creative commons

COMMONS DEED

저작자표시-비영리-변경금지 2.0 대한민국

이용자는 아래의 조건을 따르는 경우에 한하여 자유롭게

- 이 저작물을 복제, 배포, 전송, 전시, 공연 및 방송할 수 있습니다.

다음과 같은 조건을 따라야 합니다:

저작자표시. 귀하는 원저작자를 표시하여야 합니다.

비영리. 귀하는 이 저작물을 영리 목적으로 이용할 수 없습니다.

변경금지. 귀하는 이 저작물을 개작, 변형 또는 가공할 수 없습니다.

- 귀하는, 이 저작물의 재이용이나 배포의 경우, 이 저작물에 적용된 이용허락조건을 명확하게 나타내어야 합니다.
- 저작권자로부터 별도의 허가를 받으면 이러한 조건들은 적용되지 않습니다.

저작권법에 따른 이용자의 권리는 위의 내용에 의하여 영향을 받지 않습니다.

이것은 이용허락규약(Legal Code)을 이해하기 쉽게 요약한 것입니다.

Disclaimer

dCollection

공학석사학위논문

마르코프 클러스터링을 이용한 영향력 최대화 알고리듬

Influence Maximization Algorithm Using Markov Clustering

2012년 8월

서울대학교 대학원
전기컴퓨터공학부
김 청 림

마르코프 클러스터링을 이용한 영향력 최대화 알고리듬

지도 교수 이 상 구

이 논문을 공학석사 학위논문으로 제출함
2012년 8월

서울대학교 대학원
전기컴퓨터공학부
김 청 림

김 청 림의 석사 학위논문을 인준함
2012년 8월

위 원 장 김 형 주 (인)
부위원장 이 상 구 (인)
위 원 김 선 (인)

초 록

최근들어 소셜 네트워크 서비스(SNS)가 확산됨에 따라 소셜 네트워크가 효과적인 광고 플랫폼으로 부상하였다. SNS는 구성원간 정보나 영향력을 쉽게 전파할 수 있는 매체로써 적은 비용으로 최대의 사람들에게 광고를 할 수 있기 때문이다. 따라서 사전에 구축된 소셜 네트워크 상에서 효율적으로 마케팅을 수행하는 것이 중요해졌으며, 이러한 문제를 영향력 최대화 문제라고 한다. 영향력 최대화 문제란 특정한 정보확산 모델을 가정할 때, 소셜 네트워크 상에서 영향력을 최대화 하는 k개의 노드를 찾는 방법이다.

영향력 최대화 문제에서 최적해를 찾는 것은 NP-Hard임이 증명되었기 때문에 근사해를 찾는 알고리듬이 연구되었다. 그러나 영향력 최대화 문제의 근사해를 찾는 탐욕적 알고리듬들은 영향력 계산에 몬테-카를로 시뮬레이션을 이용하기 때문에 높은 정확도를 위해서 여러번의 시뮬레이션을 수행할 경우 실행속도가 매우 느리다. 몇몇 연구들은 알고리듬의 정확도를 다소 포기하더라고 실행속도를 최대한 개선하는 방향으로 연구를 진행하였다. 해당 연구들은 영향력 최대화 문제에 알맞는 휴리스틱을 이용하여 정확성을 희생하면서 보다 빠르게 영향력이 높은 노드를 찾는 해법을 제시하고 있다.

본 연구에서는 소셜 네트워크가 무수히 많은 소규모 집단으로 이루어진 점에서 착안하여, 소셜 네트워크 그래프를 여러 군집으로 클러스터링한 후 각 군집에서 가장 영향력있는 노드를 선택하는 방식으로 영향력 최대화 문제의 해법을 찾는다. 또한 마르코프 클러스터링을 수정하여 각 클러스터 내의 끌개를 빠르게 찾는 방법을 제시한다. 이

를 이용해 얻은 끌개들을 이용하여 기존의 알고리듬을 개선하는 하이브리드 알고리듬을 제시한다. 본 논문에서 제시한 하이브리드 알고리듬들은 실제 데이터셋을 이용하여 검증한 결과, 더욱 빠른 시간 안에 기존 탐욕적 알고리듬에 근사한 영향력을 갖는 k 개의 시드 노드들을 찾아내며, 기존의 휴리스틱보다 더욱 좋은 성능을 보이는 것으로 나타났다.

주요어 : 영향력 최대화, 그래프 클러스터링, 마르코프 클러스터링
학번 : 2010-23255

목 차

I. 서론 **1**

1.1 연구의 배경 1

1.2 연구의 내용 3

1.3 논문의 구성 5

II. 문제 정의 **6**

2.1 소셜 네트워크 그래프 및 정보 확산 모델 6

2.1.1 소셜 네트워크 그래프 6

2.1.2 정보 확산 모델 7

2.2 영향력 최대화 문제 9

III. 관련 연구 **11**

3.1 탐욕적 접근법 11

3.2 휴리스틱 접근법 13

IV. 마르코프 클러스터링을 이용한 영향력 최대화 문제 **16**

4.1 마르코프 클러스터링 16

4.2 마르코프 클러스터링을 이용한 끌개 탐지 19

4.3 끌개 탐색을 이용한 새로운 영향력 최대화 알고리듬 . . 22

4.3.1 MCL 휴리스틱 22

4.3.2 MCL Greedy 휴리스틱 23

4.3.3 MCL Degree Discount 휴리스틱 24

V. 성능 평가 **26**
5.1 데이터셋 26
5.2 비교 대상 27
5.3 실험 방식 28
5.4 기존 마르코프 클러스터링과 끌개 탐지 알고리듬 비교 . 28
5.5 팽창률에 따른 알고리즘 성능 및 수행 시간 분석 30
5.6 성능 비교 31
5.6.1 IC 모델에서의 성능 비교 32
5.6.2 WC 모델에서의 성능 비교 36
5.7 그래프 크기와 수행 시간과의 관계 39

VI. 결론 및 향후연구 **41**
6.1 결론 41
6.2 향후 연구 42

참고 문헌 **44**

Abstract **47**

그 림 목 차

그림 1. 마르코프 클러스터링 예시 16

그림 2. Expand 연산 17

그림 3. Inflation 연산 18

그림 4. 팽창률이 각 알고리듬 성능에 미치는 영향 31

그림 5. 팽창률이 각 알고리듬 수행 시간에 미치는 영향 . . 31

그림 6. IC 모델을 가정할 때 NetHept Collaboration Network 에서의 영향력 전파 성능 32

그림 7. IC 모델을 가정할 때 Computational Geometry Collaboration Network에서의 영향력 전파 성능 33

그림 8. IC 모델을 가정할 때 Facebook 데이터셋에서의 영향력 전파 성능 34

그림 9. IC 모델에서 각 알고리듬 수행시간 35

그림 10. WC 모델을 가정할 때 NetHept Collaboration Network에서의 영향력 전파 성능 36

그림 11. WC 모델을 가정할 때 Computational Geometry Collaboration Network에서의 영향력 전파 성능 37

그림 12. WC 모델을 가정할 때 Facebook 데이터셋에서의 영향력 전파 성능 38

그림 13. WC 모델에서 각 알고리듬 수행시간 39

그림 14. 그래프 크기에 대한 알고리듬 수행 시간 증가 비교 40

표 목 차

표 1. 데이터셋 정보 27
표 2. precision & recall 29
표 3. runtime comparison (sec) 29

제 1 장

서론

1.1 연구의 배경

최근들어 Facebook[1]이나 Twitter[2]등의 소셜 네트워크 서비스(SNS)가 등장하고 사용자의 수가 늘어가는 추세이다. 사용자들은 SNS를 통하여 장소 혹은 시간에 구애 받지 않고 다른 사용자들과 소통하고 집단을 구성할 수 있게 되었다. SNS는 구성원간 정보, 영향력, 아이디어 등을 쉽게 전파할 수 있는 매체로 자리 잡았으며, 이러한 성질에 따라 현재 하나의 광고 플랫폼으로서 인정받았다.

이러한 추세에 힘입어 최근 이미 구축된 소셜 네트워크를 이용하여 특정 상품을 홍보하는 바이럴 마케팅(Viral marketing) 기법이 대두되었다. 바이럴 마케팅이란 소셜 네트워크 상에서 입소문(word-of-mouth) 효과를 이용하여 특정 상품에 대한 광고가 소셜 네트워크 구성원들을 통해 퍼져나가는 것을 말한다. 예를 들어, Hotmail[3]의 경우 hotmail계정을 가진 사람이 그렇지 않은 사람에게 메일을 보낼 경우 "Get your private, free email at http://www.hotmail.com"이란 메세지를 삽입하여 이메일을 수신하는 사람이 자연스럽게 발신자로부터 Hotmail에 대한 광고를 접할 수 있는 바이럴 마케팅을 수행하였다. 이러한 바이럴 마케팅의 결과로 Hotmail은 최소한의 광고 비용으로 단 2년만에

[1] http://www.facebook.com
[2] http://www.twitter.com
[3] http://www.hotmail.com

1,200만명의 가입자를 보유하는 서비스로 거듭났다[1]. 최근엔 바이럴 마케팅을 상품 판매에 접목시킨 소셜 커머스 기업들이 생겨나고 있으며, groupon[4], 티켓몬스터[5]등의 서비스들이 바이럴 마케팅을 이용하여 성공한 대표적인 기업으로 꼽히고 있다.

소셜 네트워킹 서비스를 통한 소셜 마케팅은 한정된 수의 사용자들에게 홍보를 하여 적은 비용으로도 홍보 효과를 극대화하는 것을 목표로 한다. 따라서 효과적인 마케팅을 위해서 주어진 정보 확산 모델 상에서 영향력이 큰 사용자들을 마케팅의 시발점으로 채택하는 것이 중요하다. 소셜 네트워크 상에서 한정된 k명의 사람에게 광고를 하였을 때, 그 사람들이 야기하는 입소문을 따라 미치는 영향력을 최대화하는 영향력 최대화 문제(Influence Maximazation Problem)를 생각해 볼 수 있다.

예를 들어, 소셜 커머스 회사에서 자신들이 팔 상품을 Twitter상에서 광고한다고 가정하자. 만약 한정된 k명의 트위터 사용자에게 공짜로 상품을 주면서 입소문을 퍼트려 주변인들이 해당 상품을 구매하도록 유도한다. 상품을 구매한 사용자들은 자신의 주변 사람들에게도 동일한 상품을 추천한다고 하자. 따라서 해당 상품을 성공적으로 홍보하기 위해선 최종적으로 최대한 많은 사람이 상품을 구매하도록 초기 k명의 시드 사용자를 선택해야 한다. 이는 초기에 어떠한 사람들을 선택하는지에 따라 소셜 네트워크 상에서 해당 상품에 대한 광고를 보고 최종적으로 몇명의 사용자가 해당 상품을 구매할지가 좌우되기 때문이다. 간단하게 생각해보면, 친구가 아무도 없는 사람보단 광고를 퍼트릴 수 있는 친구가 많은 사람을 택하는 것이 해당 광고가 널리 퍼질

[4]http://www.groupon.com/
[5]http://http://www.ticketmonster.co.kr/

수 있다는 것을 알 수 있다. 따라서 영향력 최대화 문제에서 초기 시드 사용자는 소셜 네트워크 상에서 영향력이 큰 k명을 선택하는 것을 기본으로 하고있다.

실제로, 국내 한 기업의 경우 자사 블로그[6]를 통해 다양한 자사 상품에 대해 지속적으로 소수의 인원으로 이루어진 체험단을 모집하여 각 인원에게 광고의 대상이 되는 상품을 지급하고, 각 체험단 인원이 작성한 상품 리뷰를 통하여 광고를 퍼트리는 식의 바이럴 마케팅을 진행하고 있다. 해당 광고는 소셜 네트워크 서비스 중 블로그를 통하여 바이럴 마케팅을 수행한다. 그러나 현재 체험단을 선정하는데 있어 각 지원자가 소셜 네트워크 서비스 상에서 어느정도의 영향을 미치는지 고려하지 않아 해당 광고가 다른 사람들에게 어느정도의 영향을 미칠지 예상하는 것이 어렵다. 이러한 상황에서 영향력 최대화 문제를 적용할 경우, 각 지원자 중 최대의 영향을 미칠 수 있는 k명의 체험단을 선택하여 해당 상품의 광고를 더욱 효과적으로 수행할 수 있는 장점이 있다.

1.2 연구의 내용

본 연구는 소셜 네트워크에 내재되어있는 커뮤니티 구조를 고려하여 영향력 최대화 문제를 푸는 방법을 다룬다. 소셜 네트워크는 다수의 커뮤니티로 이루어져 있으며, 소셜 네트워크 시각화를 포함한 다양한 응용에서 소셜 네트워크 상의 커뮤니티를 찾는 다양한 연구가 이미 진행되었다[2, 3]. 영향력 최대화 문제에서도 소셜 네트워크의 커뮤니티 구조를 고려하는 것이 중요하다.

[6] http://samsungtomorrow.com/

이는 소셜 네트워크 상의 커뮤니티 구조와 각 커뮤티니에서 영향력이 가장 큰 사용자를 알 수 있다면 해당 사용자에게 광고를 할 경우 커뮤니티 내의 대부분의 사람들이 광고를 접할 가능성이 높기 때문이다. 예를 들어, 사진 동호회에서 가장 활동이 활발한 사용자에게 새로운 카메라를 제공해주며 광고를 할 경우, 동호회의 다른 사람들이 자연스럽게 새 카메라에 대한 정보를 접할 수 있을 것이다. 이와 같이 어느 한 커뮤니티 내에서 주변 사람들의 구매를 유도하는 사람에게 광고를 할 경우, 해당 사람은 커뮤니티의 하위 구성원들에게 광고를 전달할 여지가 많다.

따라서 본 연구에서는 소셜 네트워크 상에서 커뮤니티를 찾고 그 중 가장 영향력이 높은 노드를 동시에 찾을 수 있는 마르코프 클러스터링[3]를 이용하여 그래프 클러스터링을 수행한다. 마르코프 클러스터링이 정확한 클러스터가 나오기까지 여러번의 반복적인 행렬 곱셈을 통해 클러스터링을 수행하기 때문에 $O(n^3)$의 복잡성을 가진다. 그러나 영향력 최대화 문제에 한하여 소셜 네트워크 내에서 정확한 커뮤니티 구조를 찾는 것 보다 다소 부정확한 커뮤니티일지라도 각 커뮤니티에서 가장 영향력이 높은 사용자를 찾아내는 것이 더 중요하다. 따라서 정확한 커뮤니티 구조를 찾아내는 것 보다는 커뮤니티 구조의 정확성을 다소 포기하면서 빠르게 소셜 네트워크 상에서 영향력이 큰 끌개들을 찾는 알고리듬을 제시하고, 해당 끌개들 중 최종 영향력을 최대화 할 수 있는 k개의 끌개를 선택함으로서 영향력 최대화 문제를 풀도록 한다.

해당 방법은 두가지 장점을 갖는다. 먼저, 이는 소셜 네트워크 그래프 상에서 선택될 수 있는 시드의 후보를 추려낼 수 있기 때문에 기존의 탐욕적 알고리듬이 모든 노드들에 대해 영향력을 계산하는 대신,

본 연구에서 제시하는 방법으로 얻어진 끌개들에 대해서만 탐욕적 알고리듬을 수행하여 기존 탐욕적 알고리듬의 수행 시간을 개선하는 하나의 휴리스틱으로 사용할 수 있다.

다음으로 기존의 휴리스틱에서 고려하지 않던 커뮤티니 구조를 고려함에 따라 기존의 휴리스틱에서 밀집된 시드 노드들이 정답으로 선택될 수 있는 현상을 방지하여 기존 휴리스틱이 갖는 성능적 단점을 보완하였다.

1.3 논문의 구성

본 논문의 구성은 다음과 같다. 2 장에선 정보확산 모델에 대한 설명 및 영향력 최대화 문제에 대한 개괄적인 설명 및 문제 정의를 한다. 다음으로 3 장에서는 정의된 정보확산 모델에서 영향력 최대화 문제를 푸는 기존의 알고리듬들을 탐욕적 접근법과 휴리스틱적 접근법으로 나누어 설명하고, 기존 연구들이 고려하고 있지 않은 점이 무엇인지 파악한다. 4 장에서는 그래프 클러스터링을 이용하여 소셜 네트워크 내에 내제되어있는 커뮤니티를 고려한 영향력 최대 알고리듬을 제시한다. 5 장에서는 본 논문에서 제시하는 알고리듬들의 성능을 기존 연구 및 휴리스틱과 비교하고 본 논문에서 제시한 알고리듬이 가지는 이점을 분석한다.마지막으로 6 장에서는 본 논문에서 기술한 연구의 결론과 향후 연구 방향을 제시한다.

제 2 장

문제 정의

이 장에서는 본 연구에서 대상으로 하고 있는 소셜 네트워크 그래프 모델이 무엇인지 설명하고, 소셜 네트워크 그래프 상에서 영향력 최대화 문제가 무엇인지 수학적으로 정의한다.

2.1 소셜 네트워크 그래프 및 정보 확산 모델

2.1.1 소셜 네트워크 그래프

소셜 네트워크는 노드와 변으로 구성된 가중 유향 그래프로 나타내며, 각 노드는 소셜 네트워크의 구성원을 의미하고 각 변(edge)은 구성원 사이의 상호작용을 의미한다. 가중 유향 그래프로 나타낸 소셜 네트워크의 정의는 다음과 같다.

정의 1. *가중 유향 그래프(weighted directed graph)* $G = (V,\ E)$*는 노들의 집합* V*와 변의 집합* E*의 순서쌍이다. 변* $e \in E$*는 두 노드* $u,\ v \in V$*의 순서쌍* $e = (u,\ v)$*로 표시하며, 노드* u*에서* v*로 향하는 변임을 의미한다. 변* $e = (u,\ v)$*는 구성원간의 상호작용 횟수인* $c_{u,\,v}$*를 가중치로 갖는다. 노드* $u \in V$*의 이웃 노드들의 집합* $N_G(u)$*은 다음과 같이 정의한다.*

$$N_G(u) = \{\ v\ inV \mid \exists (u,\ v) \in E\ \} \tag{2.1}$$

또한 추후에 설명할 마르코프 클러스터링이나 페이지랭크 알고

리듬을 위해 소셜 네트워크를 전이 행렬(transition matrix)로 표현하여 알고리듬을 수행하기 때문에, 위에서 정의한 소셜 네트워크 그래프를 정규화하는 과정이 필요하다. 노드 u에서 인접한 노드로 이동한다고 할 때, 노드 $v \in N_G(u)$로 이동할 확률은 각각 연결되어있는 변의 가중치에 비례한다.

$$Pr(v\|u) = \frac{(u,\ v).weight}{\sum_{w \in N_G(u)}(u,\ v).weight} \tag{2.2}$$

정의 2. *정규화 그래프 $\hat{G}$는 모든 변의 가중치를 식 2.2를 이용하여 정규화하여, 모든 변의 가중치의 합이 1이 되는 그래프를 말한다.*

$$\sum_{e=(u,\ v), v \in N_{\hat{G}}(u)} = 1 \tag{2.3}$$

2.1.2 정보 확산 모델

다음은 네트 워크 상에서 영향력이 확산되는 모델들에 대해서 설명한다. [4, 5, 6, 7, 8]는 기존 사회학 분야의 연구 중 소셜 네트워크 상에서 영향력 또는 입소문이 확산되는 모델에 대한 연구이다. 이 중 영향력 최대화 문제에 가장 널리 사용되는 두 모델을 소개한다.

첫째로, [6, 7, 9]에서 제시한 모델인 Independent Cascade(IC) 모델이 있다. 각 변을 통해 영향력이 확산될 확률을 고정값으로 정의하여 영향력 확산을 표현하는 IC 모델은 가장 많이 거론되는 정보 확산 모델로, 정의 3과 같은 방식으로 소셜 네트워크 상에서 정보를 전파한다.

정의 3. *영향력 확산 모델을 표현하기 위해 모든 노드들은 활성 혹은 비활설 상태 중 하나의 상태를 취한다. 활성 상태란 해당 노드가 영향*

을 받았음을 뜻한다. 영향력을 전파하는 초기 노드의 집합을 A_0라고 하고, A_0에 속한 노드들은 활성 상태를 갖는다. i번째 반복에 영향을 받은 노드, 즉 활성 상태로 전이된 노드들의 집합을 A_i로 정의한다. $i+1$번째 반복에서는 A_i에 속한 노드 u에 인접한 비활성 상태의 노드 v에게 각각 $p_{u,v}$의 확률로 영향을 미친다. u가 v에 성공적으로 영향을 미칠 경우 v는 A_{i+1}에 추가되며, v는 활성 상태로 전이한다. 해당 절차는 $A_{i+1} = \emptyset$일 때까지 반복한다.

노드 u가 v에게 정보를 전파할 확률인 $p_{u,v}$는 다음과 같이 정의한다.

$$p_{u,v} = 1 - (1-p)^{c_{u,v}} \tag{2.4}$$

위의 식에서 p 는 전파 확률(propagation probability)로, 한 노드가 자신에게 인접한 다른 노드 v에게 한번의 상호작용으로 영향력을 전파할 확률을 의미한다.

IC 모델상에선 연결정도(degree)가 높은 노드가 주변 노드들에게 영향력을 전파할 확률이 높고, 인접한 노드로부터 정도를 전파받을 확률또한 높다. 그러나 응용에 따라 인접한 노드들이 많은 노드는 각 인접한 노드로부터 영향력을 전파받을 확률이 낮을 수도 있다. 예를 들어, 실세계에서 친구가 100명인 사람이 100명 중 한명에게서 영향력을 받는 것보단 친구가 1명인 사람이 그 친구에게서 영향력을 받는 정도가 더욱 클 것이다. 이러한 직관에 부합하여 [10]에서 IC모델을 보완한 Weighted Cascade(WC) 모델을 제안하였다.

정의 4. *WC모델은 IC 모델과 동일하게 모든 노드들은 활성 혹은 비활성 상태 중 하나의 상태를 취한다. 영향력을 전파하는 초기 노드의 집합을 A_0라고 하고, A_0에 속한 노드들은 활성 상태를 갖는다. i번째*

반복에 영향을 받은 노드, 즉 활성 상태로 전이된 노드들의 집합을 A_i *로 정의한다.* $i+1$*번째 반복에서는* A_i*에 속한 노드* u*에 인접한 비활성 상태의 노드* v*에게 각각* $p_{u,v}$*의 확률로 영향을 미친다.* u*가* v*에 성공적으로 영향을 미칠 경우* v*는* A_{i+1}*에 추가되며,* v*는 활성 상태로 전이한다. 해당 절차는* $A_{i+1} = \emptyset$*일 때까지 반복한다.*

WC 모델에서 노드 u*가* v*에게 정보를 전파할 확률인* $p_{u,v}$*는 다음과 같이 정의한다.*

$$p_{u,v} = \frac{c_{u,v}}{\sum_{i \in N_G(v)} c_{i,v}} \tag{2.5}$$

2.2 영향력 최대화 문제

소셜 네트워크의 사용자 간 정보 교환을 통해 일어나는 입소문 현상을 최대화하기 위한 연구들이 많이 진행되어 왔다. Domingos와 Richardson[11, 12]이 처음으로 영향력 최대화 문제(Influence Maximization Problem)을 확률적 알고리듬 문제로 정의한 후, Kempe[10] 가 IC모델과 WC모델등의 정보 확산 모델 상에서 영향력을 최대로 하는 노드 k개를 찾는 문제를 최적화 문제로 정의하고 각 정보 확산 모델에 대해 영향력 최대화 문제가 NP-Hard라는 것을 수학적으로 증명하였다.

Kempe[10]가 정의한 최적화 문제 상에서 영향력 최대화 문제의 수학적 정의는 다음과 같다.

정의 5. *영향력 최대화 문제는 주어진 그래프* $G = (V, E)$*에 대해, 각* $(u, v) \in E$*에 특정한 정보 확산 모델에 따라 영향력 전파 확률이 정의된다고 할 때, 영향력 함수(influence function)* $\|S\| = k$*를 만족시키며* $f(S)$*를 최대화하는* $S \subseteq V$*를 찾는 문제이다.*

현재까지 IC 모델 및 WC 모델을 정보 확산 모델로 갖는 소셜 네트

워크 상에서 영향력 최대화 문제를 각각 알고리듬 문제[11, 12]와 최적화 문제[10, 13]로 정의하고 해답을 찾는 연구가 수행되었다.

제 3 장

관련 연구

3.1 탐욕적 접근법

Kempe가 제시한 탐욕적 알고리듬은 모든 노드들에 대해 각 노드가 미치는 영향력의 기대값을 몬테-카를로 시뮬레이션을 통해 계산하고, 각 iteration마다 가장 높은 영향력 기대값을 가지는 노드를 시드 집합 S에 추가한다. 이 알고리듬으로 인해 얻어진 해답은 영향력 함수 $f(S)$가 sub-modular한 성질을 갖기 때문에 최적해의 63%해를 보장한다는 것이 증명되었다 [10]. 해당 탐욕적 알고리즘에 대한 의사코드는 알고리듬 1과 같다.

Algorithm 1 `GeneralGreedy`(G, k)

Input : graph $G = (V,\ E)$, k for the number of seeds to be selected
Output : the set S that maximizes the influence spread

```
S = ∅
for i = 1 to k do
    for each vertex v ∈ V\S do
        s_v = 0
        for i = 1 to R do
            s_v += |f(𝒮 ∪ {v})|
        end for
        s_v = s_v / R
    end for
    S = S ∪ {arg max_{v∈V\S} {s_v}}
end for
return S
```

이렇듯 탐욕적 알고리듬은 해의 정확도를 보장할 수 있는 장점이 있으나, 몬테-카를로 시뮬레이션을 통해 영향력을 계산하기 때문에 시간이 오래 걸린다는 단점이 있다. 이는 주어진 시드 집합이 미치는 영향을 정확하게 계산할 수 없기 때문에 충분이 많은 몬테-카를로 시뮬레이션을 통해 영향력 확산에 대한 모의 실험을 하고, 영향력 확산에 대한 근사값을 구하기 때문이다. 몬테-카를로 시뮬레이션을 많이 수행할 수록 해당 시드 집합의 정확한 영향력을 근사할 수 있으나, 높은 정확도의 근사값을 구하기 위해서 각 시뮬레이션당 많은 시간이 걸리게된다.

이러한 문제를 해결하기 위해서 [14]에선 어떤 한 노드 u 의 영향력인 $f(u)$을 반복적으로 계산하는 것을 최소화 하여 실행시간을 단축시키는 방법을 제시하였다. 해당 연구는 우선순위 큐를 사용하여 영향력이 높은 노드들에 대해서만 영향력을 재계산 하는 방법인 비용효과적 게으른 전송(Cost-Effective Lazy Forward)을 이용한다. 이는 시드 집합에 새로운 노드를 추가하기 위해 모든 노드의 영향력을 계산할 필요 없이 우선순위 큐에서 가장 높은 영향력을 갖는 노드에 대해서만 영향력을 재계산한 후, 재계산 후에도 해당 노드가 가장 높은 영향력을 갖을 시 해당 노드를 시드 집합에 포함하는 방법을 사용하여 영향력 재계산 횟수를 크게 감소시켰다. CELF Greedy 알고리듬은 기존의 탐욕적 알고리듬보다 실행시간을 약700배 단축하였다. 그러나 시드 집합이 비어있을 경우 제일 첫 노드를 선택하기 위해선 기존 탐욕적 알고리듬과 동일하게 소셜 네트워크 상의 모든 노드들의 영향력을 계산해야 하기 때문에 실용적으로 사용하기엔 여전히 느리다는 단점이 있다.

[15]는 주어진 소셜 네트워크 그래프의 각 변을 선제거 하는 방

식 통해 노드 수에 비례하는 시간 내에 각 노드가 한번의 몬테-카를로 시뮬레이션을 통해 미칠 수 있는 영향력을 계산할 수 있는 방법을 제시하였다. 그러나 각 노드의 영향력 기대값을 계산하기 위해선 이러한 절차를 여러번 수행해야 하기 때문에 이 또한 몬테-카를로 시뮬레이션의 횟수가 늘어갈 수록 수행 시간이 높아져 실용적이지 않다는 문제점이 있다.

3.2 휴리스틱 접근법

탐욕적 접근방법의 수행 시간이 오래 걸림에 따라, 간단한 휴리스틱을 통해 영향력 최대화 문제를 풀어보려는 시도도 있다. 가장 대표적인 방법으로 연결정도 중앙성(degree centrality) 휴리스틱이 있다[16]. 이는 그래프 상에서 연결정도가 높은 노드가 소셜 네트워크 상에서 영향력이 높을 것이라는 직관에 따라 연결정도가 가장 높은 k 개의 노드를 선택하는 방식이다. 해당 알고리듬은 사회학에서 소셜 네트워크 상의 영향력 높은 노드를 찾기 위해 빈번하게 사용되는 휴리스틱이다. 이와 관련하여 [13]에서는 다른 휴리스틱에 비해 연결정도 중앙성 휴리스틱이 영향력 최대화 문제에서 더 좋은 성능을 보인다는 것을 실험적을 통해서 증명하였다.

그러나 연결정도 중앙성 휴리스틱의 경우 자신에게 인접한 노드들만 측정하는 것이기 때문에 소셜 네트워크의 위상을 충분히 반영하는 중심점 측도라고 보기 어렵다. 또한 단순하게 연결정도가 높은 노드만 선택하게되면 연결정도가 높은 노드들이 서로 인접하게 위치해있을 경우 비슷한 범위 내에서만 영향력을 미치게 된다는 단점이 있다.

이러한 점을 보완하기 위해 [15]에서 degree discount 휴리스틱(알고리듬 2)을 제안한다. 이는 초기 시드로 선택된 노드 u의 인접 노드인 $N_G(u)$의 연결정도를 하나씩 낮춰주어 이미 선택된 노드로 향하는 변을 제외한 변의 총 합이 가장 높은 k개의 노드를 선택하는 방식이다. degree discount heuristic은 단순한 연결정도 중앙성 휴리스틱보다 정확도 면에서 더욱 좋은 성능을 보이나, 이 또한 근본적으로 소셜 네트워크의 커뮤니티 구조를 고려하지 않아서 연결정도 중앙성 휴리스틱과 마찬가지로 서로 가까이 있는 노드를 선택할 가능성이 있다는 단점이 있다.

Algorithm 2 `DegreeDiscount Heuristic`(G, k)

Input : graph $G = (V,\ E)$, k for the number of seeds to be selected
Output : the set S that maximizes the influence spread

```
S = ∅
for each vertex v do
    compute its degree d_v
    dd_v = d_v
end for
for i = 1 to k do
    select u = arg max_v {dd_v | v ∈ V\S}
    S = S ∪ {u}
    for each neighbor v of u and v ∈ V\S do
        dd_v = dd_v − 1
    end for
end for
return S
```

마지막으로 [17]에서는 다른 중요한 노드들과 많이 연결되어 있는 top-k개의 노드를 선택하는 크(eigenvector centrality) 휴리스틱 (알고리듬 3)을 제안하였다. 이는 소셜 네트워크를 위에서 설명한 정규화 그래프로 나타낸 후 각 노드의 페이지랭크 값을 계산하여 해당 값이 가장 큰 k개의 노드를 시드 집합으로 선택하는 휴리스틱이다. Eigenvalue 휴리

스틱 또한 기존의 사회학분야의 연구에서 소셜 네트워크 상의 영향력 있는 사람을 찾기 위해 많이 사용되는 휴리스틱이다.

Algorithm 3 `PageRank Heuristic`(G, p ,k)

Input : Transition Matrix M, initial distribution $\overrightarrow{s}$, restart probability p, k for the number of seeds to be selected

Output : the set S that maximizes the influence spread

```
S = ∅
r⃗ = s⃗
repeat
    r⃗ = (1 − p) · r⃗ · M + p · s⃗
until r⃗ converges
for i = 1 to k do
    select u = arg max_v{r⃗_v | v ∈ V\S}
    S = S ∪ {u}
end for
return S
```

Chen은 [15]에서 앞으로 영향력 최대화 문제가 나가야할 방향으로 탐욕적 접근법을 갖는 알고리듬의 수행 시간을 향상시키는 것보다 기존 휴리스틱 방법의 정확성을 높이는 것이 더욱 바람직한 방법이라 설명하고 있다. 따라서 다음 장에선 기존의 휴리스틱의 정확성을 높이는 방법으로 커뮤니티 구조를 고려한 알고리듬을 제시하고, 해당 알고리듬의 정확성 및 수행 시간을 기존 연구들과 비교한다.

제 4 장

마르코프 클러스터링을 이용한 영향력 최대화 문제

4.1 마르코프 클러스터링

[3]에서 제시하는 마르코프 클러스터링은 Bioinformatics분야에서 널리 쓰이고 있는 그래프 클러스터링 기법이다. 마르코프 클러스터링은 간단한 직관을 이용하여 그래프를 클러스터링한다.

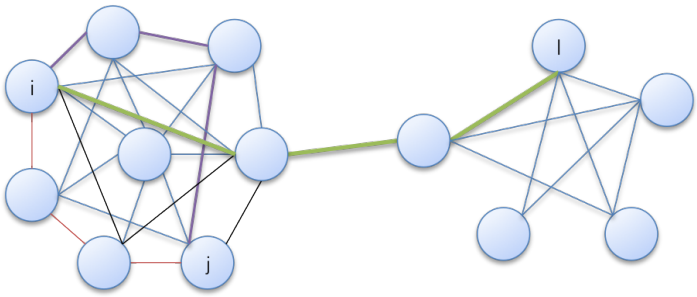

그림 1: 마르코프 클러스터링 예시

일반적으로 그래프 안에 어떠한 커뮤니티가 존재한다고 하면, 해당 커뮤니티 내의 노드에서 k번 전이하는 랜덤 워크를 실행했을 때 커뮤니티 내의 노드에 멈출 확률이 커뮤니티 외의 노드에서 멈출 확률보다 크다. 그림 1에서 볼 수 있듯이 i에서 랜덤워크를 수행하였을 때 랜덤워커가 j에 도달할 확률이 l에 도달할 확률보다 높다. 따라서 i 와 j는 같은 클러스터 내에 속할 가능성이 크다. 마르코프 클러스터링은 이

러한 직관을 이용하여 특정 횟수의 랜덤워크 후에 멈추는 노드가 같은 노드들끼리 클러스터링 하는 방법이다.

마르코프 클러스터링은 기본적으로 두가지 연산을 반복적으로 수행하여 클러스터링을 진행한다. 각각 연산은 expansion과 inflation으로 나눌 수 있다. 한번의 expansion과 inflation을 수행하는 것을 한번의 단계라 칭한다. 마르코프 클러스터링은 각 노드로부터 출발한 랜덤워커가 특정 노드에 멈출 확률을 expansion을 통해 계산한다. 여기서 expansion이란 앞에서 설명한 전이 행렬을 자기 자신과 곱하여 특정 노드로에서 길이가 2인 랜덤워크를 통해 다른 노드에 도달할 확률을 계산한다. 그림 2은 간단한 4×4행렬에서 expansion연산을 한번 수행한 예이다.

	1	2	3	4
1	0.25	0.33	0.33	0.5
2	0.25	0.33	0.33	0
3	0.25	0.33	0.33	0
4	0.25	0	0	0.5

	1	2	3	4
1	0.29	0.21	0.21	0.19
2	0.20	0.36	0.36	0.03
3	0.20	0.36	0.36	0.03
4	0.31	0.06	0.06	074

그림 2: Expand 연산

그 후, 마르코프 클러스터링은 단순하게 랜덤워크가 특정 노드로 수렴할 때까지 기다리는 것이 아니라 Inflation을 통해 전이할 확률이 높은 노드로 가는 전이 확률(transition probability)은 더욱 높이고, 반대로 전이할 확률이 낮은 노드로 가는 전이 확률을 더욱 낮춤으로써 랜덤워크가 더욱 빨리 수렴하도록 조종한다. Inflation은 전이 행렬의 각 원소를 사용자가 준 팽창률(inflation rate)만큼 제곱하는 계산을 수행한다. 그 후 전이 확률이 특정 임계치(threshold)에 미치지 못하는 변이

있을 경우 해당 변을 제거하고 행렬을 정규화 한다. 그림 3은 위의 예제에서 팽창률의 값을 2로 설정한 후 inflation연산을 수행한 결과이다. 해당 행렬에서 0.01보다 작은 전이 확률을 갖는 변은 제외하였다.

	1	2	3	4
1	0.29	0.21	0.21	0.19
2	0.20	0.36	0.36	0.03
3	0.20	0.36	0.36	0.03
4	0.31	0.06	0.06	074

	1	2	3	4
1	0.32	0.14	0.14	0.06
2	0.15	0.42	0.42	0.00
3	0.15	0.42	0.42	0.00
4	0.36	0.01	0.01	0.93

그림 3: Inflation 연산

이러한 작업은 두 연산에 의해 나온 전이 행렬이 수렴할 경우, 즉 i번째 단계와 i+1번째 단계의 상태 전이 행렬이 동일할 경우 종료하게 된다. 알고리듬 4는 앞서 설명한 과정의 의사코드로 설명한 것이다.

종료 후 나온 전이 행렬은 끌개와 해당 끌개에서 랜덤워커가 정지하는 노드들의 정보를 갖고있다. 마르코프 클러스터링이 종료된 후 얻을 수 있는 전이 행렬의 각 행은 출발 노드를 의미하며, 각 열은 도착 노드, 즉 끌개를 의미한다. 만약 어떠한 노드 u가 마르코프 클러스터링 수행 이후 끌개인 v의 클러스터에 속한다면 상태천이행 M의 u번째 행의 v번째 열은 0보다 큰 값을 갖는다. 따라서 한개 이상의 0보다 큰 값을 가지고 있는 열은 끌개임을 의미하며 해당 끌개로 모이는 노드의 개수를 해당 끌개가 갖는 클러스터의 크기라고 정의한다.

Algorithm 4 `Markov Clustering`(G, r)

Input : normalized graph $G = (V,\ E)$, inflation parameter r

$M = M(G)$
repeat
  $M = M^2$
  for $i \in V$ **do**
    for $j \in V$ **do**
      $M[i][j] = M[i][j]^r$
    end for
    for **do**
      if $M[i][j] < \theta$ **then**
        $M[i][j] = 0$
      end if
    end for
    for $j \in V$ **do**
      $M[i][j] = \frac{M[i][j]}{\sum_{k \in V} M[i][k]}$
    end for
  end for
until $Mconverges$
$H = graphinducedbynon - zeroentriesofM$
$C = clusteringinducedbyconnectedcomponentsofH$
return C

4.2 마르코프 클러스터링을 이용한 끌개 탐지

기존 마르코프 클러스터링 알고리듬은 그래프를 클러스터링하는 것을 주 목표로 하고있다. 그러나 영향력 최대화 문제에서는 정확한 그래프 클러스터를 구하는 것 보다 소셜 네트워크 상에서 영향력 있는 노드를 의미하는 끌개를 빠르게 구하는 것이 더 중요하다. 따라서 본 논문에서는 앞에서 설명한 마르코프 클러스터링을 개선하여 끌개만 빠르게 구하는 방법을 제시한다.

마르코프 클러스터링은 기본적으로 행렬 곱셉인 expansion 연산 때문에 매 단계당 $O(n^3)$에 비례한 수행 시간이 걸린다. 또한 수렴할 때

까지 iteration을 반복적으로 수행해야 하기 때문에 수행 시간이 오래 걸린다는 단점이 있다. 마르코프 클러스터링이 수행되는 절차를 살펴보면 끌개는 이른 시간에 결정이 나는 반면에 정확한 클러스터를 구하기 위해 수렴할 때 까지 반복적으로 수행한다. 앞서 설명한 바와 같이, 영향력 최대화 문제에선 정확한 클러스터보다 끌개를 더욱 빨리 찾는 것이 중요하기 때문에 최대한 적은 단계를 거쳐 끌개만 구한 후 마르코프 클러스터링을 정지하는 새로운 알고리듬을 제안하였다.

알고리듬 5는 기존 마르코프 클러스터링 알고리듬과 동일하게 진행된다. Expansion과 Inflation 연산을 반복적으로 수행하며 끌개가 될 수 있는 노드를 찾도록 한다. 설명의 편의를 위해 마르코프 클러스터링의 단계를 팽창 단계와 수축 단계로 나누도록 하자. 팽창 단계는 전이 행렬 내의 0보다 큰 가중치를 갖는 변이 늘어나는 단계을 뜻하고, 반대로 전이 행렬 내의 0보다 큰 가중치는 갖는 변이 줄어드는 단계를 수축 단계라고 한다. 팽창 단계에서는 여러번의 랜덤워크로 인해 도달할 수 있는 노드가 늘어남에 따라 전이 행렬 내에 새로운 원소가 추가되면서 0보다 큰 가중치를 갖는 변의 개수가 늘어난다. 그에 비해 수축 단계에서 랜덤워크로 인해 새로 생겨나는 변보다 inflation연산에서 특정 임계값을 넘지 못하는 변이 많아짐에 따라 총 변의 개수가 줄어들게 되는 것이다.

여기서 주의할 점은 수축 단계에서 마르코프 클러스터링의 inflation연산에 의해 전이 확률이 높은, 즉 끌개로 향하는 변은 더욱 높은 전이 확률을 갖게되고, 끌개가 아닌 다른 노드로 향하는 전이 확률이 낮은 변은 전이 확률이 더욱 더 낮아지는 경향을 보인다는 것이다. 그에 따라 수축 단계 도중 한 노드에서 자기 자신에 대한 전이 확률이 높아지는 노드는 끌개일 가능성이 높다. 그 이유는 자기 자신으로 향하

Algorithm 5 `AttractorDetection`(G, r)

Input : normalized graph $G = (V,\ E)$, inflation parameter r

```
M = M(G)
GrowingPhase = true
repeat
    prevNNZ = nnz(M)
    M = M^2
    for i ∈ V do
        for j ∈ V do
            M[i][j] = M[i][j]^r
        end for
        for  do
            if M[i][j] < θ then
                M[i][j] = 0
            end if
        end for
        for j ∈ V do
            M[i][j] = M[i][j] / Σ_{k∈V} M[i][k]
        end for
    end for
    if nnz(M) ¡ prevNNZ then
        GrowingPhase = false
        AC ← diag(M)
    end if
until GrowingPhase == true
AC = ∅
for i = 0toM.length do
    if M[i][i] ¿ AC[i] then
        AC ∪ i
    end if
end for
return AC
```

는 변의 전이 확률이 높을 경우 마르코프 클러스터링의 부익부 빈익빈 성질에 따라 자기 자신으로 전이할 확률이 지속적으로 높아지고, 해당 노드로 전이하는 노드들 또한 반복적으로 알고리듬을 수행함에 따라 해당 노드로 향하는 전이확률이 높아질 수밖에 없기 때문이다.

위에서 설명한 직관을 이용하여, 마르코프 클러스터링을 수행하는 도중 shrinking phase에 돌입하는 시점을 감지하여 팽창 단계의 마지막 iteration에서 전이 행렬의 대각행렬을 저장한 후, 수축 단계에서 해당 값이 증가하는 노드들을 끌개로 정한다. 해당 끌개 탐지 알고리듬을 이용하여 선별한 소셜 네트워크 상의 끌개 후보 집합(Attractor Candidate Set)을 AC라 명명하고, 구해진 끌개 후보 집합을 이용하여 영향력 최대화 문제에 대한 해답을 찾는 알고리듬에 응용할 수 있다.

4.3 끌개 탐색을 이용한 새로운 영향력 최대화 알고리듬

이 장에서는 앞에서 제시한 끌개 탐지 알고리듬을 이용하여 영향력 최대화 문제를 풀 수 있는 새로운 휴리스틱을 제시한다. 마르코프 클러스터링을 이용해 구한 끌개 후보들의 클러스터 크기를 이용하여 가장 큰 클러스터를 보유하고 있는 top-K개의 끌개를 선택하거나, 기존의 탐욕적 알고리듬이나 degree discount 휴리스틱을 구해진 끌개 후보들에 대해서만 적용한 새로운 휴리스틱을 제안한다.

4.3.1 MCL 휴리스틱

첫번째 휴리스틱은 앞서 설명한 끌개 후보들의 클러스터 크기가 가장 큰 top-K개의 끌개를 선택하는 휴리스틱이다. 이는 어떠한 클러스터의 끌개를 시드로 선택하였을 때, 해당 클러스터의 대부분의 노드에게 영향력이 퍼질 것이라는 직관하에 영향력이 퍼질 클러스터 크기를 최대화하는 휴리스틱이다. 알고리듬 6은 MCL 휴리스틱의 의사코드이다. 여기서 AttractorDetection 함수는 앞 장에서 제시한 끌개 후보 탐색

알고리듬을 의미한다. 뒤에서 설명할 MCL Greedy 휴리스틱이나 MCL Degree Discount 휴리스틱 또한 앞서 제시한 끌개 후보 탐색 알고리듬을 사용한다.

Algorithm 6 `MCL Heuristic`(G, k, r)

Input : graph $G = (V, E)$, k for the number of seeds to be selected, inflation rate r

Output : the set S that maximizes the influence spread

1: $S = \emptyset$
2: $AC = AttractorDetection(G, r)$
3: **for** $i = 1tok$ **do**
4: select $u = \arg\max_v \{clusterSize(v) | v \in AC \backslash S\}$
5: $S = S \cup u$
6: **end for**
7: **return** S

4.3.2 MCL Greedy 휴리스틱

두번째 휴리스틱은 마르코프 클러스터링을 이용해 구한 끌개 후보들에 대해 기존의 탐욕적 알고리듬을 수행하는 것이다. 기존의 탐욕적 알고리듬은 모든 노드들에 대해서 영향력을 계산해야 하기 때문에 수행 시간이 오래 걸린다는 단점이 있었으나, MCL Greedy 휴리스틱은 전체 노드 개수에 비해 적은 끌개 후보들에 대해서만 탐욕적 알고리듬을 수행하기 때문에 수행 시간을 대폭 개선할 수 있다.

그러나 여전히 몬테-카를로 시뮬레이션을 이용하여 영향력을 측정하기 때문에 끌개 후보들의 숫자가 늘어날 수록 MCL Greedy 휴리스틱 또한 수행 시간이 늘어난다. 소셜네트워크의 크기가 커질 수록 커뮤니티의 수가 늘어나고, 그에 따라 끌개 후보의 숫자도 늘어날 것이기 때문에, 앞서 소개한 MCL 휴리스틱과 다음에 소개한 MCL Degree Discount휴리스틱에 비해 비교적 높은 수행 시간을 보인다.

알고리듬 7은 위의 과정을 의사코드로 표현한 것이다.

Algorithm 7 `MCL Greedy Heuristic`(G, k, r)

Input : graph $G = (V, E)$, k for the number of seeds to be selected, inflation rate r

Output : the set S that maximizes the influence spread

```
S = ∅
AC = AttractorDetection(G, r)
for i = 1 to k do
    for each vertex v ∈ AC\S do
        s_v = 0
        for i = 1 to R do
            s_v += |f(𝒮 ∪ {v})|
        end for
        s_v = s_v / R
    end for
    S = S ∪ {arg max_{v∈AC\S}{s_v}}
end for
return S
```

4.3.3 MCL Degree Discount 휴리스틱

마지막으로 설명할 MCL Degree Discount 휴리스틱은 끌개 탐색 알고리듬과 기존의 휴리스틱 접근법 중 가장 좋은 성능을 보이는 Degree Discount 휴리스틱을 접목시킨 것이다. MCL Degree Discount 휴리스틱은 기존에 고려하지 않던 소셜 네트워크 상의 커뮤니티 구조를 고려한다. MCL Greedy 휴리스틱과 다르게 MCL Degree Discount 휴리스틱은 몬테-카를로 시뮬레이션을 사용하지 않기 때문에 MCL Greedy 보다 더욱 빠른 수행 속도를 갖는다.

Algorithm 8 `MCL Degree Discount Heuristic`(G, k, r)

Input : graph $G = (V,\ E)$, k for the number of seeds to be selected, inflation rate r

Output : the set S that maximizes the influence spread

```
S = ∅
AC = AttractorDetection(G, r)
for i = 1 to k do
    select u = arg max_v {dd_v | v ∈ AC\S}
    S = S ∪ {u}
    for each neighbor v of u and v ∈ AC\S do
        dd_v = dd_v − 1
    end for
end for
return S
```

제 5 장

성능 평가

이 장에서는 실제 소셜 네트워크 데이터에 대하여 기존의 영향력 최대화 알고리듬과 본 논문에서 제시하는 마르코프 클러스터링 기반의 알고리듬을 수행한 후 실험 결과를 비교하여 성능을 평가한다. 영향력 최대화 알고리듬의 성능은 각 알고리듬을 수행해 얻은 시드 노드 집합이 소셜 네트워크 상에서 최종적으로 영향력을 미치는 노드의 개수로 측정한다. 또한 각 알고리듬의 수행 시간을 비교하여 각 알고리듬이 속도 면에서 어떠한 장점 및 단점을 갖는지 살펴본다.

5.1 데이터셋

이 실험에서는 기존의 연구들[10, 13, 15]에서 사용한 'High Energy Physics - Theory Collaboration Network' 데이터셋과, 또 다른 실세계 데이터셋인 'Computational Geometry Collaboration Network'[18]를 사용한다. 실험 데이터셋으로 공저자 네트워크를 택한 이유는 실세계 페이스북 혹은 트위터 데이터셋은 그 크기가 방대하기 때문에 기존의 탐욕적 알고리듬을 수행하는데 너무 오랜 시간이 걸리기 때문에 실세계 소셜 네트워크 데이터셋을 이용하는 것은 현실적으로 불가능하다. 그에 비해 공저자 네트워크 일반적인 소셜 네트워크의 중요 특징을 내포하고 있고[19], 다양한 크기의 공저자 네트워크를 구하기 용이하여 공저자 네트워크를 실험에 사용하였다. 두 그래프 데이터 모두 논문의

저자들을 노드로 정의하고, 두 저자가 공저자 관계에 있을 경우 변으로 연결시킨다. 또한 각 변의 가중치는 같이 논문을 쓴 회수로 정의한다. 이제부터 각 데이터 셋의 이름을 HEPT와 GEOM으로 축약하기로 한다.

마지막으로 실세계 소셜 네트워크 데이터에 대해 해당 알고리듬들이 실제로 어떠한 성능을 보이는지 실험해보기 위해, 작은 크기의 페이스북 데이터에서도 실험을 진행하였다. 해당 그래프는 9개의 커뮤니티를 갖는 그래프로, NodeXL Graph Gallery[1] 에 공개된 소규모 페이스북 친구관계 그래프를 이용하였다. 해당 그래프의 이름을 FB로 축약하기로 한다.

각각 데이터셋의 노드수와 변의 개수는 다음과 같다.

표 1: 데이터셋 정보

graph data name	# of nodes	# of edges
HEPT	15,233	58,891
GEOM	7,343	11,898
FB	367	3,728

5.2 비교 대상

이 실험에서는 기존의 탐욕적 알고리듬 중 가장 빠른 new greedy 알고리듬[15], 휴리스틱 중 가장 좋은 성능을 보이는 degree discount 휴리스틱, 가장 간단한 방식인 무작위 선택, 그리고 마지막으로 페이지랭크를 이용한 고유벡터 중앙성 휴리스틱[20]를 본 논문에서 제시한 방법과 비교한다.

[1] http://wwwnodexlgraphgallery.org/Pages/Graph.aspx?graphID=584

5.3 실험 방식

이 실험은 두 데이터 셋에 대해 IC 모델 및 WC 모델상에서 영향력을 최대화 시키는 k개의 노드를 찾아내도록 한다. 이 실험에서 공통적으로 사용하는 매개변수는 k이며, k를 1부터 10까지 변화 시키며 실험을 진행하였다. 페이지랭크 값을 계산할 때 필요한 매개변수인 재시작 확률은 일반적으로 사용하는 15%로 사용한다 휴리스틱, MCL Greedy, MCL Degree Discount에서 사용하는 매개변수 팽창률은 1.1에서 2.0까지 변화 시키며 영향력 및 수행 시간에 미치는 영향을 실험하였다. 또한 탐욕적 알고리듬에서 한 노드의 영향력을 계산하기 위해 1000번의 몬테-카를로 시뮬레이션을 수행하였다.

5.4 기존 마르코프 클러스터링과 끌개 탐지 알고리듬 비교

이 실험에서는 마르코프 클러스터링을 끝까지 수행하는 것과 끌개 탐지 알고리듬을 사용하는 것의 장단점을 비교하려고 한다. 첫째로 마르코프 클러스터링을 끝까지 수행할 때 얻을 수 있는 끌개를 정답 집합으로 가정하고, 끌개 탐지 알고리듬을 이용하여 얻은 끌개를 결과 집합으로 두 데이터셋에서 정답과 결과의 precision과 recall을 계산하였다. precision과 recall은 식 5.1 및 식 5.2을 이용하여 계산하였다. 각각 마르코프 클러스터링을 끝까지 수행하여 얻은 끌개의 집합을 MCL, 끌개 탐지 알고리듬을 이용하여 얻을 끌개를 $eMCL$이라 하자.

$$precision = \frac{eMCL \cap MCL}{eMCL} \tag{5.1}$$

$$recall = \frac{eMCL \cap MCL}{MCL} \tag{5.2}$$

각각 HEPT 데이터셋과 GEOM 데이터셋에 대한 precision과 recall 값은 다음과 같다. FB 데이터셋은 그래프의 크기가 작아 eMCL과 MCL 휴리스틱간의 수행결과에서 유의미한 차이가 나지 않았다.

표 2: precision & recall

graph data name	precision	recall
HEPT	0.7692	0.7135
GEOM	0.8495	0.8234

실험 결과, HEPT와 GEOM 데이터셋에 대해 평균적으로 80%의 precision과 76%의 recall을 갖는다는 것을 볼 수 있다. 이는 끌개 탐지 알고리듬이 정확한 커뮤니티 구조를 구하는 것이 아니기 때문에 크기가 작은 커뮤니티들의 끌개를 찾지 못하기 때문이다.

다음으로 표 3에서는 각 방법의 수행 시간에 대해 살펴보도록 한다.

표 3: runtime comparison (sec)

graph data name	MCL	eMCL
HEPT	1058.97	235.11
GEOM	118.73	29.76

수행 시간을 비교하였을 때 eMCL의 경우 MCL에 비해 HEPT 데이터셋에서 약 7.45배, GEOM 데이터셋에서 약 4.87배 빠른 것을 확인할 수 있다. 실제로 기존의 마르코프 클러스터링과 끌개 탐지 알고리듬을 이용하여 영향력을 최대화하는 10개의 노드를 찾아본 결과, HEPT 데이터셋에서 10개 중 8개의 동일한 노드를 찾고, GEOM 데이터셋에선

10개 모두 동일한 노드를 찾았다. eMCL과 MCL의 성능 비교 실험은 추후 성능비교에서 더욱 자세히 기술하도록 한다. 이와 같이 영향력 최대화 문제에서 끌개 탐지 알고리듬이 기존 마르코프 클러스터링과 거의 동일한 끌개를 찾아내기 때문에 수행 시간을 대폭 개선할 수 있는 끌개 탐지 알고리듬이 유용하다는 것을 입증할 수 있다.

5.5 팽창률에 따른 알고리즘 성능 및 수행 시간 분석

이 실험에서는 마르코프 클러스터링에서 사용하는 파라미터인 팽창률이 영향력 최대화 문제에 어떠한 영향을 미치는지 알아본다. 일반적으로 마르코프 클러스터링에서 inflation rate가 1에 가까워질 수록 큰 크기의 클러스터를 소수 생성한다. 이와 반대로, 팽창률이 커질 수록 작은 크기의 클러스터를 여러개 생성한다. 따라서 큰 크기의 클러스터가 생성될 수록 끌개의 수가 줄어들어 너무 넓게 분포된 끌개들을 선택하게 된다. 반대로 크기가 작은 클러스터들이 여러개 생성될 경우, 의미없는 작은 커뮤니티가 여러개 생성되어 이 또한 영향력 최대화 문제의 성능을 감소시킨다. 이러한 경향은 그림 4에서 확인할 수 있다.

다음으로 팽창률이 각 알고리듬의 수행속도에 미치는지 영향을 알아본다. 마르코프 클러스터링 알고리듬은 팽창률이 높을수록 빠르게 수렴하는 성질을 가지고 있다. 따라서 MCL 휴리스틱이나 MCL Degree Discount 휴리스틱의 경우 팽창률이 높아질 수록 더욱 빠르게 종료하는 것을 볼 수 있다. 그러나 MCL Greedy 휴리스틱의 경우 팽창률이 높을 수록 생성되는 끌개가 많아지기 때문에 모든 끌개 후보에 대해서 영향력을 계산하기 위한 시간에 의해 더욱 느리게 종료한다.

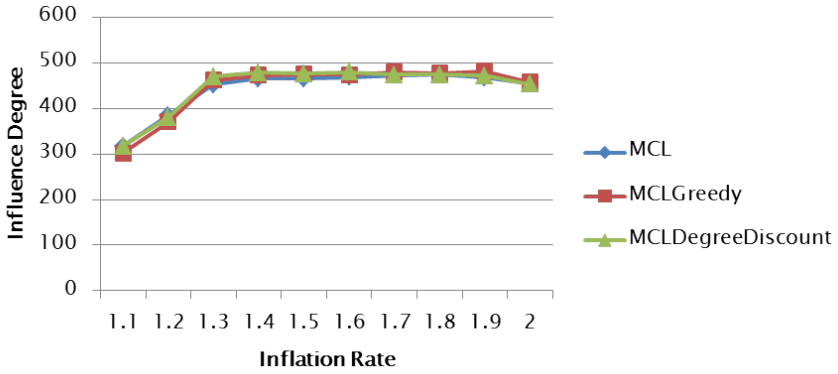

그림 4: 팽창률이 각 알고리듬 성능에 미치는 영향

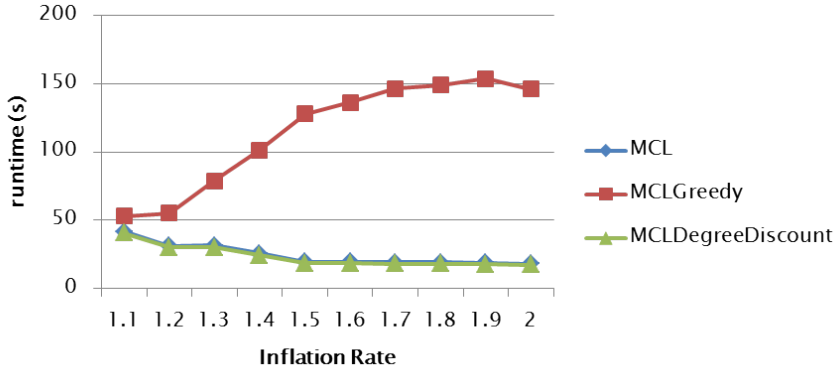

그림 5: 팽창률이 각 알고리듬 수행 시간에 미치는 영향

5.6 성능 비교

이 실험에서는 IC 모델과 WC 모델에서 k를 변화시키며 각 알고리듬이 소셜 네트워크 상에서 최종적으로 영향을 미치는 노드의 개수와 k를 10으로 고정시켰을 때 각 알고리듬의 수행시간을 비교해본다.

5.6.1 IC 모델에서의 성능 비교

그림 6과 그림 7은 IC모델 상에서 k를 변화하며 각 알고리듬들의 성능을 나타낸 그래프이다. 각 그래프를 보면 일반적으로 무작위 선별 휴리스틱의 성능이 가장 낮고, 페이지랭를 이용한 고유벡터 중앙성 휴리스틱이 그 다음으로 낮은 성능을 보인다. 그 위로 degree discount 휴리스틱이 자리하고 있고, MCL, eMCL, eMCL Greedy, eMCL Degree Discount, 탐욕적 알고리듬이 비슷한 성능을 보인다.

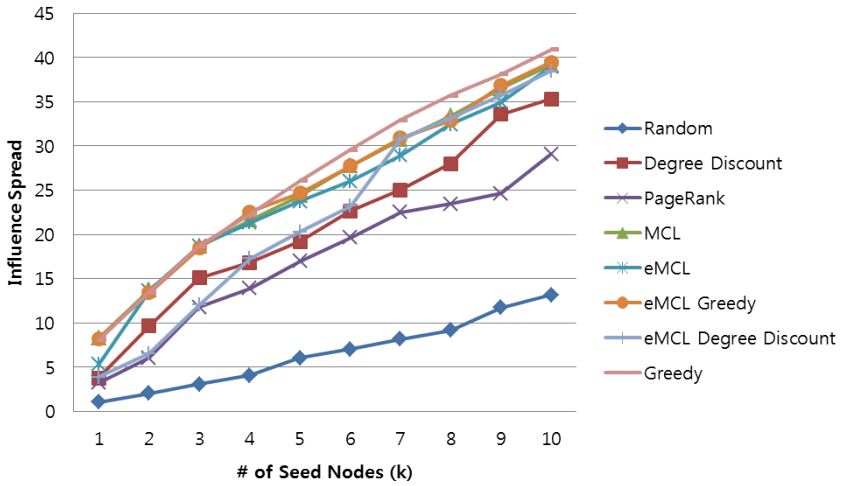

그림 6: IC 모델을 가정할 때 NetHept Collaboration Network에서의 영향력 전파 성능

k가 10일 때 그림 6에서 Degree Discount 휴리스틱에 비해 eMCL 휴리스틱, eMCL Greedy 휴리스틱, eMCL Degree Discount휴리스틱은 각각 7%, 8.3%, 5.4%의 성능 향상을 보였으며, 기존 탐욕적 알고리듬과 비교하여 평균적으로 95%의 성능을 보였다. 그림 7에선 본 논문에서 제시한 알고리듬이 Degree Discount 휴리스틱보다 각각 20.7%, 9.3%, 18.6%의 성능 향상을 보였으며, eMCL Greedy 휴리스틱이 탐욕적 알

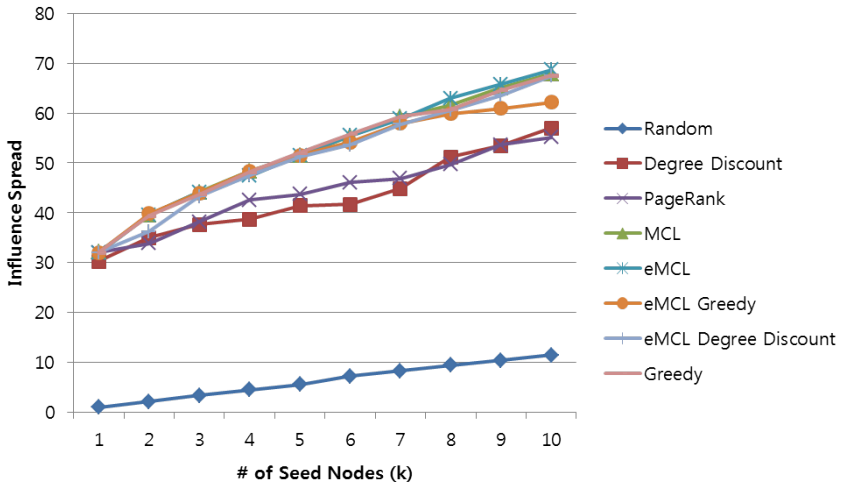

그림 7: IC 모델을 가정할 때 Computational Geometry Collaboration Network에서의 영향력 전파 성능

고리듬에 비해 92.1%의 성능을 보였으며, eMCL Degree Discount는 탐욕적 알고리듬과 동일한 성능을, eMCL 휴리스틱의 경우 탐욕적 알고리듬보다 1.7% 향상된 성능을 보였다. 또한 eMCL과 MCL알고리듬의 성능을 비교하였을 때, 두 알고리듬이 거의 동일한 성능을 보인다는 것을 알 수 있다.

추가적으로 위에서 언급한 페이스북 데이터셋에서 실행한 실험 결과는 다음과 같다.

그림 8에서 볼 수 있듯이, 실제 소셜 네트워크 데이터 상에서 실험을 수행하였을 때, 커뮤니티 구조를 고려하는 것이 기존 휴리스틱에 비해 성능을 높일 수 있다는 것을 알 수 있다. 그래프 상에서 초기 시드 선택에 의한 차이는 적으나, 시드의 개수가 많아질 수록 기존 시드와 다른 커뮤니티에 속해있는 노드를 선택하기 때문에 성능의 차이가 확연한 것을 볼 수 있다. k를 10으로 고정하였을 때, 기존 탐욕적 알고리듬

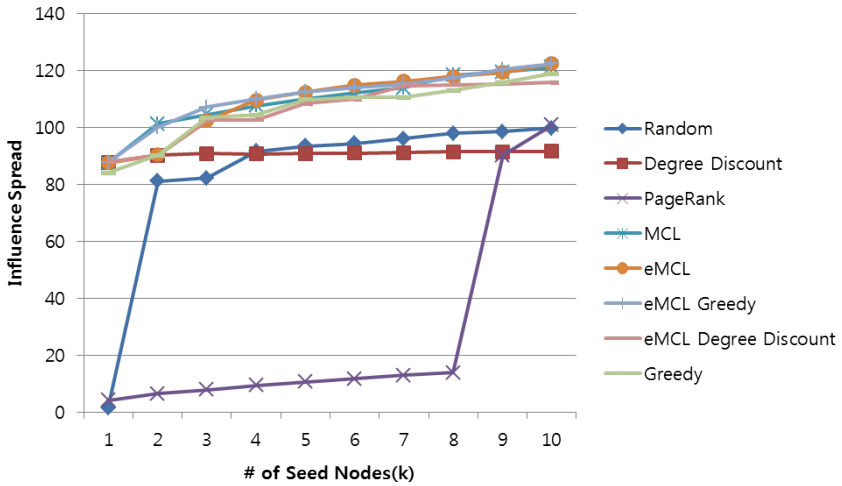

그림 8: IC 모델을 가정할 때 Facebook 데이터셋에서의 영향력 전파 성능

에 비해 eMCL Greedy가 비슷한 성능을 보이고, eMCL Degree Discount가 약 97.4%의 성능을 보였다. 그 외 MCL및 eMCL이 탐욕적 알고리듬에 비해 각각 2.9% 와 3%향상된 성능을 보였고, 그에 비해 Degree Discount 휴리스틱의 경우 탐욕적 알고리듬에 비해 77.1%의 성능을 보였다.

수행 시간과 관련하여, 그림 9에서 볼 수 있듯이 탐욕적 알고리듬이 수행 시간이 가장 길고, 탐욕적 알고리듬을 부분적으로 사용하는 eMCL Greedy 휴리스틱이 그 다음으로 오래 걸리는 것을 볼 수 있다. 탐욕적 알고리듬의 경우 수행시간이 k에 대해서 영향을 받기 때문에 더욱 큰 k에 대해선 수행시간이 더 오래 걸릴 것으로 짐작할 수 있다. 본 논문에서 제시한 알고리듬들의 경우 eMCL과 eMCL Degree Discount 휴리스틱은 기존의 탐욕적 알고리듬보다 약 15배 빠른 수행시간으로 동일하거나 더 좋은 성능의 시드 집단을 얻을 수 있었고, eMCL Greedy

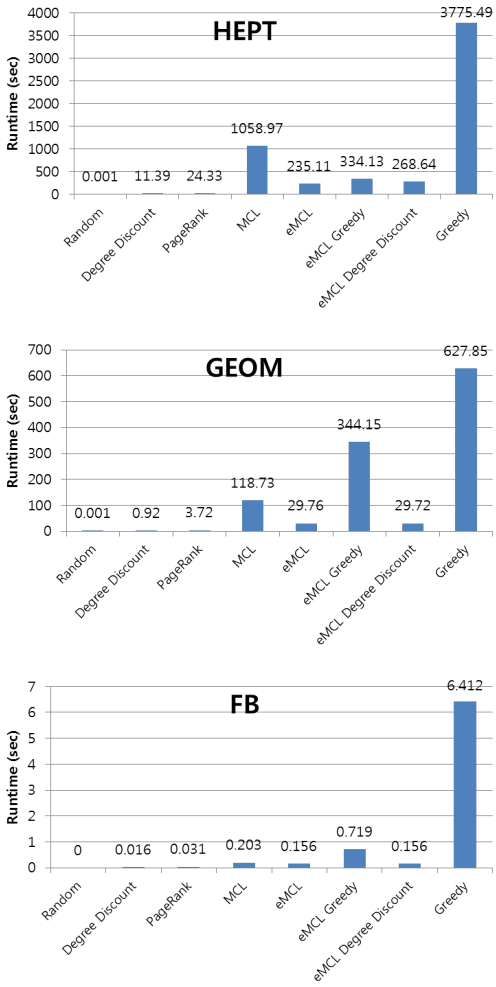

그림 9: IC 모델에서 각 알고리듬 수행시간

휴리스틱도 약 11배 빠른 수행시간으로 탐욕적 알고리듬에 준하는 성능을 보였다. GEOM 데이터셋에서도 MCL과 eMCL은 동일한 성능을 보였다.

5.6.2 WC 모델에서의 성능 비교

그림 10과 그림 11는 WC 모델 상에서 k를 변화하며 각 알고리듬들의 성능을 나타낸 그래프이다. IC 모델상에서와 거의 동일한 성향을 보이고 있다. 각 그래프를 보면 일반적으로 무작위 선별 휴리스틱의 성능이 가장 낮고, 페이지랭크를 이용한 고유벡터 중앙성 휴리스틱이 그 다음으로 낮은 성능을 보인다. 그 위로 degree discount 휴리스틱이 자리하고 있고, MCL, eMCL, eMCL Greedy, eMCL Degree Discount, 탐욕적 알고리듬이 비슷한 성능을 보이는 것을 볼 수 있다. WC 모델에서도 IC 모델과 마찬가지로 eMCL과 MCL알고리듬은 거의 동일한 성능을 보였다.

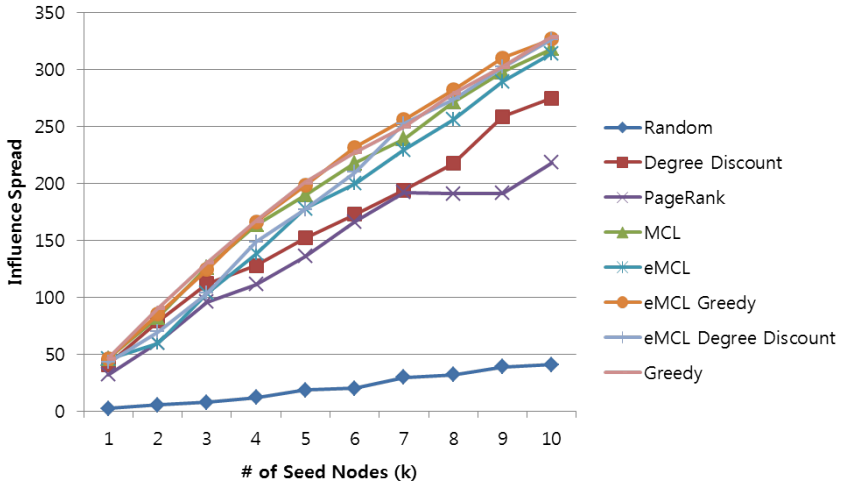

그림 10: WC 모델을 가정할 때 NetHept Collaboration Network에서의 영향력 전파 성능

k가 10일 때 그림 10에서 Degree Discount 휴리스틱에 비해 eMCL 휴리스틱, eMCL Greedy 휴리스틱, eMCL Degree Discount휴리스틱은 각각 7.9%, 13%, 12.9%의 성능 향상을 보였으며, 기존 탐욕적 알고리

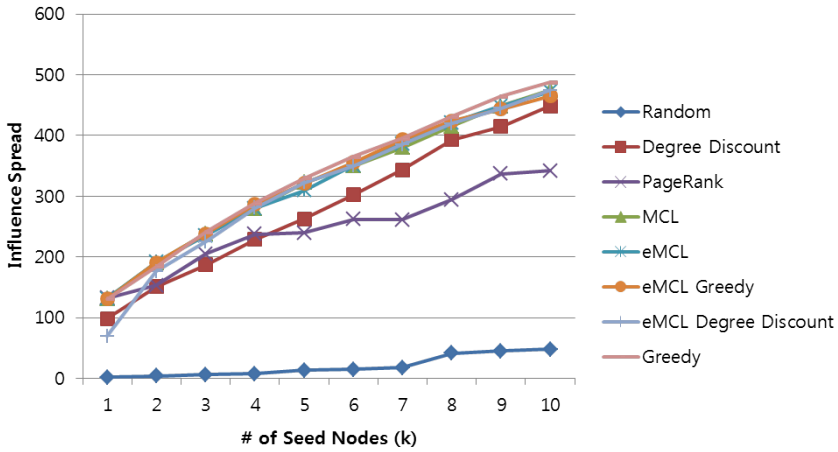

그림 11: WC 모델을 가정할 때 Computational Geometry Collaboration Network에서의 영향력 전파 성능

듬과 비교하여 eMCL 휴리스틱의 경우 94.8%의 성능을, 나머지 휴리스틱의 경우 99.3% 의 성능을 보였다. 그림 11에선 본 논문에서 제시한 알고리듬이 Degree Discount 휴리스틱보다 각각 5%, 3.4%, 5.4%의 성능 향상을 보였다. WC 모델상에서 GEOM 데이터셋에선 비교적으로 k가 낮을 때 Degree Discount 휴리스틱과 본 논문에서 제시한 휴리스틱의 성능 차이가 높았다. 탐욕적 알고리듬과 비교하였을 때, 세 휴리스틱 모두 평균적으로 96.9%의 성능을 보였다.

추가적으로 위에서 언급한 페이스북 데이터셋에서 실행한 실험 결과는 다음과 같다.

앞서 IC모델에서의 실험과 같이, WC모델에서도 비슷한 경향을 보인다는 것을 알 수 있다. 그림 12에서도 그래프 상에서 초기 시드 선택에 의한 차이는 적으나, 시드의 개수가 많아질 수록 본 논문에서 제시한 알고리듬이 더욱 좋은 성능을 보였다. k를 10으로 고정하였을

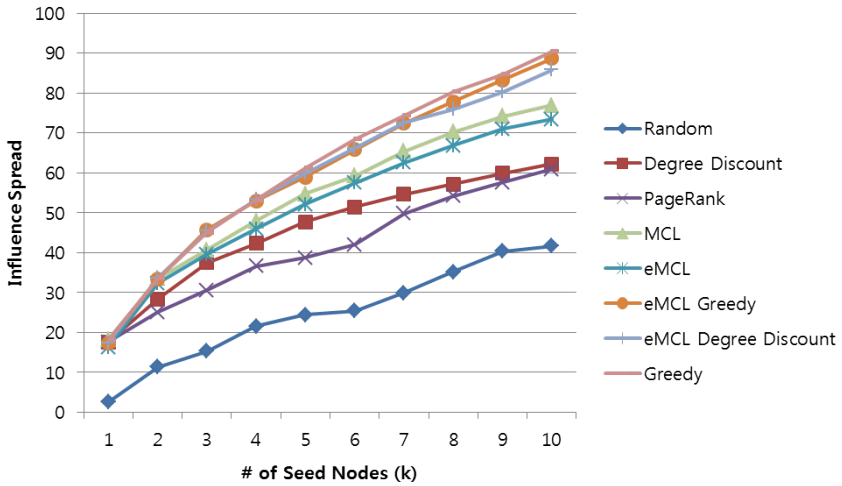

그림 12: WC 모델을 가정할 때 Facebook 데이터셋에서의 영향력 전파 성능

때, 기존 탐욕적 알고리듬에 비해 eMCL Greedy가 비슷한 성능을 보이고, eMCL Degree Discount가 약 94.9%의 성능을 보였다. 그 외 MCL 및 eMCL이 탐욕적 알고리듬에 비해 85.1%, 81.2%의 성능을 보였고, Degree Discount 휴리스틱의 경우 68.7%의 성능을 보였다.

수행 시간의 경우 그림 13에서 나타나듯이 탐욕적 알고리듬이 가장 수행 시간이 길었다. WC모델이 IC 모델에 비해 많은 노드에 영향력을 미치기 때문에 각 노드의 영향력을 계산하는 것이 더욱 오래 걸리고, 해당 시뮬레이션을 1000번 반복하기 때문에 IC 모델에서의 탐욕적 알고리듬 수행 시간보다 더욱 오래 걸린다. eMCL과 eMCL Degree Discount 휴리스틱은 기존의 탐욕적 알고리듬보다 평균적으로 46배 빠른 수행시간으로 비슷한 성능의 시드 집단을 얻을 수 있었고, eMCL Greedy 휴리스틱도 약 24.4배 빠른 수행시간으로 탐욕적 알고리듬에 준하는 성능을 보였다.

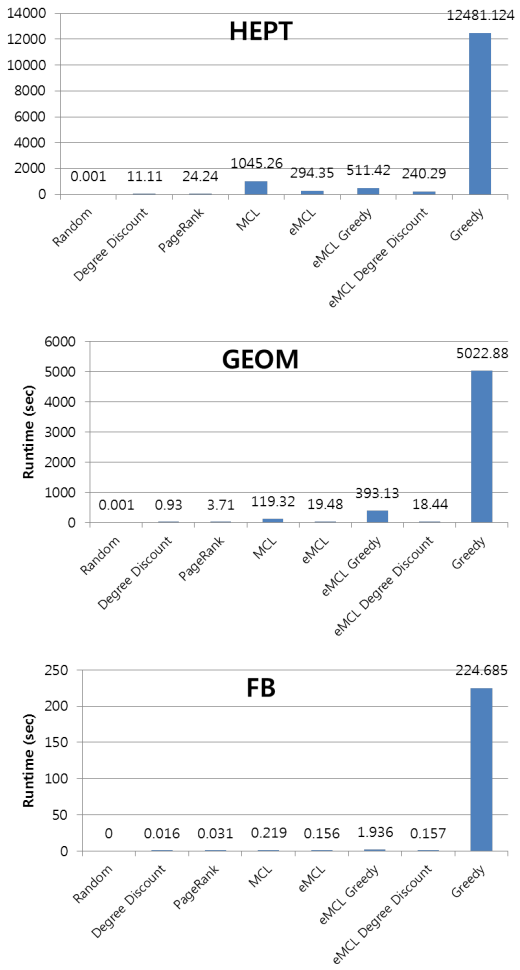

그림 13: WC 모델에서 각 알고리듬 수행시간

5.7 그래프 크기와 수행 시간과의 관계

마지막으로 그래프의 크기에 따라 각 알고리듬의 수행 시간이 증가하는 추세를 실험을 통해 비교해본다. 그림 14는 그래프의 크기를 3000개의 노드 부터 15000여개의 노드까지 증가시키며 각 알고리듬의 수행 시간 증가를 그래프로 나타내었다. 기본적으로 Degree Discount

나 Random같은 단순한 휴리스틱의 경우 노드 크기에 따른 수행 시간 증가가 미미하다. 그에 비해 탐욕적 알고리듬은 그래프의 크기가 커질수록 수행 시간이 가장 많이 증가하고 있다. 탐욕적 알고리듬에 비해 본 논문에서 제시한 휴리스틱은 수행 시간의 증가량이 4배 가량 차이 나는 것을 볼 수 있다. 그래프의 크기가 더욱 커질 수록 탐욕적 알고리듬의 수행 시간이 빠르게 증가하기 때문에 탐욕적 알고리듬의 확장성(scalability)가 낮다는 것을 알 수 있다. 그렇기 때문에 기존의 휴리스틱을 더욱 좋은 성능을 내도록 개선하는 것이 영향력 최대화 문제를 해결하는데 있어 더욱 적합하다고 할 수 있다.

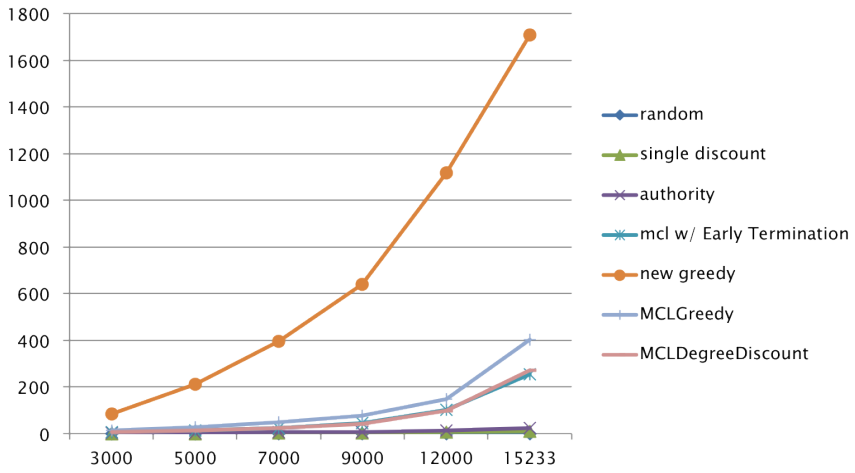

그림 14: 그래프 크기에 대한 알고리듬 수행 시간 증가 비교

제 6 장

결론 및 향후연구

6.1 결론

본 연구에서는 기존의 영향력 최대화 알고리듬들이 소셜 네트워크 상의 커뮤니티 구조를 고려하지 않는다는 점에 착안하여, 커뮤니티 구조를 고려하는 영향력 최대화 알고리듬을 제시하였다. 먼저, 마르코프 클러스터링을 개선하여 영향력 최대화 문제에서 중요시되는 끌개를 빠르게 찾아내는 알고리듬을 제시하였다. 이는 영향력 최대화 문제에서 정확한 커뮤니티의 구조를 아는 것이 중요하지 않기에 끌개 후보를 얻은 후 추가적인 계산을 하지 않아 수행 시간을 줄인다는 장점이 있다. 실제로 끌개 탐지 알고리듬이 기존 마르코프 알고리듬에 비해 개선된 수행 시간을 갖는다는 것을 실험을 통해 증명하였다.

다음으로 끌개 탐지 알고리듬을 이용하여 영향력 최대화 문제를 해결하는 알고리듬을 제시하였다. 구체적으로, 클러스터 크기를 이용한 휴리스틱을 제시하였고, 끌개 탐지 알고리듬과 기존 탐욕적 알고리듬 및 Degree Discount 휴리스틱을 접목한 알고리듬들을 제시하였다. 또한 소셜 네트워크에서 커뮤니티 구조를 고려하는 알고리듬이 기존의 휴리스틱에 비해 좋은 성능을 보인다는 것을 증명하였다.

또한 기존의 탐욕적 알고리듬이 수행시간의 문제로 인하여 실제 트위터나 페이스북 등의 소셜 네트워크 그래프에서 동작할 수 없는 단점에 비해, 본 논문에서 제시한 알고리듬은 탐욕적 알고리듬보다 더욱

큰 크기의 소셜 네트워크 그래프에서도 탐욕적 알고리듬과 비등한 영향을 가지는 노드들을 효과적으로 찾아낼 수 있을 것이라 기대할 수 있다. 따라서 실세계 소셜 네트워크 상에서도 영향력 최대화 문제를 적용하여 더욱 효율적인 바이럴 마케팅을 수행할 수 있을 것이라 기대된다.

앞선 예에 비롯해볼 때, 첫째, 기존 탐욕적 알고리듬에 비해 더욱 큰 크기의 소셜 네트워크에서 영향력을 최대화하는 k명의 체험단을 효율적으로 선택할 수 있는 장점이 있다. 둘째, 커뮤니티 구조를 고려함으로써 기존의 휴리스틱적 접근보다 더욱 큰 영향력을 미치는 사용자를 체험단으로 선택할 수 있어, 속도면이나 성능면에서 다른 알고리듬보다 더욱 경쟁력을 갖는 것을 기대할 수 있다.

6.2 향후 연구

향후 연구로는 다음과 같은 점을 개선하려고 한다. 첫째, 마르코프 클러스터링의 확장성을 개선하여 대용량 그래프에서도 효율적으로 클러스터링을 수행할 수 있도록 개선하고자 한다. 현재 마르코프 클러스터링이 Degree Discount 휴리스틱에 비해 수행 속도가 오래 걸리지만, 분산 시스템 상에서 마르코프 클러스터링을 수행할 수 있도록 개선하게 되면 더욱 빠른 시간 내에 영향력 최대화 문제를 풀 수 있을 것이다.

둘째, 앞서 제시한 끌개 탐지 알고리듬을 Shrinking Phase가 시작하자마자 종료하지 않고 사용자가 설정한 precision과 recall값을 보장하면서 최대한 빠르게 종료하도록 개선하는 것을 생각해볼 수 있다. 사용자가 직접 수행 시간과 성능 중 더욱 중요한 쪽을 선택함으로써

사용자가 처한 상황에 맞게 영향력 최대화 문제를 제어할 수 있을 것이라 기대된다.

참고 문헌

[1] A. L. Montgomery, "Applying quantitative marketing techniques to the internet," 2001.

[2] M. E. J. Newman, "Detecting community structure in networks," *The European Physical Journal B - Condensed Matter and Complex Systems*, vol. 38, pp. 321–330, Mar. 2004.

[3] S. Dongen, "A cluster algorithm for graphs," tech. rep., Amsterdam, The Netherlands, The Netherlands, 2000.

[4] F. M. Bass, "A new product growth for model consumer durables," *Management Science*, vol. 15, no. 5, pp. 215–227, 1969.

[5] J. J. Brown and P. H. Reingen, "Social ties and word-of-mouth referral behavior," *Journal of Consumer Research*, vol. 14, pp. 350–62, December 1987.

[6] J. Goldenberg, B. Libai, and E. Muller, "Talk of the network: A complex systems look at the underlying process of word-of-mouth," *Marketing Letters*, 2001.

[7] J. Goldenberg, "Using complex systems analysis to advance marketing theory development : Modeling heterogeneity effects on new product growth through stochastic cellular automata," *Academy of Marketing Science Review*, vol. 2001, no. 9, p. 1–18, 2001.

[8] M. Kimura and K. Saito, "Tractable models for information diffusion in social networks," in *PKDD'06*, pp. 259–271, 2006.

[9] D. López-Pintado, "Diffusion in complex social networks," *Games and Economic Behavior*, vol. 62, pp. 573–590, March 2008.

[10] D. Kempe, J. Kleinberg, and E. Tardos, "Maximizing the spread of influence through a social network," in *Proceedings of the ninth ACM*

SIGKDD international conference on Knowledge discovery and data mining, KDD '03, (New York, NY, USA), pp. 137–146, ACM, 2003.

[11] P. Domingos and M. Richardson, "Mining the network value of customers," in *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '01, (New York, NY, USA), pp. 57–66, ACM, 2001.

[12] M. Richardson and P. Domingos, "Mining knowledge-sharing sites for viral marketing," in *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '02, (New York, NY, USA), pp. 61–70, ACM, 2002.

[13] D. Kempe, J. Kleinberg, and Éva Tardos, "Influential nodes in a diffusion model for social networks," in *IN ICALP*, pp. 1127–1138, Springer Verlag, 2005.

[14] J. Leskovec, A. Krause, C. Guestrin, C. Faloutsos, J. VanBriesen, and N. Glance, "Cost-effective outbreak detection in networks," in *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '07, (New York, NY, USA), pp. 420–429, ACM, 2007.

[15] W. Chen, Y. Wang, and S. Yang, "Efficient influence maximization in social networks," in *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '09, (New York, NY, USA), pp. 199–208, ACM, 2009.

[16] S. Wasserman and K. Faust, vol. 63. Cambridge University Press, 1994.

[17] W. Chen, Y. Yuan, and L. Zhang, "Scalable influence maximization in social networks under the linear threshold model," in *ICDM*, pp. 88–97, 2010.

[18] V. Batagelj and A. Mrvar, "Pajek datasets," 2006.

[19] M. E. J. Newman, "The structure of scientific collaboration networks," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 98, pp. 404–409, Jan. 2001.

[20] S. Brin and L. Page, "The anatomy of a large-scale hypertextual web search engine," in *Proceedings of the seventh international conference on World Wide Web 7*, WWW7, (Amsterdam, The Netherlands, The Netherlands), pp. 107–117, Elsevier Science Publishers B. V., 1998.

Abstract

Influence Maximization Algorithm Using Markov Clustering

Chungrim Kim
School of Computer Science & Engineering
The Graduate School
Seoul National University

Social Network Services are known as a effective marketing platform in that the customers trust the advertisement provided by their friends and neighbors. Viral Marketing is a marketing technique that uses the pre-constructed social networks to perform maketing with small cost while maximizing the spread. Therefore, which seed user to select is the primary concern in viral marketing. Influence maximization problem is a well known problem to find the top-k seed users who can maximize the spread of information in a social network.

Since obtaining the global optimal solution for the influence maximization problem is proven to be NP-Hard, many greedy as well as heuristic approach has been researched. However, greedy approaches take to much time to obtain the seed node, whereas the heuristic approaches show poor performance.

To remedy such problems, we exploit the community structures in the social network to enhance the performance of the heuristic approaches. We perform markov clustering to find the natural communities in the social network and consider the most influential user in the community as the candidate for the top-k seeds.

Also, we propose a novel attractor identification algorithm that finds the influential nodes in the community with reduced runtime, and 3 new hybrid approaches for influence maximization problem. Experiments show that the proposed algorithms are more scalable than the greedy approaches, whereas the influence spread obtained by those outperforms the heuristic approaches.

Keywords : Influence Maximization, Graph Clustering, Markov Clustering
Student Number : 2010-23255

감사의 글

연구실에 들어온 지 벌써 2년이 지나 졸업을 하고 정든 연구실을 떠나게 되었습니다. 눈 깜짝할 새에 지나간 대학원 생활동안 많은 것을 배울 수 있었고, 좋은 사람들도 만날 수 있는 계기가 되었습니다. 그 동안 저에게 힘을 주고 도움이 되어주신 많은 분들께 감사 인사를 드리고 싶습니다.

먼저 저의 대학원 생활동안 학문적으로, 그리고 인간적으로도 많은 가르침을 주신 이상구 교수님께 감사드립니다. 교수님께서 지도해주신 덕분에 이 논문을 완성할 수 있었습니다. 연구실을 떠나서도 교수님의 조언과 가르침을 항상 마음에 지니며 생활하고, 교수님의 가르침에 부끄럽지 않은 사람이 될 수 있도록 노력하겠습니다.

연구실에서 만난 좋은 사람들이 있어 더욱 행복하게 연구실 생활을 할 수 있었습니다. 항상 관심을 가지고 저를 돌봐주신 재원이형, 동주형, 승석이형, 상근이형, 재휘형, 기현이형, 재석이형, 성찬이형, 종흠이형, 영기에게 감사를 드립니다. 그리고 저와 함께 일하느라 고생한 연찬이형, 병주형 그리고 동진이 형에게 감사를 드립니다. 석사과정을 함께한 민석이, 성은이, 상일이 그리고 용진이가 졸업할 때 마다 연구실의 빈 자리가 더욱 커지게 느껴진 것 같습니다. 물론 마지막까지 자기 이름이 안나와서 삐져있을 인범이에게도 감사합니다.

제 결정을 믿어주고 후원해주신 부모님과 알아서도 자기 일을 잘하는 동생 하림이에게 깊은 감사 드립니다. 가족들 덕분에 무사히 졸업을 할 수 있었습니다. 오랜 시간동안 항상 옆에서 응원해주고 힘을 준 수연이에게도 이 자리를 빌어 고맙다는 말을 전하고 싶습니다. 마

지막으로 저와 오랜 생활을 함께 한 친구들과 미처 언급하지 못한 다른 분들께도 감사 드립니다. 앞으로도 유쾌하고 열심히 사는 사람이 되겠습니다.