

GraphDTA

Prediction of drug-target binding affinity
using graph convolutional networks

2020.10.07

Hyeonsu Lee
KAICD



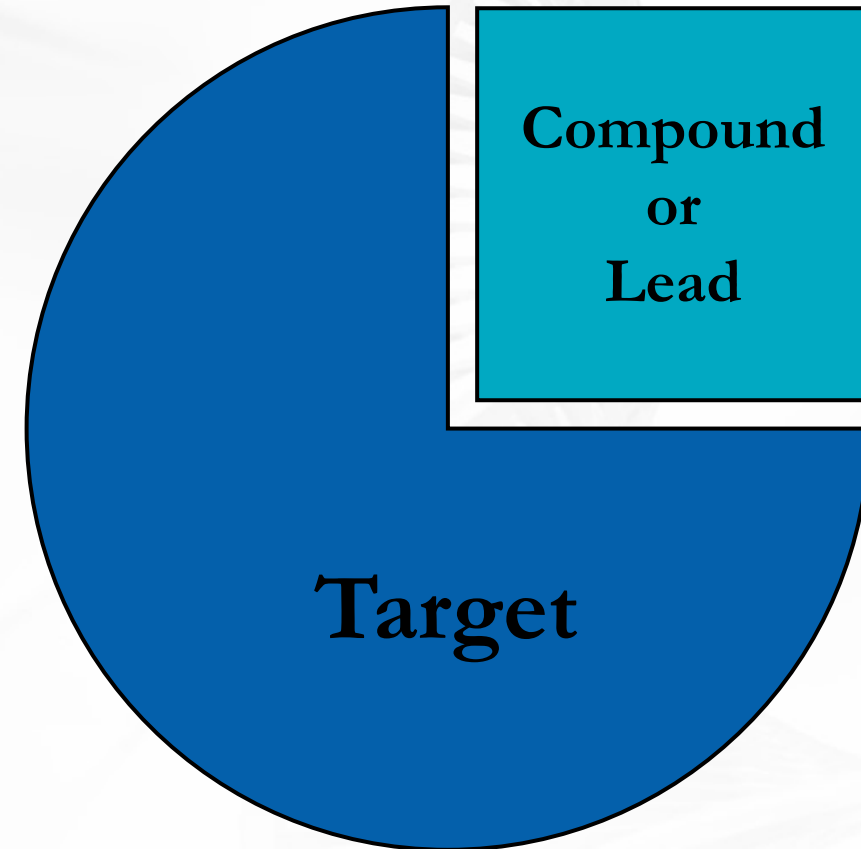
Time	Title	Contents
09:00 ~ 09:40	Introduction - GraphDTA	Introduction, Data Representation
10:00 ~ 10:40	Graph Structure	Graph, Graph Representation
11:00 ~ 12:00	Graph Convolutional Networks	Graph Convolution, Graph Convolutional Networks
12:00 ~ 13:00	Lunch Time	
13:00 ~ 13:40	GraphDTA Practice(Google Colab)	Data Structure Model Structure Predict with Pretrained Model
14:00 ~ 14:40		
15:00 ~ 15:40		
16:00 ~ 16:40	Introduction – Auto Encoder, Generative Models	AE, Stacked AE, VAE, feat.GAN
17:00 ~ 18:00	VAE Demo(Google Colab)	CHEM VAE Demo

Drug-Target Interaction

DTI

• 용어

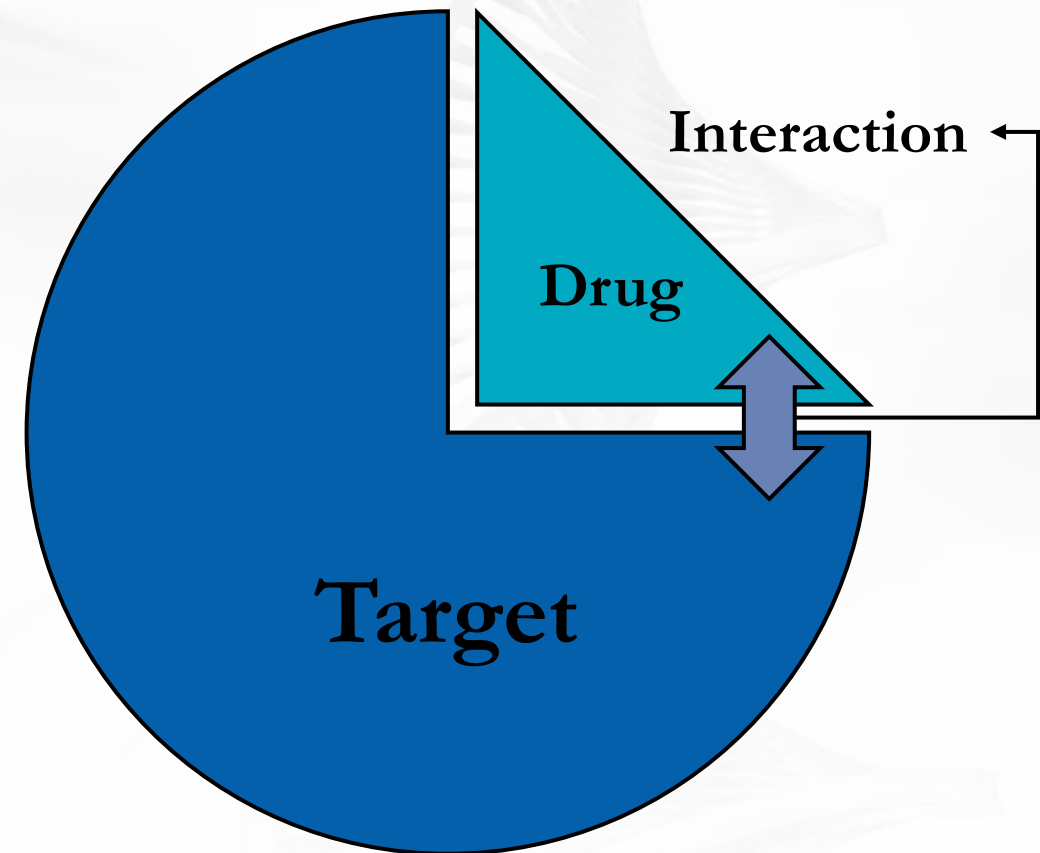
- Target : 질병과 관련된 특정 유기체 내의 분자
(단백질)
- Compound or Lead : Target과 결합하여 Target
을 제어할 수 있는 화학 물질



DTI

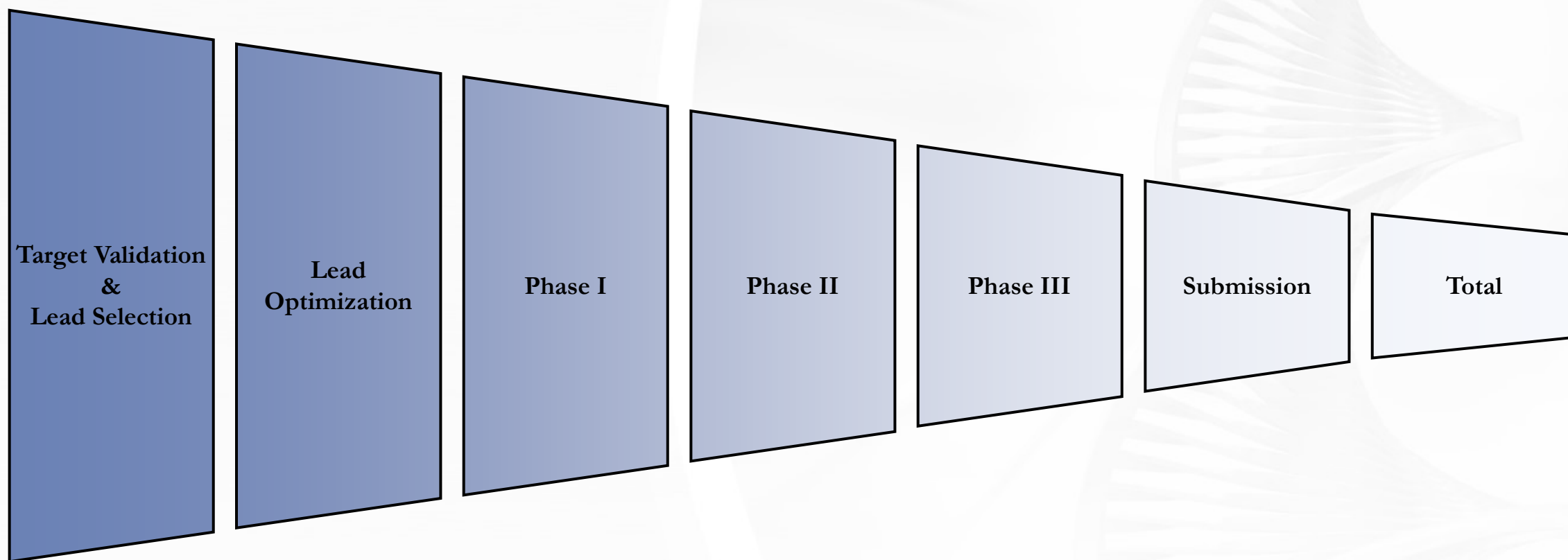
• 용어

- Properties : 질병 치료를 위해 Compound or Lead 에서 원하는 모든 특성(뛰어난 효과, 낮은 부작용, 낮은 독성)
- Drug : Properties가 완전히 최적화된 Compound
- Interaction : Target과 Drug 사이의 상호 작용
- DTI : Interaction 예측을 통한 **신약후보물질** 도출
- DTA : Binding Affinity를 통해 interaction을 예측



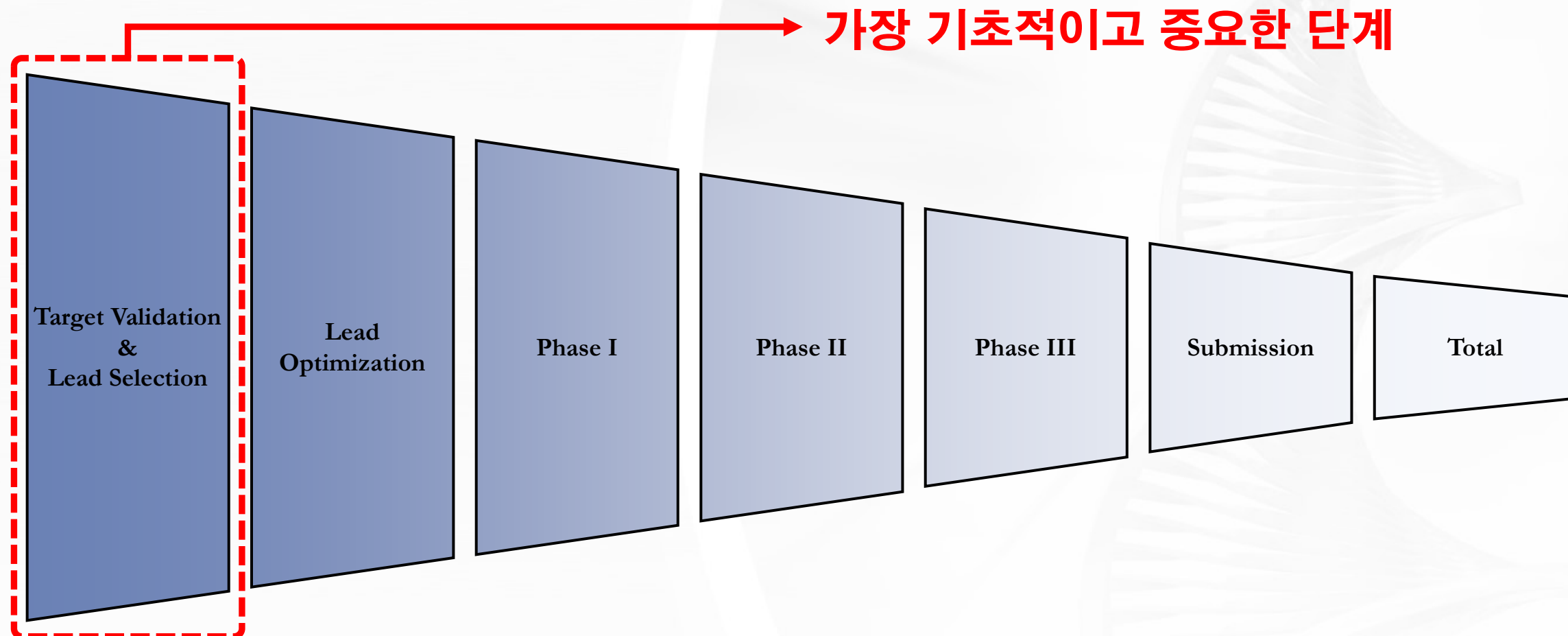
DTI

- 개요



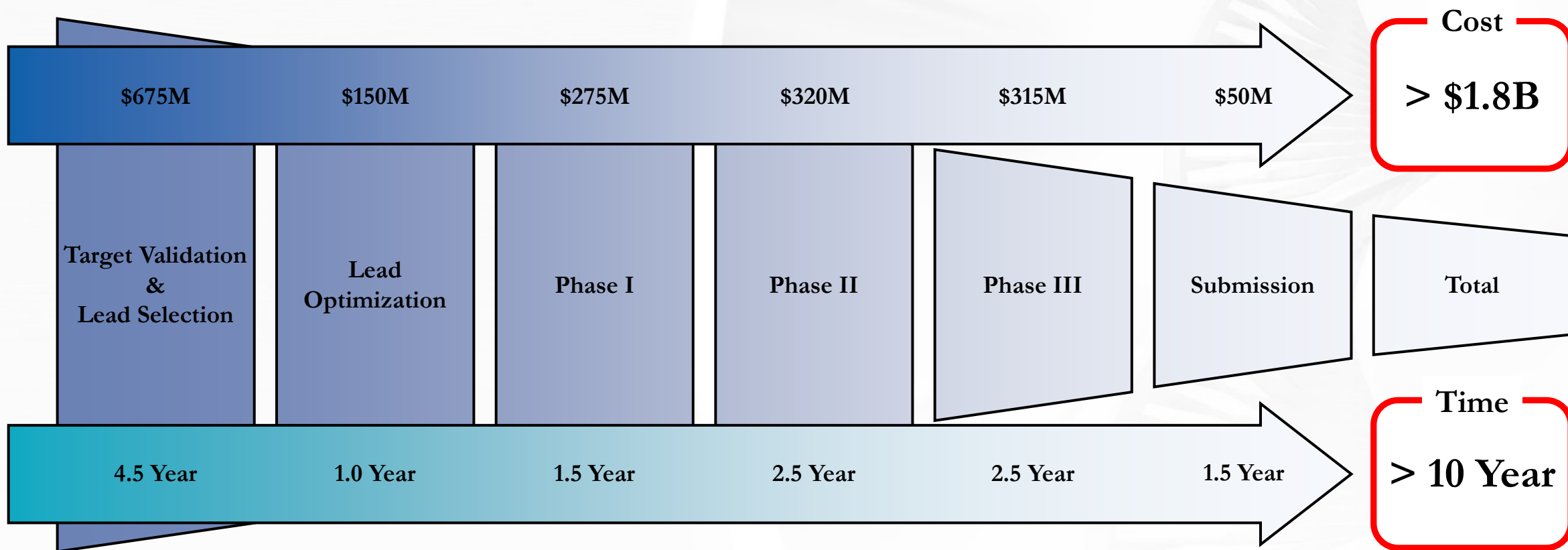
DTI

• 개요



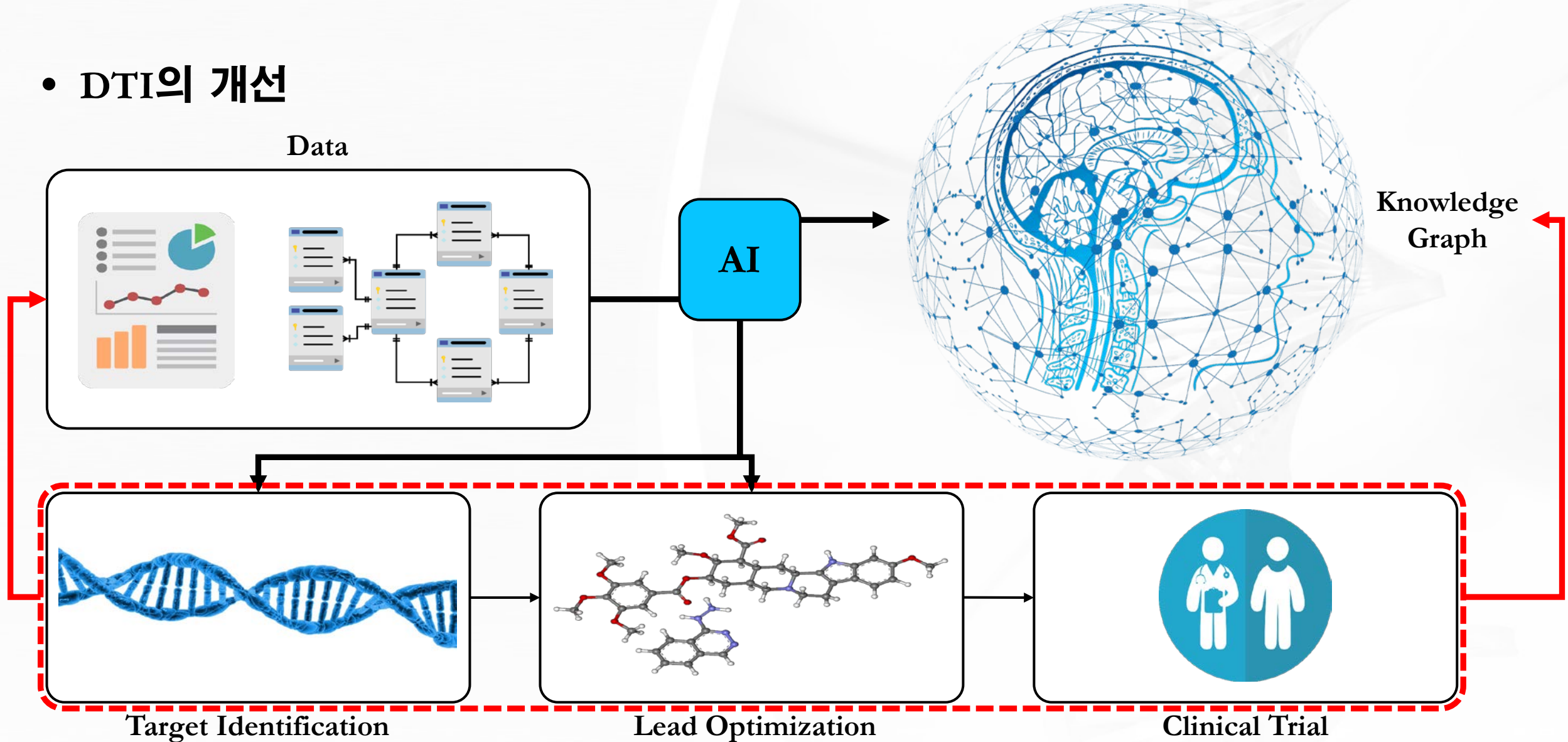
DTI

- 한계

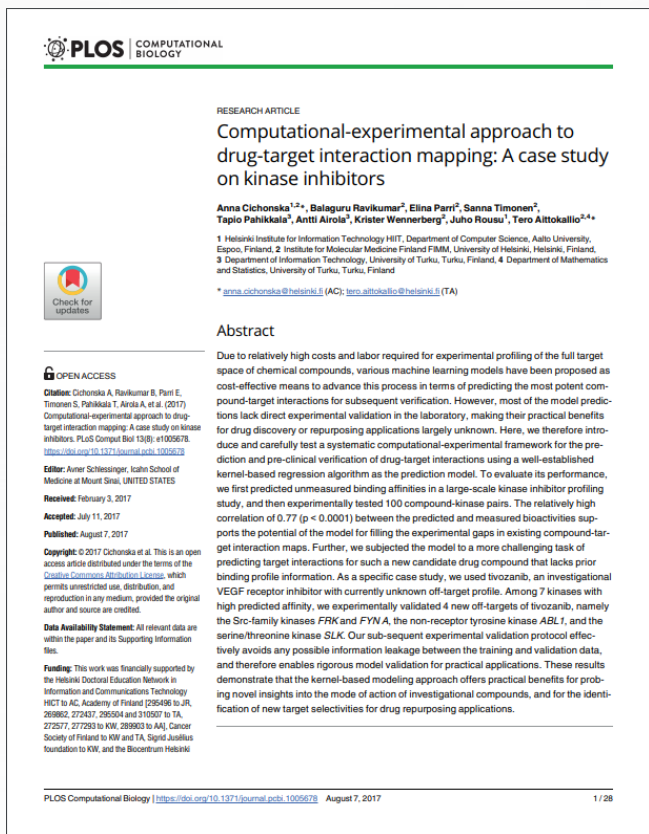


DTI

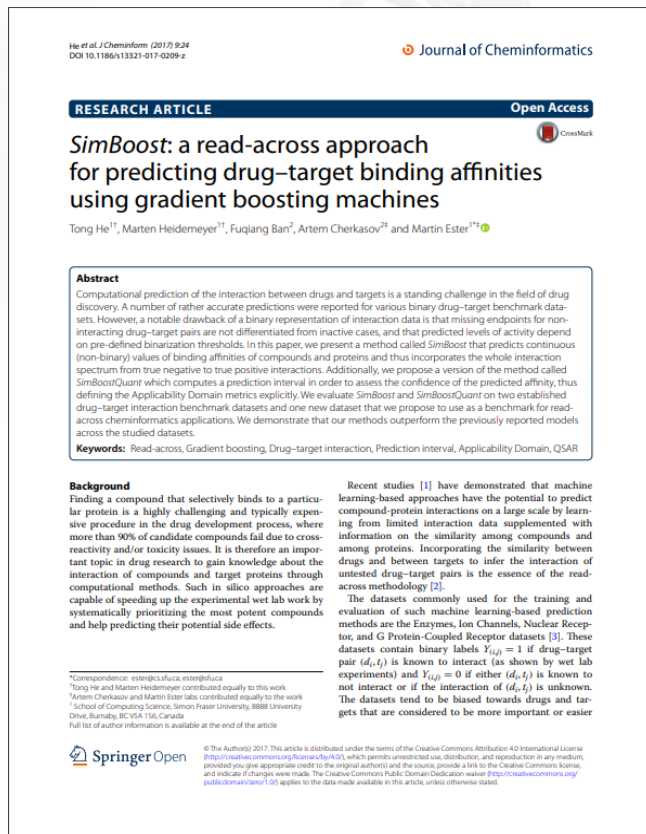
• DTI의 개선



• 기존 연구

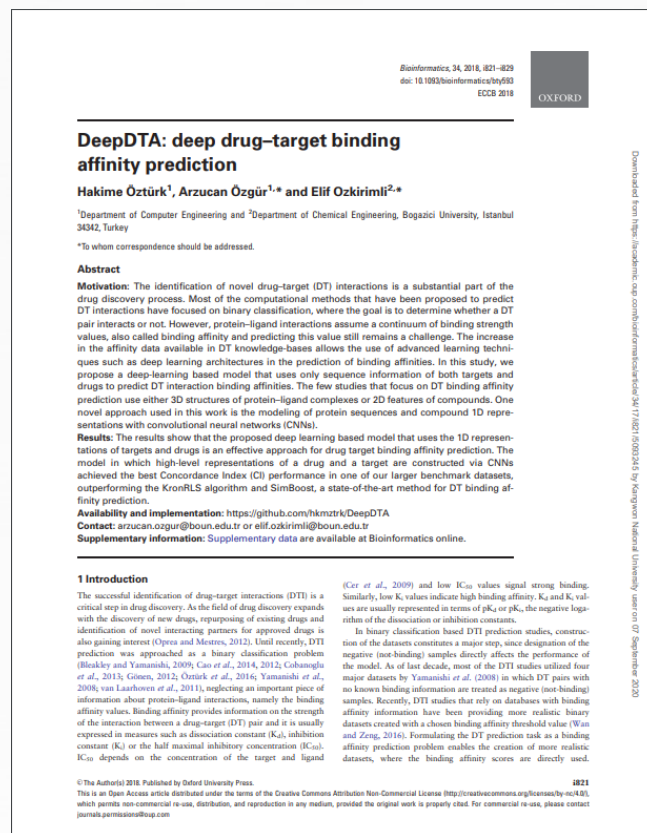


KronRLS



SimBoost

• 최근 연구



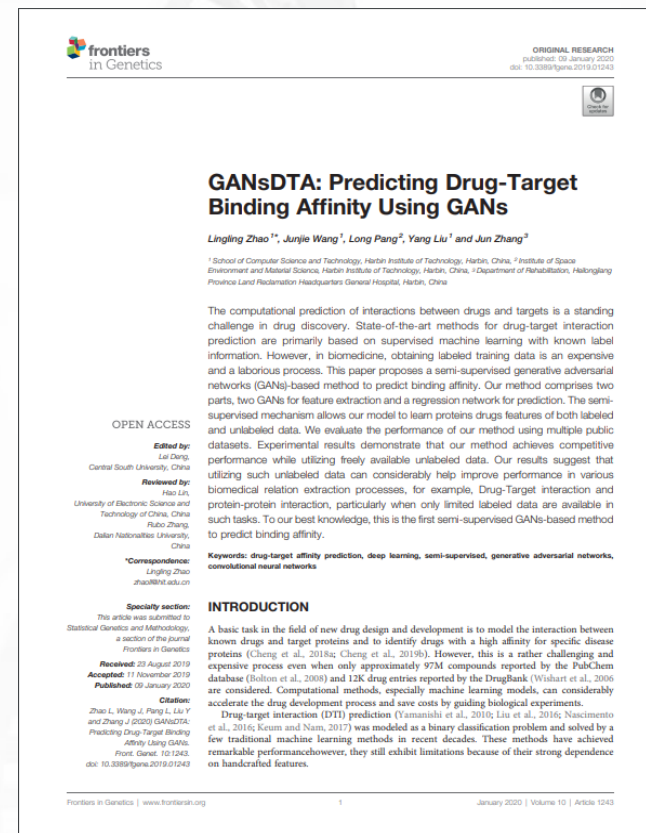
DeepDTA

Hyeonsu Lee (KAICD)



WideDTA

GraphDTA

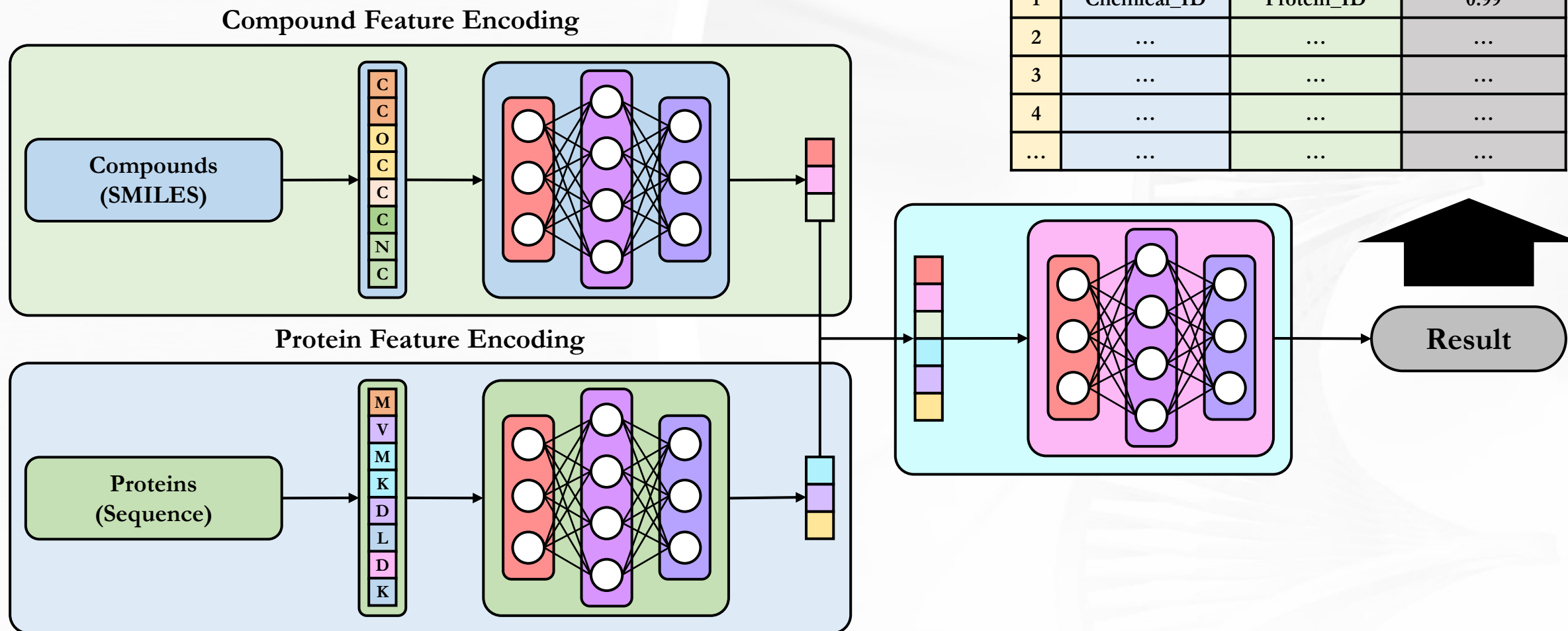


GANsDTA

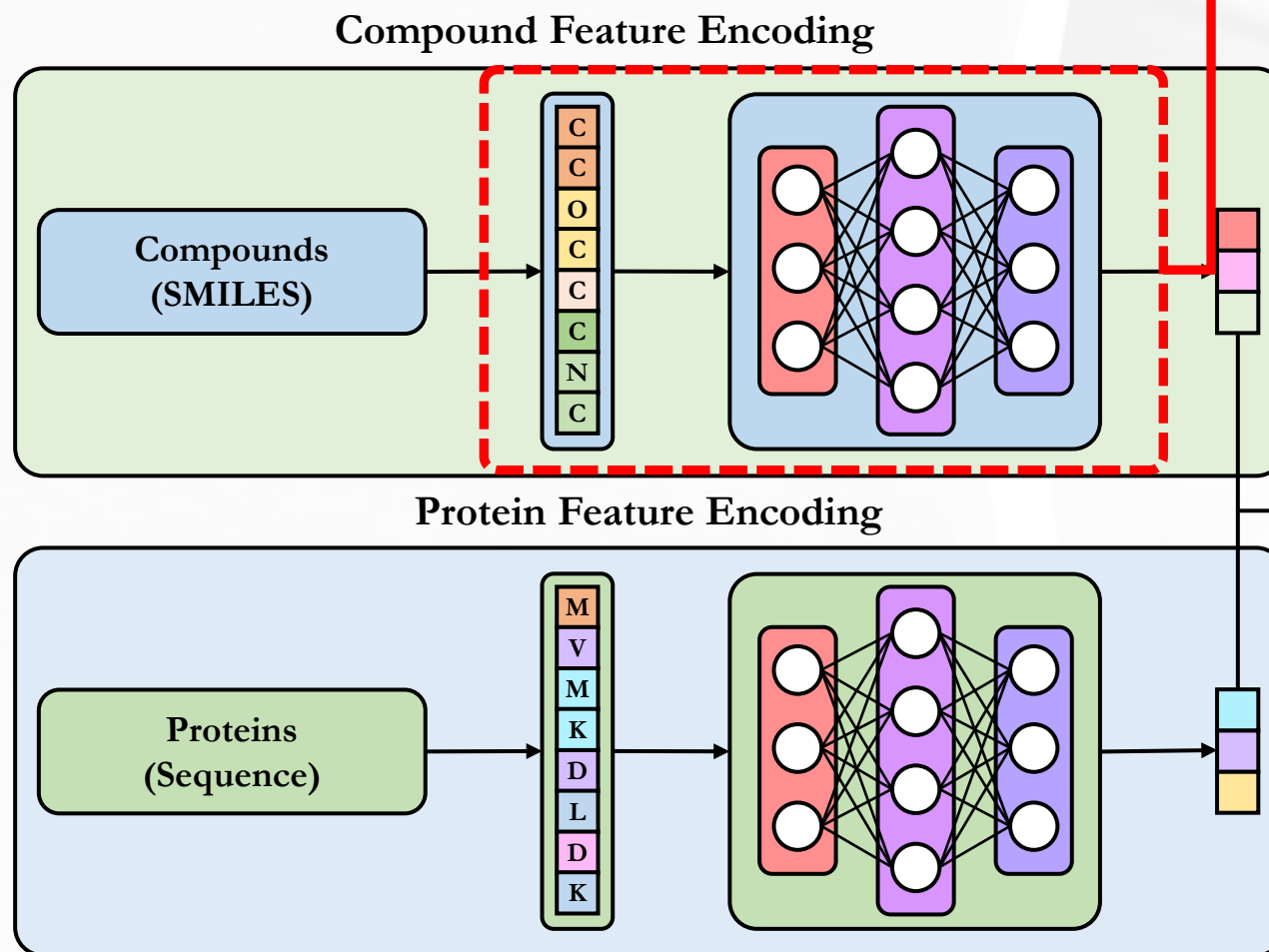
October 07, 2020

DTI

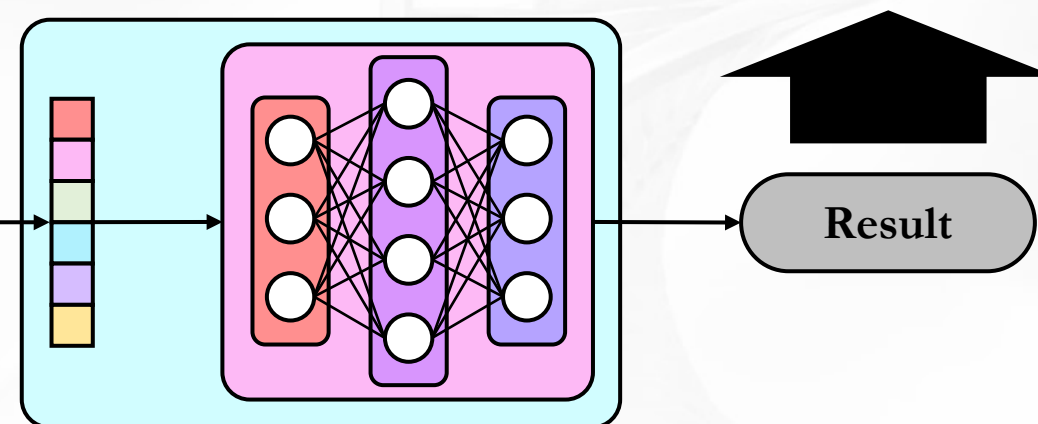
• 프로세스(DeepDTA)



DTI

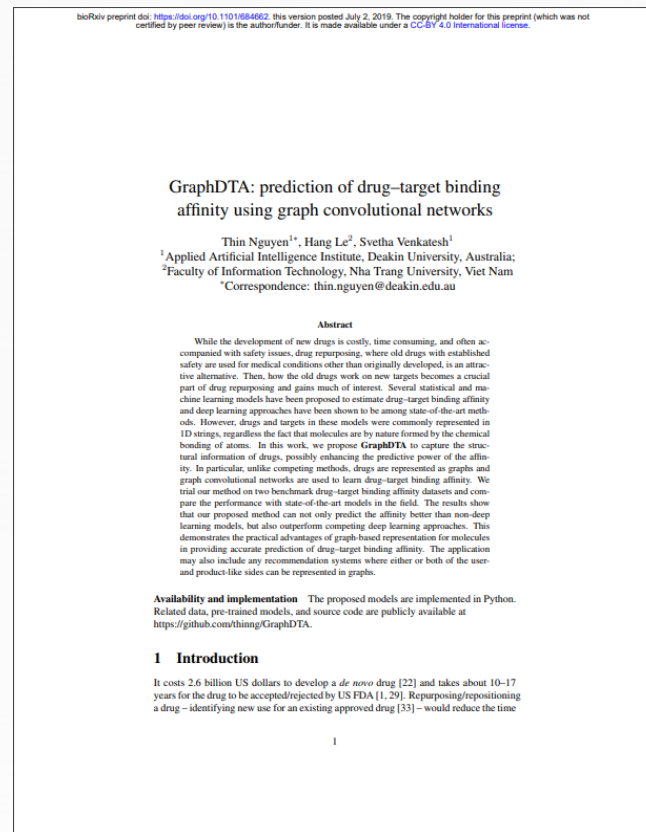
다양한 방법이 존재!

#	Compound	Protein	Prediction
1	Chemical_ID	Protein_ID	0.99
2
3
4
...



DTI

• Proposed Model - GraphDTA



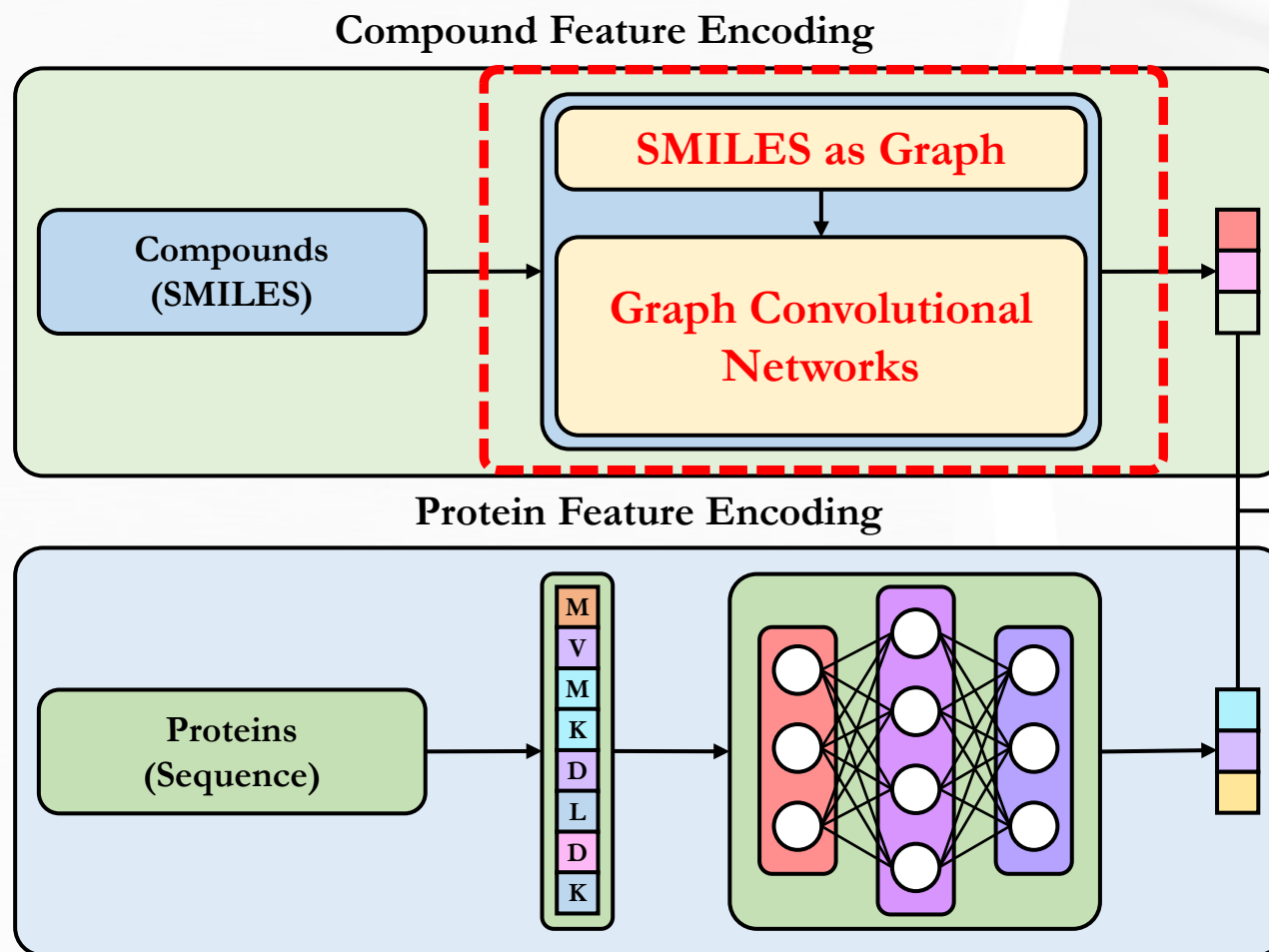
- 입력 : **그래프** 형식으로 나타낸 화학 화합물(Drug) 정보
- 입력 : 시퀀스 형태의 단백질(Target) 정보
- 출력 : 화학 화합물과 단백질 사이의 **결합 친화도** 예측 점수
- 기존 모델보다 **더 뛰어난 성능**을 보임

Method	Protein rep.	Compound rep.	CI	MSE
Baseline models				
DeepDTA	Smith-Waterman	Pubchem-Sim	0.790	0.608
DeepDTA	Smith-Waterman	1D	0.886	0.420
DeepDTA	1D	Pubchem-Sim	0.835	0.419
KronRLS	Smith-Waterman	Pubchem-Sim	0.871	0.379
SimBoost	Smith-Waterman	Pubchem-Sim	0.872	0.282
DeepDTA	1D	1D	0.878	0.261
WideDTA	1D + PDM	1D + LMCS	0.886	0.262
Proposed model - GraphDTA				
GCN [17]	1D	Graph	0.880	0.254
GAT_GCN	1D	Graph	0.881	0.245
GAT [37]	1D	Graph	0.892	0.232
GIN [40]	1D	Graph	0.893	0.229

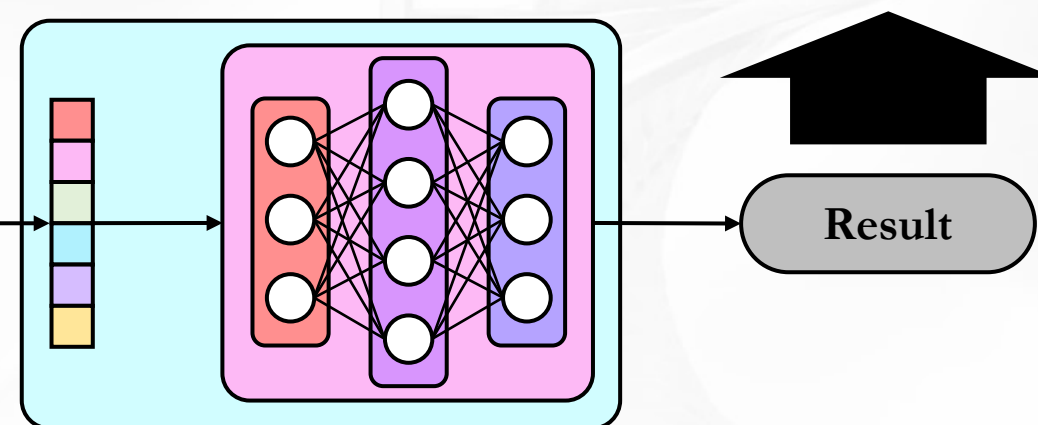
HOW?

DTI

- Proposed Model - GraphDTA

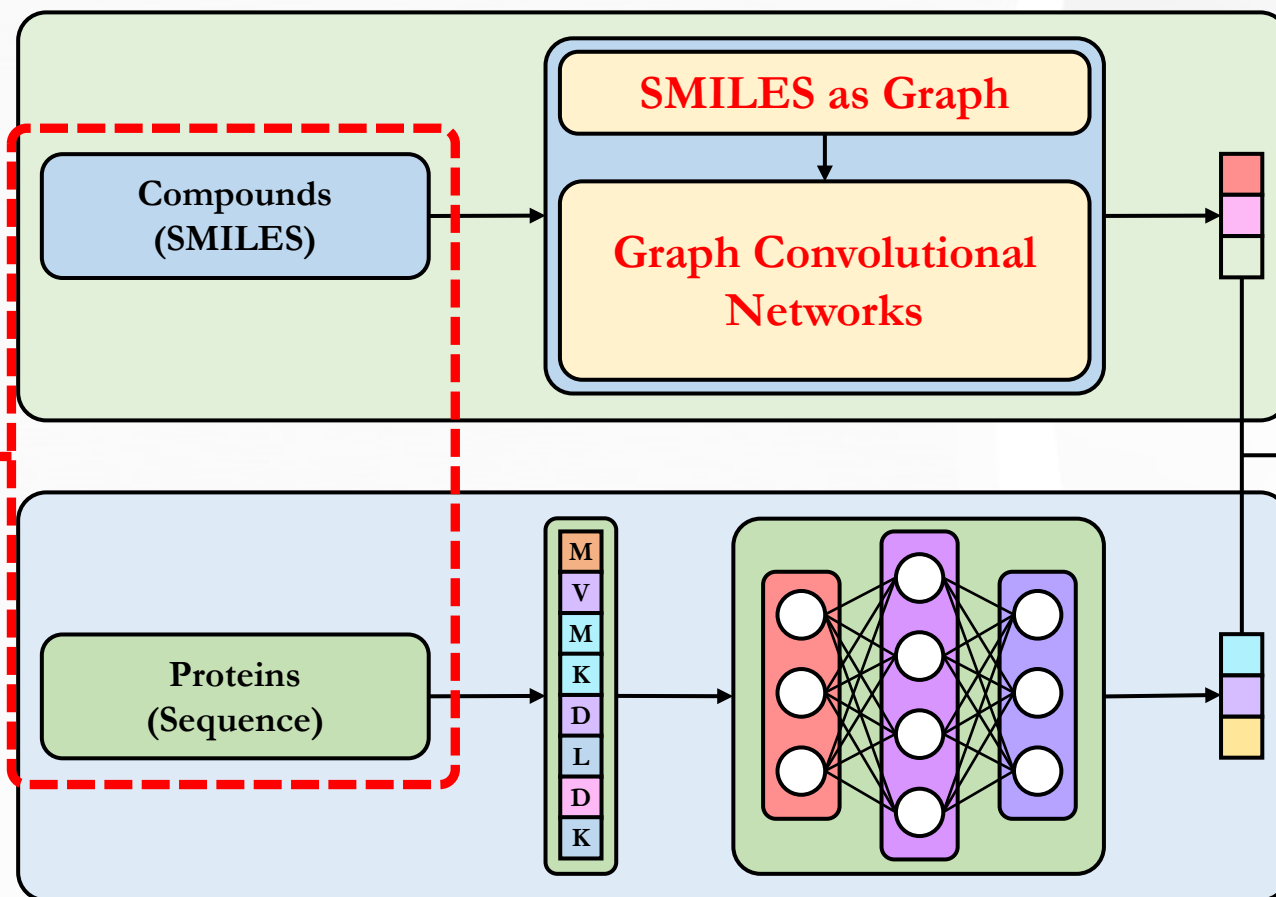


#	Compound	Protein	Prediction
1	Chemical_ID	Protein_ID	0.99
2
3
4
...



Datasets

Here!



#	Compound	Protein	Prediction
1	Chemical_ID	Protein_ID	0.99
2
3
4
...

Result

Datasets

- Davis

- 결합 친화도 점수 : 평형 이온화 상수 K_D

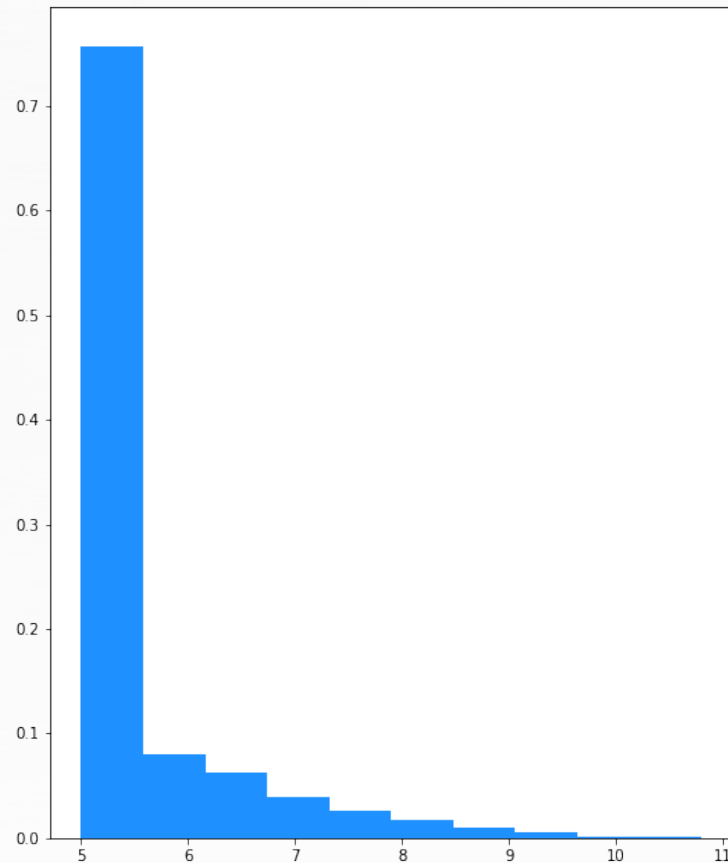
- Kiba

- 결합 친화도 점수 : KIBA score

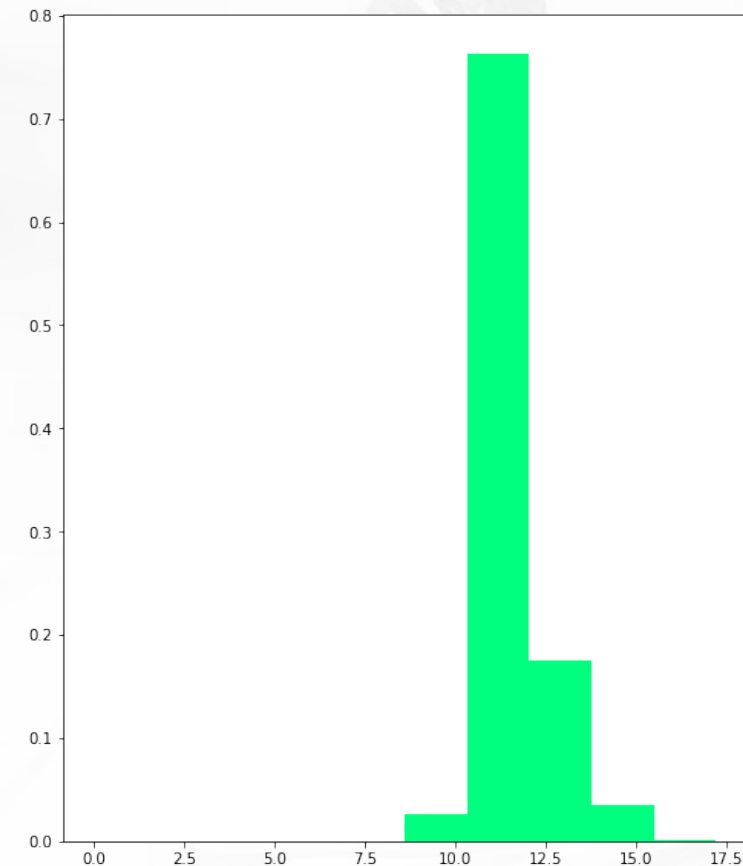
Dataset Name	Proteins	Compounds	Affinity value	Score
Davis	442	72	5.0 ~ 10.8	K_D
Kiba	229	2,116	0.0 ~ 17.2	KIBA

Datasets

- Affinity Distribution



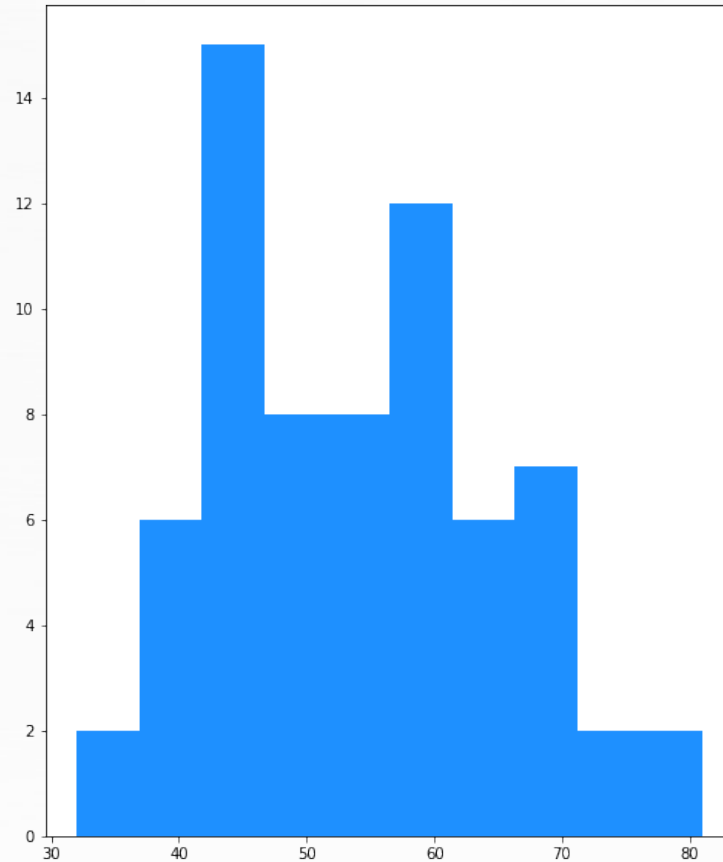
Davis Affinity Distribution



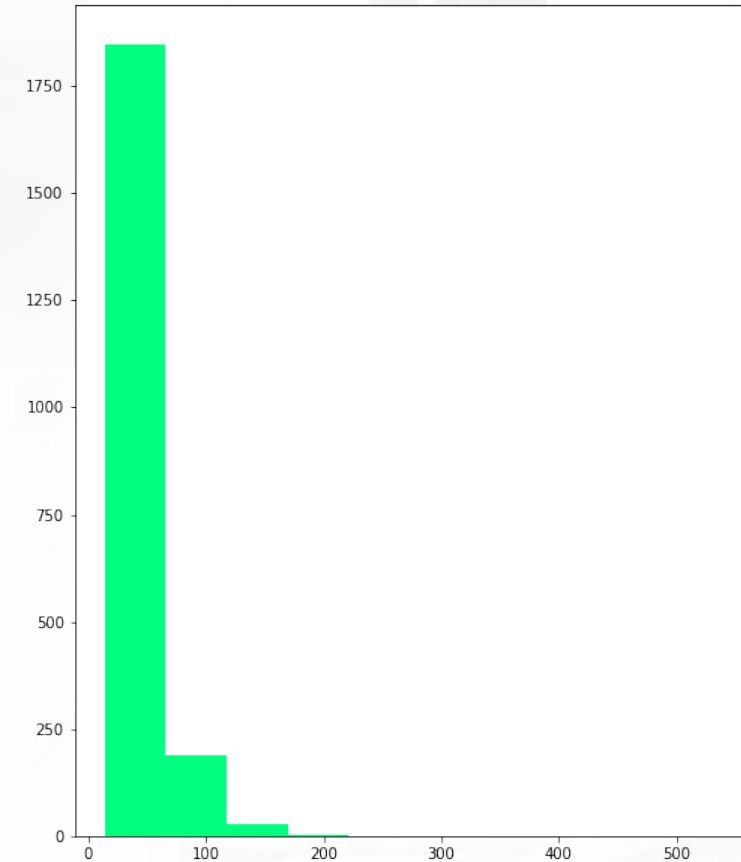
Kiba Affinity Distribution

Datasets

- Length of SMILES Distribution



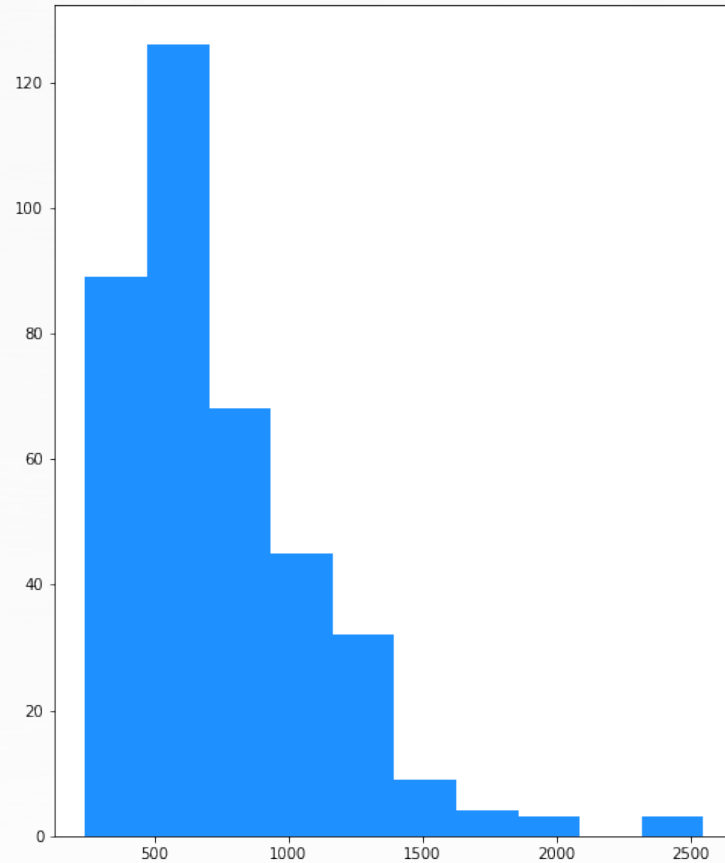
Davis Length of SMILES Distribution



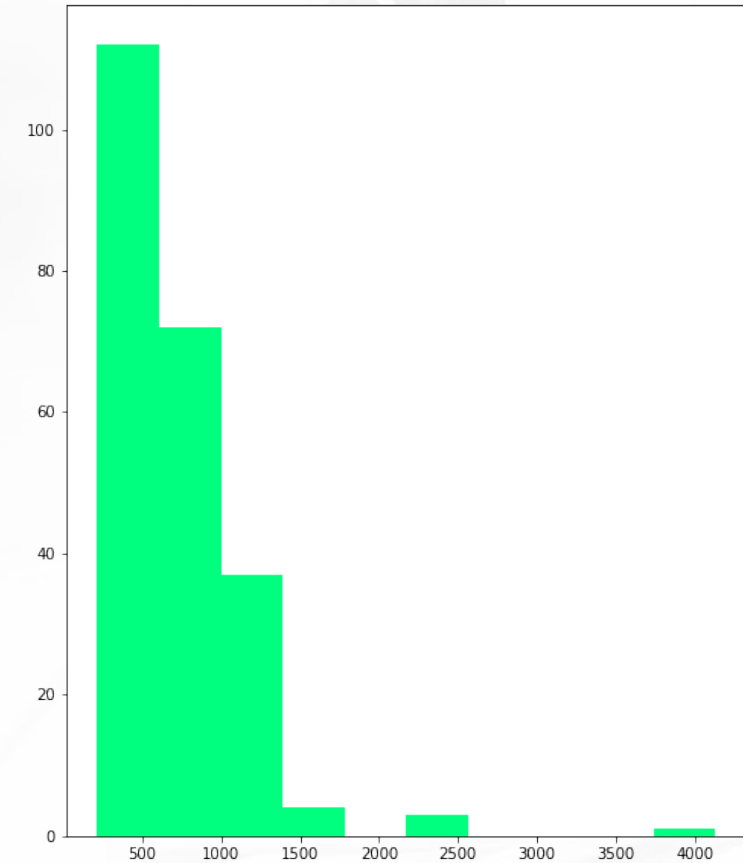
Kiba Length of SMILES Distribution

Datasets

- Length of Protein Sequence Distribution



Davis Length of Protein Sequence Distribution

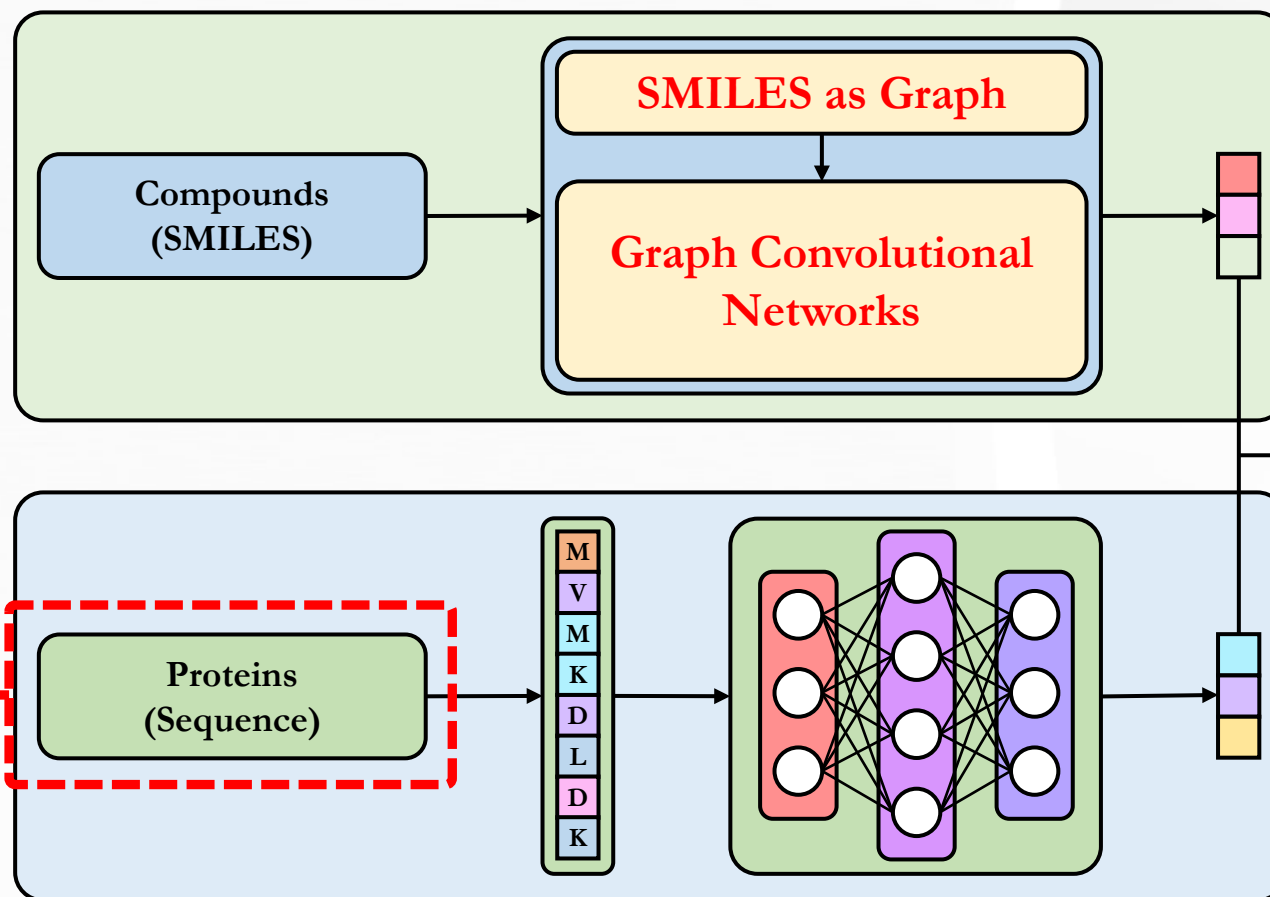


Kiba Length of Protein Sequence Distribution

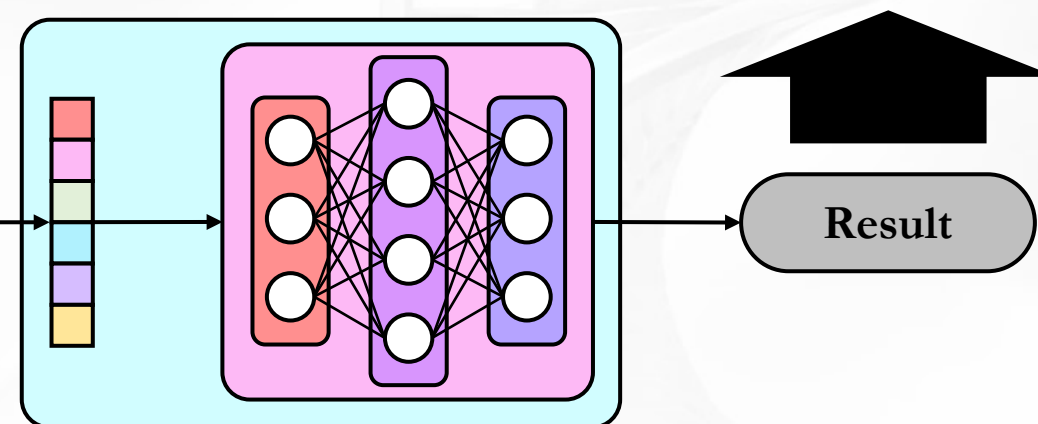
Protein Representation

- UniProt -

Here!



#	Compound	Protein	Prediction
1	Chemical_ID	Protein_ID	0.99
2
3
4
...



Protein Representation

- UniProt

- 단백질 서열 및 기능 정보에 대한 자유롭게 액세스 할 수 있는 데이터베이스
- 'J'를 제외한 25개의 아미노산으로 단백질 염기 서열 표현

<https://www.uniprot.org/>

Protein Representation

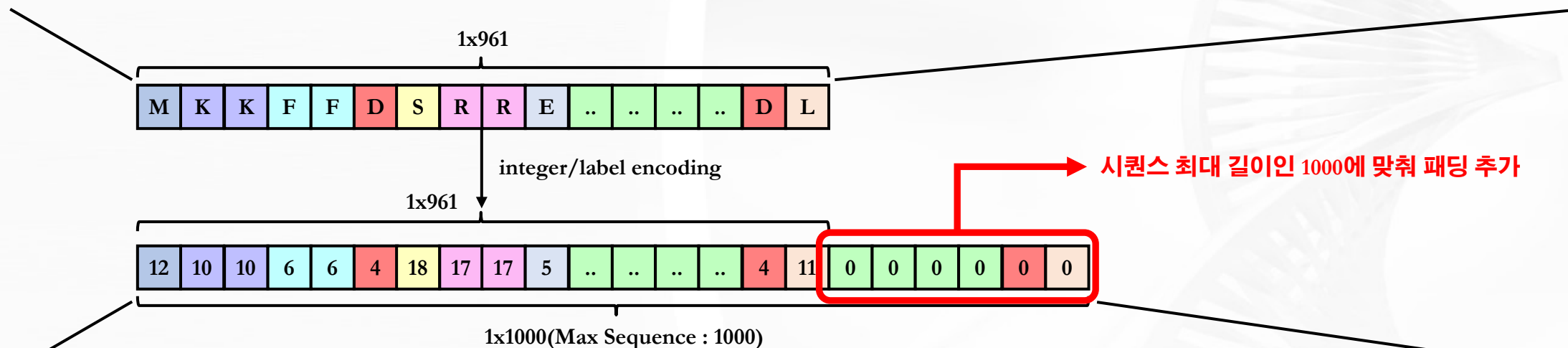
• 예시 - Real Data

Protein_ID

Protein_Sequence

```
"AAK1":
"MSRREQQGSGLGSGSSGGGSGTSLGSGYIGRVFQIGRQQTIVDEVLAEGGFATVFLVRTSNGMKCALKRMFVNNHDLQVCKREIQIMRDLGSHKNIYGYIDSSINNVSQGVWEVLILMDFCRGGQVNLNMQRLQGTFTENEVLQTFCDTCEAVARLHQCKTPIIHRDLKVENILLHGRGHVYLC
SATNKFQNPQTEGVNAVEDEIKKYITLSYRAPEMVNLVSGKIITTKADIWALGCLLYKLCYFTLPFGESQVAICDGNFTIPDNSRYSQDMHCLIRYMLEPDPDKRPDIYQVSYSFSLKKKECPIPNVQNSPIAKLPEPVKASEAAAKKTQPKARLTDPIPTTETSIAPRQRPKAGQTQPNPILPIQALTP
ATVQPPPPQAAGSSNQPLLASVPQPKPQAPPQPLPQTQAKQPAPPTPQQTSTQAQGLPAQAQATPQHQQQLFLKQQQQQQQPPPAQQQPAQTFYQQQQQATQQFQAVHPATQKPAIAQFVVVSQGGSSQQLMQNFYQQQQQQQQQQQQQLLATALHQQQLMTQQAALQKQPTMAAGQQPQPQAAAPQAP
PAIOAPVROOPKQVOTTPPPAVOGQKVGSLTPPSSPKTORAGHRRILSDVTHSAVGVPAKSKTOLLOAAAAEASLNKSKSATTTPSGSPRTSQONVYNPSEGSTWNPFDNDFSKLTAEELLNKDFAKLGEQKHPEKLGGSASELIPGFQSTQGDFAFTTSFAGTAEKRRGGGTVDGSLPLLSVSDPFPTLOV
PEKLEGLKSPDLSLLPDLPMTPDPPGSTDAVTEKADVAVESLIPGLEPPVQRLPSQTESVTSNRDLSLTGDSLLDCSLLSNPTIDLLEEFAPTAISAPVHKAAEDSNLISGFDVPEGSDKVAEDEFDPIPVLLTKNPQGGHSHRNSSGSSSESLNLRSLLLVDQLIDL", "ABL1(E259K)":
"PFWKILNPLLERGTYYYFMGQPGKVLGDQRRPSPALHFIKAGGKKESSRHGGPHCNVFEHEALQRPVASFDFPQGLSEAAARNWSENKLAGPSENDPNLFVALYDFVASGDNTLSITKGEKLRVLGYNHNGWECEAQTNGQGWVPSNYITPVNSLEKHSWYHGPVSRNAEYLLSSGINGSFLVRESE
QRSISLRYEGRVYHYRINTASDGKLYVSSSRFNTLAELVHHHSTVADGLITTLHYPAKRNKPTVYGVSPPNYDKWEMERTDITMKHKLGGGQYGEVYEGVWKYSLTVAVKTLKEDTMEVEEFLKEAAMVKEIKHPNLVQLLGVCRTREPPFYITTEFMTYGNLLDYLRECNRQEVNAVVLVYMATQISSAMEY
KNFIHRDLAARNCLVGENHLVKVADFGLSRLMTGDTYTAHAGAKFPIKWTAPESLAYNKFSIKSDVWAFVGLLWEIATYGMSPYPGIDLQSVYELLEKDYRMERPEGCEKVVYELMRACQWNPSPDRPSFAEIHQAFETMFQESSISDEVEKELGKQGVRGAVSTLLQAPLEPTKTRTSRRAAEHRDITDVP
KGQGESDPLDHEPAVSPLLPRKERGPPEGGLNEDERLLPKDKKTNLFSALIKKKKTAPTTPKRSSSFREMDGQPERRGAGEEGRDISNGALFTPLDADPAKSPKPSNGAGVPNGALRESGGSGFRSPHLWKKSSSTLTSSRLATGEEEGGGSSSKRFLRSCSASCVPHGAKDTEWRSVTLPRDLQSTGRQF
TFGGHKSEKPALPRKRAGENRSDQVTRGTVTTPPRLVKKNEEAADEVFKDIMESSPGSSPPNLTTPKPLRRQVTVAPASGLPHKEEAGKGSALGTPAAAEPTVPTSKAGSGAPGGTSKGAPEESRVRHKKHSSSPGRDKGKLSRLKPAPPPPPAASAGKAGGKPSQSPSQAAGEAVLGAKTATSLVDVAVNSD
PSQPGEGLKKPVLPAATPKPQSAKPSGTPISAPVPSTLPSASSALAGDQPSSTAFIPLISTRVSLRKTRQPPERIASGAIITKGVLDSTALCLAISRNSEQMASHSAVLEAGKNLYTFCVSYVDSIQQMRNKFATREATNKLNNLRELQICPATAGSGPAATQDFSKLLSSVKEISDIVQR", "ABL1
(F317I)":
"PFWKILNPLLERGTYYYFMGQPGKVLGDQRRPSPALHFIKAGGKKESSRHGGPHCNVFEHEALQRPVASFDFPQGLSEAAARNWSENKLAGPSENDPNLFVALYDFVASGDNTLSITKGEKLRVLGYNHNGWECEAQTNGQGWVPSNYITPVNSLEKHSWYHGPVSRNAEYLLSSGINGSFLVRESE
QRSISLRYEGRVYHYRINTASDGKLYVSSSRFNTLAELVHHHSTVADGLITTLHYPAKRNKPTVYGVSPPNYDKWEMERTDITMKHKLGGGQYGEVYEGVWKYSLTVAVKTLKEDTMEVEEFLKEAAMVKEIKHPNLVQLLGVCRTREPPFYITTEFMTYGNLLDYLRECNRQEVNAVVLVYMATQISSAMEY
KNFIHRDLAARNCLVGENHLVKVADFGLSRLMTGDTYTAHAGAKFPIKWTAPESLAYNKFSIKSDVWAFVGLLWEIATYGMSPYPGIDLQSVYELLEKDYRMERPEGCEKVVYELMRACQWNPSPDRPSFAEIHQAFETMFQESSISDEVEKELGKQGVRGAVSTLLQAPLEPTKTRTSRRAAEHRDITDVP
KGQGESDPLDHEPAVSPLLPRKERGPPEGGLNEDERLLPKDKKTNLFSALIKKKKTAPTTPKRSSSFREMDGQPERRGAGEEGRDISNGALFTPLDADPAKSPKPSNGAGVPNGALRESGGSGFRSPHLWKKSSSTLTSSRLATGEEEGGGSSSKRFLRSCSASCVPHGAKDTEWRSVTLPRDLQSTGRQF
TFGGHKSEKPALPRKRAGENRSDQVTRGTVTTPPRLVKKNEEAADEVFKDIMESSPGSSPPNLTTPKPLRRQVTVAPASGLPHKEEAGKGSALGTPAAAEPTVPTSKAGSGAPGGTSKGAPEESRVRHKKHSSSPGRDKGKLSRLKPAPPPPPAASAGKAGGKPSQSPSQAAGEAVLGAKTATSLVDVAVNSD
PSQPGEGLKKPVLPAATPKPQSAKPSGTPISAPVPSTLPSASSALAGDQPSSTAFIPLISTRVSLRKTRQPPERIASGAIITKGVLDSTALCLAISRNSEQMASHSAVLEAGKNLYTFCVSYVDSIQQMRNKFATREATNKLNNLRELQICPATAGSGPAATQDFSKLLSSVKEISDIVQR", "ABL1
(F317I)p":
"PFWKILNPLLERGTYYYFMGQPGKVLGDQRRPSPALHFIKAGGKKESSRHGGPHCNVFEHEALQRPVASFDFPQGLSEAAARNWSENKLAGPSENDPNLFVALYDFVASGDNTLSITKGEKLRVLGYNHNGWECEAQTNGQGWVPSNYITPVNSLEKHSWYHGPVSRNAEYLLSSGINGSFLVRESE
QRSISLRYEGRVYHYRINTASDGKLYVSSSRFNTLAELVHHHSTVADGLITTLHYPAKRNKPTVYGVSPPNYDKWEMERTDITMKHKLGGGQYGEVYEGVWKYSLTVAVKTLKEDTMEVEEFLKEAAMVKEIKHPNLVQLLGVCRTREPPFYITTEFMTYGNLLDYLRECNRQEVNAVVLVYMATQISSAMEY
KNFIHRDLAARNCLVGENHLVKVADFGLSRLMTGDTYTAHAGAKFPIKWTAPESLAYNKFSIKSDVWAFVGLLWEIATYGMSPYPGIDLQSVYELLEKDYRMERPEGCEKVVYELMRACQWNPSPDRPSFAEIHQAFETMFQESSISDEVEKELGKQGVRGAVSTLLQAPLEPTKTRTSRRAAEHRDITDVP
KGQGESDPLDHEPAVSPLLPRKERGPPEGGLNEDERLLPKDKKTNLFSALIKKKKTAPTTPKRSSSFREMDGQPERRGAGEEGRDISNGALFTPLDADPAKSPKPSNGAGVPNGALRESGGSGFRSPHLWKKSSSTLTSSRLATGEEEGGGSSSKRFLRSCSASCVPHGAKDTEWRSVTLPRDLQSTGRQF
TFGGHKSEKPALPRKRAGENRSDQVTRGTVTTPPRLVKKNEEAADEVFKDIMESSPGSSPPNLTTPKPLRRQVTVAPASGLPHKEEAGKGSALGTPAAAEPTVPTSKAGSGAPGGTSKGAPEESRVRHKKHSSSPGRDKGKLSRLKPAPPPPPAASAGKAGGKPSQSPSQAAGEAVLGAKTATSLVDVAVNSD
PSQPGEGLKKPVLPAATPKPQSAKPSGTPISAPVPSTLPSASSALAGDQPSSTAFIPLISTRVSLRKTRQPPERIASGAIITKGVLDSTALCLAISRNSEQMASHSAVLEAGKNLYTFCVSYVDSIQQMRNKFATREATNKLNNLRELQICPATAGSGPAATQDFSKLLSSVKEISDIVQR", "ABL1
(F317L)":
"PFWKILNPLLERGTYYYFMGQPGKVLGDQRRPSPALHFIKAGGKKESSRHGGPHCNVFEHEALQRPVASFDFPQGLSEAAARNWSENKLAGPSENDPNLFVALYDFVASGDNTLSITKGEKLRVLGYNHNGWECEAQTNGQGWVPSNYITPVNSLEKHSWYHGPVSRNAEYLLSSGINGSFLVRESE
QRSISLRYEGRVYHYRINTASDGKLYVSSSRFNTLAELVHHHSTVADGLITTLHYPAKRNKPTVYGVSPPNYDKWEMERTDITMKHKLGGGQYGEVYEGVWKYSLTVAVKTLKEDTMEVEEFLKEAAMVKEIKHPNLVQLLGVCRTREPPFYITTEFMTYGNLLDYLRECNRQEVNAVVLVYMATQISSAMEY
KNFIHRDLAARNCLVGENHLVKVADFGLSRLMTGDTYTAHAGAKFPIKWTAPESLAYNKFSIKSDVWAFVGLLWEIATYGMSPYPGIDLQSVYELLEKDYRMERPEGCEKVVYELMRACQWNPSPDRPSFAEIHQAFETMFQESSISDEVEKELGKQGVRGAVSTLLQAPLEPTKTRTSRRAAEHRDITDVP
KGQGESDPLDHEPAVSPLLPRKERGPPEGGLNEDERLLPKDKKTNLFSALIKKKKTAPTTPKRSSSFREMDGQPERRGAGEEGRDISNGALFTPLDADPAKSPKPSNGAGVPNGALRESGGSGFRSPHLWKKSSSTLTSSRLATGEEEGGGSSSKRFLRSCSASCVPHGAKDTEWRSVTLPRDLQSTGRQF
TFGGHKSEKPALPRKRAGENRSDQVTRGTVTTPPRLVKKNEEAADEVFKDIMESSPGSSPPNLTTPKPLRRQVTVAPASGLPHKEEAGKGSALGTPAAAEPTVPTSKAGSGAPGGTSKGAPEESRVRHKKHSSSPGRDKGKLSRLKPAPPPPPAASAGKAGGKPSQSPSQAAGEAVLGAKTATSLVDVAVNSD
PSQPGEGLKKPVLPAATPKPQSAKPSGTPISAPVPSTLPSASSALAGDQPSSTAFIPLISTRVSLRKTRQPPERIASGAIITKGVLDSTALCLAISRNSEQMASHSAVLEAGKNLYTFCVSYVDSIQQMRNKFATREATNKLNNLRELQICPATAGSGPAATQDFSKLLSSVKEISDIVQR", "ABL1
(F317L)p":
"PFWKILNPLLERGTYYYFMGQPGKVLGDQRRPSPALHFIKAGGKKESSRHGGPHCNVFEHEALQRPVASFDFPQGLSEAAARNWSENKLAGPSENDPNLFVALYDFVASGDNTLSITKGEKLRVLGYNHNGWECEAQTNGQGWVPSNYITPVNSLEKHSWYHGPVSRNAEYLLSSGINGSFLVRESE
QRSISLRYEGRVYHYRINTASDGKLYVSSSRFNTLAELVHHHSTVADGLITTLHYPAKRNKPTVYGVSPPNYDKWEMERTDITMKHKLGGGQYGEVYEGVWKYSLTVAVKTLKEDTMEVEEFLKEAAMVKEIKHPNLVQLLGVCRTREPPFYITTEFMTYGNLLDYLRECNRQEVNAVVLVYMATQISSAMEY
KNFIHRDLAARNCLVGENHLVKVADFGLSRLMTGDTYTAHAGAKFPIKWTAPESLAYNKFSIKSDVWAFVGLLWEIATYGMSPYPGIDLQSVYELLEKDYRMERPEGCEKVVYELMRACQWNPSPDRPSFAEIHQAFETMFQESSISDEVEKELGKQGVRGAVSTLLQAPLEPTKTRTSRRAAEHRDITDVP
KGQGESDPLDHEPAVSPLLPRKERGPPEGGLNEDERLLPKDKKTNLFSALIKKKKTAPTTPKRSSSFREMDGQPERRGAGEEGRDISNGALFTPLDADPAKSPKPSNGAGVPNGALRESGGSGFRSPHLWKKSSSTLTSSRLATGEEEGGGSSSKRFLRSCSASCVPHGAKDTEWRSVTLPRDLQSTGRQF
TFGGHKSEKPALPRKRAGENRSDQVTRGTVTTPPRLVKKNEEAADEVFKDIMESSPGSSPPNLTTPKPLRRQVTVAPASGLPHKEEAGKGSALGTPAAAEPTVPTSKAGSGAPGGTSKGAPEESRVRHKKHSSSPGRDKGKLSRLKPAPPPPPAASAGKAGGKPSQSPSQAAGEAVLGAKTATSLVDVAVNSD
PSQPGEGLKKPVLPAATPKPQSAKPSGTPISAPVPSTLPSASSALAGDQPSSTAFIPLISTRVSLRKTRQPPERIASGAIITKGVLDSTALCLAISRNSEQMASHSAVLEAGKNLYTFCVSYVDSIQQMRNKFATREATNKLNNLRELQICPATAGSGPAATQDFSKLLSSVKEISDIVQR", "ABL1
(F317L)p":
"PFWKILNPLLERGTYYYFMGQPGKVLGDQRRPSPALHFIKAGGKKESSRHGGPHCNVFEHEALQRPVASFDFPQGLSEAAARNWSENKLAGPSENDPNLFVALYDFVASGDNTLSITKGEKLRVLGYNHNGWECEAQTNGQGWVPSNYITPVNSLEKHSWYHGPVSRNAEYLLSSGINGSFLVRESE
QRSISLRYEGRVYHYRINTASDGKLYVSSSRFNTLAELVHHHSTVADGLITTLHYPAKRNKPTVYGVSPPNYDKWEMERTDITMKHKLGGGQYGEVYEGVWKYSLTVAVKTLKEDTMEVEEFLKEAAMVKEIKHPNLVQLLGVCRTREPPFYITTEFMTYGNLLDYLRECNRQEVNAVVLVYMATQISSAMEY
```

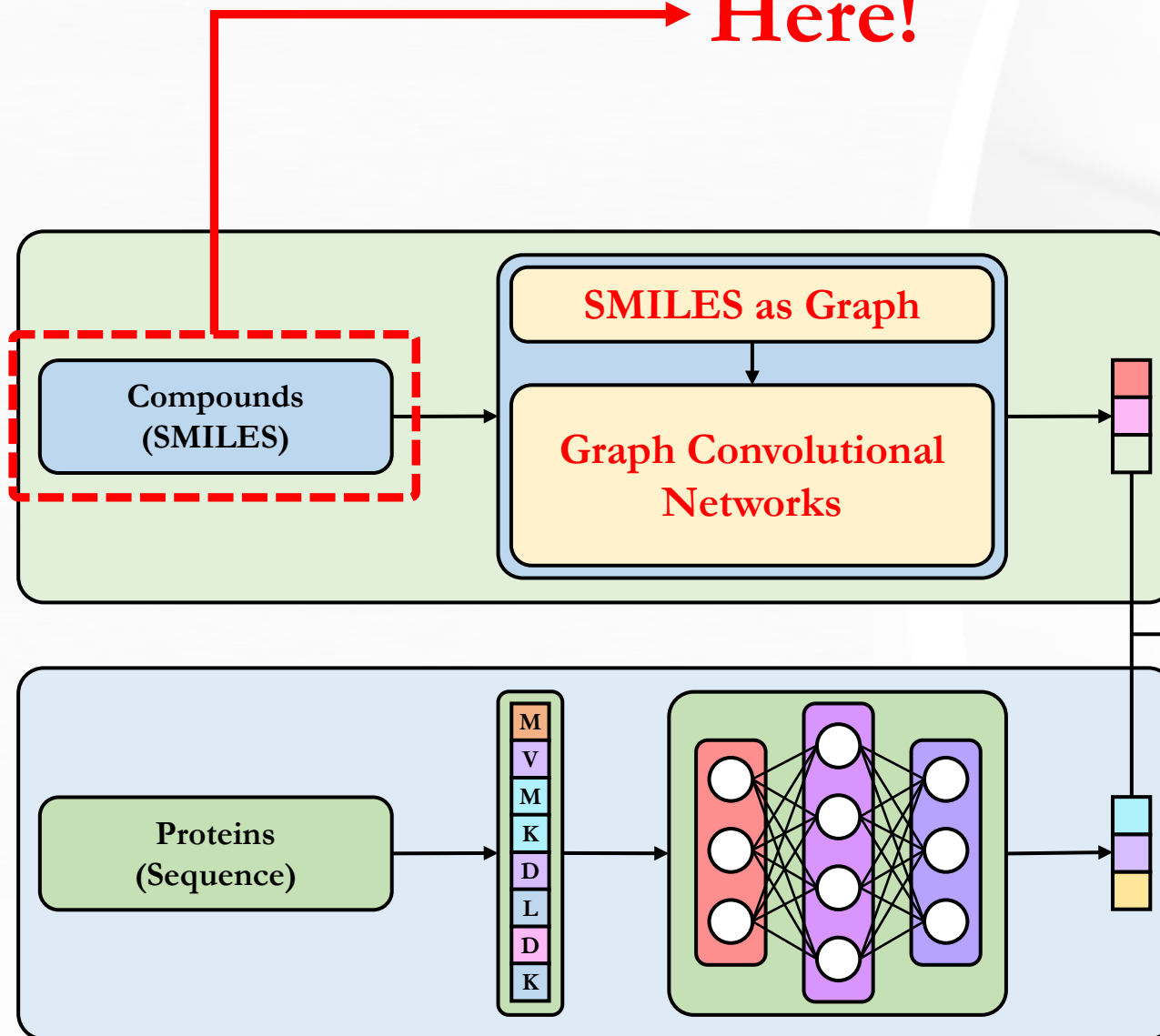

- ## AAK1 :

[illegible][illegible]

Molecule Representation

- SMILES -

Here!



#	Compound	Protein	Prediction
1	Chemical_ID	Protein_ID	0.99
2
3
4
...



Molecule Representation

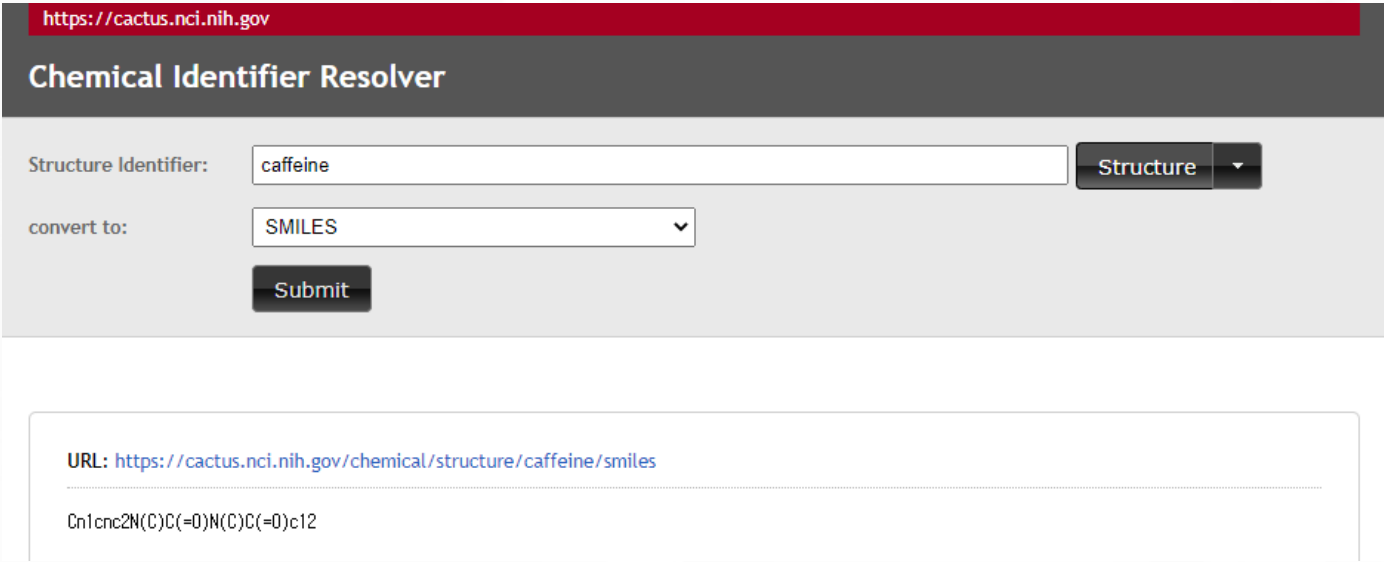
- 개요

- 화학 물질의 구조를 표현하는 방법으로 문자열, 그래프 등 다양한 형태가 존재
- 문자열로 표현하는 방법에서 대표적으로 WLN, ROSDAL, SMILES가 존재
- Davis & Kiba Datasets에서는 **SMILES를 통해 화학 물질을 표현**

Molecule Representation

- SMILES

- Simplified **M**olecular **I**nput **L**ine **E**ntry **S**ystem의 약자
- 화학물질의 구조를 문자열로 나타내는 방법
- 구조를 문자열로 변환할 때, 6가지 규칙 적용



https://cactus.nci.nih.gov

Chemical Identifier Resolver

Structure Identifier: Structure ▾

convert to: SMILES ▾

URL: <https://cactus.nci.nih.gov/chemical/structure/caffeine/smiles>

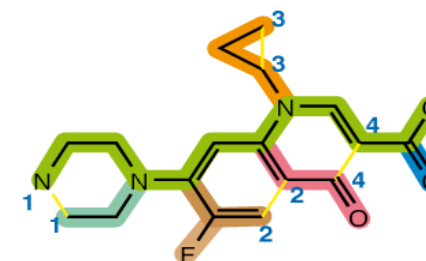
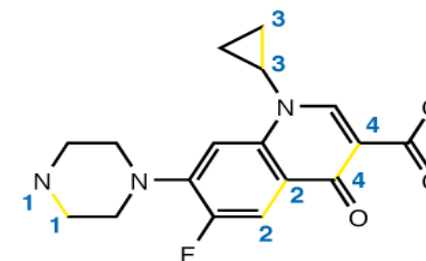
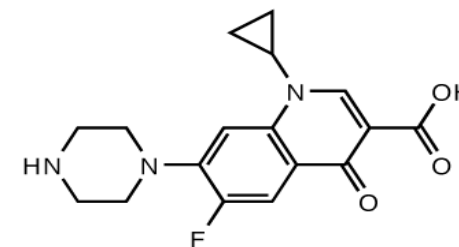
Cn1cnc2N(C)C(=O)N(C)C(=O)c12

<https://cactus.nci.nih.gov/chemical/structure>

Molecule Representation

• 규칙

- 원자는 표준 원소 기호로 표기
- 수소 원자는 가능한 모든 곳에 연결되어 있다고 가정, 표기 생략
- 이웃한 원자는 인접해서 표기
- 2중 결합은 '=', 3중 결합은 '#'으로 표기
- 가지는 '()'로 표기
- 고리는 고리를 생성하는 두 원자에 숫자를 표기
(방향족 고리는 원자를 소문자로 표기)



N1CCN(CC1)C(C(F)=C2)=CC(=C2C4=O)N(C3CC3)C=C4C(=O)O

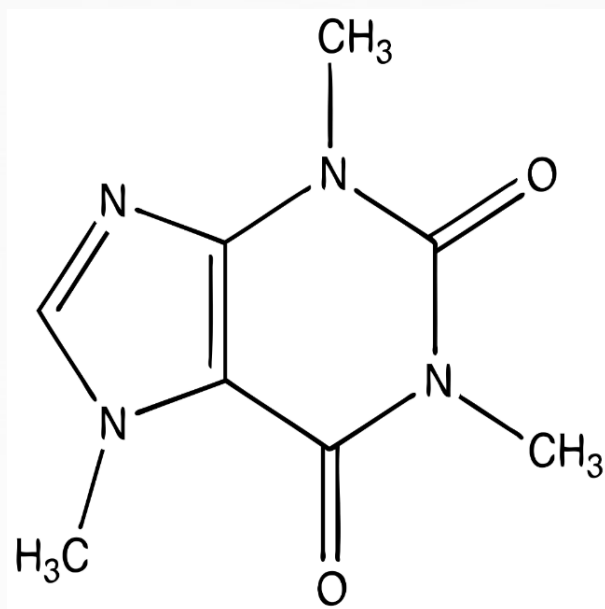
Molecule Representation

• 특성

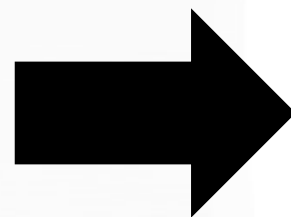
- 원자(Atoms)
- 결합(Bonds)
- 고리(Rings)
- 방향족성(Aromaticity)
- 분기(Branching)
- 입체배열(Stereochemistry)
- 동위원소(Isotopes)

Molecule Representation

- 예시



Caffeine - $\text{C}_8\text{H}_{10}\text{N}_4\text{O}_2$
[Molecule Structure]



Cn1cnc2N(C)C(=O)N(C)C(=O)c12
[SMILES]

Molecule Representation

• 예시 - Real Data

Chemical_ID SMILES_format

OC4CCOC4, "5287969": "CN1CCC(C(C1)O)C2=C(C=C(C3=C2OC(=CC3=O)C4=CC=CC=C4Cl)O)O", "6450551": "CNC(=O)C1=CC=CC=C1SC2=CC3=C(C=C2)C(=NN3)C=CC4=CC=CC=N4", "11364421": "CCC1C(=O)N(C2=CN=C(N=C2N1C3CCCCN4=C(C=C(C=C4)C(=O)NC5CCN(CC5)C)OC)C", "9926054": "CC1=CC2=C(C=C1)N=C(C3=NC=C(N23)C)NCCN.Cl", "16007391": "CCN(CCCOC1=CC2=C(C=C1)C(=NC=N2)NC3=NNC(=C3)CC(=O)NC4=CC(=CC=C4)F)CCO", "5328940": "CN1CCN(CC1)CCOC2=C(C=C3C(=C2)N=CC(=C3NC4=CC(=C(C=C4)Cl)OC)C#N)OC", "11234052": "CC1=CC2=C(N1)C=CC(=C2F)OC3=NC=NN4C3=C(C(=C4)OCC(C)O)C", "11656518": "CN1C2=C(C=C(C=C2)OC3=CC(=NC=C3)C4=NC=C(N4(F)F)N=C1NC5=CC=C(C=C5)C(F)F", "6918454": "C1CC1C0NC(=O)C2=C(C(=C(C=C2)F)F)NC3=C(C=C(C=C3)I)Cl", "156414": "C=CC(=O)NC1=C(C=C2C(=C1)C(=NC=N2)NC3=CC(=C(C=C3)F)Cl)OCCN4CCOCC4", "9933475": "CC1=CC2=C(N1)C=CC(=C2F)OC3=NC=NC4=CC(=C(C=C43)OC)OCCN5CCCC5", "11626560": "CC(C1=C(C=CC(=C1Cl)F)Cl)OC2=C(N=CC(=C2)C3=CN(N=C3)C4CCNCC4)N", "3062316": "CC1=C(C(=CC=C1)Cl)NC(=O)C2=CN=C(S2)NC3=NC(=C3)N4CCN(CC4)CCO)C", "156422": "CC1=CC=C(C=C1)N2C(=CC(=N2)C(C)(C)C)NC(=O)NC3=CC=C(C4=CC=CC=C43)OCCN5CCOCC5", "44150621": "CC(C(=O)O)O.CN1CCN(CC1)C2=CC3=C(C=C2)NC(=C4C(=C5C(=NC4=O)C=CC=C5F)N)N3.O", "176167": "CN1C=C(C2=CC=CC=C21)C3=C(C(=O)NC3=O)C4=CN(C5=CC=CC=C54)C6CCN(CC6)CC7=CC=CC=N7", "176870": "COCOC1=C(C=C2C(=C1)C(=NC=N2)NC3=CC=CC(=C3)C#C)OCCOC", "42642645": "COC1=CC2=C(C=CN=C2C=C1OCCN3CCOCC3)OC4=C(C=C(C=C4)NC(=O)C5(CC5)C(=O)NC6=CC=C(C=C6)F)F", "11717001": "C1CC(=NO)C2=C1C=C(C=C2)C3=CN(N=C3C4=CC=NC=C4)CCO", "16725726": "CCN1C2=C(C(=NC=C2OCC3CCCN3)C#CC(C)(C)O)N=C1C4=NON=C4N", "11617559": "COC1=CC=C(C=C1)COC2=C(C=C(C=C2)CC3=CN=C(N=C3N)N)OC", "123631": "COC1=C(C=C2C(=C1)N=CN=C2NC3=CC(=C(C=C3)F)Cl)OCCN4CCOCC4", "5291": "CC1=C(C=C(C=C1)NC(=O)C2=CC=C(C=C2)CN3CCN(CC3)C)NC4=NC=CC(=N4)C5=CN=CC=C5", "4908365": "CN1CCN(CC1)C(=O)C2=CC3=C(N2)C=CC(=C3)Cl", "11427553": "C1CN(CCN1)C(=O)C2=CC=C(C=C2)C=CC3=NNC4=CC=CC=C43", "208908": "CS(=O)(=O)CCNCC1=CC=C(O1C2=CC3=C(C=C2)N=CN=C3NC4=CC(=C(C=C4)OCC5=CC(=CC=C5)F)Cl", "126565": "CC12C(CC(O1)N3C4=CC=CC=C4C5=C6C(=C7C8=CC=CC=C8N2C7=C53)CNC6=O)(CO)O", "11485656": "CC1=CC(=C(C=C1)F)NC(=O)NC2=CC=C(C=C2)C3=C4(=CC=C3)NN=C4N", "9929127": "CC1=C(C=CC=N1)C(=O)NC2=C3C(=CC(=C2O)Cl)C4=C(N3)C=NC=C4", "11712649": "C1C2=CN=C(N=C2C3=C(C=C(C=C3)Cl)C(=N1)C4=C(C=CC=C4F)F)NC5=CC=C(C=C5)C(=O)O", "10074640": "CC1=C(C=C(C=C1)NC(=O)C2=CC=C(C=C2)CN3CCN(CC3)C)NC4=NC(=CS4)C5=CN=CC=C5", "51004351": "CC12C(C(CC(O1)N3C4=CC=CC=C4C5=C6C(=C7C8=CC=CC=C8N2C7=C53)CNC6=O)N(C)C(=O)C9=CC=CC=C9)OC", "11667893": "CC1(CNC2=C1C=CC(=C2)NC(=O)C3=C(N=CC=C3)NCC4=CC=NC=C4)C", "9915743": "CCOC1=C(C=C2C(=C1)N=CC(=C2NC3=CC(=C(C=C3)OCC4=CC=CC=N4)Cl)C#N)NC(=O)C=CCN(C)C", "644241": "CC1=C(C=C(C=C1)C(=O)NC2=CC(=CC(=C(N3C=C(N=C3)C)C(F)F)NC4=NC=CC(=N4)C5=CN=CC=C5", "447077": "CN1C2=NC(=NC=C2C=C(C1=O)C3=C(C=C(C=C3)Cl)Cl)NC4=CC(=CC=C4)SC", "10461815": "CC1=C(NC(=C1C(=O)N2CCCC2CN3CCCC3)C)C=C4C5=C(C=CC(=C5)S(=O)(CCC6=C(C=CC=C6Cl)Cl)NC4=O", "9884685": "C1OCCN1C2=NC(=NC3=C2OC4=C3C=CC(=N4)C5=CC(=CC=C5)O", "24180719": "CCCS(=O)(=O)NC1=C(C(=C(C=C1)F)C(=O)C2=CN3C=NC=C(C=C23)Cl)F", "25243800": "CC(C)N1C2=C(C(=C3C=C4C(C=C4C=N3)O)N1)C(=NC=N2)N", "10113978": "CC1=C(C=C(C=C1)NC2=NC=CC(=N2)N(C)C3=CC4=NN(C(=C4C=C3)C)S(=O)(=O)N", "17755052": "CS(=O)(=O)N1CCN(CC1)CC2=CC3=C(S2)C(=NC(=N3)C4=C5C=NNC5=CC(=N6CCOCC6", "11984591": "C1(C(=O)NC2=C(O1)C=CC(=N2)NC3=NC(=NC=C3F)NC4=CC(=C(C=C4)OC)OC)C.C1=CC=C(C=C1)S(=O)(=O)O", "153999": "CN(C)CC1CCN2=C(C3=CC=CC=C32)C4=C(C5=CN(CC(O1)C6=CC=CC=C65)C(=O)NC4=O", "25127112": "C1CCC(C1)C(CC#N)N2C=C(C=N2)C3=C4C=CNC4=NC=N3.OP(=O)(O)O", "176155": "CS(=O)C1=CC=C(C=C1)C2=NC(=C(N2)C3=CC=NC=C3)C4=CC=C(C=C4)F", "24779724": "CN1C=C(C=N1)C2=NN3C(=NN=C3SC4=CC5=C(C=C4)N=CC=C5)C=C2", "3025986": "CC(C)(C)C1=CN=C(O1)CSC2=CN=C(S2)NC(=O)C3CCNCC3", "10138260": "CC1=C(NC(=C1C(=O)NCC(CN2CCOCC2)O)C)C=C3C4=C(C=CC(=C4)F)NC3=O", "10127622": "CN1C=NC2=C1C=C(C(=C2F)NC3=C(C(=C(C=C3)Br)Cl)C(=O)NOCOC", "216239": "CNC(=O)C1=NC=CC(=C1)OC2=CC=C(C=C2)NC(=O)NC3=CC(=C(C=C3)Cl)C(F)F", "44259": "CC12C(C(CC(O1)N3C4=CC=CC=C4C5=C6C(=C7C8=CC=CC=C8N2C7=C53)CNC6=O)NC)OC", "5329102": "CCN(CC)CCNC(=O)C1=C(NC(=C1C)C=C2C3=C(C=CC(=C3)F)NC2=O)C", "16038120": "CC(C)S(=O)(=O)C1=CC=CC=C1NC2=NC(=NC=C2Cl)NC3=C(C=C(C=C3)N4CCC(CC4)N5CCN(CC5)C)OC", "10427712": "C1=CC(=CC(=C1)O)C2=NC3=C(N=C2C4=CC(=CC=C4)O)N=C(N=C3N)N", "16722836": "CC1=CN=C(N=C1NC2=CC(=CC=C2)S(=O)(=O)NC(C)(C)C)NC3=CC=C(C=C3)OCCN4CCCC4", "3038522": "CC(C)OC1=CC(=C(C1)NC(=O)N2CCN(CC2)C3=NC=NC4=CC(=C(C=C43)OC)OCCN5CCCCC5", "9926791": "CC1CCN(CC1N(C)C2=NC=NC3=C2C=CN3)C(=O)CC#N", "5494449": "CC1=CC(=NN1)NC2=NC(=NC(=C2)N3CCN(CC3)C)SC4=CC=C(C=C4)NC(=O)C5CC5"

Molecule Representation

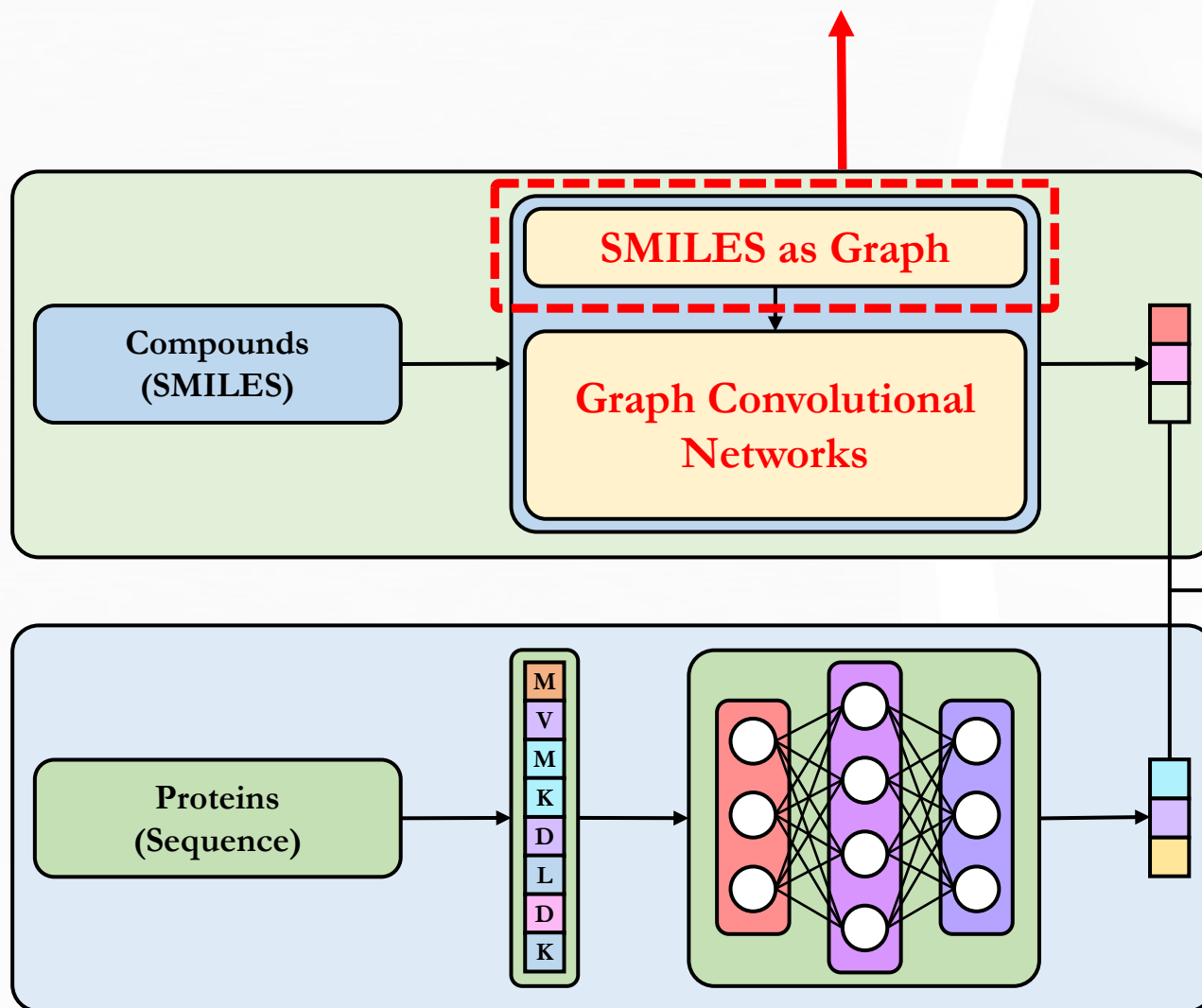
- 한계점

- 문자열로 이루어진 SMILES 특성 상, 복잡한 화학 구조를 제대로 반영하지 못함
- 화학 구조를 제대로 반영하지 못했다는 것은 **정보 유실**을 의미
- 방대한 화합물 정보를 제대로 이용하지 못함
- **신약후보물질 예측의 한계가 존재**

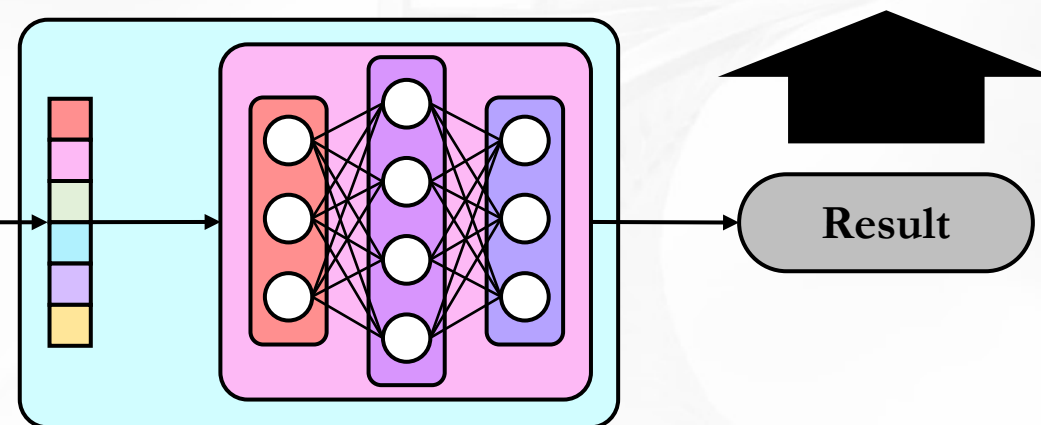
Molecule Representation

- Graph -

Here!



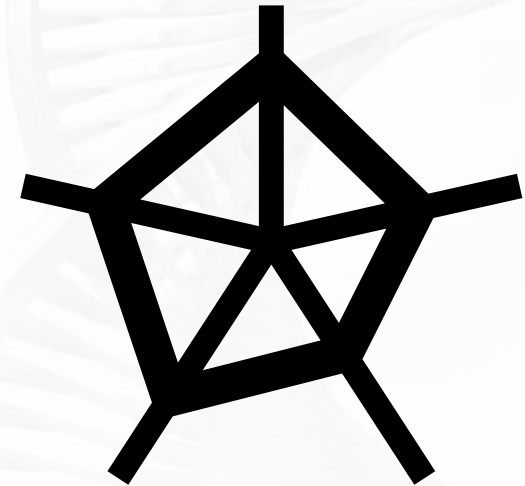
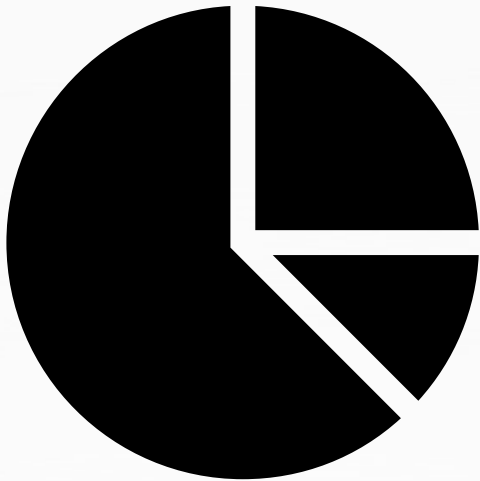
#	Compound	Protein	Prediction
1	Chemical_ID	Protein_ID	0.99
2
3
4
...



Graph

- Graph - Statistical Graph

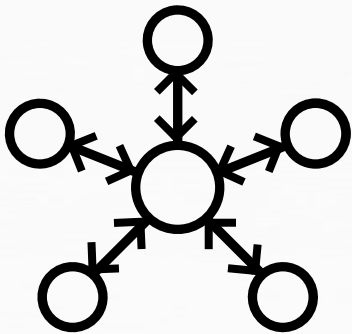
➤ 어떠한 데이터들을 그림상으로 시각화하여 나타낸 것을 의미



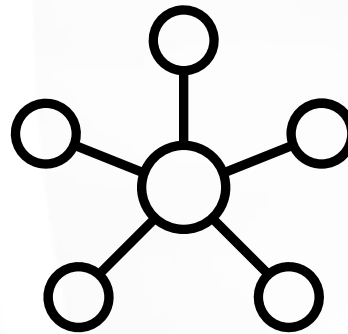
Graph

- Graph - Mathematical Graph

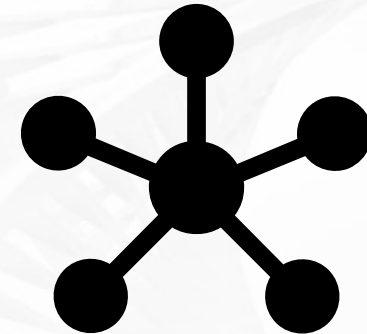
- 일부 객체들의 쌍들이 서로 연관된 객체의 집합을 이루는 구조
- 방향 / 無방향, 가중치 그래프로 구분



Directed graph



Undirected graph



Weighted graph

○ : Node

—
→ : Edge

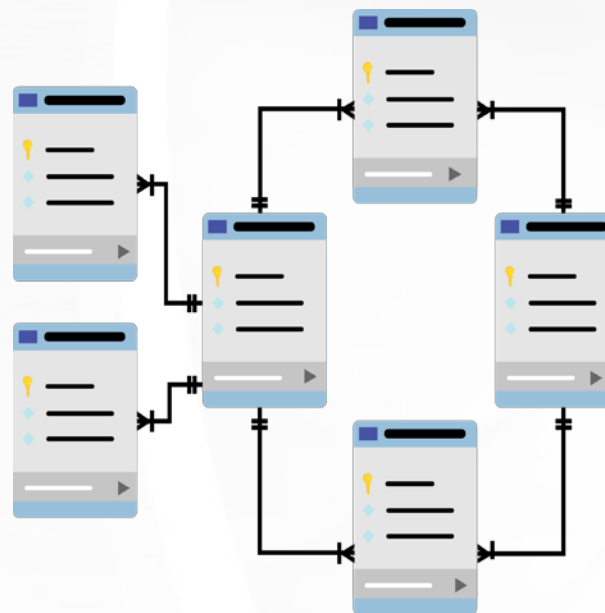
Graph

- Graph - Real Case

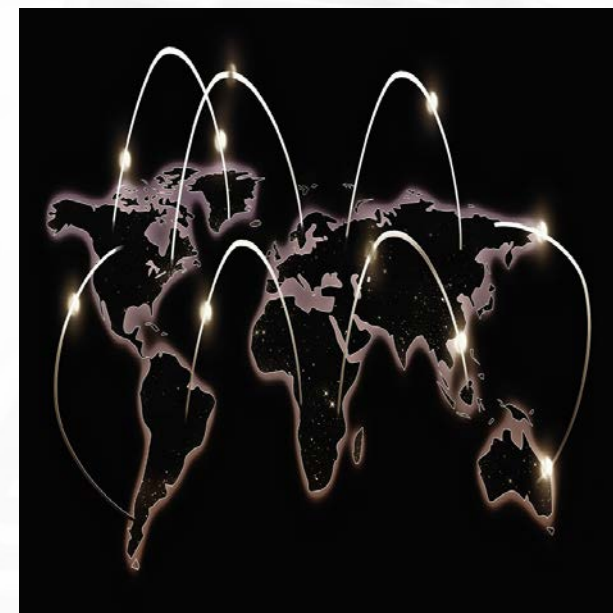
➤ 객체를 정점, 객체 사이의 관계를 간선으로 표현



사회 관계망



관계형 데이터베이스



지도

Graph

- Graph - Real Case

➤ 객체를 정점, 객체 사이의 관계를 간선으로 표현



인체 구조



지식

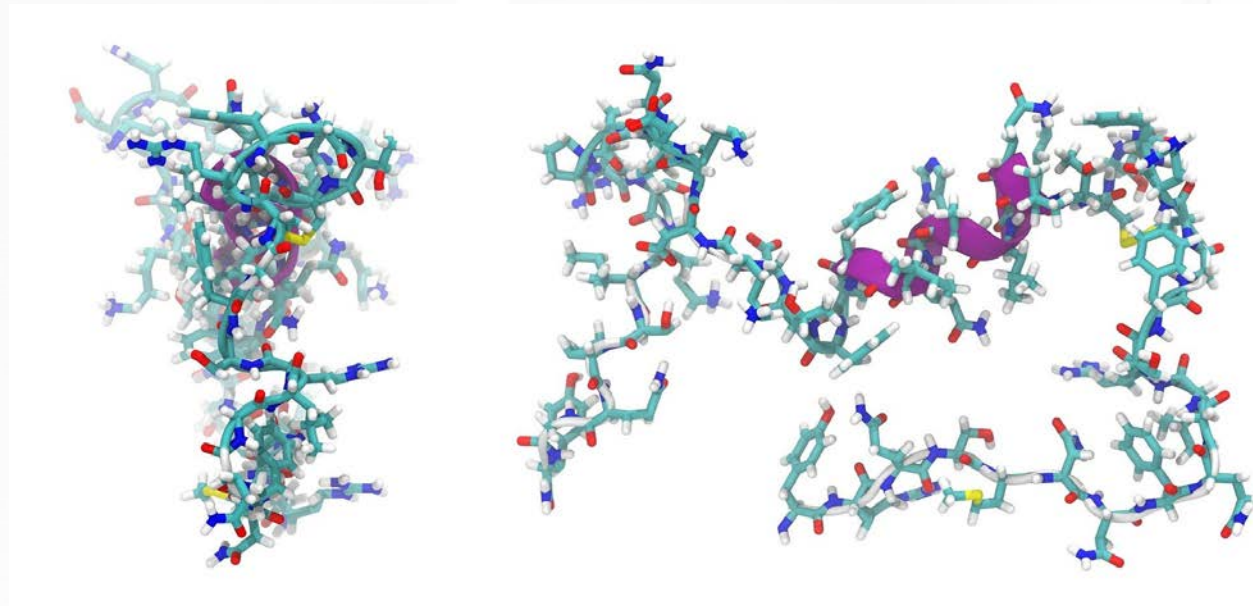


우주

Graph

- Graph - Real Case

- 객체를 정점, 객체 사이의 관계를 간선으로 표현

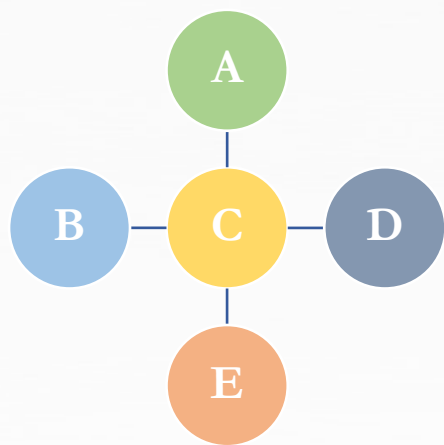


분자

Graph

• Graph - Graph Data Structure

- 정점(Vertex or Node)과 그 정점을 연결하는 간선(Edge)을 하나로 모아 놓은 비선형 자료 구조
- 정점 정보는 **특성 행렬**, 간선 정보는 **인접 행렬**로 표현



=

	A	B	C	D	E
A	0	0	1	0	0
B	0	0	1	0	0
C	1	1	0	1	1
D	0	0	1	0	0
E	0	0	1	0	0

인접 행렬

&

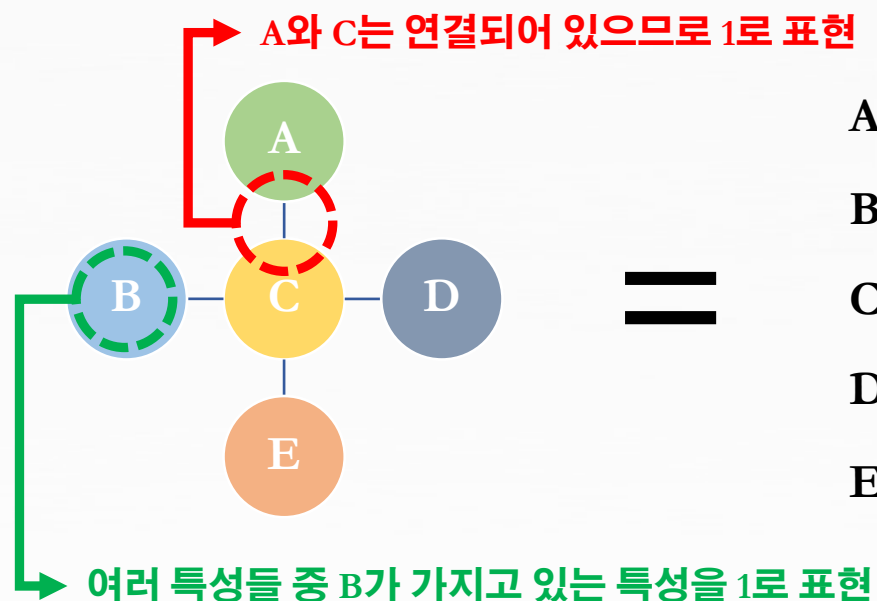
	1	2	3	4	5	6
A	0	1	0	1	0	1
B	1	1	1	0	0	1
C	1	0	0	0	1	0
D	1	1	1	0	0	1
E	0	0	1	1	0	0

특성 행렬

Graph

• Graph – Adjacency & Feature

- 인접 행렬 : 그래프에서 어느 정점들이 간선으로 연결되었는지 나타내는 정사각 행렬
- 특성 행렬 : 그래프에서 정점들이 가지고 있는 특성들을 나타내는 정사각 행렬



=

	A	B	C	D	E
A	0	0	1	0	0
B	0	0	1	0	0
C	1	1	0	1	1
D	0	0	1	0	0
E	0	0	1	0	0

인접 행렬

&

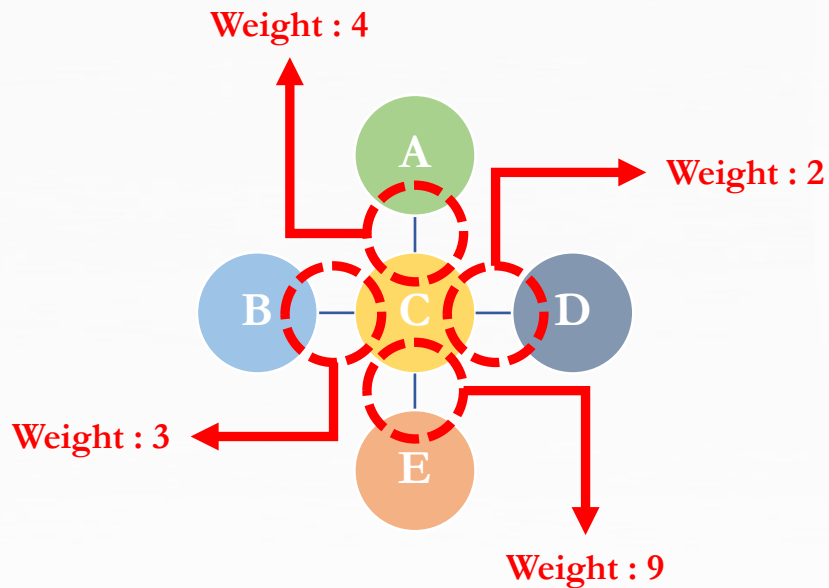
	1	2	3	4	5	6
A	0	1	0	1	0	1
B	1	1	1	0	0	1
C	1	0	0	0	1	0
D	1	1	1	0	0	1
E	0	0	1	1	0	0

특성 행렬

Graph

- Graph – Weight Matrix

- 가중치 행렬 : 인접 행렬을 이용하여 표현
- 간선이 존재하지 않는 경우, 가중치 범위 밖의 값 INF 등으로 표현



=

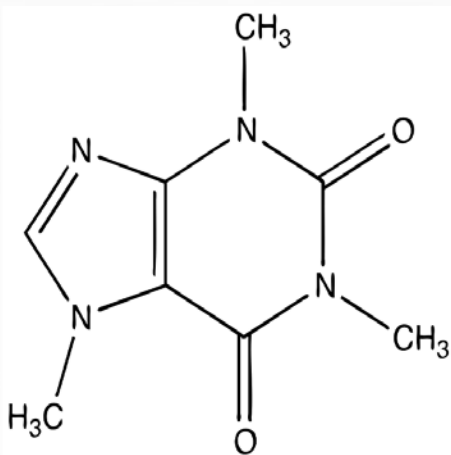
	A	B	C	D	E
A	INF	INF	4	INF	INF
B	INF	INF	3	INF	INF
C	4	3	INF	2	9
D	INF	INF	2	INF	INF
E	INF	INF	9	INF	INF

인접 행렬(가중치)

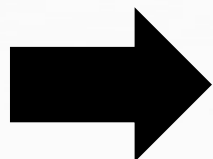
Graph – Molecule Representation

• SMILES as Graph

- **RDKit** 라이브러리를 활용
- 화학 물질의 정보를 담고 있는 데이터를 활용하여 구조 이미지(구조 식) 생성
- 문자열로 구성된 SMILES를 신경망 학습을 위한 **그래프** 형식으로 변환



Caffeine(C₈H₁₀N₄O₂)
[Molecule Structure]



Cn1cnc2N(C)C(=O)N(C)C(=O)c12
[SMILES]



0	0	1	0	0
0	0	1	0	0
1	1	0	1	1
0	0	1	0	0
0	0	1	0	0

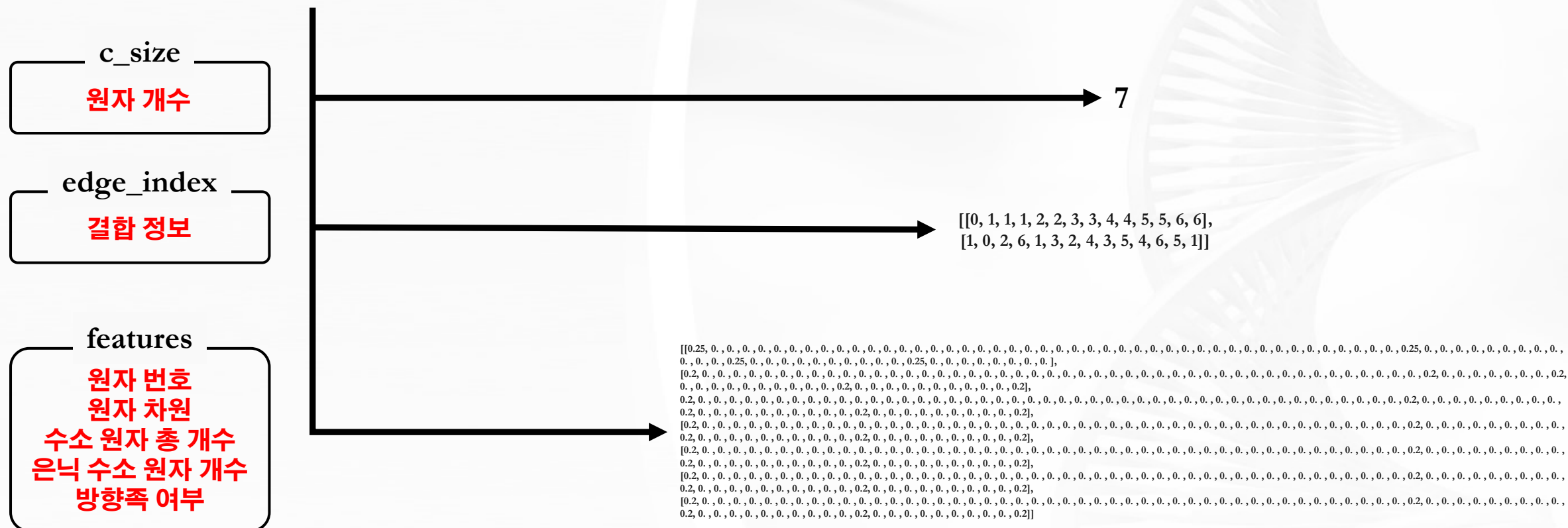
0	1	0	1	0
1	1	1	0	0
1	0	0	0	1
1	1	1	0	0
0	0	1	1	0

[Matrix]

Graph – Molecule Representation

- **SMILES to Input Presentation(Graph Representation)**

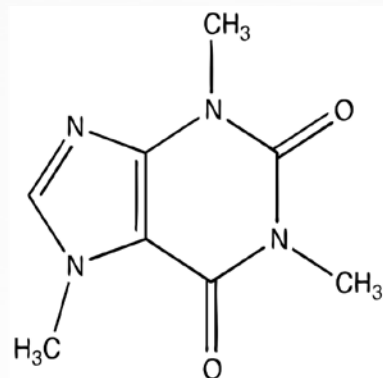
Toluene : Cc1ccccc1


$$\{\text{"Cc1ccccc1"} : c_size, edge_index, features\}$$

Graph

- Graph - Problem & Solution

- 분자처럼 원자의 속성, 연결의 종류 등을 고려해야하는 경우, 속성 그래프로 데이터를 표현해야 함
- 속성 그래프 데이터는 기존 방식으로 벡터의 형태로 변환하는 것이 불가능
- 벡터의 형태로 데이터를 입력 받는 기존의 인공지능망으로는 분자 그래프 데이터를 처리할 수 없음
- **Graph convolution**을 이용하여 **정점 또는 그래프 자체를 벡터 형태의 데이터로 변환하여 해결**



Graph Convolution



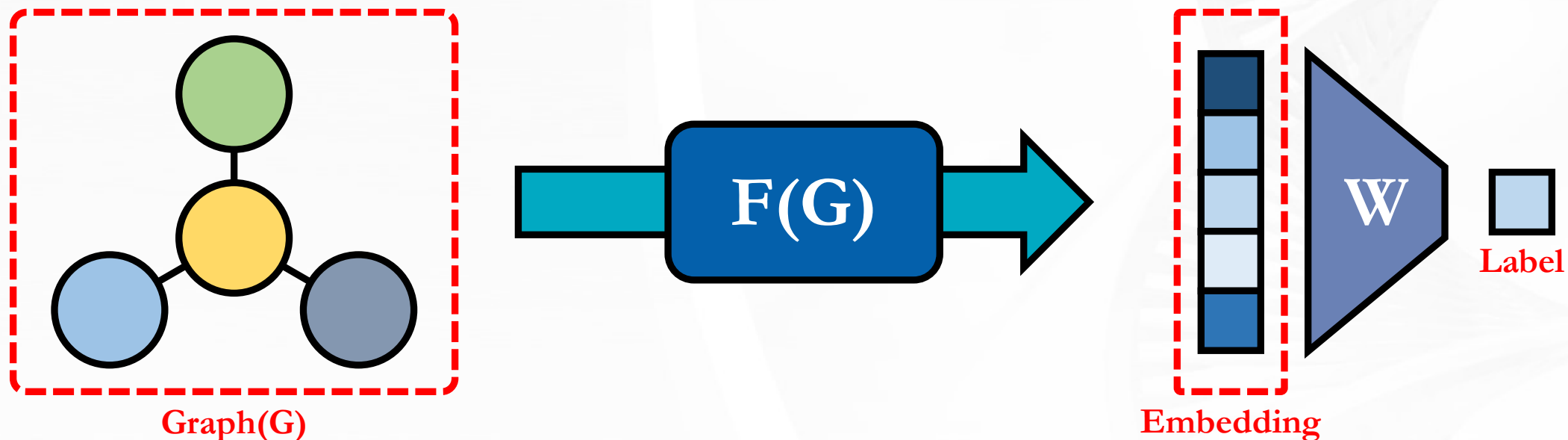
Vector

Graph Neural Networks

Graph Neural Network

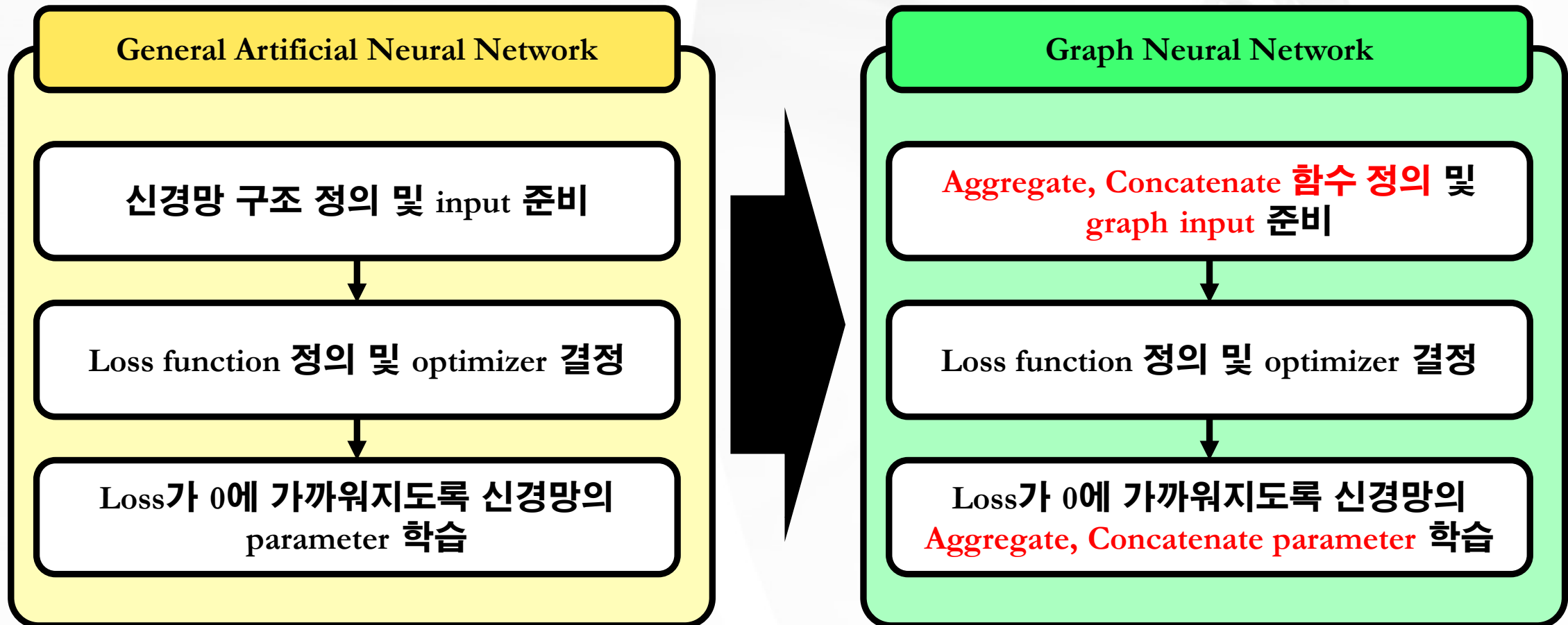
• 개요

- 그래프 구조에서 사용하는 인공 신경망
- 입력이 **그래프 구조**라는 특징을 지님
- 그래프에 존재하는 **정점 사이의 관계를 모델링**하고, 이에 대한 **표현을 생성**하는 것이 핵심



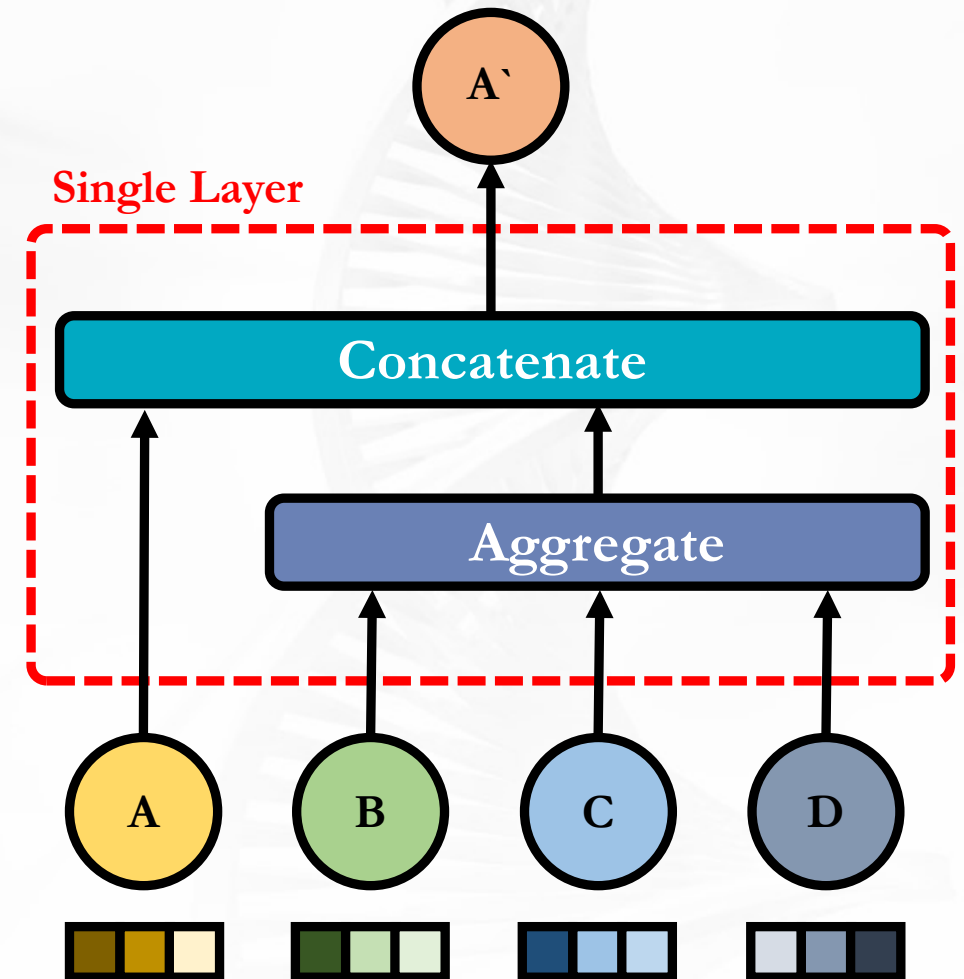
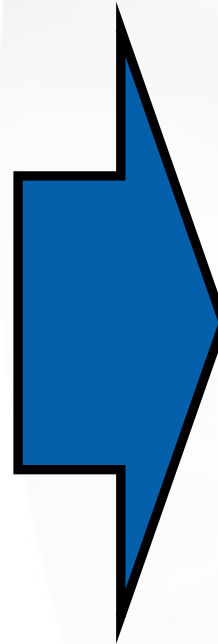
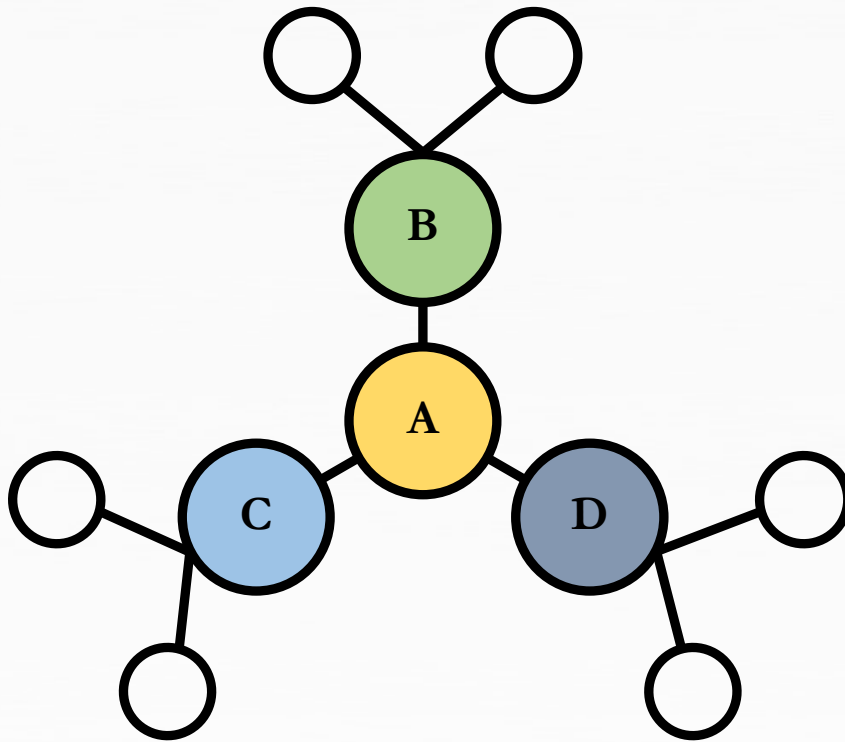
Graph Neural Network

- 학습 과정



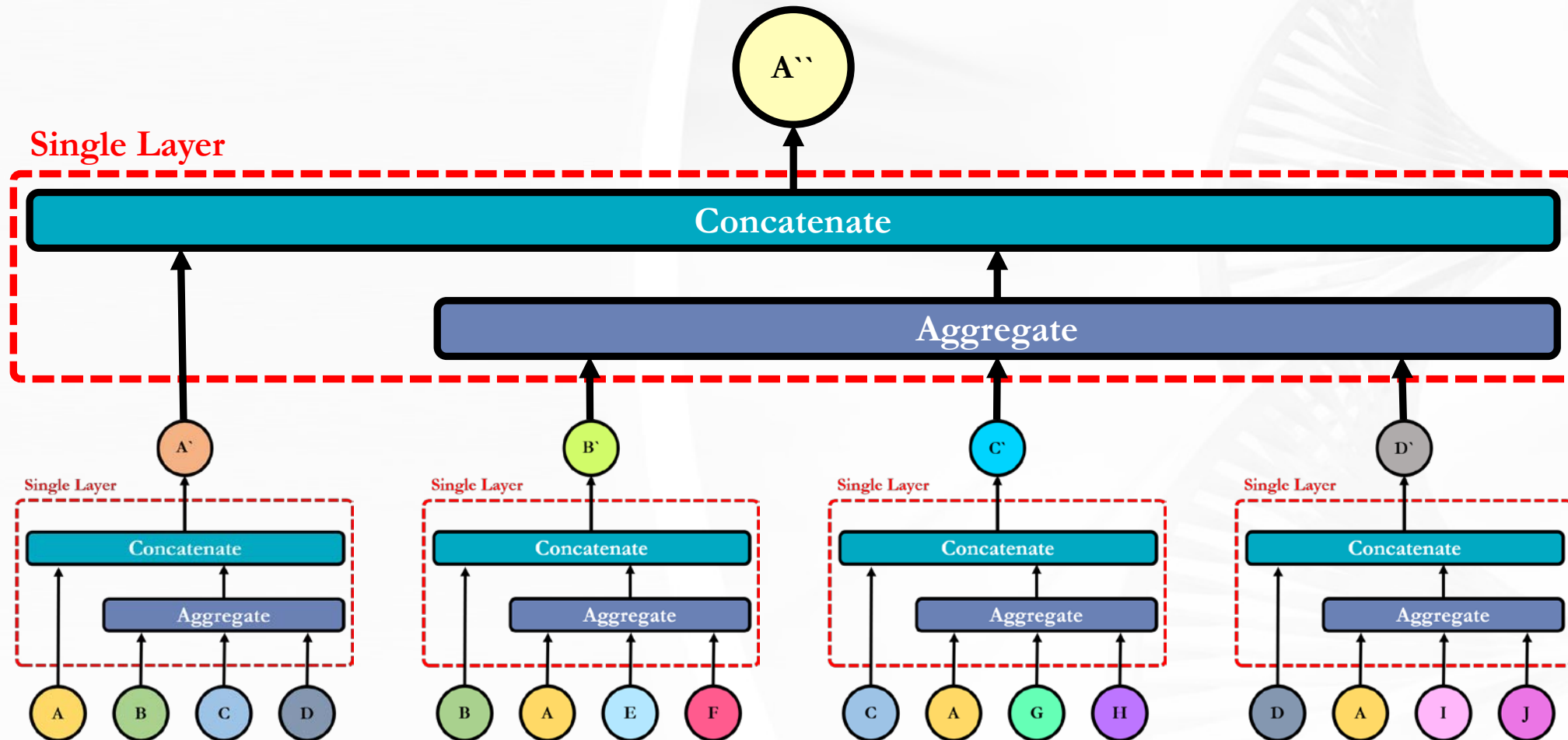
Graph Neural Network

- Neighborhoods Aggregation



Graph Neural Network

- Neighborhoods Aggregation



Graph Neural Network

- Algorithm

for v in V :

$h_{\{v\}}^{\{0\}} = v$'s feature vector

for i in range(1, $k+1$) :

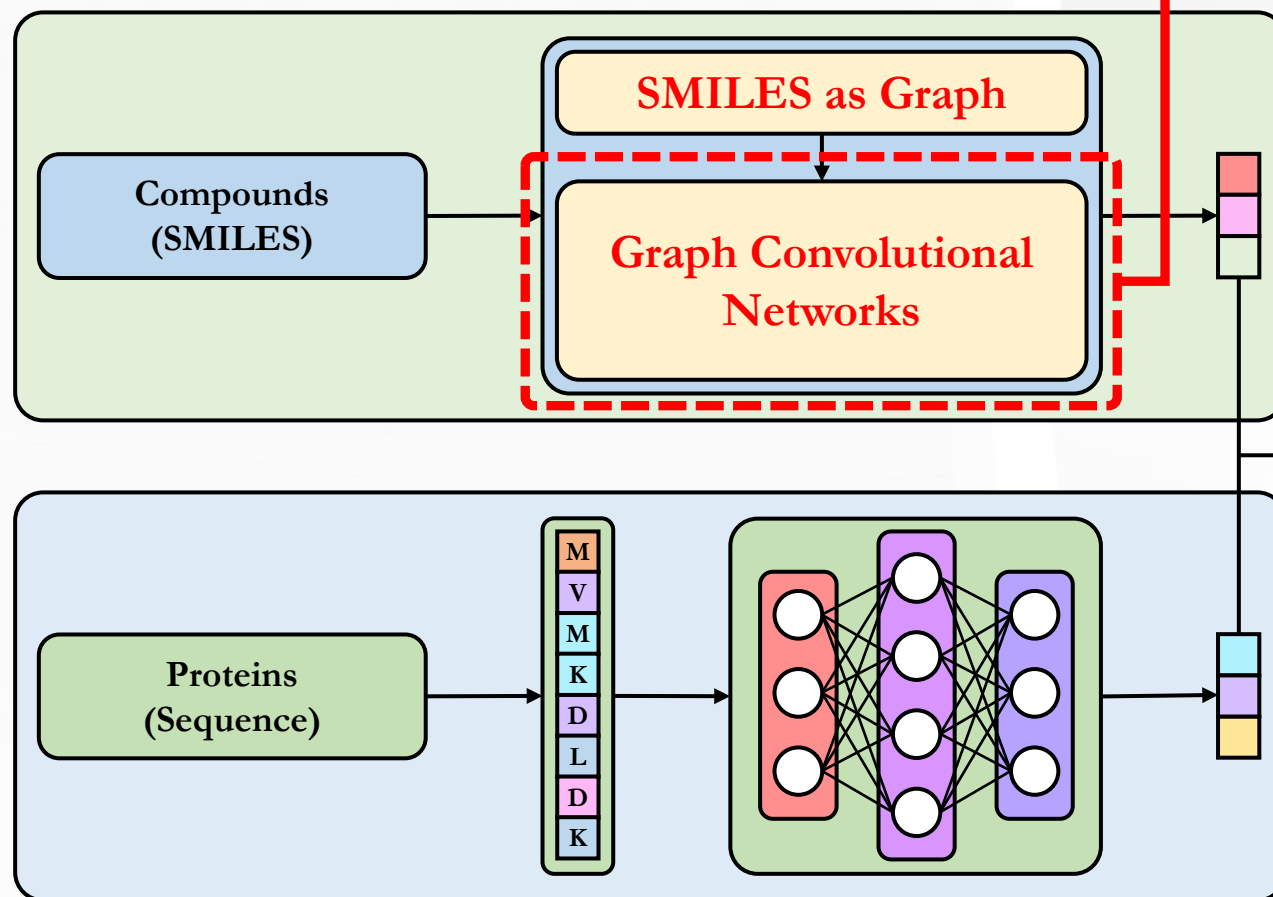
for v in V :

$a = \text{Aggregate}(\{h_{\{u\}}^{\{i-1\}} \mid \{u, v\} \in E\})$

$h_{\{v\}}^{\{i\}} = \text{Concatenate}(h_{\{v\}}^{\{i-1\}}, a)$

Graph Convolutional Networks

Here! ←

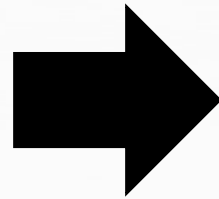
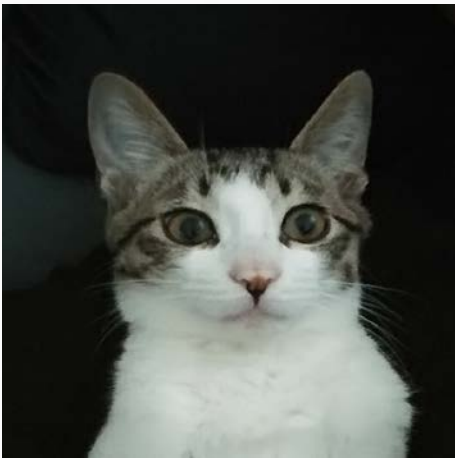


#	Compound	Protein	Prediction
1	Chemical_ID	Protein_ID	0.99
2
3
4
...

Convolution

• 개요

- 행렬 내에서 특성을 뽑기 위한 연산
- 입력 행렬과 특성 탐지를 통해 입력에 대한 특성 지도를 생성
- 특성 지도를 통해 입력 행렬의 패턴 파악



0	5	0	5	0
0	7	1	7	0
1	4	9	4	1
0	3	2	3	0
0	1	1	1	0

 $*$

1	3	0
0	2	1
0	2	3

 $=$

45	73	12
17	93	35
68	63	25

Convolution

• 합성곱 연산 과정(Matrix)

0	5	0	5	0
0	7	1	7	0
1	4	9	4	1
0	3	2	3	0
0	1	1	1	0

 \ast

1	3	0
0	2	1
0	2	3

 $=$

45		

0	5	0	5	0
0	7	1	7	0
1	4	9	4	1
0	3	2	3	0
0	1	1	1	0

 \ast

1	3	0
0	2	1
0	2	3

 $=$

45	73	

0	5	0	5	0
0	7	1	7	0
1	4	9	4	1
0	3	2	3	0
0	1	1	1	0

 \ast

1	3	0
0	2	1
0	2	3

 $=$

45	73	12

0	5	0	5	0
0	7	1	7	0
1	4	9	4	1
0	3	2	3	0
0	1	1	1	0

 \ast

1	3	0
0	2	1
0	2	3

 $=$

45	73	12
17		

Graph Convolution

- 특정 노드의 은닉 벡터를 해당 노드의 이웃 정보를 통해 표현
 - 그래프에 포함된 노드나 그래프 자체를 벡터 형태의 데이터로 변환
 - 노드의 특성과 연결 정보만을 고려하기 때문에 **간선 특성은 사용되지 않음**

은닉 노드
특성 행렬

인접 행렬

특성 행렬

$$H = \Psi(A, X) = \sigma(A X W)$$

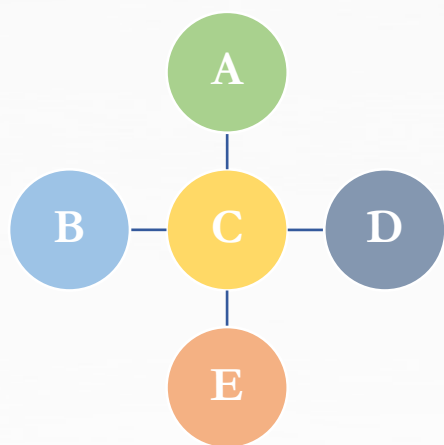
합성곱 연산

비선형 활성화 함수

가중치 행렬

Graph Convolution

• 행렬 계산 예시



=

$$G = (A, X)$$

	A	B	C	D	E
A	0	0	1	0	0
B	0	0	1	0	0
C	1	1	0	1	1
D	0	0	1	0	0
E	0	0	1	0	0

인접 행렬(A)

&

	1	2	3	4	5	6
A	0	1	0	1	0	1
B	1	1	1	0	0	1
C	1	0	0	0	1	0
D	1	1	1	0	0	1
E	0	0	1	1	0	0

특성 행렬(X)

Graph Convolution

• 행렬 계산 예시

$$\mathbf{A} \times \mathbf{X} = \mathbf{H}$$

	A	B	C	D	E
A	0	0	1	0	0
B	0	0	1	0	0
C	1	1	0	1	1
D	0	0	1	0	0
E	0	0	1	0	0

인접 행렬(A)

×

	1	2	3	4	5	6
A	0	1	0	1	0	1
B	1	1	1	0	0	1
C	1	0	0	0	1	0
D	1	1	1	0	0	1
E	0	0	1	1	0	0

특성 행렬(X)

=

	1	2	3	4	5	6
A	1	0	0	0	1	0
B	1	0	0	0	1	0
C	2	3	3	2	0	3
D	1	0	0	0	1	0
E	1	0	0	0	1	0

은닉 정점 특성 행렬(H)

Graph Convolution

• 문제점

- 인접 행렬에는 노드의 연결만 표현되어 있음
 - 합성곱 연산에서 각 노드 자체에 대한 **정보 유실**
- 인접 행렬은 정규화 되어 있지 않음
 - 특성 벡터와 곱 연산을 수행할 경우 **크기가 불안정**하게 변할 수 있음

인접 행렬의 한계

Graph Convolution

• 해결 방안 적용

- 인접 행렬에 **self-loop** 추가
- 인접 행렬을 $\mathbf{D}^{-1/2} \mathbf{A} \mathbf{D}^{-1/2}$ 로 정규화

Self-loop를 추가한 인접 행렬

$$\Psi(\tilde{\mathbf{A}}, \mathbf{X}) = \sigma(\tilde{\mathbf{D}}^{-1/2} \tilde{\mathbf{A}} \tilde{\mathbf{D}}^{-1/2} \mathbf{X} \mathbf{W})$$

$\tilde{\mathbf{A}}$ 의 대각 차수 행렬

$\tilde{\mathbf{A}}$ 의 정규화

Graph Convolution

- Self-loop 추가

Self-loop 추가

	A	B	C	D	E
A	1	0	1	0	0
B	0	1	1	0	0
C	1	1	1	1	1
D	0	0	1	1	0
E	0	0	1	0	1

인접 행렬(A)

×

Self-loop를 추가한 인접 행렬

$$\tilde{A} \times X = H$$

	1	2	3	4	5	6
A	0	1	0	1	0	1
B	1	1	1	0	0	1
C	1	0	0	0	1	0
D	1	1	1	0	0	1
E	0	0	1	1	0	0

특성 행렬(X)

=

	1	2	3	4	5	6
A	1	1	0	1	1	1
B	1	1	1	0	1	1
C	3	3	3	2	1	3
D	1	1	1	0	1	1
E	1	0	1	1	1	0

은닉 정점 특성 행렬(H)

Graph Convolution

• 인접 행렬 정규화

$$\tilde{\mathbf{D}}^{-1/2} \times \tilde{\mathbf{A}} \times \tilde{\mathbf{D}}^{-1/2}$$

	A	B	C	D	E
A	$\frac{1}{\sqrt{2}}$	0	0	0	0
B	0	$\frac{1}{\sqrt{2}}$	0	0	0
C	0	0	$\frac{1}{\sqrt{5}}$	0	0
D	0	0	0	$\frac{1}{\sqrt{2}}$	0
E	0	0	0	0	$\frac{1}{\sqrt{2}}$

차수 행렬($\tilde{\mathbf{D}}^{-1/2}$)

×

	A	B	C	D	E
A	1	0	1	0	0
B	0	1	1	0	0
C	1	1	1	1	1
D	0	0	1	1	0
E	0	0	1	0	1

인접 행렬($\tilde{\mathbf{A}}$)

×

	A	B	C	D	E
A	$\frac{1}{\sqrt{2}}$	0	0	0	0
B	0	$\frac{1}{\sqrt{2}}$	0	0	0
C	0	0	$\frac{1}{\sqrt{5}}$	0	0
D	0	0	0	$\frac{1}{\sqrt{2}}$	0
E	0	0	0	0	$\frac{1}{\sqrt{2}}$

차수 행렬($\tilde{\mathbf{D}}^{-1/2}$)

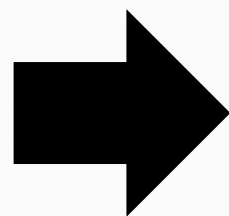
Graph Convolution

• 인접 행렬 정규화

$$\tilde{\mathbf{D}}^{-1/2} \times \tilde{\mathbf{A}} \times \tilde{\mathbf{D}}^{-1/2}$$

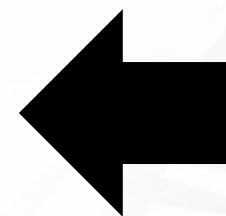
	A	B	C	D	E
A	$\frac{1}{\sqrt{2}}$	0	0	0	0
B	0	$\frac{1}{\sqrt{2}}$	0	0	0
C	0	0	$\frac{1}{\sqrt{5}}$	0	0
D	0	0	0	$\frac{1}{\sqrt{2}}$	0
E	0	0	0	0	$\frac{1}{\sqrt{2}}$

차수 행렬($\tilde{\mathbf{D}}^{-1/2}$)



	A	B	C	D	E
A	$\frac{1}{2}$	0	$\frac{1}{\sqrt{10}}$	0	0
B	0	$\frac{1}{2}$	$\frac{1}{\sqrt{10}}$	0	0
C	$\frac{1}{\sqrt{10}}$	$\frac{1}{\sqrt{10}}$	$\frac{1}{5}$	$\frac{1}{\sqrt{10}}$	$\frac{1}{\sqrt{10}}$
D	0	0	$\frac{1}{\sqrt{10}}$	$\frac{1}{2}$	0
E	0	0	$\frac{1}{\sqrt{10}}$	0	$\frac{1}{2}$

정규화 인접 행렬
($\tilde{\mathbf{D}}^{-1/2} \tilde{\mathbf{A}} \tilde{\mathbf{D}}^{-1/2}$)



	A	B	C	D	E
A	$\frac{1}{\sqrt{2}}$	0	0	0	0
B	0	$\frac{1}{\sqrt{2}}$	0	0	0
C	0	0	$\frac{1}{\sqrt{5}}$	0	0
D	0	0	0	$\frac{1}{\sqrt{2}}$	0
E	0	0	0	0	$\frac{1}{\sqrt{2}}$

차수 행렬($\tilde{\mathbf{D}}^{-1/2}$)

Graph Convolution

• 최종 계산

$$\tilde{\mathbf{D}}^{-1/2} \times \tilde{\mathbf{A}} \times \tilde{\mathbf{D}}^{-1/2} \times \mathbf{X} = \mathbf{H}$$

	A	B	C	D	E
A	$\frac{1}{2}$	0	$\frac{1}{\sqrt{10}}$	0	0
B	0	$\frac{1}{2}$	$\frac{1}{\sqrt{10}}$	0	0
C	$\frac{1}{\sqrt{10}}$	$\frac{1}{\sqrt{10}}$	$\frac{1}{5}$	$\frac{1}{\sqrt{10}}$	$\frac{1}{\sqrt{10}}$
D	0	0	$\frac{1}{\sqrt{10}}$	$\frac{1}{2}$	0
E	0	0	$\frac{1}{\sqrt{10}}$	0	$\frac{1}{2}$

정규화 인접 행렬

×

	1	2	3	4	5	6
A	0	1	0	1	0	1
B	1	1	1	0	0	1
C	1	0	0	0	1	0
D	1	1	1	0	0	1
E	0	0	1	1	0	0

특성 행렬(X)

=

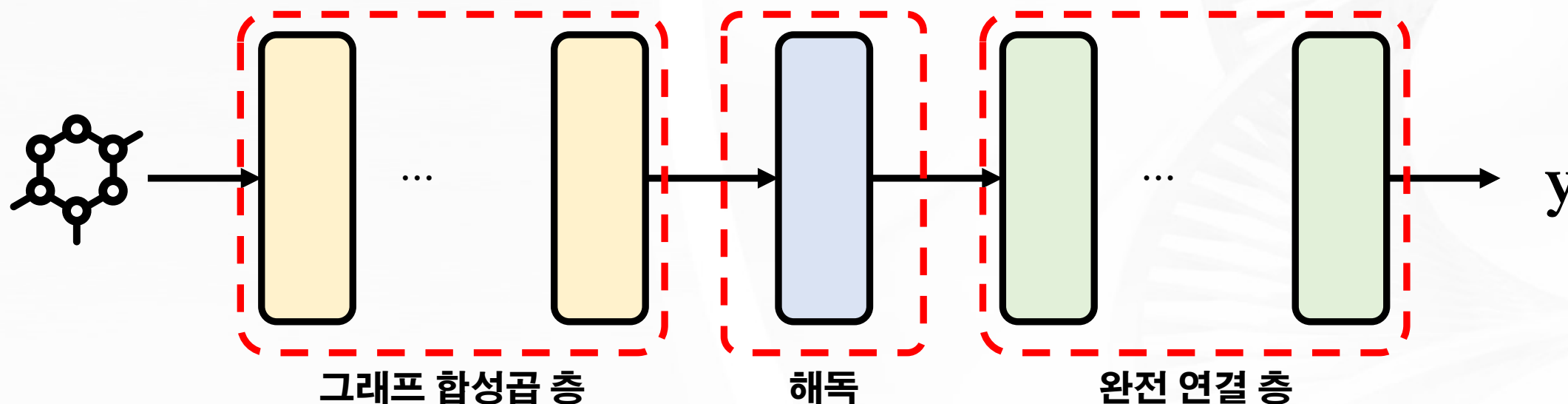
	1	2	3	4	5	6
A	$\frac{1}{\sqrt{10}}$	$\frac{1}{2}$	0	$\frac{1}{2}$	$\frac{1}{\sqrt{10}}$	$\frac{1}{2}$
B	$\frac{2+\sqrt{10}}{2\sqrt{10}}$	$\frac{1}{2}$	$\frac{1}{2}$	0	$\frac{1}{\sqrt{10}}$	$\frac{1}{2}$
C	$\frac{10+\sqrt{10}}{5\sqrt{10}}$	$\frac{3}{\sqrt{10}}$	$\frac{3}{\sqrt{10}}$	$\frac{2}{\sqrt{10}}$	$\frac{1}{5}$	$\frac{3}{\sqrt{10}}$
D	$\frac{2+\sqrt{10}}{2\sqrt{10}}$	$\frac{1}{2}$	$\frac{1}{2}$	0	$\frac{1}{\sqrt{10}}$	$\frac{1}{2}$
E	$\frac{1}{\sqrt{10}}$	0	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{\sqrt{10}}$	0

은닉 정점 특성 행렬(H)

Graph Convolutional Networks

• 개요

- 일반적으로 그래프 합성곱 층과 완전 연결 층으로 구성
- 그래프 분류 및 회귀 문제에서는 해독 과정이 필수
- 노드 분류 또는 링크 예측의 경우 해독 과정이 필요 없음



Graph Convolutional Networks

• 연산

- 각 그래프 합성곱 층은 그래프 합성곱 연산으로 정의
- 그래프 합성곱 층을 반복적으로 적용
- 그래프에 포함된 정점에서 추상화된 은닉 특성을 추출

$$\mathbf{H}^{(k)} = \sigma(\tilde{\mathbf{D}}^{-1/2} \tilde{\mathbf{A}} \tilde{\mathbf{D}}^{-1/2} \mathbf{H}^{(k-1)} \mathbf{W}^{(k)})$$

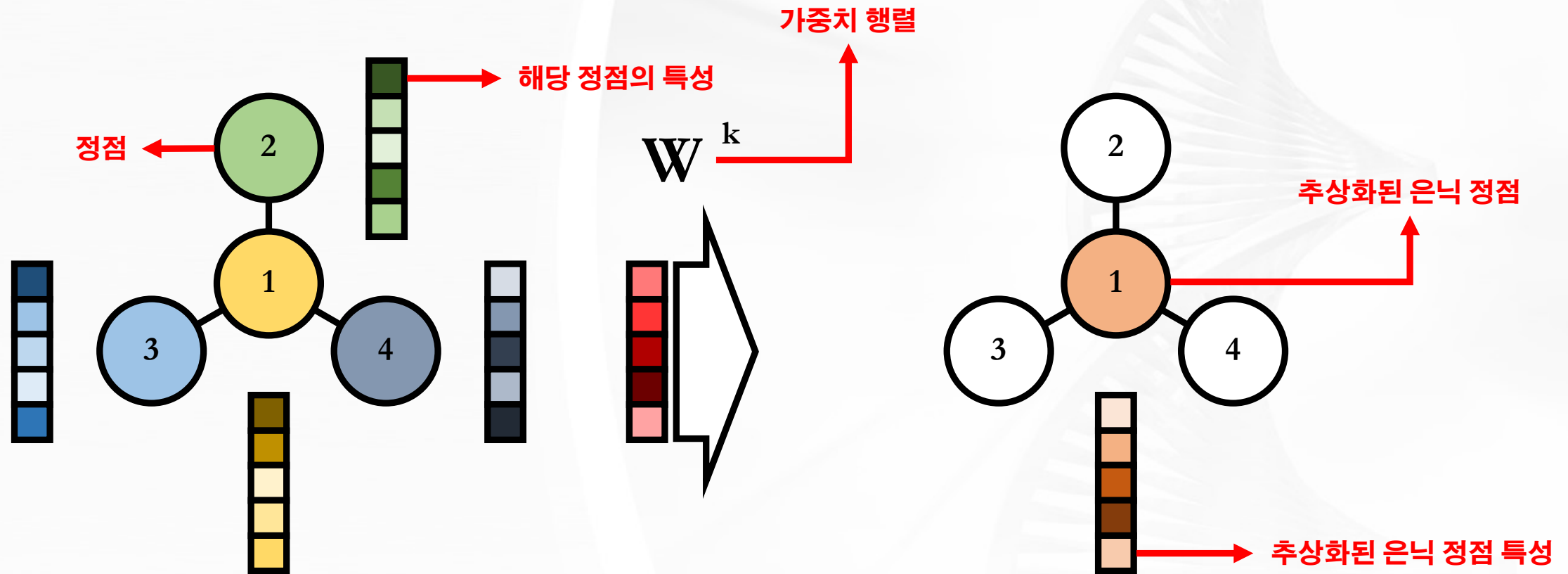
$\mathbf{H}^{(k)}$ (red arrow) → k 번째 그래프 합성곱 층 출력

$\mathbf{H}^{(k-1)}$ (green arrow) → 이전 그래프 합성곱 층 출력

$\mathbf{W}^{(k)}$ (blue arrow) → k 번째 층의 가중치 행렬

Graph Convolutional Networks

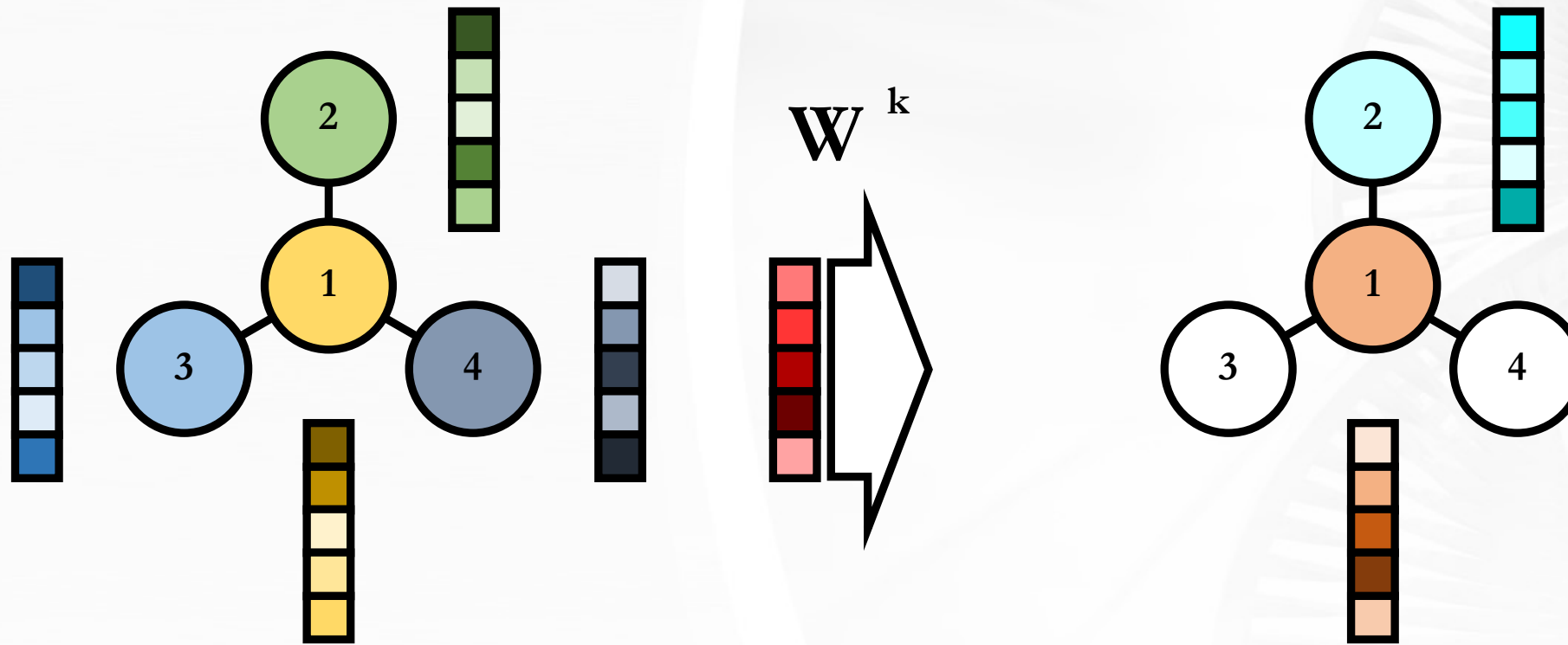
• 정점 특성 학습 과정



$$H_1^{k+1} = \sigma(H_1^k W^k + H_2^k W^k + H_3^k W^k + H_4^k W^k)$$

Graph Convolutional Networks

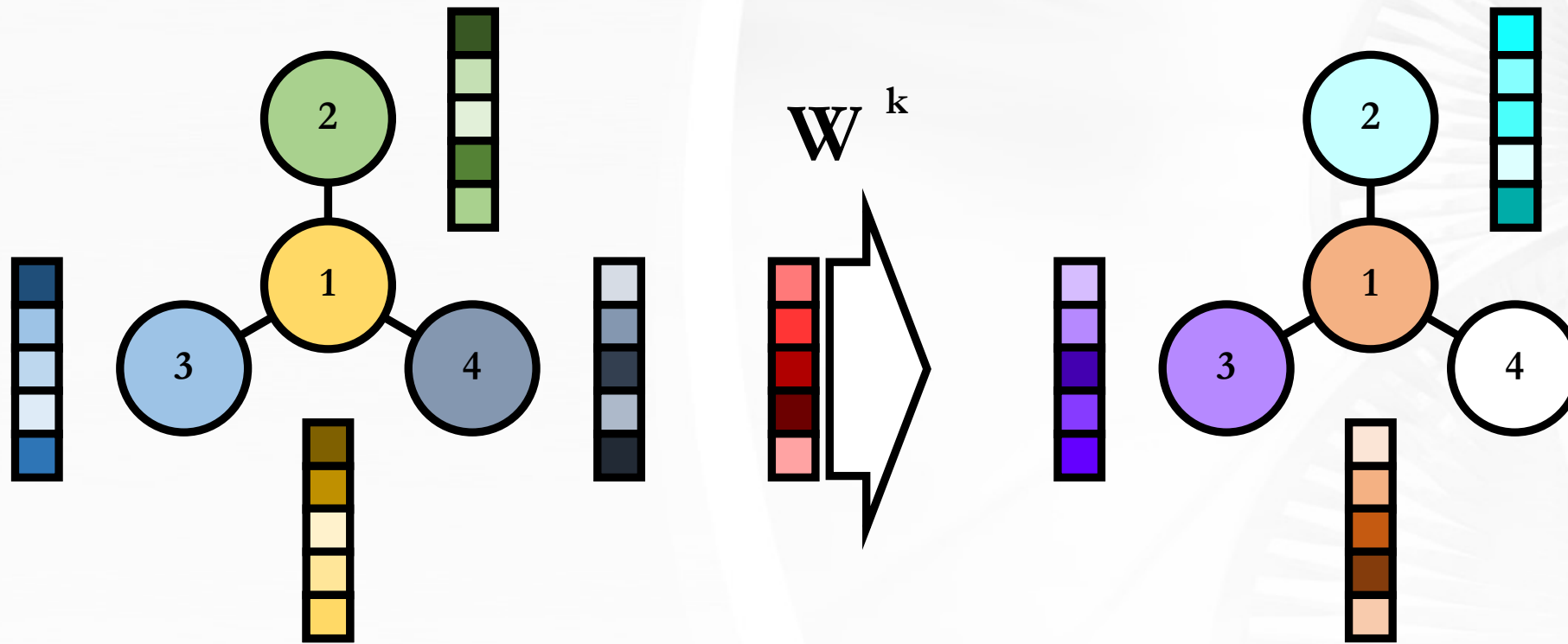
• 정점 특성 학습 과정



$$H_2^{k+1} = \sigma(H_1^k W^k + H_2^k W^k + \cancel{H_3^k W^k} + \cancel{H_4^k W^k})$$

Graph Convolutional Networks

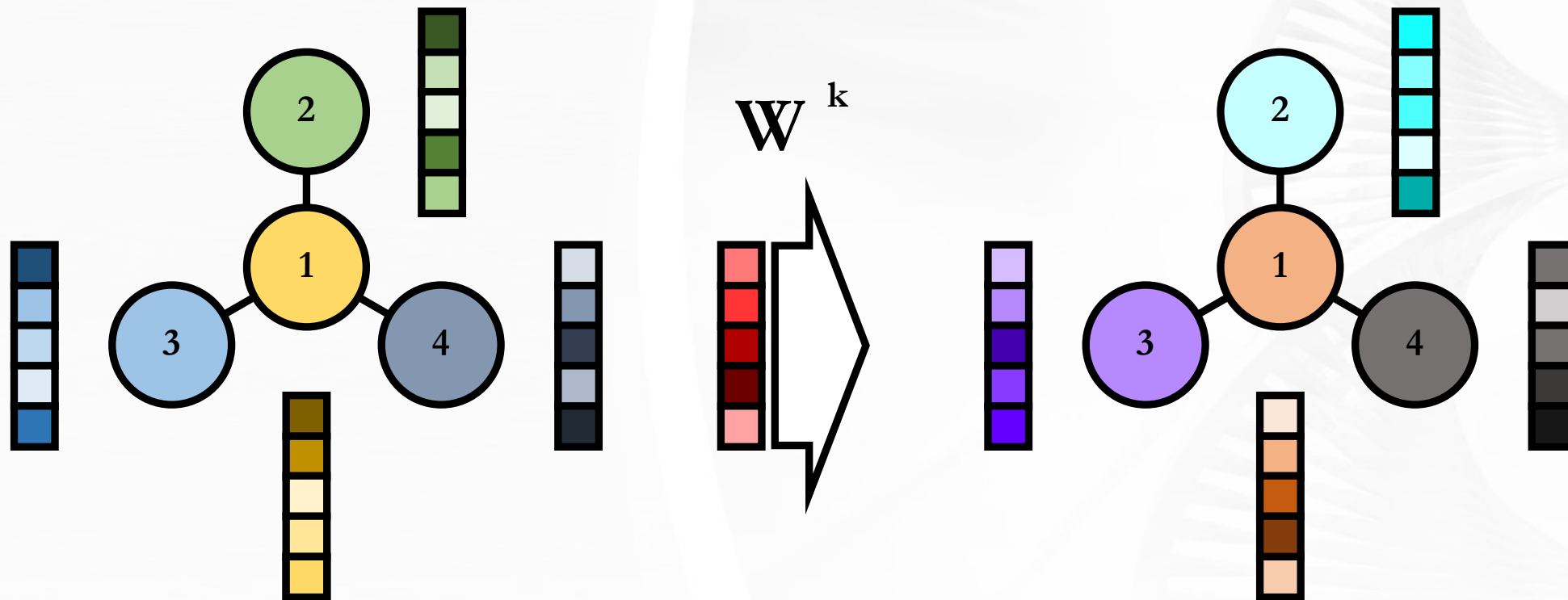
• 정점 특성 학습 과정



$$H_3^{k+1} = \sigma(H_1^k W^k + \cancel{H_2^k W^k} + H_3^k W^k + \cancel{H_4^k W^k})$$

Graph Convolutional Networks

• 정점 특성 학습 과정



$$H_4^{k+1} = \sigma(H_1^k W^k + \cancel{H_2^k W^k} + \cancel{H_3^k W^k} + H_4^k W^k)$$

Graph Convolutional Networks

• GCN 구현

- 직접 구현 시, 인접 행렬 및 특성 행렬 추출부터 batch 생성까지 많은 어려움이 따름
- PyTorch 기준, Deep Graph Library(DGL)와 PyTorch Geometric를 통해 간편하게 구현 가능
- 본 강의(논문)에서는 PyTorch Geometric으로 GCN-based Model을 구현

```
import torch
import torch.nn as nn
import torch.nn.functional as F
from torch_geometric.nn import GCNConv, global_max_pool as gmp

class GCNNet(torch.nn.Module):
    def __init__(self, n_output=1, n_filters=32, embed_dim=128, num_features_xd=78, num_features_xt=25, output_dim=128, dropout=0.2):
        super(GCNNet, self).__init__()

        self.n_output = n_output
        self.conv1 = GCNConv(num_features_xd, num_features_xd)
        self.conv2 = GCNConv(num_features_xd, num_features_xd*2)
        self.conv3 = GCNConv(num_features_xd*2, num_features_xd*4)
        self.fc_g1 = torch.nn.Linear(num_features_xd*4, 1024)
        self.fc_g2 = torch.nn.Linear(1024, output_dim)
        self.relu = nn.ReLU()
        self.dropout = nn.Dropout(dropout)

        self.embedding_xt = nn.Embedding(num_features_xt+1, embed_dim)
        self.conv_xt_l = nn.Conv1d(in_channels=1000, out_channels=n_filters, kernel_size=8)
        self.fcl_xt = nn.Linear(32*121, output_dim)

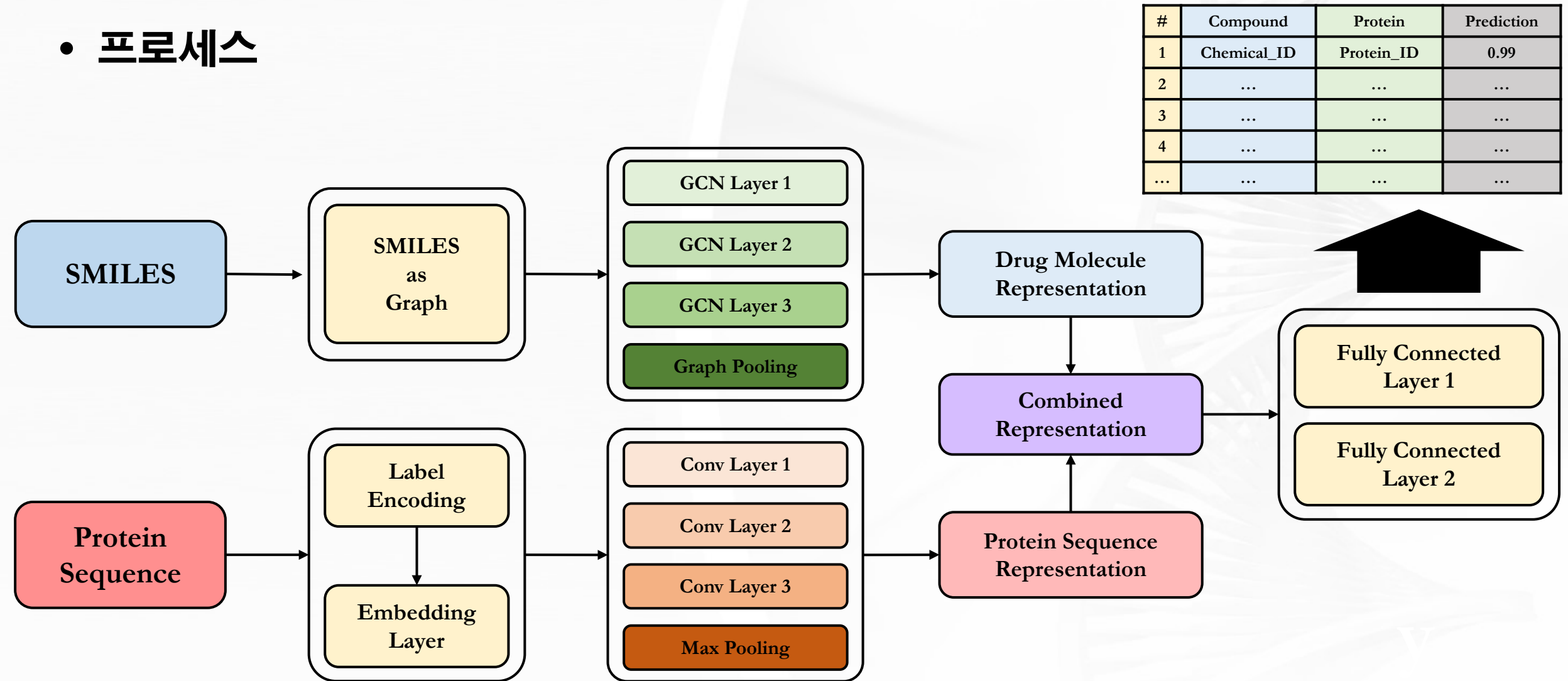
        self.fc1 = nn.Linear(2*output_dim, 1024)
        self.fc2 = nn.Linear(1024, 512)
        self.out = nn.Linear(512, self.n_output)
```

코드 구현 예시

GraphDTA Model

Model

- 프로세스



Model

• 실험 설정

- Davis, Kiba 두 데이터셋에서 DeepDTA, WideDTA와 동일한 train/test split을 사용함
- 데이터 인스턴스의 80%를 훈련에 사용했으며, 나머지 20%를 테스트에 사용함
- 기존 논문과 동일 선상에서 성능을 비교하기 위해 MSE와 CI로 성능을 평가함
- MSE : Mean Square Error, 평균 제곱 오차(낮을수록 좋은 성능)
- CI : Concordance Index, 일치 지수(높을수록 좋은 성능)
- 실습 코드에선 다양한 성능평가지표의 시각화를 위해 RMSE, PCC, SRCC 추가
- 하이퍼 파라미터 설정은 코드 리뷰에서 다룰 예정

Model

• 결과

- Davis, KIBA 두 데이터 셋에서 Baseline(DeepDTA, KronRLS, SimBoost)보다 **좋은 성능을 도출함**

Method	Protein rep.	Compound rep.	CI	MSE
Baseline models				
DeepDTA	Smith-Waterman	Pubchem-Sim	0.790	0.608
DeepDTA	Smith-Waterman	1D	0.886	0.420
DeepDTA	1D	Pubchem-Sim	0.835	0.419
KronRLS	Smith-Waterman	Pubchem-Sim	0.871	0.379
SimBoost	Smith-Waterman	Pubchem-Sim	0.872	0.282
DeepDTA	1D	1D	0.878	0.261
WideDTA	1D + PDM	1D + LMCS	0.886	0.262
Proposed model - GraphDTA				
GCN [17]	1D	Graph	0.880	0.254
GAT_GCN	1D	Graph	0.881	0.245
GAT [37]	1D	Graph	0.892	0.232
GIN [40]	1D	Graph	0.893	0.229

Davis Dataset Prediction Performance

Method	Protein rep.	Compound rep.	CI	MSE
Baseline models				
DeepDTA	1D	Pubchem-Sim	0.718	0.571
DeepDTA	Smith-Waterman	Pubchem-Sim	0.710	0.502
KronRLS	Smith-Waterman	Pubchem-Sim	0.782	0.411
SimBoost	Smith-Waterman	Pubchem-Sim	0.836	0.222
DeepDTA	Smith-Waterman	1D	0.854	0.204
DeepDTA	1D	1D	0.863	0.194
WideDTA	1D + PDM	1D + LMCS	0.875	0.179
Proposed model - GraphDTA				
GAT [37]	1D	Graph	0.866	0.179
GIN [40]	1D	Graph	0.882	0.147
GCN [17]	1D	Graph	0.889	0.139
GAT_GCN	1D	Graph	0.891	0.139

Kiba Dataset Prediction Performance

Code Review

Code Review

- Google Colab

- <https://bit.ly/33aJfZp>

- 반드시 드라이브에 사본 저장 후 진행해주세요!