

소프트웨어 엔지니어링 교육을 위한 데이터 마이닝

동국대학교 | 최은만*

1. 서론

소프트웨어를 개발하는 과정은 개발 작업만이 아니라 품질 보증 작업, 관리 작업 등 다양한 활동들로 이루어진다. 이러한 다양한 작업을 수행하는 과정에 기록되어 쌓이는 엔지니어링 데이터도 많다. 예를 들면 각종 프로세스의 측정 데이터[1], 원시코드, 버그, 설계 등 프로덕트에 대한 이력 및 측정 데이터들이다. 이렇게 개발하는 과정에 쌓이는 많은 엔지니어링 데이터를 보다 적극적으로 이용하려는 마이닝 기법[2-4]들이 연구되고 있다. 특히 측정 모델을 세우기 어려운 버그 예측을 코드 변경 작업의 히스토리 데이터를 이용하여 한다든지 텍스트 기반의 각종 커뮤니케이션 데이터를 마이닝하여 관리나 계획 작업의 의사결정에 이용하려는 연구가 주목받고 있다.

데이터 마이닝 기법은 교육에도 적극 이용되고 있다. 학습과 강의, 평가 등 교육에서 발생하는 여러 가지 데이터를 마이닝하여 교육의 현황을 분석하여 보여주거나 학습 성과를 예측하고 피교육자의 학습이탈 행동을 탐지하거나 코스웨어 계획이나 관리에도 적용[5-7]되고 있다. 예를 들면 온라인 강의에서 학습 성향을 클러스터링 하여 맞춤형 코스 개발에 적용하려는 시도[8]도 있다.

양질의 소프트웨어를 적은 비용으로 개발하려는 여러 가지 노력 중 가장 기본이 되는 것은 역시 교육이다. 소프트웨어 개발을 단지 프로그래밍과 구현으로 보고 프로세스와 설계 기술을 충분히 트레이닝 받지 못한 경우 계속 진화하여 사용되지 못하고 일회적인 S/W가 될 가능성이 크기 때문이다.

S/W를 개발하는 엔지니어를 키우는 교육은 설계, 구현, 테스트, 프로젝트 관리 기법 등 원리만을 가르치는 것이 아니라 사용자, 개발자, 관련 당사자들이 협력하여 소프트웨어를 만들 때 적용할 수 있어야 교육성과

가 크다. 따라서 S/W 엔지니어링을 교육할 때는 원리를 적용할 수 있고 협업이 가능한 실제와 유사한 캡스톤 프로젝트를 경험하게 된다.

프로젝트를 통하여 소프트웨어 엔지니어링을 교육할 때 가장 어려운 부분은 협업의 진행 상황을 정확히 파악하여 멘토링하는 일이다. 산출물을 리뷰하고 커멘트하는 방법은 대부분 작업이 완성된 후에야 이루어질 수 있다. 하지만 프로젝트 협업에 다양한 도구를 도입하고 여기에 쌓인 데이터를 마이닝하면 그룹 작업에 대한 추세를 볼 수 있고 각 팀의 작업 패턴과 특징 등을 파악할 수 있다.

이 논문에서는 소프트웨어 엔지니어링 교육에서 피교육자가 보다 효과적으로 협업할 수 있고 개인 작업의 모니터링과 그룹 작업의 문제점을 예측하기 위한 여러 가지 마이닝 기법을 소개하고 연구 동향을 정리한다.

2장에서는 소프트웨어 엔지니어링 교육 데이터의 마이닝 목적과 배경을 설명하고 연구 동향을 분석한다. 3장은 SE 교육 데이터에 대하여 4장은 데이터 마이닝 기법 및 그 결과를 소개하고 5장에 결론과 제언들을 기술한다.

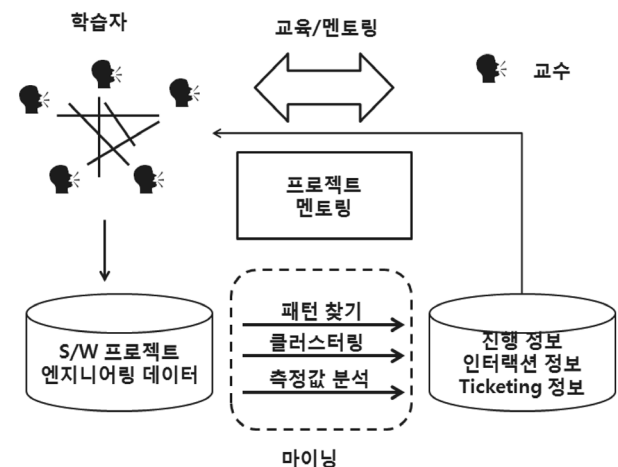


그림 1 SE 교육 데이터 마이닝 개요

* 종신회원

2. 데이터 마이닝 목적 및 관련 연구

소프트웨어 엔지니어를 위한 교육에 프로젝트 관리 시스템, 이슈트래킹 시스템, 위키, 버전 관리 도구, e-러닝 등 협업을 위한 도구를 도입하면 여러 가지 다양한 교육적 데이터를 얻을 수 있다. 문제는 이러한 대규모 데이터를 어떻게 하면 효과적으로 마이닝하여 교육과 학습을 위하여 의미 있는 패턴을 찾아 낼 것인가 하는 것이다. 즉 이 논문에서는 소프트웨어 공학 교육에서 핵심이 되는 그룹 작업 스킬을 교육하고 학습하는데 초점을 두고 있고 여기에 데이터 마이닝 기법이 어떻게 응용되는지를 기술한다.

주어진 시간에 복잡한 작업을 완성하기 위하여 팀 구성원의 결합된 노력은 필수적이다. 하지만 프로젝트 관리 또는 개발 과정을 멘토링하는 입장에서는 프로젝트를 수행하는 구성원의 다양하게 결합된 작업을 일일이 추적하고 추세를 파악하는 일은 쉽지 않다. 따라서 산출물의 공동 저작을 위하여 Wiki 도구, 작업을 정의하고 배정, 확인하는 Trac의 티켓팅 시스템, 버전 관리 기능이 있는 소프트웨어 리파지토리 등을 도입하고 온라인 협업을 위한 환경을 제공한다면 복합적인 노력들에 대한 데이터가 잘 쌓이고 여기에서 원하는 정보를 마이닝할 수 있다.

2.1 SE 데이터 마이닝의 목적

SE 교육을 위한 데이터 마이닝은 크게 세 가지 목적이 있다. 첫째는 그룹 멤버의 작업에 대한 개괄적 진행 추세를 파악하는 것이다. 교수는 이런 정보를 마이닝하여 프로젝트 진행 상황을 알 수 있고 잠재적인 문제를 파악하여 효과적인 해결방안을 제시할 수 있다. 또한 학습자는 자신의 작업을 스스로 모니터링하여 효과적으로 기여하였는지 주어진 역할을 만족시켰는지 알 수 있다.

둘째 목적은 프로젝트를 잘 진행하는 그룹과 그렇지 못한 그룹의 패턴을 파악하는 것이다. 과거의 데이터를 이용하여 개인 및 그룹 별로 우수 및 취약 그룹의 작업 패턴을 파악하여 문제를 예측할 수 있다. 비효율적인 패턴이 발견된다면 적시에 피드백 줄 수 있고 해결책을 제시하여 그룹의 모자라는 스킬을 교육할 수도 있다.

셋째 목적은 성과가 좋은 우수한 그룹은 온라인 협업 도구를 어떻게 효과적으로 사용하는지 이해하는 것이다. 이를 통하여 온라인 협업 환경을 맞춤화하여 SE 학습효과를 높일 수 있기 때문이다.

2.2 관련 연구

일반적으로 협업이 성공하려면 5가지 요소 즉 리더십, 성과 모니터링, 백업, 적응, 팀 중심[9]이 이루어져야 한다. 여기에 추가로 세 가지 메커니즘, 멘탈 모델 공유, 상호 신뢰, 밀접한 커뮤니케이션이 뒷받침 되어야 한다.

Talavera와 Guadoso[10]는 협업을 위한 상호작용을 분석하기 위하여 데이터 마이닝을 적용한다. 웹 게시판이나 전자메일, 채팅 등에서 자료를 수집하여 협업을 분석하고 클러스터링과 통계 지표[11]를 기반으로 하여 유용한 패턴이나 사실을 추출하여 협업 프로파일에 기록하였다. Prata[11]는 프로젝트 회의에서의 대화 내용을 분석하여 팀 프로젝트 내의 갈등이나 문제를 찾아내는 기계 학습 모델을 개발하였다. Soller는 프로젝트 팀의 공유된 지식을 분석하기 위하여 DECREE[12]라는 시스템을 도입하고 학습자가 팀 작업을 위하여 제안을 작성하고 합의를 도출할 때까지 수정하고 개선하게 하여 이를 분석하였다.

대부분의 연구는 학습자의 그룹 활동에 대한 분석 모델을 제시하면서 새로운 도구와 인터페이스를 강제하고 있다. 즉 일반적인 협업 도구를 사용하는 실제 환경에서는 적용하기 어려운 점이 많다. 이 논문에서는 학습자들이 일반적으로 알고 있는 도구와 환경을 사용하여 하고 이를 미리링하는 기법으로 각 개인의 작업과 상호 협력 작업을 비주얼화 하는 방법을 설명한다.

3. SE 교육 데이터

일반적으로 S/W 개발 프로젝트에서 발생하는 엔지니어링 데이터는 표 1과 같이 크게 네 가지로 나눌 수 있다. 먼저 팀 멤버들 사이에 협력을 위한 커뮤니케이션 데이터들이 있다. 그룹으로 작업하는 과정에 의견

표 1 S/W 엔지니어링 데이터

S/W 엔지니어링 데이터	데이터 사례	데이터 보관 도구
팀 커뮤니케이션	의견교환, 이슈해결, 작업 방법	Wiki, 메일링 시스템
프로세스 및 작업 관리 데이터	작업 정의, 작업 할당, 작업 수행	Trac(ticket)
설계 데이터	사용 사례, 클래스 다이어그램, 아키텍처 구조도 등	UML 도구
원시 코드 및 관리 데이터	구문트리, 실행 모델, 버전 제어 데이터	Eclipse CVS, Subversion

교환이나 이슈 해결을 위한 논의 또는 작업 지시 등이 여기에 속하며 최근에는 메일링 시스템이나 Wiki를 이용하여 협업으로 문서를 작성하며 공유한다.

프로젝트 협업에서 각 작업을 정의하고 멤버에게 할당하여 수행하게 하고 그 결과들을 데이터로 모은 것이 프로세스 및 작업 관리 데이터들이다. 현업에서는 프로젝트 프로세스 관리를 위하여 WBS, 진척도를 측정하기 위한 자료들이다. 하지만 교육 환경에서는 프로세스를 일일이 정의하고 이를 할당 및 관리하는 작업은 프로젝트의 오버헤드가 너무 크게 된다. 따라서 Trac과 같은 가벼운 도구로 ticket를 관리한다. ticket은 팀이 작업할 일 생기면 생성하고 이를 수행할 멤버에게 할당한다. 이러한 ticket들이 쌓이면 중요한 마이닝 대상 데이터들이 된다.

설계 데이터는 요구 분석과 설계 과정에 중간 산출물들이다. 즉 요구 분석 작업의 결과인 기능이나 기능 외적인 요구들이며 주로 use case나 성능 조건이 여기에 해당된다. 상세한 설계 작업 후에는 아키텍처나 모듈 안의 설계 데이터들이 작성된다. 설계 데이터들은 주로 인터페이스나 데이터 엔티티에 대한 것으로 설계 대상 시스템의 아키텍처적인 특성들을 마이닝 할 수 있다.

마지막으로 프로젝트에서 개발한 프로덕트의 핵심인 원시코드와 이를 관리하는 과정에 파생된 데이터가 있다. 대부분 Subversion(SVN)과 같은 버전 관리 시스템에 기록된 원시코드의 변경 작업 데이터이다. 이런 데이터는 원시코드에 대한 작업의 히스토리를 기록한 것으로 마이닝하여 버그를 예측한다든지 버그의 유형을 판단하는 등 유용한 정보를 얻을 수 있다.

S/W 엔지니어링 데이터는 그 유형이 크게 세 가지

타입으로 나뉜다. 첫째는 텍스트 형태로 메일이나 위키로 커뮤니케이션한 데이터나 요구 분석의 기능 리스트, 코드의 커멘트, 문서 같은 것이 여기에 해당된다. 둘째 유형은 그래프 형태로 사용 사례 다이어그램, 아키텍처 설계나 프로그램 실행 모델, 버전 진화 등이 여기에 해당된다. 마지막 유형은 순서(sequence)적인 의미가 중요한 데이터들이 있다. 프로그램의 실행되는 경로, 정적인 추적 경로 등이 여기에 해당된다.

4. 마이닝 방법과 결과

Hassan[13]에 의하면 S/W 엔지니어링 데이터의 마이닝은 정적인 기록을 살아있는 데이터로 만들기 위하여 감추어진 패턴이나 트렌드를 파악하는 것으로 정의한다. 예를 들면 프로그램 변경 히스토리를 마이닝하여 예상하지 못한 문제점이나 의존관계 또는 버그까지도 찾아낼 수 있다. 또한 프로그램에서 사용한 API 패턴을 마이닝하여 재사용 패턴을 찾을 수도 있고 재사용을 가이드 할 수도 있다.

그림 2에 표시된 일반적인 SE 데이터에 대한 마이닝에는 자주 출현하는 아이템을 파악하거나 연관 규칙을 찾아내는 방법, 클러스터링, 분류 기법 등을 사용한다. 관련이 많은 메소드 호출이나 버그 등은 연관 규칙으로 찾는 방법으로 마이닝 할 수 있다. 분류는 주로 머신 러닝 기법을 이용하는 방법으로 일정한 학습 데이터 집합을 준비하여 분류 알고리즘을 세운 후 분류 모델을 이용하여 새 데이터에 적용하여 분류한다.

SE 교육을 위한 마이닝은 그림 2에 있는 실제 프로젝트에서 작업한 중요한 결과물에 대한 마이닝 보다는 협업이나 개인 작업을 가시화 하기 위한 마이닝과 프로세스 또는 작업 패턴에 대한 마이닝이 필요하다.

표 2 SE 데이터의 유형과 마이닝 방법

S/W 엔지니어링 데이터 유형	사례	마이닝 방법
텍스트	버그 리포트, 문서, 원시코드 주석, 메일	텍스트 매칭 클러스터링 Classification
그래프	모듈 호출 그래프, 데이터 의존도, 클래스 다이어그램, 사용 사례 다이어그램	서브그래프 마이닝, 그래프 매칭, 클러스터링, Classification
시퀀스	실행/정적 추적 순서	부분순서 매칭, 연관 규칙 마이닝 클러스터링 Classification

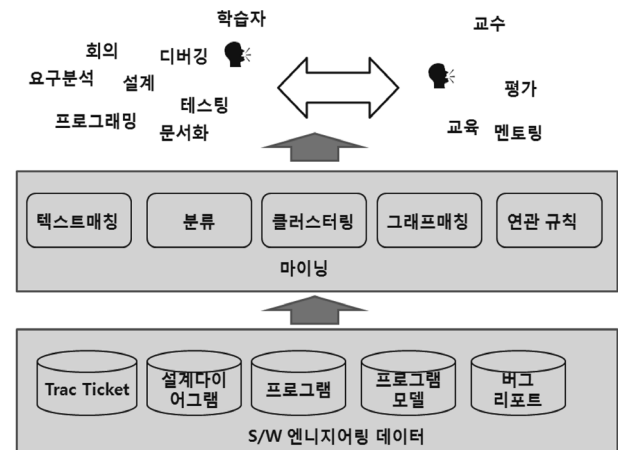


그림 2 SE 데이터 마이닝 기법

이 장에서는 여러 가지 마이닝 방법 중 SE 교육에 관련된 미러링, 패턴 매칭, 클러스터링 방법에 대하여 소개한다.

4.1 미러링 비주얼화

프로젝트를 위한 그룹 작업에서 학습자와 교수에게 당면한 중요한 도전과제는 큰 그림을 보는 것이다. 즉 프로젝트 멤버들이 효과적으로 작업하고 있는지 맡겨진 역할을 충실히 하고 있는지 확인하는 것이다. 협동 작업을 위하여 Trac과 같은 도구를 사용하면 그룹 작업 과정을 추적할 수 있고 문제가 발생했을 때 조기에 파악하여 멘토링할 수 있다.

하지만 프로젝트 엔지니어링 데이터들은 원시적이며 분량이 많고 너무 상세하여 추세를 판단할 수 없다. 따라서 마이닝하거나 추세를 파악할 수 있도록 가시적인 요약은 하여야 하는데 이를 미러링 비주얼화[14]라고 부른다. 대규모의 그룹 작업 데이터를 마이닝하여 팀이 효과적으로 작업하고 있는지 판단할 수 있는 추세를 비주얼화 하는 것이다.

우선 학생의 작업을 파악하기 위하여 리퍼지토리에 쌓인 Wiki나 작업 계획 ticket, 버전제어 작업 이력을 분석한다. 또한 팀 구성원 사이의 의사교환 흐름을 정리하는 작업도 필요하다. 리더의 경우 조금 다른 작업 패턴을 보이는데 계획에 초점을 두기 때문에 ticket을 발행하는 작업이 많게 된다. 만일 리더에게 이런 패턴이 없다면 그 역할에 문제가 있는 것으로 파악할 수 있다.

그림 3은 프로젝트에 참여하고 있는 각 개인의 시간별 기여도와 활동을 나타낸 것이다. 수직축이 시간의 흐름이며 각 개인의 활동이 하나의 축에 마치 가지가 달려 자라나는 것처럼(이를 Wattle tree라 함) 비주얼화 된다. 원은 Wiki를 작성하는 기여도(노란색)와 버그(SVN 관련 작업) 수정에 대한 기여도(주황색)를 나타낸 것으로 크기가 클수록 기여도가 높다. 부채꼴로 나타낸 것은 작업 수행을 위하여 ticket을 발행하여 작업을 할당한 것(왼쪽 진한 부채꼴)과 할당된 작업을 수행한 것(오른쪽 연한 부채꼴)이다. 진한 부채꼴이 많이 남겨질수록 미룬 일이 많다는 의미이다.

맨 왼쪽 학습자의 경우는 ticket 발행이 많은 패턴을 보이므로 리더로 추측할 수 있고 왼쪽의 두 학습자가 이 프로젝트에서 활발히 작업한 핵심 멤버임을 알 수 있다. 왼쪽에서 네 번째 학습자의 경우 개발에 큰 기여를 한 것을 알 수 있고 다섯 번째 학습자는 27일 이후에 활동이 중단되었다.

이와 같이 Wattle 트리를 통하여 프로젝트에 얼마나 참여하고 있는지를 한 눈에 보고 문제가 있다면 적절한 조치를 미리 취할 수 있다.

Wattle 다이어그램의 경우 누가 누구에게 ticket을 발행하여 작업을 의뢰했는지는 표시되지 않는다. 팀 구성원 사이의 상호작용을 잘 표현하려면 노드는 구성원, 링크는 멤버 사이의 의견 교환(Wiki작업 의뢰, SVN 지시 및 수행)이 표현된 인터랙션 그래프가 필요하다.

4.2 패턴 마이닝

프로젝트 수행에서 작업의 순서는 매우 중요한 요소이다. 예를 들어 리더가 Wiki 페이지에 있는 요구분석을 수정하고 이를 프로그램에 반영시키기 위하여 Ticket을 생성하였다면 그룹 멤버가 할당되고 Wiki 문서와 프로그램을 수정하는 작업이 후속되었을 것이다. 이런 순서의 작업 패턴이 어떤 프로젝트의 완성을 위하여 정해진 회수 이상 반복되어야 한다. 표 3과 같은 작업 순서의 패턴을 찾는 것이 패턴 마이닝의 핵심이다.

작업 순서 패턴의 마이닝은 세 가지 단계로 구성된다. 먼저 각 프로젝트 그룹에 대한 작업 이벤트의 리스트를 준비한 후 너무 긴 작업은 여러 개의 의미 있는 일련의 작업으로 분할한다. 즉 같은 프로그램의 자원에 대한 작업만을 추려내고 그룹 작업 이벤트들은 일정한 최소 단위 시간별로 잘라낸다. 마지막은 데이터 마이닝을 용이하게 하기 위하여 작업 이벤트를 인코딩한다. 준비된 작업 이벤트의 순서에 대한 패턴을 찾아 잘 하는 그룹과 못하는 그룹에 대한 작업 특징들을 찾아낼 수 있다.

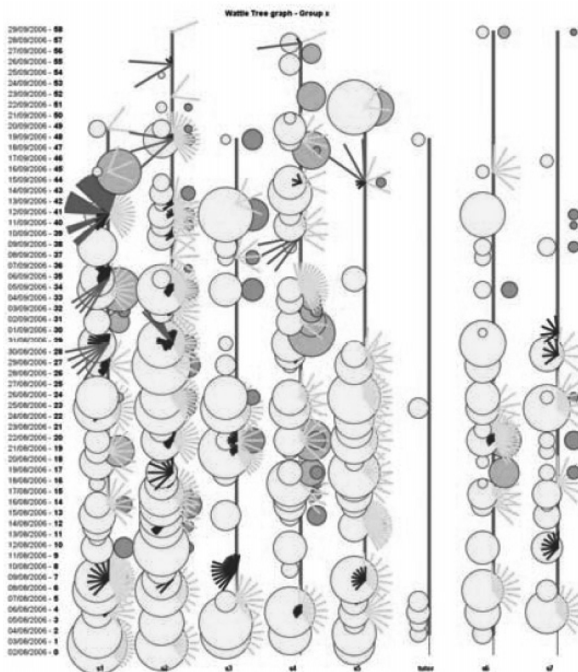


그림 3 Wattle 다이어그램 [14]

표 3 SE 데이터의 유형과 마이닝 방법

잘하는 그룹의 작업 패턴	설명
SVN과 Wiki 이벤트가 번갈아 자주 출현	SVN에 대한 Wiki 문서가 잘 반영되고 있음
SVN과 ticket이 번갈아 자주 출현	Ticket 업데이트 후 SVN 실행
Ticket이 Wiki보다 많이 사용되고 계속 발행됨	Ticket이 Wiki보다 더 작업 중심이므로 실제 한 작업을 반영하는 지표임
다수의 연속된 SVN 이벤트	그룹 리파지토리에 반영된 작업
리더가 ticket을 많이 발행하고 다른 멤버는 접수	효과적인 리더십
못하는 그룹의 작업 패턴	설명
Wiki와 Ticket 이벤트가 자주 순서가 바뀌고 교체됨	작업이 프로그램 개발로 직접 이루어지지 못함
SVN 이벤트가 적음	실제 원시코드 작업으로 이어지지 못함
리더가 ticket을 발행하고 자신이 close	리더가 효과적인 리더십을 발휘하지 못함

4.3 클러스터링

SE 교육 중 프로젝트에서 학습자 개인 또는 그룹 활동의 특징을 클러스터링 기법으로 찾아낼 수 있다[15]. 클러스터링은 여러 가지 데이터의 속성을 사용하여 동질의 그룹으로 묶는 방법이다.

클러스터링 알고리즘의 성능에 영향을 주는 두 가지 요소는 속성 선택과 클러스터의 수이다. 속성은 ticket 당 이벤트의 수, 하루에 발행되는 ticket의 수, Wiki 페이지 당 ticket의 수, Wiki 페이지의 잔존 기간, Wiki 편집 라인 수, SVN 작업 날짜 등이다.

프로젝트 진행을 잘하는 그룹은 ticket의 수락과 업데이트, Wiki 페이지 편집, SVN 작업이 자주 일어난다. 또 다른 클러스터링 그룹은 ticket의 발행은 자주 되지만 이벤트가 활발하지 않고 Wiki 페이지는 편집이 잦지만 추가되는 분량은 많지 않은 특성을 보인다. 프로젝트 진행이 잘 안되는 그룹은 ticket의 수도 적고 SVN 작업도 잘 이루어지지 않는 특징으로 클러스터링 된다.

학습자 개인별 특성으로 클러스터링 할 수도 있다. 팀 리더는 주로 ticket 발행이 많고 Wiki 작업도 많으나 SVN 작업은 많지 않다. 반대로 ticket 발행이나 Wiki 작업은 많지 않으나 SVN 작업이 많은 특성을 보이는 학습자 그룹도 있다. 이는 리더가 아니며 Trac 위주의 개발에 집중한 결과이다. Ticket 발행도 적고 Wiki 작업도 많지 않으며 SVN 작업도 적은 학습자 그룹은 프로젝트 참여가 저조한 것이다. 또 다른 클러스터링 그

룹은 ticket도 적당히 있고 Wiki 작업도 있으나 SVN 작업이 적고 있다하더라도 짧은 기간에만 이루어지는 그룹이 있다.

5. 결론

S/W 엔지니어링 교육에서 학습자의 그룹 협업 데이터를 마이닝하는 방법을 살펴보았다. 협업과정에 파생되는 여러 가지 빅 데이터의 마이닝을 통하여 교육과 학습에 도움이 되는 정보를 추출할 수 있다. 협업을 위한 플랫폼인 Trac을 사용하면 각종 설계 결과물을 Wiki로 편집하고 ticket 시스템과 S/W 형상관리 도구 SVN을 사용하여 작업 관리를 할 수 있다.

교육과 학습에 도움이 되는 정보는 미러링 비주얼화, 패턴 마이닝, 클러스터링, 세 가지 마이닝 방법으로 추출할 수 있다. 마이닝 결과를 통하여 각 그룹의 문제를 조기에 발견할 수 있고 모니터링 할 수 있어 교수와 학습자 모두 효과적인 S/W 개발 작업에 도움이 된다.

SE 데이터들이 주로 개발 현장에서 취입되고 마이닝되는 연구가 많으나 교육 데이터로서 사용된다면 프로젝트에서의 협업 패턴, 개인별 작업 특성, 프로세스 진행 상황 파악 등 여러 가지 효과를 얻을 수 있다.

교육 데이터 마이닝 과정에 제한점이 있다. 하나는 Trac 데이터가 그룹 작업의 모든 커뮤니케이션을 반영하지 못한다는 점이다. 예를 들어 개인적 대화나 문자, SNS 등을 통한 개별 의사교환은 반영되지 않는다. 둘째 클러스터링 분석을 위하여 충분한 이벤트가 쌓여야 한다는 점이다. 충분한 데이터가 쌓이지 않는다면 클러스터링의 의미가 뚜렷하지 않아 추가적인 조사 및 확인이 필요할 수 있다.

참고 문헌

- [1] J. McGarry, D. Card, C. Johne, B. Layman, E. Clark, J. Dean, F. Hall, Practical software measurement, Addison-Wesley, 2002.
- [2] H. Kagdi, M. Collard, J. Maletic, "A survey and taxonomy of approaches for mining software repositories in the context of software evolution", Journal of Software Maintenance, Vol. 19, No. 2, pp.77-131, 2007.
- [3] M. Marri, S. Thummalapenta, T. Xie, "Improving software quality via code searching and mining", Proceedings of 1st International Workshop on Search-Driven Development - Users, Infrastructure, Tools and Evaluation(SUITE 2009), pp.33-36, 2009.
- [4] A. Hassan, T. Xie, "Mining software engineering data",

- Proceedings of the 32th ACM/IEEE International Conference on Software Engineering(ICSE 2010), pp.503-504, 2010.
- [5] E. Andersen, S. Gulwani, Z. Popovic, “A trace-based framework for analyzing and synthesizing educational progressions”, Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp.773-782, 2013.
- [6] A. Fox, D. Patterson, “Crossing the software education chasm”, Communications of the ACM, Vol. 55, No. 4, pp.44-49, 2012.
- [7] N. Tillmann, J. Halleux, T. Xie, S. Gulwani, J. Bishop, “Teaching and learning programming and software engineering via interactive gaming”, Proceedings of Software Engineering Education Workshop in International Conference on Software Engineering, pp.1117-1126, 2013.
- [8] K. Masters, “A brief guide to understanding MOOCs”, The internet journal of medical education, Vol. 1, No. 2, pp.346-360, 2011.
- [9] Christopher Cheong, “From group-based learning to cooperative learning: A metacognitive approach to project-based group supervision”, Informing Science: the International Journal of an Emerging Transdiscipline, Volume 13, 2010.
- [10] Jason L. Frand, “The information-Age mindset: Changes in students and implications for higher education”, EDUCAUSE Review, 35(5), pp.15-24, September-October 2000.
- [11] Kay Reimann, K. Yacef, Mirroring of group activity to support learning as participation, International Conference on Artificial Intelligence in Education, Los Angeles, IOS Press, 2007.
- [12] B. Barros, M. Verdejo, “Analysing student interaction processes in order to improve collaboration The DEGREE approach”, Proceedings of International Journal of Artificial Intelligence in Education, pp.221-241, 2000.
- [13] A. Hassan, “The road ahead for mining software repositories”, Frontiers of Software Maintenance, pp. 48-57, 2008.
- [14] Judy Kay, Irena Koprinska, Kalina Yacef, “Educational data mining to support group work in software development projects”, Handbook of Educational Data Mining, Edited by Crist Romero, Sebasti Ventura, Mykola Pechenizkiy and Ryan S. J. D. Baker, CRC Press, pp 173-177, 2010.
- [15] A. Anya, J. Boticario, “A data mining approach to reveal representative collaboration indicators in open collaboration frameworks”, Proceeding of Educational Data Mining Conference, pp.210-219, 2009.

약 력



최은만

1982 동국대학교 전산학과(학사)
 1985 한국과학기술원 전산학과(공학석사)
 1993 일리노이 공대 전산학과(공학박사)
 1985~1988 한국표준과학연구원 연구원
 1988~1989 데이콤 주임연구원
 1998~2004 한국정보과학회 소프트웨어공학

연구회 운영위원

2002 카네기멜론대학 소프트웨어공학 과정 연수
 2000, 2007 콜로라도 주립대 전산학과 방문교수
 1993~현재 동국대학교 컴퓨터공학과 교수
 관심분야 : 소프트웨어 공학 교육, 소프트웨어 테스트, 프로세스와
 매트릭, Program Comprehension, AOP

E-mail : emchoi@dgu.ac.kr