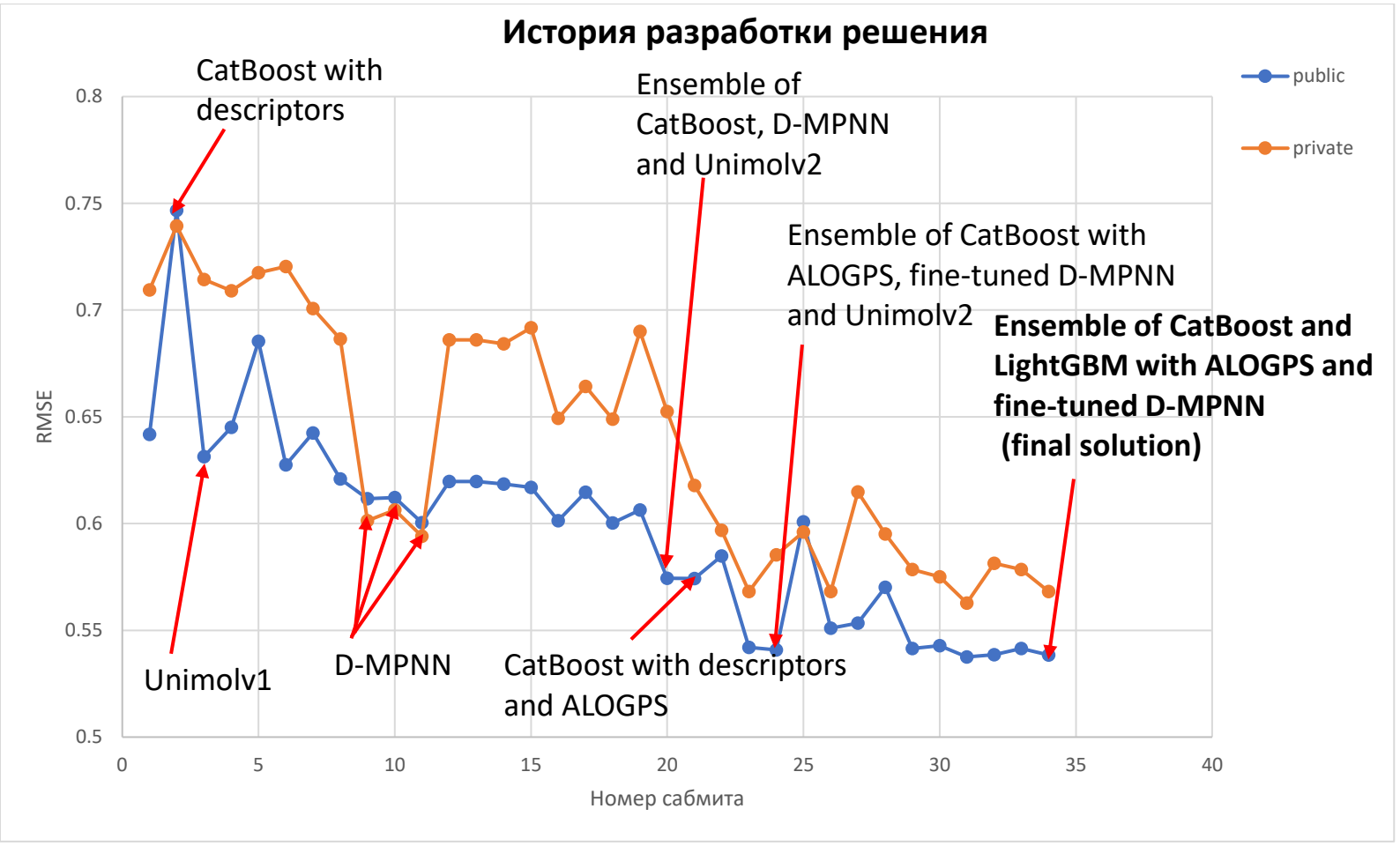
Элемент 119 от СИБУР

Решение команды zxc_ghoul3228



Репозиторий с финальным решением: https://github.com/zxcghoul3228/Sibur_element_119_2025

Подготовка данных

В самом начале был проведен анализ исходных данных, как smiles строк, так и целевых значений (logP). Все валидные smiles были стандартизированы с помощью библиотеки chython и приведены к каноническому виду с помощью библиотеки rdkit. Были обнаружены следующие аномалии: 1) наличие дубликатов с разными значениями logP; 2) невалидные smiles строки; 3) аномально высокие и низкие значения logP.

Обработка дубликатов

В обучающей выборке было около 3000 дубликатов, а именно smiles строк с несколькими различными значениями logP. Сначала все дубликаты были исключены из обучающего набора. На оставшихся данных была обучена модель градиентного бустинга на решающих деревьях из библиотеки CatBoost (см. раздел **финальное решение)**, после чего среди дубликатов были

оставлены те значения logP, которые были ближе всего к предсказанию модели. Полученный подход позволил улучшить метрики моделей (около 0.01-0.02 единиц RMSE) и показал лучшие результаты, чем исключение всех дубликатов или замена на максимальное/минимальное/среднее значения. Из 3000 дубликатов осталось около 1000 молекул, которые пополнили обучающую выборку.

Невалидные smiles

Исходный обучающий набор содержал около 450 невалидных smiles строк (пропущенные скобки, лишние символы, незамкнутые циклы, гипервалентность и т.д.). Были приняты попытки восстановления таких молекул как с помощью базовых алгоритмов (удаление некорректных символов, добавление парных скобок), так и с помощью языковых моделей. Однако неуниверсальность данных подходов, неоднозначность восстановления невалидных smiles, а также связанные с этим высокие временные затраты вынудили отказаться от данных подходов.

В силу вышеперечисленного, а также ввиду небольшого количества невалидных smiles в конечном решение некорректные молекулы удалялись и обучающей выборки.

Аномальные значения

При анализе значений logP были обнаружены как аномально высокие, так и аномально низкие значения logP. Такие значения было принято удалить из обучающей выборки. Критерий аномальности определялся путем визуального анализа данных (boxplot и др.), а также статистическими тестами. Границы обрезки были выбраны $logP<-2$ и $logP>10$.

Исключение аномальных значений позволило значительно улучшить качество моделей градиентного бустинга (до 0.1 единиц RMSE), однако почти не повлияло на нейросетевые архитектуры ввиду их устойчивости к выбросам.

Алгоритмы обучения

В процессе разработки решения было опробовано большое количество алгоритмов машинного обучения, включая классические алгоритмы, полносвязные и графовые нейронные сети, а также большие предобученные языковые модели. Кроме того, применялись различные методы ансамблирования моделей и обучения с применением разных модальностей.

Классические алгоритмы

Среди классических алгоритмов рассматривались случайный лес (Random Forest, RF), а также различные реализации градиентных бустингов на решающих деревьях (библиотеки XGBoost, LightGBM, CatBoost), поскольку

являются хорошо зарекомендовавшими себя архитектурами, в частности, для работы в табличными данными.

Получение векторных представлений (признаков) молекул из датасета проводилось путем вычисления различных 2D- и 3D-дескрипторов, фингерпринтов. Для вычисления дескрипторов использовались библиотеки rdkit, mordred [1] и mold2 (Mold2-pywrapper) [2], фингерпринты (Morgan, MACCS и RDKit) вычислялись с помощью библиотеки rdkit. Однако в конечном решении дескрипторы Mold2 и фингерпринты RDKit представлены не были. Значительный прирост к метрикам дало добавление дескрипторов logP и logS, рассчитанных с помощью программы ALOGPS [3]. Данная программа не имеет python-обертки, поэтому дескрипторы вычислялись непосредственно в интерфейсе программы, а полученные файлы доступны в Github репозитории.

Помимо классических дескрипторов и фингерпринтов, были опробованы эмбеддинги, полученные с помощью предобученных языковых моделей, таких как ChemBerta [4] и MolFormer [5], однако использование таких векторов как отдельно, так и совместно с физико-химическими дескрипторами не улучшило качество моделей.

Помимо обособленных моделей, применялись также различные техники ансамблирования моделей, включая как использование одинаковых архитектур, обученных на разных наборах признаков, так и разных алгоритмов. Данная техника позволила улучшить метрику (0.1-0.3 единиц RMSE) и является частью финального решения.

Подбор гиперпараметров осуществлялся с помощью библиотеки Optuna, а также из опыта автора. В основном перебирались такие параметры, как количество деревьев в бустинге, глубина деревьев и скорость обучения. Валидация моделей осуществлялась методом 5-fold кросс-валидации. Подбор оптимальных гиперпараметров позволил несколько улучшить метрику (0.1-0.2 единиц RMSE).

Лучшие значения метрики RMSE показали алгоритмы CatBoost и LightGBM с использованием дескрипторов и фингерпринтов из библиотек rdkit и Mordred, а также дескрипторов ALOGPS. С помощью классических алгоритмов удалось достичь метрики 0.57 на публичном лидерборде (0.62 на приватном).

Нейросетевые модели

Среди архитектур глубокого обучения сначала была опробована полносвязная нейросеть на нормированных дескрипторах, описанных в предыдущем пункте. Данная архитектура не превзошла бустинги, поэтому следующими были опробованы графовые архитектуры. Несколько графовых нейронных сетей

были опробованы [6,7], однако ключевой стала архитектура Directed Message-Passing Neural Network (D-MPNN) из библиотеки chemprop [6]. Данная модель использует технику «передачи сообщения», обновляя векторы-признаки вершин (атомов) или ребер (хим. связей) графа с учетом информации о соседях. За счет представления молекулы в виде двумерного графа данная архитектура показала большую точность предсказания logP. Для данной модели были подобраны оптимальные значения количества слоев, функции «передачи сообщений», аггрегирующие функции, а также параметры обучения (скорость обучения, число эпох, размер батча). Валидация проводилась методом 5-fold кросс-валидации.

Для улучшения качества предсказаний было принято решение использовать дополнительные данные для предварительного обучения модели. Датасет из 17000 молекул с известными значениями logP, взятый из базы данных OCHEM (название датасета «ALOGPS 3.01 (training)») [8] использовался для предобучения модели D-MPNN. Предварительно были исключены все молекулы, содержащиеся в тестовом наборе, чтобы избежать «утечки данных» и получить честные значения на лидерборде. На следующем шаге D-MPNN дообучалась на обучающей выборке, данной на соревновании. Предварительное обучение (совместно с ансамблированием нескольких моделей D-MPNN) позволило значительно улучшить метрики (0.56-0.57 на публичном лидерборде).

Кроме того, были попытки провести дообучение языковых моделей ChemBerta и MolFormer, однако высокую точность предсказаний добиться не удалось. Одним из главных разочарований была попытка дообучения архитектуры Unimol [9], которая генерирует 3D-представления молекул по их smiles и занимает первые позиции на многих бенчмарках. Однако эта модель содержит огромное число параметров (от 84M до 1.1B), что позволило провести дообучение лишь с небольшим размером батча (4) (и самой меленькой модели) и не привело к ожидаемым результатам (автору известно о методе накопления градиентов, однако на модификацию исходного кода было недостаточно времени).

Один из подходов заключался в добавлении дополнительной модальности к графовой архитектуре D-MPNN, путем конкатенации эмбеддингов исходной нейросети с эмбеддингами предобученной языковой модели и подачи такого вектора в полносвязные слои. Однако (к большому удивлению) данный подход не привел к улучшению метрик.

Финальное решение

Подготовка данных в финальном решении проводилась путем удаления невалидных smiles, удаления аномальных значений, удаления дубликатов

(оставлялись значения logP наиболее близкие к предсказанию модели CatBoost на дескрипторах rdkit, mordred и фингерпринтах morgan и maccs).

Финальный пайплайн представлял собой ансамбль из модели CatBoost и LightGBM на физико-химических дескрипторах rdkit, Mordred и фингерпринтах Morgan и MACCS, а также дескрипторах ALOGPS (веса в ансамбле 0.25 у каждой модели); пяти моделей D-MPNN, предобученных на дополнительных данных (вес 0.1 у каждой). Гиперпараметры градиентных бустингов были подобраны с помощью библиотеки Optuna, нейросетей – ручным подбором. Оценка качества моделей проводилась методом 5-fold кросс-валидации. Метрика финального пайплайна **0.538/0.569** public/private.

Автор выражает огромную благодарность организаторам соревнования за качественное проведение и интересную задачу!

Ссылки

[1] Welcome to mordred’s documentation! — mordred 1.2.1a1 documentation

[2] Mold2-pywrapper · PyPI

[3] On-line Lipophilicity/Aqueous Solubility Calculation Software

[4] seyonec/ChemBERTa-zinc-base-v1 · Hugging Face

[5] ibm-research/MoLFormer-XL-both-10pct · Hugging Face

[6] chemprop/chemprop: Message Passing Neural Networks for Molecule Property Prediction

[7] benatorc/OTGNN: OTGNN code

[8] Online Chemical Modeling Environment

[9] deepmodeling/Uni-Mol: Official Repository for the Uni-Mol Series Methods