КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ТАРАСА ШЕВЧЕНКА
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ВИСОКИХ ТЕХНОЛОГІЙ

Завідувач кафедри молекулярної біотехнології та біоінформатики
доцент Олексій Юрійович Нипорко
Протокол №____ засідання кафедри
від “____” _____20___ р.

АНАЛІЗ ЗАКОНОМІРНОСТЕЙ ФОЛДИНГУ БІЛКІВ МЕТОДАМИ ПОЯСНЕННЯ ШТУЧНОГО ІНТЕЛЕКТУ

Випускна кваліфікаційна робота магістра
студента спеціальності 091 Біологія
ОП “Біоінформатика та структурна біологія”
Попова Артема Валерійовича

Науковий керівник від кафедри
асистент кафедри молекулярної біотехнології та біоінформатики
к.ф.-м.н. **Войтешенко Іван Сергійович**

Робота виконана у відділі автоматизації програмування
Інституту Кібернетики ім. В. М. Глушкова НАН України
під керівництвом д. ф.-м. н. **В. Г. Тульчинського**

Оцінка захисту роботи

_____

Київ – 2024 р.

АНОТАЦІЯ

Попов А.В. Аналіз закономірностей фолдингу білків методами пояснення штучного інтелекту. – Випускна кваліфікаційна робота магістра за спеціальністю 091 Біологія ОП “Біоінформатика та структурна біологія”

Прогнозування структур білків за послідовностями амінокислот є складною задачею для обчислювальної біології з широкими наслідками для різних галузей, включаючи проектування ліків та дослідження хвороб. Нещодавні досягнення в глибокому навчанні дозволяють отримати високоточні моделі фолдингу, які, однак, не піддаються інтерпретації через свою black-box природу.

В цій роботі вивели частину закономірностей фолдингу білків із індивідуальних властивостей амінокислот на основі інсайдів з модуля нейромережі для *de-novo* згортки AminoBERT.

У результаті аналізу було встановлено частину основних залежностей, необхідних для передбачення вторинної та третинної структури із первинної. Для отримання та перевірки цих закономірностей було отримано локальні (12 вторинної структури та 14 третинної структури) та глобальні (2 вторинної та 1 третинної) моделі фолдингу

При аналізі векторів-вставок було проведено лінійну регресію для визначення основних фізико-хімічних закономірностей, що розміщують амінокислоти у векторному просторі вставок. Встановлено, що маса, гідрофобність, дипольний момент та частота пояснюють 48-49% варіативності дистанцій між векторами. Було також визначено, що ці властивості далі залишаються важливими для розрізнення вторинних та третинних структур .

Для визначення закономірностей вторинної структури було розраховано 12 локальних та 2 глобальні моделі, за допомогою яких виведено та підтверджено найбільш розповсюджені особливості, що

визначають вторинну структуру. Серед них була встановлена характерна залежність від позиції в послідовності, що нагадує гаусову криву, інверсія залежності від гідрофобності для бета ланцюга при віддаленні від позиції передбачення, менша залежність альфа спіралі від гідрофобності, ніж для бета ланцюга, і навпаки для дипольного моменту. Нарешті, певні структурні дескриптори також відрізняються між бета ланцюгом та альфа спіраллю - для альфа спіралі характерна залежність від поверхні, тоді як для бета ланцюга - від об’єму. Серед невпорядкованих найкраще класифікуються випадковий клубок, невпорядкований регіон та повороти, де різниця між вигинами та поворотами пов’язана різними залежностями від pKa біля точки передбачення та дипольного моменту.

Для третинної структури побудували простіші моделі для фолдингу на основі передбачення матриць дистанцій, розрахували пояснення та встановили найважливіші характеристики. Ними виявились позиційна відстань, гідрофобність, дипольний момент та частота амінокислот у послідовностях. Було підтверджено зменшення відстані між термінальними регіонами білка, що є можливим проспектом для покращення майбутньої простої глобальної моделі фолдингу білка.

Ключові слова: фолдинг білка; пояснення штучного інтелекту; передбачення вторинної структури; передбачення третинної структури

АНОТАЦІЯ

Popov A.V. Analysis of protein folding patterns using explainable artificial intelligence methods - Master's thesis in the specialty 091 Biology, educational program “Bioinformatics and Structural Biology”

Predicting protein structures from amino acid sequences is a challenging task for computational biology with broad implications for various fields, including drug design and disease research. Recent advances in deep learning have produced highly accurate folding models, which, however, are not interpretable due to their black-box nature.

In this work, we derived some of the patterns of protein folding from individual amino acid properties based on insights from the neural network module for de novo folding AminoBERT. The analysis revealed some of the main dependencies necessary for predicting secondary and tertiary structures from the primary structure. Local (12 for secondary structure prediction and 14 for tertiary structure prediction) and global (2 secondary and 1 tertiary) folding models were generated to obtain and verify these patterns.

When analyzing the embeddings, a linear regression was performed to determine the main physicochemical characteristics that cluster amino acids in the embeddings’ space. Mass, hydrophobicity, dipole moment, and amino acid frequency were found to explain 48-49% of the variation in vector distance. It was also determined that these properties further remain important for modeling secondary and tertiary structures.

To determine the patterns necessary for secondary structure prediction, 12 local and 2 global surrogate models were calculated, which were used to derive and confirm the most common features that determine the secondary structure. Among them, we found a characteristic dependence on the position in the sequence that resembles a Gaussian curve, an inversion of the dependence of hydrophobicity on the distance from the prediction position for the beta chain, a

lower dependence of the alpha helix on hydrophobicity than for the beta chain, and vice versa for the dipole moment. Finally, certain structural descriptors also differ between beta chain and alpha helix: alpha helix is characterized by a surface dependence, while beta chain is characterized by a volume dependence. Among the disordered ones, the classes with the most accurate predictions are random tangle, disordered region, and turns, where the difference between bends and turns was found to be due to different dependencies on the pKa near the prediction point and the dipole moment.

For the tertiary structure, we constructed simpler models for folding based on the prediction of distograms, calculated explanations, and identified the most important characteristics. These were the positional distance, hydrophobicity, dipole moment, and frequency of amino acids in the sequences. A decrease in the distance between protein terminal regions compared to the average was confirmed, which is a possible avenue for improving a future simple global protein folding model.

Keywords: protein folding; explainable artificial intelligence; secondary structure prediction; tertiary structure prediction

ЗМІСТ

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

AUC-MoRF - Area Under the receiving operating characteristic Curve, Most Relevant First (Площа під кривою робочої характеристики приймача, найбільш відповідні перші)

BERT - Bidirectional Encoder Representations from Transformers (Двонаправлені представлення енкодерів на основі трансформерів)

BFGS - Broyden–Fletcher–Goldfarb–Shanno algorithm (Алгоритм Бройдена-Флетчера-Гольдфарба-Шанно)

BLAST - Basic Local Alignment Search Tool (Базовий інструмент локального пошуку вирівняння)

BLOSUM - Blocks Substitution Matrix (Матриця заміни блоків)

CART - Classification and Regression Trees (Класифікаційні та регресійні дерева)

CASP - Critical Assessment of Structure Prediction (Критична оцінка прогнозування структури)

DeepLIFT - Deep Learning Important FeaTures (Глибоке навчання важливих характеристик)

EC - Enzyme Commission (Класифікація комісії з ферментів)

EIIP - Electron-Ion Interaction Potential (Потенціал взаємодії електрона з йоном)

GELU - Gaussian Error Linear Unit (Лінійний вузол гаусової похибки)

Grad-CAM - Gradient-weighted Class Activation Mapping (Карта активації класу, зважена градієнтом)

KL-дивергенція - Дивергенція Кульбака-Лейблера

LDA - Linear Discriminant Analysis (Лінійний дискримінантний аналіз)

LIME - Local Interpretable Model-agnostic Explanations (Локальні пояснення, незалежні від моделі)

LRP - Layer-wise Relevance Propagation (Поширення значущості по шарам нейромережі)

LSTM - Long Short-Term Memory (Довга короткочасна пам'ять)

MAE - Mean Absolute Error (Середня абсолютна помилка)

MLM - Masked Language Modeling (Маскувальне моделювання мови)

MMM - Multiple Mapping Method (Метод множинного відображення)

MSA - Multiple Sequence Alignment (Множинне вирівнювання послідовностей)

NLP - Natural Language Processing (Обробка природної мови)

PCA - Principal Component Analysis (Аналіз головних компонент)

PDB - Protein Data Bank (Банк даних білків)

pKa - Логарифм константи дисоціації кислоти

RBM - Restricted Boltzmann Machine (Обмежена машина Больцмана)

ReLU - Rectified Linear Unit (Випрямлений лінійний вузол)

SCOPe - Structural Classification of Proteins – extended (Розширена структурна класифікація білків)

SE(3) - Special Euclidean group in three dimensions (Спеціальна євклідова група у трьох вимірах)

SHAP - SHapley Additive exPlanations (Адитивні пояснення Шеплі)

SVD - Singular Value Decomposition (Сингулярний розклад матриці)

t-SNE - t-distributed Stochastic Neighbor Embedding (t-розподілене вкладення стохастичної близькості)

TCAV - Testing with Concept Activation Vectors (Тестування з векторами активації концептів)

XAI - Explainable Artificial Intelligence (Пояснювальний штучний інтелект)

XGBoost - eXtreme Gradient Boosting (Екстремальне градієнтне підсилення)

ВСТУП

Прогнозування структур білків за послідовностями амінокислот є складною задачею для обчислювальної біології з широкими наслідками для різних галузей, включаючи проектування ліків та дослідження хвороб. Для вирішення цієї задачі фолдингу застосовуються різноманітні методи, від моделювання по гомології, до аб інітіо статистичних методів та на основі навчання штучних нейромереж. Нещодавні досягнення в глибокому навчанні дозволяють отримати високоточні моделі фолдингу, такі як AlphaFold2, RoseTTAFold2, OmegaFold, ESMFold, RGN2 та інші, які, однак, не піддаються інтерпретації через свою black-box природу - коефіцієнти цих моделей не мають прямої кореляції із відомими властивостями білків, оскільки нейромережі самостійно шукають шлях залежності від входів (x) до виходів (y) через градієнтний спуск.

Ця проблема визначає теоретичну необхідність пояснення моделей фолдингу білка, оскільки в такому випадку хоча стає можливим отримати високоякісні передбачення структури білка, закономірності в основі фолдингу залишаються загадкою, зашифрованою тепер у вагах нейромереж для згортки білків. У біології та медицині здатність інтерпретації нейромереж набуває також особливого практичного значення, оскільки від правильного шляху прийняття рішень та помилки нейромережі може залежати життя пацієнтів - наприклад, у випадку розпізнання нейромережею пухлин чи розробці ліків. Тому для розкриття чорного ящику моделей глибокого навчання різними авторами були запропоновані різноманітні методи [1, 2, 3], які дозволяють надавати пояснення як для поодиноких прикладів, так і для всієї моделі. Ці методи були описані в рамках галузі пояснювального штучного інтелекту (XAI), які забезпечують метрики та сурогатні моделі з легкою інтерпретацією для локального та глобального пояснення індивідуальних рішень більш

складних моделей штучного інтелекту. Таким чином, вивчення процесу прийняття рішень моделями фолдингу білка дозволяє як просунутись більше до високоякісної та зрозумілої теорії фолдингу білка, так і надає можливість додатково перевіряти передбачення, отримані з цих нейромереж.

Відповідно, метою цього дослідження є виведення закономірностей фолдингу білків із індивідуальних властивостей амінокислот на основі інсайдів з нейромереж для *de-novo* згортки.

Для вирішення цієї мети були поставлені наступні задачі:

1. Підібрати відповідну нейромережу та методи пояснення штучного інтелекту для її дослідження;
2. Зібрати глобальні та локальні пояснення для вторинної та третинної структури білка;
3. Побудувати сурогатну модель, що збирає в єдине знайдені закономірності, та проаналізувати її точність.

Наукова новизна. Вперше було проведено аналіз методами пояснювального штучного інтелекту нейромережі RGN2. Було перевірено можливість застосування методів XAI на основі аналізу збурень для нейромереж фолдингу білків та проведено їхню адаптацію для пояснення в сфері біомолекулярного моделювання, і в результаті цього аналізу було отримано 26 (12 для вторинної структури, з яких 6 на основі лінійної регресії, 6 на основі XGBoost, 14 для третинної структури, з яких 7 на основі Ridge регресії, 7 на основі XGBoost) локальних та 3 (2 для вторинної структури, з яких 1 на основі XGBoost, а інша представляє формульну параметричну модель, деталі далі, 1 для третинної структури на основі XGBoost) глобальних сурогатних моделі. Було встановлено основні залежності, необхідні для передбачення вторинної та третинної структури із первинної та побудовано формулу для передбачення

вторинної структури із первинної, яка грунтується на фізико-хімічних характеристиках амінокислот, а не на статистичних параметрах.

Практичне значення роботи. Отримані глобальні моделі можна застосовувати для передбачення вторинної структури із точностями: 0.66 (на основі модуля AminoBERT нейромережі RGN2), 0.51 (XGBoost глобальний сурогат) та 0.42 (формула для розрахунку вторинної структури), організованих тут в порядку підвищення інтерпретованості, та грубої оцінки третинної структури, з середньою абсолютною похибкою 8-11 А (на основі модуля AminoBERT та для XGBoost глобального сурогату). Також, було показано ефективність застосування XAI методів на основі аналізу збурень, адаптованих для пояснення моделей у біомолекулярних дослідженнях, що мотивує до їхнього подальшого застосування.

РОЗДІЛ 1
ОГЛЯД ЛІТЕРАТУРИ

1.1. Підходи до вирішення проблеми фолдингу білків

Вперше про складність проблеми згортки білка було згадано в статті про першу в світі розшифровку структури білка з рентгено-структурного аналізу 1958 року, написану нобелівським лауреатом Джоном Кендрю, де той вказує на “відсутність хоч якихось очікуваних закономірностей” у структурі міоглобіна роздільною здатністю 6 А [4, 5]. Хоча він не втрачав оптимізму, що проблема буде вирішена, з того часу його слова про складність цієї проблеми підтвердились роботами інших вчених, наприклад, Цирусом Левінталем, який, в 1969 році, сформулював свій аргумент про визначеність шляху фолдингу, який вказував на складність обрання оптимальної конформації серед 10^{300} можливих конформацій шляхом випадкового пошуку - має бути певна процедура, що відсіює більшість з цих структур [4-6]. Це призвело до формулювання опису фолдингу як воронки, де більшість структур є некомпактними та неорганізованими, а тому (з точки зору статистичної термодинаміки полімерів) високоенергетичними, тоді як дуже маленька доля компактних структур на дні воронки є низькоенергетичними, близькими до нативної структури. Однак цих закономірностей було недостатньо для спрощення розробки алгоритмів для пошуку нативної структури. Тому для пришвидшення винайдення оптимальних алгоритмів в 1994 році був розроблений конкурс CASP [4-7], де вчені з усього світу можуть подати свої передбачення структури невідомого білка за допомогою своїх методів, що слугує як бенчмарк для обрання найкращих алгоритмів фолдингу.

З того часу було розроблено багато підходів для вирішення цієї проблеми, серед яких моделювання за шаблоном (або моделювання по гомології [7]) та моделювання аб інітіо [4, 6]. Для моделювання за

шаблоном спочатку відбувається пошук гомологічних послідовностей, для яких вже є наявні структури, після якого відбувається оптимізація бічних ланцюгів амінокислот, що відрізняються, та частин каркасу в місцях інсерцій чи делецій. У випадку ситуацій з наявністю лише дуже дальніх родичів, буває необхідно застосувати більш довгий та скрупульозний оптимізаційний процес [6, 8]. Цей алгоритм побудований на основі припущення, що схожість послідовностей вказує на схожість структури. У таких ситуаціях моделювання структури білка відбувається швидко та доволі точно. Основні кроки при виконанні моделювання на основі шаблону такі:

1. Пошук білків зі схожими послідовностями/структурами. Цей етап передбачає пошук послідовностей зі вже встановленими структурами, які так чи інакше схожі на дану послідовність. Тут можна застосовувати класичні підходи, як BLAST [9], що шукають схожі білкові послідовності на основі лише однієї даної, або спробувати щось, що дасть більше інформації про структуру білка, як-от використання еволюційної інформації для побудови моделей [6, 8], створення консенсусу та пошук за ним, аналіз проміжних послідовностей між знайденими та даною через філогенетичне дерево та інші підходи [6, 8]. Таким чином можна отримати доволі якісні моделі навіть у випадках, коли ідентичність послідовностей дуже низька (менше 25%). Іншим підходом для ідентифікації схожих білків є тредінг [7, 8], де білкова послідовність порівнюється з бібліотекою 3D структур, і на основі побудови спеціальних профілів відбувається пошук найкращих кандидатів. У даному випадку замість порівняння послідовність-послідовність, відбувається порівняння послідовність-структура, що дозволяє моделювати білки з низькою гомологією, однак з однаковим фолдом [8];

2. Відбір відповідних кандидатів. Цей етап відбувається здебільшого за рахунок аналізу послідовностей отриманих кандидатів та даної послідовності, де обирають найкращі шаблони на основі найбільшої подібності між послідовностями. Однак окрім цього, потрібно враховувати особливості взаємодій білка із середовищем - лігандами, водою, іншими білками, та якість структури (наскільки вона відповідає фізичним та статистичним принципам, що були вже встановлені, яка роздільна здатність моделі білка). Якість може бути покращена застосуванням декількох моделей-шаблонів підряд [8, 10], тому не обов'язково відбирати лише один найкращий варіант;
3. Вирівнювання обраних кандидатів по оригінальній послідовності. Вирівнювання структур є доволі важливим етапом побудови моделі на основі шаблону, особливо в ситуаціях, де ідентичність послідовностей менше 25%. Першим етапом здебільшого є процес накладання багатьох моделей шаблонів, після чого відбувається вирівнювання накладки з оригінальною послідовністю. Важливим є відсутність пробілів у елементах вторинних структур чи в мотивах, що є зануреними в кор білка. Для цього застосовується MMM метод [11], який використовує 4 параметри для оцінки, які фрагменти брати з яких частин вирівнювання - матриця середовища FUGUE, матриця замін амінокислот BLOSUM, матриця структурних замін H3P2, яка порівнює передбачену вторинну структуру та вторинну структуру обраного фрагменту, та статистичний попарний потенціал контактів для амінокислот;
4. Отримання моделі для оригінальної послідовності. Ця частина складається з шаблон-залежної та шаблон-незалежної частини. При шаблон-залежній частині відбувається побудова моделі на основі вирівняних співпадаючих частин. Програми як COMPOSER [12] застосовують доволі простий процес вирізання фрагментів та їх

об’єднання через варіативні сегменти (так звані петлі), тоді як інші пакети, як MODELLER [13], спочатку шукає множину вимог до білкової структури оригінальної послідовності на основі структур у складі вирівнювання та вимог фізичних силових полів (довжини зв’язків, кути та інше). Доволі цікавою є ідея повторення цього процесу, поки не знаходиться структура з найкращою оцінкою, наприклад, через генетичні алгоритми [8, 14]. При шаблон-незалежній частині програми виконують задачу моделювання петель - частин оригінальної послідовності, які не відповідають жодному білку серед знайдених шаблонів. Це відбувається або за рахунок пошуку схожих петель в інших білках, аналогічно кроку 1 (через бази даних петель) або моделювання аб інітіо (Монте Карло, молекулярна динаміка, генетичні алгоритми і т.д.);

5. Оцінка із застосуванням різноманітних критеріїв. Приблизним критерієм оцінки є схожість послідовностей. Якщо ідентичність двох послідовностей (даної та тієї, що застосовувалась для моделювання як шаблон) становить менше 35%, то якість падає дуже швидко зі зниженням ідентичності. Додатковими методами перевірки є сервіси PROCHECK, WHATCHECK, MolProbity, PROSA, Verify3D, які аналізують відповідність моделі статистичним (ймовірність ротамерів, цис-транс конфігурацій) та фізичним критеріям (стеричні зіткнення атомів, довжини зв’язків, кути) [8, 15-19].

Прикладом програми для моделюванні на основі шаблону є TASSER/I-TASSER [8, 20]. Автори цього підходу застосовують тредінг, тобто пошук схожого білкового фолду, на основі чого відтворюються схожі частини білка, в комбінації зі статистичними силовими полями, тоді як для відмінних застосовується аб інітіо моделювання на основі емпіричного силового поля. Воно застосовує статистичні параметри щодо частоти у

вторинних структурах, гідрофобності, водневих зв’язків, виведених зі структур в базі PDB. Різниця між TASSER та I-TASSER в додаванні другого раунду моделювання, що відчищає структуру від стеричних помилок, таких як стикання атомів [8, 20].

Однак якщо маємо справу з білками-сиротами чи дизайнерськими білками, які не схожі по послідовності на вже наявні структури, то приходиться застосовувати аб інітіо моделювання. Воно здебільшого більш складне та довге, а також ненадійне, однак існують методи покращити його точність. Наприклад, сюди відносяться методи моделювання зборкою фрагментів, де алгоритм обирає випадковий сегмент послідовності та застосовує на ньому конформацію випадково обраного фрагменту з бібліотеки фрагментів, а потім порівнює енергію до та після. Для цього використовується метод Монте Карло симуляцій та крупнозерниста функція енергії (апроксимація справжньою функції енергії, де білок представлений не у вигляді атомарного графу, а спрощеного графу - без повного представлення бічного ланцюга), що дозволяє розрахувати модель швидше та точніше [7, 8].

Загалом проблему білкового фолдингу з точки зору аб інітіо методів можна розділити на дві підзадачі: пошук оптимальної функції енергії та пошук оптимального стану в рамках цієї функції [21]. Якщо розглядати першу задачу, то тут багато варіантів вибору бажаної функції енергії (або силового поля), серед яких виділяються два кластери - засновані на фізичних апроксимаціях та статистичні. У першому випадку, однак, розрахунки займають роки чи місяці, тим самим є неоптимальними для швидкої оцінки даної структури, тоді як статистичні є дуже швидкими (дні та години), але дають доволі грубу оцінку якості структури. До фізичних належать відомі силові поля AMBER, CHARMM, OPLS та GROMOS96, під час пошуку оптимальної структури білка їх об’єднують з методами молекулярної динаміки, дозволяючи моделювати не тільки фінальну

нативну конформацію, а й загальний шлях фолдингу, тим самим отримуючи розуміння кінетики та закономірностей цього процесу в клітині. Прикладами проектів із застосуванням цього підходу є розподілений пошук нативних структур folding@home [22] та моделювання білку віліну [23]. Цей метод має інші недоліки, окрім довгого часу моделювання, такі як наявність в силових полях схильності до відтворення певного набору вторинних структур, що призводило до зниження якості фінального передбачення, порівняно навіть зі статистичними полями [21]. Зараз його застосування здебільшого обмежено до мінімізації енергії отриманої близької до нативної структури, передбаченої іншими методами, але навіть тут вони програють статистичним силовим полям. При застосуванні останніх доволі розповсюдженою є комбінування з моделюванням на основі фрагментів, оскільки емпіричні силові поля погано відтворюють вторинні структури, які залежать від тонких змін в глобальному та локальному середовищі послідовності. Окрім того, застосування фрагментів дозволяє знизити кількість можливих конформаційних станів, чим пришвидшує пошук нативного стану. Доволі відомий протокол для моделювання білків Rosetta [24] користується комбінацією описаних двох підходів - статистичних потенціалів із пошуком фрагментів та покращенням структури шляхом молекулярної динаміки на основі фізичних силових полів. Цей підхід виявився доволі успішним, але знову-таки, доволі повільним.

Другий компонент, що викликає проблеми при вирішенні проблеми фолдингу - це пошук оптимальної конформації серед неймовірної кількості помилкових чи неоптимальних структур. Для цього можна застосовувати молекулярну динаміку та Монте Карло підходи, однак загальна поверхня функції енергії, яку вони оптимізують, є дуже складною, з великою кількістю нерівностей, холмів та ямок, що значно ускладнюють застосування цих методів для оптимізації структур. Для Монте Карло

підходу характерне застосування протоколу симульованого відпалу або Метрополіс Монте Карло методу, де енергія нової структури порівнюється зі старою, і якщо вона є кращою, то автоматично приймається за нову, а якщо гіршою, то відкидається лише з певною заданою ймовірністю. Такий підхід дозволяє перескакувати локальні мінімуми та ефективно знаходити глобальний, однак для такого складного ландшафту, як енергетична функція білка, навіть цього недостатньо, і для цього методу характерно застрягати в певних метастабільних станах, які не є схожими на нативний. Молекулярна динаміка, як контраст до Монте Карло, дозволяє отримати повноцінні траєкторії фолдингу, ціною неймовірно високих розрахункових потужностей і семплить значно меншу кількість станів, ніж сучасні варіації Монте Карло, а тому застосовується тільки для моделювання траєкторії згортки [21]. Цікавою перевагою молекулярної динаміки є включення молекул води у явному вигляді, а не представленими як спеціальні потенціали, що дозволяє вирішити проблеми фолдингу білків, що не згортаються саме через помилкові взаємодії з водою. Окрім наведених, зустрічаються також методики, що використовують генетичні алгоритми (відпал конформаційного простору [25, 26], як приклад) та математичну оптимізацію шляхом розрізання конформаційного простору на половинки та оцінку значення мінімальної енергії (αBB метод [25, 26]).

Інший варіант аб інітіо передбачень представляє собою моделі машинного навчання, серед яких найкраще себе проявили нейромережі [6]. Саме їх ми розглянемо далі.

1.2. Нейромережеві моделі передбачення структури білків

Нейромережі представляють собою нелінійні апроксиматори функцій, які були вперше описані в середині 20 сторіччя на основі моделювання поведінки мозку [27, 28]. Вони складаються з декількох шарів нейронів, які приймають числові входи, обробляють їх за рахунок

матричних лінійних трансформацій і в кінці видають модифікований вихід, на який була накладена певна нелінійна функція. Загалом, шар нейромережі виглядає наступним чином:

$$Z^{[l]} = W^{[l]T} \times A^{[l-1]} + B^{[l]} \quad (1.1)$$

$$A^{[l]} = f(Z^{[l]}), \quad (1.2)$$

де $W^{[l]T}$ - матриця вагів зв'язків між нейронами цього шару $[l]$ та попереднього шару $[l-1]$, $A^{[l-1]}$ та $A^{[l]}$ - це матриці нелінійних активацій нейронів попереднього та поточного шару, а $B^{[l]}$ - матриця відхилень нейронів цього шару. Ваги та відхилення – це коефіцієнти самої моделі, які розраховуються при навчанні та дозволяють ефективно апроксимувати інформацію на виході.

Для різних архітектур та задач використовуються різні функції активації, наприклад, більшість сучасних нейромереж, серед яких різні та конволюційні моделі, використовують у проміжних шарах (та іноді в фінальних шарах теж, якщо треба передбачити дані, для яких всі можливі значення *y* більше 0) функцію ReLU:

$$f(x) = max\,(x,\ 0). \quad (1.3)$$

Тут і далі в формулах активаційних функцій x - позначення вектору, що обробляється, а $f(x)$ - позначення самої функції. Тоді як рекурентні нейромережі застосовують гіперболічний тангенс:

$$f(x) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}}. \quad (1.4)$$

Нарешті, для моделей із застосуванням уваги (трансформерів) характерна заміна ReLU на GELU, через емпіричні причини покращення навчання:

$$A(x) = x * P(x \leq X), \quad X \sim N(0,\ 1) \quad (1.5)$$

Тут $P(x \leq X)$ означає кумулятивну функцію вірогідності для нормального розподілу з характеристиками середнього $E(X) = 0$ та стандартним відхиленням $\sigma(X) = 1$.

Для перших нейромереж було характерне застосування гладкої функції, що нагадувала поведінку порогу в нейроні - так звана сигмоїда:

$$f(x) = \frac{1}{1 + e^{-x}}. \quad (1.6)$$

Зараз вона та її генералізація для взаємовиключних багатокласових передбачень (softmax) здебільшого застосовуються лише для фінального шару для генерації передбачень у вигляді ймовірності належності до того чи іншого класу. [29, 30]

Застосування декількох нелійнійних функцій підряд разом з великою кількістю нейронів дозволяє моделювати які завгодно складні нелінійні залежності за рахунок пошуку оптимальних вагів та відхилень шляхом зниження числа помилок [31]. Для пошуку значень коефіцієнтів нейромережі, які дають мінімальне значення функції втрат (процес тренування) вчені використовують математичний метод градієнтного спуску:

$$\theta_i = \theta_{i-1} - \alpha \nabla J(\theta), \quad (1.7)$$

де θ_{i-1} – попередня матриця параметрів нейромережі θ, яка задається випадково на самому початку тренування, α – швидкість навчання, що є невеликим дробовим числом, зі значенням за замовчуванням 0.001 чи менше, $\nabla J(\theta)$ – градієнт для функції помилок відносно матриці параметрів.

Перед тим, як подати на вхід до нейромережі послідовність амінокислот виникає питання перетворення літер на цифри, оскільки моделі глибокого навчання не вміють працювати з чистим текстом. Для цього використовується трюк зі сфери обробки природньої мови (NLP), який називається шар векторів-вставок (embeddings). По факту це словник, який перетворює амінокислоти послідовності білка у числові вектори

певного обраного до цього розміру [31]. Цей словник тренується разом з іншими вагами нейромережі, а тому закодовує певну інформацію про амінокислоти, однак що саме то за інформація - незрозуміло, а тому він, так само як і інші ваги нейромережі, вимагає інтерпретації.

Серед основних архітектур нейромереж застосовуються конволюційні нейромережі (вони були особливо популярні на перших історичних етапах розвитку [6]), рекурентні нейромережі та трансформери [32]. Вибір конволюційних нейромереж здебільшого виправдовується через їх застосування у сфері обробки зображень, в даному ж випадку обробляється матриця $l \times l$ (де l - довжина послідовності білка), що представляє собою матрицю попарних взаємодій, яка перетворюється у матрицю контактів чи дистанцій. Конволюційні нейромережі побудовані на ідеї операції конволюції - спеціальній операції комбінування двох функцій:

$$f(t) * g(t) = \int_{-\infty}^{\infty} f(\tau)g(t - \tau)d\tau. \quad (1.8)$$

Оскільки ми працюємо із зображеннями, то ця операція має працювати з дискретними наборами значень, а не неперевними, а тому набуває наступного вигляду:

$$f(n) * g(n) = \sum_{m=0}^{M} f(m)g(n - m). \quad (1.9)$$

Тут n є спеціальним параметром зсуву, а M - розміром послідовності дискретного сигналу. Ця операція є дуже популярною в сфері обробки сигналів, особливо зображень, де дозволяє працювати з шумом та вирізняти краї (вертикальні, горизонтальні та їх комбінації) [31]. Ідея конволюційних нейромереж - натренувати таку функцію, що складається із багатьох фільтрів невеликого розміру, які дозволяють вирізняти важливі для передбачення характеристики даних через конволюцію, надаючи їй перевагу в навчанні. Ця перевага є наслідком побудови нейромережі таким

чином, що обробка певного типу даних стає природньою для неї (тобто використовуються властивості цих даних), що вимагає менше навчання та менше параметрів на запам’ятовування особливостей датасету. Описане явище називається індуктивним упередженням [33].

Прикладами нейромереж, що побудовані на основі конволюцій, є RaptorX, AlphaFold та TrRosetta [34-36]. Усі три мережі передбачають двовимірні матриці дистанцій, які представлені як тензори $l \times l \times c$ (де c - кількість інтервалів для дистанції між амінокислотам), перетворюючи задачу регресії на задачу класифікації [34-36]. Окрім дистанцій, таким же самим чином передбачуються торсійні кути між амінокислотами. Іншим цікавим нововведенням на цій стадії розвитку нейромереж для фолдингу білків стало використання так званої коеволюційної інформації - виявилось, що з вирівнювання послідовностей білків, навіть тих, для яких немає структури, можна отримати важливу структурну інформацію на основі вивчення мутацій, що корелюють між собою [37]. Усі три моделі приймають як вхід декілька послідовностей у вигляді матриці вирівнювання (MSA) розмірів $n \times l$ (де n - кількість послідовностей), а їхня точність знижується при наданні одиничної послідовності.

Інша архітектура, що доволі часто зустрічається в нейромережах для фолдингу - це так званий трансформер [32]. Це модель, яка застосовує механізм уваги для створення мап коваріацій, що далі відсіюють неважливі попарні взаємодії. Вона розроблялась на заміну рекурентним нейромережам, які мали недолік забування важливих елементів послідовності зі зростанням її розміру та нестабільного навчання (накопичення градієнтів на довгих послідовностях призводило до їх зниження до 0 чи збільшення до нескінченності). Фундаментальна ідея механізму уваги - це вибіркове ігнорування неважливої інформації залежно від контексту та аналізованого слова, яке базується на вивченні нейромережею повної послідовності, а не її частини, що до цього робили

рекурентні нейромережі. Спочатку вхідна послідовність (представлена як матриця $N \times E$, або розмір послідовності на розмір векторів-вставок) перетворюється в три різні матриці: запит, ключ та значення (query, key та value):

$$Q = XW_Q, K = XW_K, V = XW_V. \quad (1.10)$$

Тут W_Q, W_K та W_V - відповідні матриці перетворення, які є параметрами мережі. Далі виконується матричне множення запиту на ключ з отриманням квадратної матриці коваріацій:

$$Z = QK^T. \quad (1.11)$$

Ця матриця в подальшому нормалізується (поділом на вимірність ключа, причини знову-таки емпіричні та пов'язані з кращим навчанням [38]) та перетворюється в ймовірнісні бали (скільки уваги приділяти кожному слову) із застосуванням softmax:

$$S = softmax(\frac{Z}{\sqrt{d_k}}). \quad (1.12)$$

Останнім етапом є множення балів уваги на матрицю значень, що можна інтерпретувати як "скільки порцій кожного слова треба взяти з контексту, щоб зрозуміти значення вибраного слова?":

$$A = SV. \quad (1.13)$$

Тобто загальна формула для шару уваги виглядає наступним чином:

$$A = softmax(\frac{QK^T}{\sqrt{d_k}}) \times V. \quad (1.14)$$

Ця модель показала свою ефективність у обробці послідовностей слів в тексті (NLP), а тому була очікуваним етапом розвитку і для білків, що представлені у біоінформатиці як послідовності амінокислот. Серед типових прикладів таких нейромереж для фолдингу - AlphaFold2, RoseTTAFold, ESMFold та RGN2 [39-42]. Перші дві нейромережі зберігають свою залежність від MSA інформації та застосовують

архітектуру, аналогічну до так званого SE(3) еквіваріантного трансформера. [43] Суть в тому, що структура білка має певні особливі симетрії, які можна застосувати при побудові архітектури нейромережі, а саме еквіваріантність до обертів та трансляції [44]. Тобто якщо застосувати якусь зміну до структури білка, то його обернена чи зміщена версія має змінитись так само, але з урахуванням оберту чи зміщення:

$$h(\sigma(x)) = \sigma(h(x)). \quad (1.15)$$

Цей трюк дозволяє зменшити похибку при валідації, зменшити загальну кількість параметрів та прискорює навчання [44].

Тим часом останні дві моделі намагались отримати точність AlphaFold2 без використання MSA, тобто максимально наблизитись до святого Граалю аб інітіо (де ново) фолдингу - отримання 3D структури лише з одиночного сиквенсу. Ці моделі застосовують схожий підхід для заміни MSA - моделювання мови (language modeling). Для цього у послідовностях білків відбувається маскування випадкових амінокислот з вимогою до моделі заповнити ці пробіли. Це доволі просто сформульоване завдання некерованого (а скоріше, самокерованого - self-supervised) навчання вимагає нейромережу вивчити можливі зв’язки між амінокислотами, щоб точно відновлювати послідовності білків. Для застосування цієї мережі в передбаченні структури останні шари, що необхідні для класифікації, прибирають та використовують проміжні вектори, які (в теорії) мають бути насичені найрізноманітнішою інформацією. Різниця між ESMFold та RGN2 полягає в тому, як вони декодують інформацію з цих векторів - перша модель надає їх на вхід до нейромережі, аналогічної за архітектурою до AlphaFold2, тоді як у RGN2 наявний модуль у вигляді рекурентної нейромережі [41, 42].

1.3. Пояснення моделей машинного навчання через XAI

Нейромережеві моделі є так званими моделями чорного ящику (black box), оскільки інтерпретація процесу розрахунку результату не є інтуїтивної для людини, порівняно з, наприклад, моделями лінійної регресії чи деревами прийняття рішень. Окрім того, процес градієнтного спуску визначає пошук такого шляху між входами та виходами, який оптимальний для нейромережі, але необов'язково підлягає легкій інтерпретації людиною. Все це, разом з великою кількістю параметрів у найкращих сучасних моделей глибокого навчання, призводить до неможливості наївного пояснення на основі аналізу вагів, що мотивує розробку специфічних методів для пояснення нейромереж [45-49].

Доволі розповсюдженою є наступна класифікація XAI методів по стадії, на якій відбувається інтерпретація [45, 46]:

- Ante-hoc інтерпретація - побудова моделей, що легко інтерпретуються, тобто які по своїй структурі прозорі. Здебільшого такі моделі або надто прості (в результаті отримуємо модель, яка легко інтерпретується, але яка має низьку точність), або стають настільки складними, що потребують додаткової інтерпретації після їх побудови. Сюди належать методи як лінійна регресія, дерева прийняття рішень, метод *k* найближчих сусідів та інші [45, 46].
- Post-hoc інтерпретація - інтерпретація вже після отримання готової моделі. Ділиться на два підтипи: модель-агностичні та модель-специфічні методи. Модель специфічні методи можна застосовувати до дуже специфічного набору моделей по структурі, наприклад, Grad-CAM застосовується лише для інтерпретації конволюційних нейромереж. Модель-агностичні методи, тим часом, мають перевагу в можливості їх застосування незалежно від типу моделі машинного навчання, наприклад, контрфактичні методи, що шукають мінімальну зміну у вході для отримання максимальної

зміни на виході, аналізуючи таким чином поведінку системи чорного ящику [45-46].

Іншим важливим поділом методів XAI є по обсягу інтерпретації, де відокремлюють локальне та глобальне пояснення. Локальне пояснення означає отримання інтерпретації, зрозумілої для людини, лише для одного чи пари прикладів, тоді як глобальне пояснення означає пояснення всієї моделі. До локальних пояснень належать методи на основі аналізу збурення (пертурбаційні методи) такі як LIME, SHAP, якорі та оклюзія [1-3, 50], тоді як для глобальних застосовується пошук векторів концептів за допомогою методу TCAV [51].

Окрім цих, більш-менш поширених класифікацій, існують інші, менш поширені та менш погоджені, такі як за загальним принципом пошуку пояснення, де виділяють 5 різних типів: локальні збурення (вже описані пертурбаційні методи), використання структури (наприклад, методи на основі аналізу градієнту для нейромереж), метапояснення (пояснення збираються із різноманітних методів та агрегуються з утворенням нового, більш загального та кращого пояснення), модифікація структури (дозволяє спростити модель та легше інтерпретувати, формуючи більш прозорі моделі) та пошук показових прикладів (сюди належить пошук контрфактичних прикладів); на основі даних, що аналізуються: числові, текстові, зображення та навіть даних на виході (текст, числа, правила, зображення та змішані) [45, 46, 47]. Цікавим прикладом модифікації структур для підвищення прозорості є механізм уваги, який по своїй суті будує матриці коваріації між елементами входу, а тому має сенс проводити візуалізацію цих квадратних матриць з метою пошуку закономірностей. Однак пізніше виявилось, що цей широко розповсюджений наївний метод є доволі ненадійним, а тому не рекомендується для застосування при аналізі трансформерів [52, 53].

Таке велике різноманіття методів викликає закономірне питання - які саме застосовувати та в яких ситуаціях? Це питання доволі складне, але шлях його вирішення можна візуалізувати з урахуванням наведених вище класифікацій. Перше питання, що треба вирішити, це визначення, що саме очікується від пояснення - тобто мета проведення XAI дослідження. Наприклад, зусилля можна зосередити на розумінні помилок моделі для її дебагінгу чи побудова такої моделі, яка викликає достатню довіру в користувачів. Тобто потрібно глибоке розуміння вимог зацікавленої сторони та їхніх особливостей (наявність чи відсутність експертного знання, сприйняття візуалізацій та інше). Друге питання - це відповідність методу XAI для задачі. На цю тему існує доволі невелике різноманіття робіт, але автори однієї з них рекомендують зробити спеціальні ідентифікаційні картки для кожного методу [48] і, після розуміння вимог до дослідження по інтерпретації моделі, перевіряти кожну з них на сумісність. Враховуються такі наведені характеристики як обсяг інтерпретації, його тип, тип моделі та даних, на яких модель натренована. Наприклад, Grad-CAM надає лише локальні пояснення у вигляді мап значимості, він працює лише для конволюційних моделей та пояснює лише зображення [49, 54]. Окрім того, для деяких методів відомі різноманітні обмеження, пов’язані з розвитком новітніх архітектур. Наприклад, LRP та DeepLIFT (методи на основі градієнтів) погано працюють для рекурентних нейромереж (особливо LSTM), а для трансформерів вимагають модифікацій до шарів пошарової нормалізації та уваги для збереження головних вимог цих методів (в даному випадку, це закон збереження релевантності) [55-57]. Окрім цього, багато методів, особливо заснованих на збуреннях, такі як LIME, SHAP та якорі, чи вище описаних градієнтних методів LRP, DeepLIFT та інтегрованих градієнтів, вимагають доволі довгих розрахунків та комп’ютерних потужностей для обчислення [49]. Нарешті, навіть з урахуванням всіх проблем виникає

питання того, які з цих методів дають *найкращі* пояснення. Бенчмарк 12 методів та їхніх варіацій (серед яких 9 унікальних методів та 3 варіації із SmoothGrad) на метриках максимальної чутливості (робастність методів XAI до шуму), AUC-MoRF (наскільки видалення найбільш важливих пікселів за оцінкою XAI методів впливало на передбачення оригінальною моделлю), розміру файлів (для оцінки кількості інформації у фінальних результатах) та швидкості виконання показав, що найкращими по точності є методи Grad-CAM, оклюзія та LIME, однак вони видають доволі грубу оцінку (що стає зрозуміло по найменшим серед усіх розмірам файлів), та, за виключенням першого, ці методи розраховуються доволі довго [49].

Тема пояснення нейромереж є особливо вагомою в медицині, де це дозволяє зрозуміти причини збоїв системи та вберегти здоров’я пацієнтів від можливих помилок системи, однак дослідження на тему XAI нейромереж білкового фолдингу не дуже розповсюджені, проте серед тих, що існують, розповсюджені декілька підходів: фізичний аналіз передбачень нейромереж та проміжних векторів, контрфактичні пояснення (ExplainableFold) [58], аналіз навчання (OpenFold) [59], та сурогатні прозорі моделі чи чорноящикові моделі з більш простою архітектурою. Останні будуть розглянуті в наступному підрозділі, а більшу увагу ми зосередимо на перших трьох.

Доволі розповсюдженим є такий аналіз проміжних та фінальних векторів, що пов’язує їх статистично з іншими фізичними властивостями амінокислот, доменів чи повних білків. Наприклад, при аналізі передбачень AlphaFold2 групою дослідників було встановлено, що основна перевага цієї моделі заключається в вивченні нею правильної функції енергії, яка дозволяє навіть без MSA оцінювати якість отриманих білкових структур. Окрім того, вони встановили, що MSA дозволяють серйозно звузити коло можливих нативних структур у рамках пошуку цього стану нейромережею. Загальна схема виглядає наступним чином: спочатку

AlphaFold2 визначає приблизне положення оптимальних координат (цей етап потребує MSA), після чого воно поступово покращується через застосування вивченої енергетичної функції. Таким чином, навіть якщо знати функцію енергії ідеально, то цього буде недостатньо для фолдингу, бо виникає проблема швидкого пошуку оптимуму серед великої кількості можливих просторових структур. Ця проблема була вже вказана в першому підрозділі, де розглядались характеристики аб інітіо методів. Для позбавлення AlphaFold2 від залежності на MSA, пропонується додати генераторну нейромережу до вивченої енергетичної функції, що буде представляти генератор, таким чином створюючи GAN модель на основі оригінальної моделі [60].

Інший кут, під яким можна проаналізувати AlphaFold2, це як нейромережа навчається - що стається з проміжними та фінальними векторами під час тренування. Цим займались автори OpenFold, нейромережі, яка є спробою реплікації оригінальної нейромережі разом з вивченням причин, чому в цієї моделі вийшло, а в інших ні, різноманітних переваг архітектури та інших цікавих інсайдів стосовно того, як оперує ця складна модель. Так, було встановлено, що в процесі навчання моделі білків поступово набувають нових вимірів, починаючи з точки (так координати ініціалізуються), перетворюючись в одновимірну лінію, розтягуючись в двовимірну поверхню (із формуванням приблизних вторинних структур), а в кінці ця поверхня набухає, розтягується, ніби повітряна кулька, та формує тривимірну структуру з точними передбаченнями вторинної структури. Цей процес порівнюється авторами з PCA - ніби модель спочатку вивчає одновимірну проекцію тривимірної структури, потім двовимірну, а потім повністю рекапітулює тривимірну. Наступні етапи вводять маленькі поправки в ці моделі, особливо у компоненти вторинної структури, оскільки багато менш розповсюджених вторинних структур на цих стадіях можуть передбачатись погано. У першу

чергу вивчається альфа-спіраль, далі бета-ланцюг, далі пі-спіраль, бета-поворот та інші, що нагадує частоту цих елементів у білках [59].

При аналізі методами XAI моделей трансформерів на основі BERT доволі розповсюдженим є вивчення векторів-вставок та проміжних векторів. Як приклад, в статті, що описує тренування та оцінку серії нейромереж ProtTrans, яка моделює “білкову мову” через завдання по маскувальному моделюванню (MLM), було проведено детальний опис отриманих векторів-вставок та проміжних векторів [61]. Перші, побудовані як представлення амінокислот в багатовимірному просторі, доволі чітко кластеризувались по фізико-хімічним властивостям, тоді як для других було проведено маркування по дослідженим властивостям - вторинна структура (трьохкласова система - альфа спіраль, бета ланцюг, невпорядкований регіон), розчинність (чи знаходиться білок на мембрані), локалізація в клітині (цитозольний, ядерний, мітохондріальний і т.д.), SCOPe класифікація (альфа, бета, альфа+бета, альфа/бета, мультидоменний і т.д.), організм, з якого взятий білок (передбачення на рівні доменів - віруси, бактерії, археї чи еукаріоти), навіть функції білків (передбачення по класифікації ензимів EC). Аналіз відбувався за рахунок візуалізації технікою t-SNE (проекція в двовимірний простір зі збереженням кластеризації), порівняння з випадково ініціалізованою моделлю та побудовою алгоритмів керованого навчання на основі натренованих в рамках дослідження мовних моделей. Такий варіант вивчення дозволяє чітко прослідити поток інформації в нейромережі та що кодує кожен вектор, однак вимагає розрахункових потужностей та постановки чітких гіпотез [61].

Попередні методи здебільшого зосереджені на аналізі проміжних та фінальних векторів, тоді як автори методу ExplainableFold вирішили присвятити свою увагу вивченню оптимальної методології для генерації локальних пояснень [58]. Їхній метод заснований на об’єднанні методів

направленого мутагенезу, який застосовувався дослідниками для вивчення фолдингу, із методом контрфактичних пояснень. Як вже згадувалось, суть цього методу у внесенні незначних контрольованих збурень до входів, щоб максимізувати зміну виходу моделі. Цей аналіз дозволяє виявити елементи входів, до яких система приділяє найбільше увагу при передбачені, і періодично застосовується при вивченні безпечності нейромережевих алгоритмів від т.з. адверсійних (ворожих) атак. Для нейромереж пошук таких контрфактичних прикладів є тривіальним завданням, оскільки градієнт $\nabla_X Y = \frac{dY}{dX}$ по факту шукає такий напрямок мінімальної зміни X, що призводить до максимальної зміни Y. Однак в даному випадку автори оптимізували матрицю замін (амінокислота замінювалась на невідому), щоб TM-скор був менше 0.5 для необхідного пояснення, і більше 0.5 для достатнього пояснення (тут задача була змінена на зворотню, тобто пошук такої великої підмножини змін в послідовності, щоб зміни в структурі були мінімальні). Для розрахунку мінімальної/максимальної зміни у вході застосовували L1 функцію помилок (модуль до матриці змін) та пісумовували її з оцінкою подібності структур через TM-скор. Їхній метод зміг успішно прив’язати отримані значення важливості окремих амінокислот до різноманітних біохімічних метрик заміни амінокислот, як-от дистанція Епштейна, дистанція Міята (засновані на аналізі різниці розмірів та полярності) та еволюційна оцінка (аналізує консервативність змін в певній позиції послідовності). Інша робота, що вивчала чутливість RoseTTAFold до адверсійних атак через пошук контрфактичних пояснень, виявила, що ця нейромережа не робастна, тобто дуже чутлива до зміни в послідовності білків [62]. Якщо попереднє дослідження приймало частоту замін на невідому амінокислоту як розмір збурення, то в даному випадку автори оцінювали дистанцію між двома входами через матрицю BLOSUM62, яка видає оцінки між -4 до 11 для замін однієї амінокислоти

на іншу [63]. Ця попарна метрика розраховується між двома однаковими немутованими послідовностями та парою, яка містить немутовану та мутовану амінокислоту. Як наслідок, різниця між ними дозволяє оцінити, наскільки серйозною є зміна в послідовності. Для оцінки зміни структури застосовувалась різниця дистанцій між двома амінокислотами на різних кінцях послідовності. Хоча це дуже приблизна метрика, вона дозволяє швидко знайти приблизну максимальну дистанцію в структурі (яка є інваріантною від обраної системи координат). В результаті оптимізації було показано, що частина білків дуже критично змінюються від мутацій в послідовності дистанцією 20 умовних одиниць BLOSUM62. Це дослідження не тільки показує можливість застосування методів збурення у вивченні поведінки нейромереж, а й підкреслює необхідність розуміння їх механізмів та обмежень, що отримуються лише через XAI дослідження.

1.4. Побудова сурогатних моделей

Однією з варіацій пояснення моделей машинного навчання є - побудова моделі сурогата, задача якої - відтворювати відповіді оригінальної моделі, але при цьому бути доволі простою та прозорою, щоб отримувати розуміння поведінки системи [64]. Методи, що формують сурогатні моделі, є по своїй суті модель агностичними, оскільки єдине, що треба для тренування сурогату - наявність входів та передбачених оригінальною моделлю виходів. Сурогати розділяють на локальні та глобальні, і у випадку локальних сурогатів для їх побудови застосовується метод LIME (метод локальних інтерпретованих модель-агностичних пояснень) [1]. Суть методу LIME в побудові локальної моделі, що задовольняє критерію:

$$E(x) = argmin_{g \in G} L(f, g, \pi_x) + \Omega(g), \qquad (1.16)$$

де $E(x)$ - пояснення, g є локальним сурогатом з множини усіх можливих сурогатів G, що мінімізує функцію помилок L між $f(x)$ (виходи з оригінальної моделі) та $g(x)$ (виходи з сурогату) та складність сурогату $\Omega(g)$. Важливим параметром для тренування LIME є міра близькості (π_x), яка вказує на окол, з якого беруться приклади для побудови локальної моделі. Цей гіперпараметр є однією з причин складності застосування цього методу, оскільки необхідний пошук його оптимальних значень для отримання якісних моделей. Окрім того, різні значення π_x можуть призводити до отримання діаметрально протилежних описів залежностей. Однак, як вже вказувалось раніше, цей метод надає одні з найкращих пояснень серед усіх інших XAI методів. Окрім того, він надає прості та зрозумілі інтерпретації, доступні для розуміння і без експертного знання, особливо якщо тренувати сурогат на певній трансформації оригінальних входів до моделі. Тобто, модель може бути натренована на даних, що не є зрозумілі для людини (як в NLP, наприклад, тренування йде на реченнях, що перетворені у матриці через вектори-вставки), але сурогат тренувати на властивостях цих слів, побудованих вручну дослідником. Це може надавати певне спотворення до інтерпретації, особливо якщо дослідник намагається приховати упередженість моделі [1, 64].

Для оцінки якості сурогату, чи глобального, чи локального, можна застосовувати різні метрики. Для LIME нами була вже наведена функція помилок $L(f, g, \pi_x)$, яка може бути у форме MSE, MAE для регресії чи кроссентропія та KL дивергенція для класифікації, R^2 для більш загальних задач [1, 64].

На місці сурогатів можуть бути найрізноманітніші моделі білого ящику (прозорі моделі), серед яких лінійна регресія, Ridge чи Lasso. які мають наступну спільну загальну форму:

$$\widehat{Y} = \overline{W}\,\overline{X} + \overline{B} \tag{1.17}$$

$$L(Y, \widehat{Y}) = (Y - \widehat{Y})^2 + \alpha * R(\overline{W}, \overline{B}) \tag{1.18}$$

де $\widehat{Y} \in R^m$ представляє з себе передбачення моделі, що наближують вектор істинних виходів Y, $\overline{W} \in R^{m \times n}$- матриця вагів до відповідних елементів вектора входу $\overline{X} \in R^n$, $\overline{B} \in R^m$ - вектор відхилень, $L(Y, \widehat{Y})$ - це функція помилок, що ми намагаємось оптимізувати, α - регуляризаційний коефіцієнт, а $R(\overline{W}, \overline{B})$ - регуляризаційна функція для параметрів моделі. Для лінійної регресії ця функція представляє з себе вектор нулів, для Lasso $R(\overline{W}, \overline{B}) = \sum^{nm} W_i^2 + \sum^{m} B_j^2$, а для Ridge $R(\overline{W}, \overline{B}) = \sum^{nm} |W_i| + \sum^{m} |B_j|$.

Доволі важливою перевагою застосування Lasso та Ridge проти звичайної лінійної регресії є їх робастність до мультиколінеарності, де Lasso перетворює всі коефіцієнти, окрім одного, для коваріантних властивостей на нуль, тоді як Ridge наближає їх до нуля без видалення [64].

Для класифікації розповсюдженими моделями є логістична регресія та наївний Баєс [64]:

$$\overline{p(C \mid X)} = \widehat{Y} = \sigma(\overline{W}\,\overline{X} + \overline{B}) \tag{1.19}$$

$$\overline{p(C \mid X)} = \frac{\overline{p(C)} \times \prod^{n} \overline{p(X_n \mid C)}}{\overline{p(X)}}, \tag{1.20}$$

де $\overline{p(C \mid X)} = \widehat{Y} \in R^m$ представляє з себе передбачені ймовірності кожного з класів моделі, що наближують вектор істинних ймовірностей класів Y, на основі вектора входів $\overline{X} \in R^n$, в наївному Баєсі вони представляють постеріорний розподіл ймовірностей класів, що отримується з апріорного $\overline{p(C)}$ через множення на правдоподібність $\overline{p(X \mid C)}$ та нормалізацію через ймовірність спостереження відповідних значень входів незалежно від класу $\overline{p(X)}$. Логістична регресія тренується

через метод найбільшої правдоподібності, що, якщо розписувати формулу, по факту представляє з себе мінімізацію кросентропії [64]:

$$L(Y, \widehat{Y}) = -\sum^{m} \widehat{Y}_i \log Y_i. \quad (1.20)$$

Для наївного Баєсу процедура оцінки ймовірності значно простіше і не вимагає ніякого тренування, весь процес описаний в головній формулі. Однак важливо врахувати одну з умов, що спрощують розрахунок, але розуміння якої дозволяє помітити, коли така модель провалюється - властивості, що подаються на вхід, мають бути незалежними, бо інакше розмір векторів та матриць в формулі зростає експоненційно.

Нарешті, існують моделі білого ящика, які можна ефективно застосовувати і для регресії, і для класифікації, такі як дерева прийняття рішень, k найближчих сусідів та RuleFit. Для дерев прийняття рішень характерна наступна формула:

$$\widehat{Y} = \sum^{L} k_l \times I(X_l, R_l). \quad (1.21)$$

Тут k_l представляють коефіцієнти листка дерева, а $I(X_l, R_l)$ - спеціальну функцію належності, яка вирішує, чи потрапляє властивість X_l з векторів входу $\overline{X}$ до множини R_l, яка задається певним булевим твердженням, наприклад $X_l < 10$. Для оптимізації цих моделей застосовується метод CART, який шукає такі критерії для вузлів дерева, що надають найбільше зменшення варіативності виходів (для регресії) та найбільше зменшення індексу Джині (описує, наскільки нерівномірно розподілені класи в певному вузлі, з намаганням в даному випадку зменшити кількість класів у вузлі до одного). Хоча ці моделі є доволі інтуїтивними, їх інтерпретабельність зменшується експоненційно зі зростанням либини, оскільки так само експоненційно зростає кількість

листків. Таким чином, оптимальні дерева для інтерпретації мають бути дуже маленькі, але вони мають маленьку точність [3, 64].

З дерев прийняття рішень можна отримати набір правил для побудови алгоритму на основі лінійної регресії, що дозволяє враховувати взаємодії між елементами вхідного вектору, який називається RuleFit. По факту це звичайна лінійна регресія, в якій входом є вектор $\overline{X}$ та вектор на основі перетворень з правил прийняття рішень, розмірністю в кількість вузлів в дереві, тому і функція помилок, і інтерпретація є аналогічними таким для лінійної регресії. Недоліки всі ті самі, що і для інших лінійних сурогатів - велика кількість елементів вхідного вектору погіршують інтерпретацію, а враховуючи, що частина цих елементів є даними по взаємодії, роз'яснення поведінки RuleFit моделі стає неінтуїтивним [64].

Серед прикладів застосування прозорих та інтерпретованих моделей є дослідження RBM для аналізу закономірностей початку та кінця альфа-спіралей та бета-ланцюгів. Це моделі некерованого навчання, задача яких - відновити наданий вхід після проведеного зжимання до більш низьковимірного простору. За рахунок вивченого процесу представлення великих кількостей даних у багатовимірному просторі зі значно меншою розмірністю, модель проводить кластеризацію та групування елементів, що можна застосовувати для отримання нових характеристик для різноманітних цілей. У даній роботі RBM зжимали інформацію від послідовностей амінокислот, і в результаті отримали ваги для розрізнення декількох типів альфа-спіралей та бета-ланцюгів. Було виявлено, що хоча велика кількість бета-ланцюгів і справді має чергування полярна амінокислота-неполярна амінокислота, у великої частки цих структур відбувається переривання цієї закономірності, а модель починає генерувати структури, що складаються майже виключно з гідрофобних амінокислот, тоді як для альфа-спіралей дуже розповсюдженим є початок з проліну, чергування полярних-неполярних і кінець з поліаланіну [65].

РОЗДІЛ 2
МЕТОДИ ДОСЛІДЖЕННЯ

2.1. Пошук оптимальної мережі та методів

Для того, щоб отримати інсайди з нейромереж про процес фолдингу білків, необхідно адекватний об’єкт для аналізу (нейромережа, що відповідає дослідженню) та відповідні методи XAI. Оскільки основна цікавість саме в вивченні процесу згортки, то потрібні такі нейромережі, які проводять de-novo фолдинг чисто з поодинокої послідовності. Таким чином відсіюються нейромережі, що застосовують MSA для звуження простору конформаційного пошуку. Сюди належать AlphaFold2 та RoseTTAFold. Окрім того, RoseTTAFold не підходить для дослідження, оскільки вона є недостатньо робастною, а тому при вивченні поведінки нейромережі на основі методів збурення чи оптимізації градієнтів, будемо спостерігати багато випадкових кореляцій. Вимога вивчення нейромереж, що спочатку приймають поодиноку послідовність, а далі насичують її інформацією про фізико-хімічні властивості амінокислот та структурні властивості білка, дозволяє зосередитись на лише декількох моделях, що застосовують маскувальне мовне моделювання для навчання корисним векторним представленням білків. Сюди належать мережі RGN2 та ESMFold. Основна їх різниця в структурі даунстрім модуля (що приймає як вхід векторну інформацію від трансформерної мовної моделі на основі BERT), відповідно і в швидкості виконання. ESMFold є доволі великою моделлю на основі BERT та евоформера з AlphaFold2, яка при завантаженні для передбачення займає майже всю пам’ять на GPU, а тому аналіз деякими з методів XAI (такі як на основі збурення чи інтегровані градієнти), які вимагають багатьох викликів моделі, стає неможливим через ресурсоємність, що звужує коло потенційних методів-кандидатів. Як контраст, RGN2 є навпаки, доволі зручною моделлю, яка є відносно

невеликою (її BERT модуль є меншим за такий для ESMFold, а для декодеру застосовується рекурентна архітектура, що дозволяє мати меншу кількість коефіцієнтів) і через це дозволяє пропускати під час виконання розрахунків не один приклад, а батч (декілька прикладів, представлених сукупно як матриця чи тензор). Її точність на білках-сиротах перевищує таку для AlphaFold2, а на інших білках вона поступається моделі від DeepMind доволі незначно [42]. Окрім того, на даний момент відсутні роботи, присвячені аналізу методами XAI цієї мережі. Саме тому, для дослідження була обрана модель RGN2, що є доволі невеликою та робить високоточні (на рівні AlphaFold2) передбачення de-novo без застосування MSA. Оскільки повний аналіз нейромережі займає дуже великий час, було вирішено зосередити увагу на першому модулі RGN2 під назвою AminoBERT. Ця частина відповідальна за насичення інформацією поодиноких послідовностей білків та отримання векторів, багатих структурною та фізико-хімічною інформацією (це є необхідною умовою високої точності нейромережі при виконання завдання заповнення пробілів в послідовностях у рамках самокерованого навчання), а тому була винесена гіпотеза, що цієї частини достатньо для отримання високоякісних передбачень вторинної структури та навіть дуже грубих оцінок третинної.

Оскільки AminoBERT є загальною моделлю, що видає фінальні вектори із різноманітною інформацією, було побудовано дві моделі на її основі, щоб можна було вивчити процес передбачення вторинної та третинної структури. Модель вторинної структури, яка вивчалась в цій роботі, представляє з себе AminoBERT модуль та LDA шар (по факту лінійне перетворення та softmax активація), натренований через SVD, через що було введено обмеження на кількість проаналізованих амінокислот (у результаті фінальну модель було натреновано на 75000 передбаченнях класу вторинної структури, до якої належить певна амінокислота). Стосовно третинної структури, було обрано завдання

передбачення матриці дистанцій, через що потрібно було перетворення фінальної матриці $N \times 768$, де N - довжина послідовності, на $N \times N$. Для цього застосовували білінійне перетворення, представлене наступним чином: $\widehat{Y} = XWX^T$, де $\widehat{Y}$ - передбачена матриця дистанцій, X - фінальні вектори модуля AminoBERT, W - вивчені ваги взаємодії. Модель навчалась на 4000 матрицях дистанцій, згенерованих на основі високоякісних передбачень AlphaFold2, оптимізуючи функцію втрат середньої абсолютної похибки, при цьому паралельно моніторилась кореляція між передбаченими та справжніми дистанціями.

Стосовно обрання методів дослідження, як вже вказувалось в попередньому розділі, ця задача вимагає розгляду під різними кутами: мета XAI аналізу, обсяг інтерпретації, його тип, тип моделі та даних, на яких модель натренована. У даному випадку ми намагаємось отримати інсайди в реальний фізичний феномен на основі закономірностей, виведених моделлю через тренування, шукаємо глобальне пояснення для химерної моделі, яка натренована на текстових вхідних даних та цифрових табличних вихідних даних. Якщо перше та друге є більш-менш зрозумілими вимогами, то стосовно третього пункту треба зробити невелике роз’яснення, що мається на увазі. RGN2 є химерною архітектурою в сенсі, що вона комбінує в собі трансформерний BERT модуль та рекурентний модуль. Це вже накладає обмеження на методи, які можна застосовувати - при інтерпретації через методи LRP або DeepLIFT, які є розширеннями методу Gradient x Input, для мереж рекурентного типу чи трансформерів виникає порушення головного принципу цієї групи XAI підходів, а саме принципу збереження - сума балів, що видає такий алгоритм (вони називаються релевантністю), в кожному шарі має бути однаковою. Автори цієї статті змогли розширити LRP для трансформерів, а тому і для BERT, але ціною релаксації цієї умови та вимогою не

пропускати градієнт через коефіцієнти уваги та дисперсію в шарі LayerNorm для збереження цього варіанту принципу консервації [57]. Нарешті, окрім теоретичних (чи можна застосовувати той чи інший метод та за яких умов) виникають інші, більш прагматичні питання - наявність розрахункових ресурсів (високі вимоги до обрахункових потужностей для методів на основі збурень) та якість отриманих пояснень.

На основі цих вимог було прийнято рішення взяти методи, що є модель-агностичними, а тому (як частинний випадок) не будуть залежать від архітектури нейромережі, та знехтувати проблемами обрахункової потужності з необхідністю отримати високоякісні пояснення - ці вимоги задовольняють лише методи на основі збурень, тобто LIME, SHAP та якорі. Ці методи надають здебільшого лише локальні пояснення та не пов’язують їх ніяк з властивостями амінокислот (лише з їхньою ідентичністю), а тому вимагають певної переробки. Серед трьох наведених методів лише LIME дозволяє використовувати такі вектори, які ненапряму пов’язані з оригінальними входами (вектор оригінальних входів можна отримати певною трансформацією), а тому для побудови локальних моделей використовували фізико-хімічні властивості амінокислот як “природні вставки”. Для адаптації якорів застосовували побудову сурогатних моделей на основі дерев, які прямо надають метрики покриття та точності (деталі в підрозділі про сурогатні моделі). Нарешті, SHAP використовувався для пояснення як лінійних локальних моделей в рамках нашої адаптації SHAP, так і нелінійних моделей на основі дерев в рамках адаптації якорів, через пакет SHAP для мови Python. Він надає функціонал для пояснення яких завгодно моделей з різноманітними гнучкими візуалізаціями, які особливо корисні при синтезі багатьох прикладів поведінки моделі в глобальне пояснення.

Серед інших цікавих альтернатив, які можна виконувати паралельно, є аналіз фінальних та проміжних векторів, контрфактичні

пояснення, аналіз навчання та сурогатні моделі. Аналіз фінальних та проміжних векторів є доволі простим та гнучким підходом, який може доповнювати отримані пояснення методами збурень. Наприклад, аналіз векторів-вставок дозволить зрозуміти, яка інформація є важливою нейромережі на вході, а аналіз проміжних векторів - як вона перетворюється в процесі матричних розрахунків до фінального вектору виходів. Оскільки аналізу підлягає нейромережа, навчена методикою самокерованого навчання, навіть фінальні вектори не мають очевидної інтерпретації, а тому вимагають пояснення також. Ці кінцеві матриці чисел в подальшому є необхідними для обрахунку каркасу рекурентним модулем, а тому основна інформація про тривимірну структуру білка також має бути закодована тут. Через це для перевірки наведених гіпотез було запропоновано побудувати прості моделі-декодери до основного модуля з метою передбачення вторинної та третинної структури. Контрфактичні пояснення, хоча й є доволі привабливим методом, надають лише локальне пояснення, і для отримання глобального розуміння поведінки моделі потрібно згенерувати багато таких пояснень, що насправді не відрізняється від застосування для цього методу LIME, а тому було вирішено його не застосовувати. Нарешті, сурогатні моделі є фінальним етапом синтезу отриманого знання, його генералізацією з локальних пояснень в таке глобальне пояснення, яке оперує як функція вхідних чисел, а тому для цієї роботи є святим Граалем, до якого наближались.

2.2. Пошук та генерація даних для дослідження

Оскільки метою цього дослідження було виведення закономірностей фолдингу білків з індивідуальних властивостей амінокислот, то необхідно зібрати достатньо даних про їх фізико-хімічні властивості, організувати це в табличку чи матрицю і в подальшому застосовувати її як інтерпретовані

вектори-вставки. Було зібрано наступні дані про амінокислоти: маса, об’єм, площа поверхні, гідрофобність, частота в послідовностях, pKa для карбоксильної групи, pKa для аміногрупи, pI при 25℃, дипольний момент, аромафільність та потенціал взаємодії електрон-іон (11 властивостей). Окрім того, було отримано різноманітні дані по статистичним властивостям амінокислот - так звані схильності (propensity), наприклад, схильність до альфа спіралей, чи схильність до знаходження на поверхні (5 властивостей), однак вони були застосовані скоріше лише для вивчення векторів-вставок AminoBERT. Їхня статистичність та непрямий зв’язок із іншими фізико-хімічними параметрами мотивує виключення при подальшому аналізу процесу фолдингу. Нарешті, була відкинута ще одна властивість (клас - полярна, неполярна, з сульфуром), з якої доволі важко працювати через її категоричну природу. У подальшому саме до першої групи даних корелювали отримані результати від оригінальної моделі та сурогатів.

Наприклад, таким чином провели аналіз векторів-вставок з моделі AminoBERT. Природні вставки корелювали до нейромережевих не напряму, а через визначення відстаней між векторами-вставками нейромережі та евклідову дистанцію між властивостями, оскільки нас цікавило їх відносне розташування в просторі, плюс отримана вибірка перетворюється з 20 векторів 1×768 з великою можливістю випадкових кореляцій з 20 векторами розмірністю 1×16 (матриця вагів лінійної регресії 16×768), на 190 скалярів з кореляцією на 190 векторів розмірністю 1×16 (матриця вагів лінійної регресії 1×16). Таким чином, обраний метод пошуку зв’язку дозволяє аналізувати відносні значення (кластеризацію) замість абсолютних, що є більш інтуїтивним, а також запобігає оверфітингу лінійної регресії через збільшення кількості прикладів.

В рамках глобальної задачі пояснення передбачення вторинної та третинної структури потрібно отримати не тільки моделі для цього, а й відповідні дані. Для передбачення вторинної структури застосовувався датасет Рональда Данбренка [66], що складається з 8712 послідовностей, розділених на тренувальний, валідаційний та тестувальний датасети. Він містить близько 2,2 млн токенів, представлених 20 амінокислотами та спеціальним символом для невідомої амінокислоти (X). Для передбачення було обрано набір значень, де вторинна структура закодована як один з 9 класів DSSP - H (альфа спіраль), E (бета ланцюг), C (випадковий клубок), G (3/10-спіраль), T (поворот з водневим зв'язком), S (вигин), B (залишок бета-містка), I (π-спіраль) та X (невпорядкована структура), що в подальшому були перетворені на вектори ймовірностей належності до класу [67].

Оскільки обробка такого великого датасету для побудови моделі передбачення вторинної структури займає великий час, було виділено 1000 послідовностей, на основі яких в подальшому й проводились перевірки. Для третинної структури використовувались передбачення AlphaFold2 з AlphaFoldDB, оскільки ці моделі є повними (не містять відсутніх атомів чи амінокислот), порівняно з такими в PDB, а тому їх легше обробляти. Матриці дистанцій генерувались для моделей білків з видів, позначених в базі як soybn (соя), human (людина), mycul (мікобактерія), euro (евроціоміцети) та schma (шистосоми), на основі дистанцій між C_α атомами амінокислот, лише для тих моделей, де всі ці атоми мали pLDDT (впевненість в передбаченні, що корелює з точністю AlphaFold2) > 50%, оскільки нижче цього надані координати вважались ненадійними авторами статті [68]. Відфільтрований датасет склав близько 7000 білків, однак помістити всі ці матриці в пам'ять для тренування є неможливим, а тому

було вирішено знову взяти певну вибірку з них (4500 матриць дистанцій - 4000 в тренувальний та по 250 у валідаційний та тестувальний).

Для побудови локальних сурогатів в рамках методу LIME необхідно отримати датасет, що складається з оригінальної послідовності, та її збурених версій. Генерація збурених послідовностей з невеликою дистанцією (як вимагає LIME для збереження лінійності) від оригінального вектору входів в рамках NLP парадигми проводилась на основі заміни токенів на синонімічні. Щоб збурені послідовності були достатньо близькі до основної, а також з метою врахування всіх можливих замін, в процесі генерації для нової послідовності обиралась певна випадкова частота мутації (між 1% та 20%), а сам процес вибору нового токену був стохастичним (але пропорційним скору по матриці BLOSUM62, щоб частіше обирались синонімічні заміни, якими вважались амінокислоти з великим скором). Загалом збиралось 1000 різних послідовностей та передбачень з налаштованої під це варіації моделі AminoBERT, які потім розбивались на вікна певної розмірності і далі застосовувались для передбачення. Це робилось для спрощення розрахунків та з урахуванням залежності вторинної та третинної структури в певній позиції від її відстані в текстовій послідовності. Оптимальний розмір вікна приймався за гіперпараметр та шукався під час тренування локальних сурогатів. Такий підхід призводить до проблеми мультиколінеарності, тому для створення незалежних властивостей та зменшення розмірності вставок застосовувався метод PCA з порогом у 99% варіативності оригінальних природних вставок.

Оскільки в рамках передбачення третинної структури було сформульовано через генерацію матриць дистанцій, то тут підхід з вікнами вимагав покращення та переосмислення. Подача на вхід немодифікованих природних вставок призводить до великих складнощів у процесі моделювання матриці дистанцій навіть в рамках LIME, оскільки модель

має застосовувати не тільки відносні значення властивостей (їх різницю чи схожість), а й абсолютні. Уявімо ситуацію в гідрофобному ядрі, де знаходяться три пари - дві об’ємні амінокислоти, одна об’ємна та одна маленька та дві маленькі. В даному випадку дві об’ємні будуть мати найбільшу відстань, за ними йде об’ємна та мала, і останньою йде пара малих. Іншими словами, якщо застосовувати лише схожість, то виникає проблема - перша та третя пари мають високу схожість по об’єму, але різну дистанцію. Окрім того, формування вікон проводилось з матриць дистанцій таким чином, що амінокислоти зліва могли бути відносно далеко від амінокислот справа (іншими словами, вікна могли генеруватись далеко від основної діагоналі). Ця ситуація відмінна від генерації вікон для вторинної структури, де всі амінокислоти у вікні були сусідами в послідовності, а тому вимагає пряме кодування позиції як ще однієї характеристики. Характеристики, що використовувались для передбачення матриць дистанцій, будувались на описаних вище припущеннях про важливість як абсолютних, так і відносних значень. Обмеження пам'яті (ще більше, ніж для вторинної структури) змусило нас використовувати лише вибірку у 60000 прикладів для вікон у матрицях відстаней, яку ми вибирали з верхнього трикутника матриці відстаней, щоб уникнути дублювання та спростити навчання.

2.3. Побудова сурогатних моделей та візуалізація результатів

Для отримання пояснень використовували методи на основі збурення LIME, якорі та SHAP. Оскільки вони видають лише значення для окремих токенів-амінокислот, то їх треба пов’язати із певними фізико-хімічними значеннями, для чого проводилась адаптація методів, тобто, пов’язати числа з природними вставками. Сюди належать локальні лінійні сурогати (LIME) та локальні нелінійні сурогати у вигляді дерев рішень (якорі). Для виконання LIME застосовувалось велике різноманіття лінійних моделей,

які порівнювались між собою та обирались ті, що адресували наші задачі найкраще. Так, в рамках моделювання вторинної структури необхідно передбачати ймовірність певного класу C_i - $p(C_i)$, що вимагає застосування мультикласової регресії, однак такий підхід зневажає тим, що модель видає не дискретні значення, а неперервні ймовірності, тому також спробували лінійну регресію та логістичну регресію з метою наближення ймовірностей із моделі до ймовірностей сурогата. Ці два останні підходи були відсіяні через їхню занизьку якість та теоретичну невиправданість. У подальшому шукались параметри моделі мультикласової регресії, що мінімізували похибку, через оптимізацію правдоподібності пакетом statsmodels на мові Python. Для третинної структури описаний вище підхід зі створенням 24 властивостей з 11 призводить до ще гіршої проблеми мультиколінеарності, однак, тут використання простого PCA недостатньо і, іноді, може призвести до зменшення якості моделі (оскільки PCA може виключити ознаки з низькою дисперсією, але великою важливістю для прогнозу)! Нам потрібно використовувати алгоритм, який краще підходить для цього, ніж звичайну лінійну регресію. Ми обрали часткові найменші квадрати (PLS) та Ridge регресію (лінійна регресія з L2 регуляризацією). В результаті найкращим виявився метод Ridge регресії, тобто регресії з регуляризацією, і саме його застосовували для вивчення третинної структури. Пошук оптимального параметру α (коефіцієнт регуляризації) відбувався шляхом Leave-One-Out кросвалідації, який був реалізований, як і модель Ridge регресії, через пакет scikit-learn мови Python. Оцінка якості сурогатів (як вторинної структури, так і третинної) на етапі моделювання локального пояснення відбувалась через обчислення R^2 як метрики визначення частки поясненої варіативності, а статистичних параметрів моделі - двома шляхами. Для вторинної структури пакет statsmodels видає t-критерій та статистичну значимість, а тому шукались такі коефіцієнти,

для яких остання була менше 0,05. Для третинної структури scikit-learn видає лише R^2, а тому для оцінки параметрів моделі застосовували методику бутстрапінгу - семплінгу з датасету із заміною. Всього було виконано 1000 процедур семплінгу (без урахування першого етапу, при якому обиралось оптимальне значення α) для кожного локального пояснення третинної структури, і в кожному випадку на основі цього були розраховані 2,5 та 97,5% перцентилі, необхідні для відсіювання статистично незначимих коефіцієнтів із вірогідністю 5%.

У рамках нашого розширення методу якорів продовжували аналіз локальних даних, але на цей раз будували локальні нелінійні моделі, які по своїй структурі представляють випадковий ліс (багато дерев рішень в тандемі). Якість цього методу покращується, якщо застосовувати процедуру бустингу в рамках пакету xgboost мови Python. В рамках бустингу кожна нова модель коригує помилки попередньої, а її результати додаються до результатів старої моделі. Хоча інтерпретація дерев є нескладною задачею, інтерпретація випадкового лісу, глибоких дерев і, тим паче, xgboost дерев є нетривіальною, а тому саме для цього застосовувався пакет SHAP для мови Python, що має спеціально облаштовану варіацію методу пояснення для дерев - т.з. TreeSHAP. Разом із наданням від усіх моделей-дерев метрик покриття та зменшеної варіації (оптимізація Джині індексу), ці моделі дозволили отримати додаткові інсайди про правила згортки та те, за яких умов вони працюють. Для вторинних структур оцінка моделей відбувалась за допомогою метрик точності (accuracy), влучності (precision), повноти (recall), F1 скору, макро точності та зваженої точності. Ці всі метрики відрізняються врахуванням істинних та неістинних, позитивних та негативних передбачень, а також зваженням на незбалансованість класів у датасеті. Ці метрики аналізувались в рамках методу classification_report від scikit-learn. Для

третинних структур оцінка відбувалась із застосуванням середньої абсолютної похибки як метрики точності.

Для інтеграції отриманих закономірностей здійснювали порівняння з глобальними сурогатами, що мали ідеальний баланс між інтерпретаційною здатністю та точністю. В даному випадку застосовувались дерева xgboost, що показали свою високу надійність та точність при моделюванні локальних залежностей для локальних сурогатів. Інтерпретацію проводили із застосуванням вищезгаданого пакету SHAP, а отримані результати порівнювались напряму з такими ж результатами цього ж методу на цьому ж датасеті в локальних сурогатах.

Для вторинної структури, де вдалось знайти цікаві глобальні закономірності, функціональні залежності, розраховані з попереднього етапу, застосовуються для побудови власної моделі у вигляді легко інтерпретованих формул. Ця формула складається з залежності від позиції, що надає ваги для підсумовування індивідуальних властивостей амінокислот для кожної конкретної позиції, та фінальної класифікаційної матриці, що перетворює фінальні агреговані властивості локального середовища на певний клас вторинних структур. Її оцінка відбувалась так само, як і локальних та глобальних сурогатів вторинної структури на основі дерев - з використанням методу classification_report scikit-learn, однак навчання проводили в пакеті Tensorflow, що використовується для тренування нейромереж - він надає можливості для оптимізації параметрів для нелінійних моделей шляхом градієнтного спуску. При цьому оптимізували функцію втрат під назвою категорична кросентропія, що часто застосовується під час навчання класифікаційних моделей з кількістю взаємовиключних класів більше двох. У процесі тренування намагались отримати найкращу можливу точність від моделі при найменшому розмірі, а тому вивчались, які можливі деталі моделі є неважливими чи, навпаки, можуть допомогти при класифікації, шляхом

тренування нових моделей зі схожою структурою, але певною відміною. Аналіз точності по класам також був проведений, аби встановити, які частини моделі допомагають передбаченню яких вторинних структур, а тому підтвердити чи спростувати старі гіпотези, виведені з попередніх сурогатних моделей.

Відкриття закономірностей є неможливим без візуалізації даних, але в умовах вивчення багатовимірних даних пошук оптимальних методів візуалізації є нетривіальною задачею. Однією з методик, що ми користувались доволі часто, є метод t-SNE, який оптимізує таке розміщення багатовимірних векторів у двовимірному просторі, щоб їхня кластеризація зберігалась. Цей метод є розповсюдженим при візуалізації векторів-вставок в сфері NLP, але має свої проблеми та підводні камені. По-перше, метод стохастичний, тобто вимагає генерації декількох зображень для розуміння справжньої кластеризації. По-друге, метод дуже чутливий до гіперпараметру перплексії, який визначає очікувану відстань між сусідами. Це призводить до необхідності поекспериментувати як зі значеннями перплексії, так і значенням сіду для генерації адекватної візуалізації. Хоча цей метод все ще виявився незамінним при виконанні роботи, в деяких випадках оптимальним було замінити його на методи, що показують кластеризацію напряму, як дерева ієрархічної кластеризації.

При поясненні сурогатів на основі лінійної регресії здебільшого застосовувались доволі прості візуалізації (демонстрації матриць, графіки залежності від позиції з урахуванням статистичної оцінки значень коефіцієнтів), тоді як для пояснення дерев бібліотека SHAP надає велике різноманіття візуалізацій - від доволі простих та типових (як стовпчикові гістограми чи діаграми розсіювання), до таких складних та багатоінформативних як рій бджіл, багатофакторні хіт мапи та графіки траекторій рішень.

РОЗДІЛ 3
РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ

3.1. Залежність векторів-вставок від амінокислотних властивостей

Першим етапом пояснення нейромережі було обрано вивчення векторів-вставок. Цей процес доволі детально задокументований для задач в сфері обробки мови, а тому його реалізація доволі нескладною. Оскільки нас цікавив лише зв’язок векторів-вставок для амінокислот з їхніми фізико-хімічними властивостями, то було вирішено не проводити аналіз так званих допоміжних токенів - токен початку послідовності, маскувальний токен, токен для невідомого слова та інші були проігноровані. Багатовимірні вектори-вставки амінокислот (20 векторів розмірністю 768) було зображено двома методами - через t-SNE та через дерево ієрархічної кластеризації.

Серед деяких зрозумілих інсайдів, що можна отримати на даному етапі, є те, що певні фізико-хімічні властивості справді позначаються на розміщені в просторі векторів-вставок. Дуже очевидним та таким, що легко кидається в очі, є взаємне розташування ароматичних амінокислот (триптофан, фенілаланін, тирозин), негативно заряджених амінокислот (аспарагінова та глутамінова кислота) та їх амідних похідних (аспарагін та глутамін), близькість великих позитивно заряджених амінокислот (лізин та аргінін). Окрім того, отриманий графік для цих векторів-вставок схожий із аналогічним графіком для MLM моделі ProtAlbert, розробленої в рамках дослідження з моделювання “мови” білків [61].

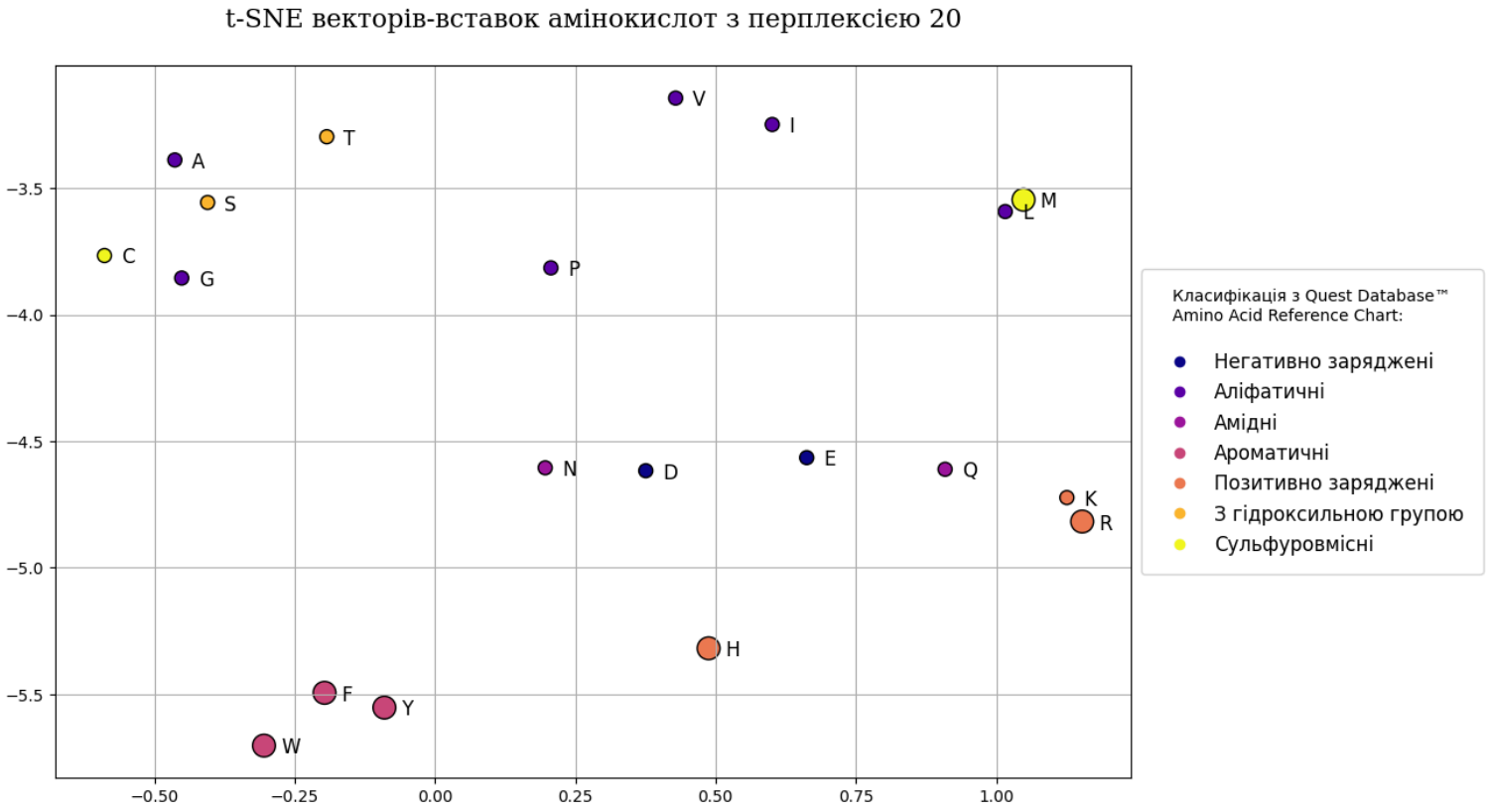


Рис. 3.1. Результат t-SNE проекції на двовимірну площину векторів амінокислот AminoBERT при перплексії 20. Великі кульки зображають амінокислоти масою більше 130 дальтон (Да)

Однак оскільки t-SNE доволі важко інтерпретувати через його стохастичну природу та залежність від параметру перплексії, було також проведено ієрархічну кластеризацію. Можна побачити, що існує певна залежність між фізико-хімічними характеристиками, але вона неочевидна. Як приклад, на найнижчих рівнях можна помітити, що аргінін та лізин як позитивно заряджені лужні амінокислоти розміщуються поблизу, схожа ситуація для аланіну та гліцину (обидві маленькі), триптофану, фенілаланіну та тирозину (ароматичні), серину та треоніну (обидві мають полярну незаряджену гідроксильну групу) та аспартату і глутамату (кислотні негативно заряджені). Однак більш високорівневі залежності встановити важче. Наприклад, незрозумілі причини розкиданості

сульфуровмісних амінокислот та більша дистанція від позитивно заряджених амінокислот до гістидину, ніж до глутаміну.

Рис. 3.2. Результат ієрархічної кластеризації векторів амінокислот AminoBERT, представлений у вигляді дендрограми, та класифікація амінокислот з Quest Database™ Amino Acid Reference Chart (кольори літер внизу)

Тому, провели порівняння з іншою класифікаційною схемою, що була розроблена першою вченою в сфері біоінформатики Маргарет Дейхофф [69]. Вона розробила систему з 6 класів амінокислот, серед яких:

1. Амінокислоти з властивістю формувати ковалентні сульфурні містки-зшивки - цистеїн;
2. Малі амінокислоти - гліцин, аланін, серин, треонін, пролін;
3. Кислотні амінокислоти (негативно заряджені) та їхні амідні похідні - аспарагінова кислота, глутаматна кислота, аспарагін та глутамін;

4. Лугові (позитивно заряджені) амінокислоти - аргінін, лізин та гістидин;
5. Гідрофобні амінокислоти - валін, лейцин, ізолейцин та метіонін;
6. Ароматичні амінокислоти - фенілаланін, тирозин та триптофан.

Рис. 3.3. Результат ієрархічної кластеризації векторів амінокислот та їхня класифікація, розроблена Маргарет Дейхофф (кольори літер внизу)

Із рис. 3.3. можна зрозуміти, що ця класифікація є значно ближчою до кластеризації амінокислот в багатовимірному просторі векторів-вставок AminoBERT. Навіть випадки, які є виключеннями, насправді в певній мірі підтверджують правило - до жовтого кластеру належать усі малі амінокислоти по Дейхофф та цистеїн. По своїй структурі цистеїн та серин дуже схожі (з заміною кисня в серину на сульфур цистеїну), що показує те, що усі амінокислоти в жовтому кластері розміщені поблизу через свій невеликий розмір. До зеленого кластеру належать усі неполярні аліфатичні

амінокислоти, до блакитного - усі ароматичні, нарешті, до синього - усі великі полярні, як незаряджені (аспарагін та глутамін), так і заряджені (усі інші).

Звичайно, подібних спостережень недостатньо, потрібні підтвердження чи спростування за допомогою статистичного аналізу. Як вказувалось у попередньому розділі, відстані між векторами-вставками нейромережі корелювали до евклідової дистанції між властивостями. Це дає оцінку характеристик, що є важливими для відносного розміщення амінокислот в просторі векторів-вставок, а не абсолютних значень елементів векторів. Отримали матрицю вагів лінійної регресії 1×16 та оцінили довірчі інтервали через пакет statsmodels. Велика кількість властивостей виявились колінеарними (проявлялось як велике число обумовленості - 3630), а тому вирішили їх почистити на основі аналізу кореляцій. На основі цього залишили лише 4 властивості, серед яких - маса, гідрофобність, частота амінокислот у послідовностях та дипольний момент. Результати регресії виявились наступними, див табл. 3.1.

Табл. 3.1. Результати регресії для дистанції між вставками та різницею властивостей, $R^2 = 0.498, R^2_{adj} = 0.487$

Параметр моделі	**Середнє**	**2.5 % перцентиль**	**97.5% перцентиль**	**t значення**
Константа перетину	4.5253	4.269	4.781	34.876
Гідрофобність	0.0091	0.007	0.011	7.697
Частота амінокислот	0.4499	0.212	0.687	3.736
Дипольний момент	0.0121	0.005	0.019	3.324
Маса	0.0161	0.012	0.020	8.160

З цього можна зробити доволі простий висновок, що основними властивостями, які визначають відносне розміщення векторів-вставок амінокислот моделі AminoBERT є гідрофобність, маса, частота амінокислот в послідовності та дипольний момент.

3.2. Закономірності вторинної структури з сурогатних моделей

Модель RGN2 (а тим паче AminoBERT) не передбачає вторинну структуру, а тому першим етапом було побудувати модель на основі фінальних векторів AminoBERT, яка передбачає вторинну структуру. В процесі підбору найпростішого та найефективнішого методу для цієї задачі зупинились на методі лінійного дискримінантного аналізу (LDA), оскільки порівняно з нейромережею як додатковим модулем до передбачень AminoBERT, точність на валідації відрізнялась незначно (60% проти 61% відповідно), при тому, що структура LDA моделі є значно простішою. Як можна побачити знизу, це лише один шар нейромережі з функцією активації softmax:

$$\widehat{Y} = softmax\,(W^T X\ +\ B), \tag{3.1}$$

де W є матрицею вагів розмірністю $1 \times E \times N_{classes}$, X - матрицею входів розмірністю $N_{data} \times E \times 1$, а B - вектором відхилень розмірністю $1 \times N_{classes}$ (тут N_{data} - розмір датасету, $N_{classes}$ - кількість класів, а E - розмірність вхідного вектору, в даному випадку це розмірність фінального вектору AminoBERT - 768). Навчання проводили через SVD на основі 150 тис. пар амінокислота-вторинна структура, а перевірку - на окремих 75 тис. парах, що складали тестовий датасет, на основі гіпотези, що цього достатньо для передбачення вторинної структури - тобто фінальні вектори AminoBERT мають так чи інакше закодовувати в собі інформацію про вторинну структуру кожної відповідної амінокислоти. Результати наведені в табл. 3.2 (наступна сторінка).

Табл. 3.2. Результати класифікації на датасеті вторинних структур за допомогою нейромережі AminoBERT з LDA

	Влучність (precision)	**Повнота (recall)**	**F1-скор (F1-score)**	**Кількість прикладів**
H	0.806	0.838	0.821	24319
E	0.700	0.748	0.723	15106
C	0.401	0.506	0.448	13679
T	0.411	0.408	0.410	7773
G	0.193	0.095	0.127	2826
S	0.266	0.124	0.170	5849
B	0.109	0.037	0.055	757
I	0.000	0.000	0.000	0
X	0.640	0.574	0.605	4691
Точність	0.606	0.606	0.606	--
Макро точність	0.392	0.370	0.373	75000
Зважена точність	0.587	0.606	0.593	75000

Таблиця 3.2, з результатами розрахунків, наведена для тестового датасету. Різниця між тестовим та тренувальним виявилась невеликою - точність для тренувального склала 62.18 %, а для тестового - 60.65 %. Ця точність є доволі схожою з результатами класифікації невеликих білкових мовних моделей, вивчених у рамках дослідження ProtTrans (від 48% до 70%). Проблеми здебільшого виникають з передбаченням таких вторинних структур, які є доволі рідкисними - G, B, S та I мають найменшу точність та найменшу представленість в датасеті (всі пари амінокислота-клас I, яких було 5, потрапили в тренувальну збірку даних).

Наступним завданням було проаналізувати цю модель та знайти, яким чином властивості амінокислот пов’язані з вторинною структурою

білка. Для цього було створено 12 локальних моделей в рамках пояснення через LIME (6 локальних лінійних моделей) та якорі в комбінації з SHAP (6 локальних нелінійних моделей - дерев прийняття рішень). Для обох методів потрібна генерація збурених (мутованих) послідовностей, яка описана в попередньому розділі у деталях.

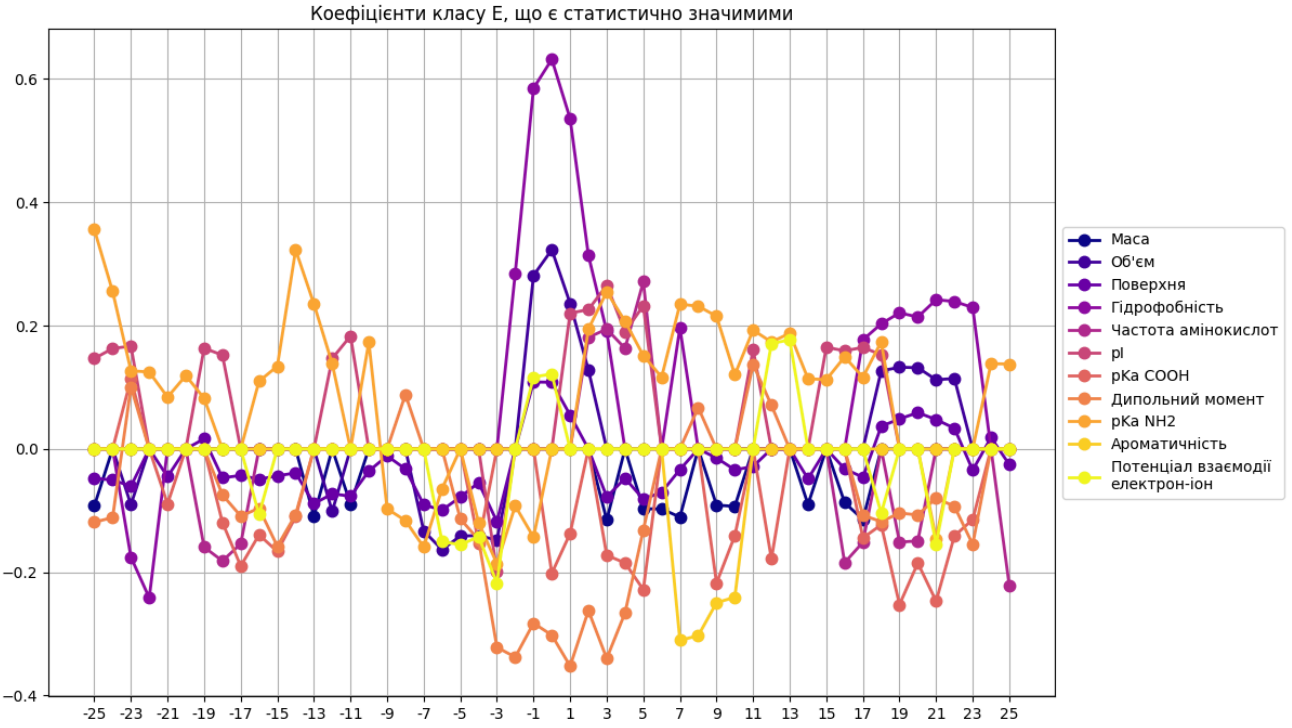


Рис. 3.4. Приклад параметрів моделі LIME для білку деацетоксицефалоспорин-С синтази (PDB ID: 1E5I) ланцюга A та класу вторинної структури E (бета-ланцюг)

Після отримання датасетів для цього завдання підібрали наступну лінійну модель для вторинної структури: багатокласова логістична регресія, що оптимізується пакетом statsmodels через процедуру оцінки максимальної правдоподібності, та для класифікації застосовує властивості амінокислоти, для якої йде передбачення, та її сусідів - 25 вліво та 25 вправо (тобто розмір вікна становив 25*2+1=51). Гіперпараметр розмір вікна в 51 амінокислоту був підібраний на основі

експерименту та отриманих псевдо-R^2. Для нелінійних моделей ця метрика застосовується замість R^2 та розраховується через відношення правдоподібності повної моделі та нульової:

$$R^2_{pseudo} = 1 - \frac{L_{null}}{L_{model}}, \quad (3.2)$$

де L_{null} - лог-правдоподібність нульової моделі, а L_{model} - лог-правдоподібність моделі, що досліджується. Для оптимізації застосовували Алгоритм Бройдена — Флетчера — Гольдфарба — Шанно (BFGS метод) [70], який дозволяє отримати результати навіть при ідеальному розділенні.

Отримана модель виявилась доволі складною (з урахуванням кількості коефіцієнтів - 51×11, та необхідності обрахунку для них статистичної значимості у вигляді інтервалів), а тому вирішили її спросити, відкинувши ті коефіцієнти, інтервали яких з ймовірністю 0,05 включають 0. Таким чином отримуємо спрощені графіки наступної форми, див. рис. 3.4.

Ми провели аналіз таких графіків для LIME моделей 6 білків (з псевдо R^2 для кожної з них 0.6959, 0.6737, 0.6112, 0.8038, 0.8304 та 0.5148), побудованих на основі MNLogit регресії модуля Statsmodels, та отримали різні закономірності, серед яких тут ми описуємо ті, що характерні для всіх білків. Наприклад, ми бачимо, що для бета-листів об’єм, гідрофобність (пряма залежність) та дипольний момент (зворотна) відіграють основну роль і знаходяться в безпосередній близькості до фактичної точки, в якій робиться прогноз для листа, що, власне, і цікавить нас. Крім того, для усіх 6 білків листки містять амінокислоти, які доволі часто зустрічаються в білкових структурах (оскільки в даному випадку брали від’ємний логарифм, то справжній коефіцієнт є додатним).

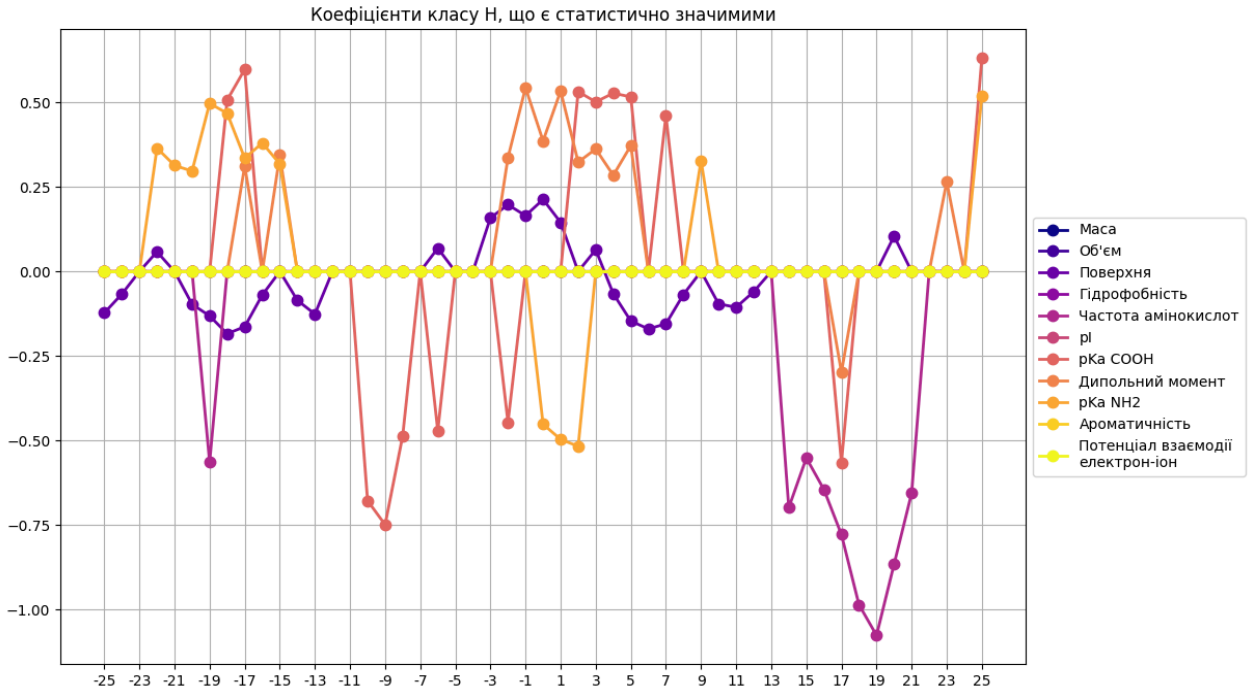


Рис. 3.5. Приклад коефіцієнтів для LIME моделі білка УДФ-глікозилтрансферази 79 ланцюг A з рису (PDB ID: 6BK0). Видно, що велика кількість коефіцієнтів не досягає порогу статистичної значимості

Деякі структури, однак, досить рідко зустрічаються в аналізованому білку, тому іноді ми отримували діаграми із невеликою часткою коефіцієнтів, які насправді є статистично значущими. Як дуже наочний приклад, наведемо деякі параметри класу H.

На щастя, нам достатньо даних з інших білків, щоб прийти до наступних висновків. Клас альфа-спіралі здебільшого позитивно корелює з гідрофобністю, площою поверхні, масою, pKa для групи COOH та дипольним моментом, з єдиною стабільною негативною залежністю у вигляді pKa для групи NH2 (можливі причини для цієї кореляції будуть описані далі, при аналізі даних з дерев). Інші кореляції є доволі хаотичними та прив’язаними до специфічних білків. Для випадкового клубка (клас C). Як і очікувалося, гідрофобність має негативну кореляцію (оскільки ця структура в основному складається з полярних амінокислот),

те ж саме стосується маси, поверхні (поблизу точки передбачення) та від'ємного логарифму частоти амінокислот (позитивний коефіцієнт для ймовірності), проте випадковий клубок позитивно корелює з pKa для аміногрупи та pI поблизу опорної точки, в той час як для pKa для групи COOH спостерігаємо негативну кореляцію.

Для інших структур, таких як вигини та повороти, ми спостерігали негативну залежність від маси, поверхні, об’єму, гідрофобності та потенціалу взаємодії електрон-іон (EIIP), поряд з рідкісними амінокислотами, що проявляються як будівельні блоки (позитивна -log p(AA) кореляція). Нарешті, серед тих класів, що залишились, тільки клас G (π-спіраль) має хоч якісь залежності, що можна детектувати. Тут основною характеристикою є коливання коефіцієнтів для дипольного моменту, поверхні, pKa для NH2 та, в деяких випадках, негативна залежність від EIIP.

Які ж характеристики вирізняють класи між собою? Для цього зберемо коефіцієнти з 6 білків в єдиний графік, і на основі нього проаналізуємо характерні відмінності. По-перше, дуже легко помітити, що велика частина цих залежностей представляють дихотомію впорядкованих проти невпорядкованих регіонів. Наприклад, залежність від маси та гідрофобності інвертується в невпорядкованих структур, порівняно з такою для впорядкованих. Однак більш цікавою є різниця всередині класів. Так, для розрізнення альфа-спіралей та бета-ланцюгів застосовується дипольний момент (позитивний коефіцієнт проти негативного відповідно), pKa для COOH та pKa для NH2 (позитивний та негативний коефіцієнти для альфа-спіралі та зворотна ситуація для бета-листа), різна ступінь залежності від гідрофобності (бета ланцюг має більш позитивний коефіцієнт для гідрофобності, ніж альфа-спіраль) та об’єму/поверхні (бета-ланцюгу характерний високий позитивний коефіцієнт для об’єму і нульовий до поверхні, тоді як альфа-спіралі -

позитивний для поверхні та нульовий для об’єму). Різниця між вигинами та поворотами з водневими зв’язками пов’язана залежностями від pKa біля точки передбачення (pKa для COOH має позитивний коефіцієнт, а pKa для NH2 - негативний для повороту, однак спостерігаємо зворотню ситуацію для вигину) та дипольного моменту (позитивна для повороту та нульова-негативна для вигину).

Тепер розглянемо проблему з іншого боку - побудуємо моделі, що виконуватимуть ціль якорів та генеруватимуть правила з наших локальних датасетів. Для отримання високоякісних моделей, для яких пропрацьовані методи пояснення, використали метод побудови дерев XGBoost, що генерує багато дерев, кожне з яких покриває помилки попереднього. Для пояснення застосовували метод TreeSHAP на основі Шаплі значень та їхніх модифікації для пояснення деревних моделей. Так згенерували, знову-таки, 6 локальний моделей, які, однак, в цей раз є нелінійними. Окрім того, для порівняння локальних моделей з глобальною базовою, було створено модель для передбачення вторинної структури, натреновану на тому самому наборі даних, що і AminoBERT+LDA. Після навчання моделей виявили, що локальні моделі мали точність 0,80-0,90 на вихідних локальних наборах даних, 0,25-0,40 на глобальному наборі даних і 0,40-0,50 на іншому локальному наборі даних, ніж той, на якому вони були навчені, що означає, що локальні моделі, хоча і погано узагальнюють невидимі дані, все ж змогли вловити деякі ідеї та кореляції, важливі для передбачення вторинної структури. Глобальна модель отримала точність 0,50 на оригінальному глобальному наборі даних, і 0,50-0,60 на локальних наборах даних, на яких вона не навчалася (для деталей див. додатки).

Після цього ми проаналізували глобальний та локальні сурогати за допомогою значень Шаплі (розрахованих за допомогою пакету SHAP та TreeExplainer, який реалізує швидкий і точний метод TreeSHAP), і отримали багато значень важливості ознак, що піддали аналізу.

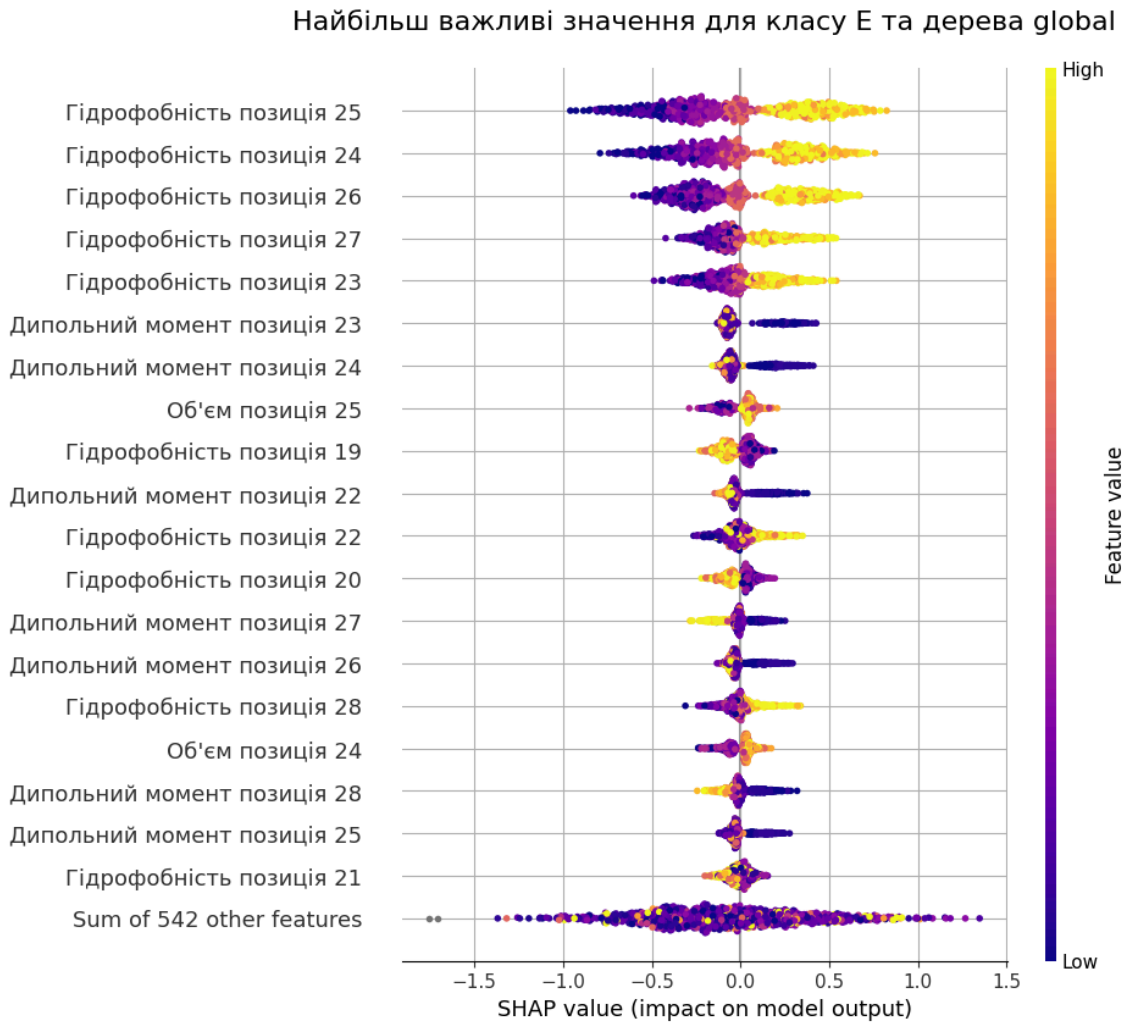


Рис. 3.6. Приклад Шаплі значень для 1000 випадкових семплів для глобального сурогату та класу бета-ланцюга

Загалом ці значення Шаплі підтверджують результати локальних моделей LIME: E має більший коефіцієнт гідрофобності, ніж H; E має від'ємний зв'язок з дипольним моментом, тоді як H - додатний, також зберігається передбачення для pKa1. Для pKa2 прогноз, здається, показує, що тільки одне значення є важливим. Ця величина з нормалізованим pKa2 близько 3 насправді є залишком проліну, і значення Шаплі тут залежить від положення, причому 26 (1 праворуч від точки відліку) є найбільшим, а інші - меншими. Це пояснює негативну кореляцію для pKa2 у LIME. Цей

величезний викид pKa2 відсутній в E, має зворотну картину в C, а також є дестабілізатором в G (спіралі іншого типу, ніж альфа спіраль, але зі схожою фізико-хімічною основою). Кореляція, яка була статистично незначимою в LIME, виявилась значимою тут - ароматичність ні разу не зустрічається серед 100 найбільш важливих характеристик в альфа-спіралі, тоді як для бета-ланцюгу всього 7 позицій є найбільш важливими, і залежність здебільшого тут позитивна.

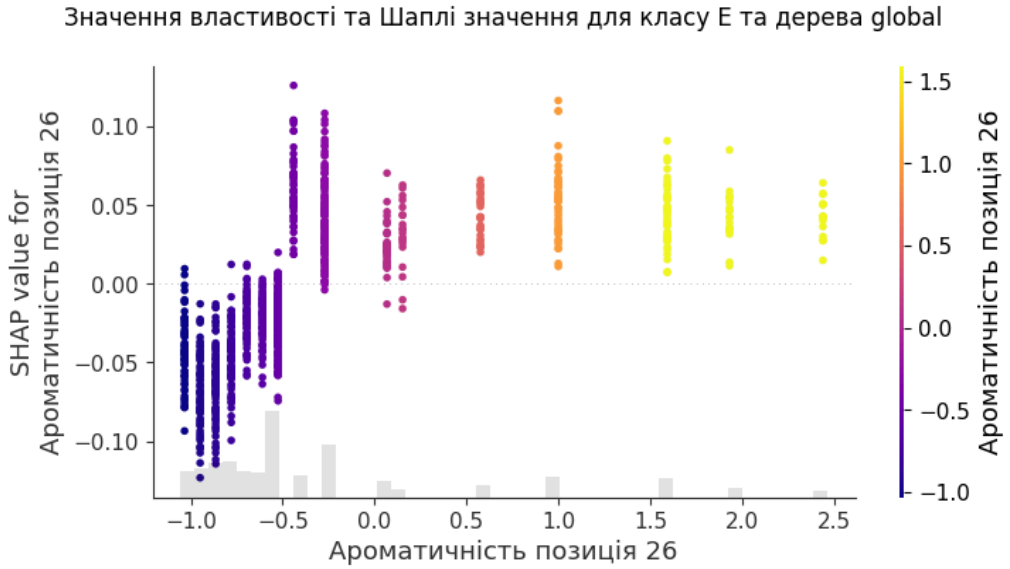


Рис. 3.7. Графіки розкиду для ароматичності та відповідних Шаплі значень для класу E (бета ланцюг)

Для виокремлення повороту від вигину описана різниця по дипольним моментам зберігається, тоді як їхня різниця по pKa стає більш заплутаною - різниця по pKa COOH зникає (щонайменше, напрямок залежності стає однаковим), тоді як для NH2 стає найбільш важливим вищезгаданий пролін (хоча загальний тренд все одно йде на зниження). Іншою цікавою характеристикою є інверсія залежності від гідрофобності для E, S та T (див рис. 3.8). Для E ця залежність представляє перехід від позитивної до негативної в 5 амінокислот від точки передбачення, тоді як для S та T залежність зворотня. Схоже на те, що таким чином модель

прив’язала властивість бета-ланцюгів утворювати бета листи через їх чергування за рахунок поворотів та вигинів, однак схожа залежність для альфа-спіралі відсутня - для неї залежність ніби поступово сходить до 0 (див рис. 3.8). Ця закономірність починається ще з узагальнених коефіцієнтів LIME (див. додатки) і спостерігається в подальшому в узагальнюючій моделі, що була створена для опису вторинної структури.

На основі отриманих закономірностей побудували загальну формулу для передбачення вторинної структури білка. Вона була представлена як агрегація значень властивостей з усіх частин білка, зважена на дистанцію з точки передбачення, після якої система видає ймовірність передбачення:

$$A_{nat} = concat\,(W_{eos}, X_{onehot}E_{nat}, W_{bos}), \tag{3.3}$$

$$D_{dif} = (X_{pos} - X^{T}_{pos}), \tag{3.4}$$

$$D_{w} = cos\,(W_{period} \cdot D_{dif} + W_{phase}) \cdot e^{-\left(\frac{D_{dif}}{W_{var}}\right)^{2}} + W_{bias}, \tag{3.5}$$

$$P\,(C \mid X) = softmax\,(W_{class} \cdot D_{w} \cdot A_{nat}). \tag{3.6}$$

Усі вектори W є тренувальними параметрами (W_{eos} та W_{bos} вказують на N та C кінці білкової послідовності, W_{period} та W_{phase} - період та фаза косинусної частини, W_{bias} - вектор відхилень, W_{class} - вектор, що пов’язує властивість та клас вторинної структури), X - вхідними (X_{onehot} - позначення амінокислот як векторів значень 1-0, X_{pos} - вектор позначень позицій амінокислот, представляє з себе діапазон від 0 до довжини білка), E_{nat} позначає природні вставки, і, нарешті, $P\,(C \mid X)$ - фінальну ймовірність кожного класу за умови знання амінокислотної послідовності. Гаусова функція моделює віддалення від точки передбачення, а косинус дозволяє моделювати ситуацію для E, S та T з інверсією залежності від гідрофобності. Нарешті, конкатенація до токену початку послідовності та

кінця дозволяє моделі приділяти увагу до кінців білка (важливо для передбачення невпорядкованої структури). Ми отримали наступну точність, див. табл. 3.3.

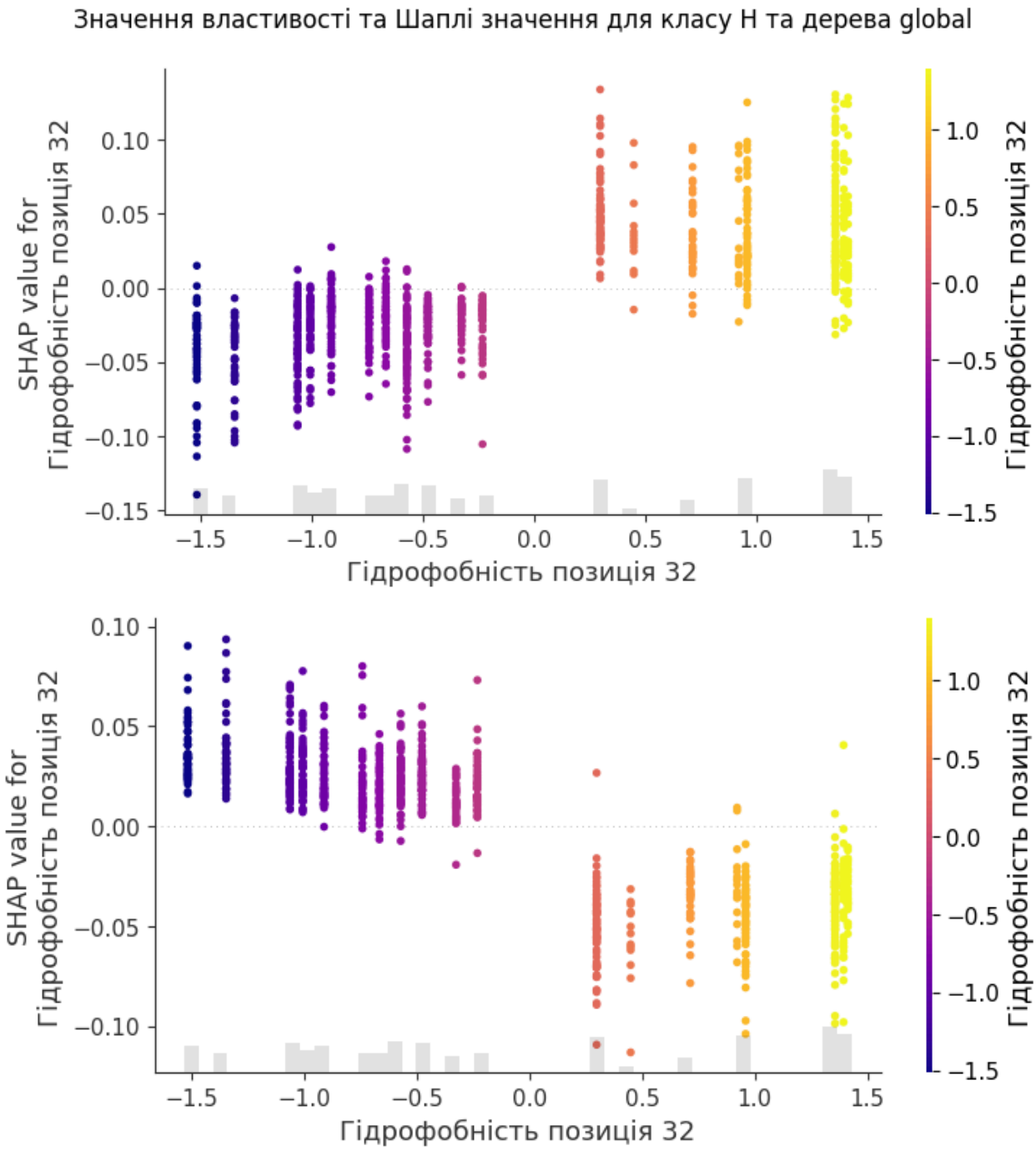


Рис. 3.8. Типовий розмір альфа-спіралі - 12 амінокислотних залишків, тоді як бета-ланцюга - 8. Чому ця інверсія спостерігається для одного класу, але не для іншого - незрозуміло

Табл 3.3. Точність формульного сурогату на тестувальному датасеті

	Влучність (precision)	**Повнота (recall)**	**F1-скор (F1-score)**	**Кількість прикладів**
H	0.49	0.68	0.57	20991
E	0.42	0.52	0.46	14008
C	0.34	0.35	0.35	12782
T	0.26	0.16	0.19	7130
G	0.15	0.00	0.01	2450
S	0.20	0.02	0.03	5364
B	0.00	0.00	0.00	755
I	0.00	0.00	0.00	5
X	0.44	0.28	0.34	3802
Точність	0.43	0.43	0.43	--
Макро точність	0.24	0.22	0.22	67287
Зважена точність	0.38	0.43	0.39	67287

У процесі підбору адекватної структури для моделі були певні роздуми стосовно необхідності тих чи інших елементів моделі. Тому провели також тренування моделей без певних компонентів та вивчили, які структури втрачають скільки точності від цих модифікацій. Відсутність W_{eos} та W_{bos}, що вказують на кінці білка, знижує точність лише X класу та на 3-6%, W_{period}, W_{phase} та відсутність косинусу - погіршує точність E, S, T та C (на 3-5%, 6-8%, 1-2% та 2-3% відповідно), W_{bias} - погіршення у влучності передбачення H (2-3%) та X (11-13%), а також у повноті для E (6-7%), нарешті оскільки базова ідея - певне усереднення чи агрегація даних з усіх амінокислот, вирішили спробувати альтернативні стратегії, як відсутність усереднення (загальне падіння точності на 17-18%) та

рівномірне усереднення (загальне падіння точності на 11-12%). У матриці W_{class} ми можемо спостерігати повторення багатьох закономірностей, пов'язаних з формуванням вторинної структури, які ми відзначали у попередніх моделях (див. додатки).

3.3. Дистанції між амінокислотами та їхні властивості

Третинна структура є найскладнішою проблемою, пов’язаною зі структурою білка, і вона має декілька варіантів представлень. Серед них є координати атомів, z-матриці, дистограми (квадратні матриці дистанцій). матриці кутів, і це ще не враховуючи представлення бічних ланцюгів амінокислот. Ми вирішили поки зосередитись на вивченні дистограм як більш простому та прямолінійному завданні. Оскільки AminoBERT не видає дистограми (бо модель була натренована на передбаченні замаскованої амінокислоти), ми вирішили адаптувати модель на цьому етапі до видачі дистограм. Було побудовано модель, що складається з AminoBERT модуля, та білінійного шару, як описано в попередньому розділі (в подальшому будемо називати цю модель AminoBERT+BiLinear). Ми отримали кореляцію 0,66-0,70 до реальних матриць дистанцій, що показує наявність структурної інформації на цьому етапі, хоча й доволі грубої - середня абсолютна похибка виявилась на рівні 8-12 А (див табл. 3.4. на наступній сторінці).

Тепер ми використовуємо той самий підхід, що й для прогнозування вторинної структури, щоб проаналізувати цю модель для прогнозування відстаней. Ми використовуємо пошук лінійного пояснення на основі методу LIME та Ridge регресії з вікнами певного розміру, який створює патчі, що вибираються з кожної матриці прогнозованих відстаней. Мотивація для використання саме цього типу лінійної моделі була описана в попередньому розділі

Табл 3.4. Точність AminoBERT+BiLinear на тренувальному, валідації та тестувальному датасеті

Тип датасету	Середня абсолютна похибка (MAE)	Кореляція (коефіцієнт Пірсона)
Тренувальний	8 A	0,70
Валідаційний	12 A	0,68
Тестувальний	12 A	0,66

Оскільки цей підхід вимагає руху в двох напрямках, позиційні відстані в одній ділянці можуть бути досить великими (ділянка у верхній правій частині матриці білкових відстаней має амінокислоти, що знаходяться на двох різних кінцях білкового ланцюга). Це вимагає від нас кодування позицій у природному вкладенні кожної амінокислоти, після чого ми обчислюємо амінокислотні відмінності, перетворюємо партії патчів на партії векторів і просимо модель передбачити відстань для центрального квадрата патча (тобто для патча розміром 9x9 нам потрібно передбачити відстань у квадраті 4-4 патча). Звичайно, різні розміри патчів дають різні результат, але найбільший приріст дав не певний великий розмір патчу, а додавання позиційних кодів. Без нього, використовуючи просто патч 1x1, ми отримали $R^2 = 0{,}006$, тоді як з ним - 0,236. Після цього ми спроектували ознаки, які мають бути найбільш корисними для прогнозування відстані, але при цьому забезпечувати пояснюваність (як це було описано в попередньому розділі).

Побачили, що загалом результати та коефіцієнти є досить мінливими, проте простежуються певні закономірності. Найпомітніша з них пов'язана з позиційною інформацією, де відносна та абсолютна позиційна інформація кодується значеннями, які створюють діагональні

патерни. Ці коефіцієнти також є найбільшими за величиною (десятки-сотні), що в Ridge регресії означає, що вони є не лише найважливішими, але й найнезалежнішими ознаками (для характеристик, що мають колінеарність з іншими, коефіцієнт спрямовується до 0). Дані про відносне положення мають послідовну картину позитивних коефіцієнтів на головній діагоналі, і чим далі від неї, тим більш від'ємними є коефіцієнти. Щодо інших коефіцієнтів, то їх значення є більш дискусійним. Великі абсолютні значення коефіцієнтів для маси зазвичай зменшуються з відстанню, проте деякі моделі LIME демонструють трошки інакшу залежність: чим ближче до квадрата 4-4, тим коефіцієнт наближається до нуля або навіть стає додатним.

Кореляції абсолютних значень поверхні та об'єму досить суперечливі, а інша закономірність з'являється лише для абсолютних значень гідрофобності, яка здебільшого має тенденцію до зменшення відстані для гідрофобних пар залишків (ймовірно, внаслідок того, що гідрофобний ефект є основною силою, що сприяє згортанню), а винятки, коли коефіцієнт тут додатний, досить рідкісні. Тенденції для абсолютних значень частоти, pI, pKa COOH також суперечливі, з деяким зсувом у бік позитивної або змішаної залежності від положення для p(aa), негативної та змішаної для pI і позитивної або змішаної для pKa COOH. Що стосується абсолютних значень дипольного моменту, pKa NH2 та ароматичності, то вони змінюються від білка до білка, що свідчить про відсутність чіткої закономірності між ними. Нарешті, дві останні закономірності щодо абсолютних значень випливають з аналізу EIIP та позицій. EIIP зберігає позитивні значення коефіцієнтів, тоді як абсолютні значення позицій демонструють діагональну закономірність, яка іноді може змінюватися між білками (верхній правий кут є найбільшим позитивним за величиною, а нижній лівий - найбільшим негативним за величиною).

Що стосується відносних величин, то тут інформація є скупою, оскільки багато коефіцієнтів не досягають необхідної статистичної значущості, однак, що стосується найбільш інформативних, то залишаються лише маса (від'ємна - велика схожість призводить до малої відстані), гідрофобність (також від'ємна), частота амінокислот (від'ємна), pKa COOH (від'ємна), положення (діагональ). Найбільш ймовірно, що ці залежності пов’язані з вторинною структурою, оскільки схожість цих характеристик в попередньому розділі визначає впорядкованість проти невпорядкованості. Мало інформації, але ймовірні взаємодії: pI (переважно позитивний - велика подібність призводить до більшої відстані), дипольний момент (негативний) і pKa NH2 (діагональ - верхній лівий нижній правий негативний, чим далі від неї, тим більш позитивними є коефіцієнти).

Звичайно, тут наявність іншої сурогатної моделі для підтвердження наших висновків була б ще більш важливою для успіху, оскільки завдання є складнішим, і недостатня статистична потужність стає ще більш серйозною проблемою (оскільки навіть при невеликому вікні 9x9 кількість ознак становить 81*24 проти 51*11 для вторинної структури). У результаті ми побудували 7 локальних дерев XGBoost і 1 глобальне дерево XGBoost, щоб проаналізувати, які ознаки роблять найбільший внесок у передбачення відстані між амінокислотами. Точність глобальної та 2 найточніших LIME моделей на тестовому наборі становить 8,100, 10,270 та 11,123 A (MAE для глобальної, LIME для моделі 8 та LIME для моделі 37). Положення, гідрофобність та дипольний момент є трьома основними компонентами прогнозування відстані. Серед них гідрофобність і дипольний момент використовуються як абсолютні величини, тоді як більша частина вхідних даних положення, важливих для прогнозування відстаней, є відносними.

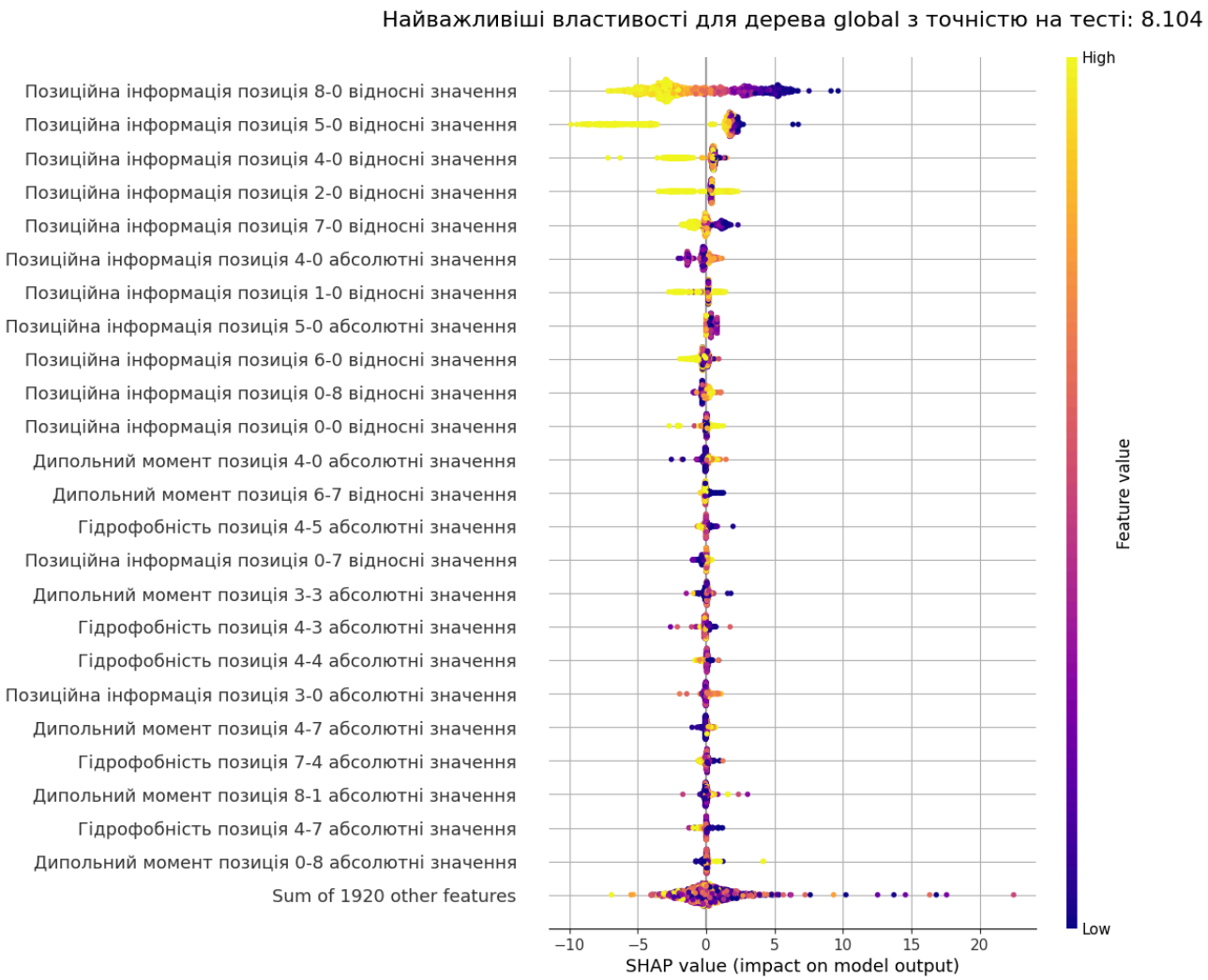


Рис. 3.9. Приклад Шаплі значень для 1000 випадкових семплів для глобального сурогату передбачення дистограм

Ми почнемо з розгляду абсолютних значень для властивостей. Невідповідності в масі зустрічаються і тут, хоча більшість значень Шаплі для глобального дерева для маси дорівнюють 0. Подібні значення Шаплі також є в дереві для моделі 8, хоча зовсім інші в дереві 37. Це показує, наскільки залежать від білка коефіцієнти для маси. Гідрофобність демонструє очікувану тенденцію до зниження, як в глобальній моделі, так і в локальних моделях для білків 8 і 37.

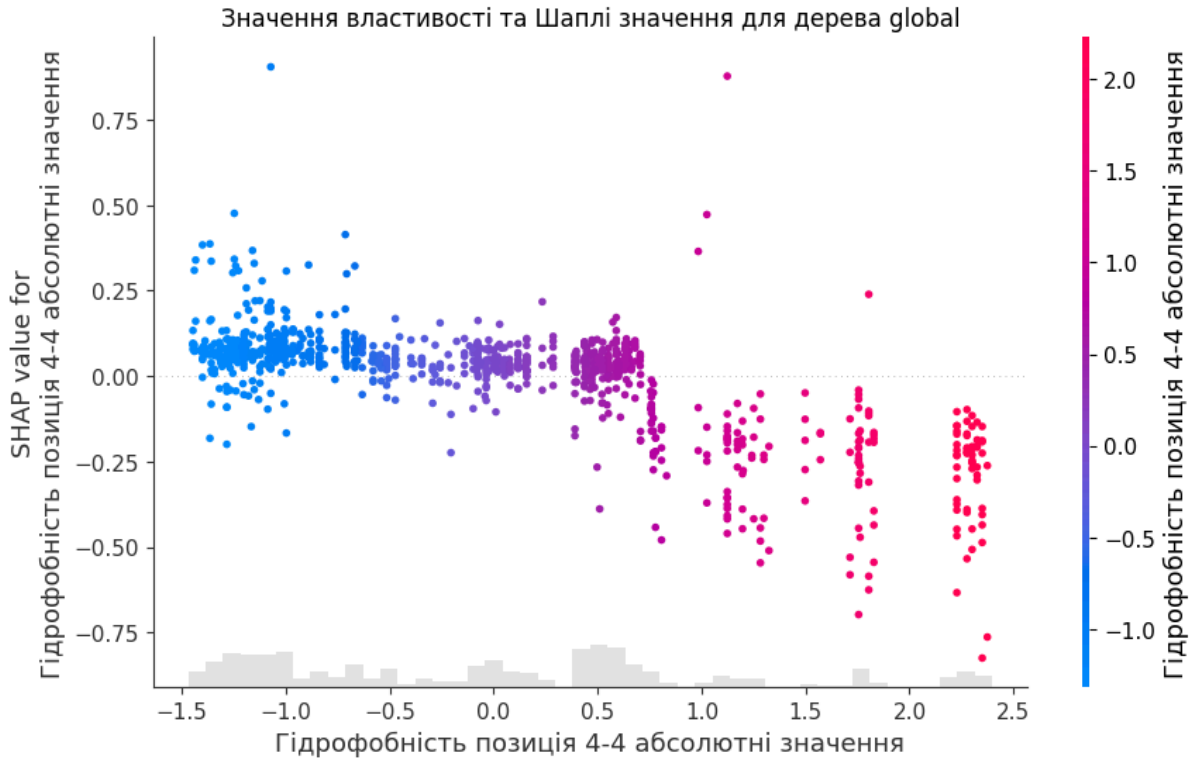


Рис. 3.10. Графіки розкиду для властивості гідрофобність та позиції у вікні 4-4

Коефіцієнти для частоти тут також негативні. Значення pI, однак, мають позитивний тренд у глобальному дереві, у дереві для послідовності 8 та 37, а ми робили висновок про загальний негативний тренд pI у лінійних моделях Ridge. Значення pKa COOH має негативну залежність від відстані, як було зазначено раніше. Ми також виявили деякі дуже чіткі зв'язки з дипольним моментом, які раніше не спостерігалися в моделях LIME Ridge - ці коефіцієнти тут є позитивними. Для аромафільності та EIIP спостерігається лише невеликий негативний тренд та невеликий позитивний тренд відповідно. Нарешті, для позиційної інформації характерна дуже чітка лінійна залежність, однак лише в основній позиції - 8-0, тоді як в інших спостерігаємо доволі незвичний графік, хоча і зі збереженням негативного тренду.

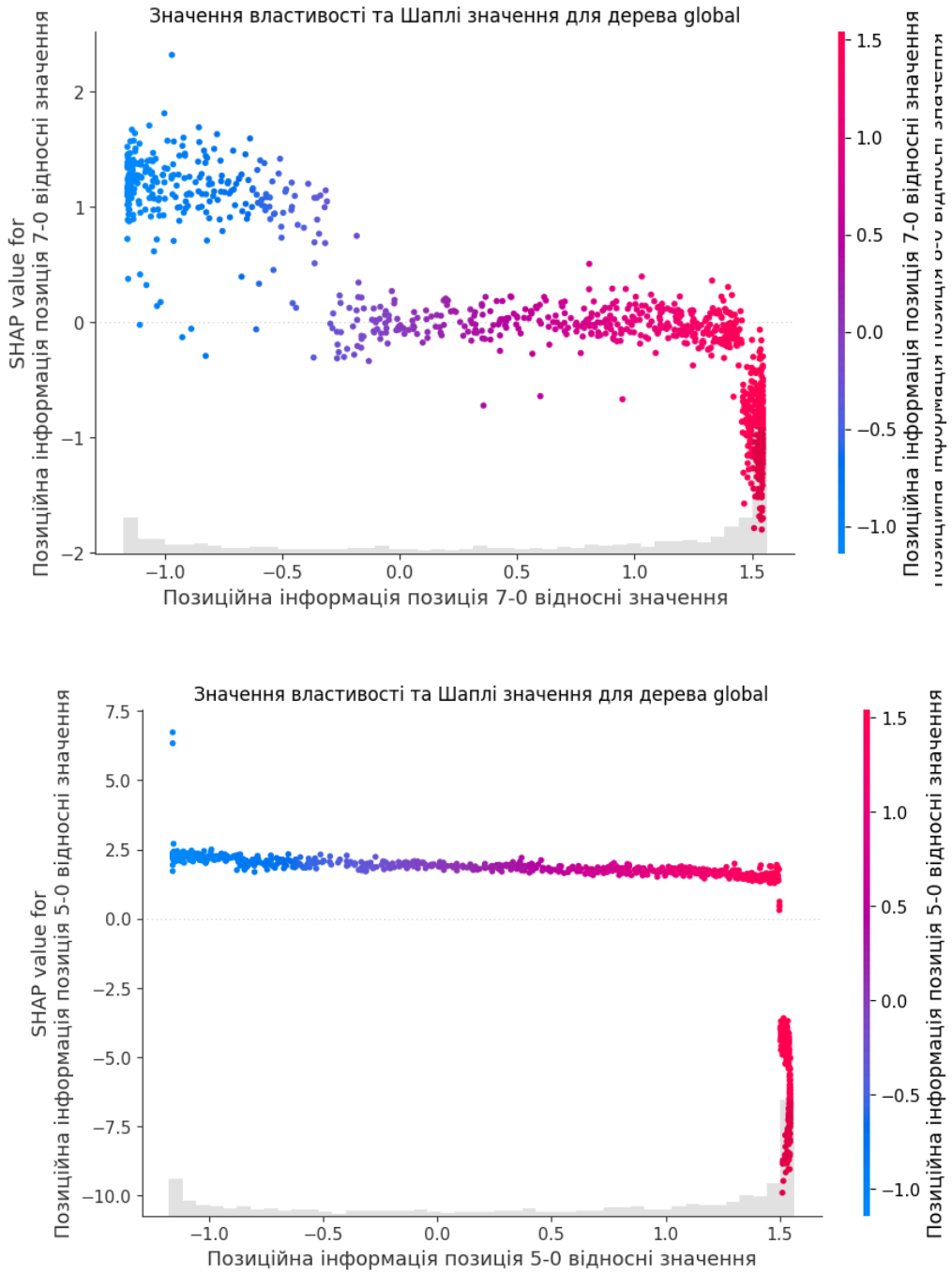


Рис. 3.11. Графік розкиду для відносної позиційної інформації при позиції у вікні 8-0 та 5-0 та відповідних Шаплі-значень. Для 8-0 помітна дуже чітка лінійна залежність, тоді як для 5-0 ця залежність починає розмиватись

Серед інших властивостей, особливо відносних, схожість дипольного моменту (негативний коефіцієнт для різниці - велика схожість зменшує відстань) є запорукою розташування поблизу амінокислот з впорядкованих ділянок. Аналогічна логіка - для гідрофобності, що теж має негативну залежність. Однак маса та pI є доволі дивною ситуацією, оскільки вони мають позитивну та негативну залежність відповідно, хоча в теорії схожість у масі означає належність до однієї вторинної структури, а велика різниця в pI має притягувати кислотні та лужні амінокислоти один до одного. Описані вище закономірності (як абсолютні, так і відносні) також зустрічаються в інших деревах - 11, 13, 18, 28 та 29, а тому можна підозрювати, що вони є в певній мірі універсальними (з поправкою на певну рідкість відносних властивостей в усіх деревах, які тому можуть виявитись випадковими кореляціями).

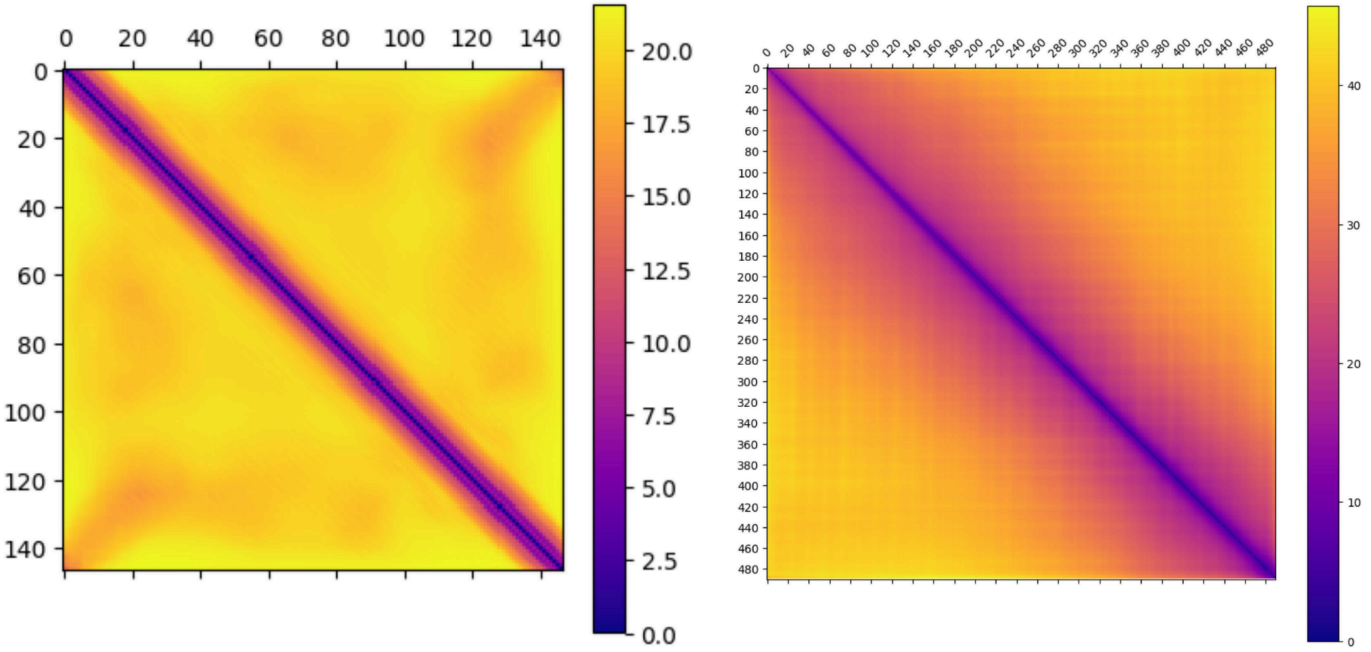


Рис. 3.12. Шаплі базові значення (середні по збуренням) для ESMFold та AminoBERT+Bilinear. Наша модель цю залежність не представляє, тоді як для ESMFold ця залежність характерна в багатьох білках

Цих доволі загальних даних було недостатньо, а тому вирішили провести аналіз іншої мережі третинної структури, що видає дистграми, аби порівняти з нашою досліджуваною. Робота в цьому напрямку ще проводиться, однак серед цікавих знахідок було виявлено трохи незвичну залежність від позиції, див. рис. 3.16.

Можна побачити, що в ESMFold дистанція між двома кінцями білка менша за середню очікувану, і коли ми продовжуємо рухатись по цій діагоналі до 20 з обох кінців, ми спостерігаємо цікавий квадрат, що показує близькість цієї ділянки до усіх інших регіонів білка. Ця ситуація була вже описана в статті 1996 року [71], де вказується на близькість не так кінців, як регіонів. І справді, якщо провести детальний аналіз цих дистограм, здебільшого ці особливі точки близькості N та C кінців - це регіони розміром близько 20 амінокислот, з розподілом, що нагадує нормальний. Схожа характеристика і для квадрату, а тому ми побудували апроксимацію цих базових Шаплі значень, яку в подальшому плануємо застосовувати для побудови глобального сурогату третинної структури на основі дистиляції AminoBERT (див. додатки). Склад цієї апроксимації: базова дистанція без урахування крайових ефектів ($\sqrt{3,8 \cdot x_{pos}}$), де 3,8 є підібраним параметром, компонент взаємодії кінців та квадрату представляє гаусову функцію з параметрами розкиду (σ) 11,33, середнім (μ) 20 та $L_{seq} - 20$, амплітудою (A) -3 (для квадрата) чи -5 (для кінців): $Ae^{-(\frac{x-\mu}{\sigma})^2}$.

ВИСНОВКИ

У результаті аналізу нейромережі AminoBERT було встановлено частину основних залежностей, необхідних для передбачення вторинної та третинної структури із первинної. Для отримання та перевірки цих закономірностей було отримано локальні (12 вторинної структури та 14 третинної структури) та глобальні (2 вторинної та 1 третинної) сурогатні моделі.

1. При аналізі векторів-вставок було встановлено основні фізико-хімічні закономірності, що розміщують амінокислоти у векторному просторі вставок - маса, гідрофобність, дипольний момент та частота. Ці властивості далі залишаються важливими для розрізнення вторинних структур та третинних.
2. Для вторинної структури характерна залежність від позиції в послідовності, що нагадує гаусову криву. У подальшому вона застосована як значення важливості амінокислотних властивостей залежно від їхньої позиції. Серед особливостей, знайдених на цьому етапі, є інверсія залежності від гідрофобності для бета ланцюга (моделюється за допомогою періодичної функції - косинусом), менша залежність альфа спіралі від гідрофобності, ніж для бета ланцюга, і навпаки для дипольного моменту. Нарешті, певні геометричні/структурні дескриптори відрізняються між бета ланцюгом та альфа спіраллю - для альфа спіралі характерна залежність від поверхні, тоді як для бета ланцюга - від об’єму. Серед невпорядкованих найкраще класифікуються: випадковий клубок, невпорядкований регіон та повороти, де різниця між вигинами та поворотами пов’язана різними залежностями від pKa біля точки передбачення та дипольного моменту.

3. Для третинної структури однією із найважливіших характеристик при передбаченні матриць дистанцій є позиційна відстань з підвищенням R^2 більше ніж на 20%, а іншими важливими характеристиками є гідрофобність, дипольний момент та частота амінокислот у послідовностях. Було підтверджено зменшення відстані між термінальними регіонами білка, що до цього постулювалось вченими і що, однак, не закладено у нашій моделі, а тому є можливим проспектом для її покращення.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Ribeiro, M.T., Singh, S. and Guestrin, C. (2016) "'Why Should I Trust You?': Explaining the Predictions of Any Classifier." arXiv. doi:10.48550/ARXIV.1602.04938
2. Ribeiro, M.T., Singh, S. and Guestrin, C. (2018) "Anchors: High-Precision Model-Agnostic Explanations," Proceedings of the AAAI Conference on Artificial Intelligence. Association for the Advancement of Artificial Intelligence (AAAI). doi:10.1609/aaai.v32i1.11491.
3. Lundberg, S. and Lee, S.-I. (2017) "A Unified Approach to Interpreting Model Predictions." arXiv. doi:10.48550/ARXIV.1705.07874.
4. Dill, K.A. and MacCallum, J.L. (2012) "The Protein-Folding Problem, 50 Years On," Science. American Association for the Advancement of Science (AAAS). doi:10.1126/science.1219021.
5. Nassar, R. et al. (2021) "The Protein Folding Problem: The Role of Theory," Journal of Molecular Biology. Elsevier BV. doi:10.1016/j.jmb.2021.167126.
6. Kuhlman, B. and Bradley, P. (2019) "Advances in protein structure prediction and design," Nature Reviews Molecular Cell Biology. Springer Science and Business Media LLC. doi:10.1038/s41580-019-0163-x.
7. Kmiecik, S. et al. (2016) "Coarse-Grained Protein Models and Their Applications," Chemical Reviews. American Chemical Society (ACS). doi:10.1021/acs.chemrev.6b00163.
8. Fiser, A. (2010) "Template-Based Protein Structure Modeling," Methods in Molecular Biology. Humana Press. doi:10.1007/978-1-60761-842-3_6.
9. Altschul, S.F. et al. (1990) "Basic local alignment search tool," Journal of Molecular Biology. Elsevier BV. doi:10.1016/s0022-2836(05)80360-2
10. Fernandez-Fuentes, N. et al. (2007) "Comparative protein structure modeling by combining multiple templates and optimizing

sequence-to-structure alignments," Bioinformatics. Oxford University Press (OUP). doi:10.1093/bioinformatics/btm377.

11. Rai, B.K. et al. (2006) "MMM: a sequence-to-structure alignment protocol," Bioinformatics. Oxford University Press (OUP). doi:10.1093/bioinformatics/btl449.
12. Sutcliffe, M.J. et al. (1987) "Knowledge based modelling of homologous proteins, part I: three-dimensional frameworks derived from the simultaneous superposition of multiple structures," "Protein Engineering, Design and Selection." Oxford University Press (OUP). doi:10.1093/protein/1.5.377.
13. Fiser, A. and Šali, A. (2003) "Modeller: Generation and Refinement of Homology-Based Protein Structure Models," Methods in Enzymology. Elsevier. doi:10.1016/s0076-6879(03)74020-8.
14.. Contreras-Moreira, B. et al. (2003) "Novel use of a genetic algorithm for protein structure prediction: Searching template and sequence alignment space," Proteins: Structure, Function, and Genetics. Wiley. doi:10.1002/prot.10549.
15. Laskowski, R.A. et al. (1993) "PROCHECK: a program to check the stereochemical quality of protein structures," Journal of Applied Crystallography. International Union of Crystallography (IUCr). doi:10.1107/s0021889892009944.
16. Hooft, R.W.W. et al. (1996) "Errors in protein structures," Nature. Springer Science and Business Media LLC. doi:10.1038/381272a0.
17. Williams, C.J. et al. (2017) "MolProbity: More and better reference data for improved all‐atom structure validation," Protein Science. Wiley. doi:10.1002/pro.3330.
18. Wiederstein, M. and Sippl, M.J. (2007) "ProSA-web: interactive web service for the recognition of errors in three-dimensional structures of

proteins," Nucleic Acids Research. Oxford University Press (OUP). doi:10.1093/nar/gkm290.

19. Eisenberg, D., Lüthy, R. and Bowie, J.U. (1997) "[20] VERIFY3D: Assessment of protein models with three-dimensional profiles," Methods in Enzymology. Elsevier. doi:10.1016/s0076-6879(97)77022-8.
20. Zhang, Y. (2007) "Template-based modeling and free modeling by I-TASSER in CASP7," Proteins: Structure, Function, and Bioinformatics. Wiley. doi:10.1002/prot.21702.
21. Lee, J., Freddolino, P.L. and Zhang, Y. (2017) "Ab Initio Protein Structure Prediction," From Protein Structure to Function with Bioinformatics. Springer Netherlands. doi:10.1007/978-94-024-1069-3_1.
22. Folding@home (2016) Folding@home. Available at: https://foldingathome.org/.
23. Ensign, D.L., Kasson, P.M. and Pande, V.S. (2007) "Heterogeneity Even at the Speed Limit of Folding: Large-scale Molecular Dynamics Study of a Fast-folding Variant of the Villin Headpiece," Journal of Molecular Biology. Elsevier BV. doi:10.1016/j.jmb.2007.09.069.
24. Simons, K.T. et al. (1999) "Ab initio protein structure prediction of CASP III targets using ROSETTA," Proteins: Structure, Function, and Genetics. Wiley. doi:10.1002/(sici)1097-0134(1999)37:3+<171::aid-prot21>3.0.co;2-z.
25. Klepeis, J.L. and Floudas, C.A. (2003) "ASTRO-FOLD: A Combinatorial and Global Optimization Framework for Ab Initio Prediction of Three-Dimensional Structures of Proteins from the Amino Acid Sequence," Biophysical Journal. Elsevier BV. doi:10.1016/s0006-3495(03)74640-2.
26. Klepeis, J.L. et al. (2004) "Ab initio prediction of the three‐dimensional structure of a de novo designed protein: A double‐blind case study,"

Proteins: Structure, Function, and Bioinformatics. Wiley. doi:10.1002/prot.20338.

27. Yadav, N., Yadav, A., & Kumar, M. (2015). History of Neural Networks. SpringerBriefs in Applied Sciences and Technology, 13–15. doi:10.1007/978-94-017-9816-7_2
28. Macukow, B. (2016). Neural Networks – State of Art, Brief History, Basic Models and Architecture. Lecture Notes in Computer Science, 3–14. doi:10.1007/978-3-319-45378-1_1
29. Lederer, J. (2021) “Activation Functions in Artificial Neural Networks: A Systematic Overview.” arXiv. doi:10.48550/ARXIV.2101.09957.
30. Szandała, T. (2020) “Review and Comparison of Commonly Used Activation Functions for Deep Neural Networks,” arXiv [Preprint]. doi:10.48550/ARXIV.2010.09458.
31. Santanu Pattanayak (2017) Pro deep learning with TensorFlow : a mathematical approach to advanced artificial intelligence in Python. Berkeley, Calif.: Apress, New York, Ny.
32. Vaswani, A. et al. (2017) “Attention Is All You Need.” arXiv. doi:10.48550/ARXIV.1706.03762.
33. Snyders, S. and Omlin, C.W. (2000) “What inductive bias gives good neural network training performance?,” Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks. IJCNN 2000. Neural Computing: New Challenges and Perspectives for the New Millennium. Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks. IJCNN 2000. Neural Computing: New Challenges and Perspectives for the New Millennium, IEEE. doi:10.1109/ijcnn.2000.861348.
34. Xu, J., McPartlon, M. and Li, J. (2021) “Improved protein structure prediction by deep learning irrespective of co-evolution information,”

Nature Machine Intelligence. Springer Science and Business Media LLC. doi:10.1038/s42256-021-00348-5.

35. Senior, A.W. et al. (2020) “Improved protein structure prediction using potentials from deep learning,” Nature. Springer Science and Business Media LLC. doi:10.1038/s41586-019-1923-7.
36. Yang, J. et al. (2020) “Improved protein structure prediction using predicted interresidue orientations,” Proceedings of the National Academy of Sciences. Proceedings of the National Academy of Sciences. doi:10.1073/pnas.1914677117.
37. Morcos, F. et al. (2011) “Direct-coupling analysis of residue coevolution captures native contacts across many protein families,” Proceedings of the National Academy of Sciences. Proceedings of the National Academy of Sciences. doi:10.1073/pnas.1111471108.
38. Hendrycks, D. and Gimpel, K. (2016) “Gaussian Error Linear Units (GELUs).” arXiv. doi:10.48550/ARXIV.1606.08415.
39. Jumper, J. et al. (2021) “Highly accurate protein structure prediction with AlphaFold,” Nature. Springer Science and Business Media LLC. doi:10.1038/s41586-021-03819-2.
40. Baek, M. et al. (2021) “Accurate prediction of protein structures and interactions using a three-track neural network,” Science. American Association for the Advancement of Science (AAAS). doi:10.1126/science.abj8754.
41. Lin, Z. et al. (2023) “Evolutionary-scale prediction of atomic-level protein structure with a language model,” Science. American Association for the Advancement of Science (AAAS). doi:10.1126/science.ade2574.
42. Chowdhury, R. et al. (2022) “Single-sequence protein structure prediction using a language model and deep learning,” Nature Biotechnology. Springer Science and Business Media LLC. doi:10.1038/s41587-022-01432-w.

43. Fuchs, F.B. et al. (2020) “SE(3)-Transformers: 3D Roto-Translation Equivariant Attention Networks.” arXiv. doi:10.48550/ARXIV.2006.10503.
44. Fuchs, F. (2020) AlphaFold 2 & Equivariance, fabianfuchsml.github.io. Available at: https://fabianfuchsml.github.io/alphafold2 (Accessed: 5 May 2024).
45. Islam, M.R. et al. (2022) “A Systematic Review of Explainable Artificial Intelligence in Terms of Different Application Domains and Tasks,” Applied Sciences. MDPI AG. doi:10.3390/app12031353.
46. Speith, T. (2022) “A Review of Taxonomies of Explainable Artificial Intelligence (XAI) Methods,” 2022 ACM Conference on Fairness, Accountability, and Transparency. FAccT ’22: 2022 ACM Conference on Fairness, Accountability, and Transparency, ACM. doi:10.1145/3531146.3534639.
47. Ivanovs, M., Kadikis, R. and Ozols, K. (2021) “Perturbation-based methods for explaining deep neural networks: A survey,” Pattern Recognition Letters. Elsevier BV. doi:10.1016/j.patrec.2021.06.030.
48. Vermeire, T. et al. (2021) “How to Choose an Explainability Method? Towards a Methodical Implementation of XAI in Practice,” Communications in Computer and Information Science. Springer International Publishing. doi:10.1007/978-3-030-93736-2_39.
49. Kakogeorgiou, I. and Karantzalos, K. (2021) “Evaluating explainable artificial intelligence methods for multi-label deep learning classification tasks in remote sensing,” International Journal of Applied Earth Observation and Geoinformation. Elsevier BV. doi:10.1016/j.jag.2021.102520.
50. Zeiler, M.D. and Fergus, R. (2013) “Visualizing and Understanding Convolutional Networks.” arXiv. doi:10.48550/ARXIV.1311.2901.

51. Kim, B. et al. (2017) “Interpretability Beyond Feature Attribution: Quantitative Testing with Concept Activation Vectors (TCAV),” arXiv [Preprint]. doi:10.48550/ARXIV.1711.11279.
52. Jain, S. and Wallace, B.C. (2019) “Attention is not Explanation.” arXiv. doi:10.48550/ARXIV.1902.10186.
53. Zini, J.E. and Awad, M. (2022) “On the Explainability of Natural Language Processing Deep Models,” ACM Computing Surveys. Association for Computing Machinery (ACM). doi:10.1145/3529755.
54. Selvaraju, R.R. et al. (2019) “Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization,” International Journal of Computer Vision. Springer Science and Business Media LLC. doi:10.1007/s11263-019-01228-7.
55. Arkhangelskaia, E. and Dutta, S. (2019) “Whatcha lookin’ at? DeepLIFTing BERT’s Attention in Question Answering.” arXiv. doi:10.48550/ARXIV.1910.06431.
56. Ancona, M. et al. (2017) “Towards better understanding of gradient-based attribution methods for Deep Neural Networks.” arXiv. doi:10.48550/ARXIV.1711.06104.
57. Ali, A. et al. (2022) “XAI for Transformers: Better Explanations through Conservative Propagation.” arXiv. doi:10.48550/ARXIV.2202.07304.
58. Tan, J. and Zhang, Y. (2023) “ExplainableFold: Understanding AlphaFold Prediction with Explainable AI.” arXiv. doi:10.48550/ARXIV.2301.11765.
59. Ahdritz, G. et al. (2022) “OpenFold: Retraining AlphaFold2 yields new insights into its learning mechanisms and capacity for generalization.” Cold Spring Harbor Laboratory. doi:10.1101/2022.11.20.517210.
60. Roney, J. P., & Ovchinnikov, S. (2022). State-of-the-Art Estimation of Protein Model Accuracy Using AlphaFold. In Physical Review Letters

(Vol. 129, Issue 23). American Physical Society (APS). https://doi.org/10.1103/physrevlett.129.238101

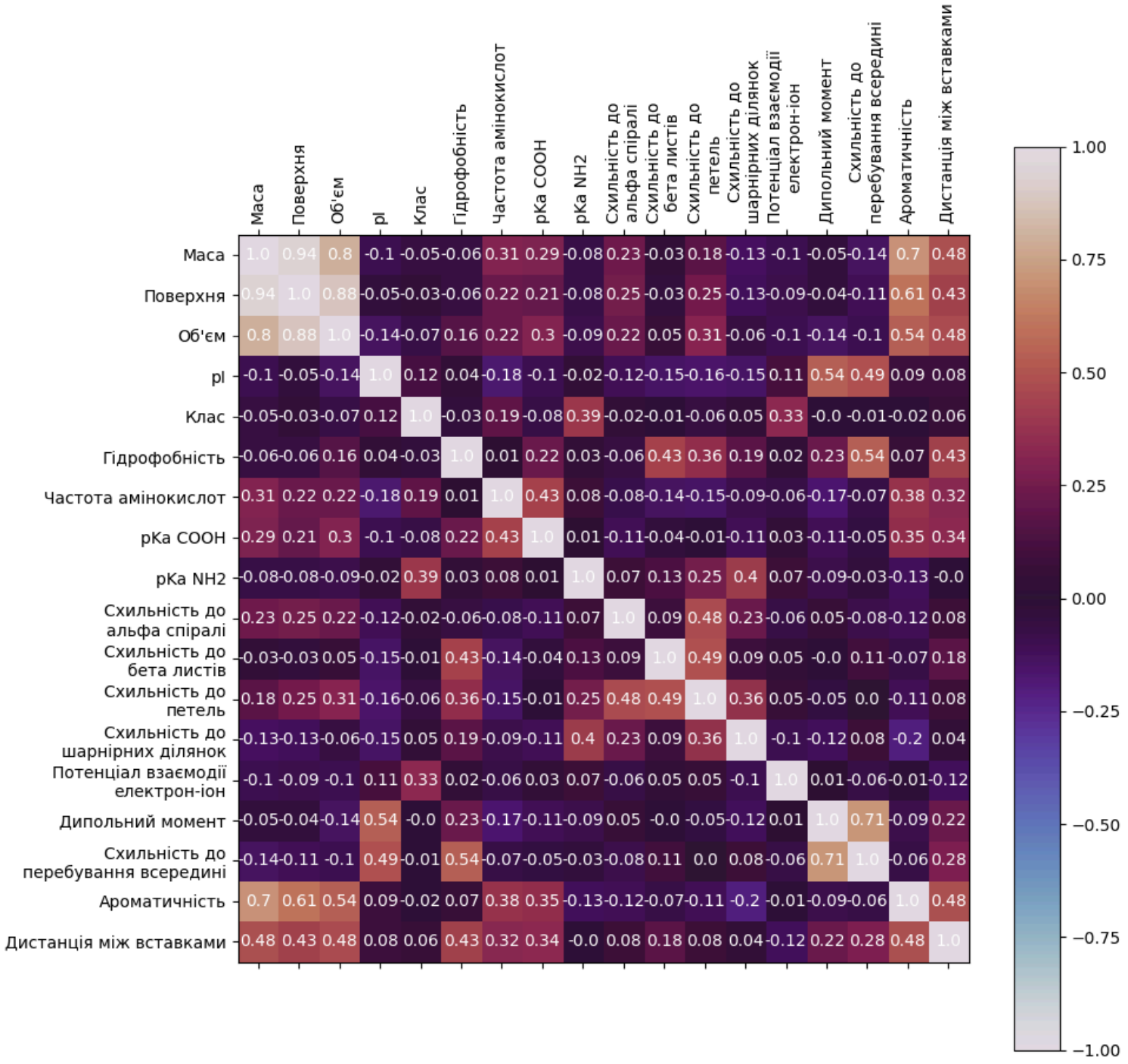
61. Elnaggar, A., Heinzinger, M., Dallago, C., Rehawi, G., Wang, Y., Jones, L., Gibbs, T., Feher, T., Angerer, C., Steinegger, M., Bhowmik, D., & Rost, B. (2022). ProtTrans: Toward Understanding the Language of Life Through Self-Supervised Learning. In IEEE Transactions on Pattern Analysis and Machine Intelligence (Vol. 44, Issue 10, pp. 7112–7127). Institute of Electrical and Electronics Engineers (IEEE). https://doi.org/10.1109/tpami.2021.3095381
62. Jha, S. K., Ramanathan, A., Ewetz, R., Velasquez, A., & Jha, S. (2021). Protein Folding Neural Networks Are Not Robust (Version 2). arXiv. https://doi.org/10.48550/ARXIV.2109.04460
63. Henikoff, S., & Henikoff, J. G. (1992). Amino acid substitution matrices from protein blocks. In Proceedings of the National Academy of Sciences (Vol. 89, Issue 22, pp. 10915–10919). Proceedings of the National Academy of Sciences. https://doi.org/10.1073/pnas.89.22.10915
64. Molnar, C. (2022) Interpretable Machine learning: a Guide for Making Black Box Models Explainable. Munich, Germany: Christoph Molnar. Available at: https://christophm.github.io/interpretable-ml-book/ (Accessed: 5 May 2024).
65. Braghetto, A., Orlandini, E., & Baiesi, M. (2023). Interpretable Machine Learning of Amino Acid Patterns in Proteins: A Statistical Ensemble Approach. In Journal of Chemical Theory and Computation (Vol. 19, Issue 17, pp. 6011–6022). American Chemical Society (ACS). https://doi.org/10.1021/acs.jctc.3c00383
66. Shapovalov, M., Dunbrack, R. L., & Vucetic, S. (2020). Multifaceted analysis of training and testing convolutional neural networks for protein secondary structure prediction. In B. Wallner (Ed.), PLOS ONE (Vol. 15,

Issue 5, p. e0232528). Public Library of Science (PLoS). https://doi.org/10.1371/journal.pone.0232528

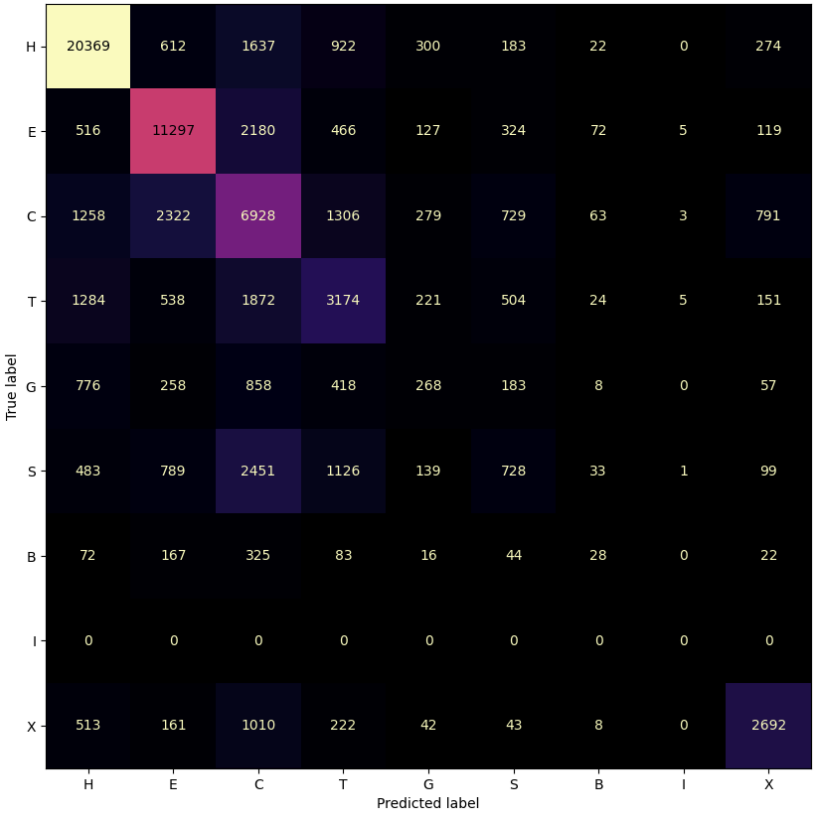
67. Kabsch, W., & Sander, C. (1983). Dictionary of protein secondary structure: Pattern recognition of hydrogen‐bonded and geometrical features. In Biopolymers (Vol. 22, Issue 12, pp. 2577–2637). Wiley. https://doi.org/10.1002/bip.360221211
68. Varadi, M., Anyango, S., Deshpande, M., Nair, S., Natassia, C., Yordanova, G., Yuan, D., Stroe, O., Wood, G., Laydon, A., Žídek, A., Green, T., Tunyasuvunakool, K., Petersen, S., Jumper, J., Clancy, E., Green, R., Vora, A., Lutfi, M., … Velankar, S. (2021). AlphaFold Protein Structure Database: massively expanding the structural coverage of protein-sequence space with high-accuracy models. In Nucleic Acids Research (Vol. 50, Issue D1, pp. D439–D444). Oxford University Press (OUP). https://doi.org/10.1093/nar/gkab1061
69. Dayhoff, M. and Schwartz, R. (1978) A Model of Evolutionary Change in Proteins. Silver Spring (MD): National Biomedical Research Foundation, pp. 345–352. Available at: https://chagall.med.cornell.edu/BioinfoCourse/PDFs/Lecture2/Dayhoff1978.pdf (Accessed: 5 May 2024).
70. Nocedal, J. and Wright, S.J. (2006) Numerical Optimization. New York: Springer.
71. Christopher, J. A., & Baldwin, T. O. (1996). Implications of N and C-Terminal Proximity for Protein Folding. In Journal of Molecular Biology (Vol. 257, Issue 1, pp. 175–187). Elsevier BV. https://doi.org/10.1006/jmbi.1996.0154

ДОДАТКИ

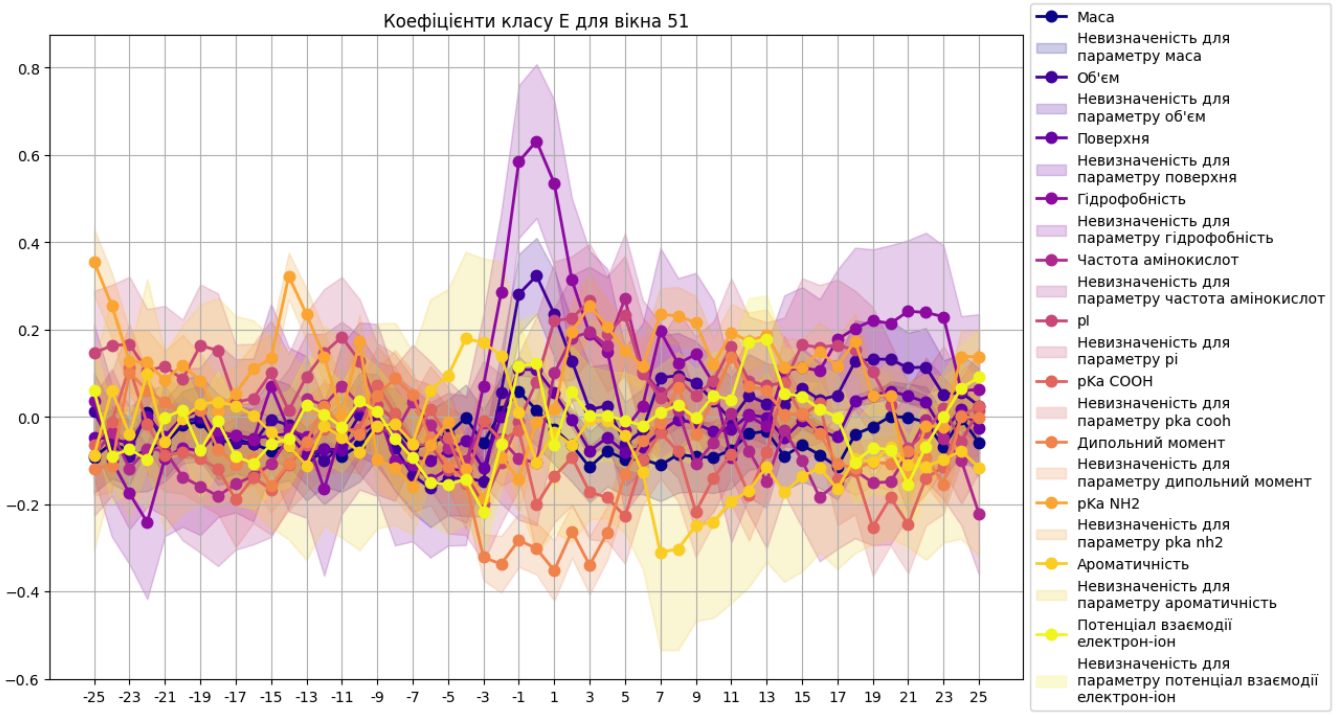
Додаток А. Матриця кореляцій між властивостями амінокислот



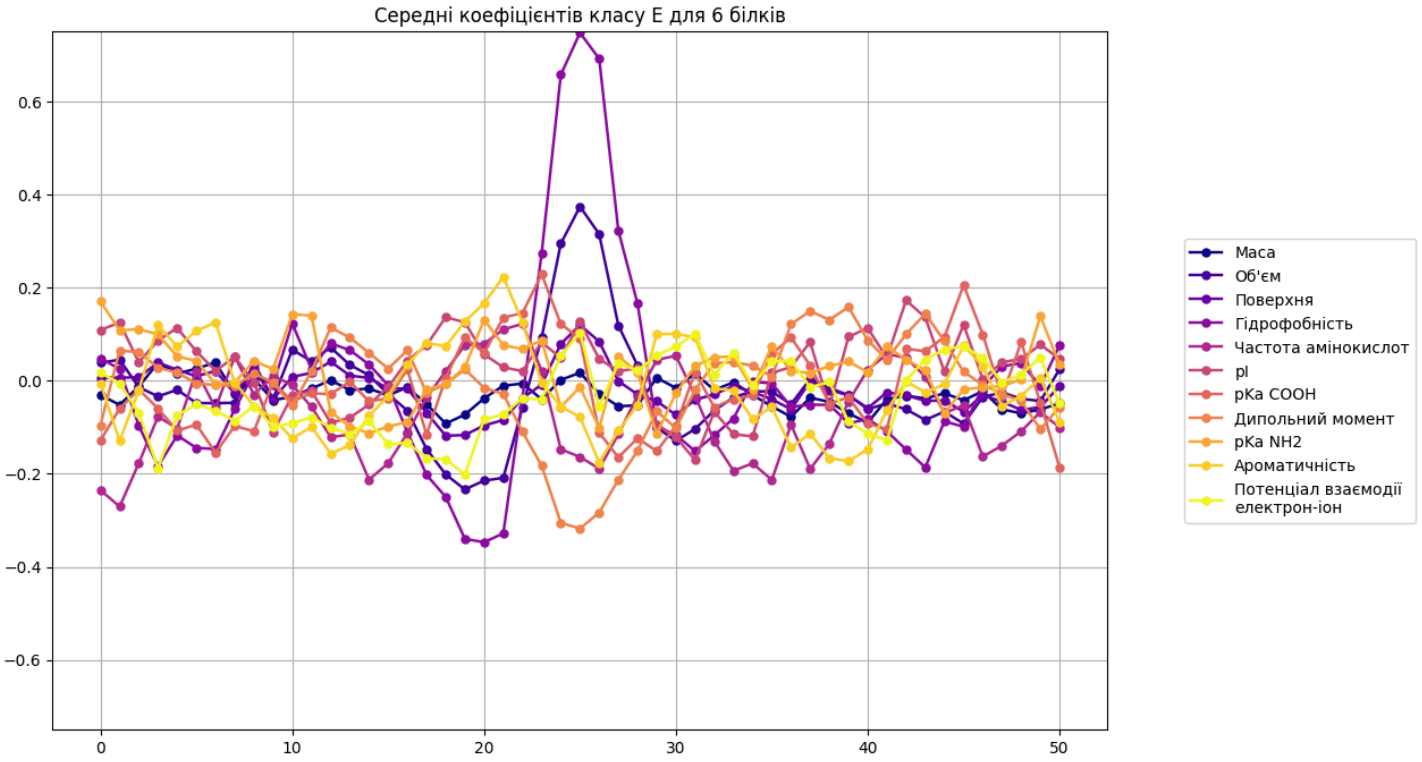
Додаток Б. Матриця плутанини для моделі AminoBERT + LDA для передбачення вторинної структури



Додаток В. Приклад представлення коефіцієнтів однієї з LIME моделей в їх повній красі - для всіх 51 позицій, 11 властивостей та з урахуванням статистичних інтервалів



Додаток Г. Приклад агрегованих коефіцієнтів для класу E з 6 білків. Агрегація відбувалась через зважження на розмір тренувального датасету (пропорційний довжині білка)

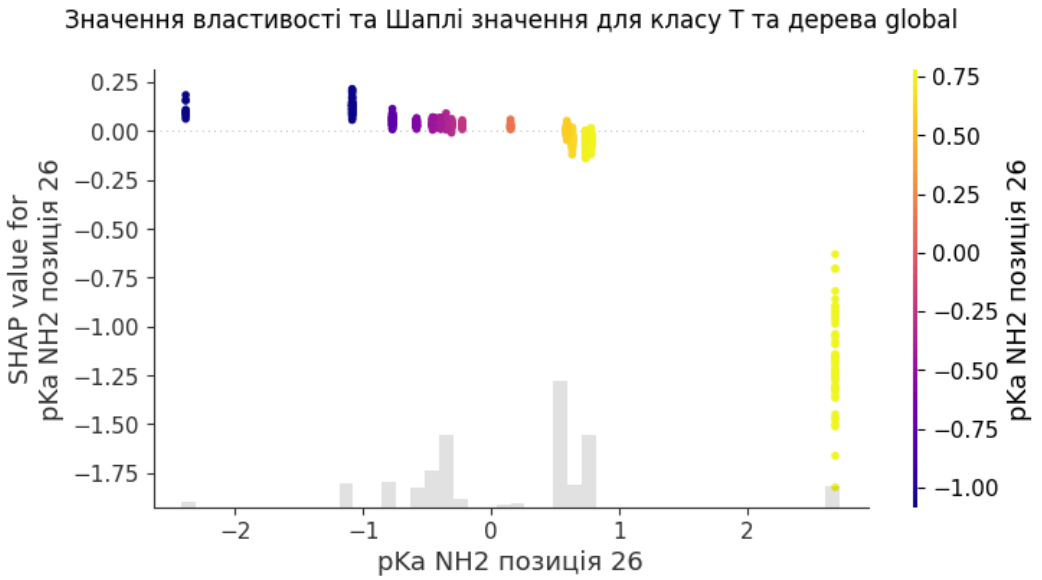


Таблиця. Приклад таблички з найважливішими властивостями для обраного дерева (білок 5XDZ ланцюг A по PDB ID)

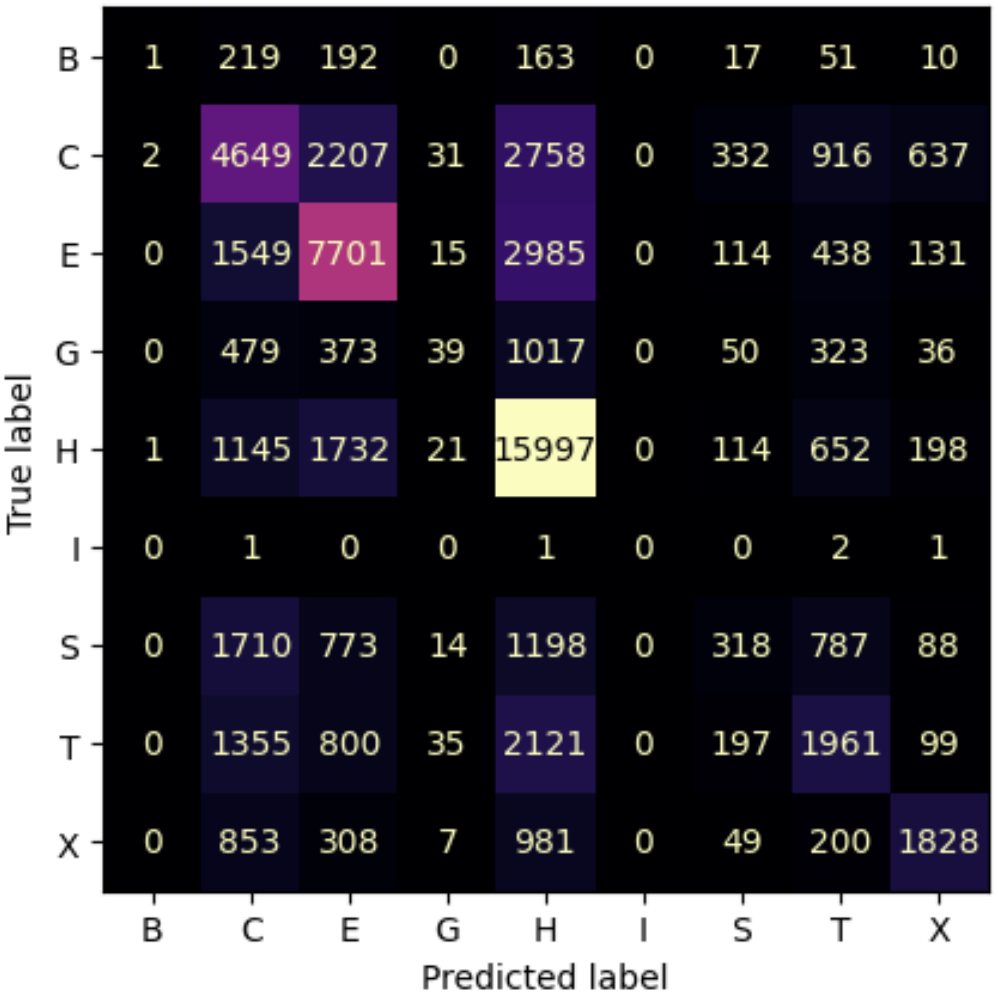
Порядок	**Властивість**	**Покриття**	**Виграш**
1	Гідрофобність позиція 25	0.2187	13734.21
2	Поверхня позиція 25	0.2187	10496.71
3	Маса позиція 26	0.2187	9710.09
4	Дипольний момент позиція 25	0.2187	8081.16
5	Маса позиція 25	0.2058	10711.58

6	Гідрофобність позиція 25	0.2413	5720.84
7	Поверхня позиція 26	0.2616	3459.86
8	Гідрофобність позиція 24	0.1626	9040.58
9	Гідрофобність позиція 25	0.2406	3360.65
10	Маса позиція 26	0.2056	4302.29
11	Гідрофобність позиція 21	0.2611	3043.79
12	Об'єм позиція 25	0.1613	6886.92
13	Маса позиція 25	0.1876	4908.33
14	pKa NH2 позиція 20	0.2187	2795.67
15	Поверхня позиція 25	0.2723	2131.50
16	Гідрофобність позиція 25	0.1902	2821.42
17	Гідрофобність позиція 25	0.2323	2102.88
18	Дипольний момент позиція 11	0.2752	1897.52
19	Дипольний момент позиція 25	0.2638	1957.76
20	Маса позиція 26	0.1168	6201.21

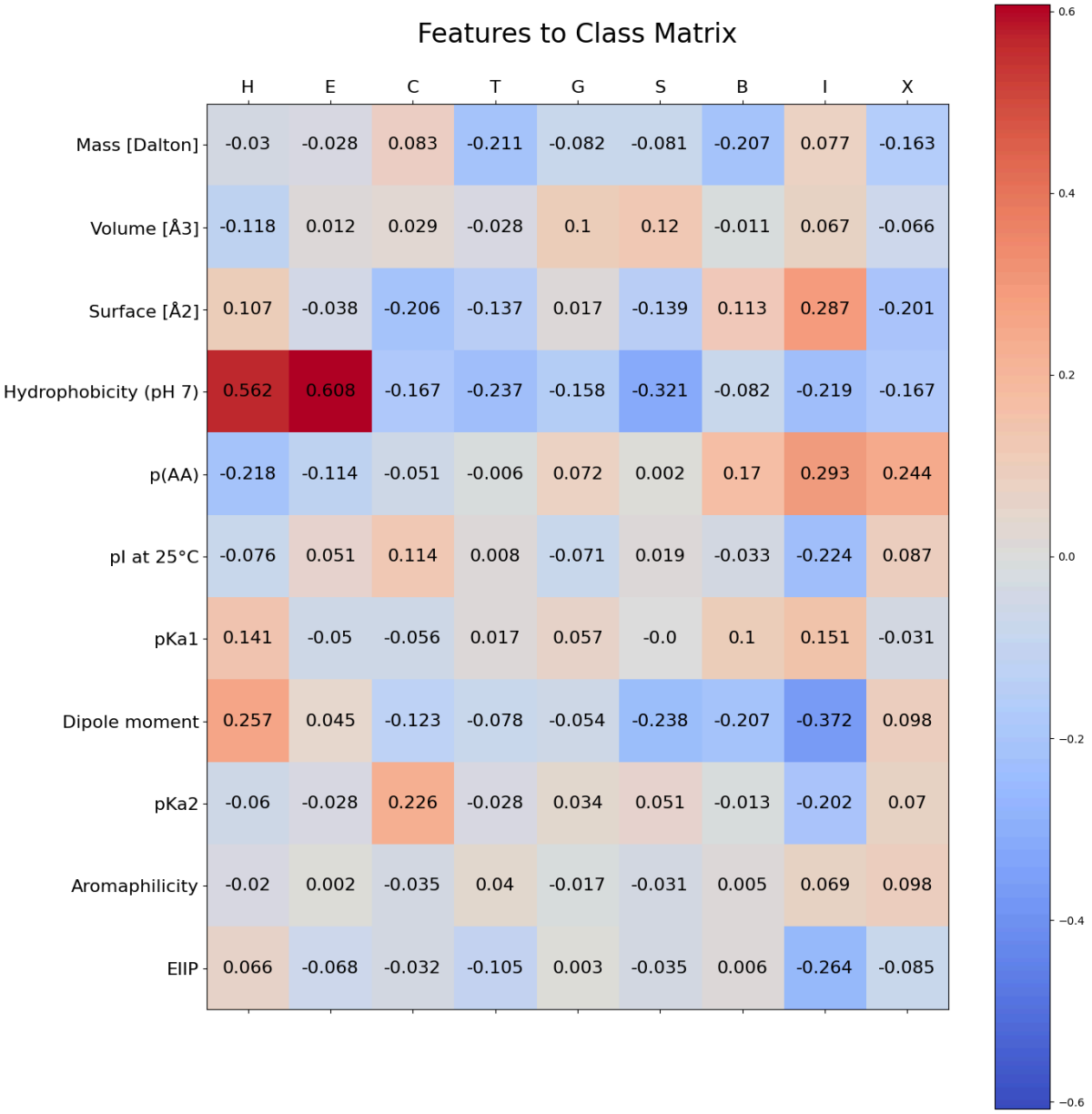
Додаток Д. Графіки розкиду для pKa NH2 та відповідних Шаплі значень для класу T (бета поворот)



Додаток Е. Матриця помилок для XGBoost глобальної сурогатної моделі для вторинної структури



Додаток Є. Матриця коефіцієнтів W_{class} з формули передбачення вторинної структури

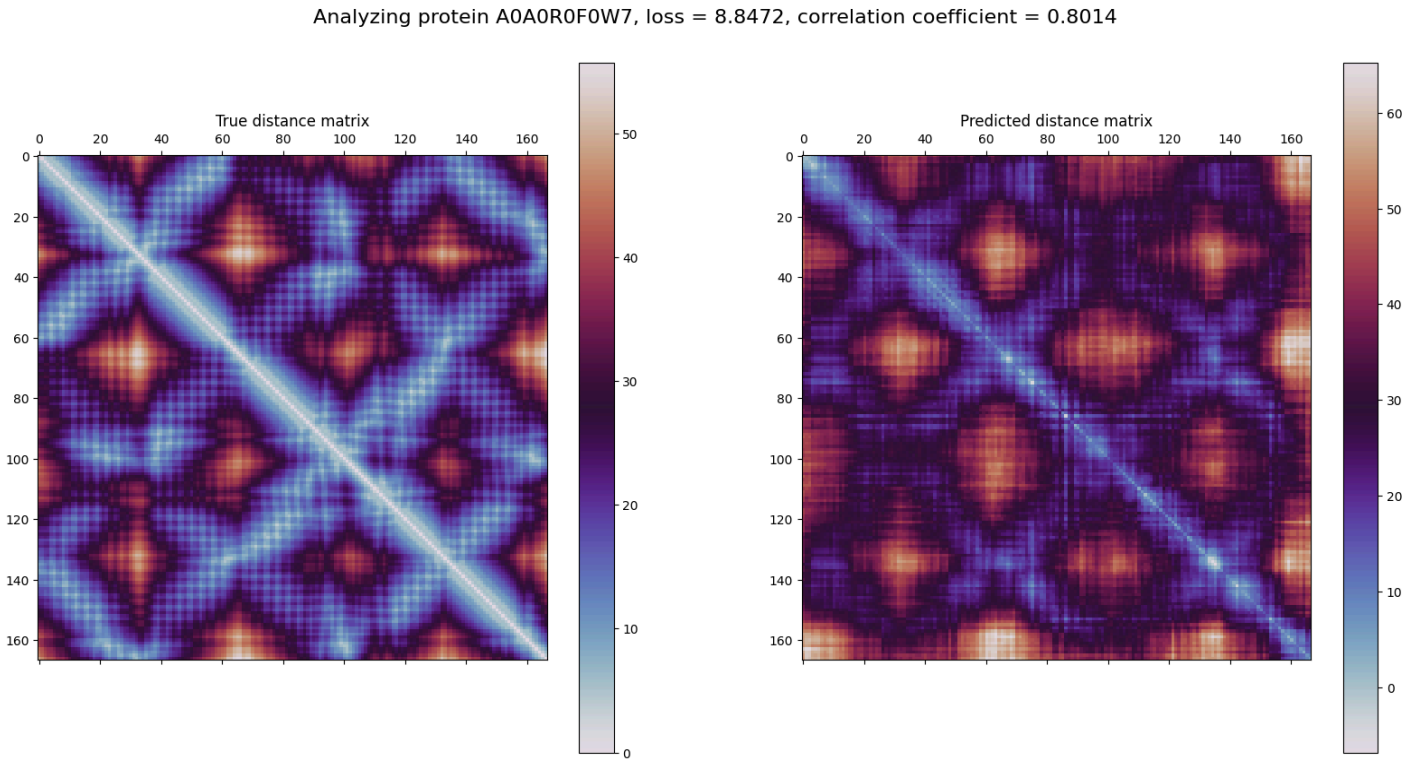


Таблиця. Результати класифікації на датасеті вторинних структур за допомогою дерев XGBoost, глобальний сурогат

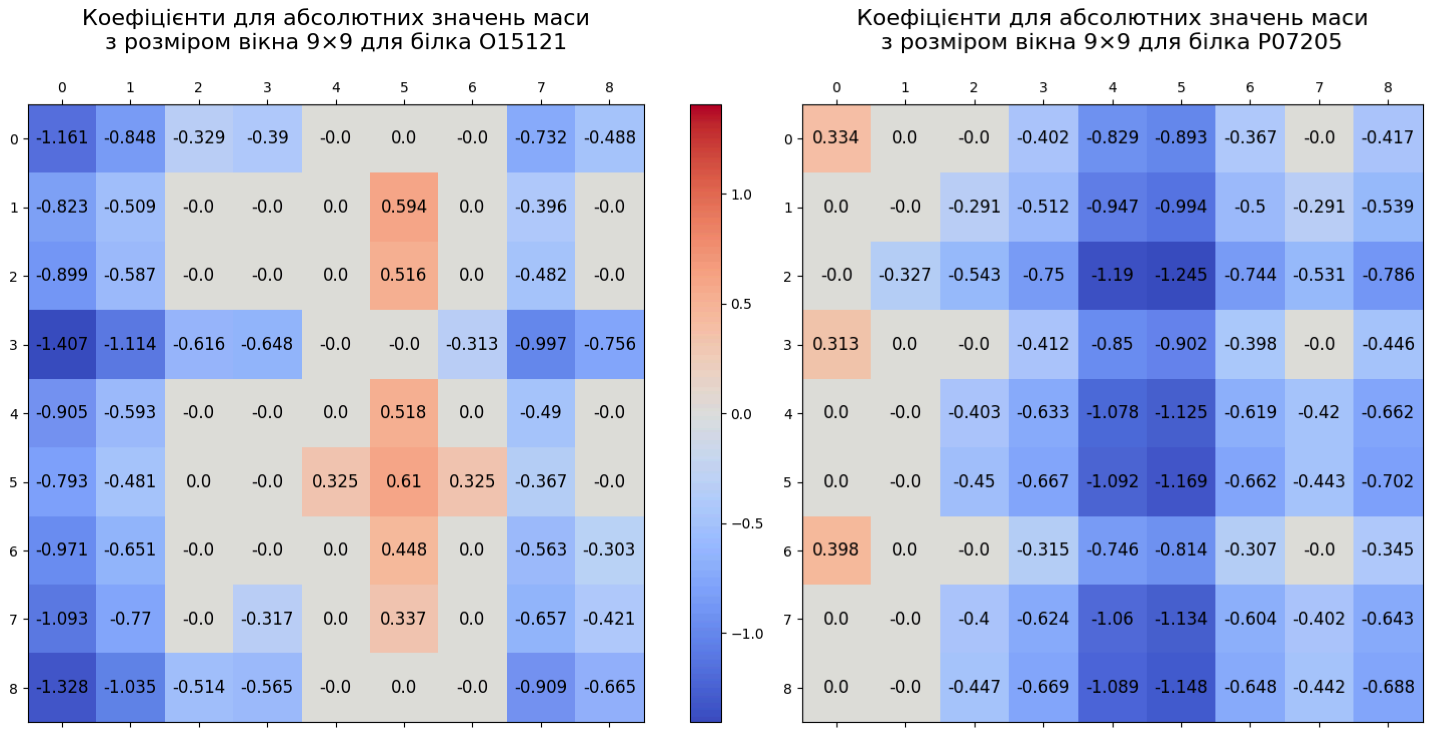
	Влучність (precision)	**Повнота (recall)**	**F1-скор (F1-score)**	**Кількість прикладів**
H	0.588	0.805	0.68	19860
E	0.547	0.595	0.57	12933

C	0.389	0.403	0.396	11532
T	0.368	0.299	0.33	6568
G	0.241	0.017	0.031	2317
S	0.267	0.065	0.105	4888
B	0.25	0.002	0.003	653
I	0.0	0.0	0.0	5
X	0.604	0.433	0.504	4226
Точність	0.516	0.516	0.516	--
Макро точність	0.361	0.291	0.291	62982

Додаток Ж. Приклад отриманих матриць дистанцій з моделі AminoBERT+BiLinear



Додаток З. Приклад коефіцієнтів для абсолютних значень маси для білка дельта-4-десатурази та фосфогліцерат-кінази 2 (код, наведений зверху - UniprotID)



Додаток И. Апроксимація та реальні Шаплі базові значення для позиційного фундаменту майбутнього третинного сурогату