문서 진화 과정 가시화를 위한 PhyloView의 개발

A Development of PhyloView for Visualize Document Evolution

부산대학교 컴퓨터공학과
Park Sun Young
E-mail : parksy@pusan.ac.kr

Revised at 2010.10.30

ABSTRACT

내용 기반의 유사 문서 탐색 시스템인 DeVAC은 사용자의 편의를 위해 클라이언트 프로그램(DeVAC Manager)를 제공한다. DeVAC Manager의 주요 모듈 중 하나인 PhyloView가 제대로 동작하지 않고, 현재 PhyloView의 소스코드가 소실되어 수정이 불가능한 상태였기 때문에 이전 보고서에서 DeVAC Manager의 소스 코드를 수정하여 동작하지 않던 PhyloView를 정상적으로 동작할 수 있게끔 수정하였다. 하지만 전체 프로그램의 구조를 해칠 수 있다는 판단 하에 본 보고서에서는 PhyloView 모듈을 전면 재 개발하였다. 개발 환경은 Java이며, 그래프 전문 라이브러리인 yFile의 최신 버전(2.7.0.1)을 사용하였다. 우선 1) 기존 버전의 기능 중 일부 불필요한 기능을 제외한 모든 기능의 정상적인 동작이 가능하도록 하고, 2) 이후 유지 보수가 쉽도록 잘 모듈화하는 것을 목표로 하였다. 또한 3) 소규모 트리에 대한 Grouping을 통해 사용자가 화면을 효율적으로 활용할 수 있도록 하는 기능을 추가하였다. 개발 결과, 기존 기능 중 상당수의 기능을 복구하였으나 아직 일부 기능을 추가하여야 하며, 모듈화는 잘 구성되었다. 실험 데이터에 대해 로딩 시간 및 레이아웃 변경 실험을 진행한 결과, 기존 200개 정도의 제한에 비해 500개 정도까지는 5초 이내에 연산이 되는 것을 확인하였다. 추후 yFile의 PhyloView의 기존 기능을 완전히 복구한 후, 사용자 편의성을 강조한 새로운 기능을 추가할 예정이다.

KEYWORDS text plagiarism, document evolution, similar document, yFiles, phylogenetic tree

1 서 론

DeVAC Manager는 내용 기반의 유사 문서 탐색 시스템인 DeVAC의 클라이언트 측 프로그램이다. 이 프로그램은 다음과 같은 기능을 수행한다.

1. DeVAC 시스템에 입력하기 위해 텍스트 파일로부터 dvc 포맷 생성
2. 입력된 데이터에 대한 결과를 분석하여 사용자에게 보여줌
3. 문서 간 진화계통도(Phylogenetic tree)를 보여줌 - PhyloView

특히, PhyloView는 DeVAC Manager의 핵심적인 기능 중 하나임에도 불구하고 제대로 동작하지 않아, 지난 보고서[1]에서 DeVAC Manager의 소스 코드를 수정함으로써 DeVAC Manager의 버전 업그레이드와 함께 기존 PhyloView의 기능을 동작하게 하였다. PhyloView는 소스 코드가 소실되어 현재의 상태로는 DeVAC Manager의 시스템 구조도 해칠 수 있고, 더 이상의 유지 보수도 불가능한 상태이다. 따라서 본 보고서에서는 그래프 라이브러리인 yFiles[2]의 새 버전인 2.7.0.1 complete를 이용하여 PhyloView를 완전히 새로 개발하는 것을 목표로 한다.

2 개발 목표 및 환경

2.1 개발 환경

새 버전의 PhyloView 개발을 위한 환경은 표1과 같다. PhyloView는 Java 기반으로 작성된다. 따라

개발 환경	
OS	Windows 7(XP, Vista에서 실행 가능)
Language	Java SE 1.6
IDE	Ecilpse 3.5.2 (Galileo)
External Library	yFiles 2.7.0.1 complete

표 1. PhyloView의 새로운 개발을 위한 개발 환경

서 Windows 계열 뿐 아니라 Linux 계열의 환경에서도 충분히 동작 가능할 것으로 판단되나, DeVAC Manager가 Windows에서만 동작하므로 실질적으로 PhyloView도 Windows 계열에서 정상 동작하는 것이 가장 중요하다. 주요 외부 라이브러리는 yFiles이다. yFiles는 그래프 시각화에 전문화된 라이브러리로, 각종 그래프 시각화와 레이아웃에 관련된 기능이 다수 포함되어 있다.

2.2 최종 개발 목표

PhyloView 새로운 버전의 최종 개발 목표는 다음과 같다.

1. 기존 PhyloView에 존재하는 모든 기능 복원(일부 불필요한 기능 삭제)
2. 개발 이후 유지 보수가 용이하도록 모듈화
3. 주기적으로 새로운 기능 추가

기능	설명
표절 계통 그래프 가시화	Zoom, Distance 조절 가능
그래프 정보 표시	Original Graph, Restructed Graph, Current State에 대한 정보 표시
레이아웃	Circular, Organic, Smart Organic, Hierarchic, Random, Othgonal 등 6가지 지원
그래프 검색	Shortest Path, Longest Path, Vertex, Circle, Inversion, Multi Entry, Strongest Connected Component
그래프 재구성	Original Graph, Weight Filtering, Minimum Spanning Tree, Directed Spanning Tree, Directed Spanning Forest, Source Search, Remove MultiEntry

표 2. PhyloView에서 제공하던 기능들

첫 번째 목표는 기존 PhyloView에 존재하는 기능 중, Random Layout 등 효용성이 매우 낮은 일부 기능을 제외하고 최대한 많은 기능을 복구하는 것이다. 기존 PhyloView에서 제공하는 기능은 표 2 와 같다. 기능의 복원은 표 2에 표시된 것들을 순차적으로 복구하여 기존의 PhyloText를 생성할 예정이다.

두 번째 목표는 PhyloView 개발 이후 기능 추가나 개선 등 유지 보수가 용이하도록 각 기능을 모듈화하는 것이다. 이를 위해 최대한 많은 기능을 클래스로 분할하고, 이를 문서화할 필요가 있다.

세 번째 목표는 PhyloView의 버전을 업그레이드하면서 필요한 기능들을 추가하는 것이다. 현재 구상중인 기능은 소규모 컴포넌트에 대한 자동 grouping을 통한 공간 효율성 제고, vertax 및 edge 개수 제한치 대폭 증가, cycle 추적 및 자동 삭제 기능, 자동 길이 조절 혹은 mouseover 등을 통한 효과적인 labeling 등이다.

3 시스템 구현

현재까지 구현된 내용은 및 추후 개발 계획은 다음과 같다.

[개발한 내용]

1. 시스템 전체 레이아웃 구성
2. 표절 계통 그래프
    1) 복원 - 표절 계통 그래프 그리기
    2) 추가 - Fit Content 기능, Zoom 기능 개선(마우스 휠 동작 가능)
    3) 삭제 - Distance 조절 기능
3. 그래프 정보 창 구성 및 내용 복원
    1) 복원 - Vertex, Edge, Cycle 정보
4. Layout 메뉴 구성 및 기능 복원 및 추가
    1) 복원 - Circular, Organic, SmartOrganic, Hierarchic, Othgonal
    2) 추가 - CompactOrthogonal
    3) 삭제 - Random Layout
5. Search, Reconstruct 메뉴 구성

[추후 개발할 내용]

1. 자동 grouping을 통한 시각화 효율성 제고
2. vertex 및 edge 최대 표시 가능 개수 증가
3. cycle 추적 및 자동 삭제 기능
4. 자동 길이 조절, mouseover 등을 통한 효과적인 labeling

추가한 기능 중 Fit Content 기능은 화면을 현재 작업창을 가장 효율적으로 사용하게끔 zoom을 자동으로 수행하는 기능이다. 이 기능을 추가함으로써 Distance 조절 기능이 불필요해져 삭제하였다.

또한 Compact Orthogonal Layout을 추가하였다. 기존에 존재하던 Random Layout은 그 효용성이 떨어져서 삭제하였다. 그 외 기존 기능에 대해서는 인터페이스가 대부분 완성되었으나, 실제 기능을 모두 다시 복원하는 데에는 시간이 조금 더 필요할 것으로 판단된다. 새로운 PhyloView의 실행 화면은 그림 1과 같다.

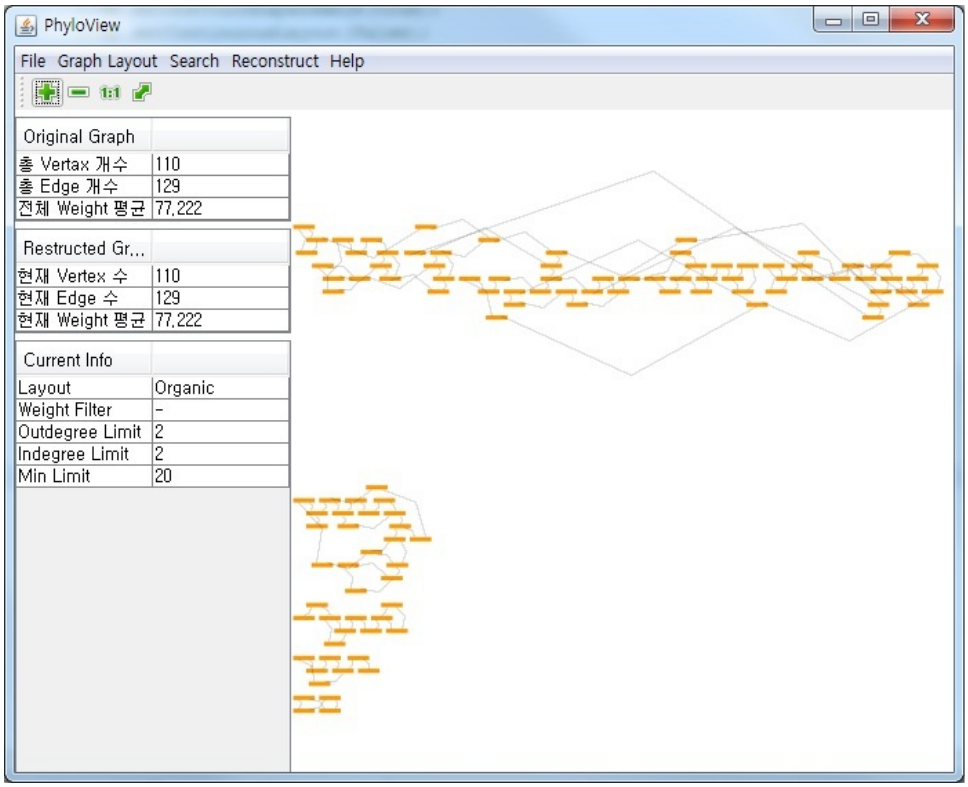

그림 1. 새로 구성한 PhyloView의 화면. 기존 버전에서 존재하는 대부분의 메뉴가 복원되어 있으나 검색 기능과 그래프 재구성 기능은 메뉴만 추가되어 있을 뿐 그 기능을 아직 완전히 복원하지는 못했다. Fit Content 기능을 추가하였으며, Zoom 기능을 개선하였다. 레이아웃 측면에서는 Compact Orthogonal Layout을 추가하고 Random Layout을 삭제하였다.

그림 2는 다양한 레이아웃을 적용하였을 때의 실행 화면이다.

4 실험

새 PhyloView의 성능을 측정하기 위하여 DeVAC Manager에서 생성한 대용량의 dpg 파일로부터 그래프를 정상적으로 그려내는지를 점검하고, 데이터 개수에 따른 초기 로딩 속도를 측정하는 실험을 진행하였다.

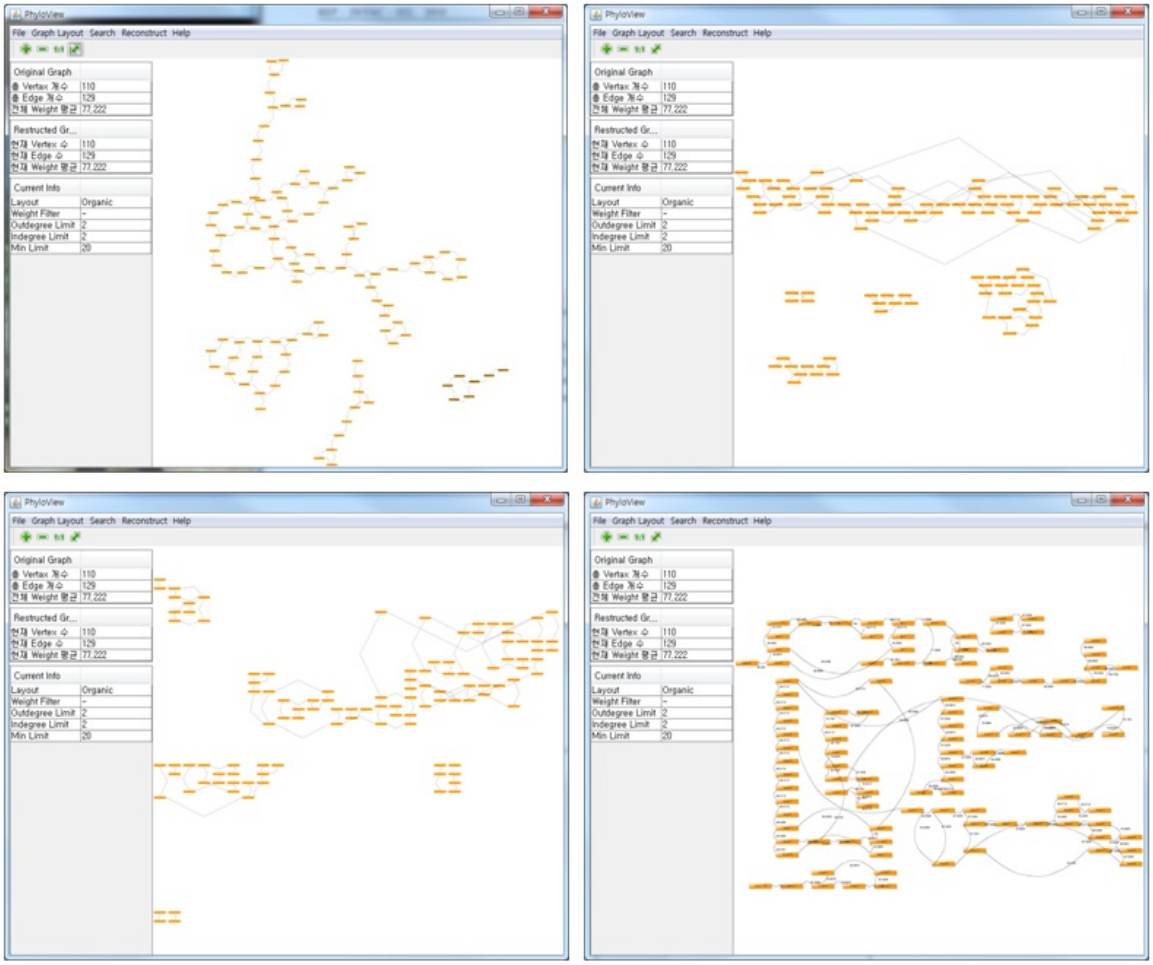

그림 2. 동일한 데이터에 대해 다양한 레이아웃을 적용한 실행 화면

4.1 실험 데이터 및 실험 방법

실험 데이터는 임의로 생성한 표절 데이터 4개와 DeVAC을 이용한 사용자의 실제 데이터 4개를 사용하였다. 각 데이터에 존재하는 vertex와 egde 개수는 표 3와 같다. 실험은 8개 데이터를 PhyloView가 읽어서 화면이 완전히 표시될 때까지의 시간을 측정하고, 그래프의 정상 표시 여부를 확인한 후, 레이아웃을 변경하는데 걸리는 시간을 측정하는 방식으로 진행하였다.

4.2 실험 결과

측정 결과는 표 4와 같다. 로딩 시간은 vertex의 개수보다 edge의 개수에 더욱 크게 영향을 받는 것으로 나타났다. 반면 그래프 레이아웃 변경 시간은 vertex의 개수와 edge의 개수 모두에 영향을 받는 것으로 보였으며, 특히 edge가 500개를 초과할 경우 레이아웃 변경 시간이 급격히 증가하여 사용자가 이용하기 불편한 수준(10초 이상)이 되는 것으로 나타났다. 즉 vertex와 edge의 개수는 최대 500개 정도로 제한하여야 할 것으로 보인다. 이전 버전에서 각 200개 정도로 제한되었던 것에 비해서는 yFiles의 최적화가 많이 진행된 것으로 판단된다.

No.	Vertex 개수	Edge 개수
1	160	100
2	320	200
3	500	499
4	1,000	999
5	182	234
6	325	1236
7	599	5418
8	618	10,412

표 3. 입력 데이터의 vertex와 edge 개수. 1번에서 4번 데이터는 임의로 생성한 실험 데이터이며, 5번에서 8번 데이터는 DeVAC에서 사용된 실제 데이터이다.

No.	로딩 시간(sec.)	레이아웃 변경 시간(sec.)	그래프 정상 표시 여부
1	1.30	0.30	O
2	1.71	0.33	O
3	2.12	3.50	O
4	2.32	10.42	O
5	1.45	2.13	O
6	3.48	15.23	O
7	80.30	342.15	O
8	200.59	436.02	O

표 4. 8개의 데이터에 대한 로딩 시간 및 정상 표시 여부, 레이아웃 변경 시간 측정 결과. vertex의 개수는 1000개 이상으로 많아져도 로딩 시간이 크게 늘어나지 않는데 비해, edge의 개수는 로딩 시간에 큰 영향을 미친다. 레이아웃 변경에 걸리는 시간은 vertex와 edge의 두 인자 모두에 큰 영향을 받는 것으로 보인다. edge의 개수가 500 $\sim$ 1000개가 넘어가면 처리 시간이 5 $\sim$ 10초를 넘기게 되어 실제 사용자가 활용하기에는 무리가 있을 것으로 보인다. 적절한 수의 vertex 및 edge 제한이 필요하다. 모든 테스트 데이터에 대해 정상적인 그래프를 출력하는 것을 확인할 수 있었다.

5 추후 계획

현재까지 개발한 PhyloView는 개발 목표의 달성도는 60% 정도이다. 추후 계획은 크게 세 부분으로 나눌 수 있다. 첫째, 기존 PhyloView에 존재하던 기능 중 아직 복원되지 못한 기능들의 완전한 복원이다. 여기에는 그래프 검색 기능, 그래프 재구성 기능과 좌측 정보 패널의 일부 정보 표시 등이 포함된다. 둘째, PhyloView 프로젝트에 대한 문서화가 필요하다. 개발이 완료되었을 때 유지 보수가 용이하도록 하여야 하고, 이를 위해서는 프로젝트 전체에 대한 문서화 작업이 반드시 필요하다. 또한

사용자 메뉴얼을 만들어 사용자가 이용하기 쉽도록 구성하여야 한다. 사용자 메뉴얼은 WinCHM[3] 이라는 도움말 생성 프로그램을 사용할 예정이며, 프로젝트 관리 및 문서화 작업에 사용할 도구는 조금 더 고려할 필요가 있다. 셋째, 새로운 기능의 추가이다. 윗절에서도 언급했지만 소규모 컴포넌트에 대한 자동 grouping을 통한 공간 효율성 제고, vertex 및 edge 개수 제한치 대폭 증가, cycle 추적 및 자동 삭제 기능, 자동 길이 조절 혹은 mouseover 등을 통한 효과적인 labeling 등 PhyloView에는 개선해야 할 기능이 많이 존재한다. 이를 하나씩 수정해나가는 작업이 필요하다.

6 결론

본 보고서에서는 DeVAC Manager의 서브 모듈인 PhyloView의 유지 보수를 위해 yFiles의 새로운 버전을 사용하여 프로그램 전체를 다시 작성하였다. 우선 기존 PhyloView에 존재하는 기능 중 그래프 보이기, 레이아웃, 그래프 정보 창 등 핵심적인 기능들을 복원하였으며, 사용자 편의를 고려해 Fit Content, 마우스 휠을 이용한 줌 기능 등을 추가하였다. yFiles의 새 버전을 사용한 결과, 기존 버전에 비해 그래프 처리속도가 증가하여 실질적으로 vertax와 edge 모두에 200개 정도로 제한이 걸려 있던 것을 500개 정도로 늘릴 수 있다는 것을 확인하였다. 추후 그래프 탐색, 재구성 등 기존 PhyloView의 모든 기능을 복원하고 시스템을 모듈화한 후 사용자 편의성을 고려한 기능들을 추가할 예정이다. 개발 이후의 유지보수에 대한 부분을 고려하여 개발 과정 전체를 문서화하여 유지할 필요가 있으며, 사용자 메뉴얼도 추가할 예정이다.

References

1. 박선영, “문서 진화 과정 추적을 위한 phyloview 개발,” Tech. Rep., GA Lab., 2010.
2. yWorks, “yfiles,” http://www.yworks.com/en/index.html.
3. Softany, “Winchm,” http://www.softany.com/winchm/.