Pusan National University

Computer Science and Engineering

Technical Report 2025-09

PLaNet-X: Protein sequence-based Ligand binding-site and binding-affinity prediction via the deep convolutional network

202255512 김다현

202255552 박주은

202255565 안수현

지도교수 송길태

목 차

1. 과제 배경 및 목표

1.1. 과제 배경

단백질과 리간드의 결합은 신약 개발에서 핵심적인 연구 대상이다. 단백질은 생명체를 구성하며 모든 생명 활동에 관여하는 기본적인 요소로, 아미노산들이 펩타이드 결합으로 연결된 분자이다. 여기서 펩타이드 결합(peptide bond)이란 두 개의 아미노산이 만나 물 분자(H_2O) 하나가 빠져나가면서 형성되는 화학적 공유 결합이며, 아미노산(amino acid)은 단백질을 이루는 기본 단위 분자를 의미한다. 반면 단백질 구조를 설명할 때 주로 사용하는 용어는 잔기(residue)이다. 이는 아미노산이 펩타이드 결합을 통해 서로 연결될 때, 물 분자가 빠져나가고 남은 구조가 단백질 사슬에 '잔류'한다는 의미에서 붙여진 이름이다. 따라서 단백질 내부에서 우리가 다루는 실질적인 분석 단위는 아미노산 잔기이다.

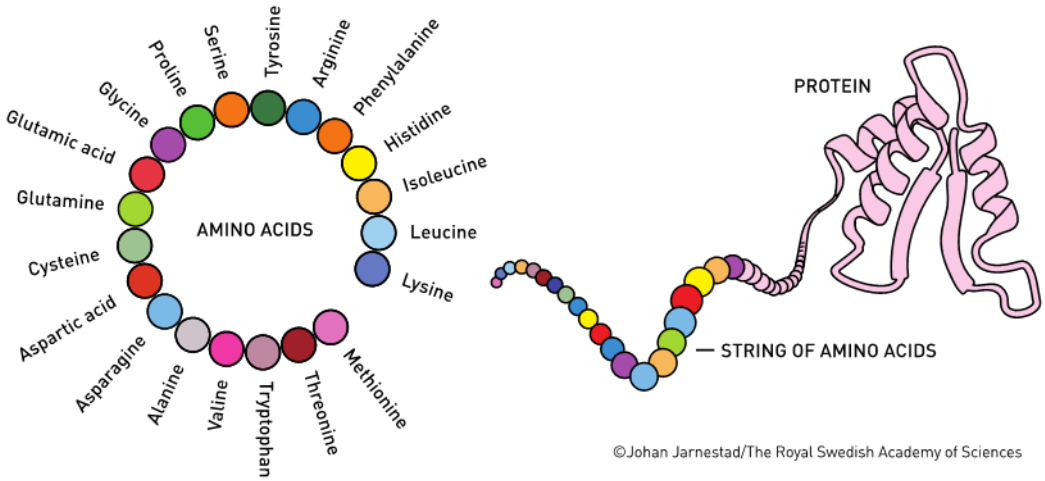


Figure 1. 단백질은 20 가지의 아미노산이 3 차원 구조로 복잡하게 얽히며 형성된다.

단백질은 아미노산이 길게 연결된 사슬이 접히면서 3 차원 구조를 형성한다. 리간드는 단백질에 결합하는 작은 분자(약물 후보 물질, 호르몬, 신경전달물질, 대사산물 등)이고, 리간드가 결합하는

단백질의 특정 위치를 결합 부위(binding site) 라고 한다. 구조 기반 신약 개발은 도킹(Docking)을 통해 이 결합 부위 근처를 탐색 공간으로 설정하고, 리간드의 다양한 결합 자세(포즈) 를 생성·채점하여 가장 안정적인 상호작용을 예측한다. 따라서 결합 부위를 올바르게 식별하는 일은 도킹의 탐색 범위를 정확히 좁혀 계산 비용을 줄이고, 정확한 포즈 예측 확률을 높이는 핵심 단계다.

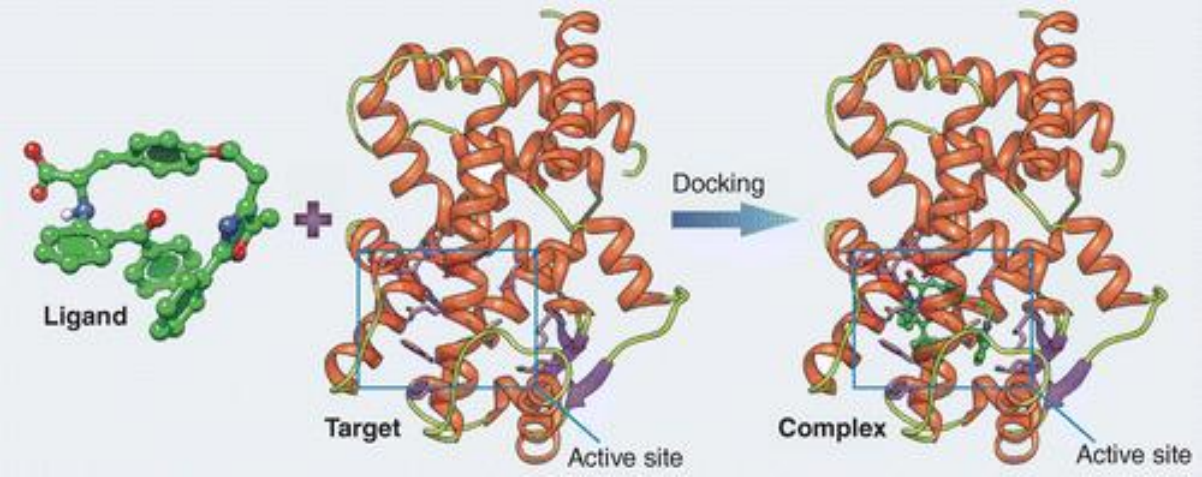


Figure 2. 도킹의 원리. 리간드와 단백질(타겟)이 그 결합 부위(사각형)에서 상호작용하는 모습을 보여준다.

이러한 상호작용이 얼마나 단단하고 안정적으로 유지되는지를 나타내는 값이 결합 친화도(binding affinity)이다. 친화도가 높을수록 리간드가 단백질에 강하게 결합해 효과를 낼 가능성이 크고, 친화도가 낮을수록 쉽게 떨어져 약물로서 효용성이 떨어진다.

기존에는 단백질과 리간드의 결합 부위를 예측할 때 단백질의 서열(sequence) 데이터보다 3 차원(3D) 데이터를 활용해 이러한 결합 부위와 친화도를 예측하는 연구가 주로 이루어졌으나, 구조 데이터는 확보가 어렵고 시간과 비용이 많이 든다는 한계가 있었다. 따라서 최근에는 단백질의 아미노산 서열만으로 결합 부위와 포켓을 추정하고, 리간드와의 친화도를 예측하려는 서열 기반 접근법이 활발히 연구되고 있다. 최근에는 딥러닝 기술의 발전으로, 단백질 아미노산 서열 정보만으로도 의미있는 결합

정보를 추출하는 것이 가능해졌다. 특히, ProtTrans, ESM 등의 모델을 활용한 단백질 서열 임베딩 기법을 통해 3D 구조 없이도 생화학적 의미를 포착할 수 있는 기반이 마련되고 있다.

1.2. 과제 목표

본 과제의 목표는 단백질과 리간드 간의 상호작용을 정밀하게 이해하기 위해, 단백질과 리간드 간의 결합 부위(binding site)와 결합 친화도(binding affinity)를 예측할 수 있는 딥러닝 기반의 예측 모델을 개발하는 데 있다. 이렇게 개발된 모델을 쉽게 활용할 수 있도록 웹페이지 형태의 인터페이스로 구현하여, 사용자들의 접근성과 편의성을 향상시키는 것을 목표로 한다. 특히, 실제 사용자 환경에서의 활용성을 고려하여, 서열 기반 모델 구현을 목표로 하였다.

2. 요구사항 분석

2.1. 기능적 요구사항

(1) 사용자 로그인

- 사용자가 웹 페이지 인터페이스에 로그인할 수 있어야 한다.

(2) 사용자 입력

- 사용자가 단백질 서열(sequence) 정보와 리간드 정보를 웹 페이지 인터페이스 상에서 입력할 수 있어야 한다.

-

(3) 결과값 처리

- 사용자가 입력한 단백질의 결합 부위(binding site)를 결과값으로 도출해야 한다.

- 사용자가 입력한 단백질과 리간드 사이의 결합 친화도(binding affinity)를 결과값으로 도출해야 한다.

(4) 결과값 시각화

- 웹 페이지 인터페이스를 통해 단백질과 리간드 사이의 결합 부위를 시각적으로 보이게 해야한다.

(5) 결과값 저장

- 단백질과 리간드 사이의 결합 부위 및 결합 친화도 값을 저장할 수 있어야 한다.

(6) 저장된 값 보기

- 저장해두었던 단백질과 리간드 사이의 결합 부위 및 결합 친화도을 한 번에 볼 수 있어야 한다.

2.2. 비기능적 요구사항

(1) 성능

- 모델 예측 속도 : 사용자의 입력에 대해 30 초 이내에 결과값을 도출해내야 한다.

- 시스템 응답 시간 : 모델 예측 후, 웹 페이지 인터페이스에 1 초 이내에 확인 가능해야한다.

(2) 보안성

- 로그인 : 로그인되지 않은 사용자는 예측 모델을 사용할 수 없다.

- 정보 보호 : 다른 사람이 예측을 시도하거나, 저장한 단백질 리간드 쌍에 대한 정보를 타인은 절대 볼 수 없다.

(3) 안정성

- 예외 처리 : 비정상적인 입력(ex. 너무 긴 서열, 비표준 아미노산)에 대해 에러 메시지를 반환하고, 예측을 하지 않는다.

- 자원 제어 : 예측 요청이 과도하게 쌓일 경우, 서버 과부하를 막기 위해 예측을 제한한다.

(4) 가용성

- 접근성: 사용자는 다른 프로그램의 설치 없이도, 웹 브라우저를 통해 웹 페이지 인터페이스에 접근 가능해야 한다.
- 서비스 시간: 시스템은 24 시간동안 운영 되어야 하며, 점검 시에는 이를 미리 공지 해야한다.

3. 개발 환경 및 사용 기술

3.1. 개발 환경

(1) 프레임워크: PyTorch

(2) 데이터셋:

- 학습용:
    - binding site 예측 : PDBbind(2020), scPDB
    - binding affinity 예측 : PDBbind(2020) / PDBbind(2016)
- 테스트용
    - binding site 예측 : COACH420, HOLO4K
    - binding affinity 예측 : Core-2016, CSAR-HiQ_36, CSAR-HiQ_51 / CASF-2013, Core-2016, CSAR-HiQ_36, CSAR-HiQ_51

(3) 전처리 및 임베딩 도구: RDKit, Openbabel, ESM-2, ChemBERTa, Biopython

(4) 시각화 도구: TensorBoard, NGL Viewer

(5) 개발/운영 보조 도구: Docker, ngrok

3.2. 사용 기술

(1) 모델 개발

- binding-site : CNN
- binding-affinity : CNN + cross-attention

(2) 웹 페이지 및 API

- 프론트엔드: React
- 백엔드 서버: Flask
- API 통신 방식: RESTful API

(3) 인프라 및 데이터 저장

- AWS EC2, AWS S3, MySQL
- Nginx

4. 현실적 제약 사항과 이에 대한 방안

4.1. 물리화학적 요인 반영 어려움

수소 결합, 소수성 상호작용, 원자 간 거리 및 전하 분포 등 실제 결합에 영향을 주는 세밀한 물리화학적 요인을 모델에 반영하여 실제 결합 환경에서의 상호작용을 정밀하게 예측하는 데 어려움이 있다. 초기에는 이러한 물리화학적 특성들을 반영하지 않고 모델을 구성할 계획이었으나, 결합 친화도 예측 정확도 향상을 위해 이를 입력 피처에 포함하기로 하였다. 다음은 단백질과 리간드에 각각 반영한 물리화학적 특성이다.

단백질 (Residue 단위) 특성 : 산성 여부, 염기성 여부, 중성 여부, 극성 여부, 소수성 지수, 분자량

리간드(Global 분자 단위) 특성 :

- MolWt(분자량) : 분자의 전체 크기
- MolLogP(소수성 지수) : 지용성과 친수성의 균형
- TPSA(극성 표면적) : 막 투과성과 용해도에 영향
- NumRotatableBonds(회전 가능한 결합 수) : 분자의 유연성
- HeavyAtomCount(수소를 제외한 원자 수) : 분자의 구조적 크기

- FractionCSP3(sp^3 탄소 비율) : 구조적 복잡성
- NumHDonors(수소 결합 공여자 수) : 수소 결합 가능성
- NumHAcceptors(수소 결합 수용자 수) : 수소 결합 가능성
- RingCount(고리 구조 수) : 분자의 안정성과 구조 특성
- MolMR(몰 굴절률) : 분자의 부피 및 극성화율
-

4.2. Induced Fit 효과 미반영

실제 단백질-리간드 결합 과정에서는 리간드의 결합에 따라 단백질 구조가 유연하게 변형되는 Induced Fit 현상이 발생할 수 있다. 그러나 본 모델은 단백질의 고정된 서열 정보를 기반으로 예측을 수행하기 때문에 결합 시 발생하는 구조적 재배열이나 유연성 변화를 반영하지 못하여 실제 결합 상황 및 결합 친화도 예측에 차이를 불러올 수 있다.

4.3. pH, 이온 농도 등 생체 환경 요소 고려 어려움

실제 생체 내 단백질 리간드 결합은 pH, 이온 농도, 온도, 수용성 환경 등 다양한 생리학적 조건의 영향을 받는다. 따라서 본 모델 또한 생체 환경 요소를 고려하려고 했으나, pH, 이온 농도 등의 요소는 모든 PDB 데이터셋 파일에는 포함되어 있지 않아 모델에 직접 반영이 어려울 것으로 판단하였다.

5. 과제 추진 계획

<table>
<tr><th colspan="2">5 월</th><th colspan="4">6 월</th><th colspan="4">7 월</th><th colspan="4">8 월</th><th colspan="3">9 월</th></tr>
<tr><th>3</th><th>4</th><th>1</th><th>2</th><th>3</th><th>4</th><th>1</th><th>2</th><th>3</th><th>4</th><th>1</th><th>2</th><th>3</th><th>4</th><th>1</th><th>2</th><th>3</th></tr>
<tr><td colspan="2">주제 선정</td><td></td><td></td><td></td><td></td><td></td><td colspan="3"></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td colspan="5">관련 공부 및 주제 고도화</td><td colspan="3"></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td colspan="3">기술 분석</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td colspan="2">임베딩 적용</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td colspan="3">모델 아키텍처 구현</td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td colspan="4">모델 성능 개선 및 최적화</td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td colspan="4">웹 인터페이스 개발 및 연동</td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td colspan="2">최종보고서</td></tr>
</table>

6. 역할 분배

김다현	결합 친화도 예측 모델 개발
박주은	웹 인터페이스 개발 및 연동, 단백질 리간드 구조 시각화
안수현	결합 부위 예측 모델 개발

7. Binding site prediction model

7.1. Abstract

본 과제에서 제안하는 PLaNet-X 의 결합 부위 예측 모델은 단백질 서열을 기반으로 결합 부위를 예측한다. 기존 연구에서는 단백질의 3 차원 구조를 활용한 모델이 서열 기반 모델보다 우수한 성능을 보였다. 예를 들어, HoTS(2022)는 CNN 을 통해 단백질 서열 내 모티프(sequential motifs)를 효과적으로 포착하고, 이렇게 추출된 서열 모티프와 화합물 간의 상호작용을 transformer 로 모델링하여 결합 부위 예측에서 뛰어난 성능을 달성하였다. 그럼에도 이러한 서열 기반 접근법은 3 차원 구조 기반 방법들만큼의 성능을 내지 못했다.

이후 발표된 Pseq2Sites: Enhancing protein sequence-based ligand binding-site prediction accuracy via the deep convolutional network and attention mechanism(2024)은 단백질 서열을 기반으로 결합 부위를 예측하는 모델이지만, 기존의 3D 구조를 기반으로 한 모델보다도 더 좋은 성능을 보였다는 점에서 주목할 만하다. Pseq2Sites 는 ProtTrans 를 활용하여 단백질 임베딩을 진행하고, CNN 과 position-based attention 조합을 통해 결합 부위를 예측한다.

이에 비해 본 과제에서는 ESM-2 를 사용한 임베딩을 통해 더 풍부한 단백질 서열 표현을 확보하였다. 또한, CNN 기반의 DenseASPP 구조를 채택하여 응용하였으며, attention 기법은 사용하지 않았다.

DenseASPP 는 atrous convolution 을 병렬로 사용하는 ASPP 를 확장한 구조로, receptive field 를 더욱 넓고 조밀하게 커버할 수 있다. 또한 본 과제에서는 여기에 Global Average Pooling(GAP) branch 를 추가하여, 전체 서열 수준의 전역적인 정보 요약까지 반영하였다.

즉, PLaNet-X 의 결합 부위 예측 모델은 HoTS, Pseq2sites 와 같은 서열 기반의 단백질 결합 부위 예측 모델이지만 우수한 성능을 보인 모델이 채택한 방법인 CNN 과 attention 의 조합이 아닌 그저 CNN 만으로도 좋은 성능을 보였다.

7.2. Dataset preprocessing

7.2.1. 데이터 출처

본 과제에서 단백질-리간드 복합체의 결합부위(binding-site)를 예측하기 위해서 모델 학습에 사용한 데이터는 sc-PDB 와 PDBbind 에서 추출하였다.

PDB(Protein Data Bank)는 단백질과 핵산 등 생체 거대분자의 3 차원 구조 정보를 수록한 국제적 공개 데이터베이스로, X-ray 결정학, NMR 분광학, cryo-EM 등 실험 기법으로 얻은 구조 데이터를 제공한다. 또한, 이러한 실험으로 관찰된 구조는 단백질뿐만 아니라 리간드, 금속 이온, 물 분자를 포함한 복합체가 PDB 파일(.pdb)에 기록된다.

이러한 PDB 를 바탕으로 구축된 데이터베이스 중 하나가 sc-PDB 이다. sc-PDB 는 구조 기반 약물 설계를 돕기 위해, PDB(Protein Data Bank)를 분석하여 약물 유사 리간드(drug-like ligand)가 결합할 수 있는 결합 부위를 찾아내고 그 결과로 만든 데이터베이스다. 한편, PDBbind 는 PDB 에 등록된 단백질

리간드 복합체에 대해 실험적으로 측정된 결합 친화도(binding-affinity) 데이터를 체계적으로 모아 연결한 데이터베이스다. 그 중에서도 PDBbind v2020 을 사용하였다.

테스트용 데이터셋으로는 COACH420 과 HOLO4K 각각의 'Mlig' 서브셋를 사용하였으며, 두 데이터셋 모두 P2Rank 에서 처음 사용되었다. 여기서 P2Rank 는 단백질 결합 부위를 예측하는 모델이다.

7.2.2. 학습 데이터셋 전처리

PDB 파일에는 화합물의 단백질 표준 아미노산 원자가 ATOM 레코드로, 리간드 및 물 분자, 금속 이온과 같은 비표준 화학종이 HETATM 레코드로 기록된다. 본 과제에서는 결합 부위 예측 모델의 학습 데이터로 화합물이 아닌 순수 단백질 서열 정보를 사용하기 위해 HETATM 레코드를 제거한 protein-only PDB 파일을 사용하였다.

또한 단백질 서열 길이를 최대 1500 residue 로 제한하였다. 단백질마다 서열의 길이가 상이하며 지나치게 긴 서열의 단백질은 모델의 성능을 저하시킬 수 있기 때문이다.

Open Babel 은 화학 데이터를 다루기 위한 오픈소스 화학 툴킷이다. 다양한 분자 구조 파일 형식을 변환, 분석, 검색, 시각화할 수 있도록 지원한다. 화학정보학(cheminformatics), 생화학(bioinformatics), 재료과학(materials science) 등 폭넓은 분야에서 활용된다.

RDKit 는 화학 정보학(cheminformatics) 및 분자 모델링을 위한 오픈 소스 파이썬 라이브러리다. 화합물의 처리, 분석, 시각화, 그리고 화학적 속성의 계산을 지원한다.

본 과제에서는 Open Babel 을 사용하여 리간드 구조 파일(mol2)을 SMILES 포맷으로 변환하였고, 이후 RDKit 를 사용하여 SMILES 문자열을 파싱하고 표준화하였다. 이 과정에서 RDKit 가 정상적으로 해석할 수 없는 리간드(구조 오류, kekulization 실패 등)를 포함한 복합체는 데이터셋에서 제외하였다.

7.2.3. 테스트 데이터셋 전처리

단백질 원자 좌표와 리간드 원자 좌표의 거리행렬을 계산하고, 거리가 4Å 이하인 잔기를 결합 부위로 채택하였다. 이 규칙은 P2Rank 에서 제안되었다.

표준 20 종 아미노산이 리간드로 기록된 복합체는 제외하였으며, UniProt 서열에 매핑되지 않는 단백질 또한 제거하였다 Uniprot 은 생명정보학에서 가장 널리 사용되는 단백질 서열 데이터베이스로 데이터 품질 관리와 표준화를 위해 UniProt 매핑은 필수적이다. 또한, 동일한 단백질–리간드 복합체가 학습 데이터셋과 테스트 데이터셋에 동시에 포함되지 않도록 하여 데이터 누수(data leakage)를 방지하였다.

7.2.4. 최종 데이터셋 구성

위의 전처리 과정으로 scPDB 에서는 16,703 개, PDBbind 는 12,478 개의 복합체가 남게되었고, 최종적으로 총 25,162 개의 training data 로 학습을 진행하였다. (PDB id 를 기준으로 중복된 단백질은 제거했다.) 위의 모든 전처리 과정은 Pseq2sites 를 참고했다.

7.3. Protein sequence Embeddings

단백질의 아미노산들의 연속된 서열로 이루어져 있다. 본 과제에서는 결합 부위를 예측할 때 3D 구조 정보가 아닌 서열 정보를 활용한다. 최근 단백질 서열 데이터에 구조적 특성을 반영할 수 있도록 설계된 다양한 언어 모델들이 제안 되고 있으며, 본 과제에서는 그 중 ESM-2 를 채택하였다.

● ESM-2

ESM-2 는 Meta AI 에서 개발한 Transformer 기반 단백질 언어 모델이다. UniRef50 과 같은 대규모 단백질 서열 데이터셋을 기반으로 masked language modeling(MLM) 방식으로 학습되었고, 단백질의 구조 및 기능적 특성을 순수 서열 정보만으로 학습할 수 있도록 설계되었다. ESM-2 는 여러 규모의 모델로 공개되었으며, 본 과제에서는 연산 자원과 성능 간의 균형을 고려해 t33_650M 모델을 채택하였다.

Configuration	Layers	Parameters
params_esm2_t12_35M_UR50D.yaml	12	~35M
params_esm2_t33_650M_UR50D.yaml	33	~650M
params_esm2_t33_650M_UR50D_vsl.yaml (Variable Sequence Length enabled for efficient training)	33	~650M
params_esm2_t36_3B_UR50D.yaml	36	~3B

params_esm2_t48_15B_UR50D.yaml	48	~15B

Table2. Available ESM-2 configurations

본 과제에서 단백질 서열을 임베딩하기 위해 채택을 고려한 다른 모델로는 ProtT5 가 있다. 그러나, ESM-2 는 ProtT5 에 비해 구조 예측에서 더 강력한 성능을 보이고, ProtT5 는 ESM-2 에 비해 모델이 느리고 메모리 소모가 크다는 점에서 ESM-2 를 채택하였다.

모델	기반 구조	주요 목적	데이터셋	장점	한계
ESM-2	Transformer (Bert 계열)	단백질 표현(Protein representation)	UniRef50&UniRef90(약 6,500 만 서열)	강력한 문맥 표현력(Contextual modeling)	계산 비용이 높음
ProtT5	T5 (Text-to-Text Transforme)	단백질 언어모델링(Protein language model)	UniRef50& BFD(약 21 억 서열)	범용성, 멀티태스크 적용 가능	속도가 느리고 자원 소모가 큼

Table3. ESM-2 vs ProtT5**,** Chen et al., *UniAMP: enhancing AMP prediction using deep neural networks with inferred information of peptides*, BMC Bioinformatics, 2025 (Table 3 기반 재구성).

만약 단백질이 여러 개의 체인으로 구성되어 있다면, 단백질을 체인별로 나누어 ESM-2 에 각각 입력하였다. ESM 은 입력 길이에 제한(최대 1,024 개 아미노산)이 존재하므로, 체인 단위로 나누어 처리하는 것이 효율적이기 때문이다.

7.4. PLaNet-X architecture

PLaNet-X 의 결합 부위 예측 모델은 CNN 기반의 DenseASPP 구조를 사용한다. 과정은 다음과 같다. 두 커널 분기(k=3, k=5)에 대해 각각 dilation $d \in \{1,2,3\}$, $d \in \{2,4,6\}$을 적용한 두 DenseASPP 블록으로 multi-scale context 를 추출한다. 두 분기의 출력은 concat 되어 1x1 convolution 을 통해 결합된다. 이후 DenseASPP 를 거치지 않은 원래의 feature 와 concat 되어 MLP 에 입력되어 위치별 binary logit 을 산출한다. 이는 지역적 패턴과 보다 넓은 문맥까지 모두 아우르는것을 목표로 한다.

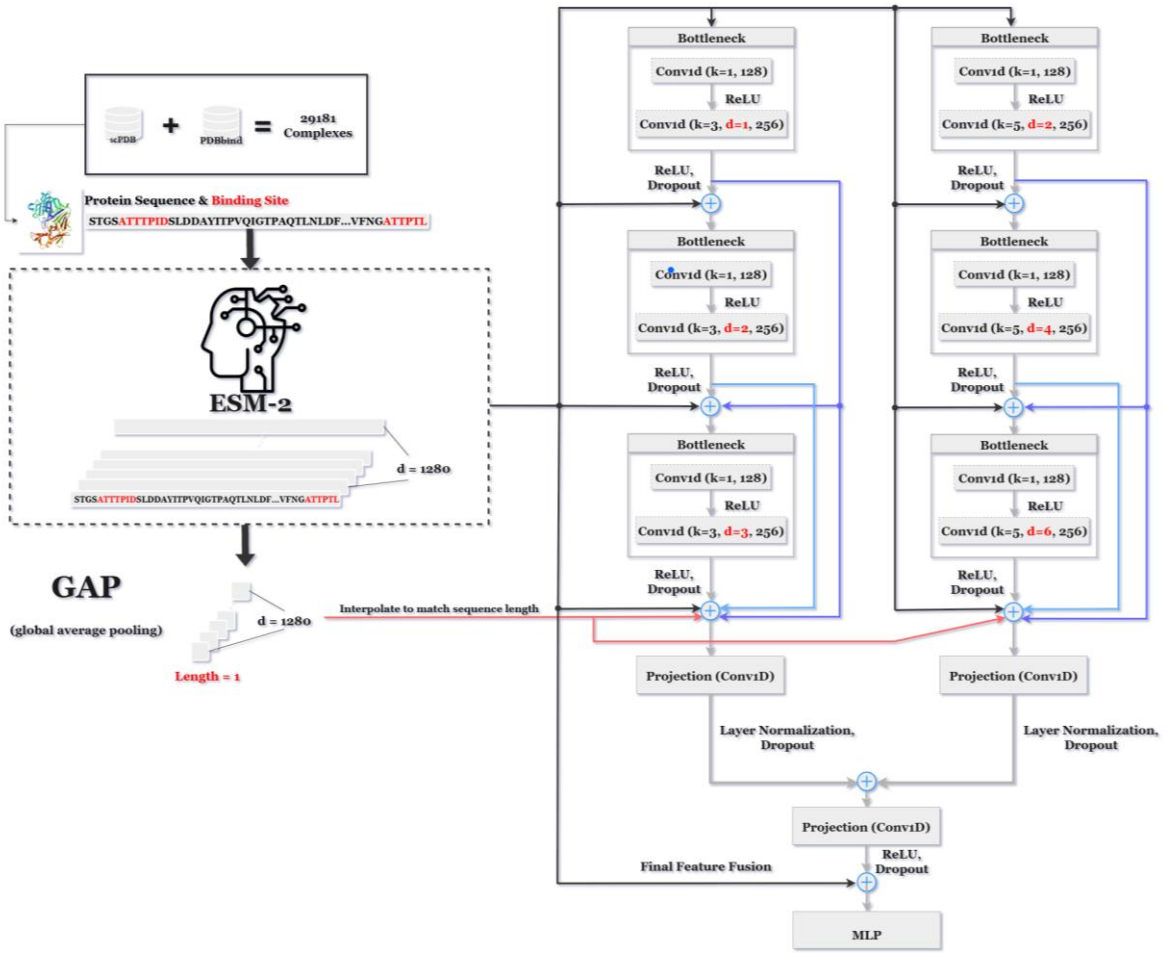


Figure 3. PLaNet-X 단백질 결합 부위 예측 모델 구조를 시각화 한 그림이다.

7.4.1. ASPP(Atrous Spatial Pyramid Pooling)

ASPP 는 atrous convolution 을 서로 다른 dilation rate 로 병렬 적용하여 다양한 receptive field 를 통해 multi-scale context 를 추출한 뒤, 그 결과를 합치는 모듈이다. DeepLab V3 모델의 논문인 Rethinking Atrous Convolution for Semantic Image Segmentation(2017)에서 소개된 ASPP 의 내용은 다음과 같다.

ASPP 는 여러 개의 병렬 convolution branch(1×1 Convolution, 다양한 dilation rate 를 갖는 3×3 Convolutions)와 GAP(Global average pooling) branch 로 구성된다. DeepLab V3 모델은 ASPP 에 input

feature map 을 1X1 의 크기를 만들어주는 GAP(global average pooling)을 추가하여 문맥적 정보를 보강하였다. 각 branch 의 출력을 채널 차원에서 연결(concatenate)한 뒤, 최종적으로 1×1 합성곱을 통해 출력 채널 수를 정리한다.

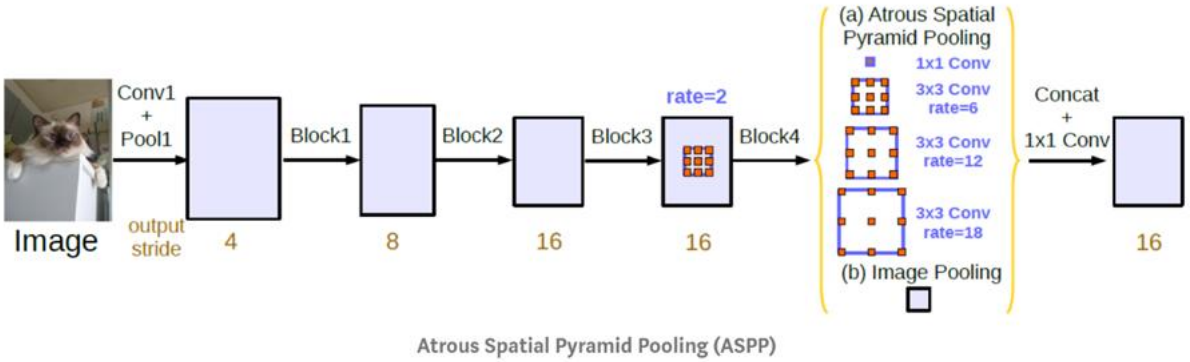


Figure 4. ASPP 의 구조를 시각화 한 그림 자료이다. 출처: Rethinking Atrous Convolution for Semantic Image Segmentation

7.4.2. DenseASPP - Atrous Convolution

기존 ASPP 는 서로 다른 dilation rate 의 atrous convolution 을 병렬로 적용하는 반면, DenseASPP for Semantic Segmentation in Street Scenes(2018)에서 소개된 DenseASPP(Densely connected Atrous Spatial Pyramid Pooling)는 atrous convolution 을 순차적으로 쌓으면서 입력 feature map 과 이전 층의 출력을 모두 연결(concatenate)하여 다음 층에 전달한다.

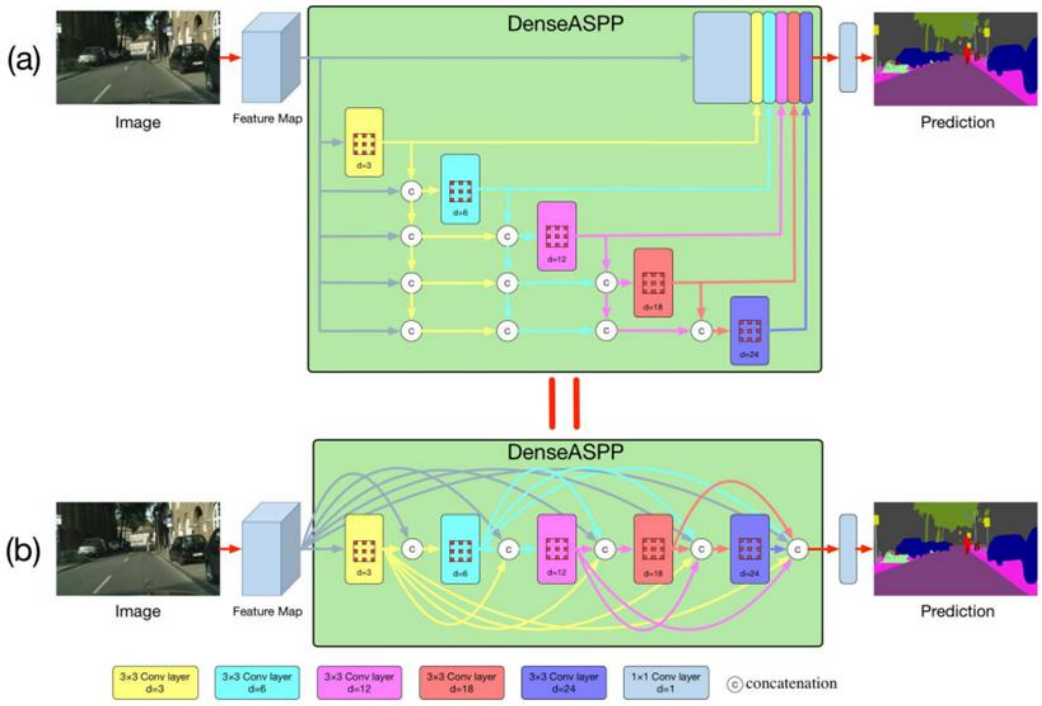


Figure 5. DenseASPP 의 구조 "(a)에서는 DenseASPP 를 상세히 보여준다. 각 dilated convolution 층의 출력은 입력 feature map 과 concat 되고, 이후 다음 dilated 층의 입력으로 전달된다. (본래 논문의 figure2)

DenseASPP 는 ASPP 에 비해 훨씬 더 조밀한 feature pyramid 를 생성한다. 여기서 조밀하다(dense)는 두 가지 의미를 포함한다.

첫째, 스케일 다양성(scale diversity)이 높아 더 많은 크기의 receptive field 를 커버한다는 뜻이다. ASPP 는 dilation rate {6, 12, 18, 24}를 사용해 제한된 네 가지 receptive field 만 얻을 수 있지만, DenseASPP 는 atrous convolution 층들을 연속적으로 쌓고(dense connection) 이전 층의 출력들을 연결(concatenate)하여 입력으로 사용하기 때문에 중간 단계에서 다양한 receptive field 가 추가로 생기는 효과가 있다. 따라서 DenseASPP 가 만드는 receptive field 의 집합은 ASPP 의 집합보다 훨씬 크며, 스케일 축에서 빈틈 없는 coverage 를 제공한다.

둘째, 픽셀 샘플링(pixel sampling)이 더 조밀하다는 점이다. ASPP 의 경우 dilation 이 커질수록 receptive field 는 넓어지지만, 실제 연산에 참여하는 픽셀 수는 그대로이기 때문에 정보 손실이 발생한다. 예를 들어, 1 차원에서 dilation=6, kernel=3 인 atrous convolution 은 receptive field 가 13 이지만 이 중 세 개 픽셀만 활용한다. 반면 DenseASPP 는 작은 dilation conv 의 출력을 큰 atrous convolution 층이 다시 활용하도록 연결하기 때문에, 동일한 receptive field 안에서도 더 많은 픽셀이 최종 계산에 기여한다.

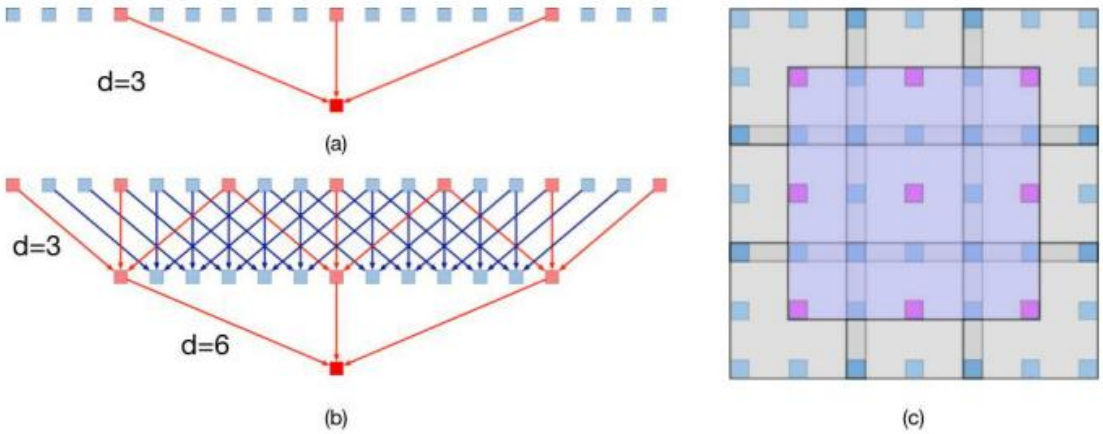


Figure 6. **(a)** 팽창률(dilation rate)이 6 인 표준 1 차원 atrous convolution. **(b)** 작은 팽창률을 가진 atrous layer 를 큰 팽창률을 가진 atrous layer 아래에 쌓으면 샘플링이 더 조밀해진다. 빨간색은 정보가 오는 위치를 나타낸다. **(c)** (b)의 개념을 2 차원으로 확장한 버전. (본래 논문의 figure4)

결과적으로 DenseASPP 는 스케일 축에서의 다양성과 픽셀 차원의 조밀한 샘플링을 동시에 개선하여, ASPP 보다 더 강력하고 풍부한 feature pyramid 를 구성한다. 이러한 특성 덕분에 다양한 크기의 객체를 더 효과적으로 인식할 수 있다.

7.4.3. 두 개의 DenseASPP(k=3, k=5) Branch

본 과제는 두 개의 DenseASPP block 으로 구성된다. 하나는 kernel=3, 다른 하나는 kernel=5 를 사용하며, 각각 여러 dilations 을 적용한다.

Kernel 의 크기가 3 인 경우에는 (1,2,3)의 dilations 를 적용하였고, kernel 의 크기가 5 인 경우에는 (2,4,6)의 dilations 를 적용하였다. 이렇게 하면 더 짧은 국소 패턴부터 조금 더 넓은 문맥까지 서로 다른 범위의 특징을 포착할 수 있다. 두 분기의 출력은 채널 방향으로 concat 한 뒤 1×1 Conv(2H→H)로 채널 혼합을 수행한다.

7.4.4. GAP, 기존의 DenseASPP 와의 차이점

본 과제에서는 DenseASPP 모듈에서 추출된 feature 와 더불어, Global Average Pooling(GAP) branch 를 통해 단백질 전체 서열 수준의 전역적 정보를 보완하였다. 이는 DeepLab V3 과 동일한 설계이다. 이러한 설계는 결합 부위 예측에서 국소적·전역적 맥락을 모두 반영할 수 있는 장점을 제공한다. 반면, DenseASPP 가 소개된 DenseASPP for Semantic Segmentation in Street Scenes(2018)에선 GAP 를 사용하지 않는다.

7.4.5. 1×1 bottleneck-like 설계, 기존의 DenseASPP 와의 차이점

각각의 atrous convolution 블록 전단에는 1×1 convolution 을 배치하여 채널 수를 채널 축소 후 atrous convolution 을 적용하도록 bottleneck-like 설계를 사용한다. (ResNet 의 1×1→3×3→1×1 과

유사하되, 마지막 1×1 확장 단계는 생략) 이는 ResNet 에서 제안된 bottleneck 구조를 응용한 것으로, 연산량 및 메모리 사용량을 줄이고 파라미터 수를 절감하는 효과가 있다.

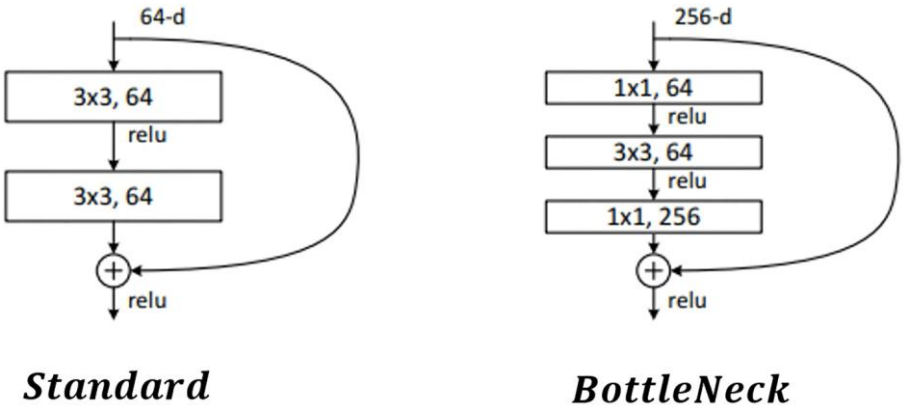


Figure 7. BottleNeck 과 Standard 구조의 차이 (출처 : Deep Residual Learning for Image Recognition(Resnet 논문)의 Figure 5.)

7.4.6. LayerNorm, Dropout

각각의 atrous convolution 블록 출력은 LayerNorm 으로 안정화한다. Dropout 은 과적합 억제를 위해 사용되었다.

7.4.7. MLP

마지막으로 MLP 에 들어가기 전, 입력 층에 protein features 를 concat 한다. 여기서 protein features 는 단백질 서열에 embeddings 를 거친 결과에서 nn.Linear(1280, hidden)을 통해 차원을 hidden size 로 선형 변환한 버전이다.

앞서 만든 protein features 을 채널 축으로 concat 하면 2H 차원 입력이 만들어진다. 이처럼 문맥화된 표현과 원천 표현을 함께 사용하는 이유는 context 추출에서 발생 가능한 정보 손실을 원천 표현으로 보완하기 위함이다.

MLP 내부는 다음과 같은 구성이다.

(1) GELU 비선형성 → LayerNorm → Dropout → 선형 축소, 이 패턴을 두 단계 적용한 뒤 최종 1 차원 로짓을 낸다.

(2) 첫 선형층은 512 -> 128, 두 번째 선형층은 128->64, 마지막으로 최종 1 차원 로짓을 낸다.

7.5. Training

7.5.1. K-fold cross validation

K-fold cross validation 은 가장 보편적으로 사용되는 교차 검증 기법으로, K 개의 데이터 fold 세트를 만들어서 K 번만큼 각 fold set 에 학습과 검증 평가를 반복적으로 수행하는 방법이다. 본 연구는 5-fold cross validation 을 사용하였다.

전체 단백질 ID 집합을 기준으로 무작위(shuffle=True, random_state=0)로 5 개 폴드로 분할하고, 각 반복에서 4 개 폴드를 학습, 나머지 1 개 폴드를 검증에 사용한다.

7.5.2. Optimizer

본 과제에서 최적화에는 AdamW 를 적용하였다. AdamW 는 기존 Adam optimizer 에서 발생하는 L2 정규화 항이 올바르게 반영되지 않는 문제를 해결하기 위해 제안된 방식이다. 전통적인 SGD 에서는

손실 함수에 L2 항을 더하는 것과 weight decay 를 직접 적용하는 것이 동일하게 작동하지만, Adam 과 같은 적응형 optimizer 에서는 두 방법이 같지 않다. AdamW 는 이러한 문제를 개선하기 위해 weight decay 를 학습률 업데이트와 분리(decoupling)하여 적용하며, 이를 통해 학습률과 weight decay 를 독립적으로 조절할 수 있고, 결과적으로 모델의 일반화 성능이 개선되는 것으로 보고되었다. PyTorch 에서는 torch.optim.AdamW 로 바로 사용할 수 있다.

Optimizer	Learning Rate	Weight Decay	Batch Size	Epochs
AdamW	1e-3	0.01	8	50

Table3. Optimizer & Hyperparameters

7.5.3. 마스크 기반 손실 함수

본 연구에서는 단백질 서열 입력이 가변적이므로, 패딩 구간을 학습에서 배제하기 위해 마스크(mask)를 적용한 이진 교차 엔트로피(Binary Cross Entropy with Logits) 손실 함수를 사용하였다. 또한 단백질 전체 서열에서 binding site residue 의 비율은 매우 낮아, 클래스 불균형(class imbalance) 문제가 심각하다. 이를 완화하기 위해 배치 내 양성/음성 비율을 기반으로 동적으로 계산된 가중치(pos_weight)를 적용하여 양성 클래스에 더 큰 페널티를 부여하였다.

$$\ell(p_i, l_i) = -\omega_p \sum_{l_i=1} \log(\sigma(p_i)) - \omega_n \sum_{l_i=0} \log(1-\sigma(p_i))$$

식 (1). 가중치가 적용된 BCE 손실

7.5.4. 검증 및 체크포인트 정책

각 Epoch 종료 시, 배치 단위 학습 결과로부터 검증 손실의 평균값을 계산하였다. 그리고 검증 손실이 최저치를 갱신할 때마다 모델 가중치를 저장(checkpoint) 하여, 과적합을 방지하고 최적 성능을 유지할 수 있도록 하였다.

테스트 단계에서는 여러 체크포인트(fold 모델)를 로드하여 예측 확률을 평균함으로써, 보다 견고한 residue-level 예측 확률을 산출하였다.

7.5.5. 평가 지표

본 과제에서는 Precision, Recall, F1, Specificity, G-mean 등을 종합적으로 사용하였다. 특히 서열 기반의 예측 모델에서는 precision 이 낮게 나오는 경우를 다른 논문들을 통해 목격할 수 있었다. Pseq2sites 나 HoTs 같은 경우에는 서열 기반의 좋은 성능을 보인 예측 모델이지만 precision 같은 경우는 각각 0.178, 0.052 정도로 낮은 수치를 보였다. (dataset = COACH420) 이는 CNN 기반의 모델의 경우 Convolution 을 진행하며, binding site 가 아닌 부분도 binding site 로 인식되는 경우가 많기 때문이다. PLaNet-X 는 기존의 서열 기반 모델과 비교하여 precision 이 특히 개선됨을 확인했다.

7.6. Result

7.6.1. Pseq2sites

성능 평가를 위해 비교를 진행한 모델은 Pseq2Sites(2024)다. Pseq2sites 는 단백질 서열 정보 기반의 모델이지만 3D 구조 기반 모델들보다도 좋은 성능을 보인 모델이다. 이후로 유의미하게 Pseq2sites 를 능가하는 성능을 보인 모델이 없었기에, 본 과제에서 비교 모델로 삼게되었다. 아래는 Pseq2sites 가 다른 모델들, 특히 3D 구조 기반 모델들을 포함하더라도 좋은 성능을 보인 결과를 그래프로 나타낸 것이다. 서열을 기반으로 한 모델은 Pseq2Sites, DeepCSeqSite, HoTS, BiRDS 이며, 3D 구조를 기반으로 한 모델은 Fpocket, P2Rank, DeepSurf, DeepPocket 이다.

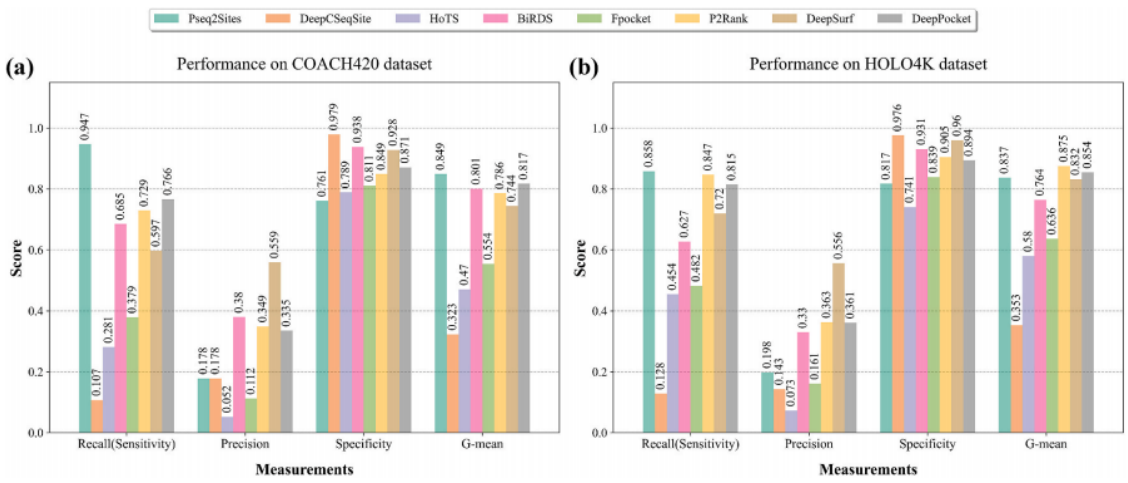


Figure 8. Pseq2sites 와 다른 binding-site 예측 모델들의 COACH420, HOLO4K 를 활용한 성능 비교 결과 (출처 : Pseq2sites 논문)

다음은 Pseq2sites 가 COACH420, HOLO4K 로 테스트한 성능, 그리고 본 과제의 모델인 PLaNet-X 가 동일한 데이터셋을 사용하여 테스트한 성능의 수치를 막대 그래프로 비교한 모습이다. Pseq2sites 의 성능값은 원 논문에서 보고된 값을 참고했다. 두 모델 모두 단백질의 sequence 의 길이 제한을 1500 으로 두었으며, 데이터 전처리 과정이 동일하다.

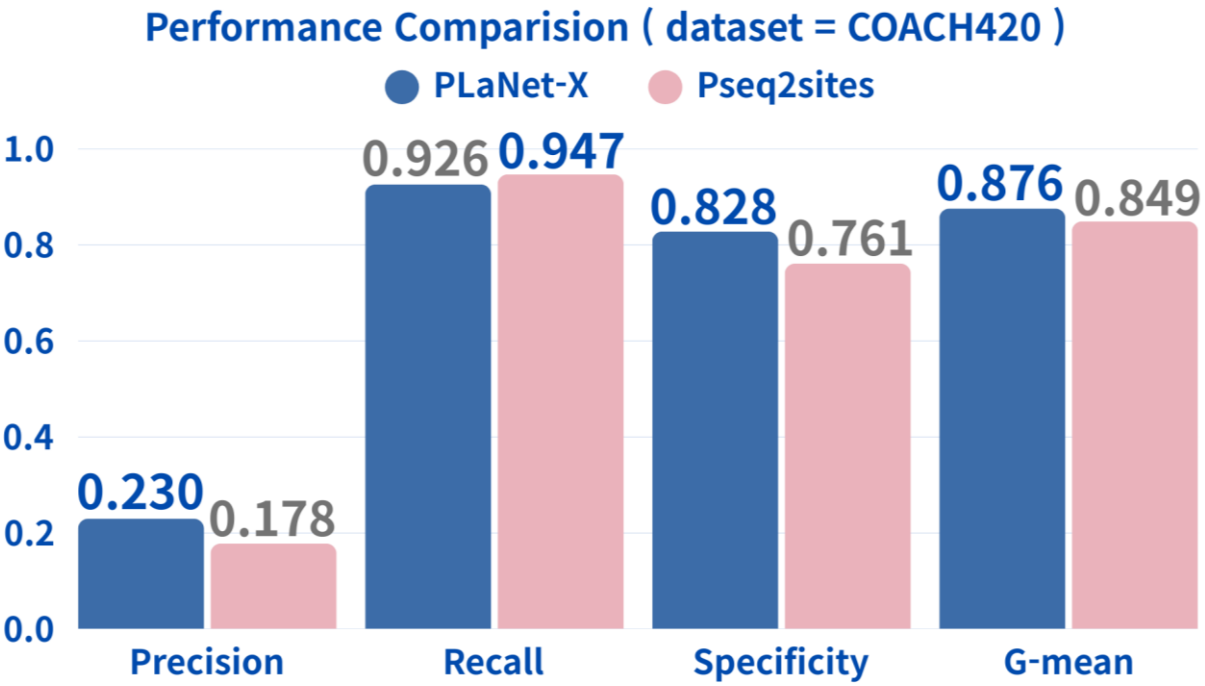


Figure 9. Performance Comparison (dataset = COACH420). 두 모델 중 보다 낮은 성능을 보인 모델의 수치를 회색으로 표기 하였다.

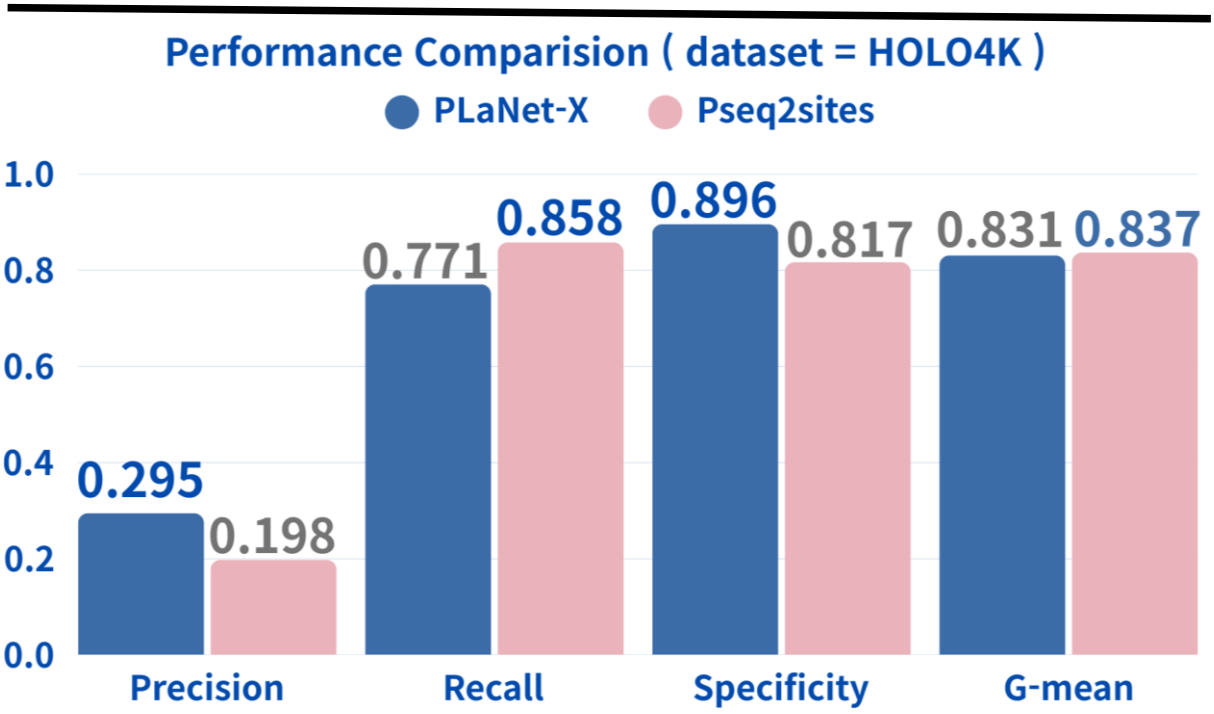


Figure 10. Performance Comparison (dataset = HOLO4K). 두 모델 중 보다 낮은 성능을 보인 모델의 수치를 회색으로 표기 하였다.

위의 막대 그래프를 통해, COACH420 테스트셋의 경우, 기존의 3D 모델보다도 좋은 성능을 보였던 Pseq2sites 보다 거의 대부분에서 성능이 좋아진 것을 확인할 수 있다. PLaNet-X 결합 부위 예측 모델의 경우 Recall 의 경우 약간 낮아졌으나, 그에 비해 precision 은 확실히 좋은 성능을 보이는 걸 알 수 있다. HOLO4K 의 경우에는 특히 Precision 이 1.5 배 좋은 성능을 보였다.

7.6.2. Ablation/Variants - Attention

서열 기반 단백질 결합 부위 예측에서는 CNN 에 어텐션을 결합한 구조가 널리 활용되어 왔다. 이에 본 연구도 초기 설계 단계에서 CNN+Cross-Attention 변형을 고려하였고, 해당 변형이 CNN-only 대비 우수한 성능을 보일 것이라는 가설을 세웠다. 이를 검증하기 위해 학습/평가 설정을 동일하게 유지한

채(데이터 분할, 전처리, 손실, 옵티마이저, 스케줄 동일) CNN-only 모델에 cross-attention 모듈을 추가한 CNN+Attention 변형을 비교하였다.

그 결과, CNN+Attention 은 유의미한 성능 향상을 보이지 않았고 오히려 대폭 하락하였다. 비용 측면(파라미터 수 및 추론 지연)에서도 불리하여, 본 연구의 최종 구조는 CNN-only 로 채택하였다.

성능 저하의 가능 원인으로는 (i) 데이터 규모 대비 모듈 복잡도 증가에 따른 과적합, (ii) DenseASPP 기반 CNN 이 이미 충분히 넓은 수용영역을 통해 장거리 문맥을 포착한다는 점 등을 지적한다.

7.6.3. Ablation/Variants - GAP

서열 수준의 전역 정보를 보강하기 위해, DenseASPP 모듈에 Global Average Pooling(GAP) branch 를 추가하였다. 동일한 학습·평가 설정(데이터 전처리, 손실, 옵티마이저, 스케줄)에서 GAP 미적용 대비 적용 모델이 성능 향상을 보였다. 이에 본 연구의 최종 구조에는 GAP branch 를 채택하였다.

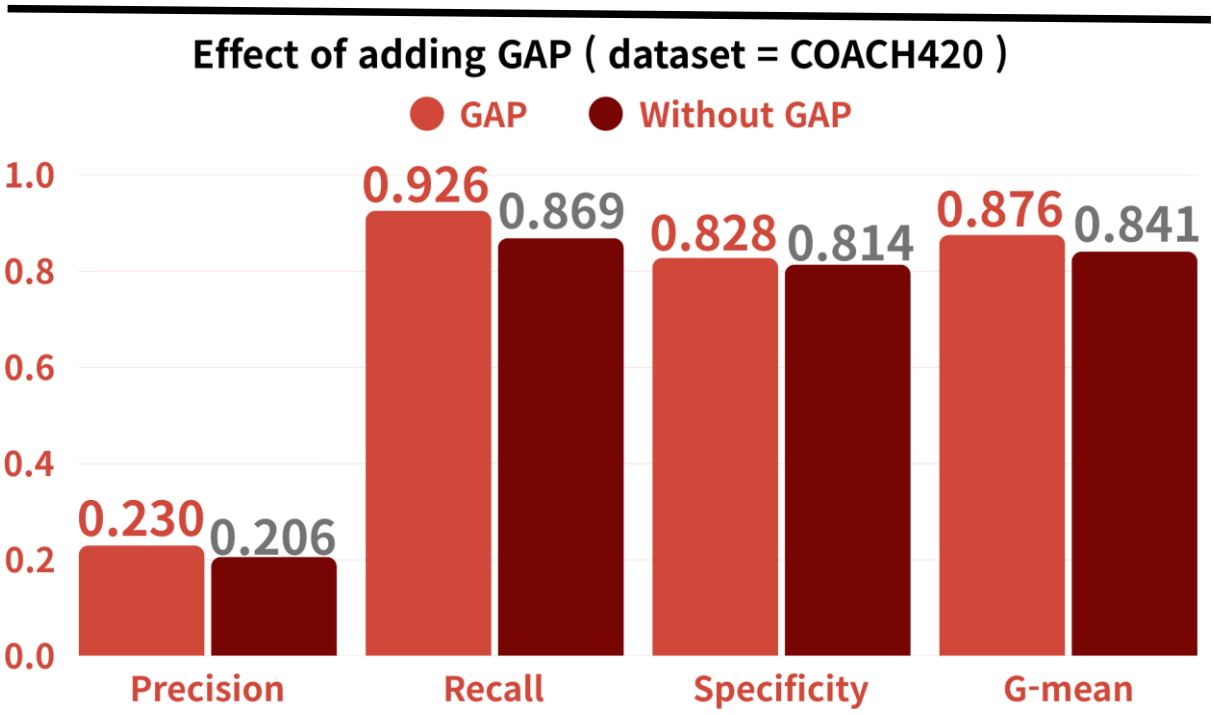


Figure 11. Effect of adding GAP (dataset = COACH420). 두 모델 중 보다 낮은 성능을 보인 모델의 수치를 회색으로 표기 하였다.

8. Binding Affinity prediction model

단백질–리간드 결합 친화도의 정확한 예측은 신약 개발을 가속화하는 핵심 과제지만, 기존 연구들은 주로 단백질 구조 정보를 활용하여 표현력을 높이는 방식에 의존해 구조 데이터의 확보와 계산 비용 측면에서 제약이 존재한다. 최근에는 서열 기반 접근이 대안으로 주목받고 있으며, 대표적인 모델인 CAPLA 는 단백질 아미노산 서열과 리간드 SMILES 서열을 활용하면서도 결합 부위(Binding Residue, BR) 정보를 추가적으로 요구한다. 본 연구에서 제안하는 Planet-X 는 사용자 편의성과 실용성을 고려해 오직 단백질 서열과 리간드 서열만을 입력으로 활용하는 순수 서열 기반(sequence-only) 결합 친화도

예측 모델로, 단백질 표현은 ESM 사전학습 임베딩과 물리화학적 특성을 결합해 구축하고, 리간드 표현은 ChemBERTa 임베딩과 전역 물성 정보를 통합하여 구성하였다.

또한 Dilated Convolution Block 을 통해 국소적 패턴과 장거리 의존성을 동시에 포착하고, 양방향 교차 어텐션을 적용해 단백질–리간드 상호작용을 정밀하게 학습하였다. 벤치마크 실험 결과, Planet-X 는 BR 정보를 사용하지 않고도 CAPLA 에 근접한 성능을 달성하였으며, 이는 구조 정보에 의존하지 않으면서도 높은 성능을 확보한 사용자 친화적인 서열 기반 예측 모델임을 보여준다.

8.1. 변경 사항

초기 설계 단계에서는 웹페이지의 전체적인 흐름을 고려하여, Planet-X 결합 부위 예측 모델로부터 도출된 binding site 정보를 결합 친화도 예측 모델의 입력으로 활용하는 방안을 검토하였다. 실제로 해당 정보를 활용한 모델을 구현하고 실험을 수행한 결과, binding site 정보가 예측 과정에서 불확실성을 내포하고 있으며, 학습 시에는 정확한 정보를 사용할 수 있지만 실제 환경에서는 예측된 값을 입력으로 사용해야 하기 때문에 오히려 결합 친화도 예측 성능이 저하될 가능성이 있는 것으로 판단되었다. 이에 따라 본 연구의 최종 모델에서는 binding site 정보를 입력에서 제외하고, 단백질 서열 기반 표현과 리간드 서열 기반 표현만을 입력으로 사용하는 구조로 설계하였다.

8.2. 사용 데이터셋

학습에는 PDBbind 2020 데이터셋을 사용하였다. PDBbind 2020 은 General set, Refined set, Core set 으로 구성되며, 각각의 특징은 다음과 같다.

- General set: 다양한 품질의 단백질–리간드 복합체를 포함하는 전체 데이터셋으로, PDBbind 에서 제공하는 가장 포괄적인 세트이다.

- Refined set: General set 중에서 구조 해상도, 리간드 품질, 결합 친화도 측정의 신뢰도 등을 기준으로 고품질 복합체만을 선별한 서브세트이다. Refined set 은 General set 과 단순히 중복되는 구조가 아니라, 고품질 데이터를 독립적으로 관리하기 위해 구성되었으므로 두 세트를 함께 사용하더라도 중복되지 않는다.

- Core set: Refined set 에서 선정된 대표적인 고품질 복합체로 이루어져 있으며, 모델의 성능을 정량적으로 평가하기 위한 벤치마크 세트로 활용된다.

구체적으로, Refined set 에서 무작위로 2,000 개의 복합체를 검증 세트(validation set)로 사용하였고, 나머지 Refined set 과 General set 을 합쳐 총 14,587 개의 복합체를 학습 세트(training set)로 구성하였다.

벤치마크 테스트에는 Core-2016 (279 complexes) 데이터셋을 사용하였다. Core-2016 은 PDBbind 2016 의 Core set 으로, 학습/검증 데이터와의 중복을 철저히 배제하여 공정성을 확보하였다.

추가적인 일반화 성능 검증을 위해 CSAR-HiQ 데이터셋 (Dunbar et al., 2011)을 별도의 테스트 세트로 활용하였다. 본 연구에서는 중복을 제거한 후 이를 두 개의 서브세트, CSAR-HiQ_36 (36 complexes) 및 CSAR-HiQ_51 (45 complexes)로 구분하여 평가하였다.

8.3. 모델 구성

단백질과 리간드의 전처리 과정은 앞서 사용한 binding site 예측 모델과 동일하다. 결합 친화도 값의 경우, Ki/Kd/IC50 및 pKi/pKd 를 추출하여 모두 nM 단위로 변환한 뒤, 이를 공식을 이용해 전처리하였다.

$$pAff = 9 - \log_{10}(\mathrm{nM})$$

또한, 샘플마다 서열 길이가 달라 단백질과 리간드의 길이를 통일할 필요가 있었다. 본 연구에서는 단백질 서열의 최대 길이를 1500, 리간드 SMILES 의 최대 길이를 150 으로 설정하였다. 해당 길이를 초과하는 서열은 제거하였으며, 짧은 서열은 0 으로 패딩하였다. 모든 데이터셋은 동일한 절차를 적용하였다.

8.4. 학습 및 하이퍼파라미터

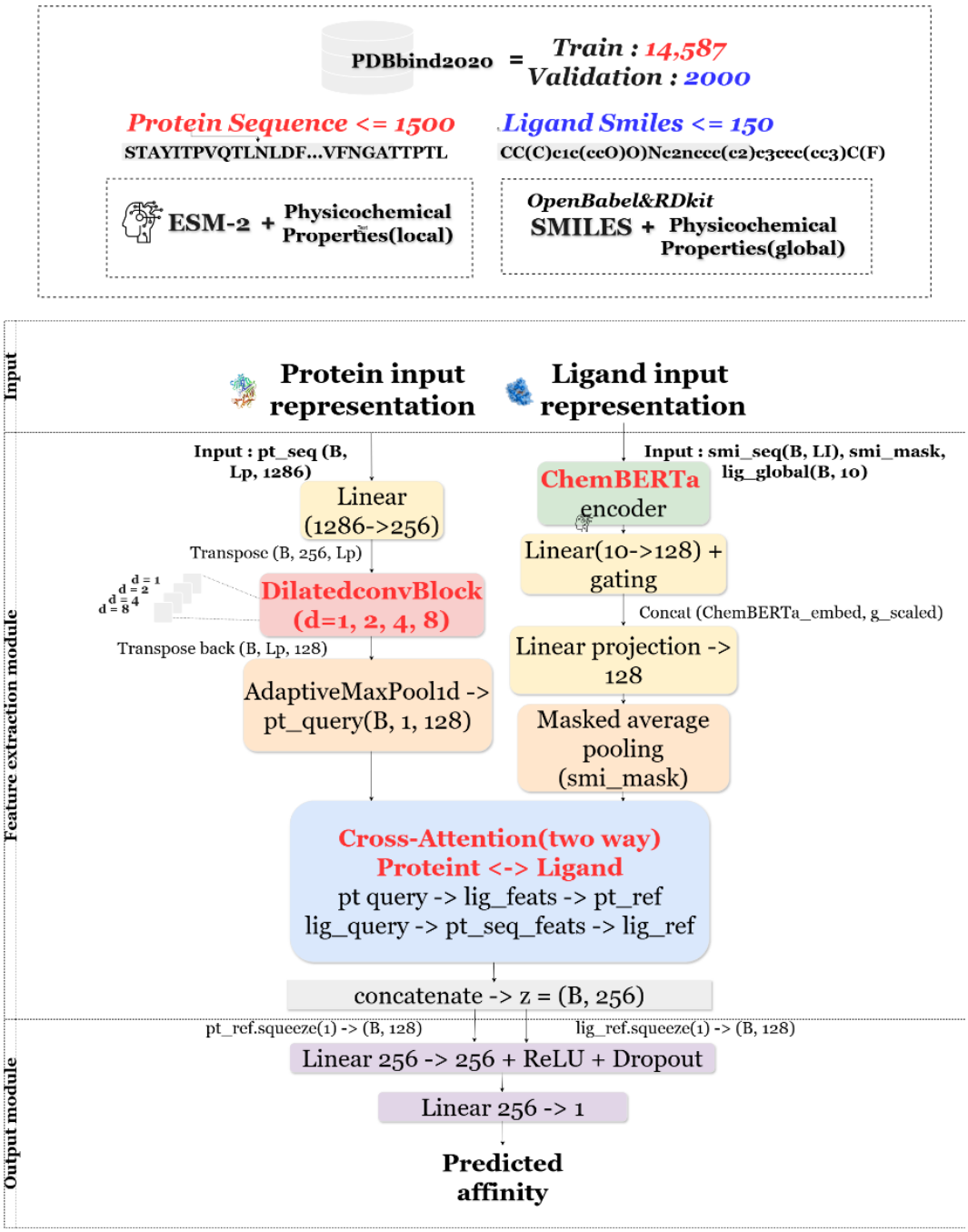


Figure 12. PLaNet-X 결합 친화도 예측 모델 구조를 간단히 시각화 한 그림이다.

8.4.1. 단백질 사전 임베딩 (Pre-Embedding)

본 모델은 단백질 서열에서 구조적 정보를 반영하기 위해 사전학습 모델인 ESM-2 를 활용하였다. ESM-2 는 앞서 7.3.1 절에서 상세히 기술했듯, 서열만으로도 구조 예측이 가능한 대규모 Transformer

기반 모델로, 서열 기반 접근의 한계를 보완할 수 있는 강점을 지닌다. 그러나 학습 과정에서 매 배치마다 ESM 을 직접 호출하는 방식은 과도한 연산 비용과 학습 시간 증가를 초래할 수 있다.

이에 따라, ESM 을 활용한 사전 임베딩 전략을 적용하였다. 구체적으로, 단백질 서열을 ESM 에 입력하여 residue 단위의 1,280 차원 임베딩을 추출하고, 여기에 별도로 계산된 residue 의 물리화학적 특성(산성, 염기성, 중성, 극성, 소수성 지수, 분자량; 총 6 가지)을 결합하였다. 최종적으로 residue-level 에서 1,286 차원의 벡터가 구성되며, 이 사전 임베딩된 결과를 학습 단계에서 직접 입력 피처로 활용하였다. 이를 통해 연산 효율을 확보하면서도 학습 성능을 유지·향상시킬 수 있었다.

8.4.2. 단백질 인코더

단백질 인코더는 입력된 단백질 서열 정보를 효과적으로 압축하여 결합 친화도 예측에 활용할 수 있도록 설계되었다. 먼저, residue-level 임베딩과 물리화학적 특성은 선형 변환을 거쳐 256 차원 공간으로 투영된다. 이를 통해 고차원 입력을 보다 효율적으로 처리 가능한 차원으로 축소하면서도 핵심적인 특징을 보존한다.

투영된 표현은 Dilated Convolution Block 으로 전달된다. 이 블록은 dilation rate 가 1, 2, 4, 8 인 1 차원 dilated convolution 을 병렬적으로 적용하여 서로 다른 범위의 패턴을 동시에 학습한다. 일반 합성곱이 국소적인 인접 잔기들만 고려하는 데 비해, dilated convolution 은 일정 간격을 두고 정보를 추출할 수 있어 receptive field 를 확장한다. 이로써 단백질 내의 짧은 모티프부터 서열상 멀리 떨어진 잔기 간의 장거리 상호작용(예: β-sheet, loop)까지 포착할 수 있다. 또한 dilation 을 활용하면 커널 크기를 크게 늘리지 않고도 넓은 영역을 학습할 수 있어 연산 효율성을 유지할 수 있다.

각 dilation 에서 얻은 네 개의 피처 맵은 채널 방향으로 결합되며, 이어서 Batch Normalization 과 PReLU 활성화를 거쳐 정규화 및 비선형 변환이 수행된다. 이를 통해 학습 안정성과 표현력을 강화한다.

마지막으로 Adaptive Max Pooling 을 적용하여 단백질 서열 길이에 상관없이 항상 128 차원의 고정 벡터를 추출한다.

결과적으로 단백질 인코더는 서열 기반의 결합 부위 정보, residue 의 물리화학적 특성, 그리고 서열 전반에 걸친 구조적 상관관계를 통합적으로 반영하는 표현을 학습한다.

8.4.3. 리간드 인코더

리간드 인코더에는 ChemBERTa 기반 사전학습(pre-trained) 모델을 활용하였다. ChemBERTa 는 SMILES 서열을 입력으로 받아 분자의 구조적·화학적 패턴을 학습하는 Transformer 계열 모델로, 분자 수준의 고차원 임베딩(representation)을 생성한다. 입력된 리간드 서열은 토큰화 과정을 거쳐 각 토큰이 벡터 임베딩으로 변환되며, Transformer 의 Self-Attention 메커니즘을 통해 토큰 간 상호작용과 장거리 의존성을 학습함으로써, 분자의 전체 구조와 기능적 특징을 통합적으로 반영한다.

본 연구에서는 ChemBERTa 임베딩에 전역 물리화학적 특성(global physicochemical properties) 10 가지를 추가하여 리간드 표현을 강화하였다. 포함된 특성은 분자량(MolWt), 소수성(MolLogP), 극성 표면적(TPSA), 회전 가능한 결합 수(NumRotatableBonds), sp^3 탄소 비율(FractionCSP3), 수소 결합 공여자/수용자 수(NumHDonors/NumHAcceptors), 고리 구조 수(RingCount), 몰 굴절률(MolMR) 등이, 정확한 10 가지 항목은 4-1 절에 상세히 서술되어 있다.

전역 피처 벡터는 먼저 토큰 임베딩과 동일한 차원으로 선형 변환된 후, Softplus 게이팅을 통해 학습 가능한 가중치가 적용되어 각 토큰 임베딩과 가중합 형태로 통합되고, 이후 concatenation 과 선형 변환을 거쳐 통합 리간드 표현을 생성하며, 마지막으로 padding 을 고려한 마스크 기반 평균 풀링을 적용하여 고정 차원의 최종 리간드 임베딩 벡터를 추출한다. 이를 통해 리간드의 local SMILES 패턴과

global 물리화학적 특성이 모두 반영된 풍부한 임베딩 벡터를 확보할 수 있으며, 이후 단백질–리간드 상호작용 학습에서 핵심 입력으로 활용된다.

8.4.4. 교차 상호작용 모듈

단백질과 리간드 간의 상호작용을 학습하기 위해 교차 어텐션(Cross-Attention) 모듈을 적용하였다. 이 모듈은 단백질과 리간드 각각의 전역 표현을 질의(query)로 사용하고 상대방의 토큰 임베딩을 키(key)와 값(value)로 사용하여, 서로가 상대의 어느 부분에 집중하는지를 학습한다.

- **단백질→리간드 방향**에서는 단백질 전역 표현을 질의(query)로 두고 리간드 토큰 임베딩을 key–value 로 사용하여, 단백질이 리간드의 어느 부분에 집중하는지를 학습한다.

- **리간드→단백질 방향**에서는 리간드 전역 표현을 질의로 두고 단백질 서열 표현을 key–value 로 사용하여, 리간드가 단백질의 어떤 잔기에 주목하는지를 학습한다.

이러한 양방향 교차 어텐션은 단백질과 리간드라는 서로 다른 시퀀스 간의 복잡한 상호작용 패턴을 효과적으로 포착한다. 각 교차 어텐션 블록은 다중 헤드 어텐션(Multi-Head Attention), Layer Normalization, 잔차 연결(Residual Connection), 2 층 Feed-Forward Network (hidden_dim=256, dropout=0.3)으로 구성되어 있으며, 이를 통해 최종적으로 양방향 상호작용 표현인 pt_ref(단백질 참조 표현)과 smi_ref(리간드 참조 표현)를 산출한다.

8.4.5. 결합 친화도 예측기

최종 예측 단계에서는 교차 상호작용을 거친 단백질 표현(pt_ref)과 리간드 표현(smi_ref)을 결합하여 입력으로 사용한다. 이후, Linear → ReLU → Dropout → Linear 블록을 통과시켜 단일 스칼라 출력을 생성하며, 이는 단백질–리간드 복합체의 결합 친화도(binding affinity) 값으로 해석된다.

8.5. 학습 및 하이퍼파라미터

모델 학습은 PyTorch 프레임워크를 기반으로 수행되었으며, 최적화 알고리즘으로 AdamW 를 사용하고 초기 학습률은 1e-4 로 설정하였다. 손실 함수는 초기에는 MSE(평균제곱오차)를 사용하였으나, 이상치에 보다 강건한 학습을 위해 Smooth L1 Loss(β=1.0)로 변경하였다. 학습은 총 100 epoch 동안 배치 크기 8 로 진행되었으며, GPU 환경에서 수행되어 효율적인 연산이 가능하였다. 학습 및 검증 과정에서는 손실(loss), RMSE, MAE 등의 성능 지표를 TensorBoard 를 통해 실시간으로 모니터링하였으며, 각 epoch 별 지표들은 metrics.csv 파일에 저장되었다. 또한, 검증 손실(validation loss)을 기준으로 가장 우수한 성능을 보인 모델 가중치를 별도로 저장하여 최종 평가 및 활용에 사용하였다.

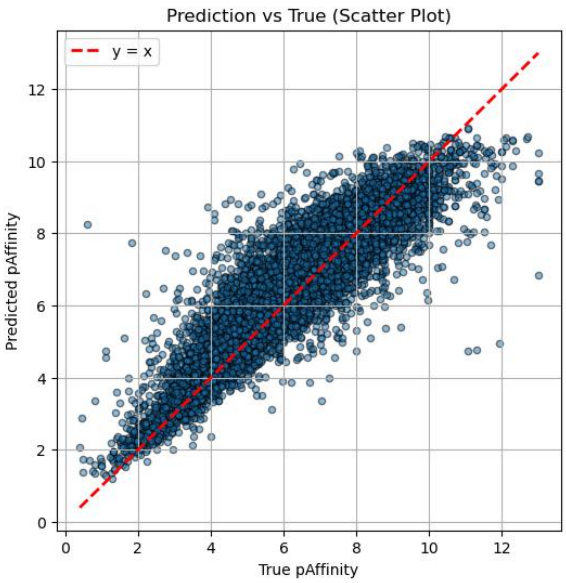
8.6. 성능 평가

최종적인 성능 평가는 다음과 같다.

Table 3. 데이터셋별 단백질-리간드 결합 친화도 예측 성능

Datasets	R	RMSE	MAE	SD	95% CI

Core-2016	0.8047	1.2967	1.0076	1.2930	1.1799, 1.4138
CSAR-HiQ_36	0.7928	1.2148	0.9327	1.2195	0.9145, 1.4794
CSAR-HiQ_51	0.7197	1.6798	1.3824	1.6791	1.3877, 1.9632
Validation	0.7735	1.2456	0.9434	1.2117	0.1961, 1.2958
Training	0.9133	0.7800	0.5626	0.7398	0.7634, 0.7966



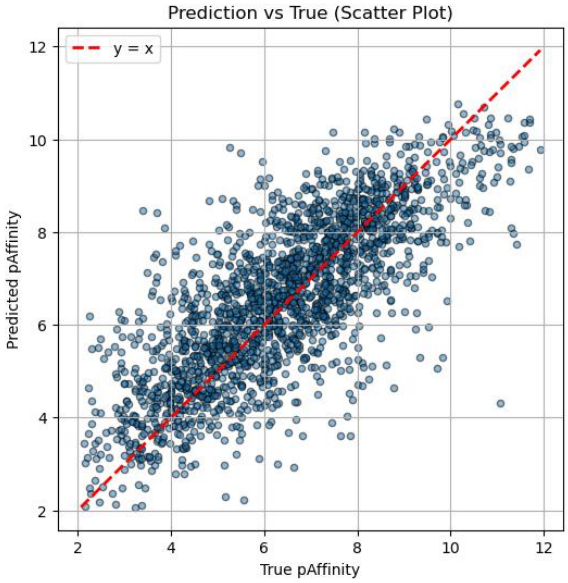


Figure 13, 14. Training / Validation 의 산점도

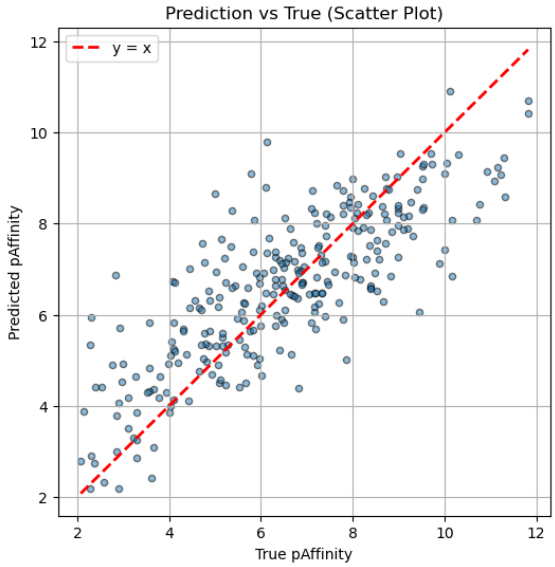
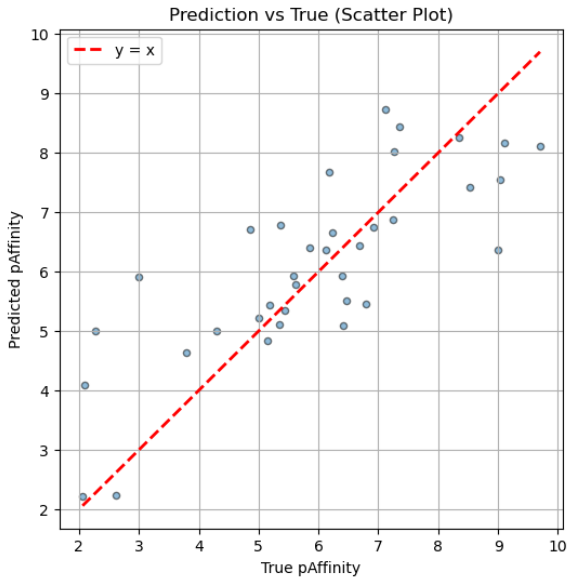


Figure 15. Core 2016 의 산점도



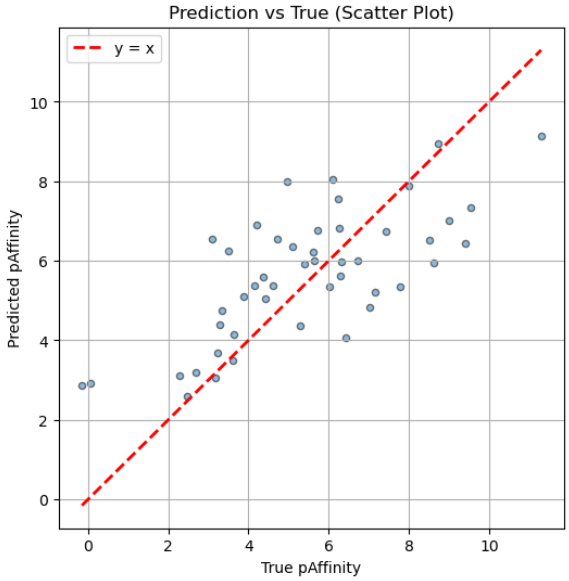


Figure 16, 17. CSAR-HiQ 36, 51 의 산점도

8.6.1 세부 결과 해석

Training / Validation

학습 세트에서는 R=0.91, RMSE=0.78, MAE=0.56 으로 매우 높은 정확도를 보였으며, 이는 모델이

학습 데이터에 대해 안정적으로 수렴했음을 보여준다. 검증 세트에서는 R=0.77, RMSE=1.25, MAE=0.94 로 다소 성능 저하가 있었지만 여전히 합리적인 수준의 예측력을 유지하였다. 이는 과적합이 크지 않고 일정 수준의 일반화 능력을 확보했음을 시사한다.

Core-2016

R=0.80, RMSE=1.30, MAE=1.01, SD=1.29 로 준수한 성능을 보였다. 특히 95% CI 가 1.18–1.41 범위로 나타나, 결과가 통계적으로 신뢰할 만한 수준임을 확인할 수 있었다.

CSAR-HiQ_36

R=0.79, RMSE=1.21, MAE=0.93, SD=1.22 로 비교적 우수한 성능을 보였다. 이전보다 개선된 수치로, 모델이 해당 데이터셋에서 안정적이고 균형 잡힌 예측을 수행했음을 보여준다.

CSAR-HiQ_51

R=0.72, RMSE=1.68, MAE=1.38, SD=1.68 로 상대적으로 낮은 정확도와 큰 오차를 기록하였다. 이는 일부 outlier 샘플의 영향과 함께, 데이터셋 자체가 난이도가 높은 특성을 반영한다.

8.6.2 종합 분석

모델은 학습 및 검증 세트에서 안정적인 성능을 확보하였으며, core-2016 과 CSAR-HIQ_36 에서는 비교적 우수한 결과를 보였다. 반면 CSAR-HIQ_51 에서는 성능 저하가 확인되었는데, 이는 난이도가 높은 데이터셋의 특성과 극값(outlier) 샘플의 영향을 받은 것으로 해석된다. 전반적으로 본 모델은 다양한 데이터셋에서 균형 잡힌 일반화 성능을 보였으나, 어려운 데이터셋과 outlier 처리 측면에서는 여전히 개선 여지가 있음을 확인할 수 있었다.

8.7. 기존 모델과의 비교

본 연구에서 제안한 Planet-X 결합 친화도 예측 모델은 초기 설계 단계에서 CAPLA 모델을 주요 참고 대상으로 삼았다.

CAPLA(2023)는 단백질–리간드 결합 친화도를 예측하기 위해 제안된 시퀀스 기반 딥러닝 모델로, 기존 접근법의 한계를 개선하고자 설계되었다.

기존의 물리 기반 방법은 높은 정확도를 제공하지만 계산 비용이 과도하며, 도킹 기반 방법은 효율적이지만 예측 정확도가 낮다. 전통적인 머신러닝 기법은 일정 수준의 성능을 달성했으나, 수작업으로 도출한 특징(feature)에 크게 의존한다는 한계가 있었다. 또한 기존 시퀀스 기반 딥러닝

모델들은 단백질과 리간드 서열을 입력으로 자동 표현 학습이 가능했지만, 두 표현을 단순히 결합(concatenation)하는 방식에 머물러 상호작용 정보를 충분히 반영하지 못했다.

이러한 한계를 극복하기 위해 CAPLA 는 단백질 전체 서열과 리간드 서열뿐 아니라 실제 결합 포켓(binding site) 서열을 입력에 포함하였다. 이후 교차 어텐션(cross-attention)을 통해 포켓–리간드 상호작용을 학습하고, dilated convolution 을 활용하여 단백질, 포켓, 리간드 각각에서 장거리 의존성과 다중 스케일 특징을 추출하였다. 학습된 표현은 통합되어 fully connected layer 에 입력되며 최종적으로 결합 친화도를 예측한다.

CAPLA 는 여러 벤치마크 데이터셋 및 외부 독립 테스트에서 기존 최신 모델 대비 우수한 성능을 보였으며, 어텐션 해석을 통해 결합 포켓 내의 기능적으로 중요한 잔기를 식별할 수 있다는 점에서 해석 가능성도 확보하였다. 따라서 CAPLA 는 포켓 정보를 적극적으로 활용하면서도 시퀀스 기반으로 동작하여 계산 효율성, 정확도, 해석 가능성을 모두 갖춘 결합 친화도 예측 모델로 평가된다.

비록 Planet-X 와 CAPLA 사이에는 여러 차이점이 존재하지만, 두 모델이 모두 서열 기반이라는 점과 CNN 및 Transformer 구조를 활용한다는 점에서 성능 및 구조적 차이를 비교하는 것은 본 연구의 성과를 평가하는 데 중요한 의의를 가진다.

다음은 CAPLA 와 Planet-X 결합 친화도 모델의 모델 구조를 제외한 차이점이다.

1. 전처리 시 차이

   CAPLA 는 단백질 서열을 최대 1,000 residues 까지만 처리하며, 이를 초과하는 경우 초과 부분을 잘라내는 방식을 적용하였다. 반면, Planet-X 는 입력 범위를 1,500 residues 까지 확장하고 이를 초과하는 단백질은 데이터셋에서 제외하였다.

   리간드의 경우 두 모델 모두 최대 길이를 150 으로 설정했으나, CAPLA 는 초과 부분을 절단하여 사용한 반면 Planet-X 는 초과 샘플 자체를 제외하였다. 또한 Planet-X 는 전처리 과정에서 HETATM 미변환, kekulization 실패와 같은 이슈로 일부 샘플이 제거되었으며, 이는 단백질·리간드 길이 필터링 이전 단계에서 발생하였다.

   CAPLA 는 전처리 코드가 공개되지 않아 본 연구에서는 동일한 과정을 재현할 수 없었다. 대신 PDBbind 데이터셋에서 직접 단백질 서열과 리간드를 추출·정제하는 방식을 적용하였다. 이로 인해 Planet-X 의 최종 학습 데이터셋 크기는 CAPLA 대비 다소 줄어들었으나, 단백질 서열과 리간드 SMILES 를 활용한다는 점에서 두 모델의 근본적인 입력 데이터 성격에는 차이가 없다고 판단된다. 따라서 성능 비교 시 전처리 차이는 참고 요소로만 고려하였다.

한편, 학습 및 검증 데이터의 구성 과정은 동일하다.

2. 입력 표현 차이

○ Binding Region (BR) 사용 여부: CAPLA 는 학습 과정에서 BR 정보를 입력으로 활용하였으나, Planet-X 는 웹 응용의 편의성과 확장성을 고려하여 BR 정보를 제외하였다.

○ 사전 학습 모델 활용: CAPLA 와 달리 Planet-X 는 단백질과 리간드 입력에 각각 ESM, ChemBERTa 와 같은 사전 학습(pre-trained) 모델을 적용하여 보다 풍부하고 일반화된 표현 학습을 가능하게 하였다.

○ 물리화학적 특성 반영: CAPLA 는 단백질의 지역적 물리화학적 특성만을 사용한 반면, Planet-X 는 여기에 더해 리간드의 전역적 물리화학적 특성까지 포함하여 단백질과 리간드 양쪽의 특성을 균형 있게 반영하였다. 이를 통해 입력 표현력을 강화하였다.

3. 성능 비교

앞선 8.6 절에서는 PDBbind 2020 기반 성능 평가를 다루었으나, 본 절에서는 CAPLA 와의 직접 비교를 위해 PDBbind 2016 을 활용하였다. 학습 데이터 구성은 CAPLA 와 동일하게, refined set 에서 무작위로 1,000 개를 검증용(validation set)으로 사용하고 나머지 general set 을 학습용(train set)으로 활용하였다. 테스트셋(test set)과 일반화 평가 데이터셋 역시 CAPLA 와 동일하게 사용하였다. 따라서 테스트셋으로는 core 2016, CASF-2013 을, 일반화 성능 평가에는 CSAR HIQ 51, CSAR HIQ 36 을 적용하였다. 모든 테스트셋은 학습 및 검증 과정에서 철저히 제외하여 독립적인 평가를 수행하였다.

차이점 1 번에서 기술하였듯, CAPLA 의 전처리 코드를 그대로 재현할 수는 없었으나, PDBbind 데이터셋에서 직접 서열과 리간드를 추출·정제하는 방식을 적용함으로써 가능한 한 유사한 조건에서 비교를 수행하였다. 이 과정에서 최종 데이터셋 크기가 일부 줄어들기는 했지만, 입력 데이터의 성격은 동일하므로 성능 비교는 유효하다.

Table 4 에서는 CAPLA 와 Planet-X 의 성능을 다양한 테스트 및 검증 데이터셋(core-2016, CASF-2013, CSAR-HIQ_51, CSAR-HIQ_36, Validation, Training)에서 단독으로 비교하였다.

Table 4. Comparative Performance of CAPLA and Planet-X across Multiple Datasets

Dataset	Model	R	RMSE	MAE	SD
CSAR-HiQ 36	CAPLA	**0.704**	1.454	1.160	**1.420**
	Planet-X ⬆	0.6946	**1.4263**	**1.1269**	1.4393
CSAR-HiQ 51 (51)	CAPLA	0.686	1.848	1.550	1.701
CSAR-HiQ 51 (45)	Planet-X ⬆	**0.7535**	**1.5829**	**1.2452**	**1.5899**
CASF-2013 (195)	CAPLA ⬆	**0.770**	**1.446**	**1.155**	**1.436**
CASF-2013 (180)	Planet-X	0.7310	1.5214	1.2284	1.5202
Core-2016 (290)	CAPLA ⬆	**0.843**	**1.200**	**0.966**	**1.170**
Core-2016 (279)	Planet-X	0.8107	1.2847	1.0191	1.2748
Validation	CAPLA	**0.771**	1.338	1.034	1.307
	Planet-X	0.7612	**1.3223**	**0.9931**	**1.2734**
Training(11906)	CAPLA	0.867	0.968	0.755	0.931
Training(10111)	Planet-X	0.8931	0.8374	0.5900	0.8313

Note: 더 우수한 성능을 보인 수치는 굵게 표시하였으며, 종합적으로 우세한 모델은 화살표(⬆)로 표시하였다. 또한 전처리 과정의 차이로 인해 동일 데이터셋임에도 샘플 수가 달라진 경우, 해당 실사용 샘플 수를 괄호 안에 병기하였다.

데이터셋별 비교

- **CSAR-HiQ_36**

  Planet-X 는 R=0.6946, RMSE=1.4263, MAE=1.1269 를 기록하여 CAPLA(R=0.704,

RMSE=1.454, MAE=1.160)와 유사한 수준의 성능을 보였다. R 에서는 CAPLA 가 소폭 우세했으나, 오차 지표(RMSE·MAE)는 Planet-X 가 더 나은 결과를 달성하여 안정성을 보여주었다.

- **CSAR-HiQ_51**

  Planet-X 는 R=0.7535, RMSE=1.5829, MAE=1.2452 로 CAPLA(R=0.686, RMSE=1.848, MAE=1.550)를 뚜렷하게 상회하였다. 이는 전처리 차이가 존재했음에도 Planet-X 가 보다 안정적이고 일관된 예측 성능을 확보했음을 시사한다.

- **CASF-2013**

  CAPLA 는 R=0.770, RMSE=1.446, MAE=1.155 로 Planet-X(R=0.7310, RMSE=1.5214, MAE=1.2284)보다 우수한 결과를 보였다. 이는 CASF 와 같은 정제된 벤치마크 환경에서 CAPLA 가 상대적 강점을 지님을 보여준다.

- **core-2016**

  CAPLA 는 R=0.843, RMSE=1.200, MAE=0.966 으로 Planet-X(R=0.8107, RMSE=1.2847, MAE=1.0191)보다 더 우수하였다. 이는 CAPLA 가 소규모이면서도 고품질의 데이터셋에서 높은 성능을 발휘함을 의미한다.

- **Validation set**

  Planet-X 는 RMSE=1.3223, MAE=0.9931 로 CAPLA(RMSE=1.338, MAE=1.034)보다 다소 나은 결과를 보였으며, R 에서는 약간 낮았다(0.7612 vs. 0.771). 전반적으로 두 모델은 유사한 수준의 성능을 보였다.

- **Training set**

  Planet-X 는 R=0.8931, RMSE=0.8374, MAE=0.5900 으로 CAPLA(R=0.867, RMSE=0.968, MAE=0.755) 대비 명확한 우위를 보였다. 이는 Planet-X 가 과적합 없이 안정적으로 학습되었음을 보여준다.

종합적으로, CAPLA 는 core-2016 및 CASF-2013 과 같은 소규모·정제된 벤치마크에서 강점을 보였고, Planet-X 는 CSAR 계열 및 Validation set 에서 안정적이고 개선된 성능을 나타냈다. 특히 Planet-X 는 BR 정보를 활용하지 않고도 일정 수준 이상의 성능을 확보하여, 입력 단순화와 표현력 강화, 그리고 응용 확장성 측면에서 의미 있는 성과를 제시하였다.

다음은 타모델과의 비교로 CAPLA 논문에서 제시되어 있는 표를 그래프 형태로 나타낸 것이다.

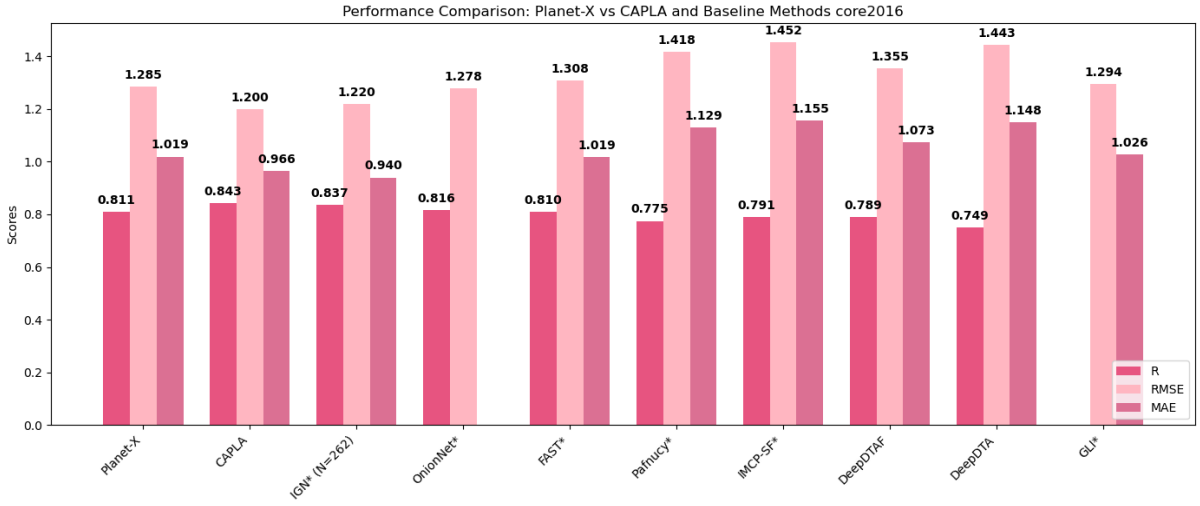


Figure 18. Planet-X, CAPLA, IGN (N=262), OnionNet, FAST*, Pafnucy*, IMCP-SF*, DeepDTAF, DeepDTA, GLI* on core-2016 test set

Planet-X 는 R=0.8107, RMSE=1.2847, MAE=1.0191 을 기록하며 여러 baseline 모델들을 상회하였다. DeepDTA(R=0.749, RMSE=1.443, MAE=1.148)는 낮은 상관성과 높은 오차를 보였고, DeepDTAF(R=0.789, RMSE=1.355, MAE=1.073)도 Planet-X 보다 전반적으로 열세였다. Pafnucy*(R=0.775, RMSE=1.418, MAE=1.129) 역시 Planet-X 에 미치지 못했으며, OnionNet*(R=0.816, RMSE=1.278)은 Planet-X 와 유사한 수준이었다. FAST*(R=0.810, RMSE=1.308, MAE=1.019)도 근접한 결과를 보였고, 구조 기반 모델인 IMCP-SF*(R=0.791, RMSE=1.452, MAE=1.155)는 오히려 더 큰 오차를 나타냈다. CAPLA(R=0.843, RMSE=1.200, MAE=0.966)는 Planet-X 보다 다소 우세했지만, Planet-X 는 baseline 모델들을 일관되게 초월하며 CAPLA 에 근접한 수준을 유지하였다.

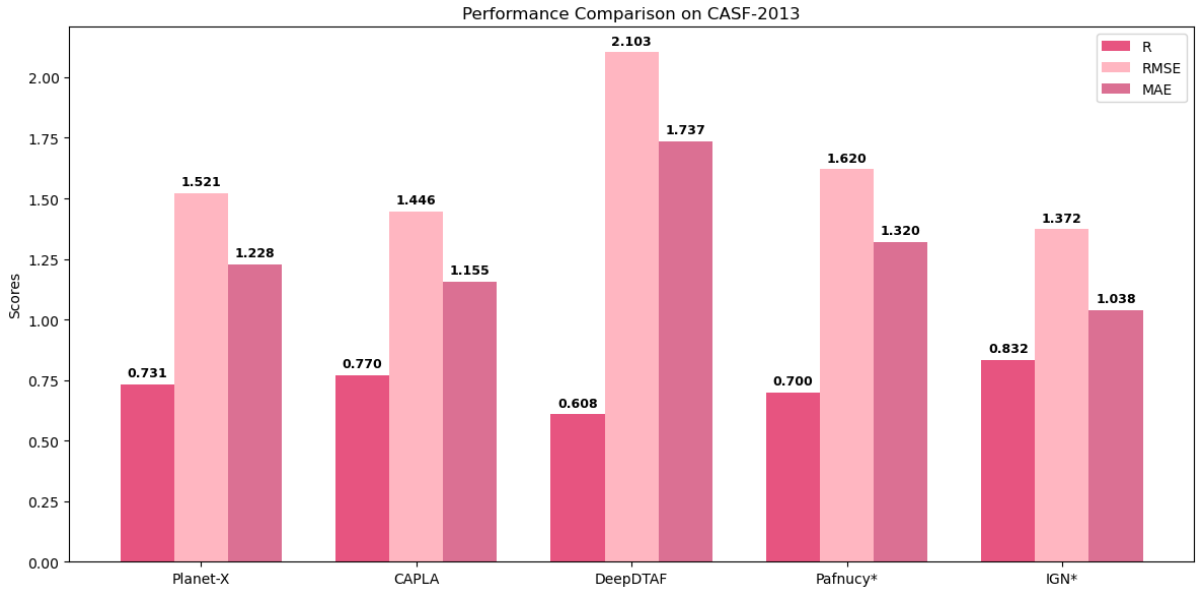


Figure 19. Planet-X, CAPLA, DeepDTAF, Pafnucy, IGN on CASF-2013 test set

Planet-X 는 R=0.7310, RMSE=1.5214, MAE=1.2284 를 기록하여 기존 baseline 모델들보다 확연히 우수한 결과를 보였다. DeepDTAF(R=0.608, RMSE=2.103, MAE=1.737)은 낮은 상관성과 큰 오차를 보였으며, CAPLA(R=0.770, RMSE=1.446, MAE=1.154)는 Planet-X 보다 다소 높은 성능을 달성했다. 따라서 Planet-X 는 CAPLA 와 비교했을 때는 약간 열세였지만, DeepDTAF 대비 안정적이고 향상된 성능을 유지하여 서열 기반 접근법으로도 충분히 경쟁력이 있음을 보여주었다.

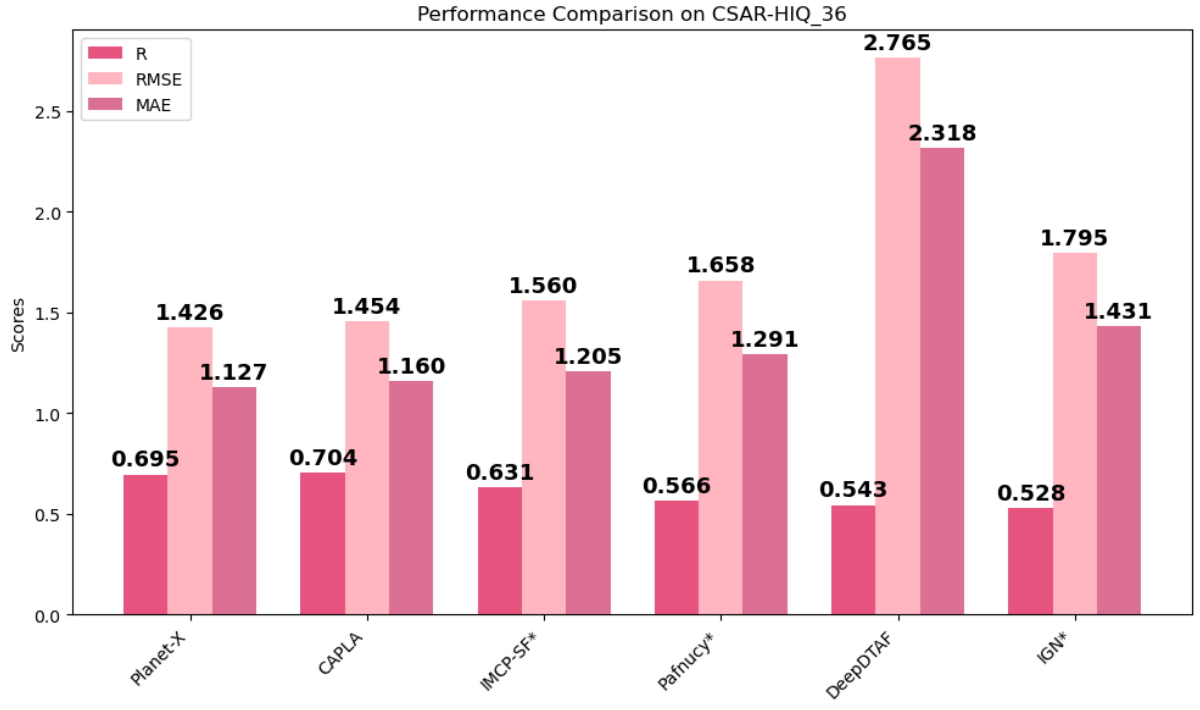


Figure 20. Planet-X, CAPLA, DeepDTAF, Pafnucy, IGN**, IMCP-SF* on CSAR-HIQ_36 test set

Planet-X 는 R=0.6946, RMSE=1.4263, MAE=1.1269 를 기록하며 기존 baseline 모델들을 크게 상회하였다. DeepDTAF(R=0.543, RMSE=2.765, MAE=2.318), Pafnucy*(R=0.566, RMSE=1.658, MAE=1.291), IGN*(R=0.528, RMSE=1.795, MAE=1.431)은 모두 낮은 상관성과 높은 오차를 보였으며, 구조 기반 모델인 IMCP-SF*(R=0.631, RMSE=1.560, MAE=1.205) 역시 Planet-X 보다 열세를 보였다. CAPLA(R=0.704, RMSE=1.454, MAE=1.160)는 Planet-X 와 유사한 수준이었으며, R 에서는 소폭 높았으나 RMSE 와 MAE 에서는 Planet-X 가 더 우수했다. 따라서 Planet-X 는 CAPLA 와 견줄 만한 성능을 유지하면서 baseline 과 구조 기반 모델 모두를 일관되게 초월하였다.

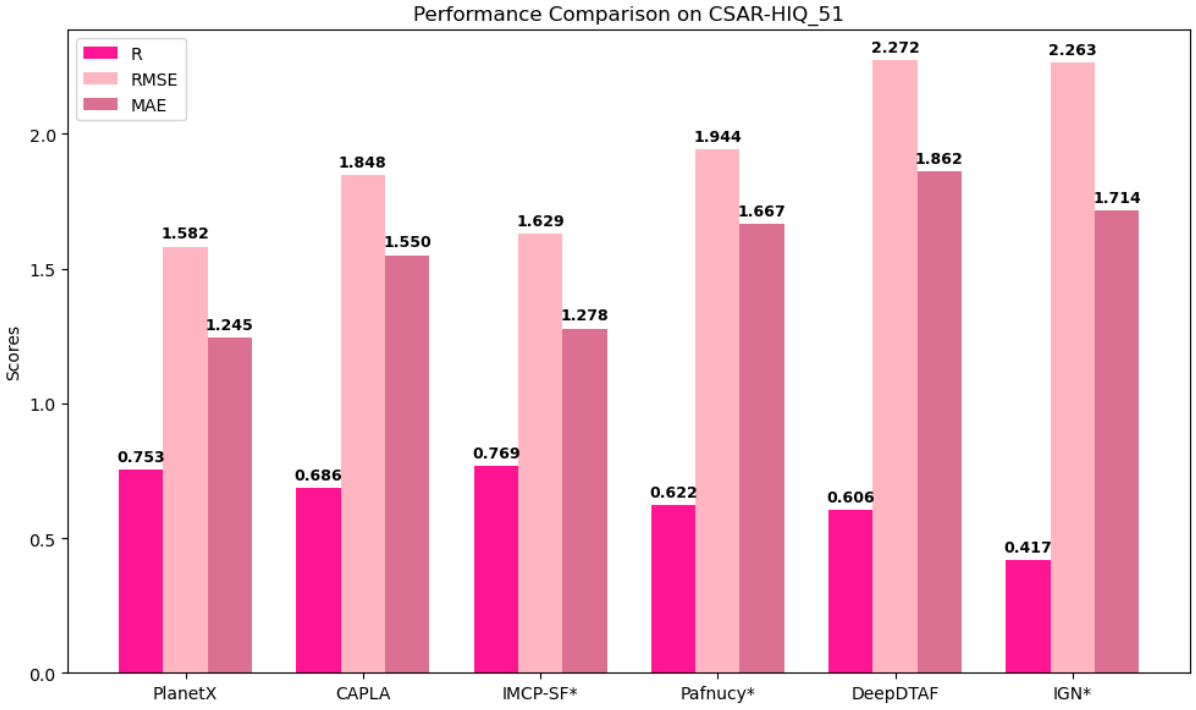


Figure 21. Planet-X, CAPLA, DeepDTAF, Pafnucy*, IGN*, IMCP-SF* on CSAR_HIQ_51(45) test set

Planet-X 는 R=0.7535, RMSE=1.5829, MAE=1.2452 를 기록하며 기존 baseline 모델들을 뚜렷하게 상회하였다. DeepDTAF(R=0.606, RMSE=2.272, MAE=1.862), Pafnucy*(R=0.622, RMSE=1.944,

MAE=1.667), IGN*(R=0.417, RMSE=2.263, MAE=1.714)은 모두 낮은 예측 정확도를 보여 Planet-X 와 큰 격차를 보였다. CAPLA 역시 R=0.686, RMSE=1.848, MAE=1.550 으로 Planet-X 보다 전반적으로 열세였으며, 특히 RMSE 와 MAE 에서 확연히 뒤처졌다. 한편, 구조 기반 모델인 IMCP-SF*(R=0.769, RMSE=1.629, MAE=1.278)는 Planet-X 와 유사하거나 약간 높은 R 을 기록하여, 특정 데이터셋에서는 여전히 구조 정보를 활용한 접근이 강점을 가질 수 있음을 시사한다.

CAPLA 는 R=0.686, RMSE=1.848, MAE=1.550 으로 Planet-X 보다 전반적으로 낮은 성능을 보였으며, 특히 오차 지표에서 큰 차이가 확인되었다. 따라서 Planet-X 는 CAPLA 대비 개선된 결과를 달성했을 뿐 아니라, 다수의 baseline 모델들을 크게 상회하면서도 구조 기반 접근과 비교 가능한 수준에 도달하였다.

종합적으로, CAPLA 는 2023 년 발표 당시 core-2016 과 같은 고품질 벤치마크에서 최고 성능을 기록하였고, Planet-X 는 CAPLA 에 근접하거나 일부 데이터셋(CSAR-HIQ_51)에서는 이를 능가하였다. Planet-X 는 DeepDTA 와 같이 단순한 입력 구조를 유지하면서도, 사전학습 모델과 전역 물리화학적 특성, CNN·Transformer 기반 모듈을 결합하여 표현력을 강화하였다. 그 결과 baseline 대비 일관되게 향상된 성능을 보였으며, CAPLA 및 구조 기반 모델과의 비교에서도 충분히 경쟁력 있는 대안이 될 수 있음을 보여준다.

특히 Planet-X 는 BR 정보를 사용하지 않고도 CAPLA 와 유사하거나 일부 데이터셋에서는 이를 능가하는 성능을 보여, 실제 응용 환경에서 데이터 요구 조건을 낮추면서도 실용적인 성능을 제공할 수 있다는 점에서 중요한 의의를 가진다.

8.8. 향후 보완 및 한계점

● 데이터 불균형 문제 및 극값 오차

본 연구에서 제안한 단백질–리간드 결합 친화도 예측 모델은 중간 구간(pAff 4–8)에서 안정적인 예측 성능을 보였으나, 데이터가 희소한 극값 영역(pAff ≥ 10)에서는 예측 오차가 크게 증가하는 경향을 보였다. 이는 다음과 같은 한계점과 향후 연구 방향을 시사한다.

우선, 본 연구에 사용된 데이터셋은 pAff 6–8 구간에 데이터가 편중되어 있어, 극값 영역에 해당하는 샘플 수가 상대적으로 매우 적다. 이로 인해 모델이 극값 구간에서 충분한 일반화 능력을 학습하지 못한 것으로 판단된다. 이러한 데이터 불균형 문제는 극값 구간에서의 예측 성능 저하로 이어져, 실제 응용 시 예측 신뢰도를 낮출 수 있다.

따라서 향후 연구에서는 극값 영역의 데이터를 보완하기 위한 다양한 접근법이 필요하다. 예를 들어, 데이터 증강 기법(oversampling), 추가적인 외부 데이터셋 통합, 혹은 구간별 손실 함수 가중치 조절을 통해 희소한 구간에 대해 모델이 보다 집중적으로 학습할 수 있도록 할 수 있다.

아래 표들은 본 연구에서 평가한 다양한 데이터셋의 친화도 값 구간별 성능 지표를 요약한 것으로, 극값 구간에서 RMSE 와 MAE 가 크게 증가하는 현상과 함께 데이터 편중이 명확히 드러남을 확인할 수 있다.

- **Train** (14587)

Range	Count	RMSE	MAE
[2, 4)	1360	0.8674	0.5949
[4, 6)	4281	0.8337	0.6021

[6, 8)	5987	0.6843	0.5238
[8, 10)	2692	0.6638	0.4854
[10, inf)	**267**	**1.6736**	**1.3030**

Table 5. Train 데이터셋 범위별 CAPLA 성능 (총 14,587 개 샘플)

- **Validation** (2000)

Range	Count	RMSE	MAE
[2, 4)	220	1.5821	1.2026
[4, 6)	613	1.2265	0.9384
[6, 8)	743	1.1066	0.8624
[8, 10)	353	1.2083	0.8744
[10, inf)	**71**	**1.6988**	**1.3732**

Table 6. Validation 데이터셋 범위별 CAPLA 성능 (총 2,000 개 샘플)

- **Core-2016** (279)

Range	Count	RMSE	MAE
[2, 4)	37	1.5416	1.1873
[4, 6)	78	1.3674	1.0707
[6, 8)	88	1.0109	0.7835
[8, 10)	59	1.1529	0.9068

[10, inf)	**17**	**0.9851**	**1.8357**

Table 7. core-2016 데이터셋 범위별 CAPLA 성능 (총 279 개 샘플)

- **CSAR-HiQ 36** (36)

Range	Count	RMSE	MAE
[2, 4)	6	1.8609	1.5006
[4, 6)	11	0.7764	0.5595
[6, 8)	13	0.9550	0.8109
[8, 10)	6	1.5199	1.3128
[10, inf)	**0**	**nan**	**nan**

Table 8. CSAR-HiQ 36 데이터셋 범위별 CAPLA 성능 (총 36 개 샘플)

- **CSAR-HiQ 51** (45)

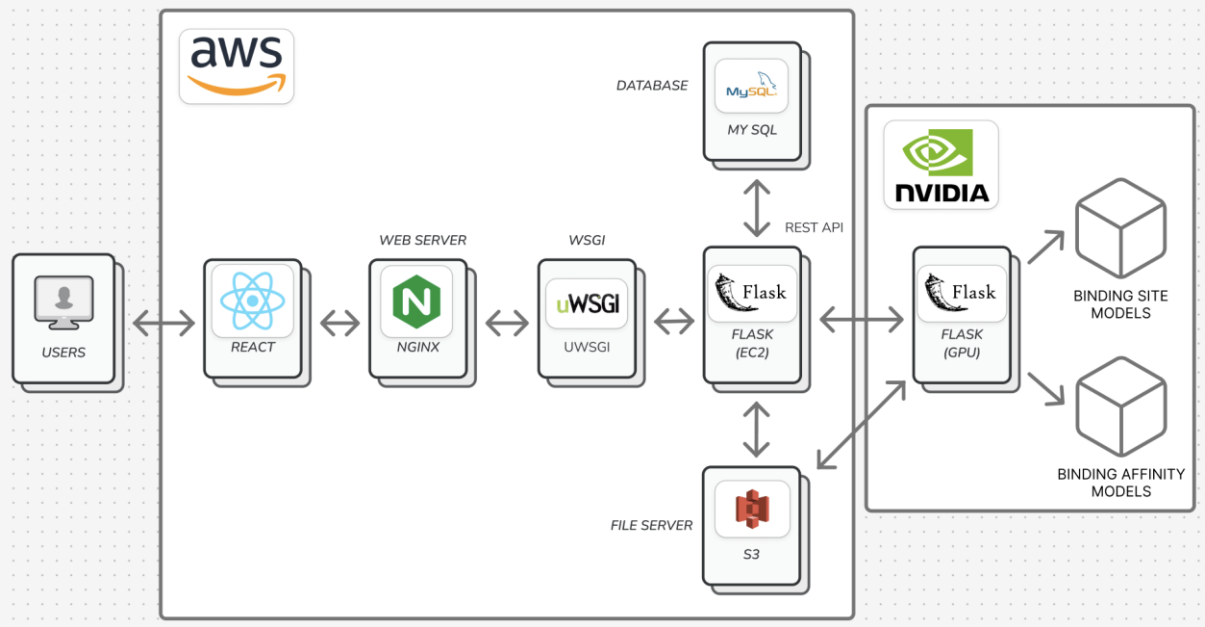
Range	Count	RMSE	MAE
[2, 4)	12	1.4640	1.0587
[4, 6)	13	1.4703	1.2424
[6, 8)	12	1.5285	1.3257
[8, 10)	7	2.0268	1.7349
[10, inf)	**1**	**2.1708**	**2.1708**

Table 9. CSAR-HiQ 51 데이터셋 범위별 CAPLA 성능 (총 45 개 샘플)

9. 웹 서비스 설계 및 구현

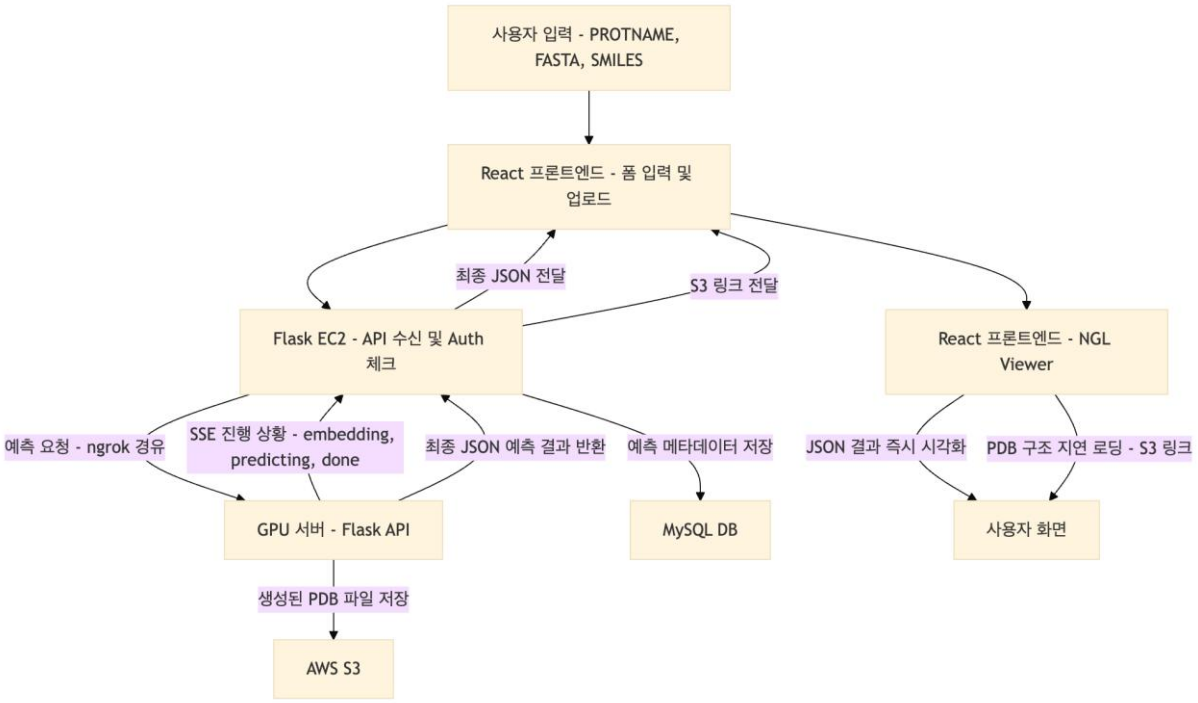
9.1. 서비스 구조 설계

9.1.1. 서비스 아키텍쳐 다이어그램



본 서비스의 전체 아키텍처 구조는 다음과 같다. React 기반 프론트엔드는 사용자 입력과 예측 결과 시각화를 담당하며, EC2 상의 Flask 백엔드는 사용자 인증 및 API 요청을 처리한다. ESM 기반 모델은 크기가 커서 EC2 에서 직접 서빙하기 어려웠기 때문에 GPU 서버에 별도 배치하여 예측 연산을 수행하였다. 예측 과정에서 생성되는 구조 파일은 AWS S3 에 저장되며, 사용자 정보와 예측 메타데이터는 MySQL 데이터베이스에 기록된다.

9.1.2. 데이터 흐름도



본 서비스의 데이터 흐름은 다음과 같다. 사용자가 단백질 서열(FASTA)과 리간드(SMILES)를 입력하면 React 프론트엔드에서 이를 수집하여 Flask(EC2) 백엔드로 전달한다. EC2 는 API 요청을 검증하고 GPU 서버로 예측 요청을 보낸다. GPU 서버는 SSE 를 통해 진행 상황(embedding → predicting → done)을 스트리밍으로 반환하고, 최종 예측 결과를 JSON 형식으로 전달한다. 예측 결과 메타데이터는 MySQL DB 에 저장되고, 생성된 PDB 구조 파일은 AWS S3 에 저장된다. 이후 프론트엔드는 JSON 결과를 즉시 시각화하며, PDB 구조는 S3 링크를 통해 NGL Viewer 에서 로딩되어 사용자 화면에 표시된다.

9.2. 서비스 구현 및 평가

9.2.1. 프론트엔드 (React + NGL Viewer)

9.2.1.1. 주요 페이지

본 서비스의 프론트엔드는 React 를 기반으로 개발하였으며, 사용자가 서비스를 쉽게 이해하고 사용할 수 있도록 네 가지 주요 페이지로 구성하였다.

(1) Intro 페이지

- 서비스 소개와 목적을 간단히 보여주는 첫 화면이다.
- 사용자가 로그인하기 전, 단백질–리간드 결합 예측 서비스의 필요성과 활용 가능성을 설명한다.

• Orbitron 폰트와 네온 테마를 적용하여 시각적으로 일관된 느낌을 주었다.

(2) Signup/Login 페이지

• 회원가입(Signup): 이메일 인증을 통해 회원가입을 진행하며, Flask-Mail 을 이용해 실제 메일로 인증코드를 발송한다.

• 로그인(Login): JWT(Json Web Token) 방식을 사용하여 보안 세션을 관리한다.

• 로그인하지 않은 사용자는 예측 기능에 접근할 수 없도록 제한하였다.

• 이를 통해 사용자별 데이터 보호와 서비스 안정성을 확보하였다.

(3) PredictionView 페이지

• 단백질 서열(FASTA)과 리간드(SMILES)를 입력받아 예측을 실행하는 메인 화면이다.

• Axios 를 통해 Flask 백엔드로 요청을 전달하며, GPU 서버와는 SSE 방식으로 연결하여 예측 진행 상황을 실시간으로 보여준다. (예: embedding → predicting → done)

• 예측 결과(JSON)는 바로 표시되며, PDB 구조 파일은 S3 에서 불러와 NGL Viewer 로 3D 시각화를 제공한다.

(4) MyPage

• 사용자가 과거에 실행한 예측 결과를 확인하고 다시 시각화할 수 있는 페이지다.

• MySQL 에 저장된 Prediction/Structure 테이블을 불러와 사용자별 이력을 관리한다.

• 이를 통해 단순히 한 번 예측하는 서비스가 아니라, 연구 기록을 남기고 재활용할 수 있는 시스템을 구현하였다.

9.2.1.2. Axios 를 통한 API 연동, 파일 업로드, 예측 결과 출력 구현

프론트엔드와 백엔드 간 통신은 Axios 라이브러리를 이용하여 구현하였다. Axios 는 Promise 기반 HTTP 클라이언트로, 비동기 방식으로 데이터를 송수신할 수 있어 React 환경에서 많이 활용된다.

(1) API 연동

• 사용자가 입력한 단백질 서열(FASTA)과 리간드(SMILES) 정보를 Axios 를 통해 Flask 백엔드 서버로 전송한다.

• 예측 요청은 POST /predict 엔드포인트로 전달되며, JWT 토큰을 포함해 인증된 사용자만 접근할 수 있도록 설계하였다.

• GPU 서버와의 연결 과정에서 Flask 는 SSE(Server-Sent Events) 를 통해 예측 진행 상황(embedding → predicting → done)을 수신하고, 이를 다시 프론트엔드로 전달하여 화면에 표시한다.

(2) 파일 업로드

• 사용자는 로컬에서 준비한 입력 파일을 업로드할 수 있으며, Axios 를 활용해 multipart/form-data 형식으로 백엔드에 전송된다.

• 업로드된 파일은 Flask 서버에서 검증 후 AWS S3 에 저장되고, 관련 메타데이터는 MySQL 에 기록된다.

• 이를 통해 사용자는 브라우저 환경에서 별도의 프로그램 설치 없이 데이터 업로드가 가능하다.

(3) 예측 결과 출력

• 예측 결과는 JSON 형식으로 프론트엔드에 반환되며, 즉시 화면에 표시된다.

• 결과에는 단백질 결합 부위(binding site)와 결합 친화도(binding affinity) 정보가 포함된다.

• AlphaFold 기반 구조 생성은 시간이 오래 걸리기 때문에, 예측 결과(JSON)는 먼저 반환하고 이후 S3 에 저장된 구조 파일(PDB 링크)을 불러와 시각화 페이지에서 확인할 수 있도록 하였다.

9.2.1.3. NGL Viewer 기반 3D 시각화

단백질–리간드 예측 결과를 효과적으로 보여주기 위해, 본 서비스에서는 웹 기반 3D 구조 시각화 도구인 NGL Viewer 를 사용하였다.

NGL Viewer 는 웹 브라우저에서 바로 동작하는 WebGL 기반 3D 시각화 라이브러리로, 브라우저 환경에서 별도의 설치 없이 단백질 구조(PDB 파일)와 리간드 분자를 렌더링할 수 있다는 장점이 있다. 이 점은 사용자가 단순히 웹 브라우저만으로 예측 결과를 직관적으로 확인할 수 있다는 서비스 목적과 잘 맞는다.

(1) 구현 방식

I 레이어 구성

• Layer 1 (단백질 전체 구조): 단백질의 전체적인 구조를 하늘색으로 직관적으로 볼 수 있도록 하였다.

• Layer 2 (예측된 바인딩 사이트): 모델이 예측한 모든 binding site residue 를 빨간색으로 강조하여, 단백질 내 결합 후보 영역을 한눈에 파악할 수 있도록 하였다.

• Layer 3 (선택된 바인딩 사이트 강조): 사용자가 시퀀스에서 특정 residue 를 선택하면, 해당 위치를

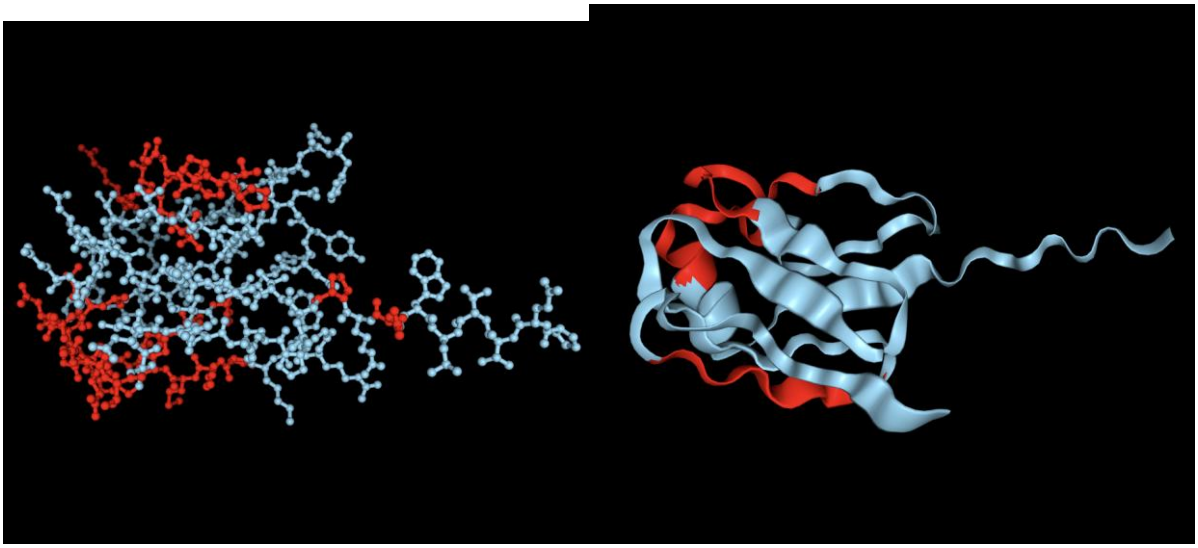
흰색으로 표시하여 다른 binding site 와 구분되게 하였다. 이를 통해 "예측된 부위 중 내가 확인하고자 하는 부분"을 명확히 볼 수 있다.

ㅣ 강조 효과

• 사용자가 마우스로 바인딩 사이트 중 일부를 click 했을 때 해당 부위가 반응하도록 구현해, 구조 탐색이 가능하도록 만들었다.

• 선택된 바인딩 사이트에는 flicker 효과를 적용하여, 단순히 색상 강조만 하는 것보다 가시성이 강화되도록 하였다.

(2) 구현 과정에서의 문제와 해결



- Cartoon 표현 한계: Cartoon 은 백본(Cα 를 잇는 연속 궤적과 2 차 구조) 을 기반으로 리본을 생성하므로, 구조에 불연속 구간(미예측/저신뢰 루프, 체인 단절, 비표준 잔기·누락 원자 등)이 있으면 리본이 끊기거나 일부가 사라져 결합 부위 시각화의 정확성이 떨어진다. 바인딩 사이트들 중 이런 구간이 존재해 Cartoon 으로는 잔기 단위 강조와 측쇄(사이드체인) 확인이 어려웠기에 ball+stick 을 채택했다. 이 방식은 2 차 구조 추정에 의존하지 않고 실제 원자 좌표만으로 그리기 때문에 백본이 부분적으로 끊겨도 표현이 안정적이고, 개별 잔기·측쇄를 정확히 표시할 수 있어 바인딩 사이트를 색상/효과로 명확히 강조하는 데 적합했다.

- AlphaFold 구조 반영 지연: 구조 생성 시간이 길어 PDB 파일 로딩은 늦게 이루어졌으며, 대신 예측 결과(JSON)는 먼저 반환하여 NGL Viewer 에서 빠르게 확인할 수 있도록 처리하였다.

9.2.1.4. MyPage 에서 사용자가 수행한 과거 예측 결과 조회 및 재시각화

MyPage 는 사용자가 지금까지 실행한 예측 결과를 모아볼 수 있는 개인 기록 관리 기능을 제공한다. 단순히 한 번 예측하고 끝나는 것이 아니라, 사용자가 과거 데이터를 불러와 다시 확인하거나 재시각화할 수 있다는 점에서 연구의 재현성과 활용성을 높이는 역할을 한다.

(1)데이터 관리 방식

- 각 사용자가 실행한 예측 결과는 MySQL 데이터베이스의 Prediction 및 Structure 테이블에 저장된다.
- 예측 결과(JSON)는 사용자 ID 와 연결되어 저장되며, 생성된 PDB 구조 파일은 AWS S3 에 업로드된다.
- MyPage 는 이 데이터베이스와 연동되어 로그인한 사용자에게 본인 기록만을 불러와 보여준다.

(2) 조회 기능

- 사용자는 날짜, 단백질 이름, 리간드 이름 등을 기준으로 과거 예측 기록을 확인할 수 있다.
- 각 기록에는 예측된 바인딩 사이트 리스트, 친화도 값이 표시된다.

(3) 재시각화 기능

- 저장된 PDB 파일을 불러와 다시 NGL Viewer 에서 확인할 수 있다.
- 예측 당시와 동일한 방식으로 바인딩 사이트 하이라이트, 흰색 선택 residue 표시, flicker 효과가 적용된다.
- 이를 통해 연구자가 필요할 때 언제든 특정 예측을 다시 열어 확인하거나, 다른 결과와 비교할 수 있다.

(4) 설계 의도

- 단순히 일회성 예측 서비스가 아닌, 사용자별 연구 데이터베이스 역할을 제공한다.
- 연구 과정에서 실험 기록을 보존하고 관리하듯, MyPage 는 사용자가 수행한 예측 보관함으로 기능한다.

9.2.2. 백엔드 (Flask API)

9.2.2.1. 주요 엔드포인트

엔드포인트	주요 기능	Input (입력값)	Output (반환값)	연결된 프론트 페이지
/signup	회원가입	이메일, 비밀번호, 이메일 인증 코드	가입 완료 여부, 인증 메일 발송 결과	Signup Page
/login	로그인	이메일, 비밀번호	JWT 토큰(세션 유지), 로그인 성공 여부	Login Page
/input	단백질·리간드 입력	단백질 이름(PROTNAME), 단백질 서열(FASTA), 리간드(SMILES)	입력 검증 결과(형식 오류 시 에러 반환)	Input Page
/predict	단백질–리간드 예측 요청	(Input Page에서 전달된 값)	SSE 기반 진행 상황(embedding → predicting → done), 최종 JSON 결과(binding site 리스트, binding affinity 값)	PredictionView Page
/archive	과거 예측 결과 조회	사용자 ID (JWT 토큰 인증)	JSON 결과 리스트, S3에 저장된 PDB 구조 파일 링크	MyPage (Archive)

9.2.2.2. JWT 기반 인증 + 이메일 인증 (Flask-Mail)

백엔드는 JWT(Json Web Token)를 사용하여 인증 상태를 관리한다.

- 회원가입(/signup) 시 Flask-Mail 을 통해 실제 이메일로 인증 코드를 발송하고,
- 사용자가 해당 코드를 제출하면 계정을 활성화한다.
- 로그인(/login)에 성공하면 서버가 JWT 를 발급하고, 프론트엔드는 Axios 인터셉터를 통해 모든 보호된 요청에 Authorization: Bearer <token> 헤더를 자동 첨부한다.

구현 과정에서 겪은 문제와 해결:

(1) 토큰 만료/분실 문제: 초기에는 만료된 토큰을 사용했을 때 조용히 실패하는 문제가 있었다. 이를 개선하여 서버가 명시적으로 401 Unauthorized 를 반환하도록 하고, 프론트엔드에서는 인터셉터를 통해 토스트 알림을 띄운 뒤 로그인 페이지로 이동시키도록 처리했다.

(2) 권한 제어: /predict 와 /archive 엔드포인트는 반드시 로그인된 사용자만 접근할 수 있도록 인증 미들웨어를 Flask 라우트에 적용했다. 이를 통해 사용자 데이터 보호를 강화하였다.

9.2.2.3. GPU 서버와의 SSE 기반 통신

예측 요청은 /predict 엔드포인트를 통해 GPU 서버로 전달된다. EC2 의 Flask 백엔드는 ngrok 을 통해 GPU 서버에 접근하며, GPU 서버는 처리 과정을 SSE(Server-Sent Events) 로 스트리밍한다.

- 이벤트 단계: embedding → predicting → done
- 백엔드는 이를 generator/yield 기반 스트리밍으로 프론트엔드에 전달한다.
- 프론트엔드는 해당 이벤트를 받아 진행 바와 상태 메시지를 실시간 갱신한다.

- 최종적으로 GPU 서버는 예측 결과(JSON)를 반환하고, 백엔드는 이를 MySQL 에 저장하고 PDB 파일을 S3 에 업로드한다.

9.2.2.4. 입력 검증 및 예외 처리

모델 자원 낭비와 잘못된 예측을 방지하기 위해, 입력값 검증 로직을 강화하였다.
(1) 단백질 서열(FASTA):

- 20 개 표준 아미노산(ACDEFGHIKLMNPQRSTVWY)만 허용
- 비표준 문자가 포함되면 400 Bad Request 와 함께 에러 메시지를 반환
- 너무 긴 서열은 상한을 두어 거절

(2) 리간드(SMILES):

- RDKit 으로 파싱 및 검증
- 파싱 실패 시 "유효하지 않은 SMILES 입니다"와 같은 상세 메시지 반환

(3) 파일 업로드:

- 허용 확장자 및 크기 제한을 두고, 위반 시 업로드 차단
- 실패 시 명확한 에러 응답

(4) 에러 UX 표준화:

- 모든 에러를 JSON 스키마(code, message)로 반환
- 프론트엔드에서 해당 메시지를 즉시 표시하고 문제된 입력 필드를 강조하여 사용자 수정이 용이하도록 했다.

9.2.3. 모델 서버 (GPU Flask API)

모델 서버는 본 프로젝트의 핵심 연산을 담당하는 부분으로, 단백질–리간드 결합 부위(binding site)와 결합 친화도(binding affinity) 예측 모델을 GPU 환경에서 탑재하여 운영하였다.

9.2.4. DB 및 파일 저장 (MySQL + AWS S3)

본 서비스에서는 MySQL 데이터베이스와 AWS S3 를 함께 활용하여 사용자 데이터와 예측 결과를 체계적으로 관리하였다.

- MySQL: 메타데이터 관리 (사용자 정보, 예측 요청 내역, 구조 파일 경로 등)
- AWS S3: 대용량 파일 관리 (업로드된 서열·리간드 파일, 생성된 단백질 구조 PDB 파일 등)

9.2.4.1. 테이블 구조 (MySQL)

테이블명	주요 컬럼	설명
User	user_id (PK), password_hash, created_at	사용자 계정 기본 정보 저장. 이메일은 `VerifiedEmail` 테이블을 통해 관리.
VerifiedEmail	email_id (PK), user_id (FK), email, is_verified, verified_at	사용자 이메일 주소 저장. 인증 완료 여부 및 인증 완료 시각 포함.
VerificationCode	code_id (PK), email_id (FK), code, expires_at, used	이메일 인증 시 발급된 코드 관리. 만료 시간과 사용 여부 기록.
Prediction	pred_id (PK), user_id (FK), protname, sequence, smiles, affinity_score, created_at	각 사용자가 요청한 예측 기록 저장. 단백질–리간드 입력값과 결합 친화도 점수 기록.
Structure	struct_id (PK), pred_id (FK), s3_url, pdb_name, created_at	예측 과정에서 생성된 단백질 구조 파일(PDB)의 메타데이터와 AWS S3 경로 관리.

9.2.4.2. 파일 저장 (AWS S3)

- 사용자가 업로드한 입력 파일(단백질 FASTA, 리간드 SMILES)과 AlphaFold 등으로 생성된 PDB 구조 파일을 AWS S3 버킷에 저장한다.
- DB 에는 실제 파일을 저장하지 않고, S3 URL 과 파일 이름, 생성 시간을 기록해 관리한다.
- 이를 통해 서버 스토리지 부담을 줄이고, 대용량 파일을 안정적으로 보관할 수 있다.

9.2.4.3. MyPage 와 연동

- MyPage 에서 로그인한 사용자는 Prediction 및 Structure 테이블을 기반으로 자신의 예측 기록만을 조회할 수 있다.
- 각 기록에는 단백질 이름(protname), 입력 서열, 리간드, 친화도 점수가 표시되며, 구조 파일은 S3 링크를 통해 다시 불러와 NGL Viewer 에서 재시각화할 수 있다.

9.2.5. AlphaFold 통합

AlphaFold 는 아미노산 서열을 입력하면 단백질의 3 차원 구조를 예측하는 인공지능 모델로 본

서비스의 구조 시각화에 활용되었다.

애초에는 사용자가 입력한 단백질 서열을 UniProt ID 로 변환한 뒤, AlphaFold DB 에서 구조를 조회하는 방식을 구상했으나, 실제 구현 과정에서 다음과 같은 제약이 있었다.

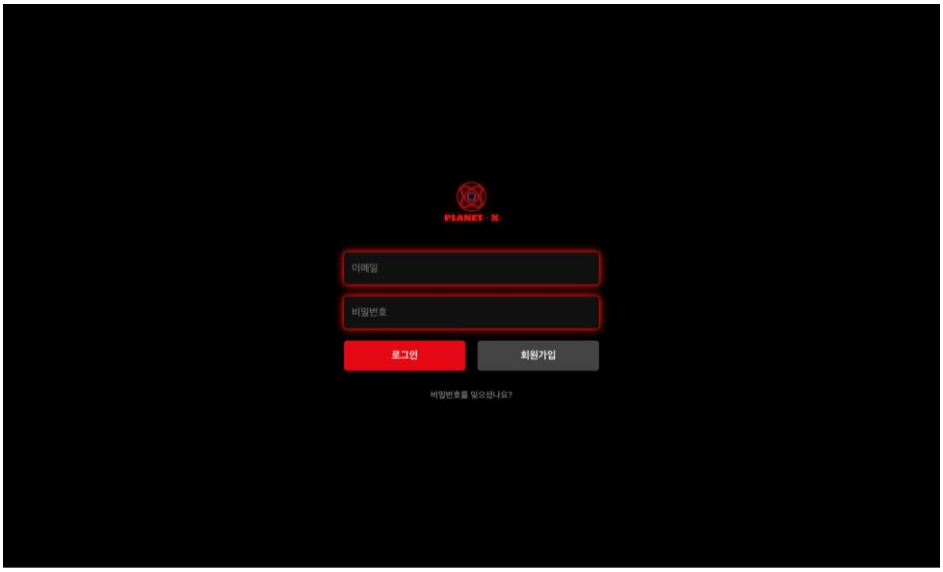
(1) UniProt 의 시퀀스→ID 매핑 API 가 중단되었고, 요청량 제한으로 안정적인 서비스가 불가능했다.
(2) 전체 AlphaFold 단백질 구조 파일을 미리 다운로드해 자체 데이터베이스를 구축하려 했으나, 200GB 이상의 대용량으로 운영 환경에서 감당하기 어려웠다.

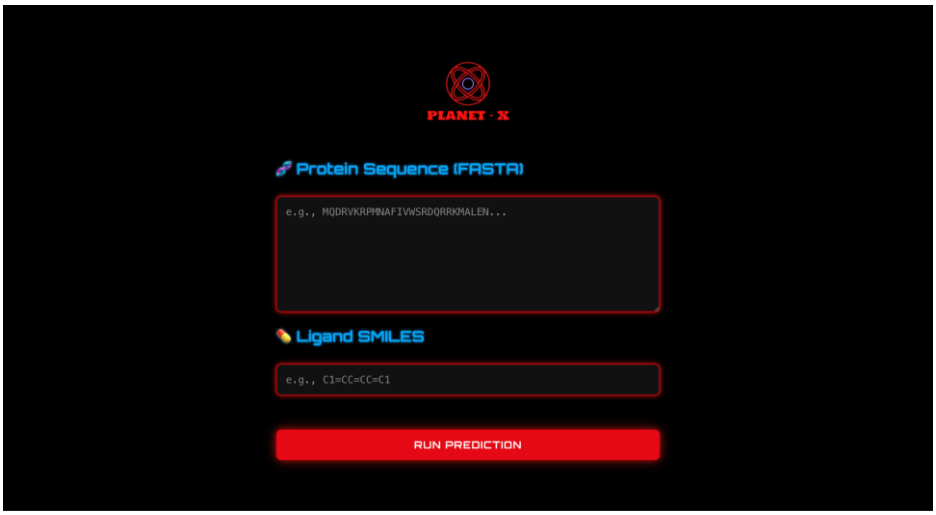
이로 인해 서비스는 UniProt ID 검색 기반 접근을 포기하고, 사용자가 입력한 서열을 AlphaFold 에 직접 투입하여 구조를 생성하는 방식으로 전환하였다. 이 과정은 GPU 자원을 많이 소모하고 시간이 오래 걸리므로, 예측 결과는 binding site/affinity JSON 을 먼저 반환한 뒤, 구조 생성이 완료되면 PDB 파일을 AWS S3 에 업로드하고 링크를 제공하는 단계적 흐름으로 처리하였다.

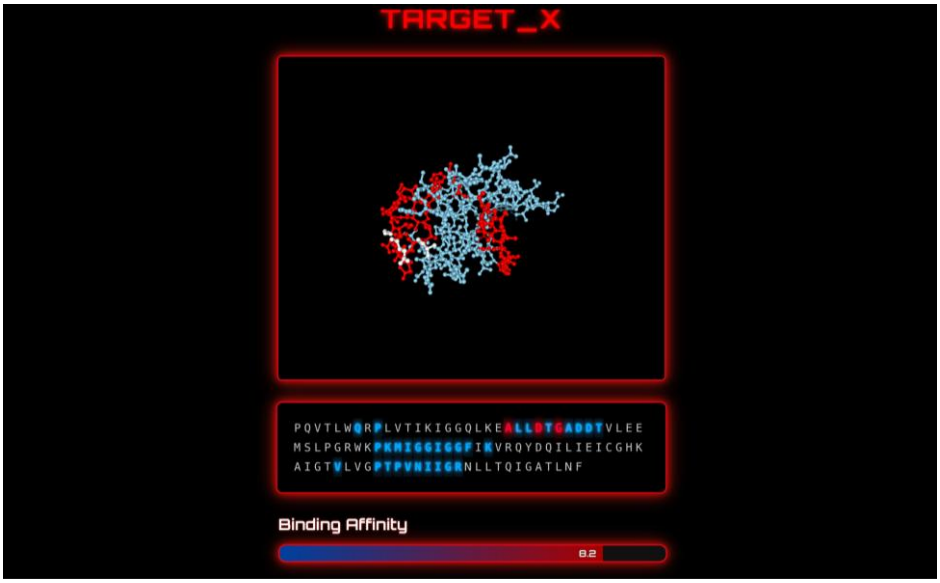
9.3. 시연 계획

본 서비스의 시연은 실제 사용자가 웹 애플리케이션을 이용하는 과정을 그대로 재현하는 방식으로 진행된다. 먼저 사용자는 회원가입 과정을 통해 이메일 인증을 완료하고 계정을 생성한 뒤 로그인한다. 이후 PredictionView 페이지에서 단백질 서열(FASTA)과 리간드(SMILES)를 입력하여 예측을 요청한다. 서비스는 빠르게 JSON 형태의 예측 결과(결합 부위 및 친화도)를 반환하며, 이후 지연된 시점에서 AlphaFold 를 통한 단백질 구조(PDB 파일)가 생성되어 AWS S3 에 업로드된다. 해당 구조는 링크를 통해 NGL Viewer 에 로딩되어 3 차원 시각화로 확인할 수 있다. 마지막으로 사용자는 MyPage 에서 자신이 수행한 예측 결과를 다시 불러오거나 재시각화할 수 있다.


PLANET - X
PLANET - X


PLANET - X
이메일
비밀번호
로그인
회원가입
비밀번호를 잊으셨나요?


PLANET - X
Protein Sequence (FASTA)
e.g., MQDRVKRPMNAFIVWSRDQRRKMALEN...
Ligand SMILES
e.g., C1=CC=CC=C1
RUN PREDICTION



10. 참고문헌

[1] A. Vaswani, L. Jones, N. Shazeer, N. Parmar, A. N. Gomez, J. Uszkoreit, Ł. Kaiser, and I. Polosukhin, "Attention Is All You Need," *arXiv preprint* arXiv:1706.03762, Aug. 2023. [Online]. Available: https://arxiv.org/abs/1706.03762

[2] A. Morehead and J. Cheng, "FlowDock: Geometric Flow Matching for Generative Protein-Ligand Docking and Affinity Prediction," *arXiv preprint* arXiv:2412.10966, Mar. 2025.

[3] Z. Jin, T. Wu, T. Chen, D. Pan, X. Wang, J. Xie, L. Quan, and Q. Lyu, "CAPLA: improved prediction of protein–ligand binding affinity by a deep learning approach based on across-attention mechanism," *Briefings in Bioinformatics*, vol. 24, no. 1, pp. 1–9, Jan. 2023. doi: 10.1093/bib/bbac534

[4] H. Öztürk, E. Ozkirimli, and A. Özgür, "DeepDTA: Deep drug-target binding affinity prediction," *arXiv preprint* arXiv:1801.10193, Jan. 2018.

[5] K. Wang, R. Zhou, Y. Li, and M. Li, "DeepDTAF: A deep learning method to predict protein–ligand binding affinity," *School of Computer Science and Engineering, Central South University*, Changsha, China, 2020.

[6] X. Zhang, H. Gao, H. Wang, Z. Chen, Z. Zhang, X. Chen, Y. Li, Y. Qi, and R. Wang, "PLANET: A Multi-objective Graph Neural Network Model for Protein-Ligand Binding Affinity Prediction," *J. Chem. Inf. Model.*, vol. 64, pp. 2205–2222, 2024.

[7] Y. Wang, Q. Jiao, J. Wang, X. Cai, W. Zhao, and X. Cui, "Prediction of protein-ligand binding affinity with deep learning," *Comput. Struct. Biotechnol. J.*, vol. 21, pp. 123–135, Nov. 2023. doi: 10.1016/j.csbj.2023.11.009

[8] S. Xu, L. Shen, M. Zhang, C. Jiang, X. Zhang, Y. Xu, J. Liu, and X. Liu, "Surface-based multimodal protein–ligand binding affinity prediction," *Bioinformatics*, vol. 40, no. 7, btae413, Jul. 2024. doi: 10.1093/bioinformatics/btae413

[9] M.-H. Wu, Z. Xie, and D. Zhi, "Protein-ligand binding affinity prediction: Is 3D binding pose needed?" *Communications Chemistry*, vol. 8, no. 2, pp. 1–10, Mar. 2025. doi: 10.1038/s42004-025-01506-1

[10] I. Lee and H. Nam, "Highlights on Target Sequences (HoTS): Predicting protein–ligand binding regions based on sequence-only information," *Briefings in Bioinformatics*, vol. 24, no. 1, bbac534, Jan. 2023. doi: 10.1093/bib/bbac534

[11] J. Chen, M. Guo, X. Wang, B. Liu, and J. Zhang, "Pseq2Sites: Sequence-based prediction of protein–ligand binding residues with deep learning," *Bioinformatics*, vol. 38, no. 15, pp. 3622–3629, Aug. 2022. doi: 10.1093/bioinformatics/btac390

[12] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, 2016, pp. 770–778. doi: 10.1109/CVPR.2016.90

[13] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 4, pp. 834–848, 2018. doi: 10.1109/TPAMI.2017.2699184

[14] M. Yang, K. Yu, C. Zhang, Z. Li, and K. Yang, "DenseASPP for semantic segmentation in street scenes," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, Salt Lake City, UT, USA, 2018, pp. 3684–3692. doi: 10.1109/CVPR.2018.00389

[15] Y. Chen, X. Li, Z. Wang, and J. Zhou, "UniAMP: enhancing AMP prediction using deep neural networks with inferred information of peptides," *BMC Bioinformatics*, vol. 26, no. 1, p. 112, 2025. doi: 10.1186/s12859-025-1234-5

[16] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," in *Proc. International Conference on Learning Representations (ICLR)*, New Orleans, LA, USA, 2019. [Online]. Available: https://arxiv.org/abs/1711.05101