

# Node2Vec: Scalable Feature Learning for Networks



Aditya Grover, Jure Leskovec  
KDD '16: Proceedings of the 22nd ACM SIGKDD

21.03.03 논문 리뷰, 이보현

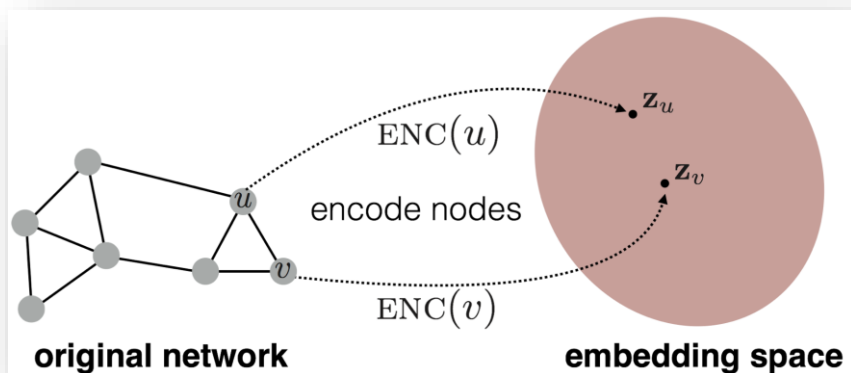


# INDEX 목차

1. INTRODUCTION
2. RELATED WORK
3. FEATURE LEARNING FRAMEWORK
4. EXPERIMENTS
5. DISCUSSION AND CONCLUSION



## • 기존 feature representation learning 학습의 문제점



### feature representation learning

: 그래프의 feature(특징)을 잘 표현하는 임베딩 벡터를 얻는 것이 목표

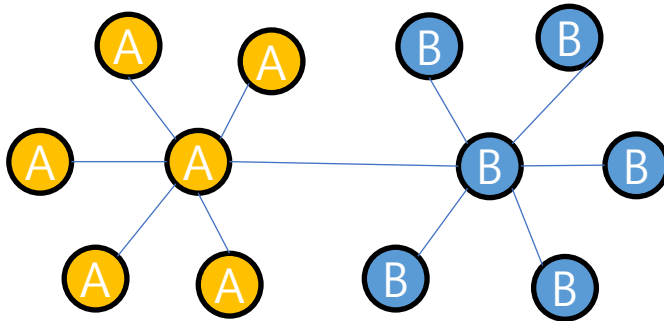
< 기존 feature representation learning 학습 방법(비지도 학습) >

접근법	종류	문제점
Dimensionality Reduction	Laplacian eigenmaps	고유값 분해로 인한 시간 소요가 큼. 과적합(overfitting)으로 인한 후속과제 성능 저조
노드 간 연결성 기반 기술	Deep walk, Line	효율적인 알고리즘이지만, <b>그래프 고유의 연결 패턴</b> 이 잘 반영되지 않은 접근 방법



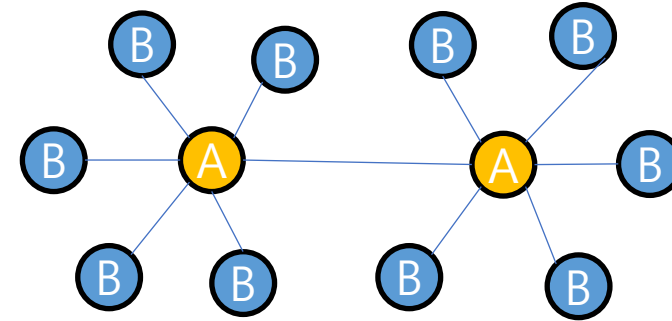
## • 기존 feature representation learning 학습의 문제점

### 그래프 고유의 연결 패턴



**Homophily**

: 그래프에서 가까이 위치하는 노드들이 유사한 경우



**Structural equivalence**

: 그래프에서 비슷한 이웃 구조를 가진 노드들이 유사한 경우

-> 실제 세상의 그래프는 이 두 패턴(homophily, structural equivalence)이 혼합되어 나타남.

-> feature representation learning 과정에 이를 반영할 필요가 있음

목표 : 그래프에 상관 없이, flexible 하고 general 한 feature representation learning 방법 제시



## • Word2vec

### Word2Vec

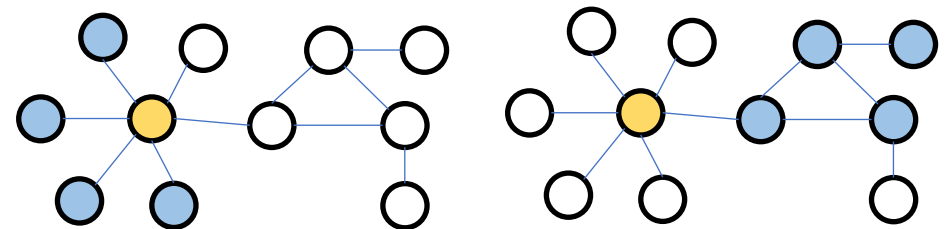
Skip-gram  
input data → Embedding

- 가점  
: 어느 한 단어로 부터 가까이에 위치하는 단어들은 높은 유사도를 가진다.

### Node2Vec

Sampling  
strategy → Embedding

- 가점  
: 어느 한 노드로부터 sampling 된 노드들은 높은 유사도를 가진다.
- 그래프는 순서가 있는 데이터가 아님
- Sampling Strategy  
: 유사 노드를 선택하는 방법



## • Node2vec 목적 함수

로그 가능도 최대화  
(maximizes the log-probability)

U 노드를 임베딩 벡터로 표현했을 때,  
U 노드의 유사 노드가 그 근처에서 관찰될 확률

$$\max_f \sum_{u \in V} \log Pr(N_S(u) | f(u)).$$

s라는 Sampling Strategy 방법으로  
sampling된 노드 U의 유사 노드들

u 노드의 feature 표현  
(임베딩 벡터)

가정

1. Conditional independence  
: 한 노드가 주어졌을 때, 다른 노드들이 관찰 될 확률은 독립적
2. Symmetry in feature space  
: 노드 간의 관계는 임베딩 공간에서도 유지됨.



## • Node2vec 목적 함수

가정에 의한 목적 함수 간단화

$$\max_f \sum_{u \in V} \left[ -\log Z_u + \sum_{n_i \in N_S(u)} f(n_i) \cdot f(u) \right]$$

↓

$$Z_u = \sum_{v \in V} \exp(f(u) \cdot f(v))$$

최적화 진행

- Negative sampling
  - 노드 수가 많을 경우  $Z_u$  계산량이 크므로, k개의 노드만 뽑아 정규화 진행.
- SGA(stochastic gradient ascent)
  - 확률적 경사 증가법으로 최적의 feature 계산



## • Sampling Strategy – 2<sup>nd</sup> order Random walk

- 2<sup>nd</sup> order Random walk : p, q 파라미터에 의한 임의보행

- 전이 확률     현재 임의 보행 과정:  $t \rightarrow v \rightarrow ?$

$$P(c_i = x \mid c_{i-1} = v) = \begin{cases} \frac{\pi_{vx}}{Z} & \text{if } (v, x) \in E \\ 0 & \text{otherwise} \end{cases}$$

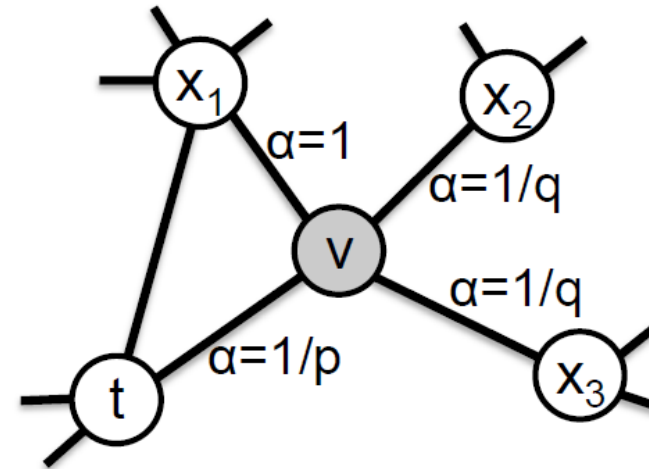
$$\pi_{vx} = \alpha_{pq}(t, x) \cdot \underline{w_{vx}}$$

고정 엣지 가중치  
(static edge weights)

탐색 편향  
(Search Bias)

$$\alpha_{pq}(t, x) = \begin{cases} \frac{1}{p} & \text{if } d_{tx} = 0 \\ 1 & \text{if } d_{tx} = 1 \\ \frac{1}{q} & \text{if } d_{tx} = 2 \end{cases}$$

- 탐색 편향(Search bias)



- return parameter, p
- In-out parameter, q





## • Sampling Strategy – q parameters

- In-out parameter, q
  - q 값이 크면 ( $>1$ )
 

이전 노드로부터 가까워서 임의 보행  $\rightarrow$  BFS

    - **BFS : 너비 우선 탐색**
      - 미시적 관점에서 임의 보행
      - Structural equivalence 반영된 feature learning 결과를 얻음
  - q 값이 작으면 ( $<1$ )
 

이전 노드로부터 멀리서 임의 보행  $\rightarrow$  DFS

    - **DFS : 깊이 우선 탐색**
      - 거시적 관점에서 임의 보행
      - Homophily 반영된 feature learning 결과를 얻음

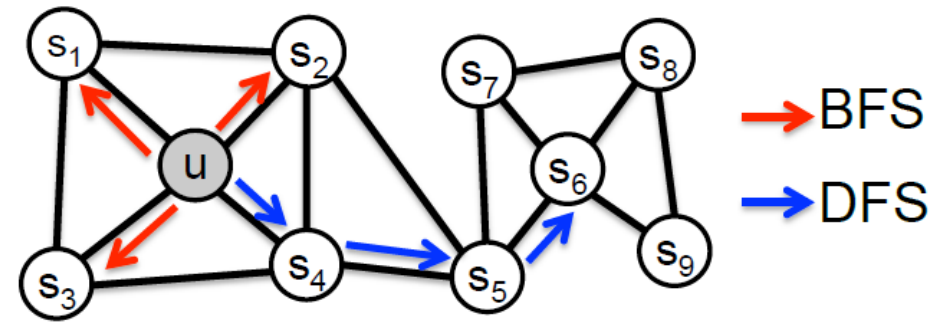


Figure 1: BFS and DFS search strategies from node  $u$  ( $k = 3$ ).

< u 노드로 부터의 BFS, DFS sampling strategy로 임의 보행 >



- **node2vec**

## 임의 보행의 장점 (Benefits of random walks)

### 1. 메모리 효율성

- 모든 노드의 인접한 노드들을 저장하는 공간 복잡도  $O(|E|)$
- 2<sup>nd</sup> order random walk 시, 각 노드의 이웃 노드간 상호 연결성을 저장하는 공간 복잡도  $O(a^2|V|)$   
 $a$ = 그래프 평균 연결성

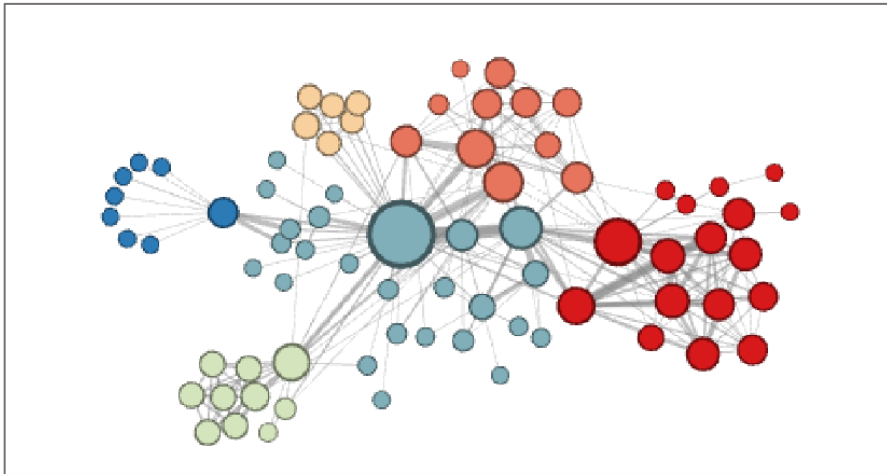
### 2. 시간 효율성

- 임의 보행 결과를 재 사용함으로써 샘플링 속도 증가

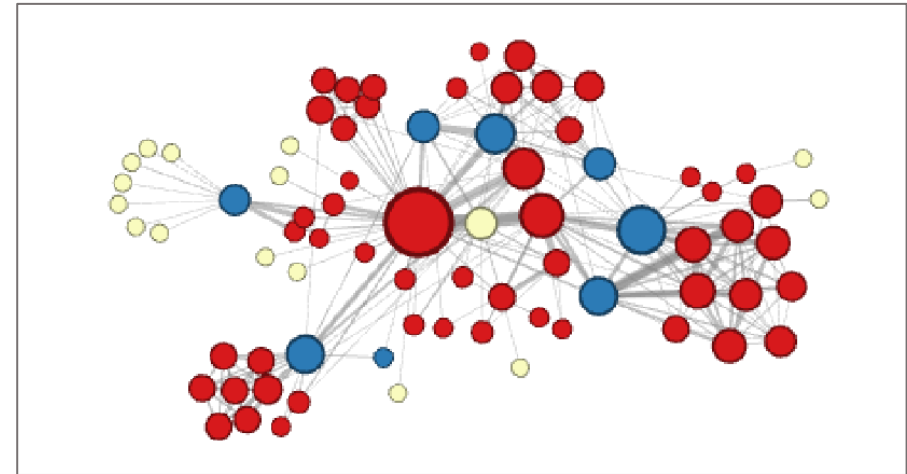


## • Case Study: Les Misérables network

- Graph : 레미제라블 소설 속 등장 인물과 인물 간의 관계
- Node : 등장 인물
- Edge : 함께 등장하는 인물
- **결과** : 비슷한 feature 끼리 같은 색상으로 표현



- 소설 속 상호 작용이 많은(같이 등장하는) 인물들끼리 분류 됨
- Homophily 가중 결과
- $p = 1, q = 0.5$



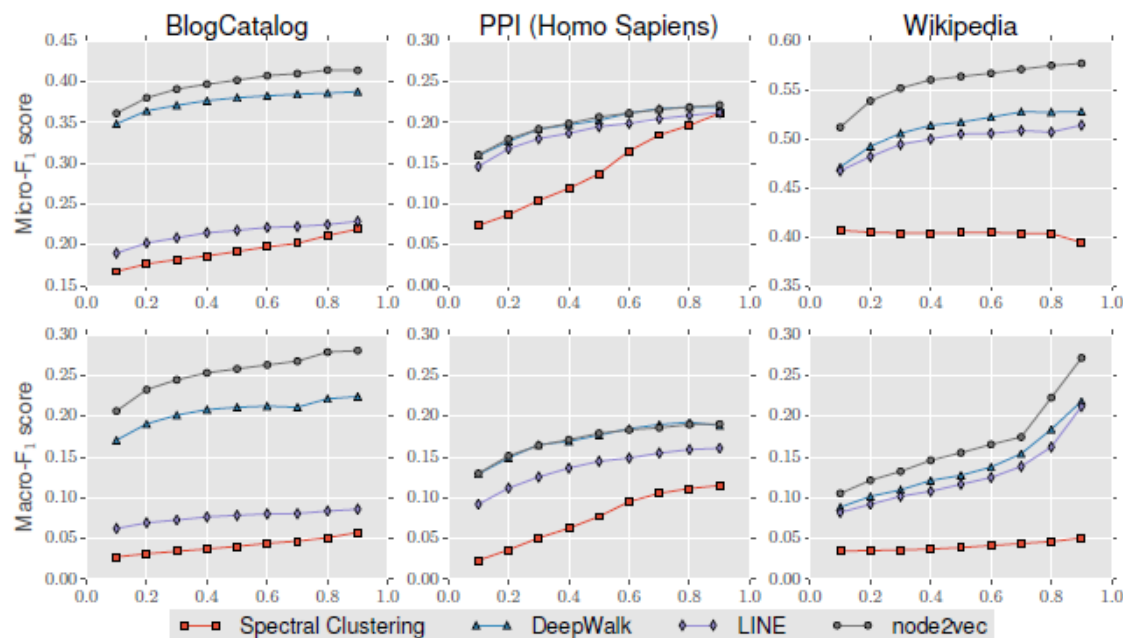
- 소설 속 비슷한 역할(메인 캐릭터, 서브 캐릭터, 씬 사이의 연결 캐릭터)끼리 분류 됨
- Structural equivalence 가중 결과
- $p = 1, q = 2$

- **의미** : Node2Vec 방법으로 Homophily, Structural equivalence 모두 표현 가능함 입증



## • Multi-label classification / Link prediction

- 비교 feature representation learning 학습 방법 : Spectral Clustering, DeepWalk, LINE, node2vec



< Multi-label classification 결과 >

-> 전반적으로 node2vec 이 최상의 성과를 보임

< Link prediction 결과 >

-> Hadamard 방법(edge feature 표현) 안정적(노란색 구간)

-> 그 중에서 node2vec이 최상의 성과를 보임

Op	Algorithm	Dataset		
		Facebook	PPI	arXiv
(a)	Common Neighbors	0.8100	0.7142	0.8153
	Jaccard's Coefficient	0.8880	0.7018	0.8067
	Adamic-Adar	0.8289	0.7126	0.8315
	Pref. Attachment	0.7137	0.6670	0.6996
	Spectral Clustering	0.5960	0.6588	0.5812
(b)	DeepWalk	0.7238	0.6923	0.7066
	LINE	0.7029	0.6330	0.6516
	node2vec	0.7266	0.7543	0.7221
	Spectral Clustering	0.6192	0.4920	0.5740
	DeepWalk	0.9680	0.7441	0.9340
(c)	LINE	0.9490	0.7249	0.8902
	node2vec	0.9680	0.7719	0.9366
	Spectral Clustering	0.7200	0.6356	0.7099
	DeepWalk	0.9574	0.6026	0.8282
	LINE	0.9483	0.7024	0.8809
(d)	node2vec	0.9602	0.6292	0.8468
	Spectral Clustering	0.7107	0.6026	0.6765
	DeepWalk	0.9584	0.6118	0.8305
	LINE	0.9460	0.7106	0.8862
	node2vec	0.9606	0.6236	0.8477



- **DISCUSSION AND CONCLUSION**

- Node2Vec
  - 2개의 파라미터  $p, q$  로 임의 보행 방법을 유연하게 조절할 수 있음(Homophily, Structural equivalence )
  - 노드 수 증가에 따른 feature representation learning 학습 속도도 선형적으로 증가
  - 노이즈가 있는 그래프 데이터에도 robust한 성능
  - Multi-label classification 과 Link prediction 에서 우수한 성능을 보임



## • References

- 논문
  - Aditya Grover, Jure Leskovec (2016) node2vec: Scalable Feature Learning for Networks. *KDD '16: Proceedings of the 22nd ACM SIGKDD*
- 블로그
  - <https://2hyes.tistory.com/145>
- 강의 자료
  - CS224W: Machine Learning with Graphs Jure Leskovec, Stanford University
  - 부스트 캠프 AI Tech Week5(그래프) 교육 자료 신기정 교수님, 네이버 커넥트 재단



# Q&A

