식물 TF-miRNA Regulation Network DB의 생성과 연동

Create and Bind Database for TF-miRNA Regulation Network in Plant

김선영
Kim SeonYeong
부산대학교 컴퓨터공학과
s.y.kim@pusan.ac.kr

ABSTRACT

지난 연구에서는 식물 TF-miRNA relation 데이터를 파일로 입력받아 yFiles를 이용해 식물 TF-miRNA regulation network를 그래프로 가시화하였다. 현재는 파일로 데이터가 입력되고 있지만, 이들 데이터는 추후 TF-miRNA regulation 데이터베이스가 구축이 되면 데이터베이스와의 연동을 통해 입력받는 형식으로 바꿔야 한다. 현재 본 연구실에서 데이터베이스 구축 작업을 수행하고 있으나 아직 작업이 완료되지 않아, 본 보고서에서는 이전 파일 정보를 바탕으로 임시 데이터베이스를 생성하여, 이와 가시화 시스템을 연동하는 작업을 수행하였다. 이를 통해 데이터의 입력을 데이터베이스와 연동하여 받을 경우에도 시스템이 정상적으로 동작함을 확인하고 그 성능을 예측해볼 수 있다. 임시 TF-miRNA relation 데이터베이스의 스키마는 파일 데이터의 속성을 바탕으로 설계하였다.

Keywords TF-miRNA regulation network, Temporary Database, binding DB

1 서 론

지난 연구에서는 식물 TF-miRNA relation 데이터를 파일로 입력받아 yFiles를 이용해 식물 TF-miRNA regulation network를 그래프로 가시화하였다[1]. 이 작업은 현존하는 TF-miRNA regulation DB 중 식물과 관련된 내용이 없기에, 이 데이터베이스를 구축하여 식물의 생장, 발달단계, 외부자극에 관여하는 유전자 및 매커니즘에 관심있는 생물학자들이 TF-miRNA regulator와 연관된 기작 [2] 들을 한 눈에 이해할 수 있도록 시각화한다는 점에서 의미가 있었다. 그러나 본 연구실에서 진행 중인 식물 TF-miRNA regulation DB는 아직 그 작업이 완료되지 않았기에 임시로 파일 데이터를 이용하여 시스템이 데이터를 입력받도록 설계했었다. 추후 TF-miRNA regulation 데이터베이스가 구축되는 대로 DB와의 연동을 통해 데이터를 입력받는 형식으로 시스템을 개선해야 한다. 본 보고서에서는 실제 데이터베이스가 구축되기 전에 이전 파일 정보를 바탕으로 임시 데이터베이스를 생성하여, 이와 가시화 시스템을 연동하는 작업을 수행하였다. 이를 통해 데이터의 입력을 데이터베이스와 연동하여 받을 경우에도 시스템이 정상적으로 수행됨을 확인하고 그 성능을 예측해 볼 수 있다. 임시 TF-miRNA relation 데이터베이스의 스키마는 파일 데이터의 속성을 바탕으로 설계하였다.

2 관련 연구

본 보고서에서 진행하는 TF-miRNA 가시화 프로그램의 경우 miRNA와 TF의 연관 기작을 파악하기 쉽게 하는 것이 그 목적이다. 이와 같이 복잡한 생물학적 네트워크 연관 기작들을 파악하기 쉽게 도와주는 Visualization에 대한 연구들이 많이 진행되고 있다.

PseudoViewer3는 그 연구들 중 하나로 Pseudoknot과 함께 RNA의 두 번째 구조를 overlap없이 자동으로 생성하여 그려주는 프로그램이다[3]. Pseudoknot은 이차 구조의 loop에 있는 염기와 loop 외부에 있는 염기와의 결합으로 생성되는 삼차구조인데[4], 이것을 포함한 RNA 구조를 평면에 나타내는 것이 복잡하기 때문에 시각화하기가 어렵다[3]. 때문에 PseudoViewer3의 연구가 의미있는 것으로, PseudoViewer는 RNA 구조를 PNG, GIF, EPS, SVG format등으로 하나의 파일을 생성해서 저장해주며, web application과 web service를 모두 지원하는데 web service client의 경우 C#과 Java 샘플로 제공한다.

VISIBIOweb BioPAX 포맷의 pathway model을 위해 레이아웃과 웹을 기반으로한 pathway 시각화를 제공하는 무료 오픈 소스이다. VISIBIOweb을 이용하여 SBGN의 표준 표기법을 사용한 pathway model의 잘 만들어진 view를 얻을 수 있으며, 원하는 대로 웹 페이지에 하나의 view로써 내장할 수도 있다. VISIBIOweb 의 자동 레이아웃 컴포넌트는 HTTP를 이용해서 다른 툴로부터 계획에 따라 접근할 수도 있다. VISIBIOweb은 클라이언트가 Google maps API를 기반으로 한 작은 JavaScript application 이고, 서버는 Eclipse Graphical Editing Framework를 기반으로 구성한 컴포넌트로 구성되어 있다[5].

VARNA는 PseudoViewer3와 유사한 RNA의 이차 구조 주석과 시각화, 자동 drawing을 위한 도구로써 데이터베이스와 web server를 위한 소프트웨어이다. VARNA는 dbn, ct, bpseq, RNAML 포맷을 사용한 input/output 과 JPEG, PNG, SVG, EPS, XFIG포맷으로 export하는 것을 지원한다. 또한 web server내에서나 command-line argument를 통해서 포인트와 클릭을 통한 접근 모두를 사용해한 결과 drawing의 구조적인 주석과 수동으로 한 수정을 모두 허용한다[6].

생물학적인 결과를 Visualization하려는 연구들은 이 외에도 더 진행되고 있는데, 최신 연구에서 visualization 연구들이 공통적으로 지원하는 것은 web service이다. 따라서 TF-miRNA도 추후 Java Applet을 통한 web service를 지원할 계획이다.

3 데이터베이스 구조

임시 데이터베이스의 스키마는 그림 1와 같다. No는 tuple의 인덱스 번호를 뜻하고 Item1, Item2는 각각 source regulator, target regulator를 의미한다. Direction은 source regulator와 target regulator의 조절 방향을 나타내는 것으로, Function의 유형에 따라 조절의 종류가 달라지며 그래프에서는 edge의 모양이 달라진다. Species는 해당 item들이 속한 식물의 종을 나타내고, Signaling pathway는 외부 자극의 전달 과정을 의미하는 것이다. 여기서 가시화 그래프에 나타나는 정보로는 item의 정보, 각 item 간의 관계를 나타내는 Direction과 Function 정보이다. Species와 Signaling pathway 정보는 가시화한 그래프에 직접적으로 나타나지 않고, 사용자가 이들 정보를 선택함에 따라서 TF-miRNA regulation network가 변화한다.

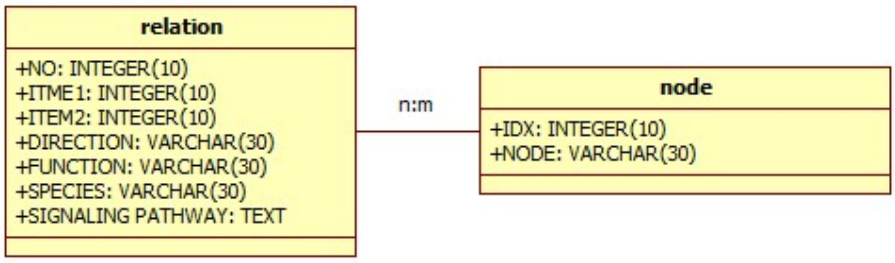


그림 1. 테이블 relation의 스키마. No는 인덱스 번호, item1과 item2는 각각 source regulator, target regulator의 번호를 의미하며, 해당 regulator의 정보는 node 테이블에 포함되어 있다. Direction과 Function은 item간의 관계 정보를 나타낸다. Species와 Signaling pathway는 가시화한 그래프에는 나타나지 않는 정보로 사용자가 선택하면 그에 해당하는 그래프를 볼 수 있다.

지금의 relation 테이블은 데이터의 불필요한 중복이 심하지만 데이터 량이 매우 적기 때문에 데이터의 중복을 피하기 위한 정규화를 시도함으로써 발생하는 손실의 정도가 더 커질 수 있다는 문제점이 있다. 가령 물리적 접근이 복잡해진다든가 길이가 매우 짧은 데이터가 많이 생길 수 있다는 점이 그것이다. 추후 regulator의 추가 정보가 발생할 가능성이 매우 높으므로 현재는 손실을 감안하고 정규화를 수행하였다. 그러나 추후 데이터 양이 일만 개 이상의 규모가 될 경우 node와 edge에 따른 정규화가 반드시 필요할 것으로 사료되며, 이 경우 데이터의 정확성이 높아지고 유연한 데이터를 구축할 수 있다는 점에서 필요한 작업이라고 할 수 있으나, regulator에 추가적인 정보가 들어오지 않을 경우 성능을 고려하여 비정규화를 할 수도 있다.

4 가시화 시스템과 데이터베이스와의 연동

현재 가시화 시스템은 Java language로 구현되어 있고 J2SDK 1.6의 환경에서 동작하며 Java Extensive Java Class Library인 yFiles를 이용하여 TF-miRNA regulation network를 그래프로 가시화하고 있다. 서버환경은 APMSETUP 7으로 구성하여 Apache 2.2.14, PHP 5.2.12, mySQL 5.1.39 버전을 사용하였으며, Java에서 JDBC를 사용하여 데이터베이스를 연동하였다.
아래 코드는 JDBC로 mysql과 Java를 연동한 코드[7]이다. Java에서 지원하는 DriverManager, Statement 클래스를 사용하여 연동하였음을 확인할 수 있다.

```
public class ConnDB :
  private Statement stmt;
  public ConnDB() throws Exception:
       Class.forName(”com.mysql.jdbc.Driver”).newInstance();
      String url = ”jdbc:mysql://localhost/tfmirna?useUnicode=true&characterEncoding=utf8
       &user=root&password=apmsetup”;
       Connection con = DriverManager.getConnection(url);
       stmt = con.createStatement();
```

기존의 파일 포맷 데이터를 파싱하여 데이터베이스에 입력한 간략한 모습은 그림 2 와 같다. 그림 2 의 (a)는 기존의 파일 포맷 데이터의 일부분을 나타낸 것이다. 이 데이터들을 파싱하여 (b)와 같이 데이터베이스화 하였다. 이 과정에서 srcItem, tarItem은 데이터를 가시화할 때 그래프의 node가 되는 부분으로, 추가적인 정보가 생길 수 있기 때문에 (c)와 같이 정규화하였다. 현재는 데이터가 소규모이므로 오히려 불필요한 정규화를 수행한 것으로 여겨질 수 있으나, 추후 대량의 데이터와 각 regulator에 추가적인 정보가 생길 경우 많은 중복을 피할 수 있을 것으로 생각되어 정규화를 수행하였다. 추후 regulator 에 추가 정보가 발생하지 않는다면 정규화한 테이블을 다시 합치거나, 다른 추가 정보를 가진 필드를 정규화할 수도 있다.

```
no  item1       item2        direction  function    species     signaling pathway
1   ath-miR160  ARF10        m2TF    repression  arabidopsis auxin signaling
2   ath-miR160  ARF16        m2TF    repression  arabidopsis auxin signaling
3   ath-miR160  ARF17        m2TF    repression  arabidopsis auxin signaling
4   ARF17       ath-miR160   TF2m    repression  arabidopsis auxin signaling
5   ath-miR164  NAC1         m2TF    repression  arabidopsis auxin signaling
```

(a)

No	srcItem	tarItem	direc	func	species	sigPathway
1	1	8	m2TF	repression	arabidopsis	auxin signaling
2	1	9	m2TF	repression	arabidopsis	auxin signaling
6	2	13	m2TF	repression	arabidopsis	auxin signaling
5	10	1	TF2m	repression	arabidopsis	auxin signaling

(b)

Idx	node
1	ath-miR160
2	ath-miR164
3	ath-miR167
4	ath-miR390
5	ath-miR395
6	ath-miR398
7	ath-miR399
8	ARF10
9	ARF16

(c)

그림 2. (a) 기존 파일 포맷 데이터의 일부분. (b) 파일 포맷 데이터를 파싱하여 데이터베이스화한 모습으로 srcItem과 tarItem은 정규화하여 (c)와 같이 Node 테이블을 새로 생성함. (c) (b)의 srcItem, tarItem의 중복을 피하기 위해 정규화한 테이블.

데이터베이스와의 연동을 통한 TF-miRNA regulation network 그래프의 가시화가 정상적으로 이루

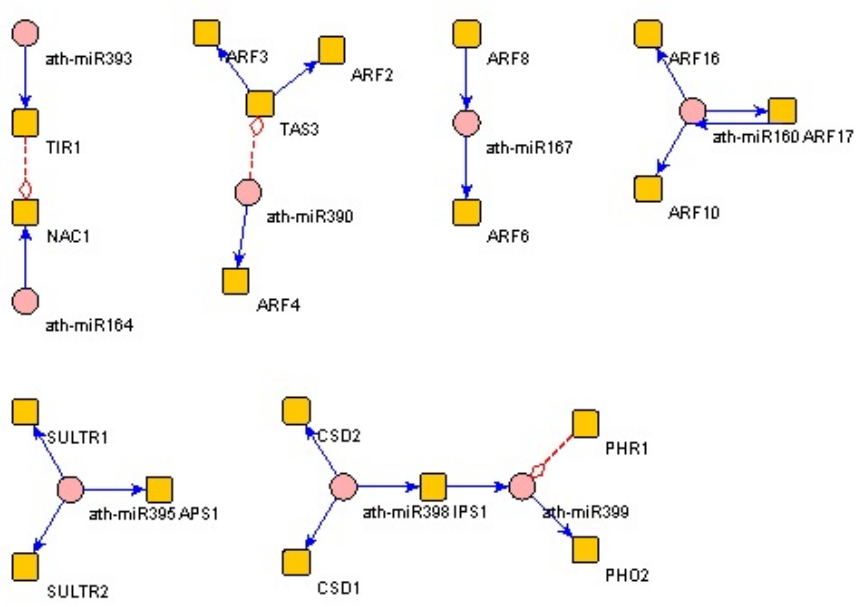
어지는지 판단하기 위하여 동일한 데이터를 파일 포맷과 데이터베이스에 각각 저장한 후 이 데이터들이 동일한 그래프를 그리는지 검사해 보았다.
그림 3 의 (a)는 파일 포맷 데이터와 가시화 시스템를 연동했을 때의 모습이고 (b)는 DB와 가시화 시스템을 연동하였을 때의 모습으로, 두 그래프가 완전히 동일한 것을 확인할 수 있다. 이로써 파일 포맷의 데이터를 데이터베이스에 저장하여 이를 가시화하여도 동일한 그래프를 그릴 수 있다는 것을 확인하였다.

5 결론 및 추후 연구

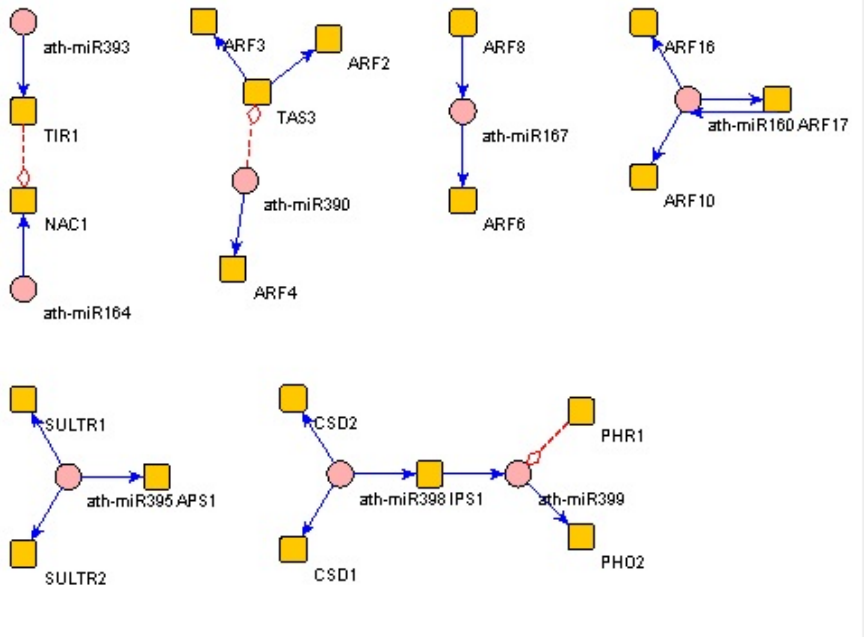
본 보고서에서는 yFiles를 이용해 식물 TF-miRNA regulation network를 그래프로 가시화했던 지난 연구에서 사용한 식물 TF-miRNA relation 데이터를 파일로 입력받는 방식 대신, 실제 TF-miRNA regulation 데이터베이스가 구축될 경우를 대비하여 가상의 데이터베이스를 생성해 이와 시스템을 연동하는 작업을 수행하였다. 종래의 파일 입력 방식을 개선한 임시 데이터베이스 연동 방식은 실제 데이터베이스가 구축되면 바로 대체할 수 있기 위한 사전 작업으로, 임시 데이터베이스의 스키마는 파일 데이터의 속성을 참조하여 설계하였다.
파일 포맷 데이터는 불필요한 데이터 중복이 많이 일어나고 있는데, 현재는 데이터량이 적어 이들 중복을 제거하기 위해 정규화를 수행할 경우 정규화하기 전보다 불필요한 데이터 중복이 더 많이 발생한다. 만약 현재 데이터베이스의 필드 중 추가 정보가 발생하는 필드가 존재할 경우, 이는 반드시 정규화를 통해 데이터 중복을 제거해야 한다. 그러나 필드 중 추가 정보가 발생하는 것이 없다면 정규화하는 것이 더욱 데이터 중복을 심하게 할 수 있으므로 불필요하다.
임시 데이터베이스의 동작 여부는 파일 포맷 데이터를 가시화한 그래프와 임시 데이터베이스와 가시화 시스템을 연동한 후의 가시화 그래프가 정확히 일치함으로써, 데이터베이스 연동 후 시스템이 정상적으로 동작함을 확인하였다. 추후 실제 데이터베이스가 구축되면 임시 데이터베이스 대신 실제 스키마에 기반하여 테이블을 일부분 재설계해야 할 필요가 있고, Java applet을 통해 가시화 시스템을 웹서비스 할 수 있도록 시스템을 업데이트할 계획이다.

참고 문헌

1. Kim SeonYeong, "Yfiles를 이용한 mirna-tf regulation network의 그래프 가시화," *Technical Report*, 2010.
2. Wang-Xia Wang, Bobby Gaffney, Arthur G.Hunt, and Guiliang Tang, "Micrornas(mirnas) and plant development," *ENCYCLOPEDIA*, 2007.
3. Yanga Byun and Kyungsook Han, "Pseudoviewer3 : generating planar drawings of large-scale rna structures with pseudoknots," *Bioinformatics*, vol. 25, no. 11, pp. 1435–1437, 2009.
4. 이동규 and 한경숙, "유전자 알고리즘을 이용한 rna pseudoknot 예측," in *proc. of 한국정보과학회 춘계학술발표회*, vol. 29, pp. 173–187, 2002.
5. Alptug Dilek, Mehmet E. Belviranli, and Ugur Dogrusoz, "Visibioweb: visualization and layout services for biopax pathway models," *Nucleic Acids Research*, vol. 38, no. 10, pp. 150–154, 2010.
6. Kevin Darty, Alain Denise, and Yann Ponty, "Varna : Interactive drawing and editing of the rna secondary structure," *Bioinformatics*, vol. 25, no. 15, pp. 1974–1975, 2009.



(a)



(b)

그림 3. (a)는 파일 포맷 데이터와 가시화 시스템를 연동했을 때의 모습, 28개의 Regulator 중 8개는 miRNA, 20개는 protein으로 구성된 것을 확인할 수 있다. (b)는 DB와 가시화 시스템을 연동하였을 때의 모습, (a)와 마찬가지로 28개의 Regulator 중 8개는 miRNA, 20개는 protein으로 각 regulator간의 연관 기작도 동일한 것을 확인할 수 있다. 그러므로 파일 포맷의 데이터를 데이터베이스에 저장하여 이를 가시화하여도 동일한 그래프를 그릴 수 있다는 것을 확인하였다.

7. Richard G. Baldwin, "Gamelan.com," http://www.developer.com/java/data/article.php/3417381/Using-JDBC-with-MySQL-Getting-Started.htm.