

학사학위 청구논문

지도교수 허준석

# GResNet의 개념을 이용한 분자 특성 예측 Program의 성능 개선

성균관대학교 자연과학대학

화학과

김영오

학사학위 청구논문

지도교수 허준석

# GResNet의 개념을 이용한 분자 특성 예측 Program의 성능 개선

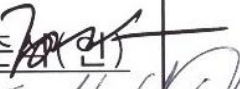
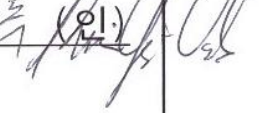
이 논문을 이학 학사학위 청구논문으로 제출합니다.

2021년 11 월 30 일

성균관대학교 자연과학대학  
화학과  
김영오

이 논문을 김영오의 이학 학사학위 논문으로 인정함.

2021년 11 월 30 일

지도교수 허준 (인)   
심사위원 김영오 (인) 

## 1. 연구배경

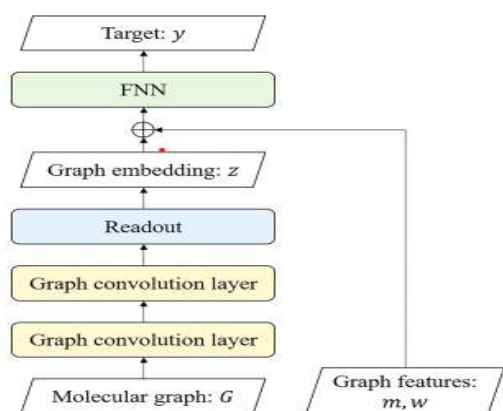
인공지능을 이용해서 분자의 특성인 solubility, lipophilicity, solvation energy, atomization energy, binding affinity 등을 예측하고자 하는 연구들이 활발히 나오고 있다. 하지만 보통 이런 연구에서 사용하는 데이터셋의 값들은 log scale인 경우가 많다. 통계적으로 참값과 가까운 값들이 잘 예측되더라도 outlier가 빈번히 발생하면 이 알고리즘을 Fermi Estimate 로써 사용했더라도 그 신뢰도가 떨어질 것이다. 본 연구에서는 그래프 합성곱 방법을 연구한 논문을 바탕으로, 인공지능 구조인 GResNet의 개념을 응용해 통계적으로 각 분자의 화학적 특성값들의 outlier를 통제할 수 있는 방법을 제안하고자 한다.

## 2. 도입

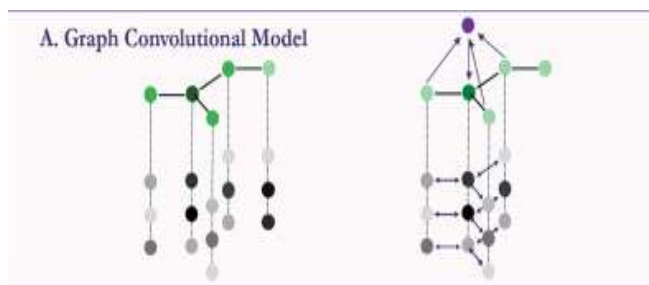
우선 본 연구에서 사용한 인공지능 모델은 분자식을 CCCC=O처럼 선형적인 data를 모델에 입력해 해당 dataset에 있는 분자 특성 수치를 예상하는 프로그램이다. 이 연구의 선행 연구로 EGCN[1]이 있다. 참고문헌[1]에서는 합성곱층 뒤에 분자의 전체적인 구조와 규모 특성을 잃어버려 예측의 성능이 떨어지는 점을 분자 속의 ring 개수와 분자량을 삽입해서 모델의 성능을 개선하였다. 본 연구에서는 다른 방식으로 합성곱층에서 전에 있던 정보를 유지하면서 가공할 수 있도록 GResNet의 개념을 사용하여 선행 연구와 같이 분자의 전체적인 특성을 잃어버리지 않으려는 의도를 유지하였으며, 동시에 각 모델들을 나뉘어 가지처럼 분화시키고 합성곱층들의 끝에서 합쳐서 서로 다르게 가공된 정보로부터 예측값을 내도록 하였다. 이를 위해서 대표적으로 분자의 수용도를 나타내는 데이터셋인 ESOL 및 다른 데이터셋들을 사용하여 새로 구축한 모델의 성능개선을 검증하였고, lipophilicity 데이터셋을 사용해 어떤 모델이 선택되는 tradeoff 과정도 보여주었다.

## 2. 선행연구

- EGCN(Extended Graph Convolution Network)



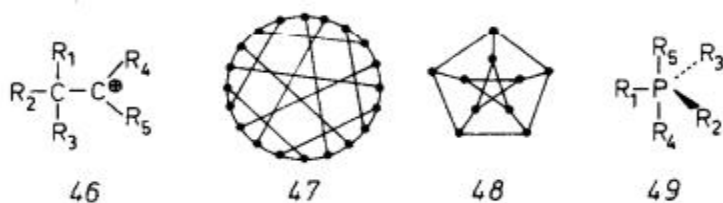
[그림1] Architecture of extended GCN (EGCN).  
m: graph feature vector related to the scale;  
w: graph feature vector related to the global structure[1]



[그림2] Graph convolutions. The graph convolutions featurization support most graph-based models. It computes an initial feature vector and a neighbor list for each atom. The feature vector summarizes the atom's local chemical environment[2]

본 연구에 앞서 선행연구를 살펴보면, 그림[2]와 같은 그래프 합성곱층을 쌓아 GCN을 만들면 층별로 연산해야 하기에 time cost가 더 증가함과 동시에 참값과의 오차가 커지는 경향이 있다. 합성곱은 sparse한 data에서 주요 특징들을 추출하여 사용하는 것이지만 분자는 자체의 특성으로 인해 주요 특징들을 뽑아내는 장점이 있는 반면 분자의 특성이 입체적 특성과 전체적인 모양에 의해 차이가 나기 때문에 정보를 축약해서 추출한다는 단점이 존재하게 된다. 그래서 본 연구에서는 그림[1]처럼 정보를 삽입해주는 형식으로 이러한 단점을 보완하였다.

아래 그림[3]은 분자를 그래프라는 식으로 표현하고 이를 축약하는 것의 문제점을 보여준다. 46에서 20-vertex graph로 나타내면 47이고 이때 두 탄소를 구분할 수 없다. 실제로도 46에서 1,2-alkyl shift에 의해 탄소양이온의 위치가 자주 바뀐다. 사람은 양이온을 R의 넘버링으로 구분이 가능하지만 node 그래프 상에서 구분이 안 된다. 탄소끼리 구분할 수 없기 때문에 두 탄소를 contraction할 수 있다. 20 vertices / 2 carbons = 10 vertices / carbon에 의해서 10-vertex graph로 바꾸면 48그래프가 된다. 48그래프는 47의 minor가



그림[3] Reaction Graphs

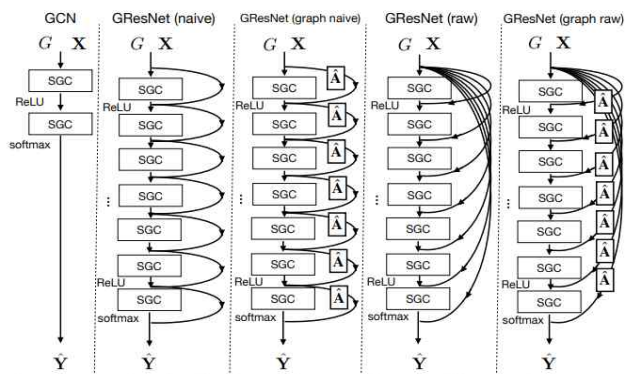
그림[3]은 분자를 그래프라는 방식으로 표현하고 이를 축약하는 것의 문제점을 보여준다.  
출처[3]

되는 것이다. 그리고 48그래프는 출처[3]에 의하면 49분자를 그래프로 표현한 것이라고 한다. 따라서 분자를 그래프로 나타내고 축약하면 전혀 다른 분자를 구분할 수 없게 된다.

부연설명1: 47그래프에서 20 vertex가 필요한 이유를 알아보자. 한 탄소에 대해 처음 R기를 둘 경우의 수 5와 그 다음 탄소를 둘 경우의 수는 4다. (R기의 종류가 5가지이기 때문이다.) 나머지 탄소의 위치는 자동으로 결정되며 R/S와 같은 입체화학을 고려치 않아  $5 * 4 = 20$ 이고 만일 탄소를 하나로 합치면  $\frac{1}{2}$ 의 경우의 수가 되기 때문인 것으로 파악된다. 탄소 사이의 edge를 제외하고 탄소는 최대 3개씩 결합을 가져서 3-regular로 표현한 것이다. 다만 탄소양이온 부분은 Alkyl shift에 의해서 구분되지 않아서 3-regular인 것이다.

부연설명2: 47에서 다음의 3가지 방법들을 사용해서 48이 될 수 있어서 48은 47의 minor다. 1) edge를 지운다. 2) edge를 contraction한다. 이것은 해당 edge를 없애면서 연결되어 있던 두 node를 하나로 합치는 것이다. 3) 홀로 있는 edge를 제거한다.

- GResNet(Graph ResNet)



그림[4] GResNet 구조들  
출처[4]

선행연구에서 사용한 인공지능 모델의 구조가 GCN을 단순히 쌓기만 한 것이었다면 본 연구에서는 위의 4가지 GResNet 구조 중 GResNet (naive)의 구조를 이용하고자 한다. GResNet (raw)의 경우는 data의 양이 지속적으로 신경망 층수를 거치면서 감소한 결과로 오직 target값 1개만 나와야 하기 때문에 비교군으로만 사용했다. 이때, GResNet (naive)는 점진적으로 data의 양이 감소한다. 반면에 GResNet (raw)는 점진적으로 감소하지 않고 층수 후반부에서 갑자기 줄어들어 1개의 값을 출력해야 한다. 따라서 GResNet (naive)에 집중했다. ResNet이란 구조는 여러 합성곱층을 쌓으면서 그 층 사이에 skip connection이란 방법을 사용해서 vanishing gradient 문제를 해결하려는 구조이다. 그리고 skip connection은 단계를 건너뛰어 현재의 합성곱층으로 이전의 정보를 입력하는 연결 방식이다. ResNet은 이것을 여러 번 사용했으며 그 대상이 graph이기 때문에 GResNet이다. 본 연구에서는 가공된 intermediate representations를 이용해서 기존 EGCN과 비교하거나 같이 사용되었기 때문에 GResNet (naive) 모델을 주로 사용하였다.

vanishing gradient 문제: 은닉층을 많이 거칠수록 전달되는 오차가 크게 줄어들어 학습이 되지 않는 현상이 발생하는데, 이를 기울기 소멸 문제라고 한다. 기울기가 거의 0으로 소멸되어 버리면 네트워크의 학습은 매우 느려지고, 학습이 다 이루어지지 않은 상태에서 멈출 것이다. 이를 지역 최솟값에 도달한다고 표현하기도 한다.[5]

- ESOL 및 다른 데이터셋들

dataset	target property	# of molecules	source
ESOL <sup>26</sup>	aqueous solubility	1128	exp.
FreeSolv <sup>27</sup>	solvation energy	642	both
lipophilicity <sup>28</sup>	water distribution coefficient	4200	exp.
PDBbind <sup>29</sup>	binding affinity	9880	exp.
QM7 <sup>30,31</sup>	atomization energy	6830	calc.

그림[5] Dataset 종류  
출처[1]

각 dataset은 input에 해당하는 SMILES와 target value에 해당하는 수치가 들어있다. 원본 EGCN 모델 코드와 그림[5]의 데이터셋은 supplement에 설명해 놓았다.

• 성능 측정 (MSE, RMSE, MAE, R2 score)

$$\bar{e}_\gamma = \left[ \frac{\sum_{i=1}^n w_i |e_i|^\gamma}{\sum_{i=1}^n w_i} \right]^{1/\gamma} \quad \text{RMSE} = \left[ n^{-1} \sum_{i=1}^n |e_i|^2 \right]^{1/2} \quad \text{MAE} = \left[ n^{-1} \sum_{i=1}^n |e_i| \right]$$

그림[6-1]

그림[6-2]

그림[6-3] 출처[6]

6-1은 Average model-estimation error로 6-2와 6-3의 일반형이라고 할 수 있다. e는 input x와 target y의 차이이며 본 연구에서 w는 모두 1로 본다.  $\gamma=1$ 이면 MAE가 되며  $\gamma=2$ 이면 6-2가 된다.

Variable	Case 1	Case 2	Case 3	Case 4	Case 5
$e_1$	2	1	1	0	0
$e_2$	2	1	1	0	0
$e_3$	2	3	1	1	0
$e_4$	2	3	5	7	8
$\sum  e_i $	8	8	8	8	8
MAE	2	2	2	2	2
$\sum  e_i ^2$	16	20	28	50	64
RMSE	2.0	2.2	2.6	3.5	4.0

그림[7] 출처[6]

그림과 같이 몇몇 경우에서 잘 맞추었다고 하더라도 오차 사이의 편차를 평가하는 데에는 MAE보다 MSE와 RMSE가 더 적절하다. 즉, MSE가 작을수록 outlier가 줄어든다.

MAE, RMSE, MSE는 0에 가까울수록 좋으며 MAE는 예측값이 참값과의 거리를 나타내며 accuracy를 알 수 있다. RMSE와 MSE는 precision을 알기 위해서 사용했으며 본 연구에서는 outlier를 감소시키기 위해서 위의 두 가지 방법을 통해 모델의 성능을 비교, 분석하였다.

MAE는 mean absolute error로 input x와 target y의 차의 절댓값을 합한 뒤에 batch 크기로 나눈 값이다. pytorch에선 L1Loss로 사용할 수 있다.

MSE는 mean squared error로 input x와 target y의 차의 제곱을 합한 뒤에 batch 크기로 나눈 값이다. pytorch에선 MSELoss로 사용할 수 있다. 그리고 RMSE는 MSE의 제곱근으로 그 경향성이 같다.

MAE가 작아지는 것이 전반적인 예측값이 참값에 근접했다는 것을 보여준다. RMSE도 MAE의 수치 증감 경향성을 어느 정도 같이 가지만 갖는 의미는 서로 다를 수 있다.  $MAE \leq RMSE \leq \sqrt{n} \cdot MAE$ 에서 알 수 있듯이 RMSE는 경계가 있고 n이 증가할수록 RMSE의 값이 비일관적으로 커질 수 있기 때문에 다른 정보를 주지 않는다면 RMSE만으로 중심

적인 경향(평균 오차)를 결정할 수 없다.[6] 따라서 본 연구는 MSE의 개선을 통해서

$$\frac{\sum(\hat{y}_i - \bar{y})^2}{\sum(y_i - \bar{y})^2} = 1 - \frac{\sum e_i^2}{\sum(y_i - \bar{y})^2}$$

그림[8] 출처[7]

결정계수 공식이며 e는 error다.

outlier를 줄이려고 시도하였으며 모델의 성능을 조정하고 타협하는 수단으로는 MAE를 사용하였다.

R2\_score는 결정계수라고도 하며 식으로는 “1-(MSE/분산)”이다. 회귀식(回歸式)의 적합도를 재는 척도이다. 회귀분석에서 종속변수 Y의 데이터  $y_i$ 에 대하여,  $y_i$ 의 총변동합에 대한 변동합의 비율을 나타낸다.[7] 본 연구에서는 MAE, MSE, RMSE 외에도 R2\_score를 추가하여 모델의 성능을 비교, 분석하였다.

### • SMILES

SMILES(Simplified Molecular-Input Line-Entry System)는 분자의 화학식을 문자열로 나타내는 데 널리 사용되는 방법이다. 예를 들어 ‘OCCc1c(C)[n+](cs1)Cc2cnc(C)nc2N은 비타민 B<sub>1</sub>으로도 알려진 중요한 영양소인 티아민을 표현한 것이다.출처[8]

본 연구에서 사용되는 dataset은 보통 input data로 분자의 SMILES가 있고 그에 짝지어진 target값으로 구성되어 있다.

## 3. 연구방법 및 결과

### • GResNet (naive)의 개념을 이용해서 여러 층 구조 만들기

먼저, 기존에 사용된 GNN(Graph neural networks) 모델은 층수가 많아짐에 따라서 모델이 훈련 데이터에 반응하지 않고 학습되지 않는 상태(suspended animation problem)가 발생한다.[4] 앞서 설명한 vanishing gradient 문제이다. 참고문헌[4]에서는 bias term의 여부에 따라서 이 문제가 발생하는 경계를 다음과 같이 구분하였다. “depth  $\geq 5$  for GCN with the bias term disabled or depth  $\geq 8$  for GCN with the bias term enabled on the Cora dataset.” 본 연구에서는 위와 같은 문제발생의 가능성을 검증함과 동시에 서로 다른 경우

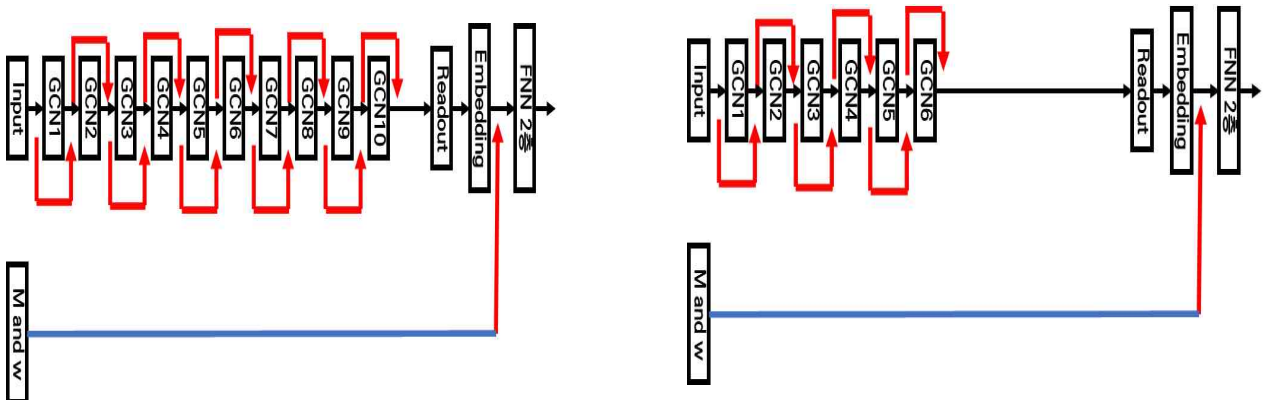


의 결과를 비교하기 위해서, GResNet (naive) 구조를 바탕으로 10층, 6층, 3층 이하의 모델 구조들을 만들었다. 각각 8층 이상의 합성곱 층을 쌓은 경우, 층수가 8층과 5층 사이의 경우, 5층 미만의 경우로 나누었으며, GResNet (raw)의 경우는 6층과 3층 이하만 만들었다.

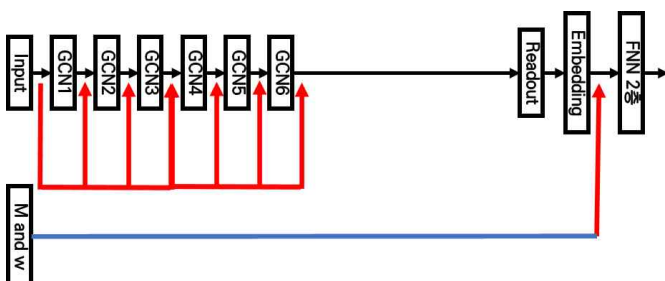
- 본 연구에서 구현하여 사용된 새로운 모델들에 쓰인 highway 방식

이 연구에서 쓰인 인공지능 모델은, EGCN의 합성곱층들과 GResNet (naive)의 합성곱 층이 input에서부터 갈라져 나와서 서로 독립된 층을 통과하다가 FNN(Fully connected neural network)에 들어가기 전에 다시 하나로 합쳐지는 방식이다. 또한, 각 루트는 서로 다른 정도로 가공된 정보를 갖고서 FNN에 들어가게 된다. 즉, GResNet의 highway로써 합성곱층의 맨처음과 맨마지막을 이어준 것이라고 볼 수 있다. 하지만 원래의 input을 그대로 가기 보단 합성곱을 거쳐 갈 수 있다는 차이가 있다.

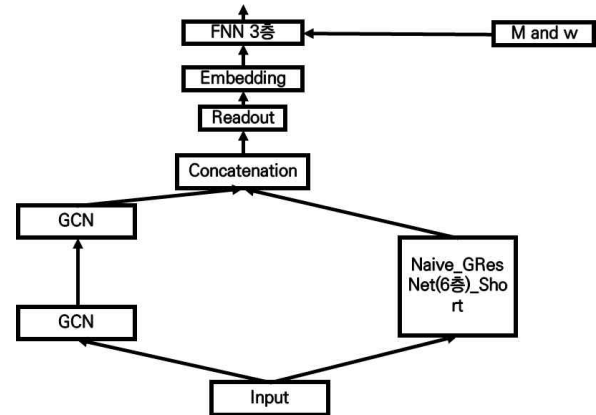
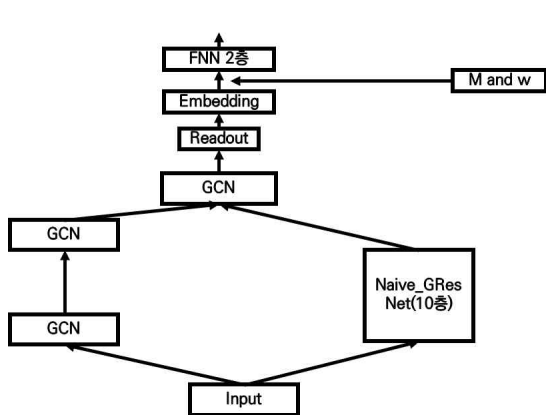
- 학습모델들



그림[9-1] GResNet\_EGCN(naive)      그림[9-2] GResNet\_EGCN\_Short  
이후 'GResNet\_숫자'인 모델은 GCN의 층수만 달리한 모델들이다.



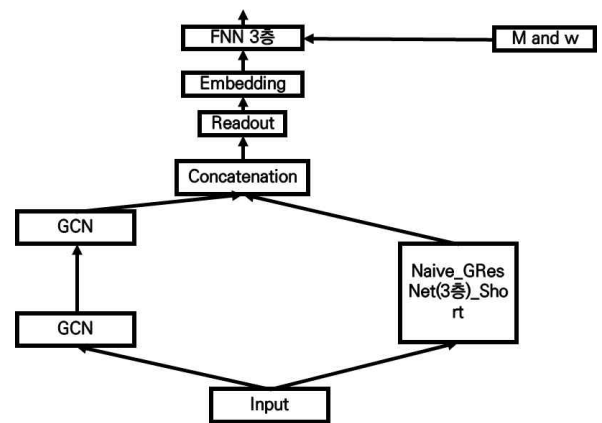
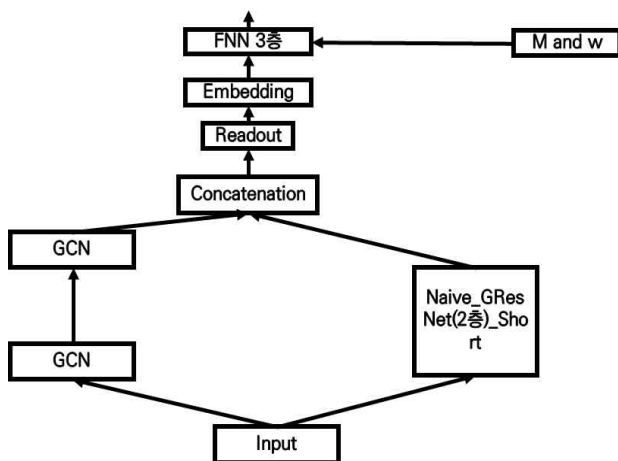
그림[10] raw\_숫자 model이다. 여기에서 숫자란 GCN의 층수를 의미한다.



그림[11-1] GED(GResNet\_EGCN\_Depthwise)

그림[11-2] Short\_GED

11-1의 readout 전 마지막 GCN과 11-2의 FNN이 한층 더 있는 것은 factor 수가 급격하게 변하는 것을 방지하기 위해서 만들어 놓은 buffer layer이다.



그림[12-1]Very\_Short\_GED

그림[12-2] Very\_Short\_GED\_3

이 두 모델들은 층수에 따른 성능을 비교할 때 tradeoff에 사용된다.

EGCN은 선행연구[1]에서 가져왔다. GED Series에서 highway 부분에서 GCN이 2층인 것은 EGCN에서 GCN을 2층으로 두었기 때문에 고정된 것으로 취급했다. 그림[9-1]과 같이 새로 구축한 GResNet\_EGCN(naive) 모델은 합성곱층을 10층으로 두었다. GResNet\_EGCN\_Short는 그림[9-2]와 같이 합성곱층을 6층으로 만들었다. 이 둘의 공통점은 skip connection의 도약이 1개의 층간의 도약인 점이다. 그림[10]과 같이 새로운 raw\_숫자 model들은 naive 모델들과 달리 skip connection의 도약이 처음에서 갈라져 나와서 모든 층간에 삽입된다는 점이다. raw 이름 뒤의 숫자는 합성곱 층의 개수다. GED Series는 GResNet (naive)의 층수를 다르게 하면서 highway로 합성곱층 2개를 설정한 모델이다.

GED Series에서 GResNet (naive)의 합성곱의 층수는 다음과 같이 모델을 구축하였다. GED는 10층이고, Short\_GED는 6층이고, Very\_Short\_GED\_3은 3층이며 Very\_Short\_GED는 2층이다. GED에서 concatenation 다음에 GCN과 다른 GED Series에서 FNN이 하나 더 있는 것은 층간 data의 양이 급변하지 않게 하기 위해 만들어 놓은 buffer층이다.

- 각 모델을 독립 시행으로 10회씩 실행해 loss값을 통계적으로 얻기

각 모델들의 loss값은 MAE, MSE, RMSE 등이 있다. 모든 모델들은 실행될 때마다 loss값이 다소 차이가 있었다. 몇몇 dataset의 target값이 log값은 점을 감안했을 때, 이 차이는 모델의 성능을 측정하기에 바람직하지 않기에 안정된 loss값을 얻기 위해서 각 모델을 독립적으로 10번씩 실행하여 loss값을 얻었다. 그리고서 그 값들을 산술평균하고 최대, 최소, 표준편차 등도 구했다. 또한 실행하는 환경은 연구실 PC와 서버 리눅스이며, 뒤에서 제시할 결과의 한 table당 실행환경과 batch size는 같다. 그러나 다른 table 간에는 batch\_size가 같지 않을 수 있다. 참고로 epoch는 300으로 고정하였다.

### 3. 연구 결과 및 고찰

본 연구는 인공지능 모델을 이용한 특성(solubility, lipophilicity, solvation energy, atomization energy, binding affinity) 예측의 성능 개선을 목표로 하여 GResNet의 개념을 바탕으로 합성곱층 수를 조절하여 모델을 구축하여 성능을 비교, 분석하였다. 성능개선의 기준은 참고문헌[1]의 EGCN모델로 삼았으며 본 연구에서 새로 생성한 모델들 중 5가지 특성 예측 부분에서 Short\_GED와 GED가 MSE 부분에서 모두 개선되었다. 그러나 lipophilicity는 MAE 부분에서 다소 악화되었다.

- Loss값 table과 개선을 graph

본 연구의 개선율은 참고문헌[1]의 공식을 사용하며  $(EGCN\_loss - \text{다른 모델의 loss}) / EGCN\_loss * 100$  %로 나타내었다. 여기에서는 table로 정리하였으며 나머지 차트는 참고문헌 뒤에 부록으로 첨부하였다. 또한 세부 사항은 MAE만 표시하였으며 MSE의 세부 사항(표준편차, 최대최소, 구간 길이)은 엑셀 파일에 정리하여 첨부하였다. 표본분산은 각 dataset의 target값의 표본분산이다. 본래 수치들의 소숫점 아래 숫자들이 길어서 소숫점 3째 자리에서 반올림하였다. 그림[16]부터 그림[19]까지에서 개선율은 선행연구[1]의 EGCN의 성능을 0으로 두고 나타낸 상대적 개선율이다. 따라서 개선율이 양수가 나오면 개선된 것이고 음수가 나오면 악화된 것으로 판별하였다.

esol_MAE	avg_loss	std_loss	max	min	length	avg_time	improvement(%)
EGCN(기준)	0.74	0.03	0.79	0.69	0.11	624.13	0
GResNet_EGCN(naive)	0.77	0.02	0.8	0.72	0.08	2203.37	-3.53
GResNet_EGCN(naive)_Short	0.7	0.01	0.72	0.69	0.04	1437.54	4.82
GResNet_1	0.81	0.01	0.83	0.79	0.04	411.92	-9.59
GResNet_2	0.72	0.01	0.76	0.7	0.05	631.27	2.16
GResNet_3	0.73	0.02	0.75	0.71	0.05	843.64	1.33
raw_2	0.72	0.01	0.75	0.7	0.05	638.73	2.39
raw_3	0.74	0.02	0.77	0.72	0.06	843.4	-0.32
raw_6	0.71	0.01	0.72	0.7	0.02	1445.34	3.76
GED	0.71	0.02	0.75	0.68	0.07	2803.46	3.34
Short_GED	0.66	0.02	0.68	0.63	0.05	1831.39	10.86
Very_Short_GED	0.7	0.02	0.75	0.68	0.06	1022.76	4.8
Very_Short_GED_3	0.69	0.02	0.72	0.65	0.07	1228.14	7.03
mini-batch size = 32	랩실의 PC로 돌렸다.						

esol_MSE	avg_loss	improvement(%)	RMSE	R2_score
EGCN(기준)	0.89	0	0.94	0.8
GResNet_EGCN(naive)	0.73	17.85	0.85	0.83
GResNet_EGCN(naive)_Short	0.74	16.94	0.86	0.83
GResNet_1	1.17	-32.44	1.08	0.73
GResNet_2	1	-12.33	1	0.77
GResNet_3	0.79	10.55	0.89	0.82
raw_2	0.83	6.87	0.91	0.81
raw_3	0.98	-10.69	0.99	0.78
raw_6	0.81	8.11	0.9	0.82
GED	0.72	19.26	0.85	0.84
Short_GED	0.69	21.87	0.83	0.84
Very_Short_GED	0.76	14.24	0.87	0.83
Very_Short_GED_3	0.75	14.83	0.87	0.83
4.4 표본분산				

그림[13-1] ESOL에서의 MAE

그림[13-2] ESOL에서의 MSE

freesolv_MAE	avg_loss	std_loss	max	min	length	avg_time	improvement(%)
EGCN(기준)	1.87	0.01	1.89	1.86	0.03	49.08	0
GResNet_EGCN(naive)	1.69	0.02	1.72	1.67	0.06	167.55	9.66
GResNet_EGCN(naive)_Short	1.84	0.03	1.9	1.79	0.11	110.39	2
GResNet_1	2.23	0	2.24	2.22	0.01	35.9	-19.08
GResNet_2	1.8	0.02	1.83	1.77	0.06	46.37	3.97
GResNet_3	2.24	0.02	2.27	2.22	0.05	57.04	-19.77
raw_2	1.91	0.02	1.94	1.86	0.08	45.56	-2.08
raw_3	1.92	0.02	1.95	1.88	0.06	55.91	-2.72
raw_6	1.79	0.02	1.82	1.77	0.05	106.9	4.54
GED	1.74	0.04	1.82	1.69	0.14	315.79	7.09
Short_GED	1.66	0.01	1.69	1.64	0.05	138.6	11.18
Very_Short_GED	1.8	0.01	1.82	1.78	0.04	64.42	3.97
Very_Short_GED_3	1.73	0.03	1.8	1.67	0.13	73.81	7.89
batch_size = 256	서버 리눅스로 돌림						

freesolv_MSE	avg_loss	improvement(%)	RMSE	R2_score
EGCN(기준)	8.43	0	2.9	0.43
GResNet_EGCN(naive)	6.18	26.68	2.49	0.58
GResNet_EGCN(naive)_Short	7.4	12.26	2.72	0.5
GResNet_1	10.71	-27.01	3.27	0.28
GResNet_2	7.36	12.69	2.71	0.5
GResNet_3	7.5	11.02	2.74	0.49
raw_2	6.88	18.45	2.62	0.54
raw_3	8.2	2.75	2.86	0.45
raw_6	7.19	14.74	2.68	0.51
GED	7.11	15.75	2.67	0.52
Short_GED	7.06	16.24	2.66	0.52
Very_Short_GED	6.55	22.31	2.56	0.56
Very_Short_GED_3	7.35	12.89	2.71	0.5
표본분산	14.81			

그림[14-1] FreeSolv MAE

그림[14-2] FreeSolv MSE

lipo_MAE	avg_loss	std_loss	max	min	length	avg_time	improvement(%)
EGCN(기준)	0.86	0.01	0.87	0.85	0.02	2267.5	0
GResNet_EGCN(naive)	0.87	0.01	0.88	0.85	0.03	8014.11	-1.45
GResNet_EGCN(naive)_Short	0.88	0.01	0.89	0.88	0.02	5162.84	-3.23
GResNet_1	0.89	0	0.89	0.88	0.01	1475.4	-3.88
GResNet_2	0.84	0.01	0.86	0.83	0.03	2279.2	1.73
GResNet_3	0.85	0.01	0.86	0.84	0.02	3011.85	0.63
raw_2	0.86	0.01	0.87	0.85	0.02	2272.97	-0.95
raw_3	0.88	0.01	0.89	0.87	0.02	3016.49	-2.94
raw_6	0.88	0	0.89	0.87	0.02	5245.51	-2.93
GED	0.9	0	0.9	0.9	0.01	10224.71	-5.38
Short_GED	0.89	0	0.9	0.89	0.01	6690.81	-4.44
Very_Short_GED	0.85	0.01	0.87	0.82	0.05	3744.53	0.96
Very_Short_GED_3	0.87	0.01	0.89	0.86	0.03	4461.17	-2.1
batch_size = 32	랩실 PC						

lipo_MSE	avg_loss	improvement(%)	RMSE	R2_score
EGCN(기준)	1.19	0	1.09	0.18
GResNet_EGCN(naive)	1.05	11.73	1.03	0.27
GResNet_EGCN(naive)_Short	1.13	5.13	1.06	0.22
GResNet_1	1.23	-2.89	1.11	0.15
GResNet_2	1.12	5.75	1.06	0.22
GResNet_3	1.12	6.29	1.06	0.23
raw_2	1.07	10.54	1.03	0.26
raw_3	1.23	-3.08	1.11	0.15
raw_6	1.14	4.2	1.07	0.21
GED	1.07	10.03	1.04	0.26
Short_GED	1.09	8.95	1.04	0.25
Very_Short_GED	1.12	6.08	1.06	0.23
Very_Short_GED_3	1.08	9.06	1.04	0.25
표본분산	1.45			

그림[15-1] lipophilicity MAE

그림[15-2] lipophilicity MSE

qm7_MAE	avg_loss	std_loss	max	min	length	avg_time	improvement(%)	qm7_MSE	avg_loss	improvement(%)	RMSE	R2_score
EGCN(기준)	0.34	0	0.35	0.34	0.01	1604.67	0	EGCN(기준)	0.29	0	0.54	0.71
GResNet_EGCN(naive)	0.31	0	0.31	0.3	0.01	4089.58	10.52	GResNet_EGCN(naive)	0.28	5.83	0.53	0.72
GResNet_EGCN(naive)_Short	0.33	0	0.33	0.32	0.01	2793	4.73	GResNet_EGCN(naive)_Short	0.29	1.69	0.54	0.71
GResNet_1	0.24	0	0.25	0.24	0.01	1273.34	29.26	GResNet_1	0.24	17.69	0.49	0.76
GResNet_2	0.34	0	0.35	0.33	0.01	1598.97	0.78	GResNet_2	0.28	3.25	0.53	0.72
GResNet_3	0.32	0.01	0.33	0.31	0.02	1903.49	7.29	GResNet_3	0.28	6.28	0.52	0.73
raw_2	0.24	0	0.24	0.23	0.01	1600.29	30.99	raw_2	0.24	18.74	0.49	0.76
raw_3	0.23	0	0.24	0.23	0.01	1900.64	31.75	raw_3	0.24	17.87	0.49	0.76
raw_6	0.24	0	0.24	0.24	0.01	2792.9	29.87	raw_6	0.24	17.76	0.49	0.76
GED	0.33	0.01	0.35	0.32	0.03	4841.66	2.6	GED	0.28	4.25	0.53	0.72
Short_GED	0.31	0.01	0.32	0.3	0.02	3365.69	10.12	Short_GED	0.26	10.02	0.51	0.74
Very_Short_GED	0.33	0	0.33	0.32	0.01	2183.92	5.11	Very_Short_GED	0.28	4.83	0.53	0.72
Very_Short_GED_3	0.3	0	0.3	0.3	0.01	2484.03	12.86	Very_Short_GED_3	0.28	5.14	0.53	0.72
batch_size = 128	랩실 PC							표본분산	1			

그림[16-1] QM7 MAE

[그림16-2] QM7 MSE

pdbbind_MAE	avg_loss	std_loss	max	min	length	avg_time	improvement(%)	PDBbind_MSE	avg_loss	improvement(%)	RMSE	R2_score
EGCN(기준)	0.74	0.03	0.79	0.69	0.11	624.13	0	EGCN(기준)	2.16	0	1.47	0.27
GResNet_EGCN(naive)	0.77	0.02	0.8	0.72	0.08	2203.37	-3.53	GResNet_EGCN(naive)	2.17	-0.59	1.47	0.27
GResNet_EGCN(naive)_Short	0.7	0.01	0.72	0.69	0.04	1437.54	4.82	GResNet_EGCN(naive)_Short	2.19	-1.32	1.48	0.26
GResNet_1	0.81	0.01	0.83	0.79	0.04	411.92	-9.59	GResNet_1	2.18	-0.99	1.48	0.26
GResNet_2	0.72	0.01	0.76	0.7	0.05	631.27	2.15	GResNet_2	2.23	-3.39	1.49	0.25
GResNet_3	0.73	0.02	0.75	0.71	0.05	843.64	1.33	GResNet_3	2.14	0.96	1.46	0.28
raw_2	0.72	0.01	0.75	0.7	0.05	638.73	2.39	raw_2	2.17	-0.54	1.47	0.27
raw_3	0.74	0.02	0.77	0.72	0.06	843.4	-0.32	raw_3	2.19	-1.23	1.48	0.26
raw_6	0.71	0.01	0.72	0.7	0.02	1445.34	3.76	raw_6	2.27	-4.81	1.51	0.24
GED	0.71	0.02	0.75	0.68	0.07	2803.46	3.34	GED	2.05	5.05	1.43	0.31
Short_GED	0.66	0.02	0.68	0.63	0.05	1831.39	10.86	Short_GED	2.15	0.73	1.46	0.28
Very_Short_GED	0.7	0.02	0.75	0.68	0.06	1022.76	4.8	Very_Short_GED	2.18	-0.84	1.48	0.27
Very_Short_GED_3	0.69	0.02	0.72	0.65	0.07	1228.14	7.03	Very_Short_GED_3	2.15	0.47	1.47	0.27
batch_size=128	랩실PC							표본분산	2.96			

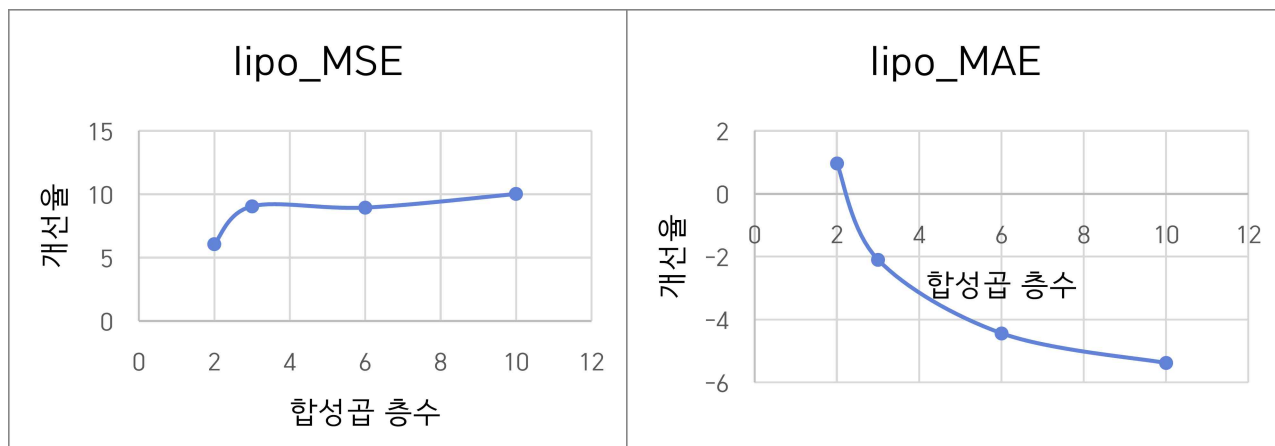
그림[17-1] PDBbind MAE

[그림17-2] PDBbind MSE

- GED 모델들의 층수별 개선 추세를 lipophilicity와 esol을 통해 대조해서 보자.

GResNet\_EGCN 모델과 GResNet\_EGCN\_Short 모델은 PDBbind MSE 부분에서 성능이 오히려 악화되기도 한다. 그래서 기준점인 EGCN을 새로 만든 모델들에 첨가한 합성곱 3층 이상의 GED들을 살펴보면, 모든 5가지 특성 예측의 MSE에서 잘 개선되고 lipophilicity를 제외하고 MAE에서도 개선되었다.





그림[18-1] GED Series MAE in lipophilicity

그림[18-2] GED Series MSE in lipophilicity

각 분자의 특성 예측에 있어 최종적으로 1개의 target값만을 출력해야 한다. 그런데 합성곱층에 data의 양이 너무 많으면 나중에 층 사이의 개수 차이가 크게 나기 때문에 점점 factor 수가 감소하는 naive 쪽으로 집중해서 모델을 만들었다. 그 결과, 각 모델의 경로를 합하는 구조에서도 naive 위주로 새롭게 구축하였다.

lipophilicity에서 MSE의 개선효과가 있던 GED Series를 그래프로 표현해서 tradeoff를 진행해보았으며 그 결과는 다음과 같다. MSE는 3층 이상에서는 비슷한 개선율을 갖고 있다. MAE에서는 2층을 제외하고 다소 성능이 나빠졌다. 본 연구에서는 MSE를 먼저 개선하고 MAE를 부가적으로 보고 있기 때문에 2층을 제외하였다. lipo\_MAE에서 나머지 층들이 악화되었기 때문에 여기에서 판단하기 어렵다. 그러나 그림[19]에서 MAE의 개선율을 보면 대체로 GED에 비해 Short\_GED가 개선되었다. MSE에서는 GED와 Short\_GED가 5가지의 분자특성 예측에 모두 개선효과가 있었기 때문에 MAE에 의해서 Short\_GED가 선택된다.

esol_MAE	1등	2등	3등	4등
	Short_GED	Very_Short_GED_3	Very_Short_GED	GED
	10.862	7.032	4.8	3.761
FreeSolv_MAE	1등	2등	3등	4등
	Short_GED	Very_Short_GED_3	GED	Very_Short_GED
	11.176	7.892	7.086	3.969
QM7_MAE	1등	2등	3등	4등
	Very_Short_GED_3	Short_GED	Very_Short_GED	GED
	12.861	10.116	5.111	2.595
lipophilicity_MAE	1등	2등	3등	4등
	Very_Short_GED	Very_Short_GED_3	Short_GED	GED
	0.963	-2.098	-4.435	-5.38
PDBbind_MAE	1등	2등	3등	4등
	Short_GED	Very_Short_GED_3	Very_Short_GED	GED
	10.862	7.032	4.8	3.338

esol_MSE	1등	2등	3등	4등
	Short_GED	GED	Very_Short_GED_3	Very_Short_GED
	21.872	19.263	14.827	14.239
FreeSolv_MSE	1등	2등	3등	4등
	Very_Short_GED	Short_GED	GED	Very_Short_GED_3
	22.306	16.241	15.747	12.894
QM7_MSE	1등	2등	3등	4등
	Short_GED	Very_Short_GED_3	Very_Short_GED	GED
	10.017	5.137	4.83	4.251
lipophilicity_MSE	1등	2등	3등	4등
	GED	Very_Short_GED_3	Short_GED	Very_Short_GED
	10.03	9.06	8.954	6.075
PDBbind_MSE	1등	2등	3등	4등
	GED	Short_GED	Very_Short_GED_3	Very_Short_GED
	2.06	0.729	0.469	-0.837

그림[19-1] MAE 개선율 등수

그림[19-2] MSE 개선율 등수

그림[19]에서 각 dataset별로 GED Series의 개선율에 대해 등수를 나타냈다.

lipophilicity만 본다면 MSE는 3층 이후로 뚜렷한 개선이 없으며 대체로 개선되었다. 그리고 MAE는 2층 이외에는 다 악화되었다. 다른 dataset에서 개선효과가 있었다 하더라도 이곳에서는 다른 경향성을 보이는 이유는 이 dataset의 특징을 살펴봐야 한다. 아래의 그림은 각 dataset에서 target값의 최대와 최소 그리고 둘의 간격을 나타냈다.

	ESOL	QM7	FreeSolv	PDBbind	lipophilicity
min	-11.6	-2.90351	-25.47	0.4	-1.5
max	1.56	5.114429	3.43	9.3	4.5
length	13.16	8.017939	28.9	8.9	6

그림[20] Dataset 별로 target값의 특징

본 연구 결과를 살펴보면, 가장 성능이 좋은 Short\_GED 모델의 경우 다른 데이터셋에서는 모두 성능개선을 나타낸 반면, lipophilicity 데이터셋만 MAE의 성능개선을 보이지 않았는데 그 이유를 살펴보면 다음과 같다. lipophilicity의 target 값은 가장 짧은 구간 길이 (최대값과 최소값의 차이) 를 가지고 있다. 즉, 제시되는 물질들 간의 특성 차이를 가장 반영하지 못한 dataset이다. ESOL과 lipophilicity의 length (max - min) 차이가 7 정도이며 이 값은 작은 차이라고 할 수 없다. 왜냐하면 ESOL과 lipophilicity의 값들은 상용로그가 적용된 값이기 때문에 작은 수치여도 실제 값은 크다. 그리고 ESOL과 비교하자면 주어진 물질들은 대체로 탄화수소 골격에 작용기가 붙어서 aqueous solubility와 분배계수가 달라진다. ESOL에서 모두 기본 골격은 소수성이지만 작용기 효과에 따라서 상대적으로 넓은 구간에서 구분된다. 물론 기본 골격의 구조와 크기의 영향도 있다. 그러나 lipophilicity의 값을 나타내는 logP가 (octanol/물) 분배계수를 나타내기 때문에 이 부분은 lipophilicity에서 더 크게 나타난다. 그러므로 분배계수가 크게 나오려면 물보단 octanol과 더 친해야 한다. 즉, 친수성에 영향을 미치는 작용기보다 소수성인 탄화수소 골격의 크기와 구조에 더 민감하다. 그러므로 층수가 높아짐에 따라 전체구조보다 특성 추출에 집중하기 때문에 다른 데이터셋보다 더 민감하게 악화되는 것이다. 이를 보완하고자 참고문헌[1]의 EGCN에선 분자의 ring 개수와 분자량을 삽입했고 본 연구에서는 합성곱층에 skip connection을 적용해서 MSE를 개선해 outlier를 줄였다.

#### • PDBbind dataset의 특징

비록 target 값의 구간이 길지만 이 데이터들의 특징은 바로 N:1로 대응된다는 점이다. 여러 화합물이 같은 binding affinity 값을 갖는다. 이렇게 되면 입력 정보 중 일부는 화합물을 구분하거나 target 값을 회귀하는 데에 기여도가 낮아질 수 있다. 반면에 합성곱층이 깊어

질수록 sparse한 data에서 더 특징을 추출하므로써 MSE 성능이 오른 것으로 보인다.

- Short\_GED가 tradeoff로 봤을 때 가장 보편적으로 개선된 model이다.

lipophilicity의 MAE에서 다소의 악화되지만 Short\_GED가 가장 개선된 모델이다. 다른 MAE에서 10% 이상 개선되었으며 MSE에서는 ESOL에서 22% 개선되는 등 좋은 결과를 얻었다. 그러나 MSE로 보면 GED도 건전한 성능을 내지만 Short\_GED에 비해 MAE에서 좋지 못한 성능을 보인다. 따라서 Short\_GED는 처음에 MSE를 개선해서 outlier를 감소시킨다는 목적에는 부합한다.

## 4. 향후 연구

GResNet의 취지에 맞게 더 깊은 합성곱 층을 쌓으면서도 성능을 개선한다는 것은 MSE 관점에서 보면 Short\_GED 모델은 성공적이라고 할 수 있다. 또한 MAE에서도 대체로 개선된 점을 볼 수 있다. 그러나 lipophilicity의 MAE를 고려해서 합성곱 층의 수를 줄이면 Very\_Short\_GED 모델이 비록 Short\_GED보다 다른 dataset에서의 MAE 개선율이 낮지만 모든 MAE의 경우에서도 개선된 결과를 얻을 수 있다. 하지만 Very\_Short\_GED 모델은 PDBbind MSE에서 악화되며 합성곱 층수를 6층에서 2층으로 줄였기 때문에 본래의 GResNet의 취지를 잘 달성했다고 볼 수 없다. 이것은 합성곱 층을 쌓을수록 원래의 정보와 전체적인 연결성을 잃어가기 때문으로 볼 수 있다. 이 점은 향후 그림[4]의 naive graph와 raw graph의 normalized adjacency matrix를 input으로 추가해서 보다 더 나은 성능으로 개선될 것이다.

## Supplement

“<https://github.com/KRICT-DATA/EGCN>” 에서 선행연구[1]을 이용할 수 있다. 각 데이터셋은 ChEMBL database, GDB-13 database, PBE0/tier2 basis set, <https://github.com/wengong-jin/chemprop>과 같은 곳에서 파생되었다.

## 5. 참고문헌

[1]GyoungS. Na, Hyun Woo Kim, and HyunjuChang.**Costless Performance Improvement in Machine Learning for Graph-Based Molecular Analysis.** *Journal of Chemical Information and Modeling.* 60 (3). 1137–1145. (2020).DOI: 10.1021/acs.jcim.9b00816

[2]Wu, Zhenqin and Ramsundar, Bharath and Feinberg, Evan N. and Gomes, Joseph and Geniesse, Caleb and Pappu, Aneesh S. and Leswing, Karl and Pande, Vijay. MoleculeNet: a benchmark for molecular machine learning. *Chem. Sci.* 9(2). 513–530. (2018). doi(10.1039/C7SC02664A). "datasets, Graph convolutional



models”

[3] Alexandru T. Balaban. Applications of graph theory in chemistry. Journal of Chemical Information and Computer Sciences. 25 (3), 334–343. (1985).

DOI: 10.1021/ci00047a033. "REACTION GRAPHS“

[4] Zhang, J.; and Meng, L. GResNet: Graph Residual Network for Reviving Deep GNNs from Suspended Animation. ArXiv abs/1909.05729. (2019).

[5] 고병철, Vanishing Gradient Problem(기울기 소멸 문제), 경인교육대학교 미래인재연구소 & 인공지능교육 연구소,

<http://computing.or.kr/14804/vanishing-gradient-problem%EA%B8%B0%EC%9A%B8%EA%B8%B0-%EC%86%8C%EB%A9%B8-%EB%AC%B8%EC%A0%9C/>

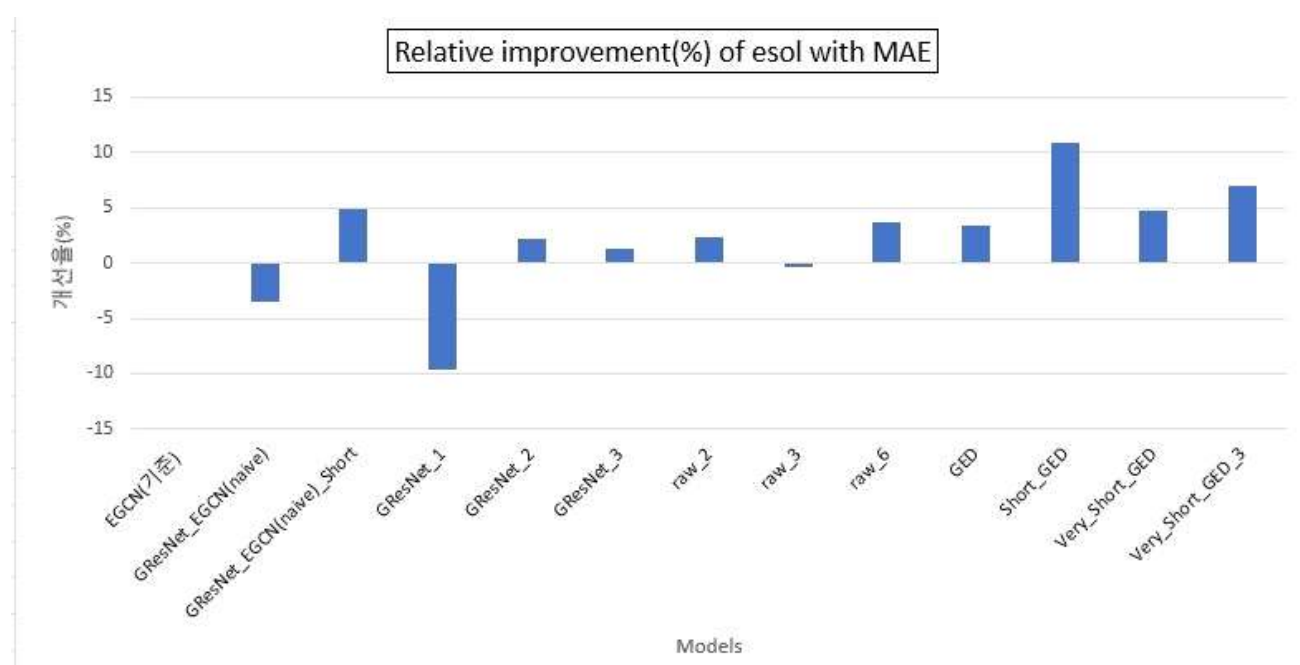
[6]Willmott, C., & Matsuura, K. Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance. Climate Research, 30, 79 – 82. (2005).

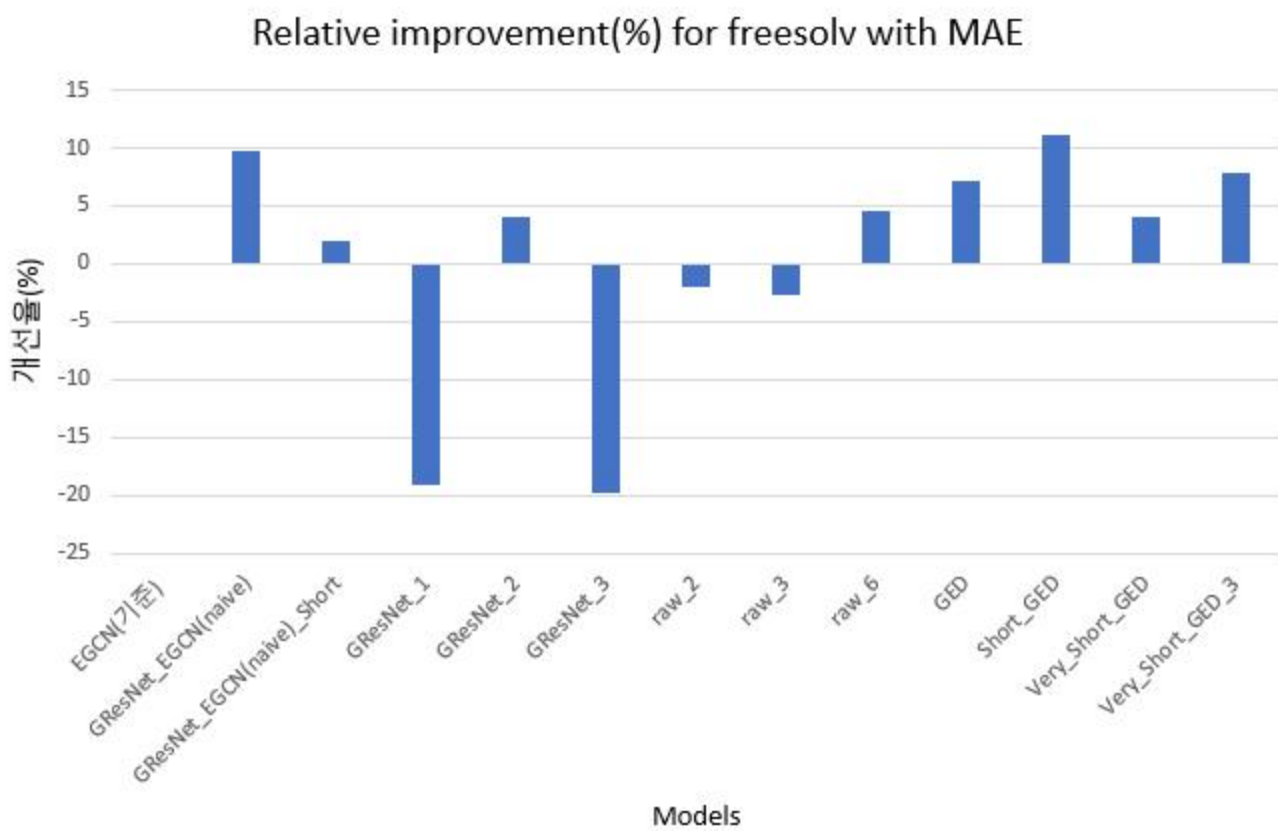
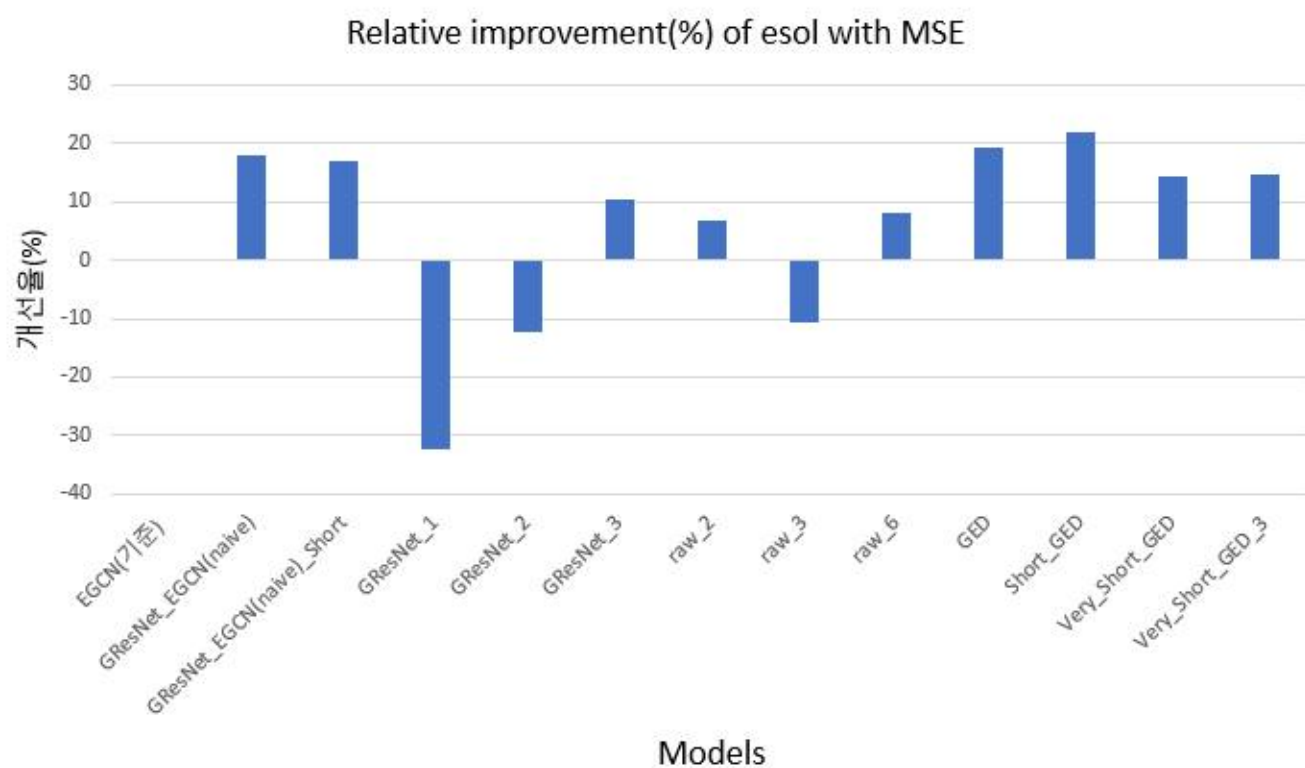
doi:10.3354/cr030079

[7] 결정계수, 두산백과, <https://terms.naver.com/entry.naver?docId=1059530&cid=40942&categoryId=32212>

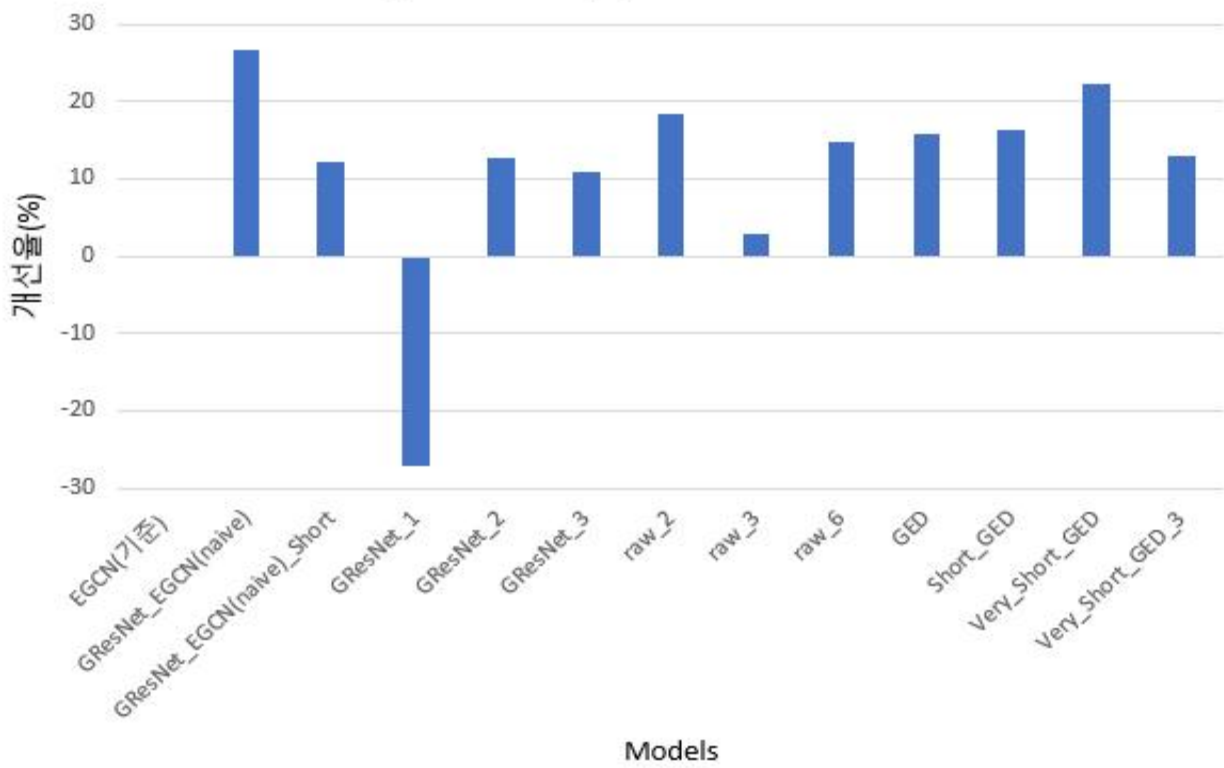
[8] 바라스 람순다르, 피터이스트먼, 패트릭 윌터스, 비제이 판테. (2020). 『생명과학을 위한 딥러닝』 (김태운, 역). 서울: 에이콘출판주식회사. 2019. 74–78 page. “SMILES”

## 5. 부록

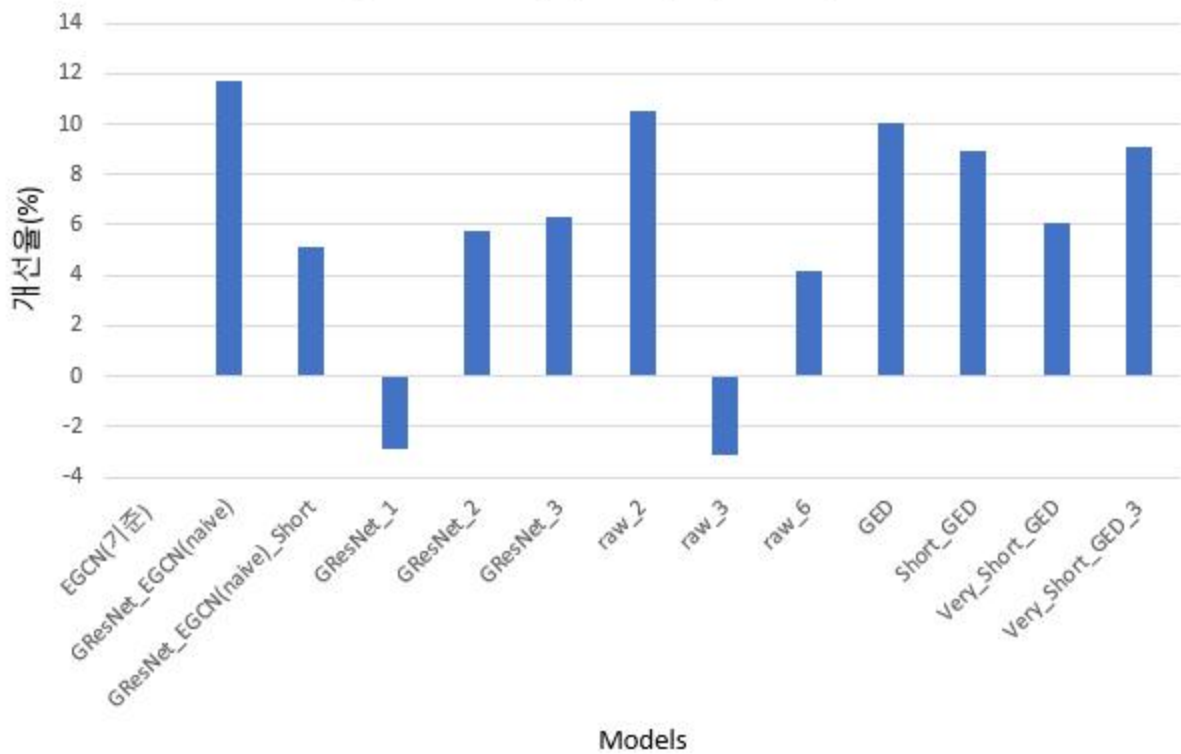




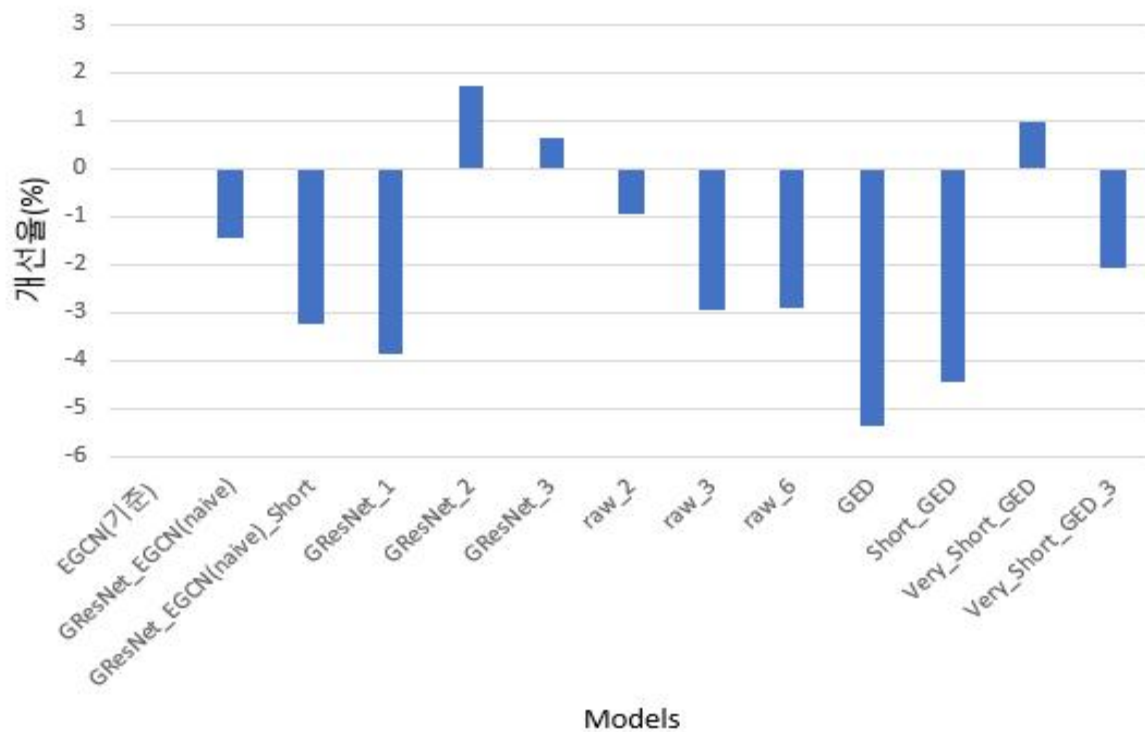
Relative improvement(%) for freesolv with MSE



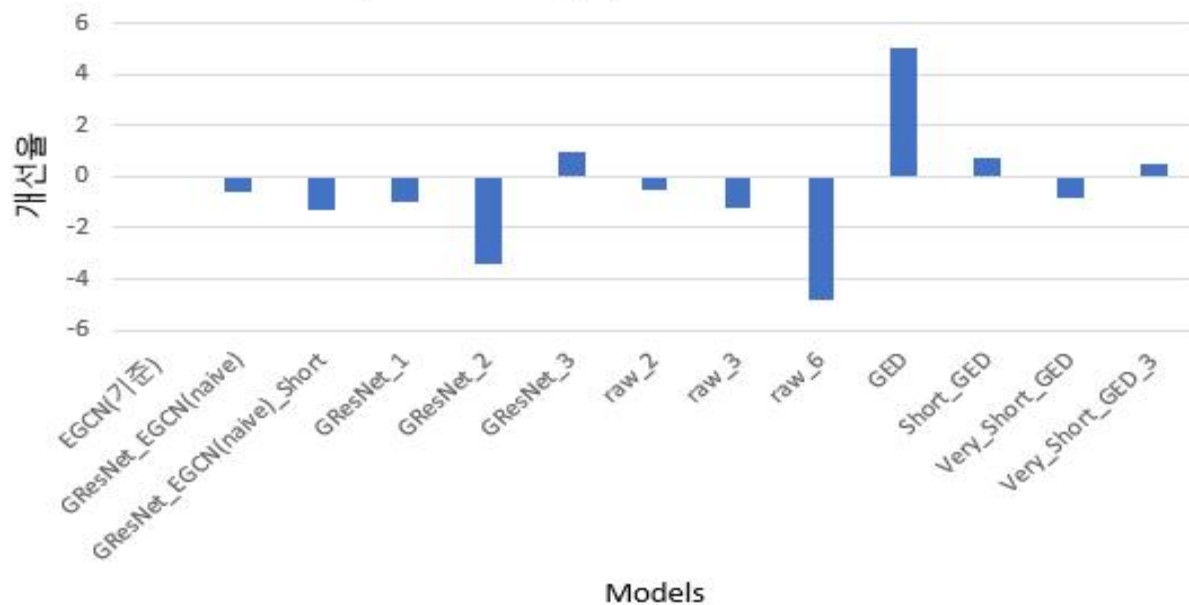
Relative improvement(%) for lipophilicity with MSE



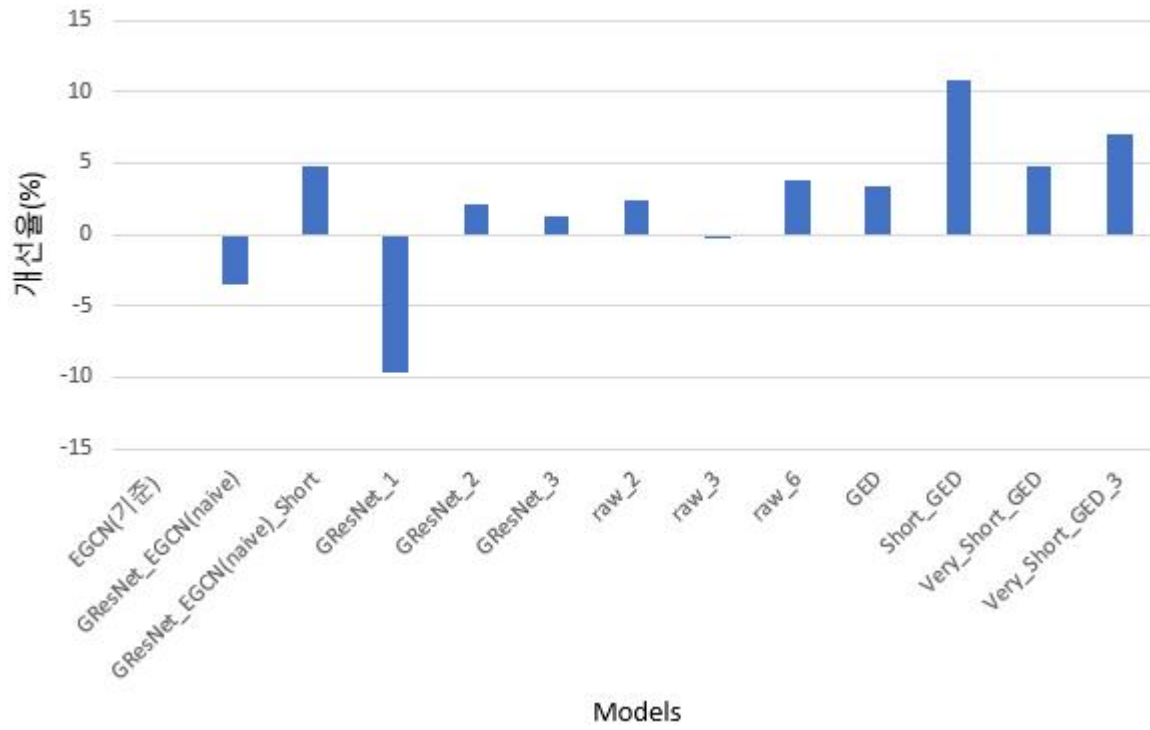
Relative improvement(%) for lipophilicity with MAE



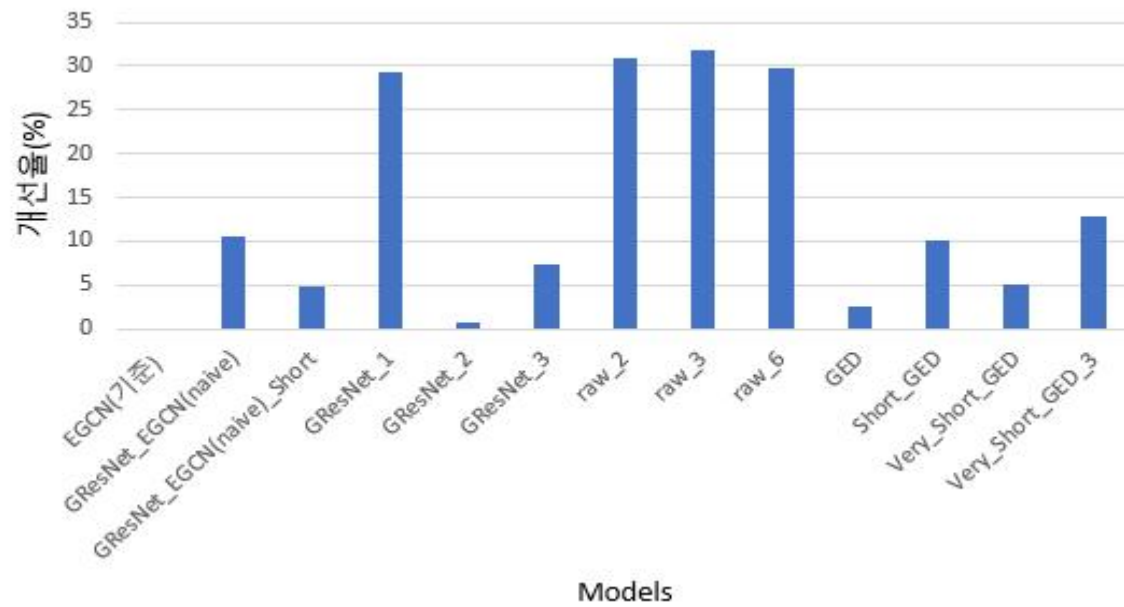
Relative improvement(%) for PDBbind with MSE



Relative improvement(%) for PDBbind with MAE



Relative improvement(%) for QM7 with MAE



Relative improvement(%) for QM7 with MSE

