KSBi-BIML 2024

Bioinformatics & Machine Learning(BIML)
Workshop for Life and Medical Scientists

생명정보학 & 머신러닝 워크샵 (오프라인)

Deep Learning in Bioinformatics

노미나 _ 한양대학교

KSBI KOREAN SOCIETY FOR BIOINFORMATICS | 한국생명정보학회

본 강의 자료는 한국생명정보학회가 주관하는 BIML 2024 워크샵 오프라인 수업을 목적으로 제작된 것으로 해당 목적 이외의 다른 용도로 사용할 수 없음을 분명하게 알립니다.

이를 다른 사람과 공유하거나 복제, 배포, 전송할 수 없으며 만약 이러한 사항을 위반할 경우 발생하는 **모든 법적 책임은 전적으로 불법 행위자 본인에게 있음을 경고**합니다.

KSBi-BIML 2024

Bioinformatics & Machine Learning(BIML)
Workshop for Life and Medical Scientists

안녕하십니까?

한국생명정보학회가 개최하는 동계 교육 워크샵인 BIML-2024에 여러분을 초대합니다. 생명정보학 분야의 연구자들에게 최신 동향의 데이터 분석기술을 이론과 실습을 겸비해 전달하고자 도입한 전문 교육 프로그램인 BIML 워크샵은 2015년에 시작하여 올해로 벌써 10년 차를 맞이하게 되었습니다. BIML 워크샵은 국내 생명정보학 분야의 최초이자 최고 수준의 교육프로그램으로 크게 인공지능과 생명정보분석 두 개의 분야로 구성되어 있습니다. 올해 인공지능 분야에서는 최근 생명정보 분석에서도 응용이 확대되고 있는 다양한 인공지능 기반 자료모델링 기법들에 대한 현장 강의가 진행될 예정이며, 관련하여 심층학습을 이용한 단백질구조예측, 유전체분석, 신약개발에 대한 이론과 실습 강의가 함께 제공될 예정입니다. 또한 단일세포오믹스, 공간오믹스, 메타오믹스, 그리고 롱리드염기서열 자료 분석에 대한 현장 강의는 많은 연구자의 연구 수월성 확보에 큰 도움을 줄 것으로 기대하고 있습니다.

올해 BIML의 가장 큰 변화는 최근 연구 수요가 급증하고 있는 의료정보자료 분석에 대한 현장 강의를 추가하였다는 것입니다. 특히 의료정보자료 분석을 많이 수행하시는 의과학자 및 의료정보 연구자들께서 본 강좌를 통해 많은 도움을 받으실 수 있기를 기대하고 있습니다. 또한 다양한 생명정보학 분야에 대한 온라인 강좌 프로그램도 점차 증가하고 있는 생명정보 분석기술의 다양화에 발맞추기 위해 작년과 비교해 5강좌 이상을 신규로 추가했습니다. 올해는 무료 강좌 5개를 포함하여 35개 이상의 온라인 강좌가 개설되어 제공되며, 연구 주제에 따른 연관된 강좌 추천 및 강연료 할인 프로그램도 제공되며, 온라인을 통한 Q&A 세션도 마련될 예정입니다. BIML-2024는 국내 주요 연구중심 대학의 전임 교원이자 각 분야 최고 전문가들의 강의로 구성되었기에 해당 분야의 기초부터 최신 연구 동향까지 포함하는 수준 높은 내용의 강의가 될 것이라 확신합니다.

BIML-2024을 준비하기까지 너무나 많은 수고를 해주신 운영위원회의 정성원, 우현구, 백대현, 김태민, 김준일, 김상우, 장혜식, 박종은 교수님과 KOBIC 이병욱 박사님께 커다란 감사를 드립니다. 마지막으로 부족한 시간에도 불구하고 강의 부탁을 흔쾌히 허락하시고 훌륭한 현장 강의와 온라인 강의를 준비하시는데 노고를 아끼지 않으신 모든 강사분들께 깊은 감사를 드립니다.

2024년 2월

한국생명정보학회장 이 인 석

강의 시간표

DAY1 : 2월 24일 (토)

시간	강 의 (자연과학대학 28동 101호)
12:30-12:50	등록
12:50-13:00	공지사항 전달
13:00-14:30	**의료빅데이터/인공지능 총론** 김헌성 교수(가톨릭대학교)
14:30-14:45	휴식
14:45-16:15	**의료영상 인공지능의 이해 및 의료영상 레이블링 실습** 백서연 교수(연석대학교)
16:15-16:30	휴식
16:30-18:00	**의료 정보처리 자동화 실습 / 독자적인 어플리케이션 만들기** 김선근 대표(원닥 주식회사), 서사도 조교

시간	강 의 (자연과학대학 28동 102호)
12:30-12:50	등록
12:50-13:00	공지사항 전달
13:00-14:20	**EMR 데이터를 활용한 머신러닝 기반 예후예측:** **Decision Tree-based Models + EMR 샘플 데이터 실습 (MIMIC sample dataset)** 고태훈 교수(가톨릭대학교)
14:20-14:40	휴식
14:40-16:00	**Chest X-ray 영상을 활용한 딥러닝 기반 폐질환 진단:** **Convolutional Neural Network + 의료영상 샘플 데이터 실습 (NIH Chest X-ray14)** 고태훈 교수(가톨릭대학교)
16:00-16:20	휴식
16:20-17:40	**심전도 데이터를 활용한 딥러닝 기반 부정맥 탐지: Recurrent Neural Network +** **Transformer + 심전도 샘플 데이터 실습 (MIT-BIH Arrhythmia Database)** 고태훈 교수(가톨릭대학교)

DAY1 : 2월 26일 (월)

시간	강 의 (자연과학대학 28동 101호)
09:00-09:20	등록
09:20-09:30	공지사항 전달
09:30-10:50	**DNN (이론)** 이상근 교수(고려대학교)
10:50-11:00	휴식
11:00-12:10	**CNN (이론)** 이상근 교수(고려대학교)
12:10-13:40	점심
13:40-15:10	**RNN, ChatGPT, XAI (이론)** 이상근 교수(고려대학교)
15:10-15:20	휴식
15:20-16:50	**CNN/RNN 모델 구조 정의, 학습 알고리즘 적용, 성능 평가, 시각화 방법 (Tensorflow 실습)** 이정현 조교, 한성민 조교

시간	강 의 (자연과학대학 28동 102호)
09:00-09:20	등록
09:20-09:30	공지사항 전달
09:30-11:00	**Best practice for single-cell data analysis** 박종은 교수(KAIST)
11:00-11:10	휴식
11:10-12:40	**Practice1: Scanpy basic workflow** 정성민 조교, 고용준 조교
12:40-14:10	점심
14:10-15:30	**Public database, data integration, reference mapping, multiomics** 박종은 교수(KAIST)
15:30-15:40	휴식
15:40-16:50	**Practice2: Advanced single-cell analysis (siVI universe)** 정성민 조교, 고용준 조교

DAY1 : 2월 27일 (화)

시간	강 의 (자연과학대학 28동 101호)
09:00-09:20	등록
09:20-09:30	공지사항 전달
09:30-10:50	**AI-based protein structure prediction** **- Intro to protein structure prediction** **- Early AI-based approaches** **- AlphaFold and RoseTTAFold** 백민경 교수(서울대학교)
10:50-11:00	휴식
11:00-12:10	**단백질 구조 예측 실습** **- ColabFold를 활용한 단백질 구조 및 상호작용 예측** **- Tips &Tricks for better structure modeling** 백민경 교수(서울대학교)
12:10-13:40	점심
13:40-15:10	**AI-based protein design** **- Intro to protein design** **- Protein backbone design using RFdiffusion** **- Protein sequence design using ProteinMPNN** 백민경 교수(서울대학교)
15:10-15:20	휴식
15:20-16:50	**단백질 디자인 실습** **- RFdiffusion 및 ProteinMPNN의 활용법 실습** 백민경 교수(서울대학교)

시간	강 의 (자연과학대학 28동 102호)
09:00-09:20	등록
09:20-09:30	공지사항 전달
09:30-11:00	**Introduction to Single-cell biology** 최정민 교수(고려대학교)
11:00-11:10	휴식
11:10-12:40	**i. Unsupervised Spatial transcriptome analysis** **ii. Tumor Boundary Determination in Spatial Transcriptomics** 유광민 조교, 이문영 조교
12:40-14:10	점심
14:10-15:30	**i. Deconvolution Analysis Using Single-cell RNA Sequencing and Spatial Transcriptomics** **ii. Cell-Cell Interaction Analysis in Spatial Transcriptomics** 김지현 조교, 최승지 조교
15:30-15:40	휴식
15:40-16:50	**i. Open Chromatin Region Analysis and Biological Interpretation of Using scATAC-seq Dataset** **ii. Construction of Gene Regulatory Networks Based on Integrated Analysis of scATAC-seq and scRNA-seq Datasets** 천하림 조교, 이호진 조교

DAY1 : 2월 28일 (수)

시간	강 의 (자연과학대학 28동 101호)
09:00-09:20	등록
09:20-09:30	공지사항 전달
09:30-11:00	**Introduction to Transformers (이론)** 전민지 교수 (고려대학교)
11:00-11:10	휴식
11:10-12:40	**Introduction to Transformers (실습)** 봉현수 조교, 임우택 조교
12:40-14:10	점심
14:10-15:40	**Deep learning in Bioinformatics** 노미나 교수(한양대학교)
15:40-15:50	휴식
15:50-17:20	**Deep learning model을 이용한 실습** 박예솔 조교

시간	강 의 (자연과학대학 28동 102호)
09:00-09:20	등록
09:20-09:30	공지사항 전달
09:30-10:50	**마이크로바이옴 기본 이론** 이선재 교수(GIST)
10:50-11:00	휴식
11:00-12:10	**16S rRNA amplicon seq. - DADA2** 조준우 조교, 백재우 조교
12:10-13:40	점심
13:40-14:40	**최신 메타지놈 분석 기법의 현황** 이선재 교수(GIST)
14:40-14:50	휴식
14:50-16:50	**Shotgun metagenome 분석 (Linux)** 조준우 조교, 백재우 조교

DAY1 : 2월 29일 (목)

시간	강 의 (자연과학대학 28동 101호)
09:00-09:20	등록
09:20-09:30	공지사항 전달
09:30-10:50	**화학정보학 기초(Cheminformatics) / 약물특성 및 약물다움(druglikeness)** **Molecular Notations &Descriptors / AI 신약개발을 위한 Databases** **AI 신약개발을 위한 Programming 기초** 김동섭 교수(KAIST)
10:50-11:00	휴식
11:00-12:10	**Google Colab에 RDKit 설치 / 화합물 정보 읽기 실습** **Bioactivity database 검색 및 정보 읽기 실습** **Molecular descriptor (fingerprint) 생성 및 similarity 계산 실습** 정수재 조교, 나민주 조교
12:10-13:40	점심
13:40-15:10	**AI 신약개발을 위한 기계학습법 기초 / QSAR 모델링 기초 / AI 신약개발을 위한 딥러닝 모델** **Virtual screening (ligand-based, structure-based) 및 de novo design** 김동섭 교수(KAIST)
15:10-15:20	휴식
15:20-16:50	**QSAR modeling 전체 과정 실습/ 화합물의 Bioactivity 예측 모델 개발** **Virtual screening 과정을 통한 신약후보물질 발굴 실습** 정수재 조교, 나민주 조교

시간	강 의 (자연과학대학 28동 102호)
09:00-09:20	등록
09:20-09:30	공지사항 전달
09:30-11:00	**Single cell multiomics 이론 / Gene regulatory network 이론** 김준일 교수(숭실대학교)
11:00-11:10	휴식
11:10-12:40	**Seurat/Signac, ArchR, TENET+ 실습** 김현규 조교, 정회빈 조교
12:40-14:10	점심
14:10-15:40	**롱리드 시퀀싱 소개 및 유전체 조립 실습** 김준 교수(충남대학교)
15:40-15:50	휴식
15:50-17:20	**변이 분석 및 시각화 실습** 김준 교수(충남대학교)

강의개요

Deep Learning in Bioinformatics

최근 다양한 생물 종, 인체 부위 및 질환 관련 오믹스 데이터는 점점 더 많이 생산되고 있으며, 중요한 의생명과학의 문제에 대한 해답을 구하고자 활용되고 있다. 신약 개발을 포함한 많은 문제에서 대규모 분자구조를 이용한 기능, 반응성 예측도 매우 중요한 문제이다. 딥러닝은 비선형 변환의 조합을 이용하여 데이터의 높은 수준의 추상화를 통해 특성을 추출함으로써 예측이나 분류와 같은 문제를 해결하고자 한다. 따라서 딥러닝은 의생명과학 분야에서도 좋은 성과를 보이고 있다.

본 강의에서는 대용량 유전체, 분자구조, 텍스트 데이터와 다양한 딥러닝 모델을 이용하여 유전인자나 그들의 기능을 예측하는 방법들을 소개한다.

강의는 다음의 내용을 포함한다:

- 딥러닝 모델 개요
- 시퀀스 데이터를 이용한 유전인자 예측
- 분자구조 데이터를 이용한 기능 예측
- 텍스트 데이터를 이용한 정보 추출

* 참고강의교재:

  Deep Learning (Aaron Courville, Ian Goodfellow, and Yoshua Bengio, 2015)

  Deep Learning for the Life Sciences (Peter Eastman, Patrick Walters, Bharath Ramsundar, Vijay S. Pande, 2019)

* 교육생준비물: 노트북 (Google Colab 이용)

* 강의 난이도: 중/고급

* 강의: 노미나교수 (한양대학교 컴퓨터소프트웨어학부)

Curriculum Vitae

Speaker Name: Mina Rho, Ph.D.

▶ **Personal Info**

Name	Mina Rho
Title	Professor
Affiliation	Hanyang University

▶ **Contact Information**

Address	222, Wangsimni-ro, Sungdong-Gu, Seoul, 04763
Email	minarho@hanyang.ac.kr
Phone Number	010-3460-9257

Research Interest

Translational bioinformatics, machine learning, and (meta)genomics

Educational Experience

1998	B.S. in Computer Science, Ewha Womans University, Korea
2001	M.S. in Computer Engineering, Boston University, USA
2009	Ph.D. in Computer Science, Indiana University, USA

Professional Experience

2009-2012	Postdoctoral Associate, Dept. of Computer Science, Indiana University, Seoul, Korea
2012-2013	Assistant Professor, Dept. of Biostatistics and Bioinformatics, Roswell Park Cancer Institute, Buffalo, USA
2013-2017	Assistant Professor, Dept. of Computer Science, Hanyang University, Seoul, Korea
2017-2023	Associate Professor, Dept. of Computer Science, Hanyang University, Seoul, Korea
2023-Current	Professor, Dept. of Computer Science, Hanyang University, Seoul, Korea

Selected Publications (5 maximum)

1. HJ Gwak, M Rho (2022) "ViBE: a deep learning model to classify viruses using metagenome sequencing data", Briefings in Bioinformatics: bbac204
2. J Jeon, J Lee, SM Jung, JH Shin, WJ Song, M Rho (2021) "Genomic Determinants Encode for the Reactivity and Regioselectivity of Flavin-Dependent Halogenases in Bacterial Genomes and Metagenomes", mSystems: 6(3)
3. Y Park, J Lee, H Moon, Y Choi, M Rho (2021) "Discovering microbe-disease associations from the literature using a hierarchical long short-term memory network and an ensemble parser model", Scientific Reports: 11(5874)
4. HJ Gwak, M Rho (2020) "Data-driven modeling for species-level taxonomic assignment from 16S rRNA: Application to human microbiomes", Frontiers in microbiology :11
5. SK Lim, D Kim, DC Moon, Y Cho, M Rho (2020) "Antibiotic resistomes discovered in the gut microbiomes of swine and cattle", Giga Science :9(5)

KSBI | 한국생명정보학회

KSBi-BIML 2023

Deep Learning in Bioinformatics

Machine Learning

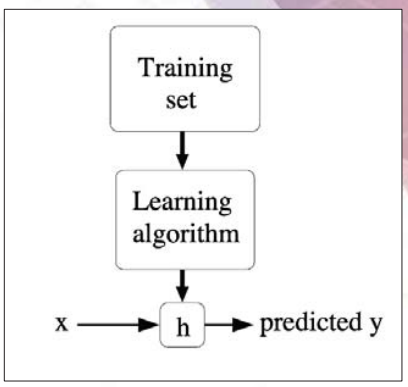
Function approximation

Problem setting:

Set of instances (examples) $X = \{ x^1, \ldots, x^n \}$

Unknown target function $f: X \rightarrow Y$

Set of function hypothesis $H = \{h \mid h : X \rightarrow Y\}$



Input:

Training examples $\{ (x^i, y^i) \}$ of unknown target function f

Output:

Hypothesis $h \in H$ that best approximates target function f, $h \approx f$

2

Deep Learning

REVIEW

doi:10.1038/nature14539

Deep learning

Yann LeCun[1,2], Yoshua Bengio[3] & Geoffrey Hinton[4,5]

Deep learning allows computational models that are composed of multiple processing layers to learn representations of data with multiple levels of abstraction. These methods have dramatically improved the state-of-the-art in speech recognition, visual object recognition, object detection and many other domains such as drug discovery and genomics. Deep learning discovers intricate structure in large data sets by using the backpropagation algorithm to indicate how a machine should change its internal parameters that are used to compute the representation in each layer from the representation in the previous layer. Deep convolutional nets have brought about breakthroughs in processing images, video, speech and audio, whereas recurrent nets have shone light on sequential data such as text and speech.

[1]Facebook AI Research, 770 Broadway, New York, New York 10003 USA. [2]New York University, 715 Broadway, New York, New York 10003, USA. [3]Department of Computer Science and Operations Research Université de Montréal, Pavillon André-Aisenstadt, PO Box 6128 Centre-Ville STN Montréal, Quebec H3C 3J7, Canada. [4]Google, 1600 Amphitheatre Parkway, Mountain View, California 94043, USA. [5]Department of Computer Science, University of Toronto, 6 King's College Road, Toronto, Ontario M5S 3G4, Canada.

3

What is Representation?

	# of words	# of attached files	# of links	# of malicious words	.....	spam
mail #1	256	0	3	7	.....	1 (Yes)
mail #2	56	1	0	3	.....	0 (No)
mail #3	24	1	0	1	.....	0
mail #4	672	0	0	0	.....	0
mail #5	67	2	4	3	.....	1
mail #6	48	0	2	6	.....	0
mail #7	79	1	3	8	.....	1

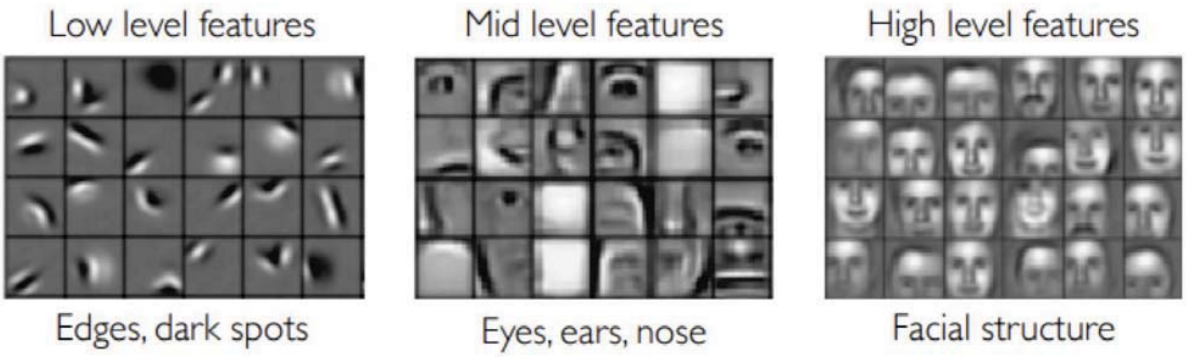
$$X^1 = (x_1^1, x_2^1, ..., x_n^1)$$

:

4

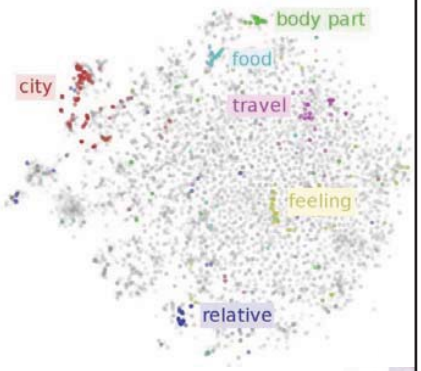
Learning representations for images

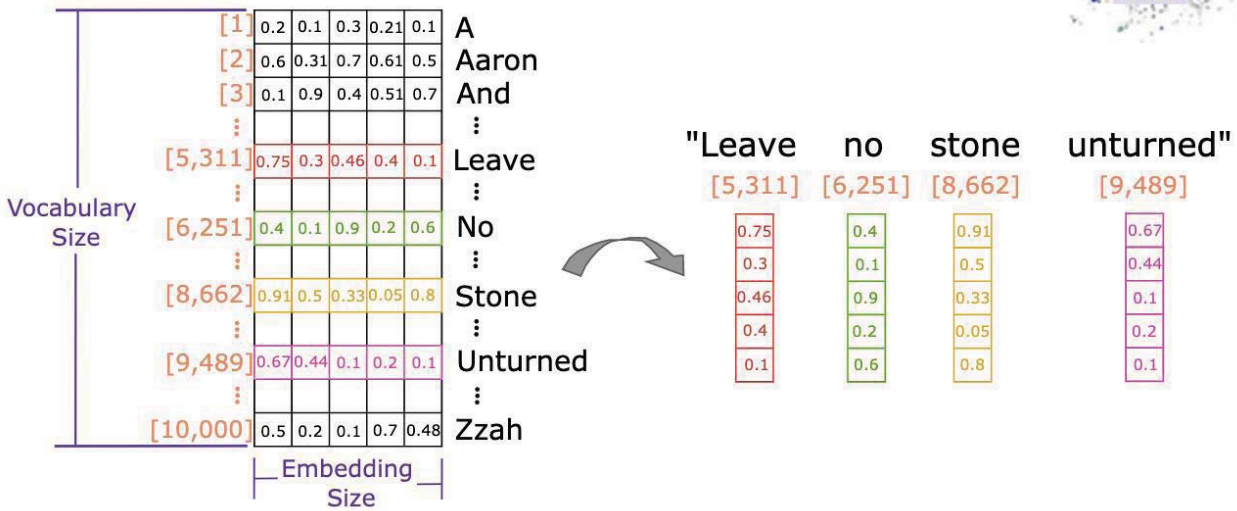
- Hand-engineered features are time-consuming and not scalable in practice.
- Can we learn the underlying features directly from the data?



$$X^1 = (x_1^1, x_2^1, \ldots, x_n^1) \longrightarrow Z^1 = (z_1^1, z_2^1, \ldots, z_m^1)$$

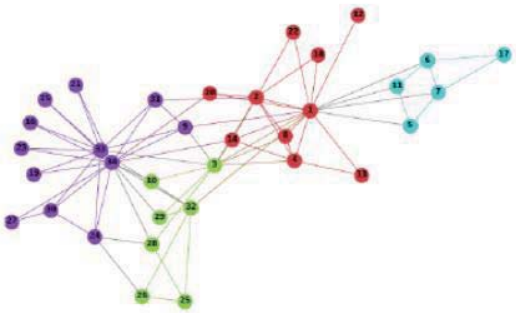
Learning representations for words



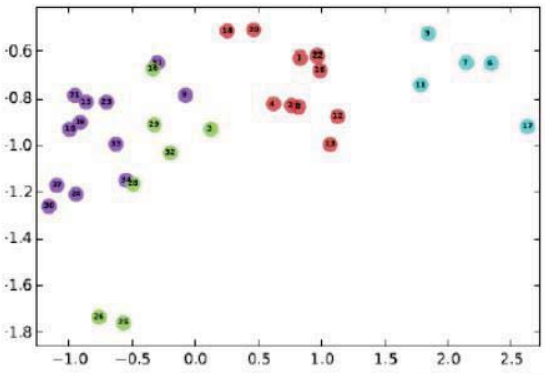


eLife 2023

Learning representations for graphs



Graph



2-dimentional embedding

-Nodes are individuals and connected if the corresponding individuals are friends.

-The nodes are colored according to the different communities.

-Two dimensional node representation

-The distances between nodes in the embedding space reflect similarity in the original graph.

Learning representations for genomic sequences

ACGCGCTGATGCCCGACACTGACTGACGCG

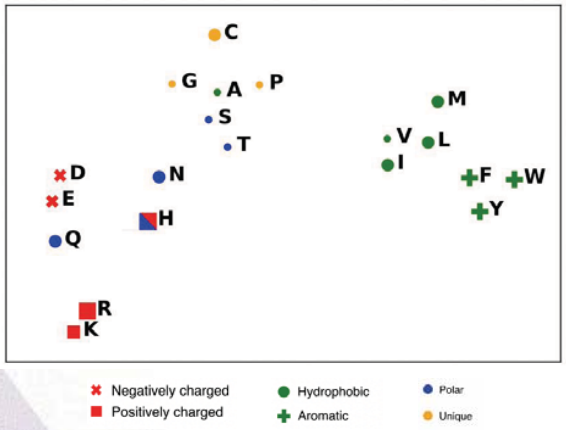
$X^1 = (x_1^1, x_2^1, \ldots)$ → (A, C, G, C, …)

(ACG, CGC, GCG, …)

Learning representations for protein

Protein function is encoded in the amino acid sequence

Sequences can diverge during evolution while maintaining the same function

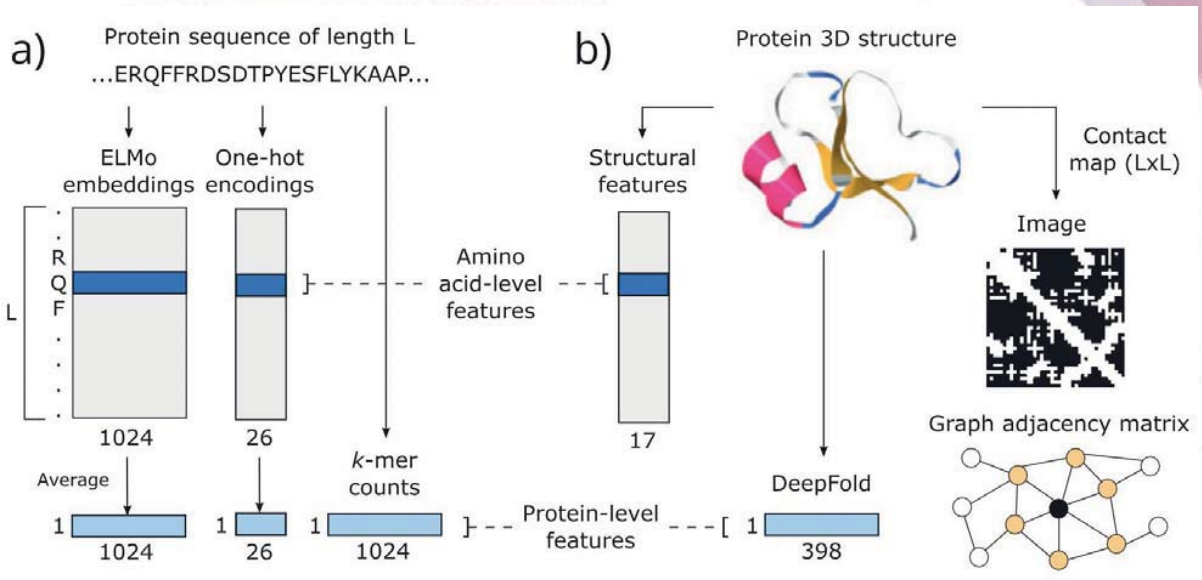


	C	S	T	A	G	P	D	E	Q	N	H	R	K	M	I	L	V	W	Y	F	
C	9																				C
S	-1	4																			S
T	-1	1	5																		T
A	0	1	0	4																	A
G	-3	0	-2	0	6																G
P	-3	-1	-1	-1	-2	7															P
D	-3	0	-1	-2	-1	-1	6														D
E	-4	0	-1	-1	-2	-1	2	5													E
Q	-3	0	-1	-1	-2	-1	0	2	5												Q
N	-3	1	0	-2	0	-2	1	0	0	6											N
H	-3	-1	-2	-2	-2	-2	-1	0	0	1	8										H
R	-3	-1	-1	-1	-2	-2	-2	0	1	0	0	5									R
K	-3	0	-1	-1	-2	-1	-1	1	1	0	-1	2	5								K
M	-1	-1	-1	-1	-3	-2	-3	-2	0	-2	-2	-1	-1	5							M
I	-1	-2	-1	-1	-4	-3	-3	-3	-3	-3	-3	-3	-3	1	4						I
L	-1	-2	-1	-1	-4	-3	-4	-3	-2	-3	-3	-2	-2	2	2	4					L
V	-1	-2	0	0	-3	-2	-3	-2	-2	-3	-3	-3	-2	1	3	1	4				V
W	-2	-3	-2	-3	-2	-4	-4	-3	-2	-4	-2	-3	-3	-1	-3	-2	-3	11			W
Y	-2	-2	-2	-2	-3	-3	-3	-2	-1	-2	2	-2	-2	-1	-1	-1	-1	2	7		Y
F	-2	-2	-2	-2	-3	-4	-3	-3	-3	-3	-1	-3	-3	0	0	0	-1	1	3	6	F
	C	S	T	A	G	P	D	E	Q	N	H	R	K	M	I	L	V	W	Y	F	

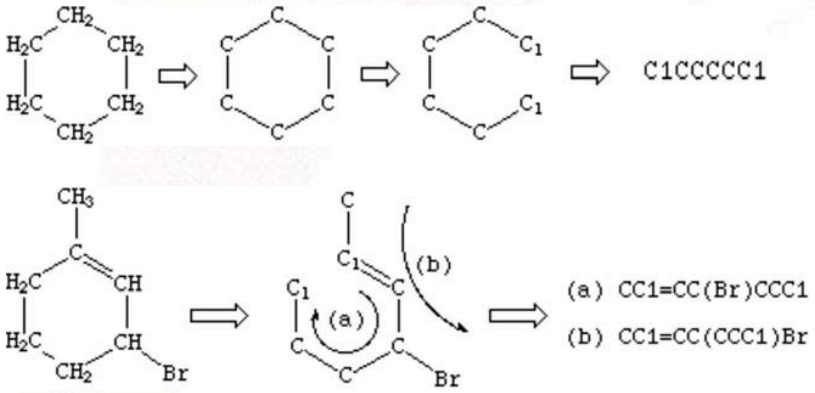
Scoring matrix BLOSUM

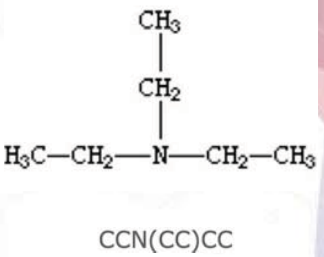
*

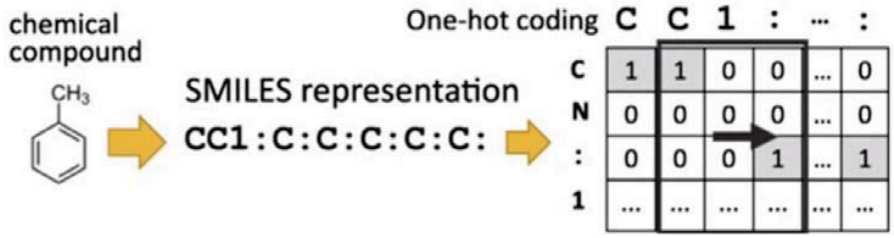
Learning representations for protein



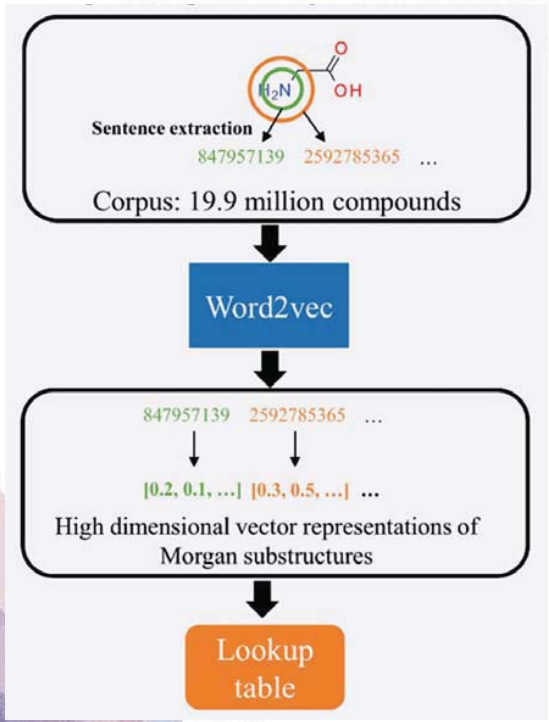
Learning representations for chemical compounds







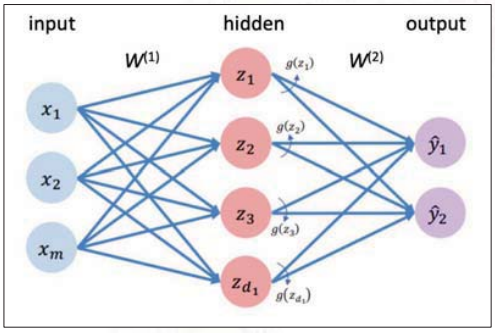
Learning representations for chemical compounds

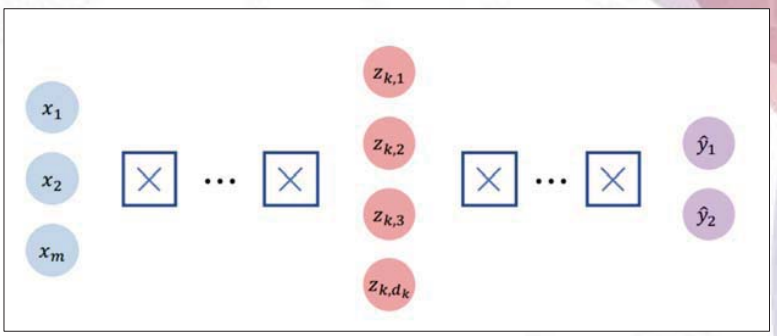


To obtain words from each molecule, the Morgan algorithm was used to generate all atom identifiers at radii 0 and 1, resulting 119 and 19,831 identifiers.

Identifiers are ordered in the same order as the atoms in the canonical SMILES representation for consistency reasons.

Models that are composed of multiple processing layers





$$z_i = w_{0,i}^{(1)} + \sum_{j=1}^{m} x_j w_{j,i}^{(1)}$$

$$\hat{y}_i = g(w_{0,i}^{(2)} + \sum_{j=1}^{d_1} g(z_j) w_{j,i}^{(2)})$$

$$z_{k,i} = w_{0,i}^{(k)} + \sum_{j=1}^{n_{k-1}} g(z_{k-1,j}) w_{j,i}^{(k)}$$

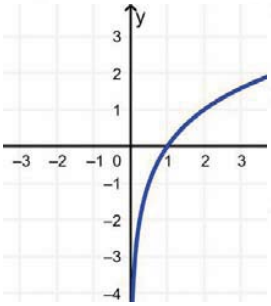
Loss optimization

- The loss of the network measures the cost incurred from incorrect predictions
- Cross entropy loss can be used with models that output a probability between 0 and 1

$$L(\hat{y}, y) = -(y log \hat{y} + (1-y) log(1-\hat{y}))$$

$$y = 1 : L(\hat{y}, y) = -log \hat{y}$$

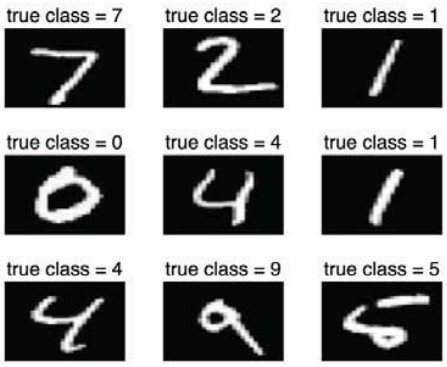
$$y = 0 : L(\hat{y}, y) = -log(1-\hat{y})$$



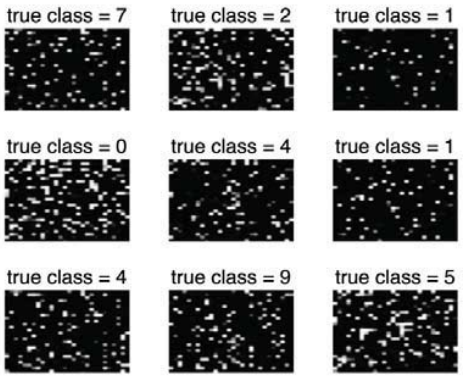
$$J(W) = \frac{1}{N} \sum_{i=1}^{N} L(f(x; W), y)$$

$$W^* = \underset{W}{argmin} J(W)$$

Neural Network



original image 28 x 28 grayscale 0:255

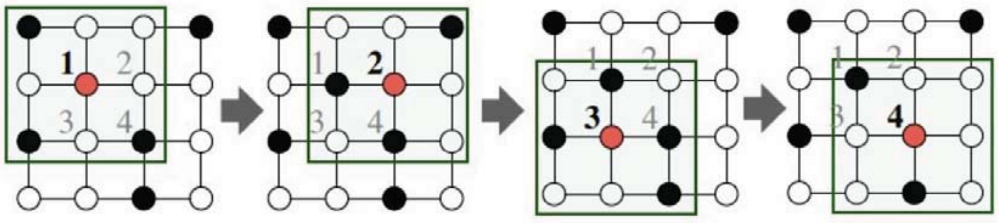


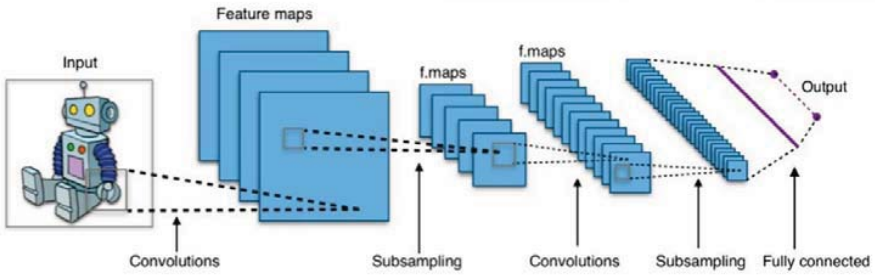
features permuted randomly

- images are 28 x 28 pixels
- represent input image as a vector $x \in \mathbb{R}^{784}$
- Learn a classifier $f(x)$ such that
  $f: x \rightarrow \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$

15

Convolution neural network

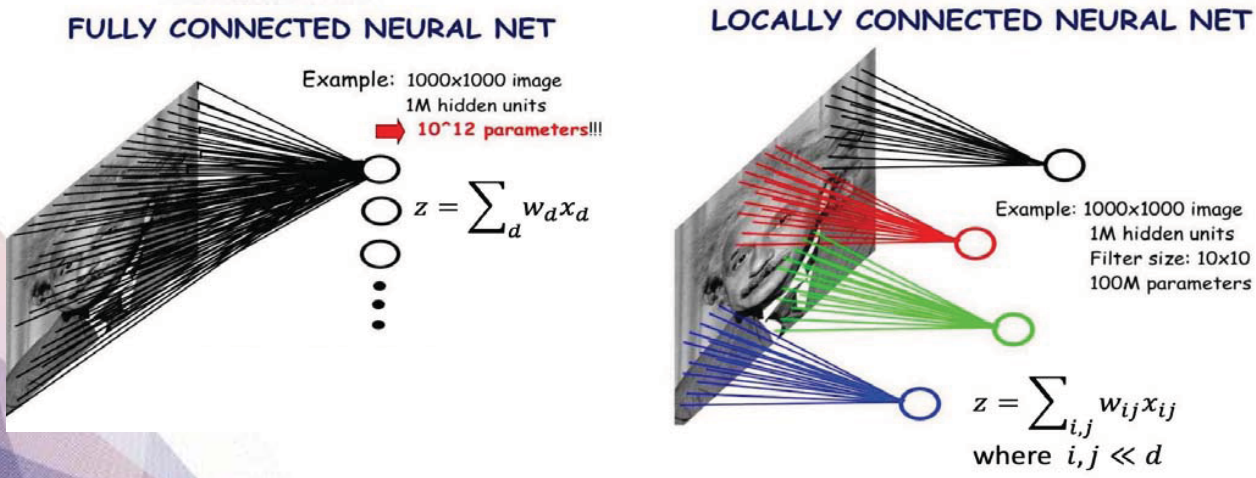




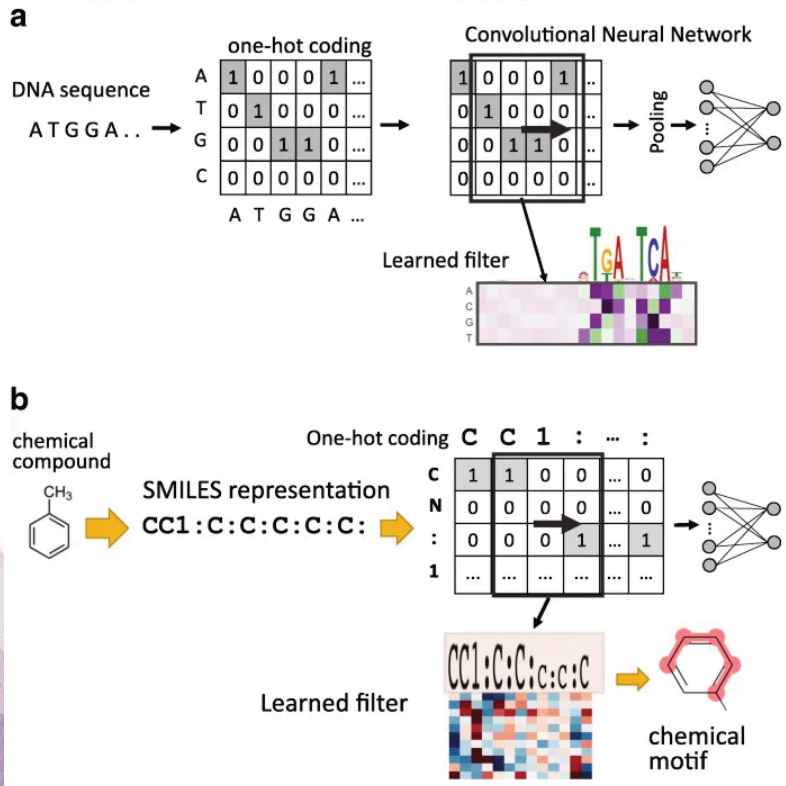
16

Convolution neural network

- Each convolution can be seen as a locally-connected neuron sliding on the entire input features

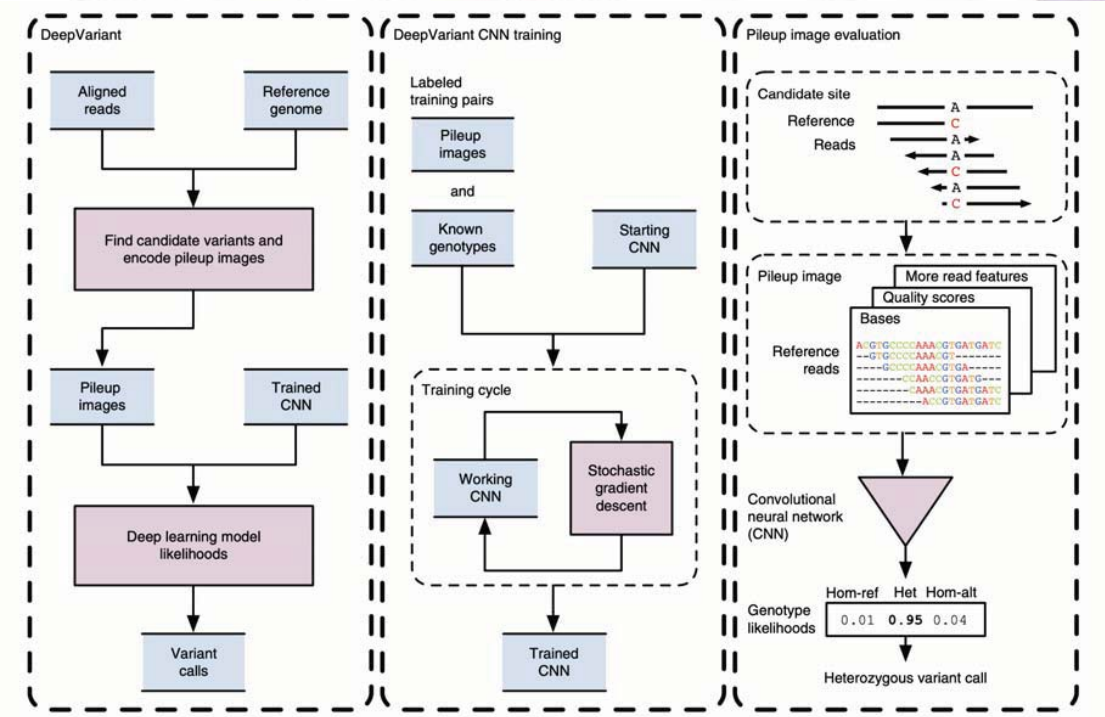


Learning representation with CNN



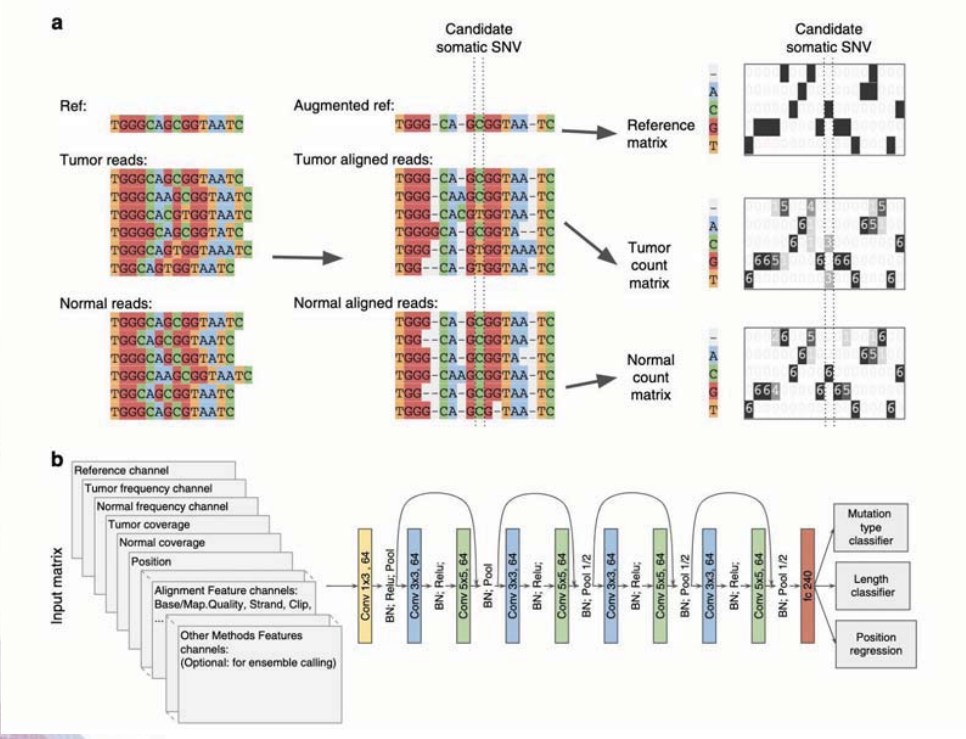
BMC Bioinformatics 2018

Learning representation with CNN: DeepVariant



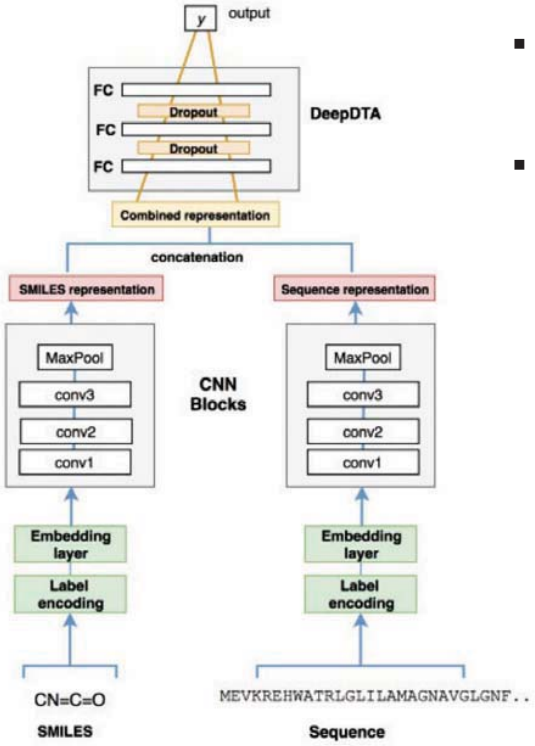
Nature Biotechnology 2018

Learning representation with CNN: NeuSomatic



Nature Communications 2019

Prediction of drug-target interaction using CNN



- The identification of novel drug–target (DT) interactions is a substantial part of the drug discovery process
- A deep-learning based model that uses only sequence information of both targets and drugs to predict DT interaction binding affinities

Table 6. The average r_m^2 and AUPR scores of the test set trained on five different training sets for the KIBA dataset

	Proteins	Compounds	r_m^2 (std)	AUPR (std)
KronRLS (Pahikkala *et al.*, 2014)	S–W	Pubchem Sim	0.342 (0.001)	0.635 (0.004)
SimBoost (He *et al.*, 2017)	S–W	Pubchem Sim	0.629 (0.007)	0.760 (0.003)
DeepDTA	CNN	CNN	0.673 (0.009)	0.788 (0.004)

Word embedding with a simple neural network

output distribution

$$\hat{\boldsymbol{y}} = \text{softmax}(\boldsymbol{U}\boldsymbol{h} + \boldsymbol{b}_2) \in \mathbb{R}^{|V|}$$

hidden layer

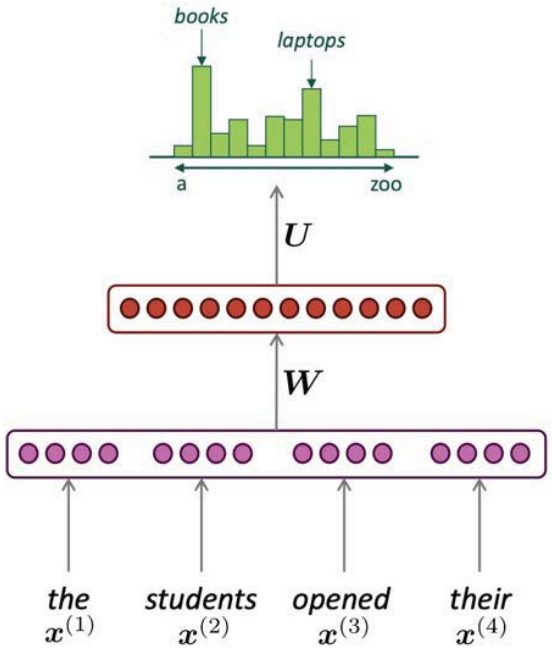
$$\boldsymbol{h} = f(\boldsymbol{W}\boldsymbol{e} + \boldsymbol{b}_1)$$

concatenated word embeddings

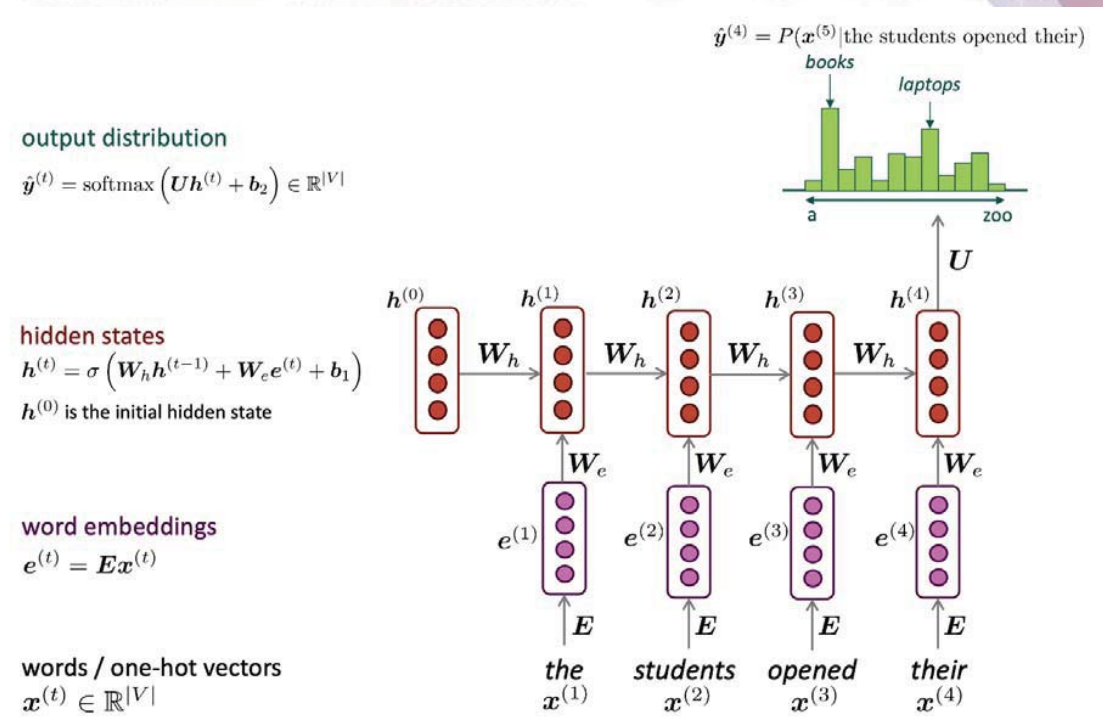
$$\boldsymbol{e} = [\boldsymbol{e}^{(1)}; \boldsymbol{e}^{(2)}; \boldsymbol{e}^{(3)}; \boldsymbol{e}^{(4)}]$$

words / one-hot vectors

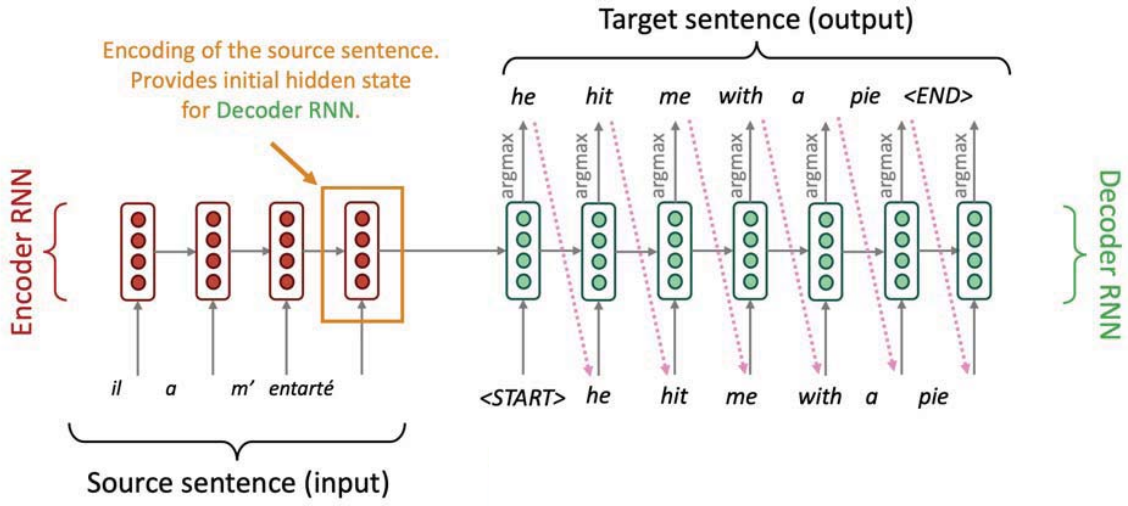
$$\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)}, \boldsymbol{x}^{(3)}, \boldsymbol{x}^{(4)}$$



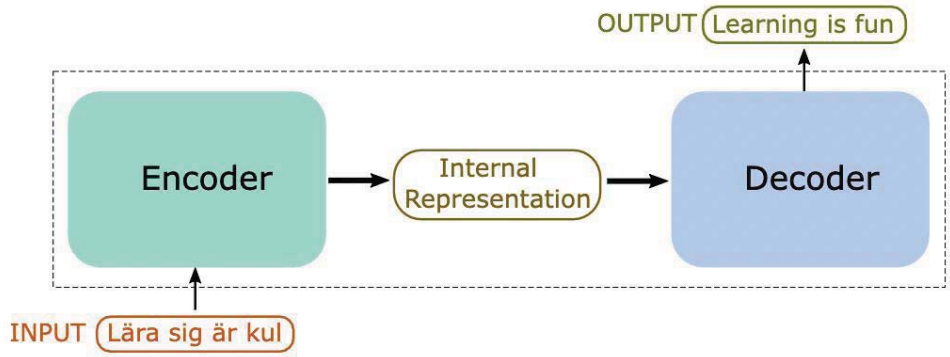
Embedding in RNN

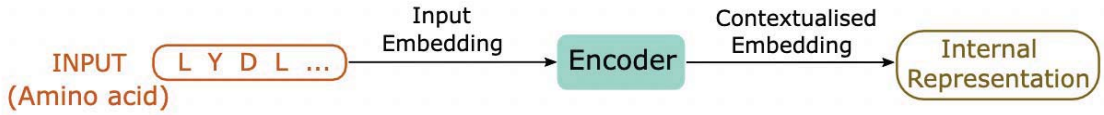


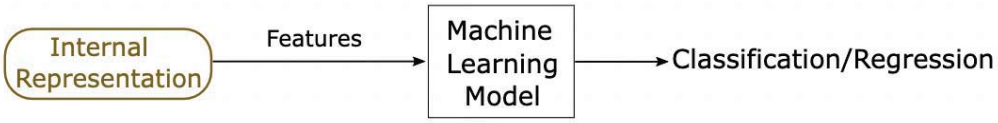
Encoder-decoder model



Encoder-decoder model



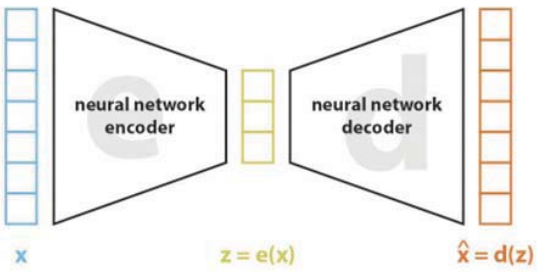




Elife 2023

Encoder-decoder model

Use encoder-decoder model to generate efficient representations

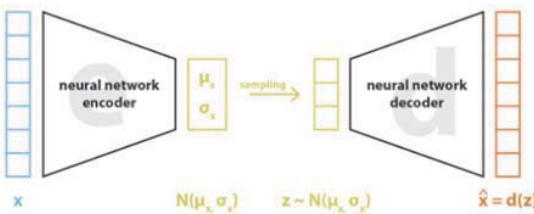


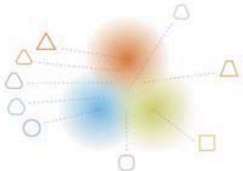
$$L = ||x - \hat{x}||^2$$
$$= ||x - d(e(x))||^2$$

Instead of encoding an input as a single point, encode it as a distribution over the latent space.

$$L = ||x - \hat{x}||^2 + KL[N(\mu_x, \sigma_x), N(0,1)]$$

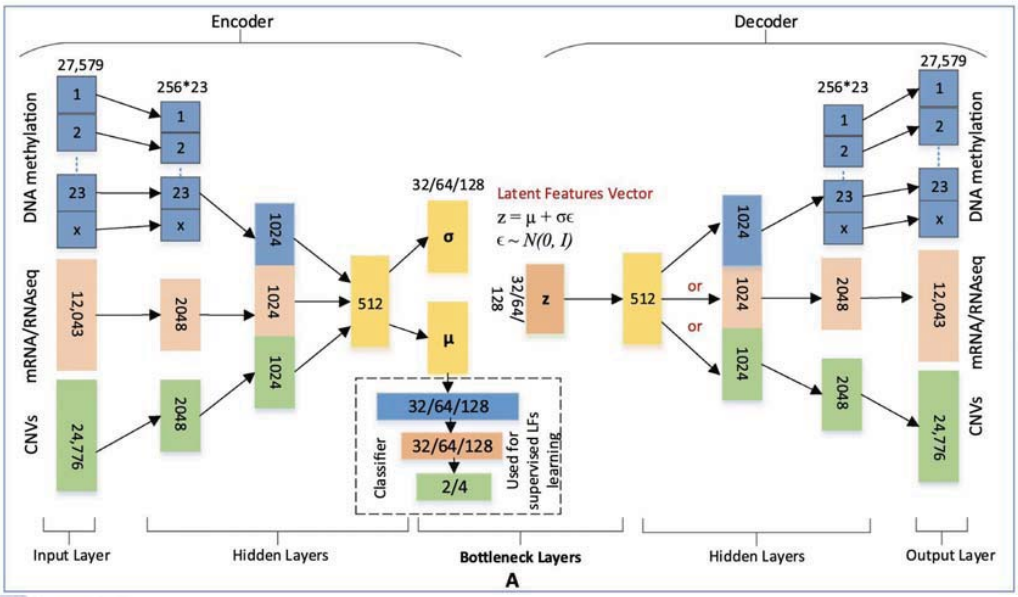
reconstruction — regularization





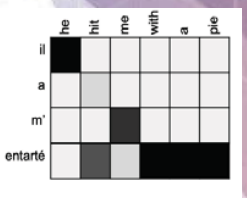
Encoder-decoder model

Integrated multi-omics analysis of ovarian cancer using variational autoencoders



Sci Rep 2021

Attention model



$$a_t = \sum_{i=1}^{N} \alpha_i^t h_i \in \mathbb{R}^h$$

Weighted sum of encoder hidden states based on the attention distribution

Softmax

$$\alpha^t = \text{softmax}(e^t) \in \mathbb{R}^N$$

Dot product

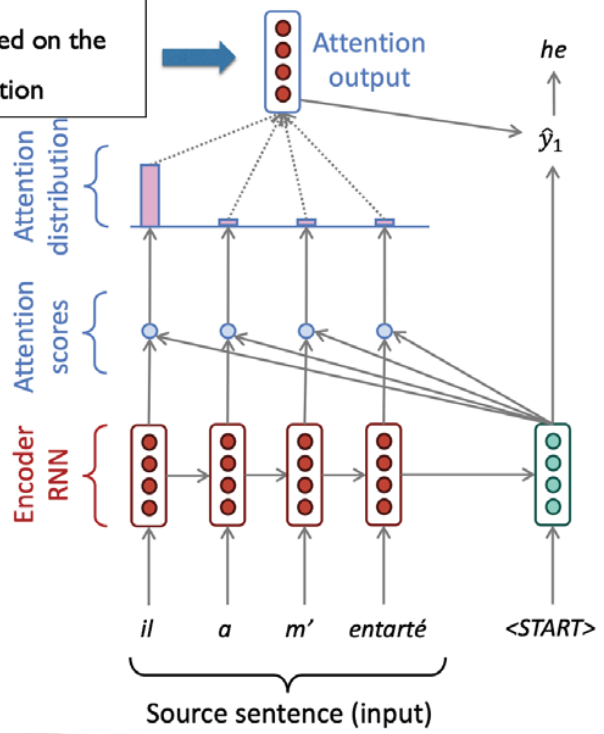
$$e^t = [s_t^T h_1, \ldots, s_t^T h_N] \in \mathbb{R}^N$$

Encoder hidden states

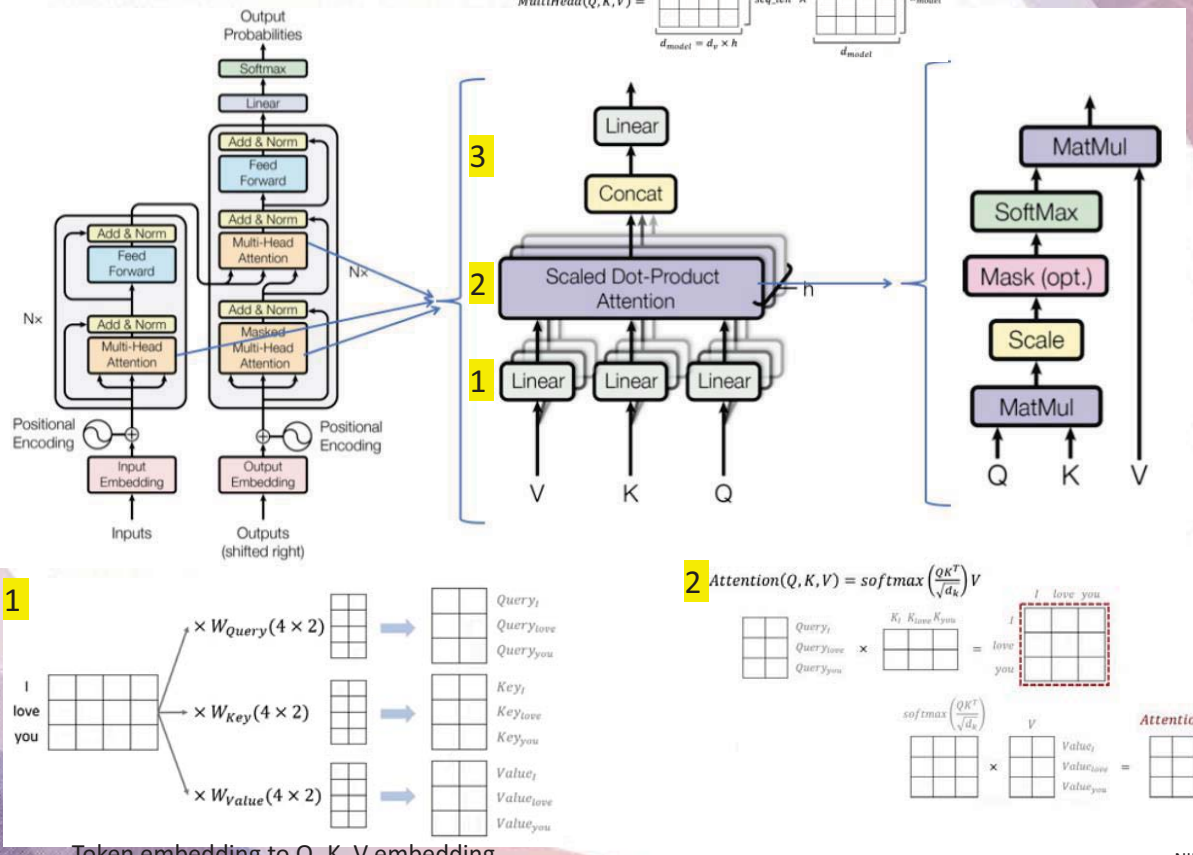
$$h_1, \ldots, h_N \in \mathbb{R}^h$$

Decoder hidden states

$$s_t \in \mathbb{R}^h$$

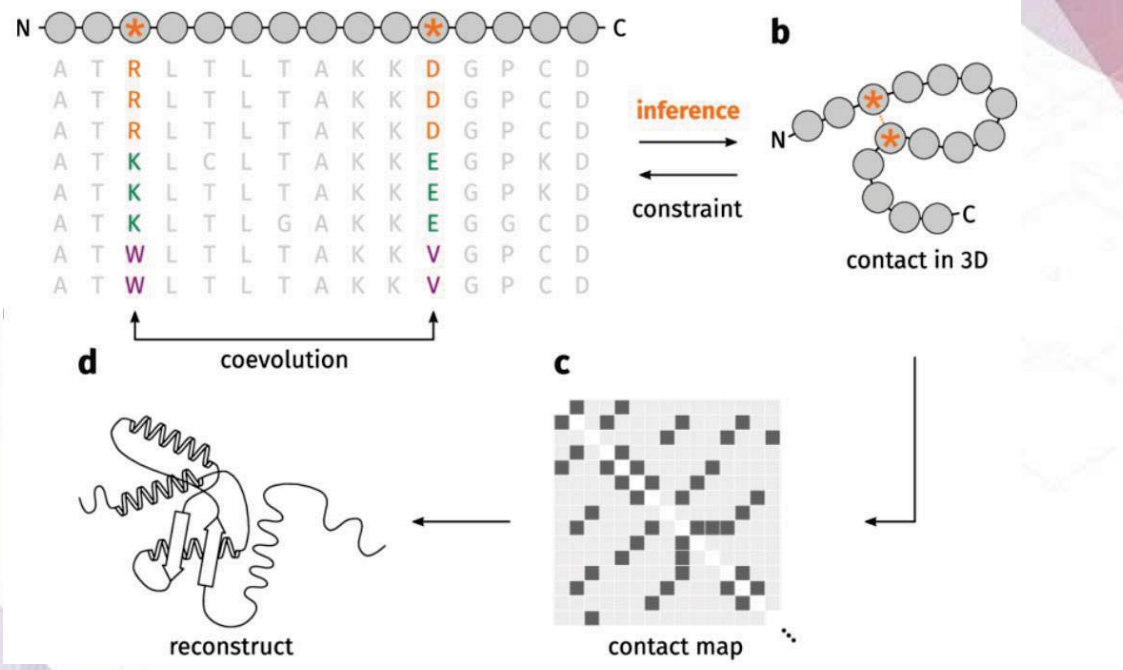


Self- attention model



NIPS, 2017

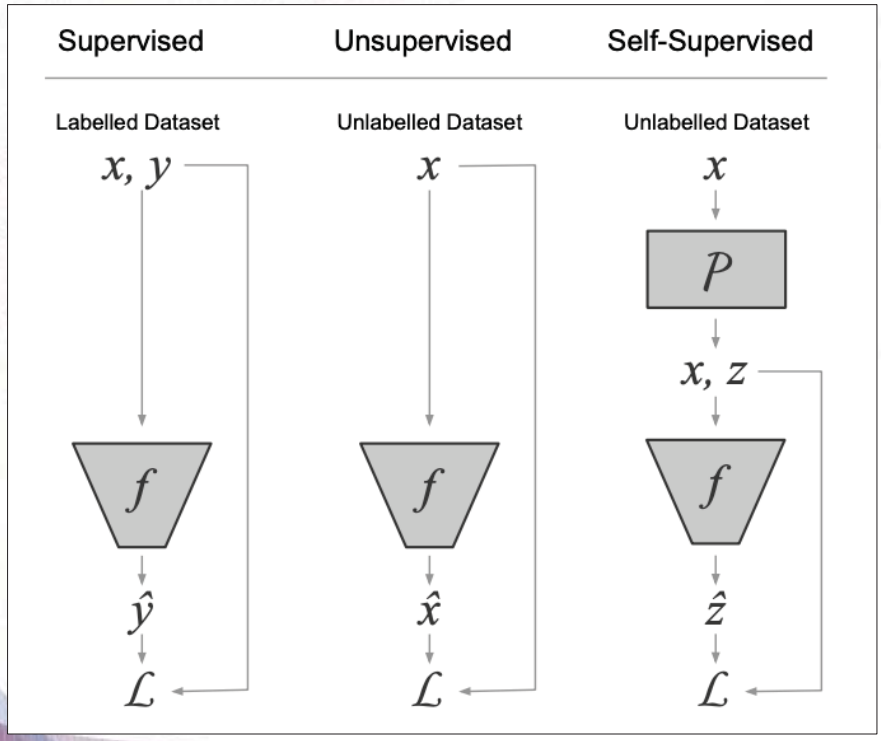
Self- attention model



30

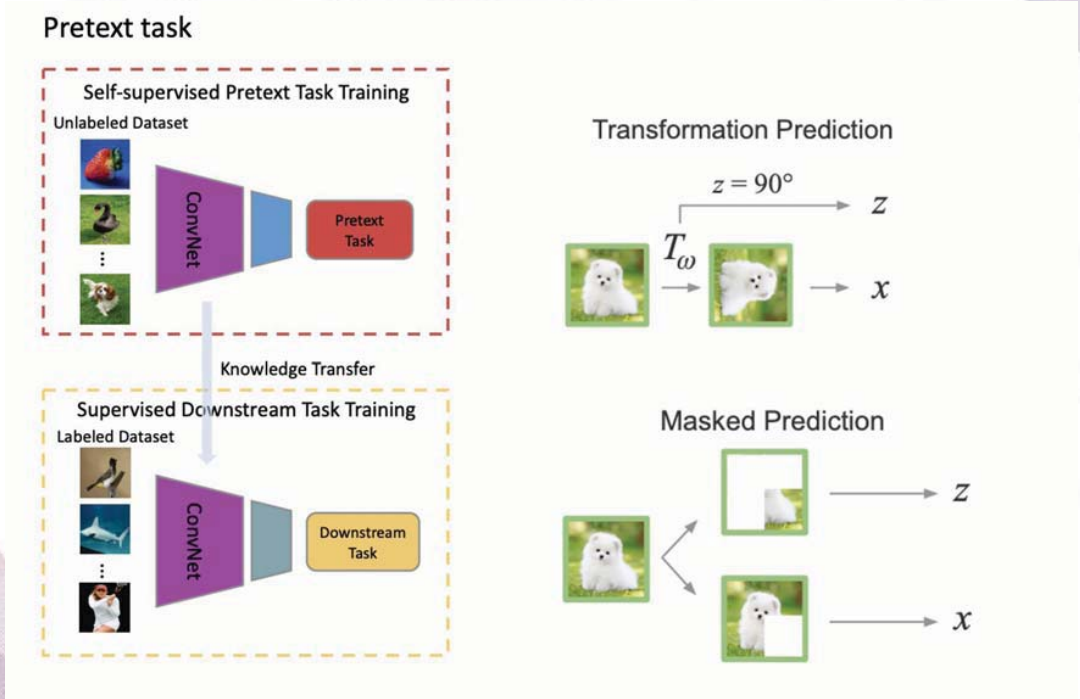
Nature 2021

Self-supervised learning



31

Self-supervised learning with image data



32

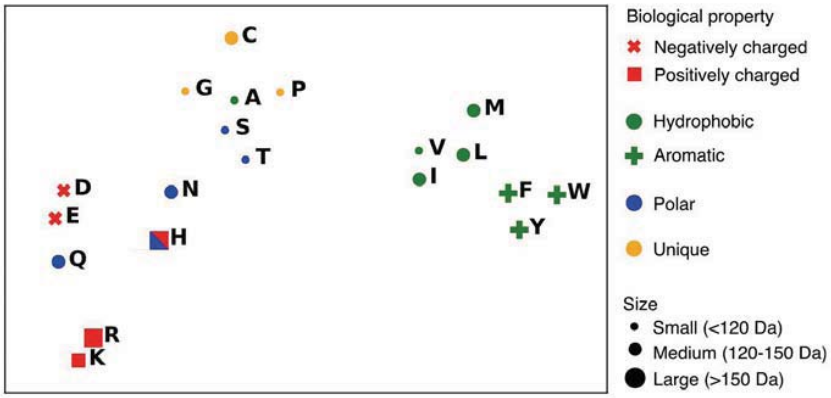
Self-supervised learning with protein sequences

Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences

Alexander Rives[a,b,1,2], Joshua Meier[a,1], Tom Sercu[a,1], Siddharth Goyal[a,1], Zeming Lin[b], Jason Liu[a], Demi Guo[c,3], Myle Ott[a], C. Lawrence Zitnick[a], Jerry Ma[d,e,3], and Rob Fergus[b]

[a]Facebook AI Research, New York, NY 10003; [b]Department of Computer Science, New York University, New York, NY 10012; [c]Harvard University, Cambridge, MA 02138; [d]Booth School of Business, University of Chicago, Chicago, IL 60637; and [e]Yale Law School, New Haven, CT 06511

Edited by David T. Jones, University College London, London, United Kingdom, and accepted by Editorial Board Member William H. Press December 16, 2020 (received for review August 6, 2020)



Trained a deep contextual language model on 86 billion amino acids across 250 million protein sequence

33

PNAS 2021

Self-supervised learning with protein sequences

	Model		Params	Training	ECE
(a)	Oracle				1
	Uniform Random				25
(b)	*n*-gram	4-gram		UR50/S	17.18
(c)	LSTM	Small	28.4 M	UR50/S	14.42
	LSTM	Large	113.4 M	UR50/S	13.54
(d)	Transformer	6-layer	42.6 M	UR50/S	11.79
	Transformer	12-layer	85.1 M	UR50/S	10.45
(e)	Transformer	34-layer	669.2 M	UR100	10.32
	Transformer	34-layer	669.2 M	UR50/S	8.54
	Transformer	34-layer	669.2 M	UR50/D	8.46
(f)	Transformer	10% data	669.2 M	UR50/S	10.99
	Transformer	1% data	669.2 M	UR50/S	15.01
	Transformer	0.1% data	669.2 M	UR50/S	17.50

Exponentiated cross entropy (ECE) metric, which is the exponential of the model's loss averaged per token

- the low-diversity dataset (UR100) uses the UniRef100 representative sequences
- the high-diversity sparse dataset (UR50/S) uses the UniRef50 representative sequences
- the high-diversity dense dataset (UR50/D) samples the UniRef100 sequences evenly across the UniRef50 clusters

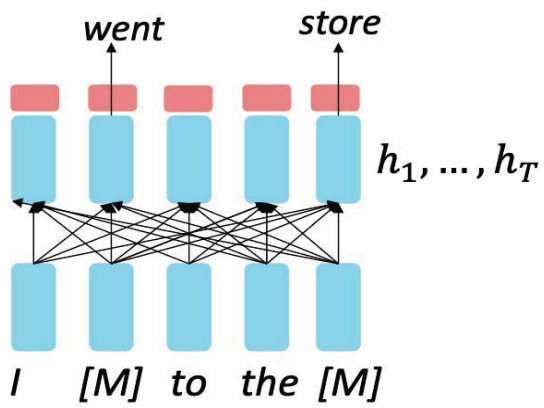
34

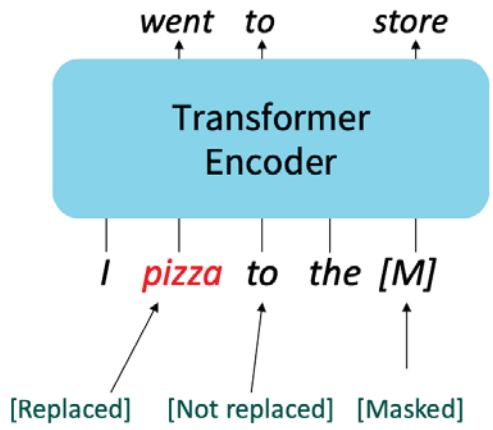
PNAS 2021

Masked Language Model

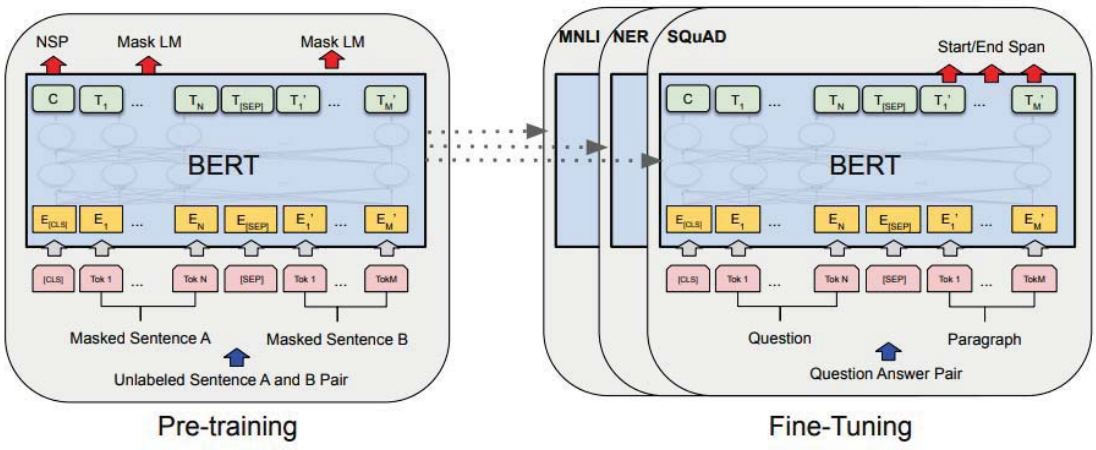
$\tilde{x}$ is the masked version of x, we're learning $p(x|\tilde{x})$

- replace some fraction of words in the input with a special [MASK] token
- predict these words.



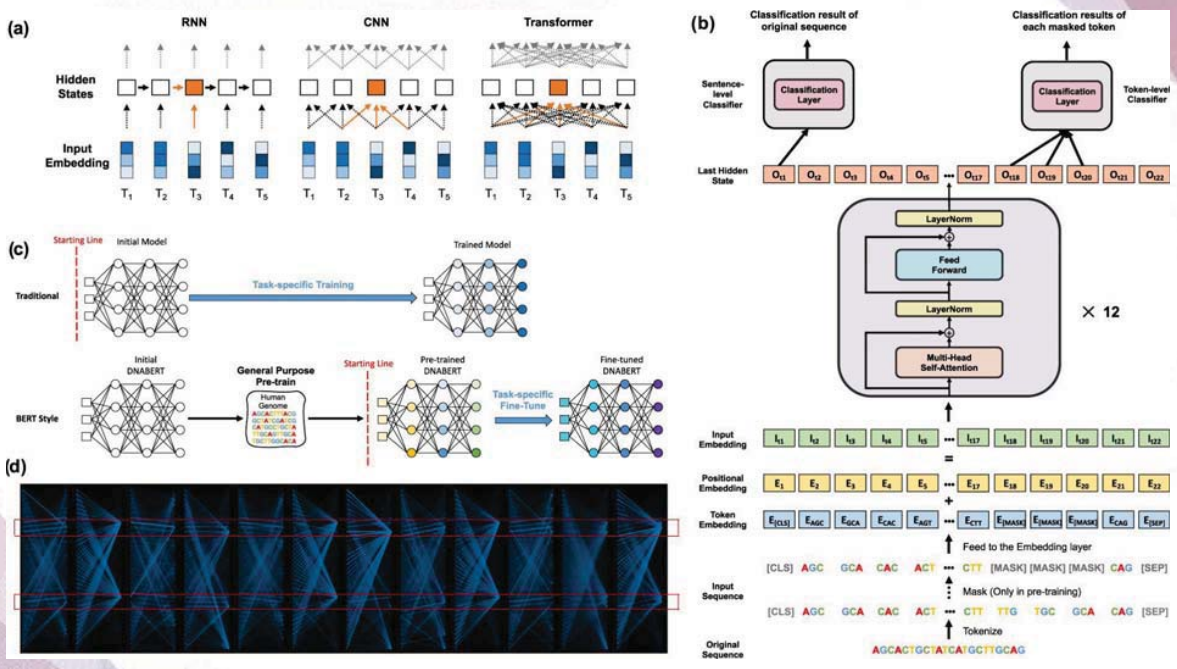


Pre-training of deep bidirectional transformers for language understanding (BERT)

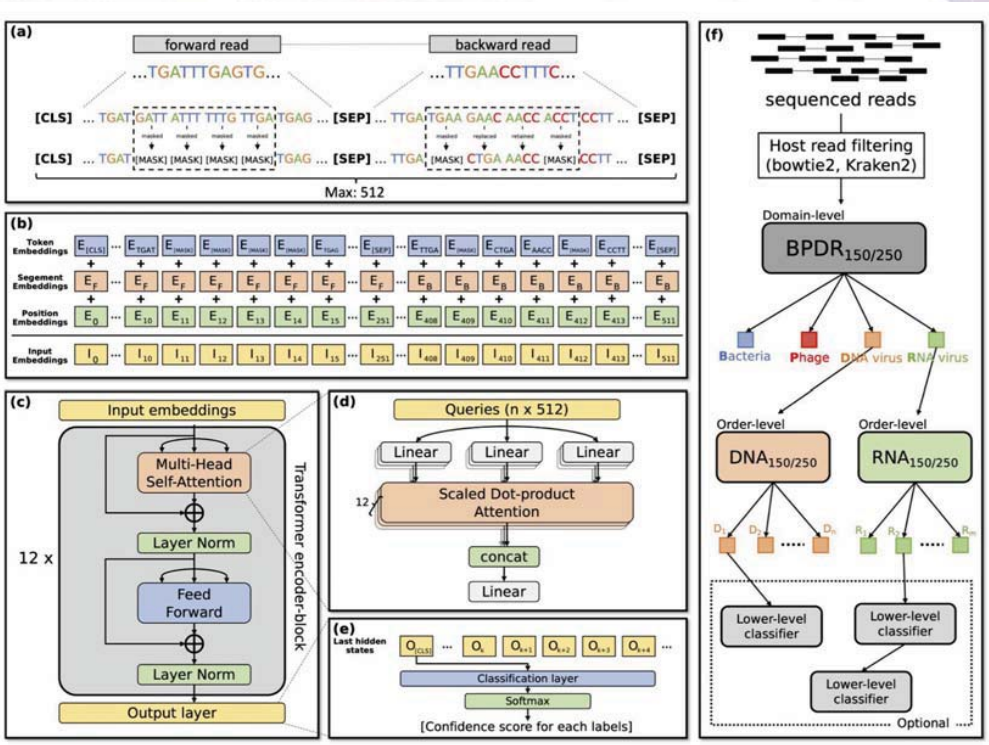


- Apart from output layers, the same architectures are used in both pre-training and fine-tuning
- The same pre-trained model parameters are used to initialize models for different down-stream tasks
- During fine-tuning, all parameters are fine-tuned.

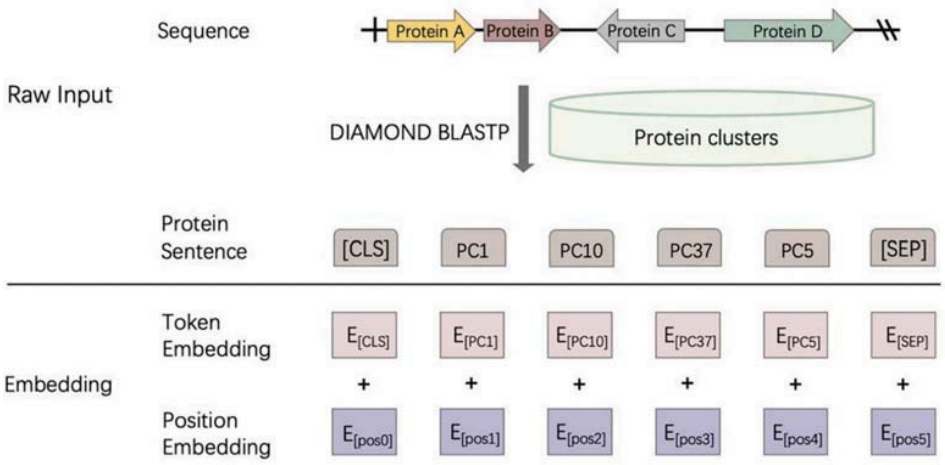
Pre-trained bidirectional encoder representation for DNA



Identifying eukaryotic viruses using BERT and metagenome sequencing



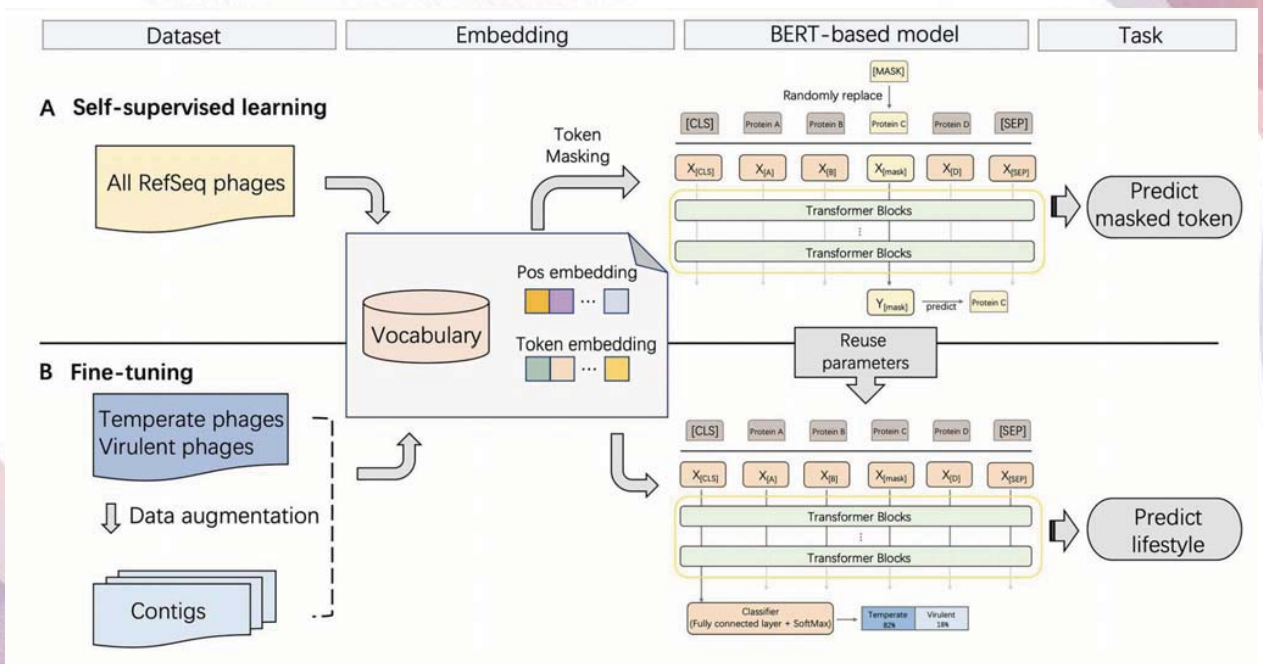
Predicting the lifestyle for bacteriophages using BERT



- Used all the phage proteins from the RefSeq database
- Clustered to generate protein token ID

Briefings in Bioinformatics 2023

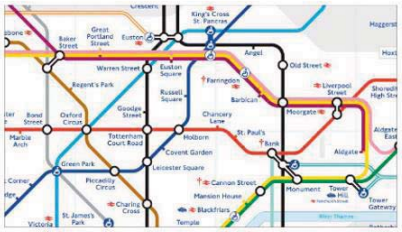
Predicting the lifestyle for bacteriophages using BERT



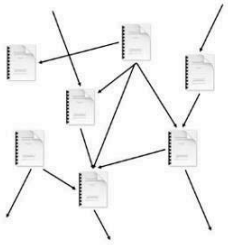
Briefings in Bioinformatics, 2023

Various networks (Graphs)

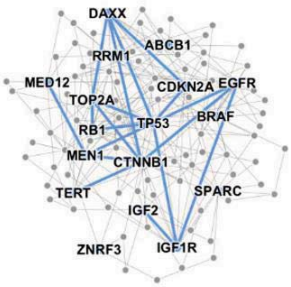
Social Network



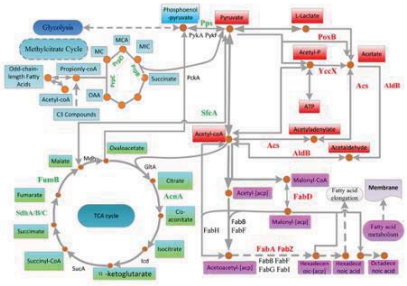
Subway map



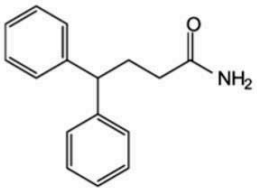
Citation network



Gene network



Metabolic pathway



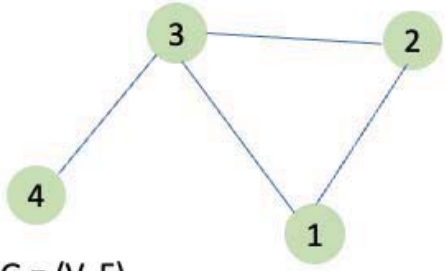
Molecules

- Complex domains have relational structures, which can be represented as graphs

*

Graph representation

- Graph is a data structure that consists of two components: nodes and links
- We use graph data structure to represent relations (links) between entities (nodes)
- Relations can be represented by using adjacency matrix



G = (V, E)

V = {1, 2, 3, 4}; E = {(4, 3), (3, 2), (1, 2), (1, 3)}

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

Adjacency matrix

*

Graph representation

Graph representation is defined as follows

$$G = (V, E, R, T)$$

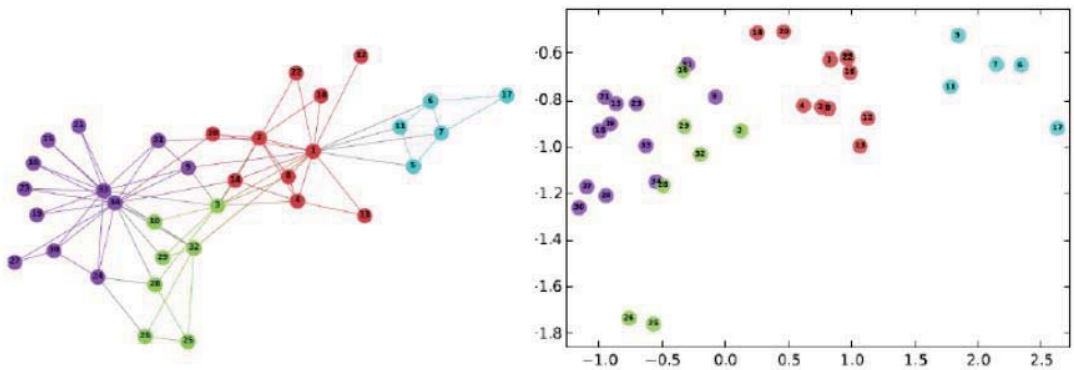
$v_i \in V$	Nodes with node types
$e_{i,j} \in E$	Edges between node i and node j
$T(v_i)$	Nodes type
$r_{i,j} \in R$	Relation type

Nodes and edges can have features

43

*

Graph representation learning



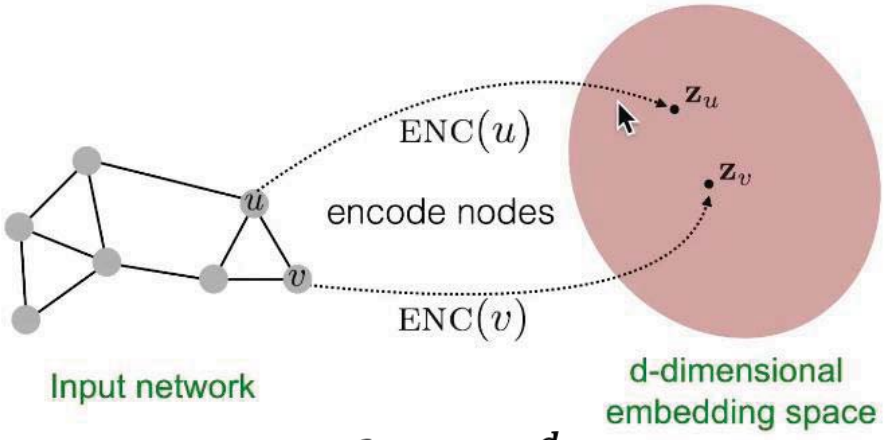
Graph 2-dimentional embedding

- A graph does not exist in Euclidean space (hard to represent by any coordinate system)
- Graph representation learning (embedding) transforms nodes and edges including their features into vector space that maximally preserve properties of the graph structure and information

44

*

Graph representation learning (node embedding)



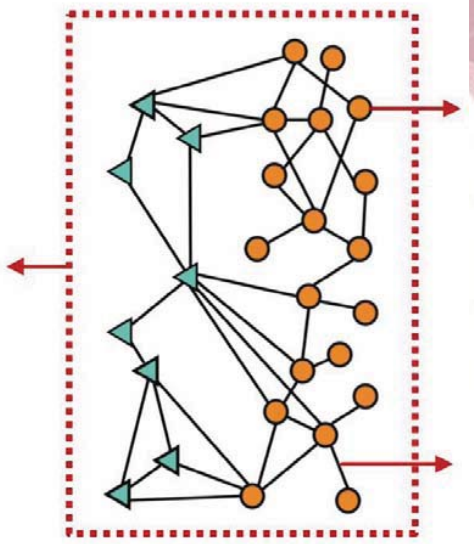
$$f: u \rightarrow \mathbb{R}^d$$

Node embedding maps nodes to d-dimensional **embeddings** such that similar nodes in the network are embedded close together

*

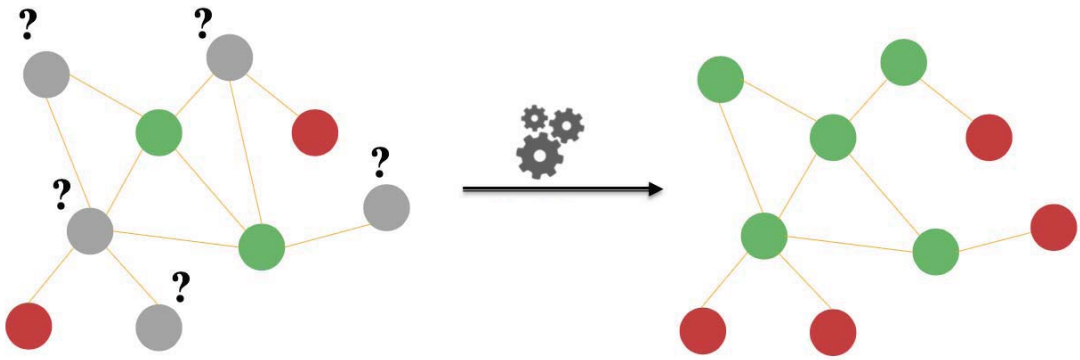
Tasks based on the graph structure

- Node classification
  → predict a type of a given node
- Link prediction
  → predict whether two nodes are linked
- Community detection
  → identify densely linked clusters of nodes
- Network similarity
  → predict how similar two (sub)graphs are
- Graph-level prediction/classification



*

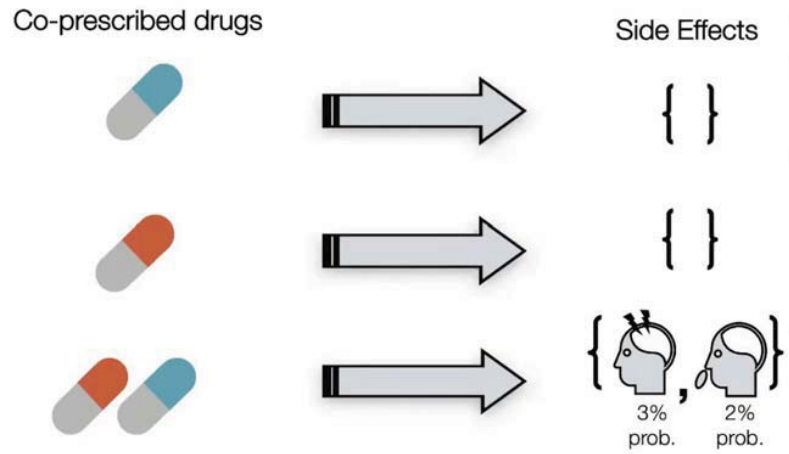
Node classification



47

*

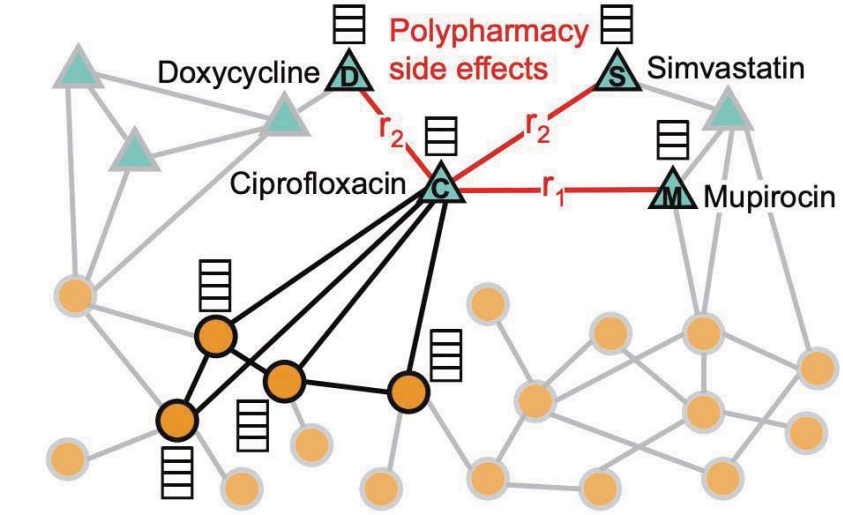
Link prediction



48

*

Link prediction



Drug Protein Node feature vector

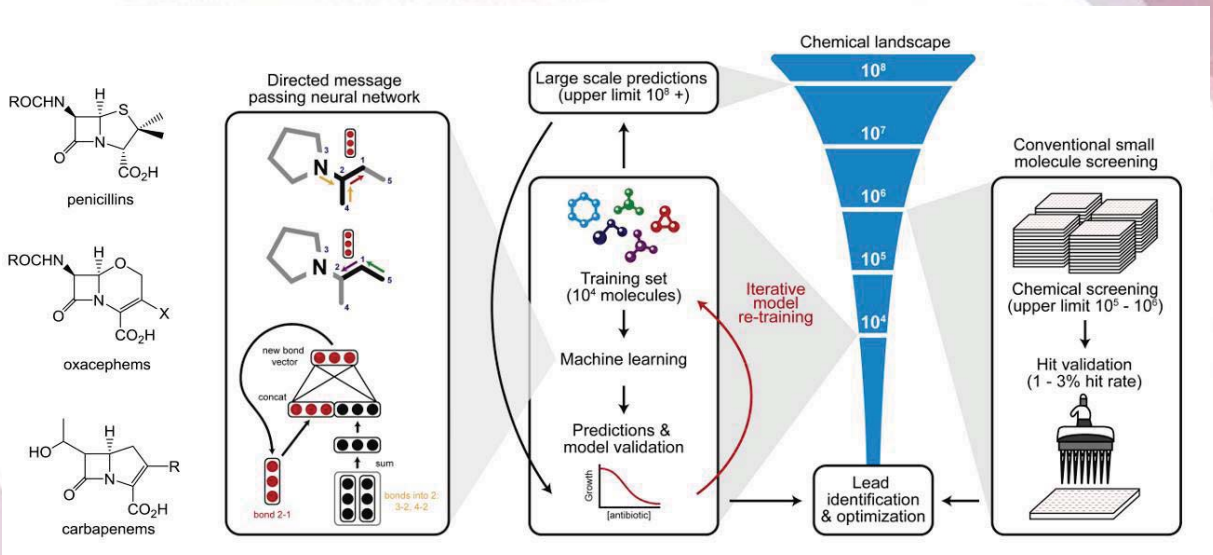
r_1 Gastrointestinal bleed side effect Drug-protein interaction

r_2 Bradycardia side effect Protein-protein interaction

49

*

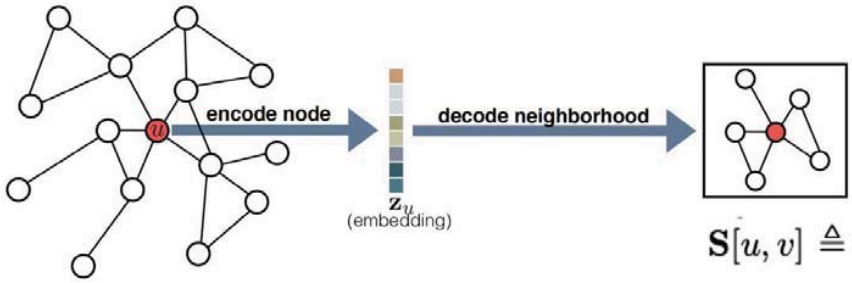
Graph-level prediction/classification



50

Cell, 2020

Graph Neural Network: how to do embedding



$$\mathbf{S}[u,v] \triangleq \mathbf{A}[u,v]$$

$$\text{ENC} : \mathcal{V} \rightarrow \mathbb{R}^d.$$

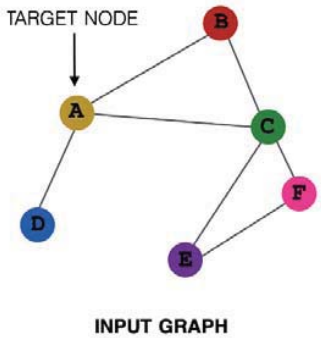
$$\text{DEC} : \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R}^+$$

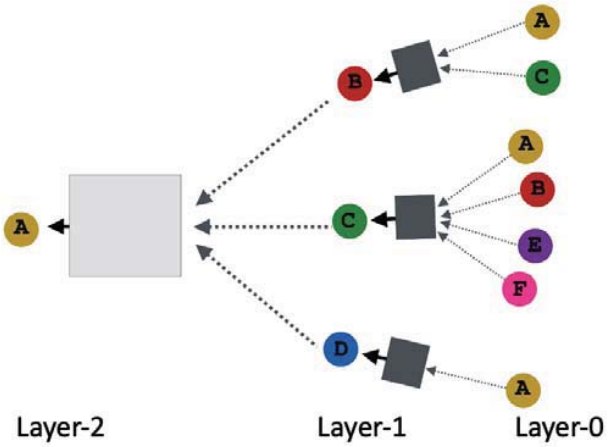
$$\text{DEC}(\text{ENC}(u), \text{ENC}(v)) = \text{DEC}(\mathbf{z}_u, \mathbf{z}_v) \approx \mathbf{z}_v^{\mathsf{T}} \mathbf{z}_u$$

$$\mathcal{L} = \sum_{(u,v)\in\mathcal{D}} \ell\left(\text{DEC}(\mathbf{z}_u, \mathbf{z}_v), \mathbf{S}[u,v]\right) \approx \sum_{(u,v)\in\mathcal{D}} \text{DEC}(\mathbf{z}_u, \mathbf{z}_v) \cdot \mathbf{S}[u,v].$$

51

Graph Neural Network: aggregation



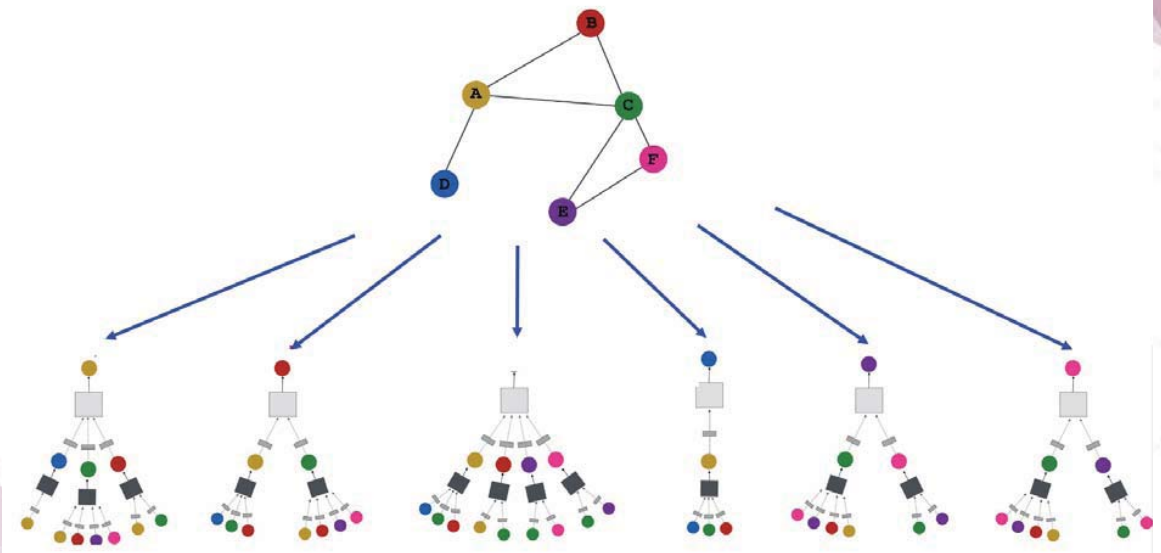


- Key idea is to generate node embeddings based on local network neighborhoods

→ nodes can aggregate information from their neighbors using neural network

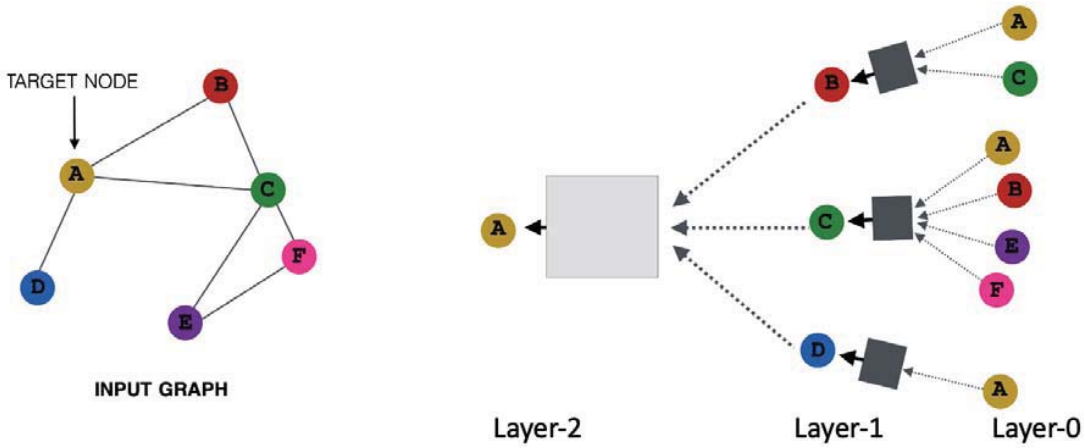
52

Graph Neural Network: aggregation



53

Graph Neural Network: aggregation



$$\mathbf{h}_u^{(k+1)} = \text{UPDATE}^{(k)}\left(\mathbf{h}_u^{(k)}, \text{AGGREGATE}^{(k)}(\{\mathbf{h}_v^{(k)}, \forall v \in \mathcal{N}(u)\})\right)$$

$$\mathbf{h}_u^{(k)} = \sigma\left(\mathbf{W}_{\text{self}}^{(k)}\mathbf{h}_u^{(k-1)} + \mathbf{W}_{\text{neigh}}^{(k)} \sum_{v \in \mathcal{N}(u)} \mathbf{h}_v^{(k-1)} + \mathbf{b}^{(k)}\right)$$

Graph Neural Network: aggregation

$$\mathbf{h}_u^{(k)} = \sigma\left(\mathbf{W}_{\text{self}}^{(k)}\mathbf{h}_u^{(k-1)} + \mathbf{W}_{\text{neigh}}^{(k)} \sum_{v \in \mathcal{N}(u)} \mathbf{h}_v^{(k-1)} + \mathbf{b}^{(k)}\right)$$

$$\mathbf{W}_{\text{self}}^{(k)}, \mathbf{W}_{\text{neigh}}^{(k)} \in \mathbb{R}^{d^{(k)} \times d^{(k-1)}}$$

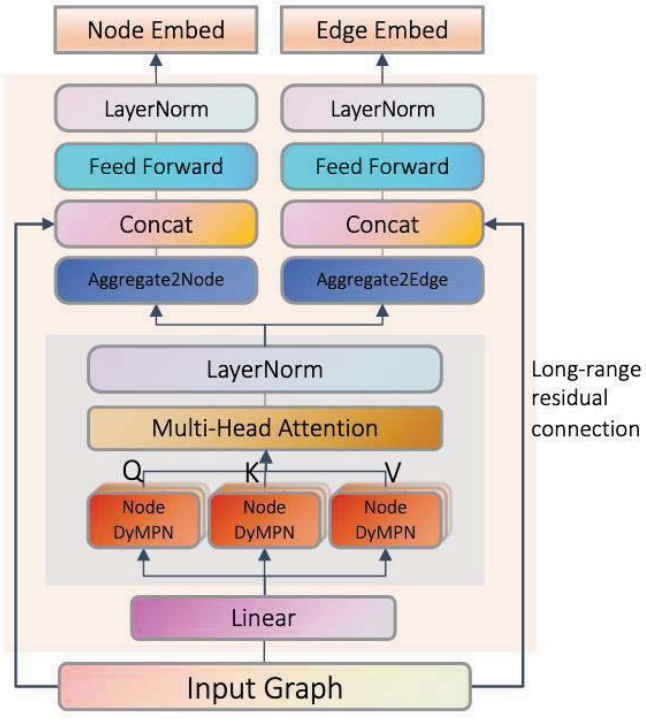
σ an element-wise non-linearity function

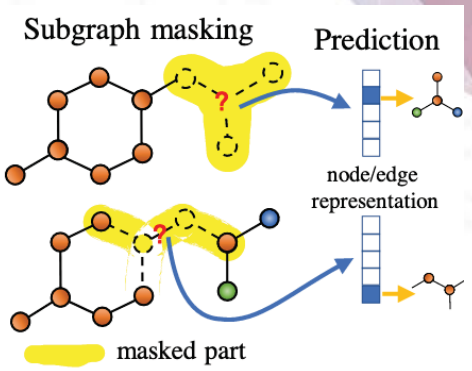
$$\mathbf{h}_u^{(0)} = \mathbf{x}_u, \forall u \in \mathcal{V}.$$

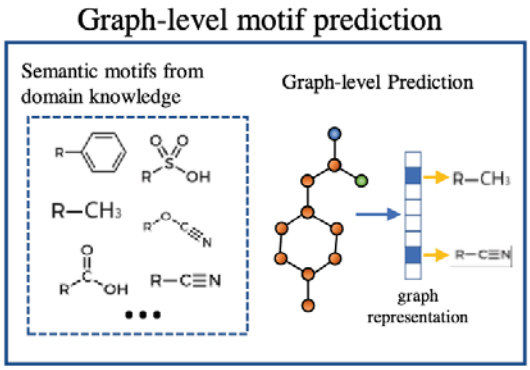
$$\mathbf{z}_u = \mathbf{h}_u^{(K)}, \forall u \in \mathcal{V}$$

55

Self-supervised graph transformer on large-scale molecular data

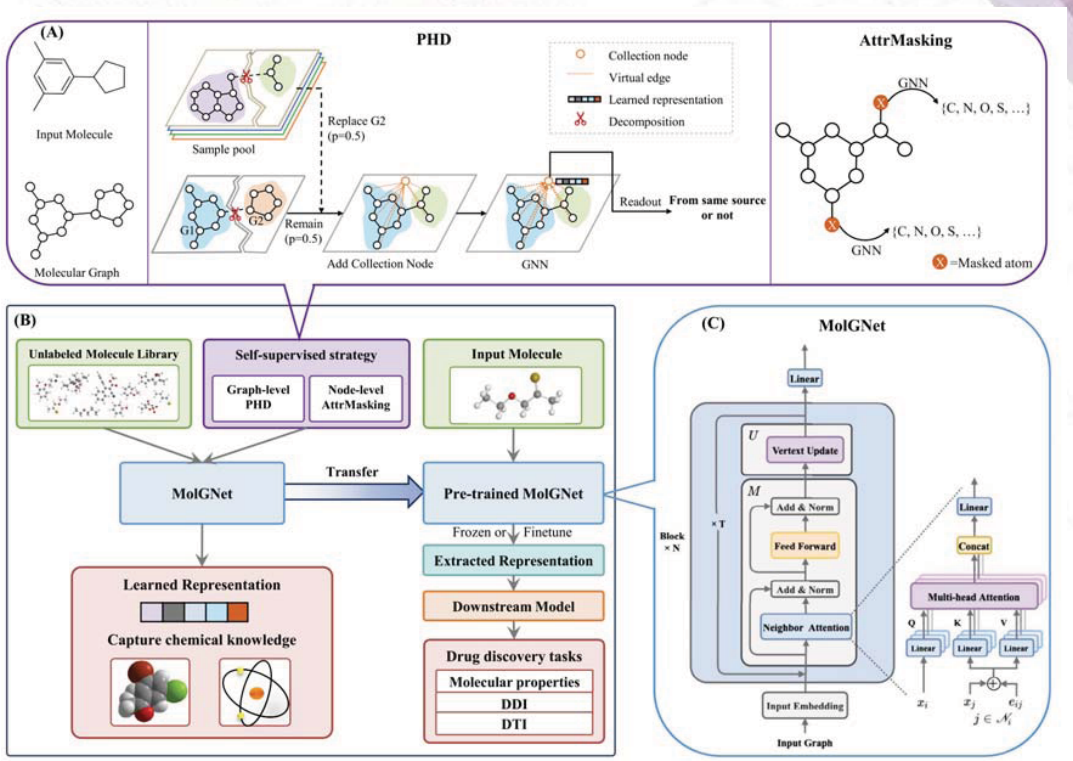






NeurIPS 2020

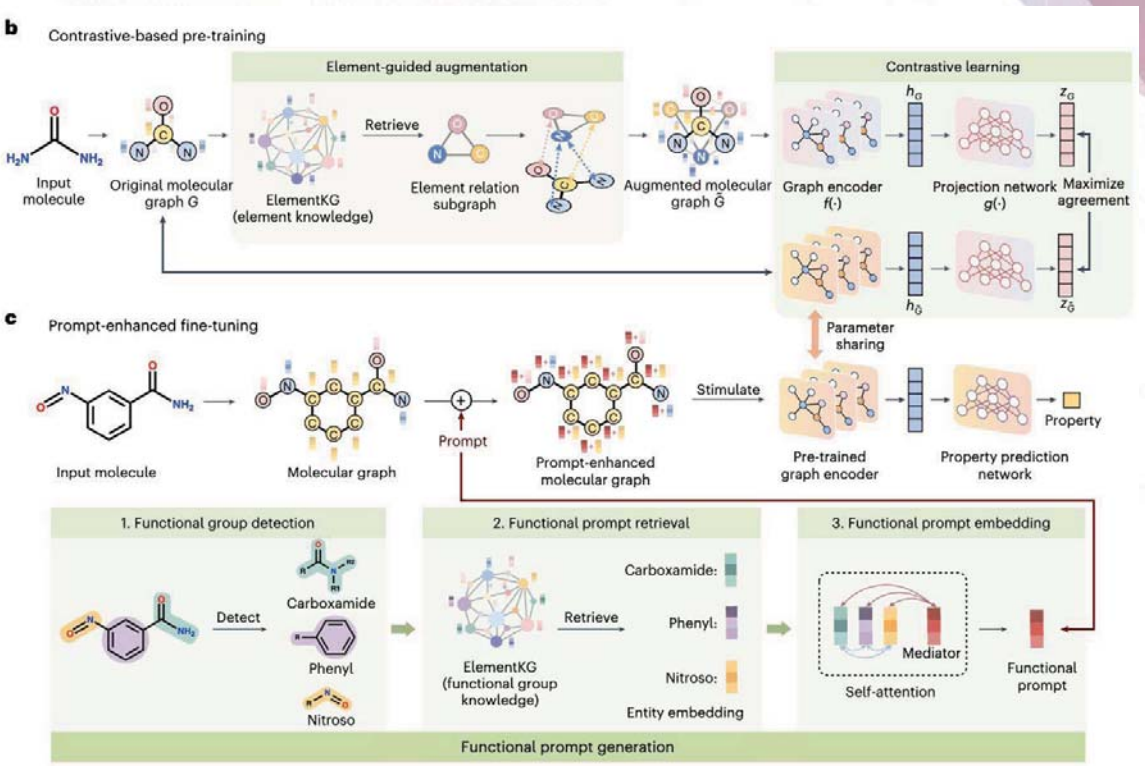
Self-supervised framework for learning global representation



57

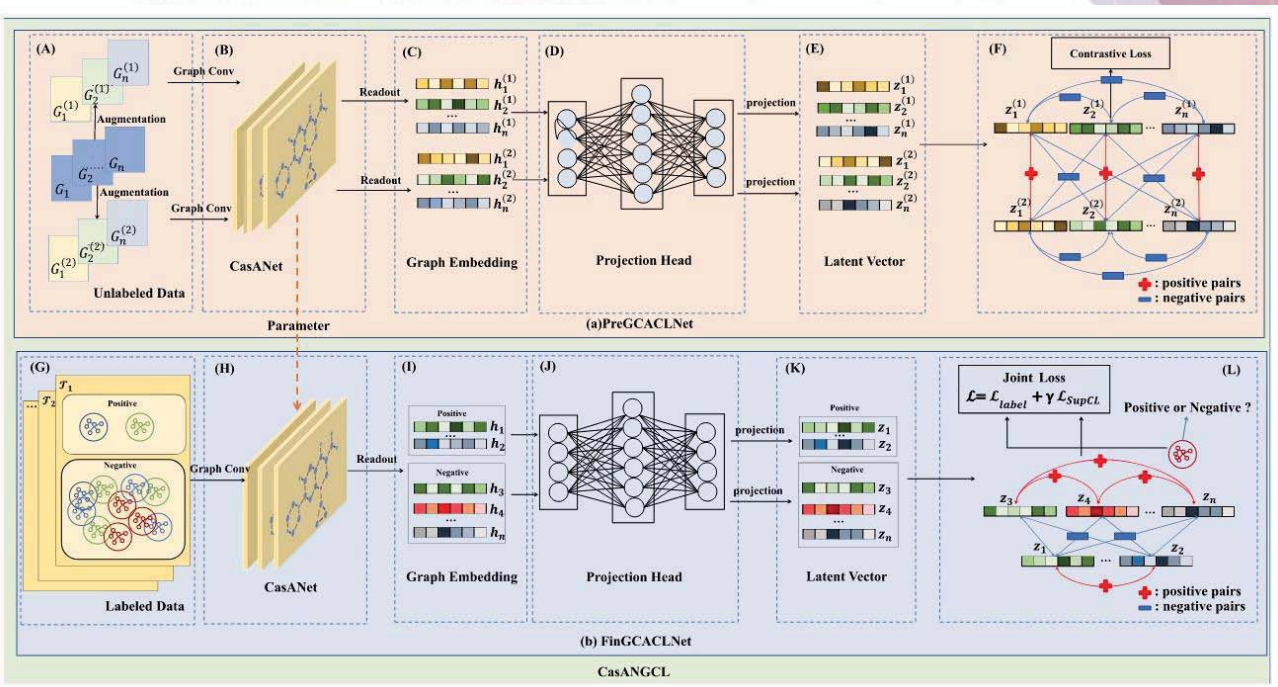
Briefings in Bioinformatics, 2021

Knowledge graph-enhanced molecular contrastive learning



Nature Machine Intelligence, 2023

Graph contrastive learning for molecular property prediction



Briefings in Bioinformatics, 2023