
How Powerful Are Graph Neural Networks?

School of Industrial and Management Engineering, Korea University

Lee Kyung Yoo

Contents

- ❖ Research Purpose
- ❖ Graph Isomorphism Network (GIN)
- ❖ Less powerful but still interesting GNNs
- ❖ Experiments
- ❖ Conclusion

Research Purpose

❖ How Powerful Are Graph Neural Networks? (ICLR 2019)

- MIT, Stanford University에서 연구하였으며 2022년 2월 5일 기준으로 2261회 인용

Published as a conference paper at ICLR 2019

HOW POWERFUL ARE GRAPH NEURAL NETWORKS?

Keyulu Xu ^{*†}
MIT
keyulu@mit.edu

Weihua Hu ^{*‡}
Stanford University
weihuahu@stanford.edu

Jure Leskovec
Stanford University
jure@cs.stanford.edu

Stefanie Jegelka
MIT
stefje@mit.edu

ABSTRACT

Graph Neural Networks (GNNs) are an effective framework for representation learning of graphs. GNNs follow a neighborhood aggregation scheme, where the representation vector of a node is computed by recursively aggregating and transforming representation vectors of its neighboring nodes. Many GNN variants have been proposed and have achieved state-of-the-art results on both node and graph classification tasks. However, despite GNNs revolutionizing graph representation learning, there is limited understanding of their representational properties and limitations. Here, we present a theoretical framework for analyzing the expressive power of GNNs to capture different graph structures. Our results characterize the discriminative power of popular GNN variants, such as Graph Convolutional Networks and GraphSAGE, and show that they cannot learn to distinguish certain simple graph structures. We then develop a simple architecture that is provably the most expressive among the class of GNNs and is as powerful as the Weisfeiler-Lehman graph isomorphism test. We empirically validate our theoretical findings on a number of graph classification benchmarks, and demonstrate that our model achieves state-of-the-art performance.

Research Purpose

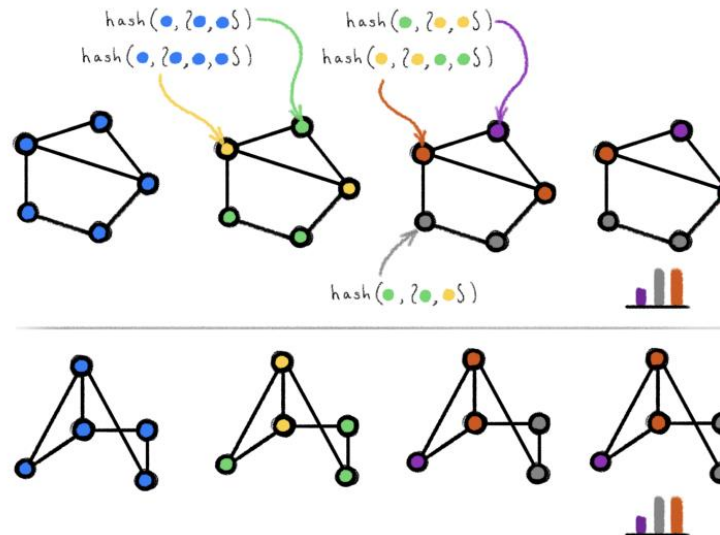
❖ How Powerful Are Graph Neural Networks? (ICLR 2019)

- **Graph Neural Networks (GNN) 의 representational power에 대한 이해 및 한계 파악의 부족**
 - ✓ GNN은 그래프의 representation learning을 위한 효과적인 프레임워크로서 node / graph classification 등의 분야에서 우수한 성능을 보임
 - ✓ 이에 따라 다양한 neighborhood aggregation 및 graph-level pooling 방식을 사용하는 GNN이 다수 제안됨
 - ✓ 그러나, 이러한 모델들의 설계는 주로 경험적인 직관 혹은 실험을 통한 시행 착오로 이루어짐
 - ✓ 원인으로서는 GNN의 속성 및 한계에 대한 이론적인 이해와 그 중에서도 representational capacity에 대한 제대로된 분석이 부족한 탓으로 볼 수 있음
- **본 연구는 GNN의 representational power에 대해 분석하는 이론적인 프레임워크를 제시**
 - ✓ 본 연구는 Weisfeiler-Lehman (WL) Test를 기반으로 GNN의 representational capacity를 분석함
 - ✓ 추가적으로 기존 GNN 모델 대비 더 큰 representational power를 가지는 새로운 알고리즘을 제안함

Research Purpose

❖ Weisfeiler-Lehman (WL) Test

- 본 연구는 기본적으로 GNN의 표현력을 WL graph isomorphism test 를 통해 설명
- WL graph isomorphism test는 효과적인 그래프 동형 테스트 기법으로 평가받고 있음
 - ✓ 기본적으로 graph isomorphism 문제는 두 개의 그래프가 위상적으로 동일한지 판단하는 문제
 - ✓ 이를 통해 isomorphic하다고 판단된 두 그래프는 그래프 요소들 사이의 일대일 대응이 성립하여 동일한 특성을 지님

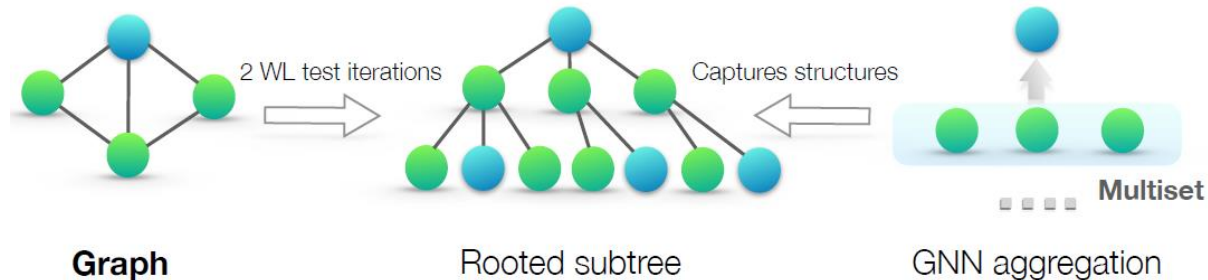


< Example of Isomorphic graphs >

Research Purpose

❖ Weisfeiler-Lehman (WL) Test

- GNN과 비슷하게, WL test 또한 node가 주어졌을 때 해당 node와 이웃 feature vectors를 통합하여 node의 feature vector를 반복적으로 업데이트 하는 방식임
 - ✓ 일정 수준 반복 이후 두 그래프 내 node들의 레이블이 달라지면 두 그래프는 non-isomorphic하다고 판단
 - ✓ WL test는 대다수 그래프의 isomorphism을 효과적으로 구분해낼 수 있는데, 이는 neighborhood aggregation 이후 node의 feature vector를 업데이트 하는 과정이 injective(= 그래프의 두 node가 서로 다른 neighborhood를 가지고 있다면 서로 다른 레이블을 가지도록) 하기 때문
- 결과적으로 GNN의 Aggregation scheme 역시 injective일 때, WL test에 준하는 discriminative / representational power를 가지는 maximally powerful GNN이 될 수 있음



< Rooted subtree structure derived from WL test/GNN >

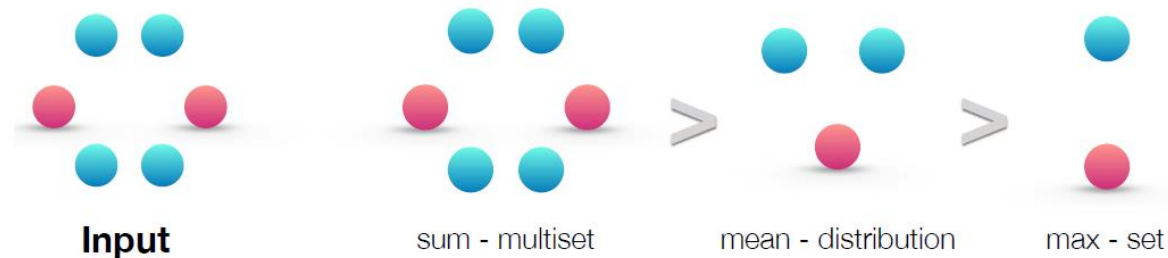
Research Purpose

❖ 기존 GNN의 Neighborhood Aggregation Scheme

- Neighborhood aggregation을 통해 이웃들의 representation을 통합하여 node의 feature를 업데이트
- 이에 따라 Aggregate 및 Combine 함수를 각각의 알고리즘에서 어떻게 정의하는지에 따라 전체 네트워크의 표현력은 상이함

$$a_v^{(k)} = \text{MAX} \left(\left\{ \text{ReLU} \left(W \cdot h_u^{(k-1)} \right), \forall u \in \mathcal{N}(v) \right\} \right) : \text{GraphSAGE (Aggregate)}$$

$$h_v^{(k)} = \text{ReLU} \left(W \cdot \text{MEAN} \left\{ h_u^{(k-1)}, \forall u \in \mathcal{N}(v) \cup \{v\} \right\} \right) : \text{GCN (Aggregate / Combine)}$$



< Ranking by expressive power of different aggregation strategy >

Graph Isomorphism Network (GIN)

❖ Graph Isomorphism Network (GIN)

- Neighbor aggregation을 위한 injective function을 모델링하기 위해 새로운 알고리즘 제안
 - ✓ 충분히 큰 hidden dimensionality와 적절한 비선형 함수를 갖는 MLP는 일정 수준의 정확도로 어떠한 연속형 함수도 근사할 수 있다는 Universal Approximation Theorem 적용
 - ✓ 최초의 input feature가 one-hot encoding이면 그들의 합 또한 injective할 것이기 때문에 합 연산 이전에 MLP가 필요하지 않지만, one-hot encoding이 아니거나 연속형 변수가 존재한다면 MLP가 필요할 것
- GIN은 최종적으로 node representation을 아래와 같이 업데이트함
 - ✓ ϵ 의 경우 학습 가능한 파라미터 혹은 고정된 값으로 둘 수 있음


$$h_v^{(k)} = \text{MLP}^{(k)} \left(\left(1 + \epsilon^{(k)} \right) \cdot h_v^{(k-1)} + \sum_{u \in \mathcal{N}(v)} h_u^{(k-1)} \right)$$

< GIN Aggregation function >

Graph Isomorphism Network (GIN)

❖ Graph Isomorphism Network (GIN)

- Node classification이 아닌, graph classification에는 추가로 graph-level readout function이 필요
 - ✓ Readout function은 node의 feature vector들에 대한 함수
 - ✓ 이 때 node의 feature vector, 즉 representation은 레이어를 거칠수록 local에서 global하게 변함
 - ✓ 따라서 레이어 수가 너무 많을 경우 global한 특성만, layer의 수가 너무 적을 경우 local한 특성만 남게 됨
 - ✓ 결과적으로 readout function을 통해 그래프를 구분하기 위해서는 적당한 수의 layer를 거쳐야 함
- 이러한 특성을 반영하기 위해, GIN은 각 레이어의 graph representation(READOUT(·)의 output)을 concatenation으로 모두 합쳐 줌
- 이에 따라 각 레이어마다 나타나는 그래프의 구조적 정보를 모두 포함할 수 있게 됨

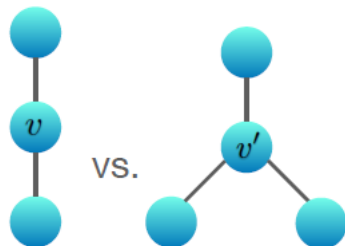

$$h_G = \text{READOUT}(\{h_v^{(K)} \mid v \in G\})$$
$$h_G = \text{CONCAT}\left(\text{READOUT}\left(\{h_v^{(k)} \mid v \in G\}\right) \mid k = 0, 1, \dots, K\right).$$

< GIN Readout function >

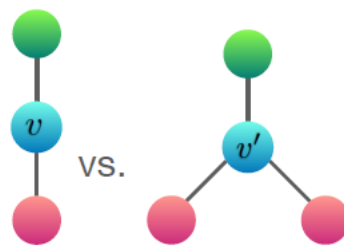
Less powerful but still interesting GNNs

❖ GNN 기반의 모델들이 포착할 수 있는 Graph 구조들의 한계

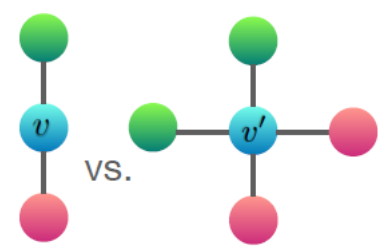
- GIN의 두 가지 특징(MLP / feature vector summation)에 대한 ablation study를 통해 입증
- 위 두 가지 특징에 대한 변화를 줄 경우, 모델의 성능이 하락함을 확인
 - ✓ MLP 대신 적용되는 1-layer perceptron은 이를 사용한 GNN이 구분하지 못하는 non-isomorphic 그래프들이 존재함에 따라 성능이 하락할 수 있음
 - ✓ 또한 bias를 고려하지 않는다면, linear mapping과 유사하게 작동하기 때문에 GNN layer들이 이웃 Feature들을 단순히 합산하는 수준으로 고려될 수 있음
 - ✓ Summation 대신 적용되는 mean-pooling 혹은 max-pooling은 multiset에 대해 injective 하지 않은 경우가 존재함에 따라 성능이 하락할 수 있음



Mean/Max 전부 실패



Max에서 실패



Mean/Max 전부 실패

< Examples of graph structures Mean/Max aggregators fail to distinguish >

Experiments

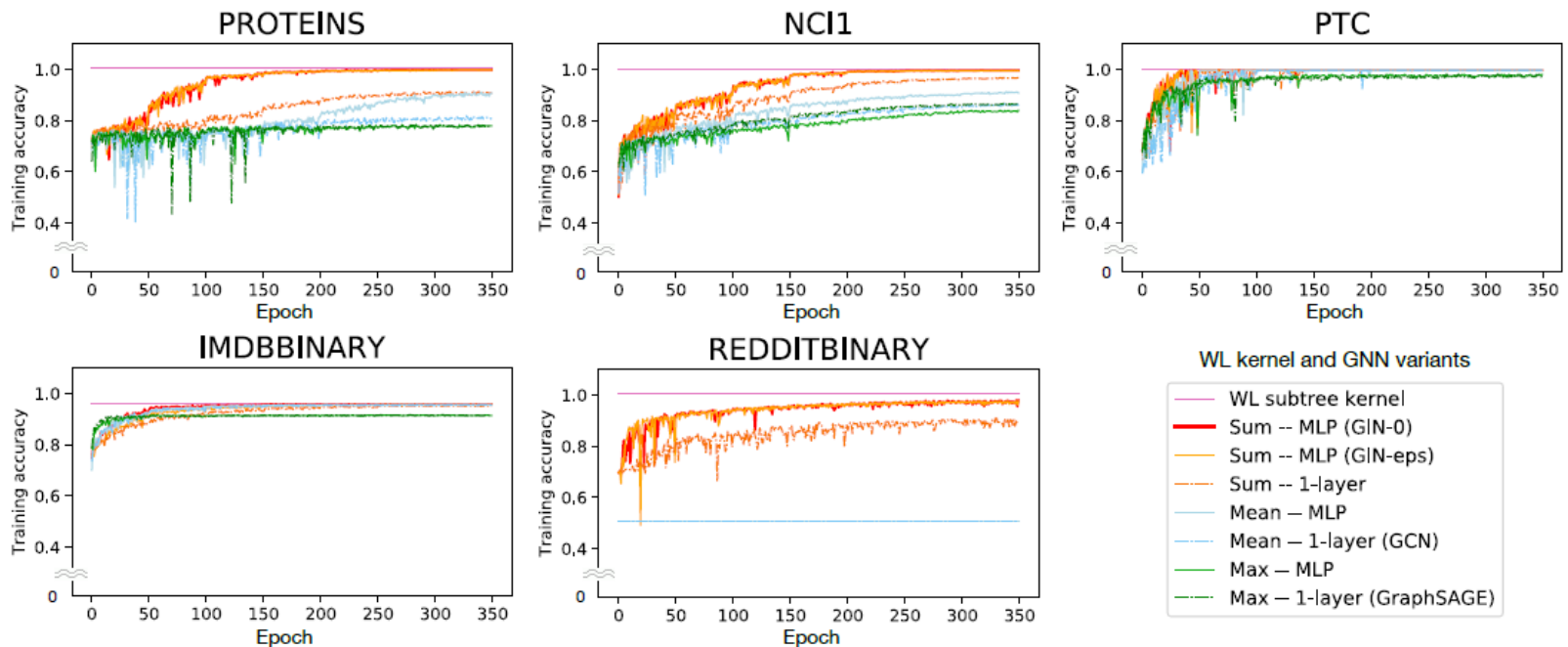
❖ Experiment Settings

- GIN 및 여러 GNN의 graph classification 성능 비교를 위한 실험 세팅
- 활용 Datasets
 - ✓ 4 Bioinformatics datasets (MUTAG, PTC, NCI1, PROTEINS)
 - ✓ 5 Social network datasets (COLLAB, IMDB-BINARY, IMDB-MULTI, REDDIT-BINARY, REDDIT-MULTI5K)
 - ✓ 실험의 목적은 모델이 input node feature에만 의존하지 않고 network 구조를 학습하도록 하는 것으로 Bioinformatics에서 node는 categorical input feature를 가지며, Social Network에서는 feature가 없음
 - ✓ Reddit 데이터셋에서는 모든 feature를 같게 만들어 사실상 의미 없도록 만들었으며, 다른 데이터셋에 대해서는 node degree를 one-hot 인코딩으로 추가하여 feature를 생성함
- Batch Normalization 적용
- Optimizer : Adam
- Learning rate : 0.01, 이후 50 epoch이 진행될 때마다 0.5씩 decay 적용
- Input layer를 포함하여 총 5개 layer 사용하였으며, 모든 MLP는 2개의 layer를 갖게 함

Experiments

❖ Results: Train set

- Representational power를 확인하기 위해 training accuracy 비교
- 모든 결과에서 GNN은 WL Test를 넘어서지는 못했는데, 이는 GNN이 WL Test에 비해 낮은 discriminative power를 갖고 있기 때문에 불가피한 결과임



*GIN-0은 ϵ 을 0으로 고정한 모델을 의미

Experiments

❖ Results: Test set

- Test set classification 정확도로 GNN의 expressive power를 가늠해보는 정도로 활용될 수 있음
- GIN이 모든 데이터셋에서 가장 우수한 결과를 보여준 것은 아님

Datasets	Datasets	IMDB-B	IMDB-M	RDT-B	RDT-M5K	COLLAB	MUTAG	PROTEINS	PTC	NCI1
	# graphs	1000	1500	2000	5000	5000	188	1113	344	4110
	# classes	2	3	2	5	3	2	2	2	2
	Avg # nodes	19.8	13.0	429.6	508.5	74.5	17.9	39.1	25.5	29.8
Baselines	WL subtree	73.8 \pm 3.9	50.9 \pm 3.8	81.0 \pm 3.1	52.5 \pm 2.1	78.9 \pm 1.9	90.4 \pm 5.7	75.0 \pm 3.1	59.9 \pm 4.3	86.0 \pm 1.8 *
	DCNN	49.1	33.5	–	–	52.1	67.0	61.3	56.6	62.6
	PATCHYSAN	71.0 \pm 2.2	45.2 \pm 2.8	86.3 \pm 1.6	49.1 \pm 0.7	72.6 \pm 2.2	92.6 \pm 4.2 *	75.9 \pm 2.8	60.0 \pm 4.8	78.6 \pm 1.9
	DGCNN	70.0	47.8	–	–	73.7	85.8	75.5	58.6	74.4
	AWL	74.5 \pm 5.9	51.5 \pm 3.6	87.9 \pm 2.5	54.7 \pm 2.9	73.9 \pm 1.9	87.9 \pm 9.8	–	–	–
GNN variants	SUM-MLP (GIN-0)	75.1 \pm 5.1	52.3 \pm 2.8	92.4 \pm 2.5	57.5 \pm 1.5	80.2 \pm 1.9	89.4 \pm 5.6	76.2 \pm 2.8	64.6 \pm 7.0	82.7 \pm 1.7
	SUM-MLP (GIN- ϵ)	74.3 \pm 5.1	52.1 \pm 3.6	92.2 \pm 2.3	57.0 \pm 1.7	80.1 \pm 1.9	89.0 \pm 6.0	75.9 \pm 3.8	63.7 \pm 8.2	82.7 \pm 1.6
	SUM-1-LAYER	74.1 \pm 5.0	52.2 \pm 2.4	90.0 \pm 2.7	55.1 \pm 1.6	80.6 \pm 1.9	90.0 \pm 8.8	76.2 \pm 2.6	63.1 \pm 5.7	82.0 \pm 1.5
	MEAN-MLP	73.7 \pm 3.7	52.3 \pm 3.1	50.0 \pm 0.0	20.0 \pm 0.0	79.2 \pm 2.3	83.5 \pm 6.3	75.5 \pm 3.4	66.6 \pm 6.9	80.9 \pm 1.8
	MEAN-1-LAYER (GCN)	74.0 \pm 3.4	51.9 \pm 3.8	50.0 \pm 0.0	20.0 \pm 0.0	79.0 \pm 1.8	85.6 \pm 5.8	76.0 \pm 3.2	64.2 \pm 4.3	80.2 \pm 2.0
	MAX-MLP	73.2 \pm 5.8	51.1 \pm 3.6	–	–	–	84.0 \pm 6.1	76.0 \pm 3.2	64.6 \pm 10.2	77.8 \pm 1.3
	MAX-1-LAYER (GraphSAGE)	72.3 \pm 5.3	50.9 \pm 2.2	–	–	–	85.1 \pm 7.6	75.9 \pm 3.2	63.9 \pm 7.7	77.7 \pm 1.5

Conclusion

❖ Conclusion

- GNN은 그래프 구조를 판별하는데 있어 WL Test 만큼이나 효과적임을 보이며, Neighbor Aggregation이나 Graph Readout Function이 이러한 효과를 증대시킬 조건을 제시
- GCN이나 GraphSAGE는 구분할 수 없는 Graph 구조에 대해 언급하고, GNN 기반의 모델들이 포착할 수 있는 Graph 구조들의 한계에 대해 정리
- GIN이라는 간단한 신경망 구조를 고안하여, 해당 알고리즘이 WL Test 만큼이나 discriminative / representational power를 갖고 있음을 증명

❖ Reference

- Xu, Keyulu, et al. "How powerful are graph neural networks?." arXiv preprint arXiv:1810.00826 (2018).
- Hamilton, William L., Rex Ying, and Jure Leskovec. "Inductive representation learning on large graphs." Proceedings of the 31st International Conference on Neural Information Processing Systems. 2017.
- Kipf, Thomas N., and Max Welling. "Semi-supervised classification with graph convolutional networks." arXiv preprint arXiv:1609.02907 (2016).
- https://greeksharifa.github.io/machine_learning/2021/06/05/GIN/

Thank You