

사회연결망 분석을 응용한 비즈니스 프로세스 자원의 업무 적합성 판단 기법

이승훈*, 김건우*, 최중민*

*한양대학교 컴퓨터공학과

e-mail:teyes99@naver.com, kgwhsg@gmail.com, jmchoi@hanyang.ac.kr

Task Conformance Checking Technique for the Business Process Resource applying Social Network Analysis

Seung-Hoon Lee*, Gun-Woo Kim*, Joongmin Choi*

*Dept. of Computer Science and Engineering, Hanyang University

요 약

최근 비즈니스 프로세스가 주목을 받으며 기업에서는 비즈니스 프로세스의 개선을 중요하게 생각하기 시작했다. 기업은 비즈니스 프로세스를 개선하기 위해 BPR(Business Process Re-engineering)등의 혁신 프로그램을 도입했지만 그 효과나 성공률이 상당히 미흡했다. 성공적인 프로세스의 혁신을 위해서는 프로세스 모델의 개편과 함께 자원 및 자원의 관리 방식의 개편도 필수적이다. 그렇지만 지금까지의 프로세스 혁신은 프로세스 모델 중심으로만 이루어졌으며 자원 및 자원의 관리 방식의 개편에 대한 연구는 미흡한 실정이다. 이런 문제점을 해결하기 위하여 최근 실제 발생한 비즈니스 프로세스 인스턴스를 기록한 프로세스 로그를 기반으로 효율적인 자원할당 및 통제방안에 대한 연구가 진행되고 있다. 본 논문에서는 사회 연결망을 응용하여 각 자원간의 연결을 고려하여 비즈니스 프로세스의 업무에 적합한 자원을 판단할 수 있는 방법을 제시하였다.

1. 서론

비즈니스 프로세스란 어떤 조직의 목표를 달성하기 위해 진행되는 시작과 끝이 있는 하나 이상의 활동 및 절차를 의미한다[1]. 기업은 업무의 수행 효율성 증대 및 비용의 절감 등을 목적으로 비즈니스 프로세스에 많은 관심을 가지고 있다. 이러한 비즈니스 프로세스를 다루기 위한 기술로 BPM(Business Process management)이나 워크플로우(Workflow)와 같은 연구가 활발히 진행되고 있다. 하지만 초기부터 완벽한 비즈니스 프로세스를 설계한다는 것은 매우 어려운 일이기 때문에 프로세스 마이닝(Process Mining) 기술을 이용한 비즈니스 프로세스 재설계(Business Process Re-engineering)에 대한 연구가 활발히 진행되고 있다. 프로세스 마이닝은 실제 실행되는 프로세스에서 발생하는 로그 정보들을 수집하여 프로세스를 향상시키는 데 유용한 정보를 얻는데 목적이 있다. 비즈니스 프로세스를 향상시키기 위해서는 프로세스 모델의 향상과 함께 자원 및 자원의 관리 방식의 개편도 필수적이다[2,3]. 그렇지만 지금까지의 프로세스 마이닝 연구는 프로세스 모델 관점에서의 연구가 주로 이루어졌으며 기업의 자원 관리의 측면에 대한 연구는 부족한 실정이다. 최근 실제 발생한 비즈니스 프로세스 인스턴스의 실행경로, 업무의 수, 각 업무의 수행시간, 각 업무를 수행하는 자원 등의 특성을 고려한 효율적인 자원할당 및 관리 등의 통제방안에 대한 연구가 진행되고 있다[2,3]. 본 논문에서는

비즈니스 프로세스 설계 이후 실제 수행을 위한 시스템 통제 분야에 초점을 맞춰 사회연결망을 응용하여 제한된 자원들을 각 자원간의 연결을 고려하여 비즈니스 프로세스의 업무에 효율적으로 배치될 수 있도록 프로세스 로그를 이용하여 각 자원간의 관계를 도출하고 이를 이용해 각 업무에 자원을 할당하는 알고리즘을 제시하고자 한다. 본 논문의 구성은 다음과 같다. 2장에서는 관련연구인 프로세스 마이닝과 사회연결망 분석에 대해 살펴보고 3장에서는 사회연결망을 응용한 자원의 업무 적합도를 판단하는 방법에 대해 살펴본다. 4장에서는 3장에서 설명한 방법을 프로세스 로그 예제를 통하여 수행해보며 5장에서는 논문의 결론을 맺는다.

2. 관련연구

2.1 프로세스 마이닝(Process Mining)

프로세스 마이닝이란 기업의 정보시스템에서 프로세스가 실행되며 기록되는 프로세스 로그를 기반으로 수행된다. 프로세스 마이닝은 프로세스 로그 안에서 의미 있는 정보를 얻어 비즈니스 프로세스를 향상시키는데 이용된다. 프로세스 마이닝은 크게 3가지 기법으로 구성된다. 첫째로는 정보 시스템의 실행결과가 기록된 프로세스 로그 정보를 추출하여 프로세스 모델을 찾아내는 프로세스 모델 발견(Discovery)기법이 있으며, 두 번째로는 발견된 프로세

스 모델과 프로세스 로그 정보를 비교하여 기업에서 의도했던 프로세스의 목표에 적합한지를 평가하는 적합성 기법(Conformance Checking), 마지막으로 발견된 프로세스 모델을 변경하여 적합성을 상승시키는 확장(Extension) 기법이 있다[4,5].

2.2 사회연결망 분석(Social Network Analysis)

사회는 다수의 사람이나 그룹의 연결과 관계로 이루어져 있으며 한 사회에 사는 사람은 다른 사람들과의 연결을 피할 수 없다. 또한 연결이나 단절의 방식 또는 구조는 다양한 사회현상을 만들어내기 때문에 사회연결망을 분석하는 것은 조직 사회에 있어 중요한 일이다[6]. 사회연결망을 분석하는 기준은 사회연상을 만들어 내는 각 자원의 특성, 사회의 구조, 각 자원과 사회와의 관계의 3가지로 나눌 수 있다. 사회연결망은 노드와 링크로 이루어진 그래프 구조로 나타낼 수 있다. 비즈니스 프로세스에서 사회연결망을 분석한다는 것은 프로세스 로그로부터 자원이나 업무 사이의 관계 및 연결의 특성과 구조를 분석하는 것을 말한다. 이를 통하여 자원 사이에 작업이 어떻게 전달되는지, 자원들이 얼마나 자주 같은 작업에 참여하는지, 또는 얼마나 비슷한 작업을 하고 있는지 등을 알아낼 수 있다[3,7]. 이 정보들은 기업 내에서 조직 구성의 문제점을 발견하고 조직을 재구성하는데 참고할 수 있으며 비즈니스 프로세스를 개선하는데 사용될 수 있다.

3. 사회연결망을 응용한 자원의 업무 적합도 판단

본 논문에서는 비즈니스 프로세스의 업무를 수행하는 자원들의 업무 적합도를 다음과 같이 판단한다. 먼저 한 업무를 여러 번 수행한 자원은 그 업무에 대한 이해도가 높다. 그리고 서로 연결된 업무에서 먼저 실행된 업무를 X라 하고 다음에 실행된 업무를 Y라 했을 때 X를 John이 수행하고 Y를 Sue가 수행했을 경우 John과 Sue가 X와 Y를 서로 같이 수행한 횟수가 많다면 서로의 업무에 대한 이해도가 높고 서로를 잘 이해하고 있다고 가정한다.

3.1 프로세스 로그(Process Log)

프로세스 로그란 비즈니스 프로세스의 실행결과가 기록된 데이터이며 프로세스 마이닝을 수행하기 위한 근간이 된다. 프로세스 로그는 아직까지 표준으로 정해진 형식이 없으나 대부분의 정보시스템에서 Case_ID, Task, Performer, Time_Stamp 등의 정보를 프로세스 로그로 제공하고 있다. 본 논문에서는 프로세스 로그의 인스턴스(Instance)로 프로세스의 Case들을 구분하기 위한 Case_ID, Case내에서 수행된 업무를 구분하기 위한 Task_ID, 업무를 수행한 사람을 기록하는 Performer, 업무가 시작된 시간과 종료된 시간을 기록하는 AssignedTime=Start, AssignedTime=Finish를 사용하였다. 아래의 <표1>은 본 논문에서 사용하는 프로세스 로그

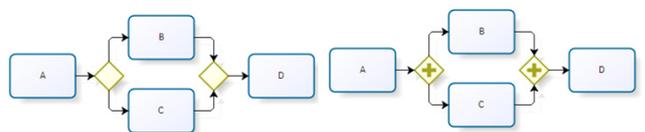
구조이다.

Case_ID	Task_ID	Performer	AssignedTime =Start	AssignedTime =Finish
...

<표 1> 기본적인 프로세스 로그 구조

3.2 게이트웨이 타입(Gateway Type) 구분

프로세스 모델에는 순차적인 업무 전달 외에 분기, 루프 등의 업무 전달방법이 존재한다. 따라서 프로세스 모델을 고려하지 않고 단순히 프로세스 로그만을 이용하여 자원간의 연결을 도출할 경우 잘못된 결과를 도출할 가능성이 있다. 그렇기 때문에 프로세스 로그를 이용하여 모델의 분기 방법을 결정하는 게이트웨이 타입을 구분하는 법이 필요하다. 먼저 XOR Gateway(그림 1.1)는 A업무를 수행 후 B나 C업무 중 하나를 수행한 후 D업무를 수행한다. AND Gateway(그림 1.2)는 A업무를 수행한 후 B와 C업무를 모두 수행하고 D업무 앞에서 동기화된 후 D업무가 수행된다. 프로세스 로그는 프로세스가 수행된 순서대로 기록되기 때문에 XOR Gateway의 경우는 A->B->D, A->C->D와 같은 패턴으로 기록될 것이기 때문에 업무의 전달 경로를 명확히 알 수 있지만 AND Gateway의 경우는 A->B->C->D, A->C->B->D와 같은 패턴으로 기록되기 때문에 실제 B업무가 C업무로 또는 C업무가 B업무로 전달되지 않음에도 불구하고 업무가 전달되는 것으로 분석될 수 있다. 이를 구분하기 위해 <표 2>와 같이 프로세스 로그에서 Case Pattern에서 중복을 없애 Common Case Pattern을 구하고 이를 2-sequence Pattern으로 나타낸다. 이제 2-sequence Pattern으로 나타낸 각 패턴에서 서로 연결된 2-sequence Pattern의 지지도와 신뢰도를 구한다. 여기서 지지도는 전체 case에서 X라는 업무와 Y라는 업무가 연결된 case 수를 전체 case수로 나눈 값 (i.e., $s(X \rightarrow Y) = \sum(X \rightarrow Y) / N$ (전체 case 수))이며 신뢰도는 업무 X와 업무 Y의 연결 수를 업무 X가 포함된 case 수로 나눈 값 (i.e., $c(X \rightarrow Y) = \sum(X \rightarrow Y) / \sum(X)$ (업무 X가 포함된 case 수))을 의미한다. 이 예제에서는 B->D와 C->D의 지지도와 신뢰도 값이 동일할 경우 AND Gateway로 판별한다.



(좌)(그림 1.1) XOR Gateway를 이용한 연결
(우)(그림 1.2) AND Gateway를 이용한 연결

Case Pattern		Common Case Pattern		2-sequence Pattern		
case1	A->B->C->D	case1	A->B->C->D	A->B	B->C	C->D
case2	A->C->B->D	case2	A->C->B->D	A->C	C->B	B->D
...	...					

<표 2> Gateway 구별을 위해 2-sequence Pattern으로 변환

3.3 Performer-Task 행렬

프로세스 로그의 Task_ID와 Performer 정보를 이용하여 어떤 자원이 어떤 업무를 몇 번 수행하였는지 알 수 있다. 이를 이용하여 어떤 자원이 어떤 업무에 얼마나 익숙한지를 유추할 수 있다.

3.4 Transfer Of Work(Performer-Performer) 행렬

프로세스 로그의 Performer 정보를 이용하여 Performer간 업무전달관계와 그 횟수를 나타내는 Transfer Of Work(Performer-Performer) 행렬을 구할 수 있다. 해당 행렬은 A업무를 John이 수행하고 B업무를 Sue가 수행하며 A와 B업무가 연결되어 있을 때 John은 Sue에게 '업무를 전달했다'라고 한다.

3.5 Transfer Of Work(Task-Task) 행렬

프로세스 로그의 Task-ID 정보를 이용하여 Task간 업무전달관계와 그 횟수를 나타내는 Transfer Of Work(Task-Task) 행렬을 구할 수 있다. 해당 행렬은 A업무가 B업무가 연결되어있을 때 'A업무는 B업무로 업무를 전달한다'라고 한다. 이 행렬을 구할 때 Gateway Type을 고려하여야한다.

3.6 업무 적합도 판단 알고리즘

자원의 업무 적합도는 자원이 해당 업무를 얼마나 많이 수행했으며 다음 업무를 수행하는 자원과의 관계가 얼마나 많이 일어나는지를 고려하여 구한다.

1. 자원이 해당 업무를 얼마나 많이 수행했는지는 Performer-Task 행렬을 통하여 알 수 있다.
2. 다음 업무를 수행하는 자원과의 관계를 구하기 위하여 현재 업무를 X, 다음 업무를 Y라고 하면 조직구조도를 참조하여 X와 Y를 수행할 수 있는 자원을 모두 구한다. 다음으로 Performer-Task 행렬에서 X와 Y를 수행할 수 있는 자원들이 실제로 수행한 업무들을 구해 축소된 행렬인 Performer'-Task' 행렬을 구한다.

3. Performer'-Task'의 전치행렬인 Task'-Performer' 행렬을 구하여 Performer'-Task'행렬과 곱을 구한다. 구해진 행렬은 Task의 업무간의 관계를 고려한 자원간의 관계를 나타낸 Performer'-Performer'행렬이다.

4. 다음으로 자원들끼리 서로 업무를 함께 한 빈도수를 이용하기 위하여 자원간의 Transfer Of Work 행렬을 고려해야 한다. Transfer Of Work(Performer-Performer) 행렬에서 X와 Y를 수행할 수 있는 자원을 모두 구한 후 X와 Y가 실제 수행한 업무들을 구해 축소된 행렬인 Transfer Of Work(Performer'-Performer')행렬을 구한다.

5. 3에서 구한 Performer'-Performer'행렬과 4에서 구한 Transfer Of Work(Performer'-Performer')행렬을 곱한

다. 이 결과는 자원의 특정업무의 수행 빈도와 자원들이 X업무와 Y업무를 같이 수행한 횟수를 모두 포함한 결과를 얻을 수 있다.

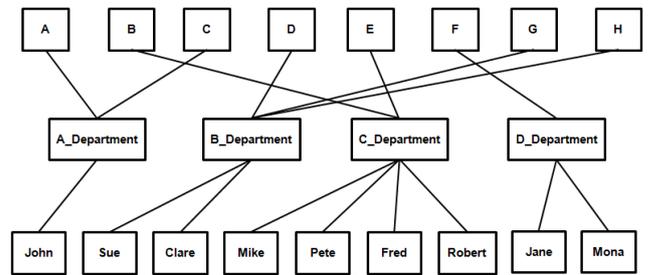
6. 결과 행렬에서 X를 수행할 수 있는 자원과 Y를 수행할 수 있는 자원의 교차점에 있는 값들을 비교하여 랭킹을 구한다.

4. 실험

본 논문에서 제시한 프로세스 로그로부터 자원들 간의 업무 적합성을 판단하는 알고리즘을 실험을 통하여 업무B와 D를 수행하는 자원 살펴보면 다음과 같다. <표 3>은 프로세스가 실행되면서 기록된 프로세스 로그의 Task_ID와 Performer를 Case_ID순으로 정렬한 것이다. (그림 2)는 해당 프로세스의 조직 구조이다. 각 업무를 수행할 수 있는 부서와 부서에 속해있는 자원들을 나타낸다.

System Log								
Case 1	(A, John)	(B, Mike)	(C, John)	(D, Clare)	(E, Pete)	(F, Mona)	(H, Sue)	
Case 2	(A, John)	(C, John)	(B, Pete)	(D, Sue)	(E, Robert)	(F, Mona)	(H, Clare)	
Case 3	(A, John)	(C, John)	(B, Fred)	(D, Sue)	(G, Clare)	(H, Clare)		
Case 4	(A, John)	(C, John)	(B, Mike)	(D, Sue)	(E, Robert)	(F, Jane)	(H, Sue)	
Case 5	(A, John)	(B, Mike)	(C, John)	(D, Sue)	(G, Clare)	(H, Clare)		
Case 6	(A, John)	(B, Fred)	(C, John)	(D, Sue)	(E, Mike)	(F, Mona)	(H, Clare)	
Case 7	(A, John)	(C, John)	(B, Mike)	(D, Clare)	(E, Robert)	(F, Jane)	(H, Sue)	
Case 8	(A, John)	(B, Mike)	(C, John)	(D, Sue)	(E, Pete)	(F, Jane)	(H, Sue)	
Case 9	(A, John)	(C, John)	(B, Pete)	(D, Sue)	(E, Robert)	(F, Jane)	(H, Clare)	
Case 10	(A, John)	(B, Mike)	(C, John)	(D, Clare)	(G, Clare)	(H, Sue)		
Case 11	(A, John)	(B, Fred)	(C, John)	(D, Sue)	(E, Fred)	(F, Mona)	(H, Clare)	
Case 12	(A, John)	(B, Fred)	(C, John)	(D, Clare)	(G, Sue)	(H, Sue)		

<표 3> 프로세스 로그의 Task_ID와 Performer를 Case_ID순으로 정렬



(그림 2) 조직 구조도

1. <표 4>와 같이 프로세스 로그로부터 Performer-Task행렬을 구한다.

Performer-Task								
	A	B	C	D	E	F	G	H
John	12	0	12	0	0	0	0	0
Sue	0	0	0	8	0	0	1	6
Mike	0	6	0	0	1	0	0	0
Pete	0	2	0	0	2	0	0	0
Jane	0	0	0	0	0	4	0	0
Clare	0	0	0	4	0	0	3	6
Fred	0	4	0	0	1	0	0	0
Robert	0	0	0	0	4	0	0	0
Mona	0	0	0	0	0	4	0	0

<표 4> Performer-Task 행렬

2. <표 5>와 같이 <표 4>와 (그림 2)를 이용하여 Performer'-Task'행렬을 구한다.

3. <표 6>과 같이 <표 5>와 <표 5>의 전치행렬을 곱하여 Performer'-Performer' 행렬을 구한다.

B, D Task의 수행자와 관계있는 Performer-Task 행렬					
	B	D	E	G	H
Sue	0	8	0	1	6
Mike	6	0	1	0	0
Pete	2	0	2	0	0
Clare	0	4	0	3	6
Fred	4	0	1	0	0
Robert	0	0	4	0	0

Task간의 관계를 고려한 Performer'-Performer' 행렬						
	Sue	Mike	Pete	Clare	Fred	Robert
Sue	101	0	0	71	0	0
Mike	0	37	14	0	25	4
Pete	0	14	8	0	10	8
Clare	71	0	0	61	0	0
Fred	0	25	10	0	17	4
Robert	0	4	8	0	4	16

(좌)<표 5> Performer'-Task' 행렬

(우)<표 6> Task간의 관계를 고려한 Performer'-Performer' 행렬

4. <표 7>과 같이 프로세스 로그와 Transfer Of Work(Performer-Performer)행렬을 이용해서 Transfer Of Work(Performer'-Performer')행렬을 구한다.

Transfer Of Work(Performer-Performer)									
	John	Sue	Mike	Pete	Jane	Clare	Fred	Robert	Mona
John	12	8	6	2	0	4	4	0	0
Sue	0	1	1	1	0	2	1	3	0
Mike	0	3	0	0	0	3	0	0	1
Pete	0	2	0	0	1	0	0	0	1
Jane	0	3	0	0	0	1	0	0	0
Clare	0	2	0	0	0	3	0	1	1
Fred	0	3	0	0	0	1	0	0	1
Robert	0	0	0	0	3	0	0	0	1
Mona	0	1	0	0	0	3	0	0	0

Transfer Of Work(Performer'-Performer')						
	Sue	Mike	Pete	Clare	Fred	Robert
Sue	1	1	1	2	1	3
Mike	3	0	0	3	0	0
Pete	2	0	0	0	0	0
Clare	2	0	0	3	0	1
Fred	3	0	0	1	0	0
Robert	0	0	0	0	0	0

<표 7> Transfer Of Work(Performer'-Performer')행렬

5. 3에서 구한 <표 6>과 4에서 구한 <표 7>행렬을 곱하여 <표 8>과같이 자원들이 B업무수행 빈도와 D업무수행 빈도, 그리고 자원들이 B업무와 D업무를 같이 수행한 횟수를 모두 포함한 결과를 얻을 수 있다.

B, D Task 수행에 적합한 Ranking						
	Sue	Mike	Pete	Clare	Fred	Robert
Sue	243	101	101	415	101	374
Mike	214	0	0	136	0	0
Pete	88	0	0	52	0	0
Clare	193	71	71	325	71	274
Fred	146	0	0	92	0	0
Robert	40	0	0	16	0	0

<표 8> B, D Task 수행에 적합한 Ranking

6. <표 9>와 같이 결과 행렬에서 B를 수행할 수 있는 자원과 Y를 수행할 수 있는 자원의 교차점에 있는 값들을 비교하여 Ranking을 얻을 수 있다. 해당 결과는 Mike가 B 업무를 수행한 횟수가 가장 많고 Sue가 D업무를 수행한 횟수가 가장 많으며 Mike와 Sue가 서로 업무를 같이 한 경우가 3번으로 가장 많기 때문에 B 업무는 Mike, D 업무는 Sue가 처리하는 경우가 가장 적합한 것을 알 수 있다. Mike와 Clare, Fred와 Sue 또한 서로 업무를 같이 한 경우가 3번으로 같지만 Clare는 Sue보다 D업무를 수행한 횟수가 적으며 Fred는 Mike보다 B업무를 수행한 횟수가 적어 Mike와 Sue보다 낮은 Ranking을 가지게 된다.

Performer가 Task를 수행한 횟수						
	Sue	Mike	Pete	Clare	Fred	Robert
B	0	6	2	0	4	0
D	8	0	0	4	0	0

(B, D)Task를 Performer Set이 수행한 횟수		
	(B, D)	횟수
Mike	Sue	3
Mike	Clare	3
Pete	Sue	2
Pete	Clare	0
Fred	Sue	3
Fred	Clare	1
Robert	Sue	0
Robert	Clare	0

1	Mike	Sue	214
2	Fred	Sue	146
3	Mike	Clare	136
4	Fred	Clare	92
5	Pete	Sue	88
6	Pete	Clare	52
7	Robert	Sue	40
8	Robert	Clare	16

<표 9> B, D Task 수행에 적합한 Ranking

5. 결론

본 논문에서는 프로세스 마이닝 기법과 사회연결망 분석 방법을 응용하여 비즈니스 프로세스의 각 업무에 적합한 자원의 Ranking을 정하는 방법을 제시하였다. 논문에서 프로세스 로그의 기본적인 정보 중 Case_ID, Task_ID, Performer정보만을 사용하여 Ranking을 정하였다. 그렇지만 프로세스 로그에는 AssignedTime_Start와 AssignedTime_Finish의 정보도 존재하기 때문에 향후 과제에서는 이를 이용하여 업무적합성을 판단하는 방법에 대한 연구를 진행할 예정이다.

참고문헌

- [1] 이승훈, 김건우, 주진웅, 이동훈, 김학수, 손진현, “로그 기반의 프로세스 마이닝을 통한 BPMN 프로세스 모델 추출”, 한국정보과학회 2009 한국컴퓨터종합학술대회 논문집 제 36권 제1호(B), p.137, 2009
- [2] 송민석, W.M.P. van der Aalst, 최인준, “비즈니스 프로세스 수행자들의 Social Network Mining에 대한 연구”, 한국경영과학회 2004 춘계 학술대회는논문집, p.531, 2004
- [3] 최인준, 송민석, 김광명, 이용혁, “프로세스 모델에서 도출한 조직간 사회관계에 대한 분석과 조직 재설계”, Journal of the Korean Institute of Industrial Engineers, Vol.33, No.1, p.11, 2007
- [4] A. Tiwari, C.J.Turner, “A review of business process mining: state-of-the-art and future trends”, Business Process Management Journal, Vol.14, No1, p.5, 2008
- [5] van der Aalst, Weijter A.J.M.M, “Process Mining: a research agenda”, Comput. Ind. Vol.53, No.3, 2004
- [6] Tomislav HERNAS, “Process-based Organization Design Model: Theoretical Review and Model Conceptualization”, University of Zagreb, Faculty of economics and business Working paper series, No.08-06, 2008
- [7] W.M.P van der Aalst, HAJO A. Reijers, 송민석, “Discovering Social Networks from Event Logs”, Computer Supported Cooperative Work, Vol.14, Issue. 6, p549, 2005