베이지안 개인화 순위 방법 기반 대용량 이종 네트워크에서의 효율적인 랜덤 워크 순위 계산*

한수민°, 김명호
한국과학기술원 전산학부
smhan@dbserver.kaist.ac.kr, mhkim@dbserver.kaist.ac.kr

Random Walk with Restart on a Large-scale Heterogeneous Network Based on the Bayesian Personalized Ranking

Sumin Han°, Myoungho Kim
School of Computing, Korea Advanced Institute of Science and Technology (KAIST)

요 약

랜덤 워크 기반 순위 계산 방법(random walk with restart)은 그래프 기반 추천 시스템에서 특정 노드에 대한 관련성이 높은 주변 노드를 검색하는 방법이다. 그러나 본 알고리즘을 대용량 이종 네트워크(large-scale heterogeneous network)에 적용하는 경우 두 가지 문제점이 생긴다. 먼저 다양한 간선의 종류를 고려하여 서로 다른 적합한 가중치를 부여하여야 하며, 가중치를 임의로 설정했을 경우 결괏값에 대한 정확성과 신뢰도가 떨어진다. 더불어 그래프가 대용량일 경우 메모리와 연산시간이 많이 소모되기 때문에 일반적인 행렬 연산방법을 적용하기 어렵다. 본 논문에서는 우선 베이지안 개인화 순위 방법(Bayesian personalized ranking)으로 간선의 종류에 따른 가중치를 학습한다. 이후 대용량에서 발생하는 문제를 해결하기 위해 BePI 알고리즘을 적용하여, 정확성과 계산성능 면에서 효율적으로 대용량 이종 네트워크에서 랜덤 워크 순위를 계산하는 방법을 제시한다.

1. 서 론

그래프란 객체를 나타내는 노드와 두 노드 간의 관계를 나타내는 간선으로 이루어진 구조이며, 소셜네트워크나 약물 작용과 같이 객체 간의 관계를 나타내는 등 다양한 분야에서 활용된다. 여기서 그래프의 객체 간의 관계를 활용하여 특정 노드와 관련이 높은 주변 노드를 검색하고 싶을 때 랜덤 워크 기반 순위 계산법(random walk with restart)을 활용할 수 있다. 그러나 본 알고리즘을 일반화하여 대용량의 이종 네트워크에 적용하여 사용할 경우 크게 아래의 두 가지의 문제점을 가진다.

먼저, 간선의 종류가 다양한 이종 네트워크에서 간선의 가중치(중요도)를 결정하는 것이 어렵다. 예를 들어 사용자에게 영화를 추천해주기 위해 대용량 그래프를 사용할 경우, '사용자→영화' 의 정보의 가치와 '영화→장르' 의 정보의 가치는 다르게 고려해야 한다. 그러나 간선의 종류에 따른 가중치를 사람이 임의로 주는 것은 매우 주관적이다. 따라서 결괏값에 대한 신뢰도를 높일 수 있도록 간선의 가중치를 결정하는 객관적인 방법이 필요하다.

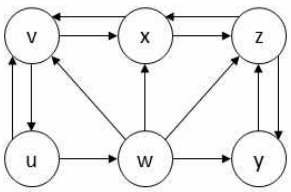
둘째로, 그래프가 대용량일 경우 메모리와 계산시간이 많이 소요되는 문제가 발생한다. 따라서 효율적으로 메모리를 활용하고 계산시간을 보다 단축해줄 방법이 필요하다.

본 논문에서는 위의 두 가지 문제점을 해결하기 위해 베이지안 개인화 순위 방법(Bayesian personalized ranking)으로 간선의 가중치를 기계적으로 학습하고, 대용량에서의 랜덤 워크 계산의 효율성을 위해 BePI를 적용한다. 더불어 각 알고리즘을 적용하는 과정에서 계산효율을 높이는 방법을 제안한다.

* 본 연구는 미래창조과학부 및 한국연구재단의 (재)유전자동의 보감사업단(2013M3A9C4078137), 과학기술정보통신부 및 한국연구재단의 중견연구자지원사업(NRF-2016R1A2B4014658)의 연구비 지원에 의해 수행되었습니다.

2. 배경 지식 및 관련 연구

2.1. 랜덤 워크 순위 계산법 (Random Walk with Restart)



(a) Network graph

$$A_{adj} = \begin{pmatrix} 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 \end{pmatrix}$$

(b) Adjacency matrix

그림 1. 그래프(a)와 인접 행렬(b); 각 행과 열은 노드 u, v, w, x, y, z이며 간선 $u \rightarrow v$가 존재하면 $\mathrm{A}_{\mathrm{adj}}[u,v] = 1$이다.

랜덤 워크 기반 순위 계산법이란 질의 노드(query node)로부터 연관성이 높은 노드들을 검색하는 알고리즘이다. 크게 랜덤 워크 부분과 재시작 부분으로 나누어져 있으며, 수식으로 아래와 같이 표현된다.

$$\mathrm{r}^{(i+1)} = \alpha \mathrm{W} \mathrm{r}^{(i)} + (1-\alpha)\mathrm{q}$$

여기서 $\mathrm{r}^{(i)}$는 i번째 단계에서 각 노드의 순위 점수 벡터이며 위 계산을 $\mathrm{r}^{(i+1)} = \mathrm{r}^{(i)}$로 수렴할 때까지 반복한다. α는 랜덤 워크 비율이며 주로 0.7~0.9의 값을 사용한다. W는 전환 행렬(transition matrix)로서 각 노드로부터의 다음 단계에 머무르게 될 노드의 랜덤 워크 확률을 계산하는 데 필요하다. q는 질의 노드에 해당하는 값만 1이고 나머지가 0인 원-핫 벡터(one-hot vector)이다.

동종 네트워크(homogeneous network)에서 전환 행렬은 인접 행렬(adjacency matrix)을 행 정규화(row normalize)한 것의 전치(transpose)인 $\widetilde{\mathrm{A}_{\mathrm{adj}}}^{T}$과 같다. 결과적으로 동종 네트워크에서의 전환 행렬은 열 정규화(column normalized)되어 한 노드에서 뻗어 나가는 각 간선의 가중치가 일정하다. 그러나 이종 네트

워크의 경우 각 간선의 중요도에 따라 각기 다른 가중치를 가지게 된다. 이러한 간선의 종류에 따른 가중치를 기계적으로 학습하기 위해 아래의 베이지안 개인화 순위법을 적용한다.

2.2. 베이지안 개인화 순위법(Bayesian personalized ranking) [1, 2]

베이지안 개인화 순위법은 세 노드 u, v, x 가 간선의 집합 E에서 $(u,v)\in E$이고 $(u,x)\notin E$인 경우 $u\rightarrow v$의 선호도 계산 값이 $u\rightarrow x$의 값보다 커야 한다는 점에서 착안하여 고안된 방법이다. 추천알고리즘에서 간선 예측(edge prediction)에 활용할 수 있으며 임의의 선호도 계산 알고리즘에 적용할 수 있다. 본 논문에서는 랜덤 워크 순위 계산 알고리즘에 대해 간선 종류에 따른 가중치를 매개변수 Θ로 하여 적용하였으며, 아래의 BPR 값이 커지도록 Θ를 학습하면 적합한 값을 얻을 수 있다.

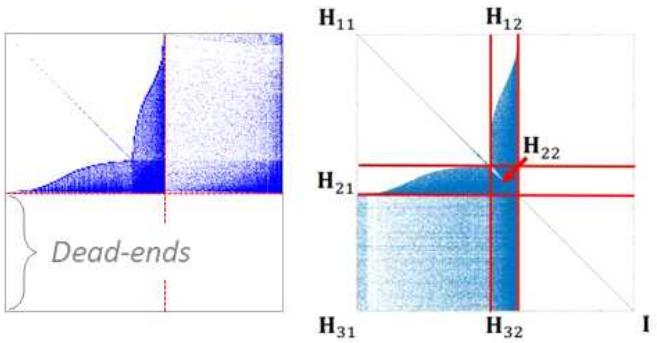
$$\mathrm{BPR} = \ln \prod_{\substack{(u,v)\in E\\(u,x)\notin E}} \sigma(r_{vu}(\Theta) - r_{xu}(\Theta))$$

여기서 시그모이드(sigmoid) $\sigma(z) = 1/(1-\exp(-z))$ 함수는 z값이 양수면 1에 가깝고 음수면 0에 가까워지는 특징이 있다. $r_{vu}(\Theta)$와 $r_{xu}(\Theta)$는 각각 u에서부터 질의한 v와 x의 순위 알고리즘 계산 결괏값이다. 결과적으로, 아래와 같이 경사 하강법(gradient descent)을 통해 적합한 Θ값을 찾을 수 있다.

$$\Theta \leftarrow \Theta + \eta \frac{\partial \mathrm{BPR}}{\partial \Theta}$$

여기서 η는 학습률(learning rate) 이다.

2.3. SlashBurn[3]과 BePI[4]



(a) Adjacency matrix after node reordering by SlashBurn

(b) Separation of H = I - αW matrix for BePI

그림 2. (a)노드 순서를 재정렬한 인접 행렬과 (b)BePI 알고리즘 계산을 위한 행렬의 부분

SlashBurn은 연결된 간선이 상대적으로 높은 중심(Hub) 노드와 연결된 간선이 거의 없는 주변(Spokes) 노드를 구분하여 노드 순서를 재정렬 해주는 알고리즘이다. <그림 2(a)>와 같이 밖으로 나가는 간선이 없는 막다른(Dead-end) 노드를 제외하면 안쪽 부분은 화살표 방향을 띄게 되는데, 왼쪽 위에서부터 거의 색이 차 있는 부분이 없는 부분이 주변 노드이고 색이 촘촘하고 길게 뻗어있는 화살표 끝부분이 중심 노드이다.

BePI는 이를 바탕으로 고속 랜덤 워크 계산을 가능하게 하는 알고리즘이다. 랜덤워크 공식을 유도하면

$$\mathrm{r}^{(i+1)} = \alpha \mathrm{W}\mathrm{r}^{(i)} + (1-\alpha)\mathrm{q} \Leftrightarrow \mathrm{r} = (1-\alpha)\mathrm{H}^{-1}\mathrm{q}$$
$$\text{where} \quad \mathrm{H} = \mathrm{I} - \alpha \mathrm{W}$$

가 되는데, H 행렬을 <그림2(b)>와 같이 부분을 나누면

$$\begin{bmatrix} \mathrm{r}_1 \\ \mathrm{r}_2 \\ \mathrm{r}_3 \end{bmatrix} = \begin{bmatrix} \mathrm{H}_{11}^{-1}(\mathrm{cq}_1 - \mathrm{H}_{12}\mathrm{r}_2) \\ \mathrm{S}^{-1}(\mathrm{cq}_2 - \mathrm{cH}_{21}\mathrm{H}_{11}^{-1}\mathrm{q}_1) \\ \mathrm{cq}_3 - \mathrm{H}_{31}\mathrm{r}_1 - \mathrm{H}_{32}\mathrm{r}_2 \end{bmatrix}, \quad \mathrm{S} = \mathrm{H}_{22} - \mathrm{H}_{21}\mathrm{H}_{11}^{-1}\mathrm{H}_{12}$$

위와 같다. 이때 H_{11}는 블록 대각 행렬(block diagonal matrix) 이며 H_{11}^{-1}는 각각의 작은 블록의 역행렬을 구하므로 계산시간이 적게 든다. 또한 H_{22}와 같은 크기를 가지는 S행렬을 이용하여 r_2를 계산할 때, GMRES(Generalized minimal residual method) 알고리즘을 사용하여 직접 S^{-1}를 계산하지 않고 $\mathrm{Sr}_2 = \hat{\mathrm{q}}_2$의 꼴을 만족하는 r_2를 반복적인(iterative) 방법으로 특정 값에 수렴하도록 계산하여서 일반적인 역행렬 계산시간보다 훨씬 단축한다. 결론적으로 BePI는 S나 H_{11}^{-1}와 같은 일부 행렬을 미리 계산하고 직접적인 역행렬 계산 대신 GMRES 알고리즘을 이용하여 계산시간을 단축한다고 볼 수 있다.

3. 제안 방법

베이지안 개인화 순위법을 랜덤 워크 기반 순위 계산법에 적용하기 위해 우선 매개변수 Θ를 간선의 종류에 따른 가중치의 집합 $(p_{\tau_1}, p_{\tau_2}, \ldots, p_{\tau_m})^T$로 둔다. 이후 전환 행렬을 아래와 같이 정의한다.

$$\mathrm{W}_{ij} = p(v_i|v_j) := \frac{p_{\psi(ji)}}{|D_j^{\psi(ji)}|}$$

여기서 $\psi(ji) \in \{\tau_1, \tau_2, \ldots, \tau_m\}$는 노드 v_j에서 노드 v_i로 연결된 간선의 종류를 뜻하며, $D_j^{\psi(ji)}$은 노드 v_j에서 $\psi(ji)$타입의 간선으로 연결된 모든 노드의 집합을 의미한다. 2.2.에서 제시된 BPR 식을 편미분 하여 쓰면 다음과 같이 쓸 수 있는데,

$$\frac{\partial \mathrm{BPR}}{\partial \Theta} = \sum_{\substack{(u,v)\in E\\(u,x)\notin E}} \frac{1}{1+e^{r_{vu}(\Theta)-r_{xu}(\Theta)}} \frac{\partial}{\partial \Theta}[r_{vu}(\Theta) - r_{xu}(\Theta)]$$

여기서 재귀적인 랜덤 워크 계산식을 풀어쓰면,

$$\mathrm{r}^{(k)} = \alpha^k \mathrm{W}^k \mathrm{q} + \sum_{i=0}^{k} \alpha^i (1-\alpha)\mathrm{W}^i \mathrm{q}$$

와 같아지며, 편미분 식을 간소화하여 유도하기 위해 $k=3$일 때까지만을 고려하여 그때의 식을 p_τ에 대해 미분할 경우

$$\begin{aligned}\frac{\partial r_{ui}}{\partial p_\tau} = \alpha^3 &\left(\sum_{\psi(ip)=\tau} \frac{\mathrm{W}_{pq}\mathrm{W}_{qu}}{|D_p^\tau|} + \sum_{\psi(pq)=\tau} \frac{\mathrm{W}_{ip}\mathrm{W}_{qu}}{|D_q^\tau|} \right. \\ &\left. + \sum_{\psi(qu)=\tau} \frac{\mathrm{W}_{ip}\mathrm{W}_{pq}}{|D_u^\tau|} \right) + \alpha^2(1-\alpha) \left(\sum_{\psi(ip)=\tau} \frac{\mathrm{W}_{pu}}{|D_p^\tau|} \right. \\ &\left. + \sum_{\psi(pu)=\tau} \frac{\mathrm{W}_{ip}}{|D_u^\tau|} \right) + \alpha(1-\alpha) \sum_{\psi(iu)=\tau} \frac{1}{|D_u^\tau|}\end{aligned}$$

가 되어, 이를 이용해 $\partial \mathrm{BPR}/\partial \Theta$를 계산할 수 있다.

여기서 $\partial r_{ui}/\partial p_\tau$ 를 계산하는 과정에서 $k=1,2,3$ 단계의 랜덤 워크 계산 결과에 대해 $\psi(ji)=\tau$의 경로를 거쳐 갈 경우 해당 부분을 편미분 하여

$$\partial \mathrm{W}_{ij}/\partial p_{\psi(ji)} = 1/|D_j^{\psi(ji)}|$$

으로 계산되었음을 알 수 있는데, 여기서 같은 종류의 메타 경로(meta path)를 미리 묶어 다음과 같이 $\partial r_{ui}/\partial p_\tau$ 의 식을

$$\frac{\partial r_{ui}}{\partial p_\tau} = \sum_{Z\in Metapath} \alpha_Z \beta(Z,u,i) \left(\frac{\partial}{\partial p_\tau} \prod_{\mu\in Z} p_\mu \right)$$

로 더 빠르게 계산할 수 있다. 여기서 $Z \in Metapath$ 는 u에서 i로 이동할 때 거치는 간선 개수가 $k_Z \leqq k$ 가 되는 모든 메타 경로를 의미하며 이는 그래프 스키마에서 너비 우선 탐색(Breadth-first search)을 통해 사전에 찾아낼 수 있다. 또한

$$\alpha_Z = \begin{cases} \alpha^k & \text{if } k = k_Z \\ \alpha^k(1-\alpha) & \text{otherwise} \end{cases}, \ \beta(Z,u,i) = \prod_{\mu \in Z} \frac{1}{|D^\mu|}\Bigg|_{u \to i}$$

로 표현할 수 있는데, α_Z는 k_Z에 따라 다르게 붙는 상수이며 $\beta(Z,u,i)$는 Z경로를 거치는 과정에서 발생하는 분모 부분을 미리 계산하는 것이다. 이 과정을 통해 매번 학습 과정에서 값이 변화하는 p_τ값이 관여하지 않는 α_Z와 $\beta(Z,u,i)$를 미리 계산하여 대용량에서의 학습시간을 훨씬 단축할 수 있다.

이처럼 BPR을 이용하여 $\Theta = (p_{\tau_1}, p_{\tau_2}, \dots, p_{\tau_m})^T$ 에 적합한 p_τ값을 찾아낼 수 있다. 이후 2.3.에서 소개한 BePI 알고리즘으로 빠르게 결과를 계산한다.

4. 결 과

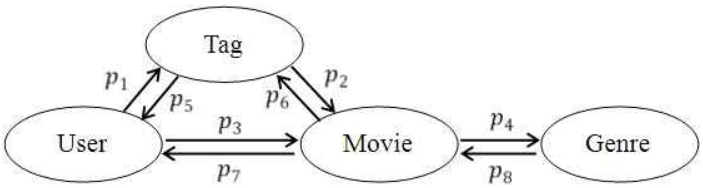


그림 3. MovieLens 20M 그래프 스키마

본 논문에서는 MovieLens 20M* 데이터를 사용하여 사용자에게 영화를 추천하도록 하는 실험을 수행하였다. 해당 데이터는 사용자가 영화에 대해 평가한 약 2천만 개의 데이터를 가지고 있으며 키워드를 나타내는 태그(tag), 그리고 영화의 장르에 대한 정보가 함께 고려되어 있다. 그래프의 스키마(schema)와 학습하는 $\Theta = (p_1, p_2, \dots, p_8)^T$는 <그림3>과 같다.

표 1. MovieLens 20M 노드 및 간선의 개수

노드	개수	간선	개수
User	125426	User-Movie (p_3, p_7)	4546132
Movie	27278	User-Tag (p_1, p_5)	931128
Tag	38644	Tag-Movie (p_2, p_6)	931128
Genre	20	Movie-Genre (p_4, p_8)	54406

본 데이터 중 User-Movie의 간선의 경우 각 사용자가 영화에 대해 평가한 평균 점수의 1.2배보다 높은 경우를 간선이 존재하는 것으로 가정하고 실험하였고, 선정된 약 454만 개의 간선 중에서 1%만을 골라 학습 간선(E_{train})으로 선정하고, 나머지를 테스트 간선(E_{test})으로 취급하여 실험을 진행하였다.

베이지안 개인화 순위 학습의 경우 $k=4$의 경로에 대하여 $(u,v) \in E_{train}$이고 $(u,x) \notin E = E_{train} \cup E_{test}$ 인 (u,v,x)쌍에 대해 수행하였다. 테스트는 $(u,v) \in E_{test}$에서 무작위의 10만 개 간선을 골라 $(u,x) \notin E$인 모든 x에 대해 $r_{vu} > r_{xu}$의 조건을 검증하는 AUC(Area under the curve)를 수행하였다.

$$\mathrm{AUC} = \frac{1}{|E_{test}|} \sum_{(u,v) \in E_{test}} \left[\sum_{(u,x) \notin E} \frac{u(r_{vu} - r_{xu})}{|(u,x) \notin E|} \right]$$

여기서 $u(z)$는 z가 양수이면 1이고 음수이면 0을 가지는 함수이며, $r_{vu} \neq 0 \ \wedge \ r_{xu} \neq 0$ 을 만족하는 경우에 대해서만 각 테스트 결과의 평균에 반영하였다.

표 2. AUC 계산 결과.

동종 AUC(%)	간선 개수(개)	동종(%)	이종(%)	성능향상(%)
0~30	1218	20.341	**26.751**	6.410
30~50	2091	41.097	**45.793**	4.697
50~80	8053	68.719	**72.381**	3.662
80~100	45318	**94.725**	94.077	-0.648
≤50	3309	33.457	**38.784**	5.327
≤80	11362	58.449	**62.596**	4.147
all	56680	87.453	**87.766**	0.313

<표2>에서 첫 번째 열의 "동종 정확도"는 동종 네트워크로 계산한 결괏값에 따른 성능향상을 비교 분석하기 위해 분류한 것이며 각각에 대해 이종 네트워크로 학습한 결괏값과 비교하였다. 그 결과 동종 네트워크에서의 AUC가 이미 높게 나왔던 테스트 간선보다 AUC가 낮았던 구간에서 이종 네트워크로의 계산이 더 많은 성능향상이 되었다는 것을 알 수 있었다.

또한, Paralution** 모듈을 사용하여 BePI를 구현하여 실행한 결과 tolerance≤1e-15 (직전 결괏값으로부터 벡터 거리)로 수렴조건을 두었을 때 전처리(Preprocessing)는 42.188 초, 질의(Query) 시간은 1000개의 무작위 테스트에서 0.115 초를 보였다. 반면 같은 조건의 tolerance에서 랜덤 워크 계산을 반복적으로 수렴할 때까지 수행할 경우 0.226 초의 수행시간이 소모되었으며, 질의 시간을 경우 BePI의 계산성능이 2배 정도 빠르다는 것을 알 수 있다.

5. 결 론

본 논문에서는 대용량 이종 네트워크에서 효율적으로 랜덤 워크 기반 순위 계산법을 적용할 수 있는 방법을 제안하였다. 먼저 베이지안 개인화 순위법으로 이종 네크워크에서의 간선의 종류에 따른 가중치를 학습하고, 이후 BePI 알고리즘을 활용한다면 한 번의 전처리과정 이후, 더욱 빠르게 질의 처리를 수행할 수 있다. 또한, 대용량의 그래프에서 영화를 추천하는 실험을 수행한 결과 가중치를 반영한 이종 네트워크 계산 정확도가 동종 네트워크가 낮은 정확도를 보이던 구간에서 더욱 높은 성능향상을 보였는데, 이는 기존에 데이터가 충분하지 못하여 동종 네트워크가 좋은 성능을 보이지 못했던 구간에서 이종 네트워크로의 계산이 더 좋은 추천 성능을 보였다는 것을 의미한다. 따라서 본 연구를 활용한다면 데이터가 충분하지 못하여 낮은 추천 정확도를 보이던 이종 네트워크를 활용하는 시스템에 더욱 효과적으로 적용될 수 있을 것으로 기대된다.

참 고 문 헌

[1] S.Rendle et al., BPR: Bayesian Personalized Ranking from Implicit Feedback, UAI, 2009.

[2] Z.Jiang et al., Recommendation in Heterogeneous Information Networks Based on Generalized Random Walk Model and Bayesian Personalized Ranking, WSDM, 2018.

[3] Y.Lim et al., SlashBurn: Graph Compression and Mining beyond Caveman Communities, IEEE TKDE, 2014.

[4] J.Jung et al., BePI: Fast and Memory-Efficient Method for Billion-Scale Random Walk with Restart, SIGMOID, 2017.

* https://grouplens.org/datasets/movielens/20m/

** http://www.paralution.com/