

# 랜덤 워크를 활용한 그래프 랭킹 기반 추천 시스템

서울대학교 | 진우정·정진홍·강 유\*

## 1. 서론

그래프 랭킹 기반 추천 시스템은 소셜 네트워크와 같은 그래프에서 랭킹을 통해 특정 사용자와 관련성이 높은 사용자를 추천해주는 기법이다. 소셜 네트워크가 발전함에 따라 소셜 네트워크 상에서 친구를 추천해주는 많은 연구가 진행되어 왔다[1,2]. 친구를 추천하기 위한 기술은 여러 기업에 의해서 사용되고 있으며 대표적으로 페이스북에서 친구 추천 기능을 예로 들 수 있다.

소셜 네트워크 상에서 추천을 위해 사용되는 랜덤 워크 기반 랭킹 기법으로 Random Walk with Restart (이하 RWR)가 있다. RWR은 한 사용자 기준으로 다른 사용자의 중요도를 나타내는 랭킹 기법이며 개인화된 추천에 적합해 많은 추천 시스템에서 사용되어 왔다. 하지만 그래프 규모가 증가함에 따라 간단한 방법으로 RWR을 빠르게 계산하는데 한계가 있어, 대규모 그래프에서 고속으로 RWR을 계산할 수 있는 방법이 활발하게 연구되고 있다.

본 논문에서는 그래프 랭킹 기반 추천을 가속화하기 위한 랜덤 워크 기법에 관련된 연구들을 정리, 안내하고, 국제적으로 진행된 최신 연구들의 연구방향과 결과를 정리하여 소개한다. 또한 랜덤 워크에서 아직 해결되지 않은 연구과제들을 제시하며 향후 랜덤 워크를 활용한 그래프 랭킹 기반 추천 시스템의 연구 방향에 대한 논의를 진행한다.

먼저, 2장에서 그래프 랭킹 기반 추천 시스템에 대한 이해와 랜덤 워크가 무엇인지 소개하고 3장에서 랜덤 워크를 기반으로 한 기존 기법들을 소개한다. 4장

에서는 랜덤 워크를 활용하는 최근의 연구 동향을 소개하고 안내한다. 이를 통해 랜덤 워크를 활용한 그래프 랭킹 기반 추천 시스템의 발전 방향을 이해하고 추천 시스템에 대한 학문적 관심을 제고하고자 한다. 마지막으로 5장에서 결론을 맺는다.

## 2. 그래프 랭킹 기반 추천 시스템의 이해

### 2.1 그래프 랭킹 기반 추천 시스템의 개요

그래프 랭킹을 이용한 추천 시스템이란 그래프에서 각 노드에 해당하는 사용자들의 랭킹을 계산한 뒤 높은 랭킹 순으로 유저를 추천하는 것을 말한다. 그래프 랭킹은 여러 방법에 의해서 구해질 수 있는데 본 논문에서는 랜덤 워크를 활용한 랭킹 기반 추천 시스템에 대하여 논하고자 한다.

### 2.2 랜덤 워크의 이해

랜덤 워크는 그래프 상에서 어떤 시작 노드를 기점으로 무작위로 주변 간선을 선택하여 이웃 노드로 움직이는 것을 말한다. 랜덤 워크를 통해 반복적으로 무한히 이동하다보면 각 노드를 방문할 확률이 나오게 된다. 예를 들어 그림 1과 같이 노드들이 연결되어 있다고 하자. 먼저 시작 노드에는 랜덤 서퍼(Random surfer)가 존재하는데 랜덤 서퍼는 웹페이지를 이동할 수 있는 가상 사용자로서 간선을 따라 각 노드들을 방문할 수 있다. 이 예에서 랜덤 서퍼는 노드 A에서 시작하여 간선을 따라 무작위로 움직인다. 이 때 확률은 두 개의 간선이 존재하므로 1/2이다. 만약 B로 움직였다면 B에서는 간선이 3개이므로 1/3의 확률로 이웃 노드로 이동할 수 있다. 이와 같은 작업을 계속 하다보면 각 노드를 어느 정도 방문하는 지에 대한 확률이 나오게 되며 이 값이 크면 많이 방문한다는 의미를 가지게 된다. 따라서 이 확률로 랭킹을 매길 수 있다. 또한 랜덤 서퍼가 이웃 노드로 이동하는 것에 더해 일정 확률로 어느 노드로 점프하느냐에 따라 랜덤 워크의 세부 기법이 분류된다.

\* 종신회원

† 이 논문은 2015년도 정부(미래창조과학부)의 재원으로 정보통신기술진흥센터의 지원을 받아 수행된 연구임 (No.R0190-15-2012, 빅데이터 처리 고도화 핵심기술개발 사업 총괄 및 고성능 컴퓨팅 기술을 활용한 성능 가속화 기술 개발). 이 연구를 위해 연구장비를 지원하고 공간을 제공한 서울대학교 컴퓨터연구소에 감사드립니다. 서울대학교 공학연구원의 지원에도 감사를 드립니다.

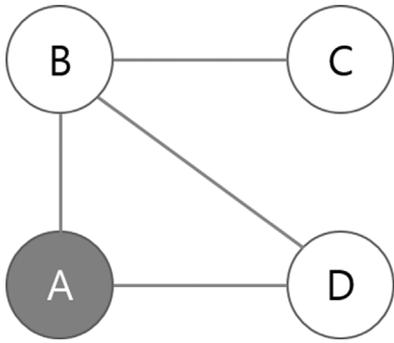


그림 1 예제 그래프. 랜덤 서퍼는 현재 노드에서 무작위로 이웃 노드에 방문 한다.

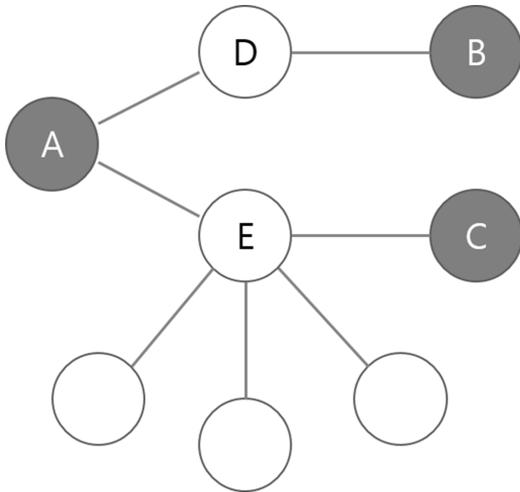


그림 2 피자 배달원 문제. 콜 그래프에서 E는 다수의 일반인으로부터 전화를 받는 피자 배달원이며, A 입장에서 피자 배달원과 연결된 C보다는 일반인과 연결된 B가 친구일 가능성이 높다. 랜덤 워크는 중간 노드의 차수를 고려한 중요도를 계산 할 수 있어 C보다 B의 중요성을 더 높게 계산해준다.

그러면 그래프 상에서 중요도를 계산하는데 랜덤 워크가 가진 장점은 무엇인지 피자 배달원 문제(Pizza delivery guy problem)를 통해 알아보자. 그림 2는 누가 누구에게 전화를 걸었는지를 나타내는 콜 그래프이며, 이 그래프에서 노드 A로부터 상대적으로 중요한 노드를 찾고 싶다고 하자. 노드 D는 A와 B만 연결되어 있고 노드 E는 A와 B뿐만 아니라 여러 노드와 연결되어 있다(여기서 노드 E는 여러 노드에 연결되어 있으므로 피자 배달원이라 생각할 수 있다). 먼저 최단 거리로 중요도를 매긴다고 가정해보자. 그러면 노드 B와 C는 A로부터 같은 거리에 있으므로 중요도가 같다.

그러나 랜덤 워크를 이용하여 중요도를 계산하면 결과는 달라진다. 노드 A에서 랜덤 서퍼가 출발하여 C에 도달하기 위해서는 E를 통과해야 하는데 E에서는 C 외에 다른 노드으로도 갈 수 있으므로 C에 랜덤 서퍼가 도

달할 확률은 낮아지게 된다. 반면, A에서 B로 가기 위해서는 D를 통과해야 하는데 D가 연결된 곳은 A와 B뿐이므로 B에 도달할 확률은 C보다 높다. 따라서 노드 B의 중요도는 노드 C의 중요도보다 높게 된다. 즉, 랜덤 워크는 거리뿐만 아니라 중간 노드들의 차수도 고려하는 기법이므로 이는 곧 친구를 추천하는데 있어 장점을 가진다. 실제로 E라는 노드는 차수가 높으므로 피자 배달원이나 유명인 등 나와는 크게 관련이 없는 노드일 수 있으며 그와 연결된 C는 나와 친구일 가능성이 낮다.

### 2.3 랜덤 워크를 활용한 그래프 랭킹 기법의 분류

랜덤 워크를 활용한 그래프 랭킹 기법으로 그래프 상에서 랜덤 서퍼의 행동에 따라 1) PageRank, 2) Random Walk with Restart로 구분된다.

#### • PageRank

PageRank는 랜덤 워크의 한 종류로서 월드 와이드 웹과 같은 하이퍼링크로 연결된 웹 페이지에 상대적 중요도를 계산하는 방법이다[3,4]. PageRank에서 랜덤 서퍼는 임의의 웹페이지에서 시작하여 이 웹페이지에 연결된 다른 웹페이지로 무작위로 이동할 수 있고 또한 연결 유무에 상관없이 임의의 웹페이지로 일정 확률로 이동할 수 있다. 어떤 웹페이지가 높은 페이지랭크 값을 가진다는 것은 랜덤 서퍼가 그 웹페이지를 방문할 확률이 높다는 것이며 이는 또한 상대적 중요도가 높다는 것이다.

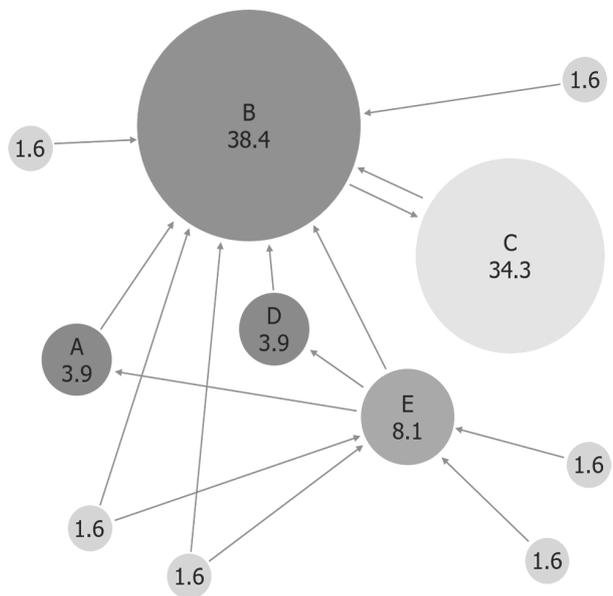


그림 3 페이지랭크 개념도. B가 다수의 중요한 노드로부터 가장 많은 연결을 갖고 있으므로 가장 높은 중요도를 가지게 되고, C는 높은 중요도를 가진 B로부터 연결이 되어있으므로 중요도가 높다.

예를 들어, 그림 3의 네트워크에서 각 웹페이지들은 페이지랭크 계산을 통해 표기된 숫자와 같이 상대적 중요도를 가지게 된다. 웹페이지 B는 여러 웹페이지로부터 연결이 되어 있으므로 중요성이 높다고 생각할 수 있다. 웹페이지 C는 웹페이지 간에 연결이 적지만 높은 중요도를 가진 웹페이지 B로부터 연결이 되어 있으므로 B와 마찬가지로 중요도가 높아진다. 그러므로 이 예제에서는 웹페이지 B와 C가 가장 중요하다고 볼 수 있다.

• Random Walk with Restart

PageRank는 모든 웹페이지에 관한 상대적 중요도를 고려해 글로벌 랭킹을 구하는 방법인 반면, RWR은 어느 한 웹페이지에 관한 상대적인 중요도를 고려해 개인화된 랭킹을 구한다[5]. RWR에서는 랜덤 서퍼가 웹페이지에 연결된 다른 웹페이지로 일정 확률로 이동할 수 있고 또는 정해진 한 웹페이지로 일정 확률로 이동할 수 있다. 이는 다음과 같은 식으로 기술할 수 있다.

$$r = (1 - c)\tilde{A}^T r + cq$$

$r$ 은 RWR 점수 벡터, 즉 질의 노드에 대한 각 노드의 중요도이며  $c$ 는 질의 노드로의 재시작 확률(일반적으로 0.15),  $q$ 는 질의 노드에 해당하는 값이 1이고 나머지는 0인 벡터,  $A$ 는 인접행렬이다.  $\tilde{A}$ 는 각 행의 합이 1이 되도록  $A$ 를 정규화한 행렬을 의미한다.

이와 같이 랜덤 워크를 활용한 그래프 랭킹 기법의 분류에 대해 알아보았다. PageRank는 그래프에서 글로벌 랭킹을 구하는 반면, RWR은 질의 노드에 개인화된 중요도를 계산하기 때문에 친구나 문서를 추천하는데 적합한 랭킹 기법이다. 다음 장에서 RWR을 어떻게 계산하는지 알아보도록 하자.

### 3. RWR 계산을 위한 기법들

#### 3.1 전통적인 RWR 계산 기법

RWR을 계산하기 위한 기본적인 방법으로 1) 멱 반복 방법(power iteration method)과 2) 역행렬을 이용한 방법이 있다. 멱 반복 방법은 아래의 식과 같이 반복적으로 RWR 점수 벡터( $r$ )를 계산하여 값이 수렴할 경우 계산을 끝내는 방법이다.

$$r^t \leftarrow (1 - c)\tilde{A}^T r^{t-1} + cq$$

이 방법은 인접 행렬을 저장하는 공간 외에 부가적

인 저장 공간과 전처리 시간이 필요하지 않아 대규모 그래프에서도 RWR을 계산할 수 있다는 장점이 있지만, 많은 행렬-벡터 곱이 필요하기 때문에 그래프 크기가 커지면 계산 시간이 느려지고 질의 노드가 바뀌면 계산을 처음부터 다시 해야 한다는 단점이 있다. 두 번째 방법은 역행렬을 이용하여 아래의 식과 같이 RWR 점수 벡터를 구하는 방법이다.

$$r = c(I - (1 - c)\tilde{A}^T)^{-1}q = cH^{-1}q$$

여기서  $H$ 는  $(I - (1 - c)\tilde{A}^T)$ 로 정의된다. 만일  $H$ 의 역행렬을 전처리를 통해 미리 가지고 있을 수 있다면, 한 번의 행렬-벡터 곱으로 RWR 점수 벡터를 구할 수 있어 빠르게 RWR 질의를 처리할 수 있다. 하지만 대규모 그래프에서 역행렬을 계산하는데 매우 많은 시간이 걸리며 역행렬의 결과를 메모리에 저장하기 위해서 매우 많은 양의 저장 공간이 필요하기 때문에, 대규모 그래프에서 RWR을 계산하는 것이 어렵다.

#### 3.2 근사 RWR 계산 기법

위와 같이 전통적인 RWR 계산 기법은 대규모 그래프에서 RWR 점수 벡터를 빠르게 구하는데 한계가 있기 때문에 대규모 그래프에서 실시간으로 RWR을 계산하는 응용에는 적절치 않다. 이러한 한계를 극복하고자 다소 랭킹의 정확도를 잃더라도 RWR 점수를 근사하여 빠르게 계산하는 근사 기법들이 제안되었다[6-8].

• Approximate NF(Neighborhood Formation)[6]

RWR 점수가 질의 노드 주변에서만 유의미한 값을 가지고 대부분의 노드에서는 0에 가깝게 분포한다는 점에 착안하여, 그래프 분할 알고리즘을 통해 여러 부분 그래프로 나눈 뒤 질의 노드가 있는 부분 그래프에서만 RWR을 계산하는 방법이 진행되었다[6]. 이 방법은 질의 노드가 속하지 않는 여러 부분 그래프에서 노드들의 RWR 점수는 0으로 두고, 질의 노드를 가지는 부분 그래프의 인접행렬에서 멱 반복 기법을 적용해 RWR 점수를 구한다. 이는 전체 그래프가 아닌 부분 그래프에서 RWR 계산을 하므로 RWR 점수 벡터를 빠르게 얻을 수 있지만, 그래프가 효과적으로 분할되지 않을 경우, 부분 그래프 간 상당 부분의 연결이 무시되기 때문에 RWR 기반의 랭킹의 정확도가 많이 떨어지게 된다.

• RPPR(Restricted Personalized PageRank)[7]

미리 부분 그래프를 찾는 방법이 아닌, 반복적으로 RWR 값을 구하는 계산 과정 중에 부분 그래프를 찾

아내는 RPPR 알고리즘 연구도 진행되었다[7]. 이 방법은 처음에는 부분 그래프를 질의 노드로만 한정시킨다. 그리고 RWR 계산을 하면서 부분 그래프 경계에 있는 노드로부터 연결된 노드의 RWR 값이 일정 값보다 크면 부분 그래프에 포함시키는 방법이다. 이 역시 전체 그래프가 아닌 질의 노드 근처의 부분 그래프만 계산하므로 RWR 계산이 빠르지만, 부분 그래프에 연결된 노드의 RWR 값이 일정 값보다 작으면 이 값이 무시되어 랭킹에 반영되지 않는 단점이 있다.

• B\_LIN, NB\_LIN[8]

앞의 두 방법의 단점을 보완하여 부분 그래프간의 연결을 근사적으로 고려하면서 RWR 계산을 빠르게 하는 B\_LIN과 NB\_LIN 알고리즘이 제안되었다[8]. B\_LIN은 먼저 그래프 분할 알고리즘을 사용하여 그래프를 분할한다. 그러면 전체 그래프는 부분 그래프 내부를 연결하는 간선의 인접행렬(내부 행렬)과 부분 그래프 간 연결되는 간선의 인접행렬(교차 행렬)로 표현된다. 그리고 나서 교차 행렬에 고유값 분해(eigenvalue decomposition)와 같은 행렬 분해 기법과 낮은 계수 근사법(low-rank approximation)을 사용하면 교차 행렬을 효과적으로 압축하여 근사할 수 있다. NB\_LIN은 B\_LIN과 다르게 그래프 분할을 적용하지 않고 주어진 인접 행렬에 낮은 계수 근사법을 바로 적용한 방법이다. 이러한 방법들을 통해 높은 정확도를 유지하며 빠르게 RWR을 계산할 수 있다. 하지만 이 방법 역시 그래프 분할이 효과적으로 이루어지지 않으면 교차 행렬의 원소의 개수가 많아지면서 낮은 계수 근사법의 정확도가 떨어지고, 결과적으로 계산된 RWR 점수의 정확도가 감소하게 된다.

3.3 전처리 기법 기반 RWR

역행렬을 이용한 방법을 이용하면 RWR 점수 벡터를 빠르게 구할 수 있다. 그러나 앞서 살펴본 바와 같이 역행렬을 이용한 방법은 많은 저장 공간을 필요로 하므로 대규모 그래프에서 현실적으로 적용하기 힘들다. 따라서 RWR 계산을 빠르게 처리하면서 역행렬과 관련된 연산을 효율적으로 하는 전처리 기법 연구가 진행되었다[9-12].

• 행렬 분해 기반 계산 기법[9,10]

노드의 순서를 재배열함과 동시에  $H$ 를 행렬 분해 기법을 이용하여 전처리 비용을 효과적으로 단축할 수 있는 기법이 제안되었다. 행렬 분해 기법은 임의의 행렬을 직교행렬과 상삼각행렬로 분해하는 QR 분해[9]나, 하삼각행렬과 상삼각행렬로 분해하는 LU 분해[10]가 사용된다. 이러한 행렬 분해는 노드의 순서를

재배열함으로써 행렬 분해를 빠르게 하고 분해된 행렬을 희박하게 만들 수 있다. 따라서  $H$ 의 역행렬을 직접적으로 구하는 것보다 노드의 순서를 재배열한 뒤 행렬 분해를 하고, 분해된 행렬들의 역행렬을 구하는 것이 더 효율적이다. 이 방법을 통해 RWR 점수 벡터를 비교적 빠르고 정확하게 구할 수 있지만, 행렬 분해 기반 기법이 역행렬을 구하는 것과 같은 시간 복잡도를 가지기 때문에 더 큰 규모의 그래프에서의 처리에 한계를 가진다.

• BEAR[11,12]

앞서 설명한 바와 같이 역행렬과 관련된 연산은 노드의 순서를 재배열함으로써 효율적으로 수행될 수 있다. 이러한 점에 착안하여 노드의 순서를 재배열하여 그래프를 효과적으로 압축시킨 뒤 RWR 계산을 빠르고 정확하게 하는 BEAR 기법이 제안되었다[11,12]. BEAR는 먼저 SLASHBURN[13,14]을 사용하여 노드 재배열을 한다. SLASHBURN은 노드 재배열을 통해 인접 행렬의 0 아닌 원소들을 특정 부분에 집중시켜 그래프를 효과적으로 압축한다. 예를 들어 그림 4와 같이 하이퍼링크로 연결된 웹페이지들을 그래프로 표현한 인접행렬이 있다. 이 그래프에 SLASHBURN을 적용하면 인접행렬의 원소들은  $H_{11}$  행렬의 대각선에 집중되어 블록 대각행렬을 이루며 나머지 원소들은  $H_{12}$ ,  $H_{21}$ ,  $H_{22}$ 에 집중된다. 따라서 역행렬을 직접적으로 구해 RWR을 계산하는 것보다, 블록 제거 기법을 통해  $H_{11}$ 의 대각성분의 역행렬을 구하고 분할된 부분 행렬의 조합으로 구성되는 행렬(Schur complement)의 역행렬을 구하면 효율적으로 RWR을 계산할 수 있다. 또한 구해지는 부분 역행렬에 LU 분해를 적용하면 전처리 시간과 메모리 사용량을 효과적으로 줄일 수 있다.

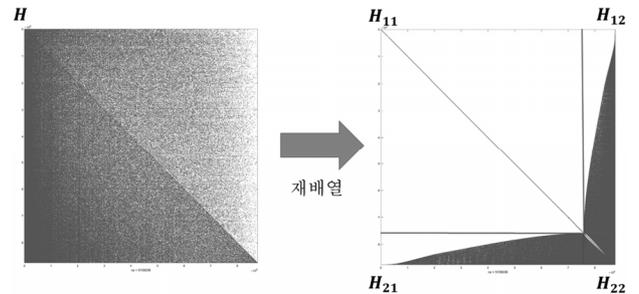


그림 4 SLASHBURN 적용 전, 후의 행렬. 하이퍼링크로 연결된 웹페이지들을 표현하는 인접행렬에 SLASHBURN을 적용하면 노드가 재배열됨에 따라 행렬 원소들이 대각성분에 모이게 되고 나머지 원소들은 차원이 낮은 행렬에 집중된다.

데이터 출처: <http://snap.stanford.edu/data/web-Google.html>

## 4. 랜덤 워크 활용 추천 시스템 최신 연구 동향

RWR을 큰 그래프에 대하여 빠르게 계산하는 연구에 덧붙여, 최근 랜덤 워크 기반 그래프 랭킹 추천 시스템의 연구 동향은 크게 두 갈래로 나뉠 수 있다: 1) 부호가 있는 그래프에서의 추천, 2) 학습을 통한 효과적 랭킹 기법. 본 장에서는 각 갈래에 대한 최신 연구 동향을 안내한다.

### 4.1 부호가 있는 그래프에서의 추천

최근에는 소셜 네트워크 서비스에서 사용자 간의 좋고 싫음을 나타낼 수 있는 기능도 제공하고 있다. 그래프에서 좋음(신뢰)을 나타내는 것은 간선에 양수로, 싫음(불신뢰)을 나타내는 것은 음수로 표현이 가능하다. 그러나 RWR 계산은 양수 간선만 가지는 그래프에서 이루어지므로 간선에 음수가 있는 그래프에서는 적용이 불가능하다. 게다가 질의 노드로부터 연속적으로 음수 간선이 있는 경우에는 이것이 신뢰를 의미하는 것인지 신뢰하지 않음을 의미하는 것인지 해석이 불분명하다. 이러한 문제점 때문에 그동안 음수 간선이 있는 그래프에서의 랭킹 기법 연구가 어려웠으며 이를 극복하기 위한 개선된 RWR을 고안하는 연구가 이루어지고 있다.

최근에는 양수만 있는 간선과 음수만 있는 간선을 따로 분리하여 RWR을 계산하는 방법이 제안되었다 [15]. 이 방법은 양수 간선과 음수 간선이 섞여있는 그래프를 두 개의 그래프로 나눈다. 한 그래프에는 양수의 간선만 존재하고 다른 그래프에는 음수의 간선만 존재한다. 양수의 간선만 존재하는 그래프에 RWR을 계산하고 음수의 간선만 존재하는 그래프에 음수 값을 양수로 치환하여 RWR을 계산한다. 최종적인 RWR 값은 이 두 값의 차이로 이루어진다. 하지만 이 방법은 하나의 그래프를 두 개의 그래프로 분리하여 생각하기 때문에 양수와 음수의 연관관계를 설명하지 못한다.

그래서 이 방법을 보완한 Troll-Trust Model [16] 알고리즘이 고안되었다. 이 알고리즘은 각 노드의 신뢰도와 한 노드가 다른 노드를 신뢰할 확률을 정의한다. 각 노드의 신뢰도는 같은 값으로 초기화되고 신뢰할 확률을 반복적으로 인접한 노드에 전파한다. 만약 한 노드가 신뢰할만한 노드로부터 많은 신뢰를 받고 있다면 그 노드는 높은 신뢰도 값을 가지게 된다. 이 신뢰도 값을 가지고 어떤 노드가 가장 신뢰할만한지 알 수 있다. 하지만 이 알고리즘은 어느 한 질의 노드에 관한 신뢰도가 아닌 전체 노드에 관한 신뢰도를 구하기 때문에 개인화된 신뢰도를 측정하지 못한다.

## 4.2 학습을 통한 효과적 랭킹 기법

RWR을 이용한 추천 시스템에 기계 학습을 사용하는 기법의 연구가 진행되어 온 바 있다 [1]. 최근에 기계 학습의 관심이 증가함에 따라 기계 학습과 추천 시스템의 결합이 이루어지고 있다. 이는 학습 기법을 사용하여 추천 시스템의 정확도를 높이는 방법이다.

이러한 연구들 중 주목할 만한 연구로 Supervised Random Walks(이하 SRW) [1]를 들 수 있다. SRW는 기계 학습 기법인 로지스틱 회귀와 RWR을 결합한 방법이다. 먼저 몇 개의 질의 노드에 관해 친구가 될 노드와 그렇지 않은 노드들이 주어지고 그래프의 각 연결마다 특징(feature)이 주어지고 있다고 가정하자. 특징은 각 노드들의 연결이 얼마나 오래되었는지 또는 각 노드의 차수 등이 될 수 있다. 이 특징은 매개변수와 결합되어 연결의 세기를 결정하게 된다. 특징과 매개변수 값의 곱이 크다면 연결은 강하게 되어있는 것이고 작다면 연결은 약하게 되어있는 것이다. 그러면 미리 계산된 연결의 강도를 간선 값으로 하여 RWR을 계산할 수 있는데 친구가 될 노드의 RWR 값이 그렇지 않은 노드의 RWR 값보다 크도록 매개변수를 최적화한다. 이 최적화된 매개변수와 특징을 곱하여 다른 질의 노드의 RWR을 계산하면 RWR 값이 큰 순서대로 친구를 추천해줄 수 있다.

## 5. 결 론

소셜 네트워크가 널리 쓰임에 따라 그래프에서의 친구 추천 기법에 대한 관심이 커지고 있다. 본 논문에서는 랜덤 워크를 활용한 그래프 랭킹 기반 추천 시스템에 대한 개략적인 안내와 더불어 RWR을 가속화하기 위한 알고리즘들을 안내하고 최근 5년간 주목받고 있는 두 개의 연구 흐름에 대해 안내를 하였다. 최근에는 RWR 계산의 가속화뿐만 아니라 간선에 부호가 있는 사인 그래프에서의 RWR 적용 연구와, 학습을 통한 효과적인 랭킹 기법을 통해 추천 시스템의 정교화와 고도화를 위한 연구들이 진행되어 왔다.

랜덤 워크는 식이 명료하고 수학적 이론이 뒷받침되기 때문에 추천 시스템뿐만 아니라 다양한 소셜 네트워크 현상을 이해하고 설명하는 데 풍부하게 활용될 것으로 보인다. 더욱이 아직 해결하지 못한 문제들과 알고리즘적 도전 분야가 많기 때문에 향후 다양한 연구 분야에서 랜덤 워크를 활용한다면 산업적으로나 학문적으로 많은 발전이 이루어질 것으로 기대된다.

## 참고문헌

- [ 1 ] L. Backstrom and J. Leskovec, "Supervised random walks: predicting and recommending links in social networks", In Proceedings of the fourth ACM international conference on Web search and data mining, pp. 635-644, 2011.
- [ 2 ] D. Liben-Nowell and J. Kleinberg, "The link prediction problem for social networks", Journal of the American society for information science and technology, Vol. 58, No. 7, pp. 1019-1031, 2007.
- [ 3 ] L. Page and et al., "The PageRank citation ranking: bringing order to the web", 1999.
- [ 4 ] S. Brin and L. Page, "Reprint of: The anatomy of a large-scale hypertextual web search engine", Computer networks, Vol. 56, No. 18, pp. 3825-3833, 2012.
- [ 5 ] T. Haveliwala H, "Topic-sensitive pagerank", In Proceedings of the 11th international conference on World Wide Web, pp. 517-526, 2002.
- [ 6 ] J. Sun and et al., "Neighborhood formation and anomaly detection in bipartite graphs", In Data Mining, Fifth IEEE International Conference on, p. 8, 2005.
- [ 7 ] D. Gleich and M. Polito, "Approximating personalized pagerank with minimal use of web graph data", Internet Mathematics, Vol. 3, No. 3, pp. 257-294, 2006.
- [ 8 ] H. Tong, C. Faloutsos and J. Pan, "Fast random walk with restart and its applications", 2006.
- [ 9 ] Y. Fujiwara and et al., "Efficient personalized pagerank with accuracy assurance", In Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 15-23, 2012.
- [10] Y. Fujiwara and et al., "Fast and exact top-k search for random walk with restart", Proceedings of the VLDB Endowment, Vol. 5, No. 5, pp. 442-453, 2012.
- [11] K. Shin and et al., "BEAR: Block Elimination Approach for Random Walk with Restart on Large Graphs", In Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data, pp. 1571-1585, 2015.
- [12] J. Jung and et al., "Random Walk with Restart on Large Graphs Using Block Elimination", ACM Transactions on Database Systems (TODS), Vol. 41, No. 2, p. 12, 2016.
- [13] U. Kang and C. Faloutsos, "Beyond 'caveman communities': Hubs and spokes for graph compression and mining" In Data Mining (ICDM), IEEE 11th International Conference on. IEEE, pp. 300-309, 2011.
- [14] Y. Lim, U. Kang and C. Faloutsos, "Slashburn: Graph compression and mining beyond caveman communities", Knowledge and Data Engineering, IEEE Transactions, Vol. 26, No. 12, pp. 3077-3089, 2014.
- [15] M. Shahriari and M. Jalili, "Ranking nodes in signed social networks", Social Network Analysis and Mining, Vol. 4, No. 1, pp. 1-12, 2014.
- [16] Z. Wu, C. Aggarwal and J. Sun, "The Troll-Trust Model for Ranking in Signed Networks", In Proceedings of the Ninth ACM International Conference on Web Search and Data Mining, pp. 447-456, 2016.

## 약 력



### 진우 정

2010~현재 서울대학교 전기정보공학부 학부과정  
관심분야: 데이터 마이닝, 그래프 링크 예측  
Email: woojung211@snu.ac.kr



### 정진홍

2014 전북대학교 컴퓨터공학부 학사  
2015.8 한국과학기술원 전산학과 석사  
2015.9~현재 서울대학교 컴퓨터공학부 박사과정  
관심분야: 대용량 그래프 마이닝, 그래프 랭킹  
Email: jinhongjung@snu.ac.kr



### 강 유

2003 서울대학교 컴퓨터공학부 학사  
2012.5 미국 Carnegie Mellon University, Computer Science 박사  
2012.6~12 미국 Carnegie Mellon University, Post doctoral fellow  
2013~2015.8 한국과학기술원 전산학과 조교수  
2015.9~현재 서울대학교 컴퓨터공학부 조교수  
관심분야: 데이터 마이닝, 빅 데이터, 대용량 기계 학습  
Email: ukang@snu.ac.kr