핵심기술 연구 구현 결과서 Toxicity Converter

□ 연구 개요

○ 배경

- 독성 예측 인공지능(AI)은 신약 후보물질의 안전성 평가, 산업 근로자 보호, 환경 독성 모니터링 등 다양한 분야에서 수요가 급속히 증가[1].
- 이에 따라, 화합물의 독성을 정량적 및 정성적으로 예측하기 위한 머신러닝 기반 연구가 활발히 진행 중이며[2]-[5], 전통적인 독성 실험에 비해 시간과 비용 절감 및 대규모 화학 스크리닝 가능

○ 관련 연구

- ToxicBlend [6]
  - · XGBoost, 완전 연결 신경망, 그래프 컨볼루션 네트워크를 결합한 앙상블 모델을 사용하여 ToxCast 및 Tox21 데이터에서 독성 예측을 수행. 다양한 표현 방식(QSAR 설명자, PubChem 지문, SMILES)과 앙상블 접근을 통한 성능 개선 달성
- eToxPred [7]
  - · 머신러닝 기법을 사용해 작은 유기 화합물의 합성 용이성과 독성 가능성을 예측합니다. 분자 지문을 주요 입력으로 활용하며, 독성 예측 정확도 약 72% 달성

○ 기존 연구의 한계점 및 개선 사항

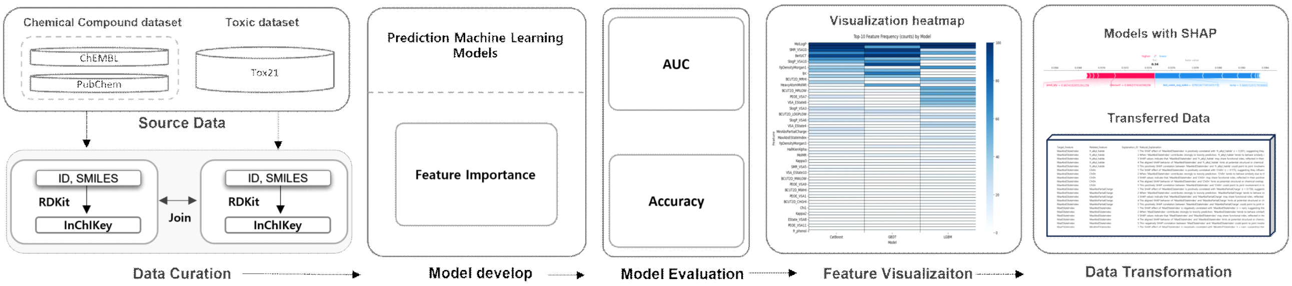
- 기존의 feature importance의 경우 비반복적 단편적인 결과를 기반의 연구가 대부분이었으며, 이들 연구의 경우 주요 결과에 대한 일반화 능력 결여

○ 연구 목표

- 본 연구의 목적은 ADMET 독성 예측에서의 주요 분자 특성(feature) 식별과 일반화 가능한 중요도 추출을 위해, 반복적이고 다모델 기반의 피처 중요도 산출 프레임워크 (Toxicity Converter)를 구축을 위함.

□ 주요 설계

❍ Toxicity Converter 설계도



❍ 수도 코드 (Pesudo code)

```
Input:  Toxicity Source Data D
Params: I (number of prediction iterations),
        K (inner steps per iteration),
        M = {LightGBM, GBDT, CatBoost},
        τ = "median" (feature selection threshold)
Output: ADMET Toxicity Converter TC, Signature SG
function Toxicity_Converter(D):
    Initialize Predictive Models M
    Initialize Result Buffers R_auc, R_features
for i = 1 to I do
        (X_train, X_test, y_train, y_test) ← StratifiedSplit(D, seed=i)
        (X_train, X_test) ← ImputeAndClean(X_train, X_test, strategy="median")
for k = 1 to K do
            for each model m in M do
                m_base ← Fit(m, X_train, y_train)
                FI ← GetImportance(m_base)
                S  ← SelectFeatures(FI, threshold=τ)
                m_final ← Fit(m, X_train[S], y_train)
                y_prob  ← PredictProba(m_final, X_test[S])
                auc     ← AUC(y_test, y_prob)
                Record(R_auc, (i, k, m, auc))
                Record(R_features, (i, k, m, S, FI))
            end for
        end for
    end for
    SG ← AggregateFeatureSignature(R_features)
    TC ← BuildConverterArtifact(M, τ, R_auc)
    return TC, SG
```

□ 연구 실험 결과

❍ 방법론

- LGBM [8], GBDT [9], 및 CatBoost [10]을 사용하여 독성 예측 모델을 구현
  · 각 모델의 특징 중요도는 내장된 특징 중요도 계산 기술을 사용하여 계산되었으며, 반복 실험의 결과는 집계 및 분석
- 평가 지표로 AUC (ROC 곡선 아래 면적) [11]를 사용

❍ 실험 결과

- Feature Importance

	model	feature	importance
0	GBDT	MolLogP	0.1216134066
0	GBDT	BertzCT	0.1081478689
0	GBDT	fr_phenol_noOrthoHl	0.03299427252
0	GBDT	Ipc	0.02665888459
0	GBDT	BCUT2D_LOGPHI	0.02450955623
0	GBDT	SlogP_VSA8	0.02104009124
0	GBDT	HeavyAtomMolWt	0.02085616541
0	GBDT	SMR_VSA10	0.02019077285
0	GBDT	qed	0.01901110155
0	GBDT	SlogP_VSA2	0.0181210404
0	GBDT	FpDensityMorgan1	0.01641331366
0	GBDT	FractionCSP3	0.01582546751
0	GBDT	SMR_VSA3	0.01528255425
0	GBDT	SlogP_VSA10	0.01525405491
0	GBDT	fr_allylic_oxid	0.01387549663
0	GBDT	MolMR	0.01223270177
0	GBDT	BCUT2D_MRHI	0.0113019463
0	GBDT	VSA_EState4	0.01127794336
0	GBDT	BCUT2D_LOGPLOW	0.01127409296
0	GBDT	SlogP_VSA5	0.01116478282
0	GBDT	SMR_VSA5	0.01114907528
0	GBDT	VSA_EState8	0.01094287784
0	GBDT	VSA_EState10	0.01057555565
0	GBDT	VSA_EState3	0.01028305372
0	GBDT	MinEStateIndex	0.01009646997
0	GBDT	fr_bicyclic	0.009896213466
0	GBDT	fr_C_S	0.009822127966
0	GBDT	PEOE_VSA11	0.00980396043

Gradient Boosting Decision Tree

	model	feature	importance
0	CatBoost	MolLogP	5.199525347
0	CatBoost	BertzCT	2.924536186
0	CatBoost	SlogP_VSA2	2.442128578
0	CatBoost	qed	1.983895131
0	CatBoost	SMR_VSA10	1.790176892
0	CatBoost	MinEStateIndex	1.690557499
0	CatBoost	PEOE_VSA7	1.679715344
0	CatBoost	SlogP_VSA8	1.603701826
0	CatBoost	EState_VSA9	1.596079316
0	CatBoost	FpDensityMorgan1	1.581848187
0	CatBoost	BCUT2D_LOGPHI	1.504405854
0	CatBoost	SMR_VSA6	1.402306255
0	CatBoost	FractionCSP3	1.393793282
0	CatBoost	fr_allylic_oxid	1.387565573
0	CatBoost	PEOE_VSA8	1.385262965
0	CatBoost	TPSA	1.270047903
0	CatBoost	EState_VSA3	1.249167047
0	CatBoost	BCUT2D_MRLOW	1.242701489
0	CatBoost	SlogP_VSA10	1.237635519
0	CatBoost	fr_phenol	1.23559278
0	CatBoost	EState_VSA4	1.23435812
0	CatBoost	VSA_EState8	1.227188765
0	CatBoost	MolMR	1.207992592
0	CatBoost	AvgIpc	1.194168809
0	CatBoost	MaxPartialCharge	1.193072978
0	CatBoost	BCUT2D_LOGPLOW	1.145658561
0	CatBoost	BCUT2D_CHGHI	1.139250685
0	CatBoost	HeavyAtomMolWt	1.122499136

CatBoost

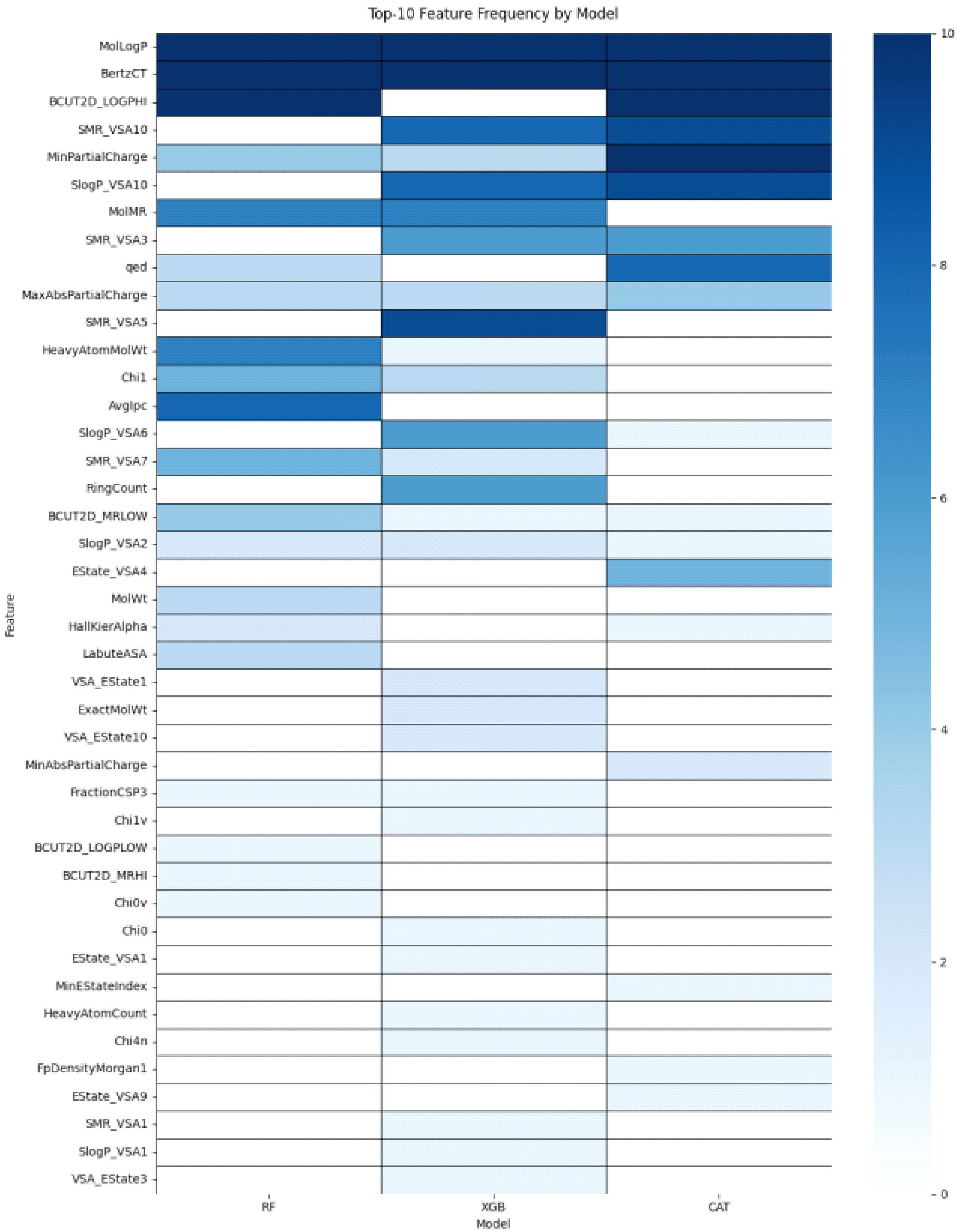
	model	feature	importance
0	LightGBM	MolLogP	230
0	LightGBM	qed	174
0	LightGBM	BCUT2D_MRLOW	165
0	LightGBM	MinAbsEStateIndex	159
0	LightGBM	VSA_EState4	157
0	LightGBM	FpDensityMorgan1	154
0	LightGBM	VSA_EState5	149
0	LightGBM	EState_VSA9	143
0	LightGBM	VSA_EState8	143
0	LightGBM	PEOE_VSA7	137
0	LightGBM	EState_VSA4	137
0	LightGBM	FpDensityMorgan2	137
0	LightGBM	SlogP_VSA2	133
0	LightGBM	BCUT2D_LOGPHI	133
0	LightGBM	SPS	131
0	LightGBM	BCUT2D_LOGPLOW	129
0	LightGBM	PEOE_VSA8	129
0	LightGBM	BalabanJ	128
0	LightGBM	SMR_VSA10	126
0	LightGBM	BCUT2D_MRHI	124
0	LightGBM	SlogP_VSA3	123
0	LightGBM	AvgIpc	120
0	LightGBM	MinEStateIndex	119
0	LightGBM	HallKierAlpha	118
0	LightGBM	BCUT2D_MWHI	115
0	LightGBM	EState_VSA3	115
0	LightGBM	MinPartialCharge	113
0	LightGBM	MinAbsPartialCharge	109

LightGBM

- **AUC values from 100 runs for each model**

run	model	auc	n_features
0	LightGBM	0.800924	105
0	GBDT	0.80007	105
0	CatBoost	0.803204	105
1	LightGBM	0.793631	105
1	GBDT	0.79285	105
1	CatBoost	0.800554	105
2	LightGBM	0.789584	105
2	GBDT	0.784122	105
2	CatBoost	0.790216	105

- Visualization

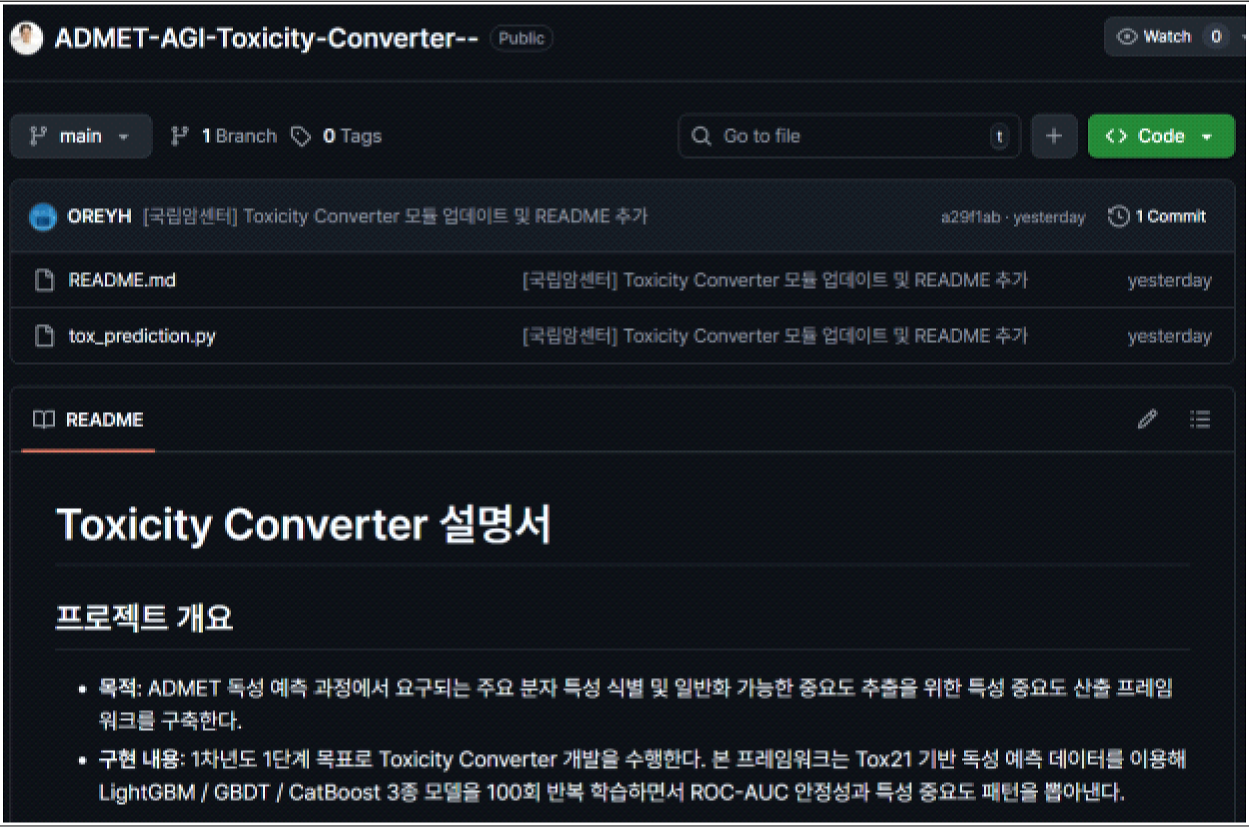


□ **공개SW 성과**

○ 소스코드

```python
import os, glob, numpy as np, pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.metrics import roc_auc_score
from sklearn.impute import SimpleImputer
from sklearn.feature_selection import SelectFromModel
from sklearn.ensemble import GradientBoostingClassifier
import lightgbm as lgb
from catboost import CatBoostClassifier, Pool

# (옵션) 시각화
import matplotlib.pyplot as plt
import seaborn as sns

# ----------------------------
# 0) 경로 & 출력 폴더
# ----------------------------
DATA_PATH = r"C:\Users\nicep\project\Canonicalized_Tox21_with_rdkit_descriptors.csv"
OUT_MODELS, OUT_RESULTS = "models", "results"
os.makedirs(OUT_MODELS, exist_ok=True); os.makedirs(OUT_RESULTS, exist_ok=True)

# ----------------------------
# 1) 데이터 로드 & 피처/타깃 분리
# ----------------------------
df = pd.read_csv(DATA_PATH)

drop_cols = ["ASSAY_NAME","LABEL","SMILES","Can_SMILES"]
X = df.drop(columns=[c for c in drop_cols if c in df.columns], errors="ignore")
X = X.apply(pd.to_numeric, errors="coerce").replace([np.inf, -np.inf], np.nan)
X = X.dropna(axis=1, how="all")
y = pd.to_numeric(df["LABEL"], errors="coerce")

# LABEL 결측 제거
if y.isna().any():
    valid_idx = ~y.isna()
    X = X.loc[valid_idx]
    y = y.loc[valid_idx]
```

○ Git-Hub 공개: https://github.com/KwangSun-Ryu/ADMET-AGI-Toxicity-Converter--



□ 참고문헌

1. Masarone, S., et al.: Advancing predictive toxicology: overcoming hurdles and shaping the future. Digital Discovery 4(2), 303–315 (2025)
2. Bai, Changsen, et al. "Machine Learning Enabled Drug Induced Toxicity Prediction." Advanced Science 12.16 (2025): 2413405.
3. Jaganathan, Keerthana, Hilal Tayara, and Kil To Chong. "An explainable supervised machine learning model for predicting respiratory toxicity of chemicals using optimal molecular descriptors." Pharmaceutics 14.4 (2022): 832.
4. Guo, Wenjing, et al. "Review of machine learning and deep learning models for toxicity prediction." Experimental Biology and Medicine 248.21 (2023): 1952-1973.
5. Zhang, Ruiqiu, et al. "Artificial Intelligence-Driven Drug Toxicity Prediction: Advances, Challenges, and Future Directions." Toxics 13.7 (2025): 525.
6. Zaslavskiy, Mikhail, et al. "ToxicBlend: virtual screening of toxic compounds with ensemble predictors." Computational Toxicology 10 (2019): 81-88.
7. Pu, L., Naderi, M., Liu, T. et al. eToxPred: a machine learning-based approach to estimate the toxicity of drug candidates. BMC Pharmacol Toxicol 20, 2 (2019). https://doi.org/10.1186/s40360-018-0282-61.Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma,
8. W., Ye, Q., Liu, T.Y.: LightGBM: a highly efficient gradient boosting decision tree. Adv. Neural Inf. Process. Syst. 30, 3149–3157 (2017)
9. Friedman, J.H.: Greedy function approximation: a gradient boosting machine. Ann. Stat. 29(5), 1189–1232 (2001)
10. Prokhorenkova, L., Gusev, G., Vorobev, A., Dorogush, A.V., Gulin, A.: CatBoost: unbiased boosting with categorical features. Adv. Neural Inf. Process. Syst. 31, 6638–6648 (2018)
11. Hanley, J.A., McNeil, B.J.: The meaning and use of the area under a receiver operating characteristic (ROC) curve. Radiology143(1), 29–36 (1982).