 작업 가운데 하나이다. 신속하고 자동화된 서열 비교 능력은 새로운 서열에 대한 기능 판별 및 분석 등 모든 작업을 용이하게 한다. 본 논문에서는 동종의 단백질 서열들을 다중 정렬하여 일치하는 구간을 찾아내고, 그 구간에서 아미노산 코드와 위치정보를 이용해 동종 서열들 간의 특정한 패턴 규칙을 찾아내고, 새로운 서열에서 어떤 서열 패턴 특징이 발생하는지를 찾아냄으로써 서열을 분류하는 방법을 제안한다.

1. 서론

최근에 인간 게놈 프로젝트의 성공과 더불어 바이오 관련 실험 장비들의 자동화와 발달로 인하여, 생명 현상의 신비를 밝히는데 기본이 되는 바이오 데이터들이 양적 질적인 면에서 급격히 증가하고 있다. 이에 따라 소극적으로 진행되어 오던 서열 데이터 분석 방법에 다양한 정보검색 기술을 접목한 방법으로 서열로부터 지식을 추출하고자 하는 노력들이 시도 되고 있다[1].

기존에 다중 정렬을 이용한 서열 매칭 기법에서는 두 개의 단백질 서열 사이에 얼마나 많은 수의 서열이 일치하는지에 중점을 둔 상동성 검색이 주된 관심이었다. 이는 높은 상동성을 갖는 서열을 찾는 데는 유리하지만 진화적으로 먼 거리에 있는 종들 간의 유사성을 찾는 데는 한계가 있다. 또한 서열 전체를 비교하기 때문에 시간과 처리에 많은 오버헤드가 걸리고 기능상으로 의미 있는 아미노산과 그렇지 않은 아미노산을 같은 값으로 처리하는 단점이 있다.

본 논문에서는 동종의 서열을 다중 정렬하고 정렬된 서열에서 일치하는 서열의 갭이 없는 구간을 찾아내고 그 구간에서 아미노산 코드와 위치 정보를 이용한 서열 패턴 탐색을 수행하고 이 빈발 패턴을 다른 서열에서 탐색함으로써 그 서열의 분류를 결정한 다.

2. 관련연구

2.1 서열 정렬

서열 정렬은 서로 다른 서열의 상동성을 검사하기 위해 서열에 공백을 삽입하거나 일치하지 않는 서열을 삭제하는 방법 등을 이용하여 최대한 많은 수의 아미노산이 일치하도록 만드는 기법이다. 예를 들어 서열 원소가 일치하면 +1, 불일치하면 -1, 공백을 넣을 때는 -2라는 점수를 주고 가능한 정렬 중 가장 서열의 점수 합계가 높은 결과를 찾는다. 서열을 정렬하는 방법은 크게 세가지로 나눠볼 수 있는데 서열 전체에서의 상동성을 찾는 전체 정렬과, 전체 서열의 특정 부위에서 상동성을 찾는 국부 정렬, 그리고 여러 개의 서열을 동시에 비교하는 다중 서열 정렬이 있다[2]. 특히 다중 서열 정렬 기법은 단백질 서열에 가장 보편적으로 적용되는 기법으로써 여러 개의 서열을 비교하여 정렬된 각 서열에 의해 만들어진 단백질간의 진화적, 구조적 유사성을 표현하는데 적합한 기법이다[3]. 다중 정렬을 위한 도구는 여러 개가 있는데 그 중 일치하지 않는 서열에 갭을 삽입하여 정렬하는 ClustalW와 일치하지 않는 서열을 삭제하여 갭이 없이 완벽히 일치하는 구간을 찾는 BLOCKS가 있다. 이러한 다중정렬 기법을 이용하여 서열을 정렬하면 서열간의 유사한 부분을 찾아볼 수 있는데, 그림 1과 같이 동종의 서열과 이종의 서열을 비교해 보면 동종의 서열에서 많은 아미노산들이 일치하는 것을 알 수 있다[2].

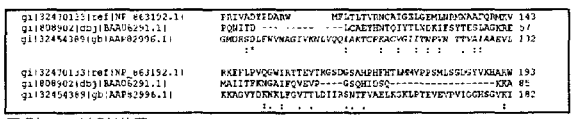

그림 1. 서열비교

2.2 순차 패턴 탐사

순차 패턴은 한 트랜잭션 안에서 발생하는 항목들간의 연관 규칙에 시간의 변이를 추가한 것이다. 순차 패턴에서는 주어진 트랜잭션 데이터베이스에서 최소 지지도(식 1)를 만족하는 모든 시퀀스들 사이에서 최대 시퀀스를 찾는 것이다[4]. 여기서 시퀀스는 트랜잭션 시간에 따라 정렬된 트랜잭션들의 리스트를 말한다. 그러나 바이오 서열 데이터에서는 각 균주의 서열을 트랜잭션의 리스트로 간주하여 최소 지지도를 만족하는 최대 시퀀스를 찾을 수 있다.

순차 패턴 생성 알고리즘에는 AprioriAll, AprioriSome, DynamicSome 알고리즘 등이 있다. 이러한 알고리즘을 이용하여 트랜잭션 데이터 베이스에서 특정한 패턴을 발견할 수 있는데, 바이오 서열 데이터는 단지 20개 아미노산의 조합으로 이루어져 있어서 서열이 길어질 경우 각각의 아미노산 발생 확률이 다른 일반 데이터베이스에 존재하는 아이템의 발생 확률보다 높기 때문에 의미 없는 패턴을 찾을 수 있는 문제를 가지고 있다.

$$\text{support} = \frac{\#\ of\ transaction\ containing\ all\ the\ item\ in\ X \cup Y}{total\ \#\ of\ transactions\ in\ the\ database} \quad (1)$$

3. 다중 정렬된 서열의 순차패턴 탐사 및 분류

본 논문에서는 동종의 유전자 서열을 ClustalW를 이용하여 다중 정렬한다. 정렬된 서열 데이터에서 일치하는 서열들의 갭이 없는 구간에서 빈발 패턴을 발견하기 위해 AprioriAll 알고리즘을 이용하여 순차 패턴을 탐사한다. 그러나 바이오 서열 데이터는 단지 20개의 아미노산 조합으로 이루어진 데이터이므로 같은 아미노산이 또다시 발생할 확률이 높기 때문에 아미노산과 서열내의 아미노산 위치정보를 함께 결합하여 새로이 표기한 후 빈발 패턴을 생성하고 이렇게 생성된 빈발 패턴을 이용하여 서열을 분류한다. 그림 2 는 본 논문에서 제안하는 서열 분류 시스템(Cla_PO_ASS)의 전체 구성도를 보여준다.

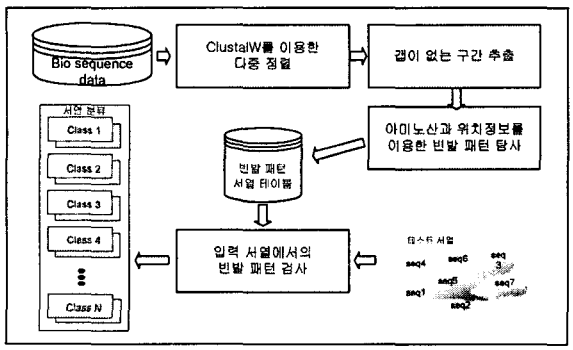

그림 2. 제안된 서열 분류 구성도

3.1 ClustalW를 이용한 다중 서열 정렬

동종의 서열들 중에는 상당한 부분의 아미노산이 일치하고 있고 그 중에서도 특히 많은 서열에서 공통적으로 존재하는 서열은 중요한 의미를 담고 있다. 본 논문에서는 이러한 서열을 찾기 위해 먼저 동종의 서열 데이터를 다중 정렬한다. ClustalW는 European Bioinformatics Institute(EBI)에서 만든 DNA, 혹은 단백질의 아미노산 서열을 다중 정렬 시켜주는 다양한 소프트웨어 중에 하나이다. BLOSUM, PAM 등의 다양한 matrix를 사용하고 있으며 갭의 위치에 따른 보정이 가능하고, 정확한 정렬을 위한 수동정렬 기능 등을 사용하여 정확한 다중 정렬을 수행하기 때문에 가장 많이 쓰이는 응용 프로그램 중에 하나이다[3]. 본 논문에서 찾으려고 하는 빈발 패턴은 기본적으로 정확한 정렬을 했을 때 보다 좋은 결과를 얻을 수 있기 때문에 우수한 평가를 받고 있는 ClustalW를 이용하여, 동종의 서열들 중에서 공통적으로 존재하는 의미 있는 서열들을 찾아내기 위한 전처리 단계로써, 서열들을 다중 정렬한다. 그림 3는 다중 정렬된 Lactobacillus속에 속한 서열들의 모습이다.

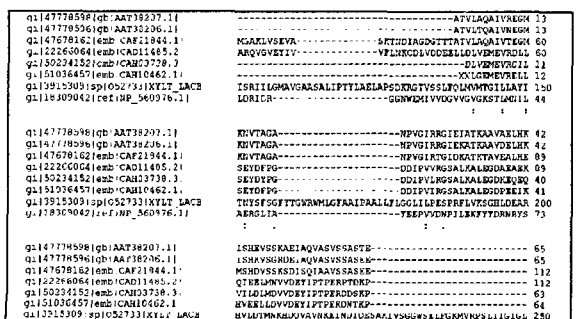

그림 3. 다중 정렬된 Lactobacillus속 서열

그림 3와 같이 다중 정렬된 서열은 짝 정수가 가장 높도록 정렬이 되므로 최대한 많은 서열들이 일치 하도록 정렬된다. 이때 그림 3에서 보이는 바와 같이 동종의 서열을 다중 정렬할 경우 일치하는 서열들의 구간에서 갭이 포함되어 있지 않은 부분이 나타난다.

3.2 위치정보를 이용한 순차 패턴 탐사

순차 패턴을 탐사 하기 위한 알고리즘 중의 하나인 AprioriAll 알고리즘은 원래의 데이터베이스를 고객 시퀀스로 이루어진 데이터베이스로 변환하여 빈발 항목 집합을 찾는다. 그러나 바이오 서열 데이터에 이 알고리즘을 적용할 경우 두 가지를 고려해야 한다. 첫 번째는 각각의 균주를 하나의 고객 시퀀스로 간주하여 빈발 항목 집합을 찾는 것이다. 두 번째는 서열 데이터는 단지 20개의 아미노산 조합으로 이루어 지기 때문에 서열의 길이가 길어질수록 하나의 아미노산이 발생한 후 또 다시 발생할 확률이 높다는 것이다. 이러한 바이오 서열 데이터에 순차 패턴 알고리즘을 그대로 적용할 경우 전혀 의미 없는 빈발 패턴을 발견할 요인이 된다. 이런 문제점을 해결하기 위하여 본 논문에서는 아미노산과 그 아미노산의 서열에서의 위치 정보를 결합한 표기법을 이용하여 아미노산을 표기한다[1]. 즉 첫 번째 자리의 아미노산 서열이 G라면 G1, 두 번째 자리의 서열이 V라면 V2로 표기한다. 다중 정렬된 모든 단백질의 아미노산 코드를 이러한 방식으로 변환하여 표기하면 S3와 S5는 분명하게 다른 아이템으로 간주될 수 있기 때문에 좀더 정확하고 의미 있는 서열들의 패턴을 발견할 수 있다. 표 1은 다중 정렬된 서열들에 위치 정보를 결합한 표기법을 보여준다.

표 1. 위치정보를 이용한 아미노산 표기법

균주	서열								
agilis	A1	E2	A3	E4	K5	Q6	I7	E8	E9
buchneri	K1	E2	Q3	E4	Q5	V6	I7	L8	D9
kimchii	P1	E2	E3	I4	K5	H6	V7	E8	E9
iners	A1	E2	L3	H4	K5	I6	S7	H8	E9

이러한 방법으로 그림 3에서 보여지는 각각의 갭이 없는 구간별로 빈발 패턴을 구하여 빈발 패턴 테이블을 작성한다. 표 2는 2구간에서 발견된 빈발 패턴들의 일부를 보여준다.

표 2. 발견된 빈발 패턴 서열

구간	길이	빈발 패턴 서열	지지도(%)
1	...	...	...
	...	...	...
	...	...	...
2	2	E2, I7	50
	2	E4, I7	50
	4	A1, E2, K5, E9	50
	4	E2, K5, E8, E9	50
	...	...	...
...	...	...	...
	...	...	...
	...	...	...

3.3 빈발 패턴을 이용한 서열 분류

서열을 분류하기 위해 3.2절에서 찾은 빈발 패턴 테이블을 이용한다. 입력 서열의 첫 번째 아미노산부터 값을 읽으며 빈발 패턴 서열들을 저장하고 있는 테이블과 비교 검색하여 일치하는 부분이 있는지를 검사한다. 예를 들어 Lactobacillus속에 속한 sharpeae의 일부 서열이 "LEGDPEQVKVIEELM" 과 같이 있을 때, 첫 번째 아미노산 L부터 시작하여 빈발 패턴 테이블과 비교한다. 앞의 5개 아미노산은 빈발 패턴 서열과 일치하는 부분을 찾지 못하고 서열의 여섯 번째의 E에서부터 빈발 패턴 서열과 일치하는

오습을 확인할 수 있다. E는 1, 2, 4번 빈발 패턴 서열의 E2와 E4에서 시작할 수 있으므로 E를 E2와 E4로 간주한 후, 3번 빈발 패턴은 제외하고 1번 2번과 4번 빈발 패턴에서 다음 아미노산을 검사한다. 1번과 2번 빈발 패턴의 다음 아미노산은 I7이고 sharpeae의 서열과 일치하지 않으므로 1번과 2번 빈발 패턴도 제외한다. 4번 빈발 패턴의 다음 아미노산은 K5이다. 이는 이전 단계에서 sharpeae의 여섯 번째 아미노산 E를 E2로 치환하고 순서대로 번호를 부여하면 5번 자리에 아미노산 K가 있어 정확하게 일치함을 확인할 수 있고, 4번 빈발 패턴의 다음 서열인 E8과 E9 역시 정확하게 일치하고 있음을 확인할 수 있다. 그러므로 Sharpeae서열은 4번 빈발 패턴을 지지하는 agilis와 kimchii서열과 상동성이 있는 서열로 분류 할 수 있다.

4. 실험 및 성능평가

본 논문에서는 유산균으로 잘 알려진 Lactobacillus속에 속하는 균주들을 이용하여 실험하였다. 실험은 Lactobacillus속에 속하는 균주들의 많은 서열들이 일치하는 특정 구간에서의 빈발 패턴을 찾기 위해 National Center for Biotechnology Information(NCBI)에서 제공하는 툴을 이용하여 Lactobacillus속에 속하는 서열 348개중 서열이 밝혀져 있는 50개의 서열을 추출하여 다중정렬 후 빈발 패턴을 찾고 이를 이용하여 NCBI에서 추출한 또 다른 서열 100개를 분류한다. 실험을 위한 시스템 구현은 Visual C++ 6.0을 이용하였고 Pentium4 2.0GHz, 256MB Ram 환경에서 실험하였다.

성능평가는 두 가지 항목을 가지고 시행하였다. 첫 번째 항목은 Lactobacillus서열 50개를 이용하여 빈발 패턴 테이블을 작성하고 이 빈발 패턴 테이블을 이용해 다른 100개의 서열을 분류하고 재현율과 정확률[5]을 구하고, BLOCKS와 ClustalW의 결과와 비교 평가한다. 또한 빈발 패턴을 찾기 위한 최소지지도를 점진적으로 증가시켜 가면서 본 논문에서 제안한 시스템의 성능을 평가하였다. 두 번째 항목은 Lactobacillus속에 속한 다른 50개의 서열을 제안한 시스템으로 분류하고 BLOCKS의 결과를 대조구로 하여 분류결과의 유사도를 측정하였다.

4.1 실험 방법 및 결과

빈발 패턴을 찾기 위한 전처리 과정으로써, Lactobacillus속에 속하는 서열 50개를 추출하여 ClustalW를 사용하여 서열들을 다중정렬한 후 정렬된 서열들의 특정 구간에서 위치정보를 결합한 변형된 아미노산을 AprioriAll 알고리즘의 입력으로 사용하여 빈발 패턴을 찾는다. 발견된 빈발 패턴을 테이블로 만들고 Lactobacillus속에 해당하는 서열 50개와 다른 종의 서열 50개를 무작위로 선정하여 입력하고 빈발 패턴 테이블을 이용하여 서열을 분류하였다. 그림 4는 재현율과 정확률을 ClustalW, BLOCKS와 비교한 결과를 평가한 그래프이다.

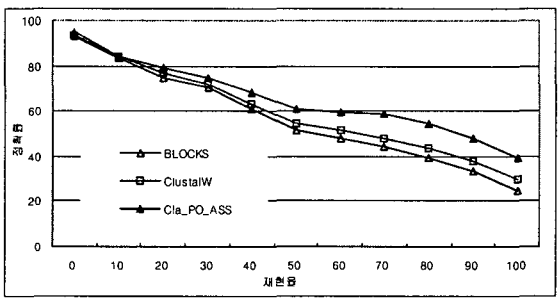

그림 4. 재현율과 정확률 비교

그림 5는 빈발 패턴 테이블을 생성하기 위한 지지도를 결정할 때 기존에 임의로 결정한 지지도를 실험에 의해, 데이터베이스에 적합한 임계치를 구하기 위해, 지지도를 점진적으로 증가시켜가며 제안한 시스템의 재현율과 정확률을 측정한 결과이다. 측정 결과 지지도가 26%일 때 재현율과 정확률이 교차되므로 이때의 지지도가 가장 적합한 임계치 임을 알 수 있다.

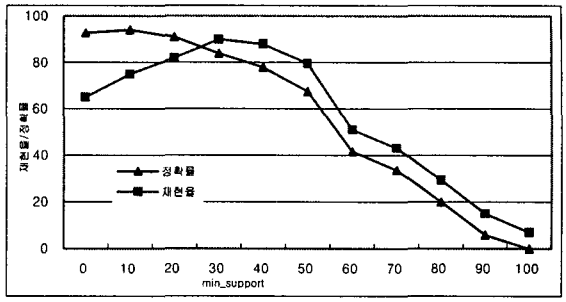

그림 5. 지지도 증가에 따른 재현율과 정확률(%)

마지막으로 Lactobacillus속 서열 50개를 제안한 시스템으로 분류하고 어떤 서열과 유사한 서열로 분류되는지를 실험하고, BLOCKS의 결과를 대조구로 하여 비교하였다. 실험 결과 50개의 서열 중 88%인 44개의 서열이 BLOCKS의 결과와 같은 분류를 하였다. BLOCKS의 결과가 정답이라는 보장을 할 수 없으나 보편적으로 사용되는 서열 분류 도구이므로 본 논문에서 제안한 시스템의 개략적인 성능을 평가 할 수 있다.

5. 결론 및 향후 연구방향

기존의 서열 분석을 위한 여러 가지 방법들은 얼마나 많은 부분이 일치하는지의 여부에 관심을 두고 상동성을 찾는 기법들이 많았다. 이러한 방법은 모든 일치하는 서열을 동등하게 처리 하였다. 본 논문에서는 이러한 점에 착안하여 동종 서열들 간의 빈발 패턴을 찾아내고 찾아진 빈발 패턴이 입력 서열에서 발견되는 지의 여부를 이용하여 서열을 분류하는 방법을 제시하였다. 바이오 서열 데이터는 단지 20개의 아미노산 조합으로 이루어진 데이터이기 때문에 하나의 아미노산이 발생한 후 또 다시 발생할 확률이 높기 때문에 위치 정보를 함께 이용하여 빈발 패턴을 발견하고 서열 분류에 적용하였다.

향후 과제로는 여러 종으로부터 각각의 빈발 패턴을 추출하여 빈발 패턴 테이블을 작성하고 임의의 서열이 입력 되었을 때 이를 적절히 분류 하는지를 실험해 본다. 또한 보다 빠른 서열 분류를 위한 기법을 실험해 본다.

Acknowledgement

본 연구는 정보통신부 대학 IT 연구센터 육성 지원사업의 연구결과로 수행되었습니다.

6. 참고문헌

[1] Y. Gao, K. Mathee, G. Narasimhan, X. Wang, "Motif Detection in Protein Sequences", IEEE, Vol. 22, No. 24, pp.63-72, 1999.

[2] Cynthia Gibas, Per Jambeck, "Developing Bioinformatics Computer Skills", O'REILLY, pp.37-44,253, 2001

[3] J.D. Thompson, D.G. Higgins, and T.j. Gibson., "ClustalW: improving the sensitivity of progressive multiple sequence alignment through sequence weighting", Nucleic Acids Research, Vol. 22, No. 22, pp. 4673-4680, 1994.

[4] R. Agrawal and R. Srikant, "Mining Sequential Patterns", In Proceedings of the 11th International Conference on Data Engineering, pages 3-14, 1995.

[5] R. Baeza-Yates, B. Ribeiro-Neto, Modern Information Retrieval, Addison Wesley, pp.75-79, 1999.