ISSN 1598-0170 (Print)
ISSN 2287-1136 (Online)
http://www.jksii.or.kr

대규모 워크플로우 소셜 네트워크의 추정 근접 중심도 랭킹 알고리즘 성능 분석☆

Performance Analysis of an Estimated Closeness Centrality Ranking Algorithm in Large-Scale Workflow-supported Social Networks

김 자 원[1] 안 현[2] 김 광 훈[2*]
Jawon Kim Hyun Ahn Kwanghoon Kim

요 약

본 논문에서는 대규모 워크플로우 기반 소셜 네트워크를 위한 추정 근접 중심도 랭킹 알고리즘을 구현하고, 그에 대한 성능 분석을 수행한다. 기존의 근접 중심도 분석 방법은 특정 노드와 다른 모든 노드들 간의 최단거리를 구하므로 네트워크의 크기가 커짐에 따라 근접 중심도 분석 시간이 기하급수적으로 증가하는 문제점을 가진다. 이로 인해 대규모 소셜 네트워크의 근접 중심도 랭킹 과정에도 계산시간 문제를 가진다. 이러한 문제점을 개선하고자 본 논문에서는 추정기법을 활용한 근접 중심도 랭킹 알고리즘을 구현하며 기존 알고리즘과의 성능 분석을 수행한다. 이는 약 50%의 계산 시간 단축 결과를 보여주었다.

☞ 주제어 : 워크플로우 기반 소셜 네트워크, 추정 근접 중심도, 랭킹 알고리즘

ABSTRACT

This paper implements an estimated closeness centrality ranking algorithm in large-scale workflow-supported social networks and performance analyzes of the algorithm. Existing algorithm has a time complexity problem which is increasing performance time by network size. This problem also causes ranking process in large-scale workflow-supported social networks. To solve such problems, this paper conducts comparison analysis on the existing algorithm and estimated results by applying estimated-driven RankCCWSSN(Rank Closeness Centrality Workflow-supported Social Network). The RankCCWSSN algorithm proved its time-efficiency in a procedure about 50% decrease.

☞ keyword : Workflow-supported Social Network, Estimated Closeness Centrality, Ranking Algorithm

1. 서 론

최근 여러 기업들 사이에 협업, 협력이 빈번하게 발생됨에 따라, 이전에 비해 워크플로우 기반 소셜 네트워크(Workflow-Supported Social Network, WSSN)의 크기가 대규모로 증가하고 있다. 워크플로우 소셜 네트워크는 업무에 대한 업무 수행자들 간의 관계를 나타낸 그래프로, 업무 수행자를 정점으로 나타내고 그들 간의 관계를 간선으로 표현한 그래프이다. 이러한 워크플로우 소셜 네트워크에 기존 소셜 네트워크에 연결 중심도(Degree Centrality), 근접 중심도(Closeness Centrality), 사이 중심도(Betweenness Centrality)을 적용할 수 있다. 이는 인사 평가, 업무 배치 등 인적 자원 관리 측면에서 중요한 지표로 활용될 수 있다. 이는 직무성과와도 밀접하게 연관되어 기업의 직무 효율 상승을 기대할 수 있다. 하지만 근접 중심도의 경우 시간 복잡성의 문제로 인해 네트워크의 크기가 커질수록 분석 시간이 기하급수적으로 증가하는 한계점을 가진다. 이렇기 때문에 수행자간의 근접 중심도를 기반으로 하는 랭킹 과정 또한 알고리즘 계산시간에 문제점을 가진다. 본 논문에서는 이를 해결하기 위해 추정 기법을 활용한 근접 중심도 랭킹 알고리즘을 구현하고 성능 분석한다.

본 논문의 구성은 다음과 같다. 2장은 관련연구로서

[1] Dept. of Computer Engineering, Sungkyul University., Anyang-si, Gyeonggi-do, 430-742, Korea.
[2] Dept. of Computer Science, Kyonggi University., Suwon-si, Gyeonggi-do, 443-760, Korea.
* Corresponding author (kwang@kgu.ac.kr)
[Received 2 February 2015, Reviewed 9 February 2015,, Accepted 25 March 2015]
☆ 본 논문은 2014년도 인터넷정보학회 추계학술발표대회우수논문 추천에 따라 확장 및 수정된 논문임
☆ 본 연구는 한국연구재단에서 지원하는 2012년도 기초연구사업 (No. 2012006971)의 연구수행으로 인한 결과물임.

워크플로우 기반 소셜 네트워크의 이전 분석 연구에 대해 살펴본다. 또 소셜 네트워크 분석에 있어 랭킹에 관한 연구에 대해 알아본다. 3장에서는 본 저자에 의해서 제안된 대규모 워크플로우 기반 소셜 네트워크를 위한 추정 근접 중심도 랭킹알고리즘(RankCCWSSN)에 대해 알아보고 구현 방법에 대해 설명한다. 4장에서는 알고리즘을 고정된 전체 네트워크 크기에서 모집단의 크기에 따른 알고리즘 수행시간과 네트워크 크기 증가에 따른 알고리즘 수행시간에 대해 실험하여 보고 이를 기존 수식의 계산시간과 비교하여본다. 마지막으로 5장에서는 결론과 향후 연구과제에 대해 설명한다.

2. 관련연구

기존 워크플로우 프로세스에 관하여 화두가 되고 있는 연구는 워크플로우 프로세스 모델과 그에 대한 로그 데이터를 활용하여 다양한 정보를 추출 하는 것이다. W. van der Aaslst[2]에서는 이벤트 로그에 따른 프로세스 마이닝 기법에 대해 말하고, 워크플로우 프로세스가 실행됨에 따라 발생하는 로그 데이터 중 업무 수행자, 즉 사람 중심의 네트워크를 발견하는 것을 설명하고 있다. K. Kim[3]은 워크플로우 기반의 소셜 네트워크에 대해 정의하고 정보제어넷(ICN) 워크플로우 모델을 기반으로 워크플로우 소셜 네트워크를 발견하는 알고리즘을 제안했다. 알고리즘을 제안함에 따라 기존 워크플로우 모델로부터 소셜 네트워크를 발견하고, 이를 통해 기존 소셜 네트워크 분석에 사용되었던 다양한 분석 기법(연결 중심도, 근접 중심도, 사이 중심도 등)[4]을 워크플로우 기반 소셜 네트워크에 적용할 수 있게 되었다. S, Park[5, 6]에서는 비연결 워크플로우와 연결 워크플로우 기반의 소셜 네트워크 근접 중심도 분석 알고리즘을 제안하였고, 이에 대한 프레임워크를 정의하였다. 이로 인해 기업적인 측면에서의 인적자원관리에 대해 다양하게 활용될 수 있는 지표를 얻을 수 있다. 한편 소셜 네트워크 분석에 있어 랭킹에 대한 연구는 H. Park [7]에서와 같이 네트워크 속 가장 영향력 있는 사람을 찾는 것이 화두가 되고 있다. 하지만, 기업 간 협업과 협력을 통해 조직의 규모는 증가하게 되어 대규모의 모습을 띄고 이에 대한 근접 중심도 분석은 계산 시간 증가로 인해 문제점을 갖는다. 따라서 본 논문에서는 대규모 네트워크의 적합한 계산시간을 보여주는 RankCCWSSN알고리즘에 대해 구현하고 성능 분석하여, 대규모 네트워크에서의 적합성을 증명한다.

3. 추정 근접 중심도 랭킹 알고리즘 설계 및 구현

본 장에서는 대규모 소셜 네트워크 분석에 적합한 추정 기법을 활용하여 대규모 워크플로우 기반 소셜 네트워크를 위한 근접 중심도 랭킹 알고리즘을 구현한다. 구현 과정은 워크플로우 소셜 네트워크에서 추정 근접 중심도를 분석하고 그를 바탕으로 정확한 근접 중심도를 구해 랭킹 하는 것이다.

3.1 추정 근접 중심도 랭킹 알고리즘

워크플로우 기반의 소셜 네트워크는 워크플로우 프로세스가 실행됨에 따라 발생하는 로그 데이터 중 업무 수행자 중심의 네트워크, 즉 사람 중심의 네트워크를 발견하는 것을 말한다. 이러한 워크플로우 기반 소셜 네트워크의 근접 중심도를 분석하는 것은 $O(n^2)$에 해당하는 시간 복잡도를 가져 대규모 네트워크에서 사용하기 부적합함을 보인다. 이로 인해 수행자의 근접 중심도 값을 랭킹하는 과정 또한 수행시간이 지수 적으로 증가한다.

그림 1은 대규모 워크플로우 기반 소셜 네트워크를 위한 근접 중심도 랭킹 알고리즘(Rank Closeness Centrality Workflow Supported Social Network)이다. 그림 1의 알고리즘은 하나의 업무(프로세스)가 실행됨에 따라 업무를 수행하는 업무 수행자 사이의 연결 관계를 나타낸 수행자 중심의 네트워크 모델을 입력으로 받는다. 전체 네트워크 중 랜덤하게 추출한 노드들을 모집단으로 구하고, 추정기법을 이용하여 모집단에 속한 노드와 전체 노드들 사이의 근접 중심도를 추정한다. 모집단을 기반으로 분석한 근접 중심도 근사치 값을 정렬하고 k번째까지의 노드에 대해 기존 근접 중심도 수식을 적용하여 k번째까지의 노드와 전체 노드들 간의 정확한 근접 중심도를 도출한 뒤 정렬한다. 알고리즘의 결과물인 출력은 전체 네트워크에서의 근접 중심도를 구하여 순위매긴 리스트이다. 기존 근접 중심도를 구하는 방법은 네트워크 내의 전체 노드와 전체 노드간의 근접 중심도 값을 구하는 방식이지만, 이는 알고리즘 계산시간의 문제로 인해 대규모 네트워크에 적합하지 않음을 보여준다.

RANKCCWSSN(k) /* 추정 근접 중심도 기반 랭킹 알고리즘 */
Input Social Network, $G(\varphi, a)$ /* 수행자 중심의 네트워크 모델*/
Output Ranked List /*랭킹된 리스트*/

-k 순위 매길 정점의 개수
-$\boldsymbol{P}$ 업무 수행자의 집합
-l 샘플 수행자의 수
-$\widehat{\boldsymbol{C}}$ 추정된 근접 중심도의 집합
-$\boldsymbol{C}$ 측정된 근접 중심도의 집합
-$\boldsymbol{S} = \{\boldsymbol{s_1}, \cdots, \boldsymbol{s_l}\}$ 샘플링된 정점의 집합
-g 수행자들의 수

Begin Procedure
Initialize
$\boldsymbol{C} = \emptyset$
$\widehat{\boldsymbol{C}} = \emptyset$
$\boldsymbol{S} = \boldsymbol{RANDOM}(P, l)$

For $(\forall \boldsymbol{\varphi_i} \in \boldsymbol{P})$ **Do**
Begin /*근접 중심도를 추정한다.*/
$\widehat{\boldsymbol{C}}_{\boldsymbol{i}} = \frac{1}{\sum_{j=1}^{l} \frac{g d(S_j, \varphi_i)}{l(n-1)}}$
$\widehat{\boldsymbol{C}} = \widehat{\boldsymbol{C}} \cup \widehat{\boldsymbol{C}}_{\boldsymbol{i}}$
End
Sort $\widehat{\boldsymbol{C}}$
For $(\widehat{\boldsymbol{C}}_{\boldsymbol{i}} \in \widehat{\boldsymbol{C}}; \boldsymbol{1} \le \boldsymbol{i} \le \boldsymbol{k})$ **Do**
Begin /*정확한 근접 중심도를 구한다.*/
$\boldsymbol{C}_{\boldsymbol{i}} = \frac{1}{\left[\sum_{j=1}^{g} d(\varphi_i, \varphi_j)\right]} \ (\boldsymbol{i} \neq \boldsymbol{j})$
$\boldsymbol{C} = \boldsymbol{C} \cup \boldsymbol{C}_{\boldsymbol{i}}$
End
Sort $\boldsymbol{C}$
RETURN $\boldsymbol{C}$;
End Procedure

(그림 1) 추정 근접 중심도 랭킹 알고리즘
(Figure 1) The Estimated Closeness Centrality Ranking Algorithm

하지만 제안한 알고리즘은 전체 노드들 간의 근접 중심도가 아닌 모집단과 전체 노드 간의 근접 중심도를 구함으로써 효과적으로 알고리즘 수행시간을 단축할 수 있다.

그림 1의 알고리즘 중에서 근접 중심도를 추정하는 과정[8]은 다음 그림 2의 알고리즘과 같다. 기존의 근접 중심도를 구하는 방법은 특정노드가 다른 노드들 간의 최단거리 합을 구하고 그를 역수 취하는 것이다. 이러한 과정에서 노드의 수가 많아질수록 근접 중심도를 구하는 시간이 기하급수적으로 증가 하였고, 이러한 문제점을 개선하기 위하여 그림 2와 같은 근접 중심성 근사치 추정 알고리즘을 제안하였다[8]. 그림 2는 기존의 근접 중심도 알고리즘과는 달리 전체 네트워크 내에서 무작위 방식으로 모집단을 구한 뒤 모집단에 속한 노드와 전체 노드들 사이의 최단 거리 합을 구하고 역수로 하는 것으로 이루어진다.

Algorithm RAND:

1. Let k be the number of iterations needed to obtain the desired error bound.
2. In iteration i, pick vertex v_i uniformly at random from G and solve the SSSP problem with v_i as the source.
3. Let

$$\hat{c}_u = 1/\sum_{i=1}^{k} \frac{n\, d(v_i, u)}{k(n-1)}$$

be the centrality estimator for vertex u.

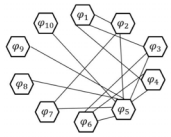
It is not hard to see that, for any k and u, the expected value of $1/\hat{c}_u$ is equal to $1/c_u$.

(그림 2) 추정 알고리즘
(Figure 2) Estimated Algorithm

모집단에 대한 근접 중심도를 분석함으로 대규모 소셜 네트워크의 중심도를 분석할 때의 알고리즘 전체 수행시간이 현저하게 줄어드는 효과를 기대할 수 있다.

3.2 알고리즘 구현

다음은 본 논문에서 제안한 알고리즘을 자바 기반으로 구현한 결과이다. 우선 소셜 네트워크를 수학적으로 분석하기 위해 알고리즘의 입력 부분인 소셜 네트워크를 그림 3과 같이 SocioMatrix형태로 변환한다[5].



	φ_1	φ_2	φ_3	φ_4	φ_5	φ_6	φ_7	φ_8	φ_9	φ_{10}
φ_1	0	1	1	0	0	0	0	0	0	0
φ_2	1	0	0	0	1	0	1	0	0	0
φ_3	1	0	0	0	1	1	0	0	0	0
φ_4	1	0	0	0	1	1	0	0	0	0
φ_5	0	1	1	1	0	1	1	1	1	1
φ_6	0	0	1	1	1	0	0	0	0	0
φ_7	0	1	0	0	1	0	0	0	0	0
φ_8	0	0	0	0	1	0	0	0	0	0
φ_9	0	0	0	0	1	0	0	0	0	0
φ_{10}	0	0	0	0	1	0	0	0	0	0

(그림 3) 워크플로우 소셜 네트워크의 예제의 SocioMatrix
(Figure 3) The SocioMatrix of Workflow Model example

변환된 SocioMatrix를 기반으로 기준이 되는 특정 네트워크의 노드에 수행자를 추가 하는 방식으로 SocioMatrix의 크기를 증가시킨다[9]. 이후 확장된 SocioMatrix에 SSSP(Single Source Shortest-Path)[10]를 적용하여 네트워크 안의 각 노드들 간의 최단거리를 구한다. 각 노드들 간의 최단거리는 그림 1의 알고리즘 수행 시 $d(\varphi_i, \varphi_j)$의 값으로 사용된다. 기존 근접 중심도 수식

의 계산 시간과 본 논문에서 제안한 알고리즘의 계산 시간을 비교하기 위하여 각각 알고리즘의 시작 부분과 종료 부분에 시간을 출력하여 해당 알고리즘의 수행시간을 측정하였다.

RankCCWSSN(Rank Closeness Centrality for Workflow-Supported Social Network)알고리즘을 구현한 결과 다음과 같은 결과 값을 얻을 수 있다.

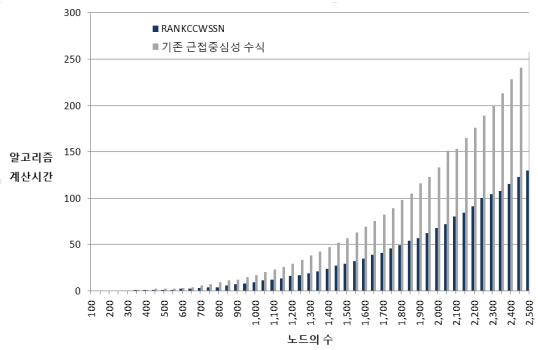
(표 1) 알고리즘의 수행 결과
(Table 1) The result of the execution of the algorithm

랭킹된 리스트		
노드 그룹	순위	근접 중심도 값
Group 1 : ($node_4$, $node_{14}$, $node_{24}$, ..., $node_{994}$)	1 ~ 100	0.001
Group 2 : ($node_5$, $node_{15}$, $node_{25}$, ..., $node_{995}$)	101 ~ 200	0.00066667
Group 3 : ($node_3$, $node_{13}$, $node_{23}$, ..., $node_{993}$)	201 ~300	0.00066667
Group 4 : ($node_2$, $node_{12}$, $node_{22}$, ..., $node_{992}$)	301 ~400	0.00066667
Group 5 : ($node_6$, $node_{16}$, $node_{26}$, ..., $node_{996}$)	401 ~500	0.000625

표 1은 알고리즘 수행 결과 값을 표로 나타낸 것이다. 전체 노드의 수는 1000개이며, 추정 근접 중심도를 구하기 위한 모집단의 개수는 전체 노드 수의 절반인 500개이다. 전체 순위 매긴 노드의 수는 근접 중심성 상위 500개이다. 표 1에서의 노드 그룹은 근 접 중심도 값이 같은 그룹을 의미하며, 이것은 그림 3과 같이 대상이 되는 소셜 네트워크를 행렬로 만들고 하나의 노드에 수행자를 추가하는 방식으로 행렬의 크기를 늘렸기 때문이다. 업무 수행자 각각의 근접 중심도 값을 구할 수 있고, 근접 중심도 값이 같은 업무 수행자들을 그룹핑하여 실제 조직의 부서, 팀과 같은 그룹 단위의 랭킹 또한 가능하다. 알고리즘의 성능 분석은 알고리즘의 시작시간과 종료시간을 출력하여 시작시간에서 종료 시간을 뺀 시간(단위 : 초)으로 수치화 하였다.

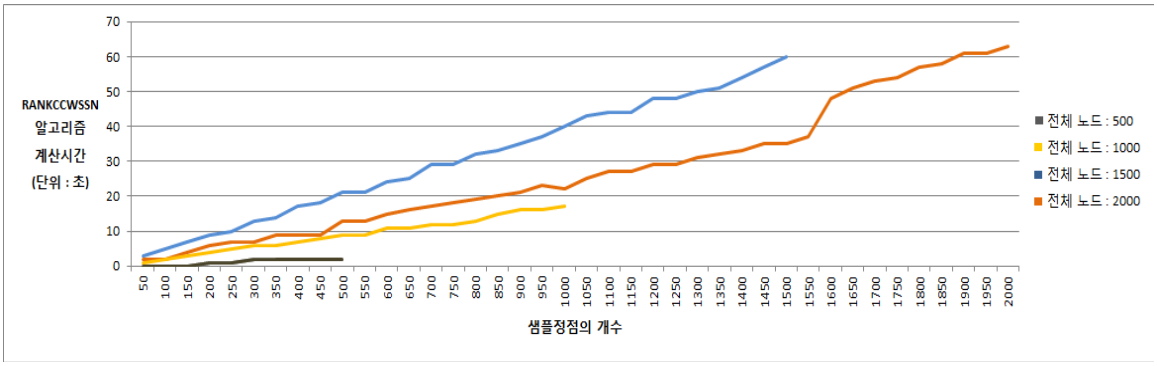
4. 알고리즘 성능 분석

본 장에서는 기존 근접 중심성 수식의 계산 시간과 제안한 알고리즘의 수행시간을 비교하여 제안한 알고리즘의 계산시간이 단축됨을 증명한다. 알고리즘 구현을 통하여 대규모 네트워크에서의 RankCCWSSN알고리즘 및 기존 근접 중심성 수식을 이용한 랭킹 방법의 소요시간을 알 수 있다. 이를 통해 본 논문에서 제안한 알고리즘이 기존 근접 중심성 수식을 이용하였을 때 보다 대규모 네트워크에 더 적합한 계산시간을 증명한다.



(그림 4) 네트워크 크기에 따른 알고리즘 성능
(Figure 4) The performance of the algorithm by network size

그림 4는 RankCCWSSN알고리즘을 사용하여 랭킹을 도출하였을 때와, 기존 근접 중심도 수식을 활용하여 랭킹을 도출하였을 때 알고리즘 계산시간을 나타낸 표이다. 그래프의 가로축은 네트워크 전체 노드의 수를 의미하고, 세로축은 알고리즘 수행시간을 초 단위로 나타낸 것이다. 실험 조건으로 전체 네트워크의 크기는 네트워크내의 노드 수를 100개부터 50개 단위씩 증가해 총 2500개 노드까지의 대규모 네트워크를 설정한다. RankCCWSSN의 경우 모집단의 수를 전체 노드의 50%로 설정하였고, 랭킹 사이즈는 10으로 통일 하였다. 전체 노드의 수가 100 ~ 400인 경우 기존 수식과 RankCCWSSN알고리즘의 수행시간은 일치한다. 하지만 노드 수가 증가함에 따라 기존 근접 중심성 수식을 이용할 때와 RankCCWSSN 알고리즘을 사용하였을 때, 두 수행시간은 절반에 가까운 큰 차이를 보인다. 그림 4의 기울기를 분석하여 보면 기존 근접 중심성 수식을 이용하여 대규모 네트워크의 랭킹을 도출하였을 때에는 세로축에 가까운 기울기를 보여주는 반면, RankCCWSSN 알고리즘을 사용하여 랭킹을 도출하였을 때에는 기울기의 정도가 보다 완만해짐을 알 수 있다. 실제로 2500개의 노드를 가진 네트워크를 분석한 경우 RankCCWSSN 알고리즘을 사용했을 때 기존 수식을 활용한 계산 시간에 비해 128초 앞당겨짐을 확인할 수 있다. 그림 5는 RankCCWSSN알고리즘을 사용하였을 때 모집



(그림 5) 모집단 크기에 따른 알고리즘 성능
(Figure 5) The performance of the algorithm by subset size

단의 크기를 다르게 하여 알고리즘 수행시간을 나타낸 것이다. 그래프의 색에 따라 전체 노드의 크기가 다르고 가로 축은 샘플 정점의 개수, 세로축은 초 단위로 나타낸 알고리즘 수행시간을 의미한다. 모집단의 크기는 50씩 증가해 전체 네트워크의 크기까지 증가하도록 설정했다. 전체 노드의 개수가 500인 경우 모집단의 크기에 따라 2초 정도의 수행시간 차이를 보였지만, 전체 노드 수 증가에 따라 모집단 크기별 수행시간도 증가함을 알 수 있다. 이러한 점들을 미루어 볼 때에 모집단의 크기를 결정하는 것은 알고리즘 수행시간과 밀접하게 연관되어 있다. 모집단의 크기에 따라 알고리즘 수행시간이 비례하는 모습을 볼 수 있다. 따라서 알고리즘을 수행시간을 고려하여 모집단의 크기를 설정하는 것이 알고리즘 수행시간을 단축시키는 중요한 부분이다. 또 그림 5로 알 수 있는 것은 기존 수식을 이용한 근접 중심성 랭킹 방법에선 알고리즘 계산시간이 지수 적으로 증가하는 모습을 띄었지만, RankCCWSSN 알고리즘의 경우 선형에 가까운 증가를 보인다.

5. 결론 및 향후연구

본 논문에서는 대규모 워크플로우 기반 소셜 네트워크를 위한 추정된 근접 중심성 랭킹 방법을 제안했다. 기존 근접 중심성을 계산하는 수식은 네트워크 크기의 증가에 따라 근접 중심성 분석 시간이 기하급수적으로 증가하는 결과를 보여주었다. 이러한 부분은 대규모 네트워크에서의 근접 중심성 수식 적용이 적합하지 않음을 보여주고, 이에 대한 랭킹과정에도 적합하지 못하다.

이러한 문제점을 개선하기 위해 본 논문에선 제안한 알고리즘(RankCCWSSN)에 대해 구현하고 성능분석을 함으로써, 기존 근접 중심성 수식만을 사용하여 랭킹과정을 수행했을 때에 비해 효과적인 알고리즘 계산 시간 단축을 증명하였다. 이러한 결과는 기업 경영 측면인사 평가 지표, 인사 관리 지표 등 인적자원관리에 다양하게 활용될 수 있다. 향후 연구 과제로는 현재 알고리즘 모집단을 전체 네트워크에서 랜덤하게 추출하는 것을 의미 있는 모집단을 구하는 것으로 대체 하여 근접 중심도 분석 가치를 높이는 것에 있다[11]. 모집단에 노드 간 연결 정도를 분석하는 연결중심도, 특정 노드의 연결 정도와 그에 대해 직접적으로 연결되어 있는 노드들의 연결 정도를 기반으로 측정하는 위세 중심도와 같은 조건들을 적용하여본다. 이는 본 논문에서 제안한 알고리즘의 정확도 상승을 기대할 수 있다.

참고문헌(Reference)

[1] Jawon Kim, Hyun Ahn, Hyunah Kim, Minjae Park, Kwanghoon Kim, "Performance Analysis of an Estimated Closeness Centrality based Ranking Algorithm in Large Scale Workflow-supported Social Networks", in Proc. of 2014 Fall Conference of KSII, Korean Society For Internet Information, Vol. 15, no. 2, pp.169-170, 2014.

[2] Wil M. P. Van Der Aalst, Hajo A. Reijers, Minseok Song, "Discovering Social Networks from Event Logs", Computer Supported Cooperative Work, Vol.

14, no. 6, pp.549-593, 2005.
http://dx.doi.org/10.1007/s10606-005-9005-9

[3] Kwanghoon Kim, "A Workflow-based Social Network Intelligence Discovery Algorithm", Journal of Korean Socieity Internet Information, Vol. 13, no. 2, pp.73-86, 2012.
http://dx.doi.org/10.7472/jksii.2012.13.2.73 73

[4] David Knoke, Song Yang, SOCIAL NETWORK ANALYSIS - 2^{nd} Edition, Series: Quantitative Applications in the Social Sciences, SAGE Publications, 2008.

[5] Seong Joo Park, Kim Hyun Ah, Kwanghoon Kim, "Workflow-based Social Network Closeness Centrality Analysis Algorithm", Korean Society For Internet Information, Vol. 13, no. 2, pp.41-42, 2012.

[6] Seong Joo Park, Hakcung Kim, Hyunah Kim, Minjae Park, Kwanghoon Kim, "A Closeness Centrality Analysis Method for Disconnected Workflow-supported Social Networks", Korean Society For Internet Information, Vol. 14, no. 2, pp.107-108, 2013.

[7] Hosung Park, Haeun Kwak, Miyoung Cha, Subok Moon, "An Influential Ranking in Social Network", The Korean Institute of information Scientists and Engineers, Vol. 28, no. 3, pp.24-30, 2010.

[8] David Eppstein, Joseph Wang, "Fast Approximation of Centrality", Journal of Graph Algorithms and Applications, Vol. 8, no. 1, pp.39-45, 2004.
http://dx.doi.org/10.1.1.30.3710

[9] Meesun Kim, Hyun Ahn, Hyunah Kim, Minjae Park, Kwanghoon Kim, "Implementation of the Betweenness Centrality Analysis Algorithm for Workflow-supported Social Networks", Korean Society For Internet Information, Vol. 15, no. 2, pp.175-176, 2014.

[10] E. W. dijkstra, "A Note On Two Problems in Connexion with Graphs", Numerische Mathematik, Vol. 1, no. 1, pp.269-271, 1959.
http://dx.doi.org/10.1007/BF01386390

[11] Dokyoung Lee, Hyun Ahn, Hyunah Kim, Kwanghoon Kim, "Accuracy Analysis of an Estimated Closeness Centrality based Ranking Algorithm in Large Scale Workflow-supported Affiliation Networks", Korean Society For Internet Information, Vol. 15, no. 2, pp.171-172, 2014.

[12] Jihye Song, Minjoon Kim, Haksung Kim, Kwanghoon Kim, "A Framework: Workflow-based Social Network Discovery and Analysis", Proceedings of the 13th International Conference on Computational Science and Engineering, pp.421-426, 2010.
http://dx.doi.org/10.1109/CSE.2010.74

[13] Sungjoo Park, Minjae Park, Hyuna Kim, Haksung Kim, Wonhyun Yoon, Thomas B. Yoon, Kwanghoon Pio Kim, "A Closeness Centrality Analysis Algorithm for Workflow-supported Social Networks", Proceedings of the 15th International Conference on Advanced Communication Technology, pp.158-161, 2013.
http://dx.doi.org/10.7472/jksii.2013.14.10.77

[14] Kazuya Okamoto, Wei Chen, Xiang-Yang Li, "Ranking of Closeness Centrality for Large-Scale Social Networks", Lecture Notes in Computer Science of Frontiers in Algorithmics, Vol. 5059, pp.186-195, 2008.
http://dx.doi.org/10.1007/978-3-540-69311-6_21

[15] Jawon Kim, Hyun Ahn, Hyunah Kim, Kwanghoon Kim, "A Ranking Algorithm of Closeness Centralities for Large Scale Workflow-supported Social Networks", Korean Society For Internet Information, Vol. 15, no. 1, pp.55-56, 2014.

[16] Dokyoung Lee, Hyun Ahn, Hyunah Kim, Kwanghoon Kim, "A Ranking Algorithm of Closeness Centralities for Large-Scale Workflow-supported Affiliation Networks", in Proc. of 2014 Spring Conference of KSII, Korean Society For Internet Information, Vol. 15, no. 1, pp.37-38, 2014.

◐ 저 자 소 개 ◑

김 자 원 (Ja-won Kim)
2012년 ~현재 성결대학교 컴퓨터 공학과(재학)
관심분야 : 워크플로우/비피엠 기술, Social Network, Social Network Analysis
E-mail : jjawon@sungkyul.ac.kr

안 현 (Hyun Ahn)
2011년 경기대학교 컴퓨터과학과(학사)
2013년 경기대학교 일반대학원 컴퓨터과학과(석사)
2013년 ~현재 경기대학교 일반대학원 컴퓨터과학과(박사 과정)
관심분야 : 워크플로우/비피엠, IoT-aware Business Process
E-mail : hahn@kgu.ac.kr

김 광 훈 (Kwang-hoon Kim)
1984.2 경기대학교 이과대학 전자계산학과 학사
1986.2 중앙대학교 일반대학원 전자계산학과 석사
1994.5 University of Colorado Boulder, Department of Computer Science, MS
1998.5 University of Colorado Boulder, Department of Computer Science, Ph.D.
1986.2~1991.8 한국전자통신연구원 연구원
2005.3~2010.2 Univ. of Colorado Boulder, Department of Computer Science, 방문교수
2007.7~2010.6 경기대학교 콘텐츠융합소프트웨어연구센터장
1998.3~현재 경기대학교 컴퓨터과학과 교수
2002.3~현재 비피엠코리아포럼 부회장
2003.1~현재 WfMC ERC Vice-chair
2003.1~현재 TTA 정보통신국제표준전문가
2000.1~현재 한국인터넷정보학회 이사, 부회장
관심분야 : 워크플로우/비피엠, Process-Aware Information Systems, Process Discovery/Rediscovery, Workflow-Supported Social Networks
E-mail : kwang@kgu.ac.kr