도메인 기반 병렬 단백질 상호작용 예측*

김철환, 정유진
한국외국어대학교 컴퓨터공학과
{redfoot, chungyj}@hufs.ac.kr

A Domain based Parallel Prediction of Protein-protein interactions

Chul-hwan Kim and Yoojin Chung
Dept. of Computer Eng, Hankuk Univ. of Foreign Studies

요 약

본 논문에서는, 단백질은 도메인으로 모듈화 되어 있으며, 단백질 간 상호작용이 도메인간 상호작용의 결과라는데 착안, SVM 을 이용하여 도메인 기반 단백질 상호작용을 예측하는 시스템을 구성하였다. 하지만, SVM 을 통한 실험은 정확한 예측 결과뿐 아니라 빠른 처리속도를 요구하게 되었다. 따라서 SVM 을 병렬화하여 빠른 처리시간을 가지는 Parallel SVM 을 적용, 도메인을 기반으로 한 단백질 상호작용을 실험하였으며, 이를 통해 정확성과 처리 속도를 측정, 비교함으로써 도메인 기반 병렬 단백질 상호작용 예측을 검증해 본다.

1. 서론

생명체의 기본 정보가 저장된 DNA 에서 생성되는 단백질은 생명 현상의 중요한 기능적 역할을 수행하기 때문에 단백질과 관련된 다양한 연구가 진행되고 있다.

그 중 Support Vector Machine(SVM)[1][2]의 사용과 새로운 feature model 의 등장에 따른 단백질 상호작용의 예측에 대한 연구가 활발하다. 하지만 SVM 은 모델 생성을 위한 데이터 생성 방법이 필요하고, 현재까지 나온 방대한 단백질 데이터를 처리하기 위해 많은 연산시간을 필요로 한다.

먼저 단백질은 도메인으로 모듈화 되어있는 형태로 나타난다. 아미노산 서열에서 특정 시퀀스로 표현 가능한데, 도메인이 중요한 이유는 단백질간 상호작용은 도메인간 상호작용의 결과이기 때문이다. 물론 현재로써는 다수의 도메인으로 구성된 단백질들끼리의 상호작용에서 정확히 어느 도메인이 상호작용에 관여하는지 알려져 있지 않으나, 단백질의 1 차 구조를 이루고 있는 도메인 종류에 기반하여 데이터를 표현, SVM 의 입력으로 활용하였다.

하지만 현재까지 나온 방대한 단백질 데이터를 처리하기 위해서는 많은 시간이 필요한데 이는 SVM 을 병렬화한 Parallel SVM 을 이용, 연산시간을 단축하였다. [3] Parallel SVM[4]은 많은 메모리 용량을 요구하던 기존 Standard SVM 의 단점을 개선한 Incremental SVM[5]이 병렬화 된 형태이며, 이는 데이터를 나눠서 처리함으로써 적은 메모리 이용과 빠른 연산이 가능하게 되었다.

이 논문에서는 도메인 기반 단백질 상호작용 예측 시스 템을 제안하고, Parallel SVM 을 이용하여 병렬화된 예측 시스템의 성능 개선을 알아 본다.

* 본 연구는 한국과학재단 목적기초연구 (R01-2003-000-10860-0) 지원으로 수행되었음

2. 실험 개론

2.1 Support Vector Machine(SVM)

상호작용 예측 시스템에 적용된 SVM 은 기본적으로 두 범주를 갖는 객체들을 분류하는 방법이다. 이는 우리가 목적으로 하고 있는 상호작용 예측 시스템의 상호작용이 " 있는가" , " 없는가" 를 구분하는데 있어 적절한 모델이며 최근에 들어 그 성능을 인정받아 다양한 분야의 예측 시스템에서 적용이 되고 있는 개념이다.

2.2 Incremental Proximal SVM

Incremental Proximal SVM Fung 과 Mangasarian 에 의해 개발 되었다.

Incremental Proximal SVM 은 기존 Standard SVM 에서 두 객체를 분리하는 최적의 하이퍼플레인을 찾는 문제에 많은 연산시간을 발생하게 만들었던 알고리즘을 간단하고 빠른 알고리즘으로 바꾸었다.

후에 Fung 과 Mangasarian 에 의해 Increments 와 Decrements 가 가능하게 개선 되었는데, 이것은 전체 데이터를 한번에 처리하는 것이 아닌, 일정한 크기로 나눈 후 각각 처리하는 방법으로 많은 연산 시간을 줄이게 하였다.

이러한 Increments 요소는 병렬화 계산을 가능하게 하였다. Heap-based Tree Topology 를 이용, leaf node 에서 나뉘어진 데이터를 각각 처리하고, 마지막 연산을 Top Node 에서 수행하는 방식으로 병렬화된 Incremental Proximal SVM 이 Amund 와 Havard 에 의해 구현되었다.

2.3 Parallel SVM 의 입력 데이터 생성

단백질은 도메인으로 모듈화된 형태로 나타난다. 도메인은 단백질의 1 차 구조인 아미노산 서열에서 특정 시퀀스로 표현 가능한데, 도메인이 중요한 이유는 단백질간 상호작용은 도메인간 상호작용의 결과이기 때문이다.

실험에 적용한 데이터는 DIP(Domain Interacting Proteins)[6]에서 제공하는 yeast 종의 단백질을 사용하였다.

하지만 DIP 의 데이터베이스는 단백질의 도메인 정보를 포함하지 않고 있다. 그래서 DIP 에서 Cross reference 로 제공하고 있는 PIR(Protein Information Resource)[7]의 데이터 베이스를 이용하였다.

PIR 에서 제공하는 데이터베이스에서 도메인 이름과 단백질이 갖고 있는 도메인의 시퀀스 내의 위치 정보를 추출하여, DIP 의 데이터베이스인 Fasta 포맷의 파일과 통합해, 새로운 Fasta 포맷의 파일을 생성하였다. [그림 1]

```
>DIP:1N|sw:P19527|pir:A21762|gi:112046|D
KARMSSLARAELEKRIDSLMDEIAFLKKVHEEEIAELQAQIQYAQISVEMDVSSKPDLSA.
>DIP:2N|sw:|pir:A23003|gi:83621|D
MKKQNLNSILLMYINYIINYFNNIHKNQLKKDWIMGYEYMYKFLMNNMTCFIKWDNNKIL:
>DIP:3N|sw:P06778|pir:A23282|gi:83448|D
MAFLSYFATENQQMQTRRLPRTAEGSGGFGVLLMNEIMDMDEKKPVFGNHSEDIQTKLDK:
>DIP:4N|sw:P04925|pir:A23544|gi:91067|D:SIG:1-22:CTP:232-254
MANLGYWLLALFVTMWTDVGLCKKRPKPGGWNTGGSRYPGQGSPGGNRYPPQGGTWGQPH
>DIP:5N|sw:P07207|pir:A24420|gi:85085|D:TMM1:27-43:EGX1:297-:
MQSQRSRRRSRAPNTWICFWINKMHAVASLPASLPLLLLTLAFANLPNTVRGTDTALVAA:
>DIP:6N|sw:P08700|pir:A24427|gi:418834|D:SIG:1-19
```

[그림 1] fasta file + domain information

그리고 단백질이 갖고 있는 도메인 정보를 표현하기 위해 각 도메인 별로 고유 번호를 할당하였다. 그 결과 2395 개의 도메인에 번호를 붙일 수 있었으며, 이 고유 번호를 바탕으로 DIP 에서 제공하고는 Yeast 의 상호작용 리스트에서 상호작용이 있는 두 단백질간에 포함되어 있는 도메인에 대한 고유 번호를 Parallel SVM 의 입력으로 사용하였다.

하지만, 할당된 1 부터 2395 번까지의 도메인 고유번호를 적용하였을 경우 SVM 모델 생성시에 vector 크기의 문제로 인해 올바른 모델이 생기지 않았다. 그래서 모델 고유번호에 각각 0.1, 0.01, 0.001, 0.0001 배수를 취해 Parallel SVM 의 입력으로 적용하여 예비 실험을 해 보았다. 그 결과 0.001 배수가 가장 좋은 수행 결과를 보여줌을 알아냈으며 Parallel SVM 의 실험 데이터로 선택하였다. [그림 2]

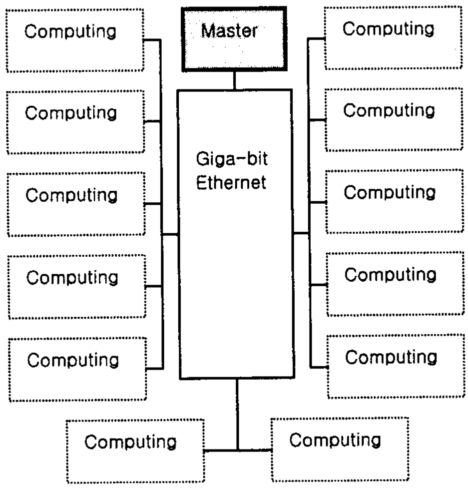
```
1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0.236000,0.2:
1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,(
1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,(
1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,(
1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,(
1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,(
1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,(
1,0.004000,0.004000,0.004000,0.004000,0.004000,
1,0.004000,0.004000,0.004000,0.004000,0.004000,
1,0.004000,0.004000,0.004000,0.004000,0.004000,
```

[그림 2] Parallel SVM 의 입력

3. 실험 및 평가

3.1 실험 환경

본 논문에서 실험을 진행하기 위해 13 Node 의 병렬 컴퓨팅 환경을 구축하였다. 각 Node 의 시스템은 PentiumIII-866Mhz 의 CPU 와 256MB 의 RAM 을 가진 컴퓨터로 이루어져 있으며, 각 Node 는 Giga-bit Ethernet 으로 연결 되었다. [그림 3] OS 는 Linux 로 이루어져 있으며, 병렬 라이브러리로 MPICH(Message Passing Interface)[8]가 이용되었다.



[그림 3] 13 Node PC-cluster

3.2 실험 데이터

DIP 에서 제공하는 15000 여 개로 한정 된 상호작용 리스트에서 도메인을 포함하고 있는 단백질 간의 상호작용 4000 여 개를 뽑아내었다. 그리고 Parallel SVM 의 입력에 필요한 부정예제는 상호작용 리스트에 존재하지 않는 단백질 쌍에서 도메인을 포함하고 있는 단백질 쌍을 선택하였다. [9]

이렇게 완성된 8000 개의 데이터에서 4000 개를 선택하여 하나의 실험을 진행하는 방식으로 데이터의 70%를 SVM 의 모델 생성에 30%를 Classify 에 이용, 총 5 회의 실험을 하였다.

3.3 실험 및 평가

Parallel SVM 을 적용한 도메인 기반 병렬 단백질 상호작용 예측에 대한 정확도 및 처리속도에 대한 직접 비교가 불가능 하여 PC-cluster 에서 실행 Node 수에 따라 Parallel SVM 에 대한 정확도 및 실행시간을 비교하였다.

실험 결과는 [표 1], [표 2]와 같다. 각 실험은 1 Node(single mode), 2 Node, 4 Node, 8 Node, 12 Node 에서 진행 되었다. [표 1]은 각 실험 별 정확도 비교 이며, [표 2]는 각 실험 별 실행시간 비교이다.

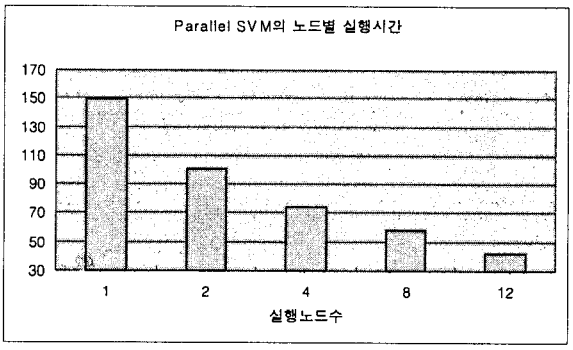
[표 1] 각 실험 별 정확도 비교

Test set	Parallel SVM의 Node 수				
	1 Node	2 Node	4 Node	8 Node	12Node
1	60.62%	58.88%	58.27%	57.23%	58.79%
2	59.98%	56.32%	54.81%	56.84%	57.47%
3	58.32%	54.11%	55.83%	52.13%	55.21%
4	61.03%	58.61%	59.31%	53.74%	57.22%
5	59.02%	55.33%	54.09%	57.84%	55.87%
평균	59.79%	56.65%	56.46%	55.55%	56.91%

[표 2] Parallel SVM 의 실행 시간(초)

Test set	Parallel SVM의 Node 수				
	1 Node	2 Node	4 Node	8 Node	12Node
1	217	192	122	51	41
2	210	188	121	51	42
3	219	195	127	51	42
4	217	191	127	51	43
5	212	190	120	51	41
평균	215	191	123	51	42

[그림 4]는 [표 2]에서의 실행 노드 수에 따른 전체 실행 시간의 평균을 그래프로 표현한 것이다.



[그림 4] 평균 실행시간 비교

4. 결론 및 향후 연구과제

[표 1]에서와 같이 Parallel SVM 을 이용하여 실험한 도메인 기반 병렬 단백질 상호작용 예측의 정확도는 최고 61.03%의 결과를 보여 주었다.

그리고 [표 2]와 [그림 4]에서 볼 수 있듯이 Parallel SVM 을 여러 Node 에서 실행했을 경우 실행시간을 비교해 보면 1 Node 대비 2 Node 일 경우 1.12 배, 4 Node 일 경우 1.74 배, 8 Node 일 경우 4.21 배, 12 Node 일 경우 5.11 배 빠른 실행 결과를 보여주고 있다. 여기서 실행 Node 수가 증가 할수록 실행 시간이 단축되는 것을 볼 수 있으며 실행 Node 수 대비 효율성을 봤을 경우 8 Node 로 실행했을 경우 최고의 Performance 를 보여줌을 알 수 있다. 여기서 도메인 기반 상호작용 예측에서 Parallel SVM 을 이용하여 병렬 실행을 했을 경우 실행시간의 단축 효과를 봄으로써 효율적임을 알 수 있다.

그러나 이 예측 시스템이 사용되기 위해서는 정확도를 높이는 방법이 더 연구되어야 하고, Node 가 더 추가되었을 경우와 실행 데이터의 개수가 대량으로 늘어났을 경우 실행 Node 수와 실행 데이터 수 간의 관계에 대해 연구를 진행해야 할 것이다.

References

[1] Vapnik, V., The Nature of Statistical Learning Theory, Springer-Verlag, New York. (1995)

[2] Burges, C.J.C., " A tutorial on support vector machines for pattern recognition" , Data Mining and Knowledge Discovery, 2, 121-167, (1998).

[3] 김세영, 정유진: 병렬 단백질 상호작용 예측 시스템 정보과학회(2004)

[4] Amund Tveit and Havard Engum : Parallelization of the Incremental Proximal Support Vector Machine Classifier using a Heap-based Tree Topology

[5] Fung, G., Mangasarian, O.L.:Incremental Support Vector Machine Classification. In Grossman, R., Mannila, H., Motwani, R., eds.:Proceedings of the Second SIAM International Conference on Data Mining, SIAM (2002), pp. 247-260

[6] DIP (Database of Interaction Proteins) http://dip.doe-mbi.ucla.edu/

[7] PIR (Protein Information Resource) http://pir.georgetown.edu/

[8] MPICH2(Message Passing Interface) http://www-unix.mcs.anl.gov/mpi/mpich2/

[9] 김철환, 정유진: 단백질 상호작용 예측을 위한 SVM 의 부정예제 생성방법론, 정보과학회 (2004)