Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет Комп'ютерних наук
(повна назва)
Кафедра Штучного інтелекту
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА
Пояснювальна записка

рівень вищої освіти другий (магістерський)

Передбачення типів взаємодій лікарняних засобів
за допомогою графових нейромереж
(тема)

Виконав:
здобувач другого року навчання,
групи СШІм-24-2
Ігор РОДІОНОВ
(власне ім'я, прізвище)
Спеціальність 122 Комп'ютерні науки
(код і повна назва спеціальності)
Тип програми освітньо-наукова
(освітньо-професійна або освітньо-наукова)
Освітня програма Системи штучного інтелекту
(повна назва освітньої програми)
Керівник проф. Лариса ЧАЛА
(посада, власне ім'я, прізвище)

Допускається до захисту

Завідувач кафедри ШІ _____ Лариса ЧАЛА
(підпис) (власне ім'я, прізвище)

2026 р.

Харківський національний університет радіоелектроніки

Факультет Комп'ютерних наук

Кафедра Штучного інтелекту

Рівень вищої освіти другий (магістерський)

Спеціальність 122 Комп'ютерні науки
(код і повна назва)

Тип програми освітньо-наукова
(освітньо-професійна або освітньо-наукова)

Освітня програма Системи штучного інтелекту
(повна назва)

ЗАТВЕРДЖУЮ:
Зав. кафедри _____
(підпис)
«_____» _____ 20 ___ р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

здобувачеві Родіонову Ігорю Олеговичу
(прізвище, ім'я, по батькові)

1. Тема роботи Передбачення типів взаємодій лікарняних засобів за допомогою графових нейромереж

затверджена наказом університету від 6 квітня 20 26 р. № 269Ст
2. Термін подання здобувачем роботи до екзаменаційної комісії 20 травня 20 26 р.
3. Вихідні дані до роботи Датасет DrugBank (6.0); наукові публікації за темою DDI та GNN; засоби Python, PyTorch, PyTorch Geometric.

4. Перелік питань, що потрібно опрацювати в роботі
1) Аналіз предметної галузі передбачення типів взаємодій лікарських препаратів
2) Методи та засоби розв'язання задачі
3) Розробка та реалізація системи
4) Експериментальне дослідження та аналіз результатів

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Строк / термін виконання етапів роботи	Примітка
1	Отримання завдання на кваліфікаційну роботу	06.04.2026	виконано
2	Аналіз предметної галузі, огляд методів та	14.04.2026	виконано
	існуючих підходів до передбачення DDI		
3	Постановка задачі дослідження та обґрунтування	22.04.2026	виконано
	методологічного підходу		
4	Експериментальне моделювання та навчання	30.04.2026	виконано
	моделі		
5	Написання пояснювальної записки	07.05.2026	виконано
6	Перевірка на академічний плагіат	11.05.2026	виконано
7	Нормоконтроль	13.05.2026	виконано
8	Рецензування	15.05.2026	виконано
9	Підготовка презентації та доповіді	17.05.2026	виконано
10	Попередній захист	19.05.2026	виконано
11	Захист перед ЕК	20.05.2026	

Дата видачі завдання 6 квітня 20 26 р.

Здобувач _____
(підпис)

Керівник роботи _____ проф. Лариса ЧАЛА
(підпис) (посада, власне ім’я, прізвище)

РЕФЕРАТ

Пояснювальна записка: 78 с., 5 рис., 8 табл., 1 дод., 22 джерела.

ВЗАЄМОДІЇ ЛІКАРСЬКИХ ПРЕПАРАТІВ, ГЛИБОКЕ НАВЧАННЯ, ГРАФОВІ НЕЙРОННІ МЕРЕЖІ, КАСКАДНА КЛАСИФІКАЦІЯ, МОВНІ МОДЕЛІ БІЛКІВ, МУЛЬТИМОДАЛЬНЕ НАВЧАННЯ, ПОЛІФАРМАЦІЯ, ХОЛОДНИЙ СТАРТ.

Об'єкт дослідження – процес передбачення типів та ефектів взаємодій лікарських препаратів на основі їх структурних і фармакологічних властивостей.

Предмет дослідження – методи та архітектури глибокого навчання для ієрархічної класифікації ефектів лікарських взаємодій в умовах cold-start оцінювання.

Мета роботи – розробка та експериментальне дослідження мультимодальної каскадної системи прогнозування лікарських взаємодій, що забезпечує передбачення для нових молекул без попередньої інформації про їх взаємодії.

Методи дослідження – графові нейронні мережі (GIN) для кодування молекулярних структур зі SMILES, мовна модель білків ESM-2 650M для представлення білкових мішеней, каскадна ієрархічна декомпозиція задачі класифікації, Focal Loss і зважений семплер для роботи з дисбалансом класів, drug-wise розбиття даних для забезпечення cold-start протоколу.

Розроблено мультимодальну каскадну систему прогнозування лікарських взаємодій на основі DrugBank 6.0, що поєднує GIN-енкодер молекулярних графів і ESM-2 ембеддинги білкових мішеней. У cold-start протоколі система досягає macro AUC 0.883 для PK-ефектів і 0.845 для non-PK міток, що є конкурентоспроможним результатом для сценарію скринінгу нових сполук.

ABSTRACT

Master's thesis contains: 78 pp., 5 fig., 8 tabl., 1 ann., 22 references.

COLD START, DEEP LEARNING, DRUG-DRUG INTERACTIONS, GRAPH NEURAL NETWORKS, HIERARCHICAL CLASSIFICATION, MULTIMODAL LEARNING, POLYPHARMACY, PROTEIN LANGUAGE MODELS

Object of research – the process of predicting types and effects of drug-drug interactions based on structural and pharmacological properties of drugs.

Subject of research – deep learning methods and architectures for hierarchical classification of drug interaction effects under cold-start evaluation conditions.

Purpose of the work – development and experimental investigation of a multimodal cascaded system for drug-drug interaction prediction that enables inference for novel molecules without prior interaction data.

Research methods – Graph Isomorphism Networks (GIN) for molecular structure encoding from SMILES, protein language model ESM-2 650M for protein target representation, hierarchical cascaded task decomposition, Focal Loss and weighted sampling for class imbalance handling, drug-wise data splitting for cold-start protocol enforcement.

A multimodal cascaded system for drug-drug interaction prediction was developed based on DrugBank 6.0, combining a GIN molecular encoder and ESM-2 protein target embeddings. Under cold-start conditions the system achieves macro AUC 0.883 for PK effects and 0.845 for non-PK labels, demonstrating competitive performance for novel compound screening.

ЗМІСТ

Перелік умовних позначень, символів, одиниць, скорочень і термінів 8
Вступ 9
1 Аналіз предметної галузі передбачення типів взаємодій лікарських препаратів 10
1.1 Загальний огляд проблеми передбачення взаємодій лікарських препаратів 10
1.2 Поняття та класифікація взаємодій лікарських препаратів 12
1.3 Методи виявлення та передбачення DDI: від лабораторних підходів до машинного навчання 15
1.3.1 Лабораторні та клінічні методи виявлення DDI 16
1.3.2 Обчислювальні підходи на основі машинного навчання 18
1.3.3 Графові нейронні мережі як основа сучасних методів передбачення DDI 20
1.4 Огляд існуючих моделей передбачення DDI 23
1.4.1 Ранні підходи на основі нейронних мереж (DeepDDI) 24
1.4.2 Моделі на основі графових нейромереж 25
1.4.3 Моделі з використанням гетерогенних графів та графів знань (TIGER, KnowDDI, ExDDI) 27
1.4.4 Порівняльний аналіз існуючих моделей передбачення DDI 29
1.4.5 Невирішені проблеми існуючих підходів 30
1.5 Постановка задачі дослідження 32
2 Методи та засоби розв’язання задачі 35
2.1 Формальна постановка задачі прогнозування лікарських взаємодій 35
2.2 Методи представлення молекулярних структур на основі графових нейронних мереж 37

2.3 Методи представлення білкових мішеней 41
2.4 Каскадна архітектура класифікації ефектів взаємодій 44
2.5 Методи навчання моделі в умовах дисбалансу класів 47
3 Розробка та реалізація системи 51
3.1 Джерела даних та попередня обробка 51
3.2 Схема бази даних та організація зберігання 52
3.3 Формування навчальної, валідаційної та тестової вибірок 54
3.4 Обчислення та збереження ембеддингів білкових послідовностей .. 55
3.5 Реалізація компонентів моделі 56
3.6 Процес навчання та підбір гіперпараметрів 59
4 Експериментальне дослідження та аналіз результатів 61
4.1 Протокол оцінювання та базові моделі 61
4.2 Результати навчання компонентів каскаду 62
4.3 Оцінювання в умовах холодного старту 65
4.4 Порівняння з існуючими підходами 67
4.5 Аналіз результатів за класами ефектів 69
4.6 Вплив умови холодного старту на якість передбачень 71
Висновки 73
Перелік джерел посилання 75
Додаток А Відомість кваліфікаційної роботи 78

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

AP – Average Precision – середня точність;

AUC – Area Under the Curve – площа під ROC-кривою;

CYP450 – Cytochrome P450 – цитохром P450, сімейство ферментів метаболізму лікарських препаратів;

DDI – Drug-Drug Interaction – взаємодія лікарських препаратів;

ESM-2 – Evolutionary Scale Modeling 2 – мовна модель для білкових послідовностей;

F1 – F1-score – F1-міра, гармонічне середнє точності та повноти;

GCN – Graph Convolutional Network – графова згорткова нейронна мережа;

GIN – Graph Isomorphism Network – мережа ізоморфізму графів;

GNN – Graph Neural Network – графова нейронна мережа;

PK – Pharmacokinetics – фармакокінетика;

SMILES – Simplified Molecular Input Line Entry System – рядковий формат запису молекулярної структури;

SSP – Structural Similarity Profile – профіль структурної подібності молекул.

ВСТУП

Поліпрагмазія – одночасне застосування кількох препаратів – є поширеною клінічною практикою, що суттєво підвищує ризик непередбачених лікарських взаємодій. Кількість можливих комбінацій зростає квадратично з кількістю призначень, тому експериментальна перевірка кожної пари є практично неможливою, а обчислювальні методи прогнозування DDI стають необхідним інструментом доклінічного скринінгу.

Сучасні підходи на основі глибокого навчання – графові нейронні мережі для молекулярних структур і мовні моделі білків для біологічного контексту – демонструють сильні результати, але переважно у warm-start режимі, де тестові препарати вже були присутні під час навчання. Такий протокол не відповідає реальному сценарію скринінгу нових молекул. Додатковою проблемою є вкрай нерівномірний розподіл клінічних ефектів, що охоплюють сотні класів і робить пряму багатокласову класифікацію малоефективною.

Метою цієї роботи є розробка мультимодальної каскадної системи прогнозування DDI, що поєднує GIN-енкодер молекулярних графів, ESM-2 ембеддинги білкових мішеней і ієрархічну декомпозицію ефектів, та її оцінювання у суворому cold-start протоколі на базі DrugBank 6.0. Результати можуть бути застосовані у фармацевтичній розробці для раннього виявлення потенційних взаємодій нових сполук і у клінічній підтримці прийняття рішень при призначенні комбінованої терапії.

1 АНАЛІЗ ПРЕДМЕТНОЇ ГАЛУЗІ ПЕРЕДБАЧЕННЯ ТИПІВ ВЗАЄМОДІЙ ЛІКАРСЬКИХ ПРЕПАРАТІВ

У даному розділі проведено аналіз предметної галузі задачі передбачення типів взаємодій лікарських препаратів. Розглянуто клінічну значущість проблеми поліфармації та її масштаб у сучасній медичній практиці, систематизовано механізми та класифікації DDI, проаналізовано еволюцію методів виявлення взаємодій від лабораторних підходів до сучасних архітектур глибокого навчання. Виконано порівняльний огляд провідних моделей передбачення DDI, визначено їхні обмеження та сформульовано постановку задачі дослідження.

1.1 Загальний огляд проблеми передбачення взаємодій лікарських препаратів

Задача передбачення взаємодій лікарських препаратів є актуальною інженерною проблемою на перетині обчислювальної біології та машинного навчання. Її практична значущість визначається простим фактом: коли людина одночасно приймає кілька препаратів, вони можуть впливати один на одного непередбачуваним чином – підсилювати або нейтралізувати ефект, або спричиняти токсичні реакції. Масштаб цієї проблеми робить ручну перевірку всіх можливих комбінацій практично неможливою, що і створює запит на автоматизовані обчислювальні методи.

Одночасний прийом великої кількості препаратів є поширеним явищем у сучасній клінічній практиці. Єдиного загальноприйнятого визначення для цього явища досі не існує – систематичний огляд зафіксував щонайменше 26 різних трактувань, проте найпоширенішим залишається кількісний критерій: одночасний прийом п'яти і більше препаратів [1]. За оцінками, понад 40% пацієнтів старше 65 років відповідають цьому критерію, а частина з них приймає десять і більше засобів одночасно [3].

Старіння населення та зростання частоти хронічних захворювань роблять цю тенденцію стійкою, перетворюючи проблему з індивідуальної клінічної на системну.

Клінічні наслідки такої ситуації є серйозними. Встановлено статистично значущий зв'язок між прийомом великої кількості препаратів та підвищеним ризиком госпіталізації і смертності навіть після врахування тяжкості основних захворювань [4]. Клінічно значущі взаємодії препаратів у літніх пацієнтів є суттєво частішими порівняно із загальною популяцією, причому потенційно шкідливі комбінації виявлялись у переважної більшості тих, хто приймав п'ять і більше засобів [2]. В онкологічних хворих такі взаємодії асоційовані зі збільшенням частоти токсичних ускладнень, зниженням ефективності протипухлинної терапії та загальної виживаності [5].

Центральним механізмом, через який виникають ці проблеми, є саме взаємодії між одночасно призначеними препаратами. Переважна більшість таких ускладнень опосередковується через два типи взаємодій: зміну концентрації одного препарату під впливом іншого, або безпосередній вплив на біологічні мішені [3]. Проте важливо те, що ефект конкретної комбінації часто не є очевидним і не може бути передбачений без аналізу молекулярних властивостей обох препаратів.

Саме тут виникає фундаментальна математична перешкода, яка і робить цю задачу інженерною. Кількість унікальних пар для N препаратів становить $N \cdot (N-1)/2$ – і вже при кількох сотнях сполук мова йде про десятки тисяч пар, при тисячах – про мільйони. База DrugBank 6.0 містить понад 14 000 лікарських засобів і понад 2 600 000 задокументованих взаємодій між ними [20]. Проте навіть ця вражаюча кількість охоплює лише незначну частку від загального простору потенційних пар: для 14 000 сполук кількість унікальних комбінацій перевищує 98 мільйонів. Масштаб комбінаторної проблеми робить вичерпне лабораторне тестування недосяжним [14].

Методи машинного навчання є на сьогодні єдиним масштабованим рішенням цієї задачі: вони здатні узагальнювати закономірності з наявних відомих взаємодій і поширювати їх на невивчені пари препаратів, передбачаючи не лише факт взаємодії, але й її конкретний тип і механізм – що є критично важливим для практичного застосування. Саме архітектурні рішення, що уможливлюють таку детальну класифікацію, становлять предмет подальшого розгляду.

1.2 Поняття та класифікація взаємодій лікарських препаратів

Взаємодія лікарських препаратів (Drug-Drug Interactions, DDI) – це зміна фармакологічного ефекту одного препарату під впливом іншого, що проявляється у зміні його терапевтичної активності, токсичності або обох характеристик одночасно. Слід зазначити, що не всі такі взаємодії є небажаними: частина з них цілеспрямовано використовується в клінічній практиці – наприклад, комбінування інгібіторів протеази з низькими дозами ритонавіру для посилення ефекту першого, або застосування фолінової кислоти для зниження токсичності метотрексату. База DrugBank 6.0 фіксує задокументовані взаємодії з деталізацією їх механізму та клінічного значення [20]. Проте переважна більшість взаємодій є небажаними, і саме їхнє передбачення становить пріоритетне завдання обчислювальної фармакології.

За механізмом виникнення всі DDI поділяються на два принципово різних типи: фармакокінетичні та фармакодинамічні.

Фармакокінетичні взаємодії виникають тоді, коли один препарат змінює концентрацію іншого в організмі. Це відбувається на будь-якому з чотирьох етапів руху препарату в організмі – всмоктування, розподіл, метаболізм і виведення. Найпростіший приклад на етапі всмоктування: антациди змінюють кислотність шлунку і знижують біодоступність інших препаратів, а деякі мінерали утворюють з антибіотиками нерозчинні

сполуки, повністю блокуючи їх засвоєння. На етапі розподілу препарати можуть конкурувати за зв'язування з білками крові, що змінює кількість активної речовини у вільній формі.

Центральним і найбільш клінічно значущим механізмом є метаболізм через систему ферментів печінки – цитохром Р450 (CYP450). Ці ферменти відповідають за розщеплення переважної більшості препаратів в організмі. Понад 70% клінічно значущих фармакокінетичних взаємодій опосередковані саме через систему CYP450 [8]. База SuperCYP систематизує відомі відносини між препаратами і цими ферментами, підкреслюючи, що один і той самий препарат може одночасно прискорювати, уповільнювати або блокувати роботу різних ферментів [7].

Механізм взаємодії через CYP450 можна пояснити так: якщо препарат А блокує фермент, який зазвичай розщеплює препарат В, то концентрація препарату В в крові зростає до потенційно токсичного рівня. І навпаки – якщо препарат А прискорює роботу цього ферменту, препарат В розщеплюється надто швидко і втрачає терапевтичний ефект. Фермент CYP3A4 є найпоширенішим і метаболізує близько половини всіх препаратів на фармацевтичному ринку. Фермент CYP2C8 відіграє ключову роль у метаболізмі ряду препаратів, і його блокування може більш ніж у вісім разів збільшувати концентрацію деяких з них, провокуючи тяжкі ускладнення [9].

Класичним прикладом клінічно значущої взаємодії через CYP450 є поєднання клопідогрелю з омепразолом. Клопідогрель є проліками – він активується в організмі через фермент CYP2C19. Омепразол блокує цей фермент, внаслідок чого клопідогрель не активується належним чином і втрачає свій антитромботичний ефект. При цьому дані лабораторних досліджень не дозволяли повністю передбачити клінічний масштаб цієї взаємодії через складний каскад перетворень за участю кількох ферментів [10]. Цей приклад ілюструє, чому навіть добре вивчені взаємодії можуть мати неочікувані наслідки – і чому автоматизовані методи передбачення є необхідними.

Для систематизації взаємодій через систему CYP450 нижче наведено таблицю 1.1, яка узагальнює ключові ферменти, препарати що з ними взаємодіють, та типові клінічні наслідки.

Таблиця 1.1 – Типи фармакокінетичних взаємодій через систему CYP450

Фермент	Препарати-субстрати	Блокатори	Прискорювачі	Клінічний наслідок
CYP3A4	Симвастатин, циклоспорин, мідазолам	Кетоконазол, кларитроміцин	Рифампіцин, карбамазепін	Токсичність або втрата ефективності
CYP2C8	Репаглінід, паклітаксел, піоглітазон	Гемфіброзил, монтелукаст	Рифампіцин	Тяжка гіпоглікемія, підвищена токсичність
CYP2C19	Клопідогрель, омепразол, циталопрам	Омепразол, флуконазол	Рифампіцин	Зниження антитромботичного ефекту
CYP2D6	Кодеїн, метопролол, флуоксетин	Флуоксетин, пароксетин	–	Надмірна седація або відсутність ефекту
CYP1A2	Теофілін, кофеїн, клозапін	Флувоксамін, ципрофлоксацин	Куріння, омепразол	Токсичність теофіліну, зміни рівня клозапіну

Фармакодинамічні взаємодії, на відміну від фармакокінетичних, не змінюють концентрацію препарату в крові – вони впливають безпосередньо на біологічні мішені. Адитивний синергізм спостерігається, коли два препарати з подібним механізмом дії посилюють загальний ефект – наприклад, комбінація двох антигіпертензивних засобів для досягнення цільового рівня тиску. Потенціювання є більш вираженим явищем – один препарат непропорційно посилює ефект іншого; класичним прикладом є поєднання опіоїдів з бензодіазепінами, що може призводити до фатальної

дихальної депресії. Антагонізм, навпаки, знижує ефективність одного з препаратів – наприклад, застосування бета-агоністів при одночасному прийомі бета-блокаторів нівелює бронхорозширюючий ефект.

За ступенем клінічної небезпеки взаємодії поділяють на чотири категорії. Протипоказані комбінації несуть ризик серйозних, часто незворотних наслідків і потребують абсолютного уникнення. Серйозні взаємодії потребують активного моніторингу або корекції доз. Помірні взаємодії допустимі за умови обґрунтованої клінічної необхідності. Незначні взаємодії мають мінімальну практичну цінність з точки зору корекції терапії.

Для того щоб задача класифікації DDI стала придатною для машинного навчання, необхідно формалізувати цю різноманітність механізмів у вигляді стандартизованої онтології. У роботі з моделлю DeepDDI запропоновано онтологію з 86 стандартизованих типів взаємодій, де кожен тип кодує одночасно механізм, напрямок ефекту та клінічний наслідок [6]. Наприклад, один тип описує зниження метаболізму препарату A при комбінуванні з препаратом B з відповідним зростанням ризику токсичності, тоді як інший кодує зниження терапевтичної ефективності через фармакодинамічний антагонізм. Така формалізація перетворює задачу на мультикласову класифікацію з 86 можливими класами на виході – саме ця постановка є стандартом для сучасних моделей і використовується в цій роботі.

1.3 Методи виявлення та передбачення DDI: від лабораторних підходів до машинного навчання

Різноманіття механізмів DDI, розглянутих у попередньому підрозділі, зумовлює принципову складність їх виявлення та систематизації. Саме ця багатовимірність стала рушієм послідовної еволюції методів – від прямих лабораторних спостережень до складних архітектур глибокого навчання.

Традиційні експериментальні методи, хоча й залишаються золотим стандартом верифікації взаємодій, принципово не масштабуються на мільйони потенційних пар сполук [11]. Це стимулювало розвиток обчислювальних підходів: ранні моделі машинного навчання забезпечили масштабованість, однак виявились нездатними вловити тонкі структурні та біологічні закономірності, що визначають тип і напрямок взаємодії. Огляди останніх років фіксують виразний зсув у бік методів глибокого навчання, зокрема графових нейронних мереж [13].

Цей підрозділ відтворює цю еволюцію: спочатку розглядаються лабораторні методи з аналізом їх обмежень, далі – класичне машинне навчання, і нарешті графові нейронні мережі як домінуюча сучасна парадигма.

1.3.1 Лабораторні та клінічні методи виявлення DDI

Традиційна ідентифікація DDI спирається на ієрархію експериментальних методів, кожен з яких надає цінну, але принципово обмежену інформацію.

На найнижчому рівні цієї ієрархії знаходяться лабораторні дослідження – тести на клітинних культурах і мікросомах печінки, які дозволяють у контрольованих умовах оцінити здатність досліджуваної сполуки прискорювати або блокувати конкретні ферменти системи CYP450 [8]. Перевагами є відтворюваність і відносно низька вартість порівняно з клінічними випробуваннями. Водночас головним обмеженням є те, що ізольована лабораторна система позбавлена транспортних білків, метаболічних каскадів і міжорганних взаємодій, характерних для живого організму. Як наслідок, лабораторні дані нерідко не корелюють з реальними клінічними спостереженнями.

Наступний рівень – клінічні фармакокінетичні випробування – усуває ці обмеження ціною різкого зростання витрат і тривалості. Клінічне

дослідження взаємодії однієї пари препаратів потребує розробки протоколу, рекрутування добровольців та аналітичного контролю; загальна вартість може перевищувати мільйон доларів і займати від кількох місяців до трьох років. Регуляторні вимоги до вивчення DDI поширюються лише на обмежений перелік найбільш ймовірних взаємодій, тоді як переважна більшість потенційних комбінацій ніколи не потрапляє до клінічних протоколів [11].

Постмаркетинговий фармаконагляд через системи спонтанної звітності – американську FAERS і європейську EudraVigilance – теоретично дозволяє виявляти взаємодії, що проявляються лише в реальній клінічній практиці. Однак цей підхід має системні вади: недозвітність побічних реакцій сягає за різними оцінками 90–95%, сигнал про нову взаємодію формується із затримкою в роки від початку застосування препарату, а причинно-наслідковий зв'язок між конкретною комбінацією і спостережуваним ефектом часто залишається невизначеним [13]. Отже, фармаконагляд є реактивним інструментом – він документує вже наявні проблеми, але не запобігає їм.

Кількісний аргумент на користь принципової недостатності всіх перерахованих підходів є переконливим. Для бази DrugBank 6.0 з понад 14 000 сполук загальна кількість унікальних пар перевищує 98 мільйонів [14]. Якщо навіть уявити лабораторію, здатну проводити одне клінічне дослідження на тиждень безперервно, повне охоплення лише цього масиву зайняло б близько двох мільйонів років. Цей розрахунок відображає реальну прірву між можливостями експериментальних методів і обсягом фармакологічного простору – саме вона і створює запит на обчислювальні підходи.

Обчислювальні методи, навпаки, здатні охоплювати мільярди потенційних пар за лічені години, спираючись на структурні та біологічні ознаки молекул, а не на безпосередній фізичний експеримент. Саме тому машинне навчання і, зокрема, графові нейронні мережі, що дозволяють

природно кодувати молекулярні структури у вигляді графів, стали провідним напрямом досліджень у галузі передбачення лікарських взаємодій.

1.3.2 Обчислювальні підходи на основі машинного навчання

Застосування методів машинного навчання до задачі передбачення DDI стало можливим лише після розв'язання фундаментальної проблеми представлення: алгоритми оперують числовими векторами, тоді як хімічна структура препарату є складним графовим об'єктом. Перетворення молекули на числовий вектор реалізується кількома принципово різними способами.

Найбільш компактним рядковим форматом є SMILES – Simplified Molecular Input Line Entry System, що кодує атоми, зв'язки і циклічні структури за допомогою ASCII-символів. Наприклад, аспірин записується як CC(=O)Oc1ccccc1C(=O)O. Однак сирий рядок SMILES безпосередньо непридатний для класичних алгоритмів, тому на практиці він слугує проміжним представленням для генерації більш формалізованих дескрипторів. Молекулярні відбитки – зокрема ECFP та MACCS-ключі – перетворюють молекулу на бінарний вектор фіксованої довжини, де кожна позиція відповідає наявності або відсутності певного структурного фрагмента. Третій підхід – фізико-хімічні дескриптори – агрегує кількісні характеристики молекули: молекулярну масу, розчинність у жирах, полярну площу поверхні, кількість донорів і акцепторів водневих зв'язків. Вибір типу представлення суттєво впливає на якість фінальної моделі і визначає, які закономірності вона здатна вловити [14].

Спираючись на ці векторні представлення, дослідники застосовували широкий спектр класичних алгоритмів. Метод опорних векторів (SVM) демонструє ефективність при відносно невеликих вибірках, однак його продуктивність погіршується при зростанні кількості класів. Випадковий

ліс забезпечує стійкість до перенавчання і дозволяє оцінювати важливість ознак. Логістична регресія і метод k-найближчих сусідів традиційно слугують базовими лініями порівняння.

Класичним раннім застосуванням машинного навчання до передбачення DDI є робота 2014 року [12], де було інтегровано чотири типи ознак – фенотипові схожості, терапевтичні класи, хімічну структурну схожість і геномні профілі. Отримані значення AUROC для ряду класифікаторів перевищували 0.9. Водночас задача формулювалась як бінарна – взаємодіють препарати чи ні – і не передбачала ідентифікації механізму чи типу взаємодії. Саме ця бінарність є першим і найважливішим структурним обмеженням класичного машинного навчання.

Систематичний аналіз обмежень класичних підходів виявляє кілька ключових проблем [11]. По-перше, переважна більшість моделей вирішує задачу наявності або відсутності взаємодії, не розрізняючи її механізм – інформацію, критичну для практичного застосування. По-друге, моделі на основі фіксованих дескрипторів не мають уявлення про реальну структуру молекули: два препарати є для них лише точками у векторному просторі. По-третє, якість моделі повністю визначається якістю вручну сконструйованих ознак, і будь-яка хімічна інформація, не відображена у використаних дескрипторах, залишається невидимою для алгоритму. По-четверте, проблема дисбалансу класів є особливо гострою при переході від бінарної до мультикласової постановки: рідкісні, але клінічно критичні типи взаємодій виявляються систематично недопредставленими. Додатковою проблемою є узагальнення на нові препарати: класичні моделі, навчені на парах відомих сполук, часто виявляються нездатними коректно передбачати взаємодії за участю нових молекул [13].

Проміжне місце між класичним машинним навчанням і сучасними архітектурами займають моделі, що безпосередньо оперують рядками SMILES як вхідними даними. SmileGNN використовує SMILES для побудови молекулярних графів і передає їх на вхід графових нейронних

мереж, демонструючи суттєве покращення метрик порівняно з підходами на основі фіксованих відбитків [21]. Ця робота є показовою ілюстрацією ключової інтуїції: якщо молекула за своєю природою є графом, то і модель має оперувати графовою структурою. Інтеграція глибокого навчання і графів знань відкрила можливість моделювати не лише хімічну схожість, але й біологічний контекст взаємодії [15].

Сукупність розглянутих обмежень формує чіткий запит на якісно інший клас моделей – здатних безпосередньо навчатись на структурних і реляційних даних без попередньої агрегації у фіксовані вектори. Відповіддю на цей запит стали графові нейронні мережі.

1.3.3 Графові нейронні мережі як основа сучасних методів передбачення DDI

Ключова слабкість класичних методів має структурну природу: перетворення молекули у вектор фіксованої довжини неминуче втрачає топологічну інформацію. Молекулярний відбиток кодує наявність певних фрагментів, але не їх взаємне розташування і локальний хімічний контекст. Два різних сполуки можуть мати ідентичні набори фрагментів, але принципово різні властивості – саме через спосіб їх з'єднання. Ця втрата є не технічним артефактом, а принциповим наслідком відображення графового об'єкта у звичайний числовий простір.

Природним представленням молекули є граф: атоми виступають вузлами з атрибутами – атомний номер, формальний заряд, ступінь вузла, ознака ароматичності – а хімічні зв'язки є ребрами з типами: одинарний, подвійний, потрійний або ароматичний. Таке представлення зберігає повну топологічну інформацію і дозволяє алгоритму безпосередньо оперувати структурою молекули. Перехід до графових архітектур став якісним зсувом у методології передбачення DDI, оскільки вперше надав моделям можливість працювати з повною структурою молекули [11].

Поряд із молекулярними графами у фармакології широко застосовуються графи знань – гетерогенні мережі, де вузли представляють сутності різних типів: препарати, білки, гени, біологічні шляхи, захворювання, – а ребра кодують відомі відносини між ними. Hetionet об'єднує понад 47 000 вузлів 11 типів і більше мільйона ребер 24 типів, інтегруючи дані з баз DrugBank, UniProt та інших джерел. DRKG охоплює близько 100 000 сутностей. Саме такі гетерогенні структури дозволяють кодувати біологічний контекст взаємодії, недоступний при аналізі ізольованих молекулярних структур [14]. Концепція гетерогенного графу є принципово важливою: різні типи ребер несуть різну семантику, і модель що розрізняє «препарат блокує фермент» від «препарат є субстратом транспортера» отримує суттєво більше інформації для передбачення типу взаємодії.

Центральним механізмом графових нейронних мереж є framework передачі повідомлень. Ідея полягає в ітеративному оновленні представлення кожного вузла на основі інформації від його безпосередніх сусідів. Формально, якщо позначити представлення вузла v на k-й ітерації як $h_v^{(k)}$, то оновлення відбувається за схемою:

$$h_v^{(k)} = \mathrm{UPDATE}\left(h_v^{(k-1)}, \mathrm{AGGREGATE}\left(h_u^{(k-1)} : u \in N(v)\right)\right), \qquad (1.1)$$

де $h_v^{(k-1)}$ – представлення самого вузла v з попередньої ітерації;

$N(v)$ – множина сусідів вузла v;

AGGREGATE – функція агрегації повідомлень від сусідів;

UPDATE – функція оновлення власного стану вузла.

Після k ітерацій вузол акумулює інформацію з усього свого k-hop оточення: для молекулярного графа це означає, що атом «знає» про свій локальний хімічний контекст у радіусі кількох зв'язків – саме ту інформацію, що визначає його роль у взаємодії з іншими молекулами. Цей

індуктивний принцип дозволяє GNN узагальнювати на нові молекули, яких не було у тренувальній вибірці – критична перевага перед класичними методами [15].

Три найбільш поширені архітектури GNN відрізняються способом реалізації функцій AGGREGATE і UPDATE. Graph Convolutional Network (GCN) використовує усереднення ознак сусідів з нормалізацією за їхніми ступенями, що забезпечує обчислювальну ефективність і стабільність навчання. Однак усі сусіди в GCN отримують однакові ваги – суттєве спрощення, адже у молекулярному графі не всі зв'язки однаково важливі. Graph Attention Network (GAT) усуває це обмеження, вводячи механізм уваги: вага кожного сусіда обчислюється динамічно під час навчання на основі схожості ознак вузлів, що дозволяє моделі самостійно визначати найбільш інформативні зв'язки. Застосування GNN до задачі DDI підтверджує, що графове кодування молекулярної структури суттєво покращує розпізнавання взаємодій порівняно з підходами на основі фіксованих відбитків [17]. Graph Isomorphism Network (GIN) є теоретично найпотужнішою з трьох архітектур: її функція агрегації є ін'єктивною – тобто різним наборам ознак сусідів гарантовано відповідають різні вихідні представлення, що робить GIN максимально дискримінативною при розрізненні молекул зі схожими підструктурами.

SmileGNN демонструє практичну реалізацію молекулярно-графового підходу: SMILES рядки перетворюються на атомарні графи, які передаються на вхід GNN для отримання молекулярних ембеддінгів, після чого ембеддінги пари препаратів об'єднуються для передбачення взаємодії [21]. Порівняно з підходами на основі фіксованих відбитків ця архітектура демонструє кращі метрики на стандартних бенчмарках. DeepDDI показала, що навіть нейромережеве оброблення векторних представлень дозволяє досягти значно кращої деталізації порівняно з класичними підходами, формалізувавши задачу як 86-класову класифікацію [6].

Існує два принципово різних способи формулювання задачі у термінах GNN [16]. У першому підході DDI-граф, де вузлами є препарати, а ребрами – відомі взаємодії, слугує безпосереднім вхідним об'єктом: задача формулюється як класифікація ребер або передбачення нових зв'язків у цьому графі. У другому підході GNN застосовується до молекулярних графів кожного препарату окремо для отримання структурних ембеддінгів, які потім об'єднуються і передаються класифікатору. Сучасні архітектури, як правило, поєднують обидва підходи: молекулярний ембеддінг кодує хімічну структуру, тоді як граф знань надає біологічний контекст. Саме ця дуальність є ключовою перевагою GNN-підходів – вони здатні одночасно враховувати структурну специфіку молекул і системний біологічний контекст їх взаємодій, що є недосяжним для жодного з класичних методів [13].

Таким чином, графові нейронні мережі вирішують одразу кілька обмежень класичних методів: вони оперують повною топологічною структурою молекули, природно інтегрують гетерогенну біологічну інформацію і підтримують мультикласову постановку задачі. Саме ці властивості зумовили домінування GNN-архітектур у сучасних дослідженнях із передбачення DDI і визначають методологічну основу цієї роботи.

1.4 Огляд існуючих моделей передбачення DDI

Утвердження графових нейронних мереж як провідної парадигми у передбаченні DDI супроводжується швидким нарощуванням архітектурної різноманітності: лише за останні роки з'явились десятки спеціалізованих моделей, що по-різному вирішують задачу інтеграції структурних і біологічних даних. Темп публікацій у цій галузі різко прискорився після 2020 року, а методологічний ландшафт став надто фрагментованим для інтуїтивного орієнтування [15], [16]. У цьому контексті систематичний

огляд існуючих підходів є необхідною передумовою обґрунтованого вибору методів для власного дослідження. Логіка викладення відповідає хронологічному і концептуальному розвитку напряму – від перших нейромережевих моделей до складних мультимодальних архітектур.

1.4.1 Ранні підходи на основі нейронних мереж (DeepDDI)

Серед перших нейромережевих підходів до передбачення типів DDI особливе місце посідає модель DeepDDI [6]. Її ключовий внесок полягав не стільки в архітектурній складності, скільки в принциповому переформулюванні задачі: замість бінарної класифікації «взаємодія є або відсутня» було запропоновано 86-класову схему, де кожен клас відповідає стандартизованому типу взаємодії з онтології DrugBank. Це рішення стало методологічним орієнтиром для всіх наступних досліджень, оскільки вперше надало обчислювальним моделям можливість передбачати механізм і напрямок взаємодії, а не лише факт її наявності.

Архітектура DeepDDI є відносно простою за сучасними стандартами: послідовність повнозв'язних шарів з функціями активації ReLU і нормалізацією батчів. Вхідним представленням слугує профіль структурної схожості – Structural Similarity Profile (SSP) – вектор, що описує схожість даного препарату до кожного з препаратів у навчальній вибірці за метрикою Танімото, обчисленою на основі молекулярних відбитків похідних від SMILES. Таким чином, пара препаратів кодується конкатенацією двох SSP-векторів, що подається на вхід мережі для класифікації серед 86 типів. Подібний підхід до кодування хімічної схожості через профілі відносно навчальної вибірки дозволяє непрямо врахувати структурний контекст молекули без явної побудови графів [14].

Модель навчалась і тестувалась на даних DrugBank, що містив близько 192 000 пар препаратів з відомими типами взаємодій. Точність класифікації перевищила 92% у задачі розпізнавання типу взаємодії для

відомих пар, а модель також демонструє здатність генерувати текстові описи передбачуваних взаємодій через шаблонне відображення класів [6].

Разом з тим DeepDDI має два структурних обмеження, що стимулювали подальший розвиток галузі. По-перше, SSP є похідним від молекулярних відбитків, а не від безпосередньої топологічної структури молекули: нова молекула, що структурно відрізняється від усіх відомих препаратів, матиме малоінформативний SSP-профіль, що знижує якість передбачення для нових сполук. По-друге, модель не використовує жодної інформації з біологічного контексту: дані про спільні мішені, ферменти або транспортери, доступні в DrugBank, залишаються поза архітектурою. Саме ці обмеження – відсутність графового представлення молекул і відсутність контексту графу знань – задали вектор розвитку для наступного покоління моделей.

1.4.2 Моделі на основі графових нейромереж

Успіх DeepDDI у формалізації мультикласової задачі одночасно виявив її головне структурне обмеження: профіль структурної схожості не є повноцінним графовим представленням молекули. Наступне покоління моделей зосередилось саме на усуненні цього недоліку.

SmileGNN стала першою спробою систематично поєднати два типи молекулярної інформації в єдиній архітектурі [21]. Мотивація полягала у спостереженні, що SMILES-рядок і молекулярний граф кодують різні аспекти структури: рядкове представлення зберігає лінійну послідовність атомів, тоді як граф явно кодує топологію зв'язків і локальне хімічне оточення кожного атома. Архітектура перетворює SMILES-рядок на молекулярний граф, який передається на вхід GNN для отримання структурного ембеддінгу молекули. Отримані ембеддінги двох препаратів конкатенуються і передаються повнозв'язному класифікатору. Порівняно з підходами на основі фіксованих відбитків SmileGNN демонструє

покращення за метриками точності та F1, особливо помітне для типів взаємодій, де локальна хімічна структура є визначальною [21]. Водночас модель не залучає зовнішню біологічну інформацію про мішені або ферменти.

Паралельно було запропоновано альтернативний підхід до графового кодування молекул для задачі DDI [17]. Архітектура використовує молекулярні графи як єдине вхідне представлення: GNN агрегує ознаки атомів і хімічних зв'язків для отримання молекулярного ембеддінгу кожного препарату, після чого ембеддінги пари об'єднуються і передаються класифікатору. Ключова ідея полягає в тому, що графове представлення зберігає топологічний контекст молекули – інформацію про локальне хімічне оточення кожного атома, – яка втрачається при перетворенні структури у вектор відбитків. Результати на DrugBank демонструють конкурентні метрики порівняно з тогочасними підходами на основі відбитків [17]. Підходи на основі графового кодування молекул закладають методологічну основу для подальшої інтеграції механізмів пояснення у процес передбачення DDI [16].

Порівняння трьох моделей – DeepDDI, SmileGNN і підходу [17] – виявляє чітку траєкторію розвитку. DeepDDI сформулювала задачу як мультикласову класифікацію, але спиралась на представлення без явної топологічної структури. SmileGNN інтегрувала молекулярний граф поряд із рядковим каналом, тоді як підхід [17] повністю перейшов до графового представлення. Із кожним кроком модель отримує доступ до більш повної структурної інформації про молекулу. Разом з тим всі три підходи поділяють спільне принципове обмеження: жодна з них не залучає зовнішній біологічний контекст [15] – інформацію про спільні мішені, ферментні профілі або біологічні шляхи. Усунення цього обмеження через інтеграцію графів знань стало визначальним напрямом розвитку наступного покоління моделей.

1.4.3 Моделі з використанням гетерогенних графів та графів знань (TIGER, KnowDDI, ExDDI)

Усвідомлення того, що молекулярна структура є необхідною, але недостатньою умовою для передбачення типу DDI, стимулювало розробку архітектур, що систематично інтегрують біологічний контекст через гетерогенні графи знань. Три моделі, розглянуті в цьому підрозділі, представляють найбільш показові кроки цього переходу.

TIGER, представлений на конференції AAAI 2024, реалізує двоканальну архітектуру [18]. Перший канал обробляє молекулярну структуру препаратів через GNN, отримуючи структурні ембеддінги з SMILES-представлень. Другий канал оперує гетерогенним графом знань, що об'єднує вузли різних типів – препарати, білки-мішені, захворювання – і ребра з різною семантикою: мішені, ферменти, транспортери, показання. Ключовою архітектурною інновацією є трансформер з урахуванням типу відношення для гетерогенного графу: на відміну від стандартних GNN, що ігнорують тип ребра при агрегації, TIGER навчає окремі матриці уваги для кожного типу відношення. Це дозволяє моделі розрізняти, чи є зв'язок між препаратом і білком відношенням інгібування, чи індукції, чи субстратної взаємодії – семантична різниця, критична для передбачення типу DDI. Вихідні вектори обох каналів об'єднуються і передаються класифікатору. Модель навчалась на комбінації даних DrugBank і TWOSIDES, демонструючи macro F1 близько 0.92 на стандартних розбивках [18]. Обмеженням TIGER залишається відносно слабка інтерпретованість: двоканальна архітектура вказує на загальну важливість графу знань, але не ідентифікує конкретний біологічний шлях або білок, що відповідає за конкретне передбачення.

Саме цю прогалину адресує KnowDDI [19]. Центральна ідея полягає у переході від глобального графу знань до локальних контекстних підграфів: для кожної пари препаратів з графу знань автоматично виокремлюється

підграф, що містить найрелевантніші сутності і відношення між цими двома молекулами. Модель навчається не на всьому графі, а на цих спеціалізованих підграфах, що дозволяє концентрувати обчислювальні ресурси на інформативних частинах біологічної мережі. Контрастивний підхід до навчання посилює сигнал від значущих зв'язків і пригнічує шум від периферійних асоціацій. Це має особливе значення для рідкісних класів взаємодій: KnowDDI демонструє суттєво кращі метрики на рідкісних типах порівняно з TIGER, забезпечуючи більш рівномірний розподіл продуктивності між частими і рідкісними класами. Критичною перевагою є інтерпретованість: для кожного передбачення модель повертає відповідний підграф – конкретний набір білків, шляхів і відношень, що зумовили класифікацію. Ця властивість наближає KnowDDI до вимог практичного застосування, де верифікованість прогнозу є не менш важливою, ніж його точність [15]. Результати на DrugBank демонструють AUPR близько 0.91 [19].

ExDDI, представлена на AAAI 2025, робить якісно інший крок: замість структурної інтерпретованості через підграфи вона генерує пояснення природною мовою для кожного передбаченого типу взаємодії [22]. Архітектура поєднує GNN-кодування молекулярних структур і граф знань із генератором текстових пояснень, що синтезує опис механізму взаємодії у зрозумілій формі. Клінічна мотивація є очевидною: недостатньо отримати клас «тип 42 – підвищення концентрації»; необхідно зрозуміти, через який механізм відбувається взаємодія і яке практичне значення це має. ExDDI генерує саме такий текст, роблячи передбачення безпосередньо придатним для використання без додаткової інтерпретації. За метриками класифікації модель є конкурентоспроможною з TIGER і KnowDDI, додаючи унікальну можливість верифікації прогнозу через природномовне обґрунтування [22]. Цей напрям є одним із найперспективніших для подолання бар'єру між обчислювальними методами і реальним практичним застосуванням [16].

Траєкторія від DeepDDI до ExDDI відображає системний тренд галузі: від моделей, що видають клас без пояснення, до інтерпретованих систем, здатних обґрунтовувати свої передбачення на рівні, доступному для верифікації людиною.

1.4.4 Порівняльний аналіз існуючих моделей передбачення DDI

Систематичне порівняння розглянутих моделей потребує єдиної системи критеріїв. Для оцінки обрано шість параметрів: рік публікації як індикатор хронологічної позиції; тип архітектурного підходу; датасет для навчання і тестування; досягнуті метрики; кількість типів DDI у вихідному просторі як показник складності задачі; та ідентифіковані обмеження. Саме поєднання цих параметрів дозволяє побачити не ізольовані характеристики окремих моделей, а траєкторію розвитку напряму в цілому.

Особливо важливим є параметр кількості типів DDI: перехід від бінарної до 86-класової постановки є не кількісним, а якісним зсувом, що кардинально змінює практичну цінність передбачень. Не менш значущим є розрізнення між підходами, що використовують виключно молекулярну структуру, і тими, що інтегрують зовнішній біологічний контекст через графи знань. Порівняльні характеристики моделей узагальнено в таблиці 1.2.

Таблиця 1.2 – Порівняльний аналіз існуючих моделей передбачення DDI

Модель	Рік	Підхід	Датасет	Метрика	Типів DDI	Обмеження
DeepDDI	2018	DNN + SSP	DrugBank	>90% Acc	86	Без граф. структури
SmileGNN	2022	SMILES + GNN	DrugBank	>0.88 Acc	бінарна	Без графу знань
Nour et al.	2022	GNN features	DrugBank	>0.90 Acc	бінарна	Обмежений контекст

Продовження таблиці 1.2

Модель	Рік	Підхід	Датасет	Метрика	Типів DDI	Обмеження
TIGER	2024	Dual-channel HGT	DrugBank/ TWOSIDES	~0.92 macro F1	86	Складна архітектура
KnowDDI	2024	Subgraph KG	DrugBank	>0.90 AUPR	86	Залежність від якості KG
ExDDI	2025	GNN + NL explain	DrugBank	конкурентна з SOTA	86	Потребує NL анотацій

Таблиця 1.2 унаочнює кілька паралельних тенденцій. По-перше, простежується чіткий рух від бінарної постановки до мультикласової: якщо у 2022 році провідні GNN-моделі вирішували задачу наявності або відсутності взаємодії, то до 2024–2025 років стандартом стала 86-класова класифікація з ідентифікацією механізму. По-друге, архітектурна складність зростає нелінійно: від повнозв'язних шарів DeepDDI через однорівневі GNN до гетерогенних трансформерів і підграфових підходів. По-третє, галузь рухається від непрозорих моделей до інтерпретованих: KnowDDI повертає пояснювальний підграф, ExDDI генерує природномовне обґрунтування. Ці тенденції окреслюють простір вимог до перспективних архітектур і безпосередньо інформують методологічні рішення, прийняті у цій роботі.

1.4.5 Невирішені проблеми існуючих підходів

Аналіз моделей виявляє ряд системних проблем, що обмежують як наукову строгість існуючих підходів, так і їх практичну придатність. Ці проблеми не є артефактами окремих архітектурних рішень – вони

відображають фундаментальні протиріччя між складністю задачі і наявними методологічними інструментами.

Обмежена генералізація. Переважна більшість опублікованих моделей навчена і оцінена виключно на DrugBank, що породжує питання: чи відображають повідомлені метрики реальну здатність моделі до узагальнення? Стандартні випадкові розбивки на тренувальну і тестову вибірки не унеможливлюють витік інформації через спільні препарати, і моделі оцінені на таких розбивках часто суттєво погіршують результати при переході до зовнішніх баз або при оцінці на препаратах відсутніх у тренуванні [13]. Різні бази даних мають різні онтології взаємодій, і перенесення моделі між ними є нетривіальним завданням, яке рідко систематично досліджується [14].

Дисбаланс класів. У реальних даних DrugBank розподіл типів взаємодій є надзвичайно нерівномірним: найчастіші класи представлені десятками тисяч прикладів, тоді як клінічно критичні але рідкісні типи можуть налічувати кілька десятків або сотень зразків. Більшість GNN-архітектур, оптимізованих за стандартними метриками точності, систематично ігнорують рідкісні класи, досягаючи високих середніх показників за рахунок повного провалу на малопредставлених категоріях [15]. Парадоксально, саме ці рідкісні взаємодії є найбільш небезпечними і найбільш потребують надійного передбачення.

Холодний старт. Моделі, що спираються на ембеддінги вузлів у графі знань, принципово не можуть обробляти препарати, відсутні у цьому графі під час навчання. Для нового лікарського засобу, що лише виходить на ринок і не має задокументованих взаємодій, такі моделі не здатні генерувати прогноз. Холодний старт є одним із найменш вирішених викликів сучасного DDI-прогнозування: лише моделі, що спираються виключно на структурні ознаки молекули, природно стійкі до цієї проблеми [16].

Інтерпретованість. Поява ExDDI з генерацією природномовних пояснень є важливим кроком, однак переважна більшість сучасних моделей,

включаючи TIGER, функціонують як чорні скриньки: вони видають клас взаємодії без жодного обґрунтування механізму [22]. Практичне впровадження систем підтримки прийняття рішень у фармакотерапії вимагає не лише точних, але й верифікованих і пояснюваних прогнозів – вимога, якій більшість сучасних архітектур не відповідає.

Неповна мультимодальна інтеграція. Більшість підходів об'єднують два джерела інформації – як правило, молекулярну структуру і граф знань – однак рідко в одній архітектурі систематично інтегруються всі три рівні: структура молекули, біологічний контекст і текстові описи механізмів взаємодій. Текстові дані DrugBank несуть значний інформаційний потенціал, який залишається здебільшого невикористаним у графових підходах [14].

Сукупність цих проблем формує чіткий дослідницький запит: необхідна архітектура, що поєднує структурне кодування молекул, інтеграцію біологічного контексту і стійкість до холодного старту, при цьому зберігаючи здатність до узагальнення на нові препарати і рідкісні класи взаємодій. Саме ці вимоги визначають постановку задачі цього дослідження.

1.5 Постановка задачі дослідження

Проведений аналіз засвідчує, що галузь передбачення DDI пройшла шлях від бінарної класифікації на основі молекулярних відбитків до складних мультимодальних архітектур, що інтегрують структурні та біологічні дані через графові представлення. Разом з тим систематичний огляд виявив п'ять відкритих проблем – обмежену генералізацію, дисбаланс класів, уразливість до холодного старту, недостатню інтерпретованість і неповну мультимодальну інтеграцію, – що свідчить про наявність суттєвого простору для вдосконалення існуючих підходів.

Об'єктом цього дослідження є процес передбачення типів взаємодій лікарських препаратів на основі структурних та фармакологічних властивостей препаратів. Предметом дослідження є методи та моделі на основі графових нейронних мереж для мультикласової класифікації типів DDI.

Метою роботи є розробка та дослідження моделі на основі GNN для передбачення типів DDI з підвищеними показниками класифікації порівняно з існуючими базовими підходами, зокрема в умовах холодного старту та нерівномірного розподілу класів.

Для досягнення поставленої мети визначено такі завдання:

– проаналізувати існуючі підходи до передбачення DDI та визначити їхні обмеження;

– здійснити вибір та попередню обробку датасету для навчання моделі;

– розробити архітектуру GNN-моделі з урахуванням виявлених прогалин існуючих підходів;

– навчити модель та виконати підбір гіперпараметрів;

– провести порівняльну оцінку з базовими моделями за стандартними метриками (F1-macro, Accuracy, AUROC);

– проаналізувати результати та дослідити інтерпретованість передбачень.

Вибір графових нейронних мереж як центрального методологічного інструменту є обґрунтованим на кількох рівнях.

По-перше, молекула є за своєю природою графовим об'єктом, і GNN дозволяють безпосередньо оперувати її топологічною структурою: ітеративна агрегація ознак сусідніх атомів відтворює хімічну інтуїцію без інформаційно збиткового перетворення у вектор фіксованої довжини. Графове кодування молекул дає стабільне покращення порівняно з підходами на основі відбитків у задачах мультикласової класифікації DDI [17], [19].

По-друге, GNN природно масштабуються на гетерогенні графи знань, де вузли різних типів і ребра різної семантики несуть взаємодоповнюючу біологічну інформацію. Агрегація з урахуванням типу відношення в гетерогенному графі дозволяє моделі розрізняти фармакокінетичні і фармакодинамічні механізми взаємодій – диференціація, недоступна для підходів що ігнорують тип ребра [18], [15]. Ця властивість є критичною для задачі передбачення конкретного типу DDI серед 86 або більше класів.

Альтернативні підходи поступаються GNN за обома критеріями: трансформери потребують значно більших обчислювальних ресурсів на фармакологічних датасетах, а класичне машинне навчання (підрозділ 1.3.2) не забезпечує мультикласову класифікацію без суттєвої втрати структурної інформації. GNN-архітектури стабільно перевершують обидва класи методів на стандартних DDI-бенчмарках [16].

Додатковим аргументом на користь GNN є їх природна стійкість до проблеми холодного старту при використанні молекулярних графів як основного представлення: нова молекула, що не була присутня у тренувальній вибірці, може бути закодована через свою структуру без залежності від наявності у графі знань. Це є принциповою перевагою перед підходами, що спираються виключно на ембеддінги вузлів графу знань [13]. Формалізація задачі як 86-класової класифікації, запропонована в роботі з DeepDDI і прийнята як стандарт наступними моделями, забезпечує необхідну деталізацію передбачень для практичного застосування [6]. Інтеграція підграфового навчання, розробленого в KnowDDI, відкриває додаткові можливості для підвищення якості на рідкісних класах і забезпечення інтерпретованості результатів [19].

Теоретичне обґрунтування і результати систематичного огляду, представлені в цьому розділі, формують методологічну основу для розробки власної архітектури, опис якої, разом із деталями реалізації, навчання і порівняльної оцінки, становить зміст наступних розділів роботи.

2 МЕТОДИ ТА ЗАСОБИ РОЗВ'ЯЗАННЯ ЗАДАЧІ

2.1 Формальна постановка задачі прогнозування лікарських взаємодій

Задача передбачення взаємодій лікарських препаратів (DDI) у більшості існуючих робіт формалізується як задача класифікації на парах препаратів. У ранніх підходах, зокрема DeepDDI [6], DDI зводилась до бінарної або багатокласової класифікації: кожній парі (drugA, drugB) ставиться у відповідність один або кілька дискретних типів взаємодії з фіксованого набору шаблонів. Підходи на основі графових нейромереж, такі як MHCADDI та SSI-DDI, використовують молекулярний граф як вхідне представлення та навчають парне кодування через механізм уваги між атомами двох молекул. Роботи, що залучають гетерогенні графи знань (TIGER, KnowDDI), формалізують задачу як передбачення зв'язків у графі. Спільним обмеженням цих підходів є або бінарна постановка (взаємодіють / не взаємодіють), або класифікація з фіксованою таксономією ефектів без урахування їх ієрархічної природи. У цьому дослідженні задача формалізується як ієрархічна класифікація з явним розрізненням рівня фізіологічної системи та рівня конкретного клінічного ефекту, що дозволяє моделювати структуру простору міток, закладену в клінічних описах DrugBank 6.0.

Кожен препарат у цьому дослідженні представлений молекулярним графом, що позначається Gi. Граф будується зі рядка SMILES засобами бібліотеки RDKit і має вигляд:

$$\mathcal{G}_i = (V_i,\ E_i,\ x_i), \tag{2.1}$$

де V_i – множина атомів-вузлів;

E_i – множина хімічних зв'язків-ребер;

x_i – матриця атомних ознак розмірністю |Vi| на 5.

Як ознаки кожного атома використовуються: атомний номер, ступінь вузла, формальний заряд, кількість неспарених електронів та ознака ароматичності. Саме ця структура є вхідними даними для GIN-енкодера, описаного у підрозділі 2.2.

Окрім молекулярного графа, кожен препарат асоційований із множиною білкових мішеней, яку позначають Pi. Кожен елемент цієї множини є трійкою (pk, τk, αk), де pk – векторне представлення поліпептиду розмірністю 1280, отримане з моделі ESM-2 650M, τk – тип взаємодії (мішень, фермент, транспортер або носій), αk – механізм дії (інгібування, індукування, субстрат тощо). Ця множина утворює «хмару білків» препарату і є вхідними даними для білкового енкодера у підрозділі 2.3.

Простір міток задачі побудований безпосередньо з текстових описів DrugBank 6.0 і містить 180 унікальних клінічних ефектів, згрупованих у 15 фізіологічних систем. Підхід DeepDDI [6] формує 86 шаблонів речень з DrugBank 5 на основі синтаксичної онтологізації тексту. У цій роботі використовується DrugBank 6.0 з більш деталізованими описами, і таксономія побудована незалежно: вона відповідає клінічним ефектам (наприклад, qtc_prolongation, rhabdomyolysis), а не синтаксичним конструкціям. Тому числа 86 і 180 не є суперечністю – це результати двох різних методологій анотування на різних версіях бази.

Кожна анотація взаємодії між парою препаратів є трійкою

$$y = (s, e, \delta), \tag{2.2}$$

де s – фізіологічна система;

e – конкретний ефект у межах цієї системи;

δ − напрям зміни (збільшення або зменшення).

Одна пара препаратів може мати кілька таких анотацій: у датасеті 1 161 893 унікальних пари породжують 1 727 888 записів про

ефекти, тобто в середньому 1.49 анотації на пару. Ієрархічна природа простору міток – спочатку система, потім ефект – безпосередньо мотивує каскадну архітектуру, описану у підрозділі 2.4.

Протокол оцінювання у цій роботі відповідає сценарію cold-start: модель повинна коректно передбачати взаємодії для молекул, яких не було в тренувальній вибірці. Розбиття виконується на рівні препаратів – множини препаратів у тренувальному, валідаційному та тестовому наборах не перетинаються між собою. Пара препаратів належить до строгого cold-start підмножини тоді і тільки тоді, коли обидва препарати відсутні у тренувальному наборі.

Розбиття має розмір 2199 / 733 / 733 препарати. У тестовій вибірці 144 967 пар (34.7%) відповідають умові cold-start; решта 273 067 пар належать до semi-cold сценарію, де один препарат із пари присутній у тренувальних даних. Саме cold-start підмножина є основним бенчмарком цієї роботи, оскільки моделює реалістичний сценарій скринінгу нових сполук.

2.2 Методи представлення молекулярних структур на основі графових нейронних мереж

Вибір методу представлення молекулярної структури безпосередньо впливає на якість DDI-моделей. Найраніші підходи, наприклад DeepDDI [6], кодують молекули як вектори хімічних підструктур (ECFP / Morgan fingerprints) або одномірні дескриптори хімічних властивостей. Ці представлення прості у вичисленні, але фіксують довжину бітового вектора наперед і не адаптуються до задачі. В роботах на основі GNN – MHCADDI, SSI-DDI – для молекулярного кодування застосовуються графові згорткові мережі (GCN) або мережі з механізмом уваги (GAT), що дають гнучкіші векторні представлення, але використовують нормалізоване усереднення сусідніх векторів. У цьому дослідженні для кодування

молекулярних структур обрано Graph Isomorphism Network (GIN), оскільки серед агрегуючих мереж вона досягає максимальної виразної сили для розрізнення графів без втрати інформації від нормалізації.

Молекула як рядок SMILES містить усю необхідну структурну інформацію (рисунок 2.1), але у формі, незручній для нейронної мережі: однаковий атомний склад може записуватись різними рядками, порядок символів не відповідає топології молекули, а локальне оточення кожного атома не виражене явно. Альтернативою є векторні відбитки – наприклад, кругові відбитки Моргана (ECFP), які кодують підструктури фіксованим бітовим вектором. Проте відбитки мають суттєвий недолік: їх розмірність фіксована наперед, а довжина хімічного контексту, що враховується, визначається радіусом і не може адаптуватись під час навчання. Графові нейронні мережі (GNN) позбавлені обох цих обмежень.

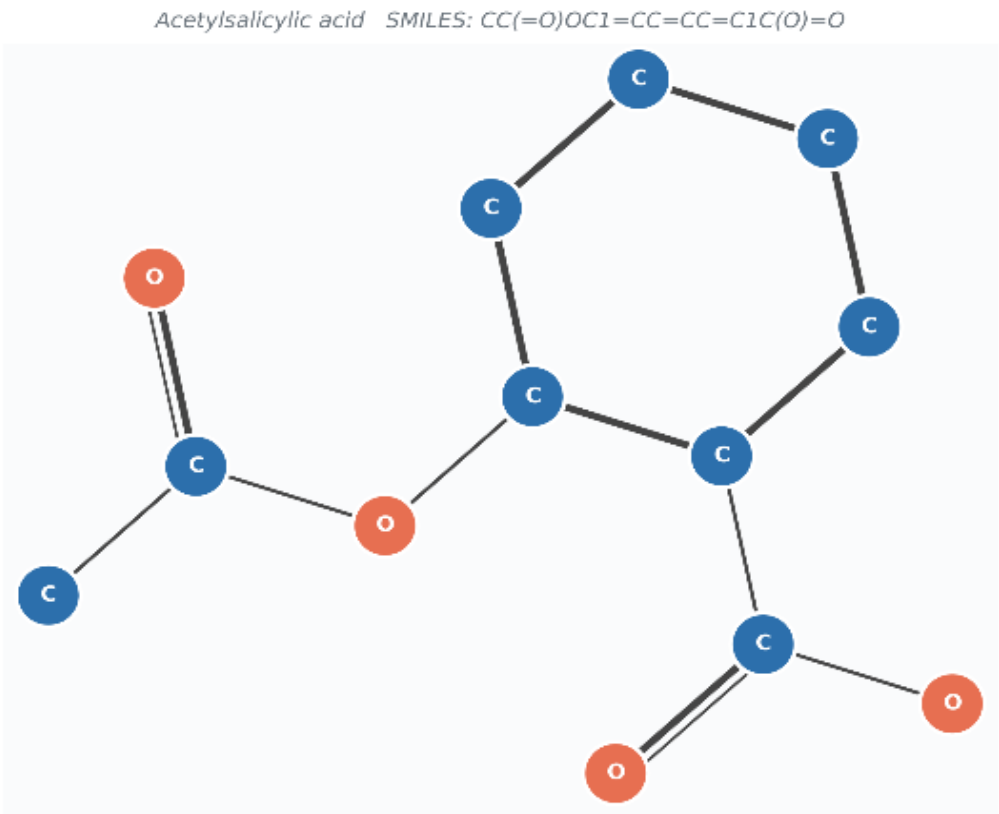


Рисунок 2.1 – Молекулярний граф для аспірину (ASA). Вузли – атоми, ребра – хімічні зв'язки

У цій роботі для кодування молекулярних структур використовується Graph Isomorphism Network (GIN). GIN обрано з теоретичних міркувань: серед моделей, що агрегують інформацію від сусідів через підсумовування, GIN досягає максимальної виразної сили – вона настільки ж потужна, як тест Вайсфайлера-Лемана для розрізнення графів. Графові згорткові мережі (GCN) і мережі з механізмом уваги (GAT) використовують нормалізоване усереднення, що теоретично слабше: деякі структурно різні підграфи можуть давати однакові представлення. Правило оновлення GIN для вузла v на шарі k записується як:

$$h_v^{(k)} = \mathrm{MLP}^{(k)}!\left(\left(1+\varepsilon^{(k)}\right)h_v^{(k-1)} + \sum_{u\in\mathcal{N}(v)} h_u^{(k-1)}\right), \qquad (2.3)$$

де N(v) – множина сусідів вузла v;

MLP – двошаровий перцептрон;

ε – параметр, що навчається разом з рештою мережі.

Таке правило зберігає внесок самого вузла через суму без нормалізації, що і забезпечує вищу виразну силу порівняно з нормалізованим усередненням.

Реалізована архітектура містить три послідовні шари GINConv. Перший шар проектує атомні ознаки з розмірності 5 у прихований простір розмірності 256, наступні два шари діють у тому ж просторі. Після кожного шару застосовується пакетна нормалізація, функція активації ReLU та Dropout зі значенням 0.1. Як атомні ознаки використовуються п'ять дескрипторів: атомний номер, ступінь вузла у графі, формальний заряд, кількість неспарених електронів та ознака ароматичності. Ці ознаки повністю описують локальну хімію кожного атома і не потребують додаткових розрахунків поза RDKit.

Після трьох шарів передавання повідомлень атомні вектори об'єднуються у вектор молекули за допомогою global mean pooling – усереднення по всіх атомах:

$$m_i = \frac{1}{|V_i|}\sum_{v \in V_i} h_v^{(3)}, \quad (2.4)$$

де m_i – вектор молекули i;

$|V_i|$ – кількість атомів;

$h_v^{(3)}$ – вектор вузла v після третього шару GIN.

Результат – вектор розмірності 256, що представляє молекулу у прихованому просторі. Загальна кількість параметрів GIN-енкодера становить 332 032.

У модифікованих версіях каскаду (PKEffectModel та EffectSignalsModel) базова архітектура розширена двома способами. По-перше, замість одного global mean pool застосовується паралельне об'єднання mean- і max-пулінгу, що дає вектор 512-d, який потім проектується лінійним шаром назад до 256-d. Max-пулінг зберігає інформацію про атоми з найбільш вираженими ознаками і в задачах з рідкісними функціональними групами доповнює усереднення. По-друге, до виходу GIN додається проекція кругового відбитка Моргана (ECFP4, 2,048 бітів):

$$m_i = f_{\text{GIN}}(\mathcal{G}_i) + f_{\text{proj}}(\text{ECFP4}_\text{i}), \quad (2.5)$$

де f_{proj} – лінійний шар з LayerNorm.

Такий residual-зв'язок дозволяє моделі явно спиратись на підструктурні відбитки, якщо GIN не встиг узагальнити потрібний паттерн за три шари. Це особливо актуально для рідкісних функціональних груп, де тренувальних прикладів замало для стабільного навчання GIN.

Вибір трьох шарів обґрунтований практично: збільшення глибини GNN понад три-чотири шари у задачах передбачення молекулярних властивостей регулярно спричиняє ефект надмірного згладжування, за якого вузли в різних частинах графа отримують схожі представлення. Для

типових малих молекул DrugBank – з медіанним діаметром графа у 5–7 зв'язків – три шари достатні, щоб кожен атом «побачив» своє хімічне оточення на відстані трьох зв'язків, що покриває більшість фармакофорних груп.

2.3 Методи представлення білкових мішеней

Фармакологічна активність малої молекули визначається її взаємодією з білковими мішенями – рецепторами, ферментами, транспортерами та носіями. Якщо два препарати конкурують за одну й ту саму мішень або обидва метаболізуються одним ферментом родини CYP450, їх взаємодія майже неминуча. Тому білкове оточення кожного препарату є природним предиктором DDI і повинно бути закодованим у векторну форму, доступну для нейронної мережі.

Існуючі підходи до кодування білкового оточення препарату в DDI-моделях поділяються на кілька груп. Перша – прості одноразові ембеддинги на основі назв мішеней: модель зберігає лише ID білка з DrugBank і навчає навчальний ембеддинг з нуля. Цей підхід, використаний, зокрема, у TIGER, не потребує послідовності білка, але повністю нестійкий до нових білків: для білка, якого немає в тренувальній вибірці, модель не може побудувати повноцінного представлення. Друга група – фізико-хімічні дескриптори амінокислотного складу, наприклад частоти амінокислот або профілі двоточної структури – вони несуть структурну інформацію, але не дозволяють моделі враховувати довжині контексту взаємодії. Третья група – Transformer-енкодери, навчені з нуля на послідовностях корпусу DDI, що не дозволяє охопити еволюційно-консервативні паттерни. У цьому дослідженні використовуються векторні представлення з попередньо навченої білкової мовної моделі ESM-2, що поєднує переваги Transformer-архітектури з навчанням на величезному білковому корпусі без необхідності додаткового навчання.

Пряме кодування амінокислотних послідовностей через навчання з нуля у цій задачі є нежиттєздатним. Датасет містить 5 367 унікальних поліпептидів із медіанною довжиною 466 амінокислот і максимумом 14 774. Transformer-енкодер, навчений лише на цьому корпусі, не зміг би захопити еволюційно-консервативні паттерни, що визначають функцію білка: для цього потрібні мільйони різноманітних послідовностей. Тому в роботі використовується попередньо навчена білкова мовна модель ESM-2 з 650M параметрів, яка тренувалась на базі UniRef50 – понад 250 мільйонів унікальних послідовностей. ESM-2 вивчає відновлення замаскованих амінокислот і у результаті кодує в своїх прихованих станах функціональну та структурну інформацію без явного завдання 3D-координат. Це дозволяє отримати якісні векторні представлення навіть для білків, кристалічна структура яких невідома.

Оскільки виклик ESM-2 під час кожного навчального кроку є обчислювально надмірним, усі 5 367 ембеддингів попередньо обчислені і збережені у базі даних PostgreSQL. Для кожного поліпептиду ESM-2 повертає матрицю прихованих станів розмірності L на 1,280, де L – довжина послідовності. Ця матриця усереднюється по позиціях, і результуючий вектор розмірності 1,280 зберігається як представлення білка. Під час навчання модель звертається до бази і отримує готові вектори без жодного звернення до ESM-2.

Кожен препарат асоційований не з одним білком, а з множиною P_i, яку у роботі називають «хмарою білків». У датасеті зафіксовано 36 576 зв'язків між препаратами та поліпептидами: 70.4% – мішені, 17.2% – ферменти, 9.8% – транспортери, 2.9% – носії. Середній розмір хмари становить 3.82 білки на препарат при медіані 1 і максимумі 309. Така скошеність означає, що більшість препаратів має одну-дві документовані мішені, тоді як широкоспектральні інгібітори CYP можуть взаємодіяти з сотнями білків.

Для агрегації інформації з хмари в єдиний вектор застосовується механізм зваженого пулінгу з увагою. Спочатку кожен ESM-2 вектор проектується лінійним шаром з LayerNorm у простір розмірності 256. До цього проекційного вектора приєднуються два навчальні ембеддинги: тип взаємодії (target / enzyme / transporter / carrier, розмірність 32) та механізм дії (інгібування, індукування, субстрат тощо, 55 варіантів, розмірність 32). Конкатенація дає вектор розмірності 320 для кожного білка. На основі цього вектора обчислюється скалярна оцінка уваги через дворівневий блок з активацією Tanh, і оцінки нормалізуються softmax по всіх білках у хмарі. Підсумкове представлення хмари отримується як зважена сума:

$$c_i = \sum_{k=1}^{|\mathcal{P}_i|} \alpha_k \, v_k, \tag{2.6}$$

де α_k – вага уваги k-го білка,

v_k – його проекційний вектор після перетворення.

Якщо у препарату немає жодного задокументованого білка, замість хмари підставляється навчальний вектор-замінник розмірності 256. Фінальний вектор сі разом із молекулярним вектором mi (підрозділ 2.2) утворює повне представлення препарату.

Увага до типу взаємодії та механізму дії є ключовим архітектурним рішенням. Модель, що агрегує білкову хмару простим усередненням, не розрізняє, чи є білок мішенню або ферментом-метаболізатором – а для передбачення фармакокінетичних ефектів ця різниця принципова. Механізм уваги дозволяє мережі самостійно навчитись надавати більшої ваги ферментам при передбаченні метаболічних взаємодій і мішеням при передбаченні фармакодинамічних.

У більш складних варіантах каскаду – PKEffectModel і EffectSignalsModel – архітектура білкового енкодера розширена до двох рівнів уваги. На першому рівні застосовується self-attention у межах хмари одного препарату (2 шари, 12 голів, прихований вимір 768), що дозволяє

білкам у хмарі взаємодіяти між собою перед агрегацією. На другому рівні – cross-attention між хмарами двох препаратів пари: вектори препарату A запитують ключі та значення препарату B і навпаки. Така двостороння взаємодія явно моделює ситуацію, коли фермент одного препарату безпосередньо релевантний для мішеней іншого – що і є механізмом більшості фармакокінетичних DDI.

2.4 Каскадна архітектура класифікації ефектів взаємодій

Найчастіше в існуючих DDI-моделях простір ефектів обробляється трьома способами. Перший – flat multilabel: модель передбачає всі ефекти одночасно без будь-якої ієрархії; цей підхід простий, але змушує модель одночасно вирішувати задачі різного рівня абстракції з однаковою вагою. Другий – двоетапна класифікація: спочатку визначається, чи є взаємодія (бінарний класифікатор), а потім розрізняється тип; такий підхід використовуються у DeepDDI і MHCADDI, але не враховує фізіологічну організацію ефектів і діє незалежно для кожної системи. Третій – повністю ієрархічні підходи, де кожний рівень ієрархії обробляється окремим класифікатором; прикладом є KnowDDI з дворівневим представленням ефекту. У цьому дослідженні пропонується каскадна архітектура з трьох незалежно навчених компонентів, що розвиває ідею ієрархічного підходу і усуває його обмеження: кожний компонент оптимізується під свою підзадачу з своєю функцією втрати без конкуренції за спільні ваги.

Пряма класифікація DDI-ефекту як задача на 180 класів має системну проблему: розподіл класів надзвичайно нерівномірний. Три найчастіших ефекти – concentration, excretion та metabolism – покривають 41.5% усіх міток, тоді як 80 з 180 класів мають менше 100 прикладів кожен. Більшість цих рідкісних ефектів належать до різних фізіологічних систем і мають різну природу: фармакокінетичні ефекти описують зміни концентрації та метаболізму, фармакодинамічні – безпосередній вплив на рецептори та

органи. Об'єднання всіх 180 класів в один класифікатор змушує модель одночасно вирішувати задачі різного рівня абстракції, що ускладнює навчання і знижує якість на рідкісних класах.

Для розв'язання цієї проблеми у роботі реалізована каскадна архітектура з трьох послідовних компонентів, які ієрархічно звужують простір передбачень. Загальна схема показана на рисунку 2.2.

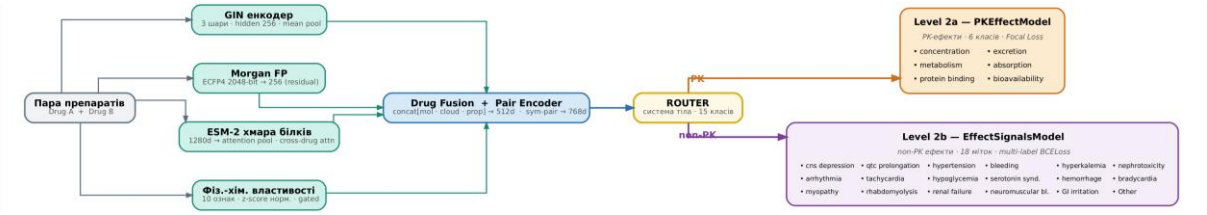


Рисунок 2.2 – Каскадна архітектура мультимодальної моделі прогнозування DDI

Пара препаратів кодується чотирма паралельними гілками (GIN-енкодер, Morgan FP, ESM-2 хмара білків, фіз.-хім. властивості), об'єднується у спільний вектор пари та подається на вхід Router. Залежно від передбаченої системи Router направляє передбачення до PKEffectModel (6 класів) або EffectSignalsModel (18 міток).

Перший компонент – Router – класифікує взаємодію за фізіологічною системою на 15 класів. PK-система є найбільшою і займає 42.4% усіх міток, тому бінарне рішення «PK або не-PK» є центральним розгалуженням каскаду. Router реалізований як дворівневий класифікатор: лінійний шар з 768 до 384 нейронів з активацією ReLU та Dropout, після чого вихідний лінійний шар дає логіти для 15 класів. Загалом Router містить 301 071 параметр.

Вхід Router – вектор пари препаратів розмірності 768, який формується через Drug Fusion і Pair Encoder. Кожен препарат окремо представлений конкатенацією молекулярного вектора m_i (256d, GIN),

вектора хмари білків c_i (256d, attention pooling) та вектора фізико-хімічних властивостей (64d, 10 ознак після z-score нормалізації та гейтованої проекції). Конкатенація дає вектор 512d на препарат. Вектор пари формується симетрично:

$$p_{ij} = \text{PairEncoder!}\left([d_i|d_j] + [d_j|d_i]\right), \quad (2.7)$$

де di і dj – вектори двох препаратів, а симетрична агрегація гарантує, що передбачення не залежить від порядку препаратів у парі.

PairEncoder проектує об'єднаний вектор у простір 768d і містить 1 774 080 параметрів.

Другий компонент – PKEffectModel (рівень 2a) – активується, якщо Router передбачив PK-систему. Він вирішує задачу 6-класової класифікації: concentration, excretion, metabolism, absorption, protein_binding та bioavailability. Ці шість ефектів охоплюють усі фармакокінетичні механізми та разом становлять 732 118 міток (42.4% датасету). Модель навчається незалежно від Router за допомогою Focal Loss з параметром gamma = 2.0, що знижує відносний внесок добре класифікованих прикладів і примушує модель зосереджуватись на важчих випадках – зокрема на рідкісних ефектах absorption і protein_binding. На тестовій вибірці PKEffectModel досягає 61.5% accuracy та 55.1% macro F1, що суттєво перевищує baseline більшості класу (частка metabolism серед PK = 43.8%).

Третій компонент – EffectSignalsModel (рівень 2b) – активується для всіх не-PK взаємодій і вирішує задачу multi-label класифікації на 18 цільових ефектів: cns_depression, qtc_prolongation, hypertension, bleeding, hyperkalemia, nephrotoxicity, arrhythmia, tachycardia, hypoglycemia, serotonin_syndrome, hemorrhage, myopathy, rhabdomyolysis, renal_failure, neuromuscular_blockade, gastrointestinal_irritation, bradycardia та узагальнений клас Other. Multi-label формулювання є обґрунтованим: одна взаємодія може одночасно спричиняти, наприклад, і гіпотонію, і

брадикардію. Функція втрат – BCEWithLogitsLoss з label smoothing 0.05, де позитивні мітки замінюються на 0.95, а негативні – на 0.05. Таке згладжування запобігає надмірній впевненості моделі на рідкісних позитивних прикладах.

Окремим навчанням кожного компонента каскаду досягається кілька переваг. По-перше, кожна модель оптимізується під свою підзадачу зі своєю функцією втрат і своїми гіперпараметрами. По-друге, помилки Router не розповсюджуються у зворотному напрямку на Level 2 – кожен рівень отримує чисті градієнти від своєї мети. По-третє, каскадна схема природно масштабується: якщо в майбутньому з'являться нові рідкісні ефекти, достатньо донавчити відповідний компонент Level 2.

Загальна кількість параметрів усієї ієрархічної моделі становить 3 616 544. Попри значний розмір простору міток архітектура залишається компактною завдяки поділу задачі між спеціалізованими компонентами.

2.5 Методи навчання моделі в умовах дисбалансу класів

Задачі класифікації з дисбалансом класів постійно виникають у біомедичних застосунках машинного навчання. Focal Loss, запропонована для детекції об'єктів на зображеннях з різким дисбалансом класів, згодом адаптована до біомедичних задач, у тому числі до прогнозування мікрбіомх. Stochastic Weight Averaging (SWA) показала покращення узагальнення в різних задачах класифікації, включаючи незбалансовані. EMA-усереднення ваг широко використовується в сучасних великих моделях (наприклад, BYOL та mean teacher) як стабілізатор навчання. У цьому дослідженні ці методи застосовуються спільно як комплементарна система протидії дисбалансу, адаптована до специфіки задачі передбачення DDI.

Дисбаланс класів у датасеті є структурною властивістю самих даних, а не артефактом збору. Відношення частоти найпоширенішого ефекту до

медіанного становить приблизно 2 010, а 80 із 180 ефектів мають менше 100 прикладів. Розподіл топ-30 ефектів наведено на рисунку 2.3 – логарифмічна шкала підкреслює, що різниця між домінуючими та рідкісними класами сягає кількох порядків.

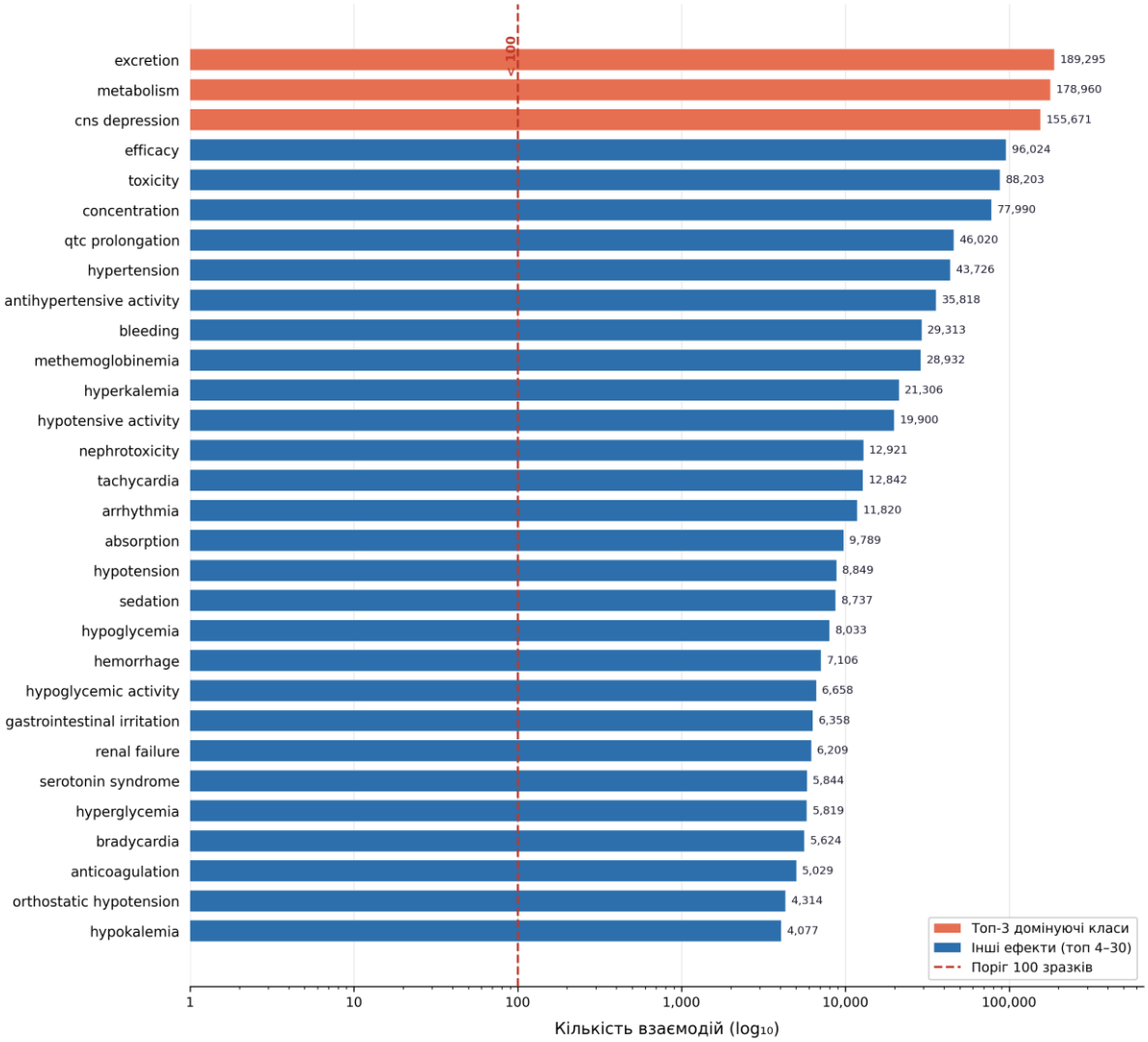


Рисунок 2.3 – Розподіл кількості взаємодій для топ-30 ефектів (логарифмічна шкала)

Пунктирна лінія позначає поріг 100 зразків. Три домінуючих ефекти (excretion, metabolism, cns_depression) виділені окремим кольором.

За такого дисбалансу стандартна крос-ентропія дає модель, що конкурентно передбачає лише домінуючі класи: мінімум функції втрат

досяжний без жодного правильного передбачення для рідкісних ефектів. У роботі застосовано три незалежних механізми протидії.

Перший – Focal Loss із параметром gamma = 2.0. Focal Loss змінює внесок кожного прикладу у загальну втрату залежно від поточної впевненості моделі у правильному класі. Для прикладу з ймовірністю правильного класу p_t функція втрат набуває вигляду

$$L = -(1 - p_t)^{\gamma} \log(p_t), \tag{2.8}$$

де pt – ймовірність правильного класу;

γ – параметр фокусування ($\gamma = 2.0$).

При gamma = 2.0 приклади, які модель вже класифікує впевнено (p_t близько до 1), отримують ваговий множник близький до нуля і практично не впливають на градієнт. Натомість важкі та рідкісні приклади з малим p_t зберігають повний внесок. Це дозволяє уникнути ситуації, коли градієнт домінується масовим класом.

Другий механізм – зважений семплер. На кожній епосі зразки обираються з ймовірністю, обернено пропорційною кореню частоти їх класу (sqrt_inverse weighting). Замість того щоб модель бачила тисячі прикладів excretion на кожен приклад skeletal_muscle_relaxation, семплер вирівнює ефективні частоти класів у межах батчу. Це доповнює Focal Loss: семплер впливає на те, які приклади потрапляють у батч, а Focal Loss – на те, скільки кожен з них важить у втраті.

Третій механізм – усереднення ваг моделі. Після закінчення основного навчання застосовуються дві техніки: Exponential Moving Average (EMA) з коефіцієнтом 0.999, що підтримує ковзне середнє ваг протягом усього навчання, та Stochastic Weight Averaging (SWA), що усереднює знімки ваг з епох 60–80. Усереднення ваг відоме як метод покращення узагальнення без додаткових даних: усереднена модель

знаходиться в ширшій плоскій мінімізуючій ямі функції втрат і менш схильна до підгонки під домінуючі класи пізніх епох навчання.

Оптимізатор – AdamW з weight decay 5e-4 і початковою швидкістю навчання 5e-5. Перші 5 епох – лінійний прогрів від нуля до номінальної швидкості (warmup), після чого планувальник ReduceLROnPlateau зменшує швидкість навчання вдвічі при відсутності покращення валідаційної втрати протягом 7 епох поспіль. Клірінг норми градієнта обмежений значенням 5.0, що запобігає стрибкам під час навчання на незбалансованих батчах.

Чекпоінт зберігається за мінімальною валідаційною втратою, а не за точністю. Збереження за точністю є систематичною помилкою в умовах дисбалансу: модель може демонструвати зростання accuracy за рахунок кращого запам'ятовування домінуючих класів при одночасній деградації на рідкісних. Валідаційна втрата (Focal Loss) зважає на всі класи і є надійнішим сигналом для вибору чекпоінту.

Протокол оцінювання відповідає cold-start сценарію, описаному у підрозділі 2.1. Основна метрика – macro F1, що усереднює F1 по всіх класах без зважування на їх частоту і тому рівноправно враховує рідкісні та домінуючі ефекти. Accuracy як метрика у цій задачі є оманливою: majority-class baseline дає 42.4% accuracy (якщо завжди передбачати PK-систему) та 17.4% (якщо завжди передбачати concentration). Тому точність наводиться як допоміжна метрика поряд з macro F1 та, для multi-label компонента, з Macro AUC і Average Precision.

Для EffectSignalsModel (Level 2b) поріг бінаризації виходів встановлюється окремо на валідаційній вибірці за максимумом macro F1 – а не фіксується за замовчуванням 0.5. Це суттєво: оптимальний поріг для рідкісних міток часто нижчий за 0.5, оскільки модель схильна занижувати свою впевненість у позитивних прикладах рідкісних класів.

3 РОЗРОБКА ТА РЕАЛІЗАЦІЯ СИСТЕМИ

3.1 Джерела даних та попередня обробка

Основним джерелом даних є база DrugBank 6.0 у форматі XML-дампу розміром 1.5 ГБ. База містить структуровану інформацію про препарати, їх фізико-хімічні властивості, білкові мішені та задокументовані двосторонні взаємодії з текстовими описами механізмів. Для парсингу використовується потокова обробка через бібліотеку lxml (streaming XML parsing), що дозволяє обробляти файл без повного завантаження у пам'ять.

З кожного запису препарату витягуються такі поля: первинний ідентифікатор DrugBank (формату DB#####), тип препарату (small molecule або biotech), рядок SMILES із блоку calculated-properties, фізико-хімічні властивості (logP, PSA, кількість акцепторів та донорів водневих зв'язків, молекулярна вага, кількість ротабельних зв'язків, pKa кислотний і основний, рефрактивність), список взаємодій з ідентифікаторами партнерів та текстовими описами, а також пов'язані поліпептиди з типом взаємодії (target, enzyme, transporter, carrier) і механізмом дії.

Фільтрація відбувається на двох рівнях. На рівні препаратів залишаються лише small molecule з непустим рядком SMILES: з 17 430 загальних записів у базі 13 166 є малими молекулами, з яких 12 313 мають задокументований SMILES. Решта 853 препарати виключаються, оскільки GIN-енкодер не може обробити структуру без молекулярного графа. На рівні взаємодій залишаються лише відфільтровані пари препаратів.

Текстові описи взаємодій перетворюються на структуровані мітки окремим етапом NLP-класифікації. Кожен опис аналізується за шаблонами, що відповідають клінічним ефектам, і маппується у трійку (система органів, ефект, напрям зміни). Результатом є 1 727 888 записів у таблиці drug_interactions_effects, тобто в середньому 1.49 анотації на пару. Унікальних ефектів виявлено 180, згрупованих у 15 фізіологічних систем.

Поліпептиди зберігаються окремо: 5 367 унікальних білків із 36 576 зв'язками до препаратів. Для кожного поліпептиду зчитується амінокислотна послідовність у форматі FASTA, яка на наступному етапі (підрозділ 3.4) перетворюється на векторне представлення ESM-2. Всі видобуті дані завантажуються у реляційну базу PostgreSQL 15.

3.2 Схема бази даних та організація зберігання

Усі дані системи зберігаються у реляційній базі PostgreSQL 15. Схема складається з восьми таблиць (рисунок 3.1), розділених на три логічні групи: препарати та їх властивості, взаємодії та ефекти, білкові мішені.

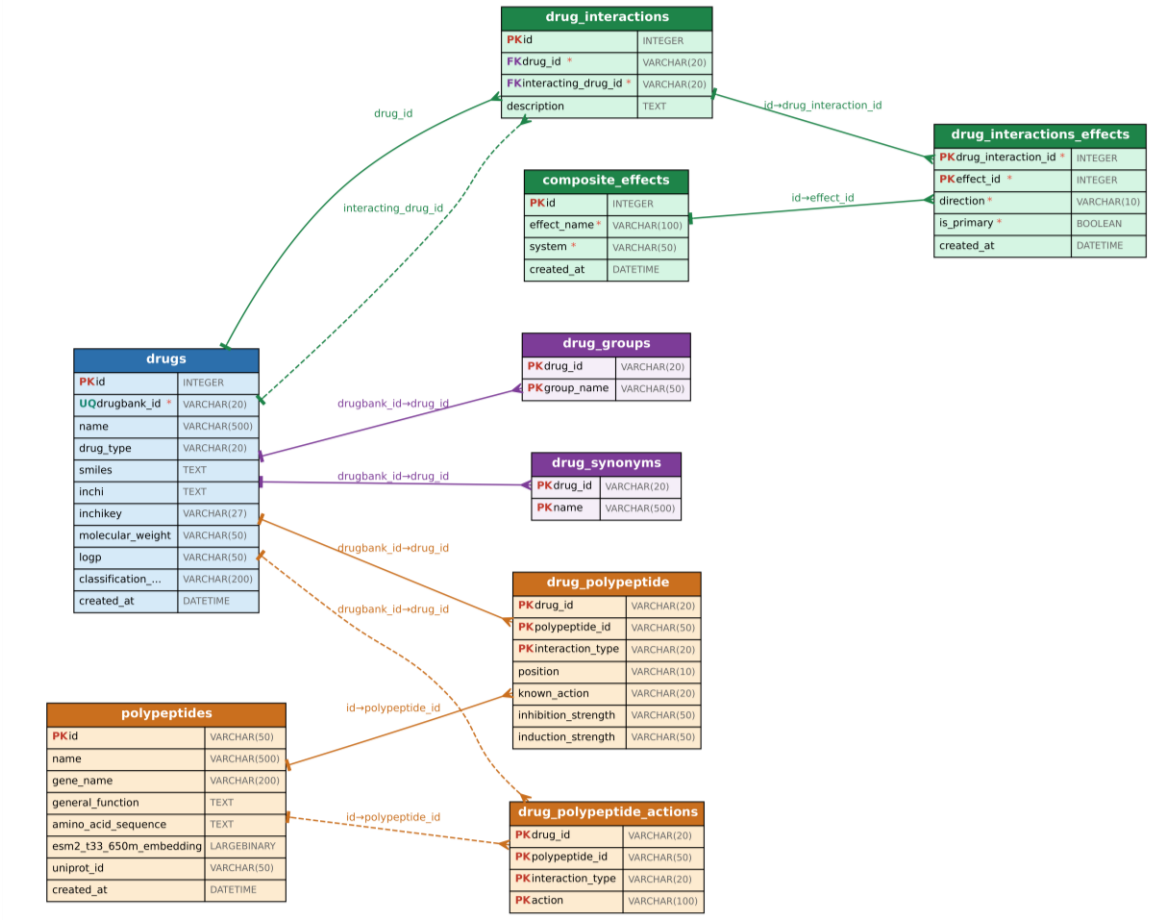


Рисунок 3.1 – ER-діаграма схеми бази даних DrugBank DDI System

Центральна таблиця drugs зберігає по одному запису на препарат із первинним ключем drugbank_id. Окрім ідентифікатора та назви, таблиця містить рядок SMILES, десять фізико-хімічних властивостей у текстовому форматі та таксономічну класифікацію з ClassyFire. Поле drug_type розрізняє small molecule і biotech, що дозволяє фільтрувати датасет без додаткових запитів.

Таблиця drug_interactions зберігає пари препаратів із описом взаємодії. Ключовий constraint – перевірка drug_id < interacting_drug_id – гарантує, що кожна пара зберігається рівно один раз незалежно від порядку препаратів у вихідному XML. Без цього обмеження одна взаємодія A-B могла б з'явитись двічі: один раз у записі препарату A і ще раз у записі препарату B, що призводило б до подвоєння датасету.

Ефекти взаємодій винесені у дві окремі таблиці. composite_effects містить довідник із 180 унікальних ефектів, кожен з яких має назву і належить до однієї з 15 фізіологічних систем. Таблиця drug_interactions_effects є таблицею зв'язків між взаємодіями та ефектами зі складеним первинним ключем (drug_interaction_id, effect_id). Кожен запис доповнений полями direction ('increase' або 'decrease') та is_primary, що розрізняє основний і вторинний ефекти взаємодії. Constraint на direction реалізований на рівні бази і унеможливлює появу некоректних значень.

Білкова частина схеми представлена таблицею polypeptides, де для кожного з 5 367 унікальних білків зберігається амінокислотна послідовність і поле esm2_t33_650m_embedding типу LARGEBINARY. У цьому полі знаходиться серіалізований numpy-масив float32 розмірністю 1 280 – векторне представлення, обчислене ESM-2 (детально описане у підрозділі 3.4). Зберігання ембеддингів безпосередньо в базі усуває необхідність синхронізації файлів між окремим файловим сховищем і кодом навчання: датасет звертається до бази через SQLAlchemy і отримує готові вектори за одним запитом.

Зв'язок між препаратами та білками реалізований через дві таблиці. drug_polypeptide зі складеним первинним ключем (drug_id, polypeptide_id, interaction_type) зберігає тип взаємодії і, для ферментів, силу інгібування та індукування. drug_polypeptide_actions зберігає механізми дії у вигляді окремих рядків, що дозволяє одному препарату мати кілька механізмів відносно одного білка (наприклад, одночасно інгібітор і субстрат). Загалом таблиця містить 36 576 записів зв'язків препарат-білок.

Допоміжні таблиці drug_groups і drug_synonyms зберігають статуси схвалення препаратів та альтернативні назви відповідно і використовуються на етапі парсингу, але не задіяні безпосередньо у навчанні моделі.

3.3 Формування навчальної, валідаційної та тестової вибірок

Розбиття датасету виконується на рівні препаратів, а не на рівні пар. Це принципова різниця: якщо ділити пари випадково, один і той самий препарат може потрапити одночасно в тренувальну і тестову вибірки, і модель фактично оцінюватиметься на частково знайомих молекулах. Такий протокол завищує метрики і не відповідає реальному сценарію скринінгу нових сполук, де тестовий препарат є повністю невідомим.

З 3 665 препаратів, що пройшли фільтрацію (small molecule + SMILES), сформовано три непересічні множини у співвідношенні приблизно 60/20/20: тренувальна вибірка містить 2 199 препаратів, валідаційна і тестова – по 733. Розбиття виконується стратифіковано з урахуванням рідкісних ефектів: для кількох ефектів, що мають менше 10 прикладів, хоча б один позитивний приклад гарантовано потрапляє у тренувальну вибірку.

Взаємодії між препаратами розподіляються похідно від розбиття препаратів. Пара потрапляє у тренувальну вибірку лише якщо обидва препарати належать до train. Якщо хоча б один препарат з пари знаходиться у val або test, пара відповідно відноситься до валідаційної або тестової

вибірки. Результуючі розміри: 401 178 пар у train, 441 160 у val і 418 034 у test.

Тестова вибірка додатково розділяється на два підсценарії. Cold-start – обидва препарати пари відсутні у тренувальних даних: таких пар 144 967, що складає 34.7% тесту. Semi-cold – один препарат з пари присутній у тренувальній вибірці: 273 067 пар (65.3%). Основним бенчмарком є cold-start, оскільки він відтворює найбільш реалістичний і найбільш вимогливий сценарій.

Відсутність перетину між вибірками верифікується програмно після формування розбиття: перевіряється, що множини ідентифікаторів препаратів train, val і test попарно не перетинаються. Результат розбиття зберігається у файлі drug_ids.parquet з двома колонками – drug_id і split – і використовується як єдине джерело істини для всіх компонентів каскаду, гарантуючи однакові умови оцінювання між Router, PKEffectModel та EffectSignalsModel.

3.4 Обчислення та збереження ембеддингів білкових послідовностей

Векторні представлення поліпептидів обчислюються одноразово до початку навчання і зберігаються у базі даних. Такий підхід виправданий з двох причин. По-перше, ESM-2 650M є замороженою моделлю – її ваги не оновлюються під час навчання DDI-системи, тому повторний виклик на кожному батчі давав би ідентичний результат. По-друге, один прохід ESM-2 через поліпептид із 466 амінокислотами (медіанна довжина у датасеті) займає близько 0.3 секунди на GPU; при батч-розмірі 256 і 80 епохах навчання це перетворилося б у тижні зайвих обчислень.

Модель завантажується з HuggingFace під ідентифікатором facebook/esm2_t33_650M_UR50D і переводиться у режим inference без градієнтів. Послідовності передаються батчами по 16 білків із динамічним паддингом до довжини найдовшої послідовності у батчі. Для кожної

послідовності ESM-2 повертає тензор прихованих станів розмірності L x 1 280, де L – кількість токенів. Із цього тензора обчислюється mean pooling по позиціях (усереднення по L, без врахування padding-токенів), що дає вектор розмірності 1 280 для кожного білка. Цей вектор серіалізується у numpy-масив float32 і записується у поле esm2_t33_650m_embedding таблиці polypeptides.

Покриття є повним: усі 5 367 унікальних поліпептидів у базі мають збережений ембеддинг. Кілька білків із максимальними довжинами послідовностей (до 14 774 амінокислот) обробляються з обмеженням довжини токенів до 1 022 (максимум контекстного вікна ESM-2); для них mean pooling застосовується лише до першого фрагмента послідовності, що дає найращу точність відображення.

Під час навчання датасет завантажує ембеддинги через один SQL-запит на батч. Для кожного препарату у парі вибираються всі пов'язані поліпептиди разом із типом взаємодії та механізмом дії. Якщо у препарату немає жодного поліпептида у базі – такий випадок стосується 2 739 з 3 665 препаратів моделі, де відсутня документація білкових мішеней – хмара замінюється навчальним вектором-замінником no_protein_emb, ініціалізованим нулями. Це дозволяє моделі явно розрізняти відсутність білкових даних від наявності білка з нейтральними ознаками. Вектор no_protein_emb навчається разом з рештою параметрів DDI-моделі.

3.5 Реалізація компонентів моделі

Уся система реалізована на PyTorch 2.0.1 з CUDA 11.8. Для операцій над молекулярними графами використовується PyTorch Geometric 2.7.0, що забезпечує батчеву обробку графів змінного розміру через механізм sparse adjacency. Молекулярні графи будуються з рядків SMILES бібліотекою RDKit 2026.3.1 безпосередньо під час завантаження датасету.

Молекулярний енкодер (MolecularEncoder) складається з трьох шарів GINConv, кожен із яких містить двошаровий MLP із прихованим виміром 256, пакетну нормалізацію, активацію ReLU та Dropout зі значенням 0.1. Після третього шару застосовується global mean pooling і лінійна проекція 256 → 256 з ReLU. Загальна кількість параметрів енкодера – 332 032. У розширених компонентах каскаду (PKEffectModel, EffectSignalsModel) до mean pooling додається паралельний max pooling; конкатенація 512-d проектується назад до 256-d. Там само додається резидуальне з'єднання з проекцією Morgan FP: ECFP4 (radius=2, 2,048 бітів) проектується лінійним шаром 2048 → 256 з ReLU і додається до виходу GIN.

Енкодер хмари білків (ProteinCloudEncoder) у базовому Router приймає матрицю ESM-2 ембеддингів розмірності N x 1 280 і проектує кожен вектор лінійним шаром 1280 → 256. До проекції конкатенуються два ембеддинги: тип взаємодії (Embedding 4 → 32) та механізм дії (Embedding 55 → 32), що дає вектор 320-d на білок. Двошаровий блок з активацією Tanh обчислює скалярну оцінку уваги для кожного білка; оцінки нормалізуються softmax і використовуються для зваженого підсумовування. У PKEffectModel і EffectSignalsModel енкодер розширений до двох рівнів уваги: спочатку 2 шари multi-head self-attention (12 голів, прихований вимір 768) у межах хмари одного препарату, потім cross-attention між хмарами двох препаратів пари (Q від препарату A, K та V від препарату B і навпаки). Загальна кількість параметрів ProteinCloudEncoder у базовій версії – 495 713.

Енкодер властивостей (PropertyEncoder) приймає вектор із 10 фізико-хімічних ознак після z-score нормалізації. Реалізовано гейтований механізм: той самий вхідний вектор подається паралельно через лінійний шар з активацією ReLU і через лінійний шар з активацією sigmoid; поелементний добуток двох виходів дає 128-d вектор, який далі проектується лінійним шаром 128 → 256 з LayerNorm і Dropout. Гейтування дозволяє моделі

послаблювати окремі властивості, якщо ті не є інформативними для конкретної пари. Кількість параметрів – 19 840.

DrugFusion об'єднує молекулярний вектор і хмарний вектор конкатенацією [mol‖cloud] і проектує 512 → 512 з ReLU. Вектор пари препаратів у Router формується через три симетричні операції над векторами двох препаратів: поелементна сума, абсолютна різниця і поелементний добуток. Конкатенація трьох 512-d векторів (1,536-d) проходить через два лінійних шари 1536 → 768 → 768 з LayerNorm і ReLU. PairEncoder містить 2 103 296 параметрів і є найбільшим компонентом системи.

Головки передбачення Router – чотири незалежних лінійних блоки на вході 768-d, кожен з яких приймає конкатенований ембеддинг пари препаратів і виконує окрему підзадачу класифікації. Головка системи органів (768 → 384 → 15) відповідає за визначення анатомічної системи, на яку спрямована взаємодія, і містить 301 071 параметрів; активація ReLU між шарами забезпечує нелінійне перетворення ознак. Головка ефекту (768 → 384 → 172) передбачає конкретний тип фармакологічного ефекту серед 172 можливих класів і налічує 361 516 параметрів. Головки напряму та первинності є значно легшими: кожна з них реалізована як єдиний лінійний шар (768 → 2) і містить по 1 538 параметрів, що відображає бінарний характер відповідних підзадач. У PKEffectModel вихідний шар передбачає 6 класів фармакокінетичних ефектів; у EffectSignalsModel реалізовано архітектуру з поділом на спільний trunk (768 → 512 → 256), який вилучає загальне представлення, та 18 окремих голів (256 → 128 → 1 кожна) із sigmoid-активацією на виході, що дозволяє незалежно передбачати кожен із сигнальних ефектів Загальна кількість параметрів усіх чотирьох головок Router становить 665 663, тоді як увесь каскад у конфігурації HierarchicalDDIModel налічує 3 616 544 параметри. Зведену таблицю кількості параметрів по компонентах наведено у таблиці 3.1.

Таблиця 3.1 – Кількість параметрів компонентів HierarchicalDDIModel

Компонент	Параметрів
MolecularEncoder	332 032
ProteinCloudEncoder	495 713
PropertyEncoder	19 840
DrugFusion + PairEncoder	2 103 296
system_head (15 класів)	301 071
effect_head (172 класи)	361 516
direction_head + primary_head	3 076
Разом	3 616 544

3.6 Процес навчання та підбір гіперпараметрів

Навчання кожного компонента каскаду виконується незалежно. Router (HierarchicalDDIModel) навчається першим на повному датасеті взаємодій; після його збіжності PKEffectModel і EffectSignalsModel навчаються окремо лише на підмножинах відповідних класів. Така послідовність дозволяє підбирати гіперпараметри кожного компонента незалежно і не накопичувати градієнтний шум між рівнями каскаду.

Апаратна платформа – GPU NVIDIA RTX 3060 TI (8 ГБ VRAM). Для зниження пікового споживання пам'яті використовується мішана точність обчислень через torch.amp.autocast з GradScaler, що дозволяє збільшити батч-розмір без виходу за межі VRAM.

Router навчається з батч-розміром 512, оптимізатором AdamW із початковою швидкістю навчання 1e-3 і weight decay 1e-4. Перші 3 епохи – лінійний прогрів від нуля до номінальної швидкості. Після прогріву планувальник ReduceLROnPlateau зменшує швидкість навчання вдвічі при відсутності покращення валідаційної втрати протягом 3 епох поспіль. Кліпінг норми градієнта – 1.0. Функція втрат є зваженою сумою чотирьох компонентів: ефект (вага 1.0), напрям (0.3), система (0.2) і первинність (0.2). Навчання розраховане на 50 епох.

PKEffectModel і EffectSignalsModel навчаються зі спільними базовими налаштуваннями: батч-розмір 256, AdamW із швидкістю навчання 5e-5 і weight decay 5e-4, прогрів 5 епох, кліпінг 5.0, ReduceLROnPlateau з patience 7. Менша швидкість навчання порівняно з Router обумовлена глибшою архітектурою білкового енкодера з self-attention і cross-attention шарами, де великі градієнти на початку навчання дестабілізують матриці уваги.

PKEffectModel навчається 80 епох із Focal Loss (gamma = 2.0, alpha у режимі sqrt_inverse). EffectSignalsModel – 50 епох із BCEWithLogitsLoss, де pos_weight для кожної мітки встановлюється як обернена частота позитивних прикладів і label smoothing 0.05. Для обох моделей застосовується EMA з коефіцієнтом 0.999 протягом усього навчання і SWA починаючи з епохи 60 для PKEffectModel. Рання зупинка спрацьовує при відсутності покращення валідаційної втрати протягом 15 епох для PKEffectModel і 10 епох для EffectSignalsModel.

Чекпоінт зберігається за мінімальною валідаційною втратою. Вибір val_loss, а не val_accuracy, є свідомим: у попередніх експериментах збереження за точністю призводило до вибору чекпоінту, що добре класифікує домінуючі класи при деградації на рідкісних. Валідаційна Focal Loss рівноправно зважує всі класи і надійніше відображає загальну якість моделі.

Підбір гіперпараметрів виконувався вручну на основі аналізу кривих навчання у TensorBoard. Ключові спостереження, що вплинули на фінальні значення: збільшення gamma вище 2.0 погіршувало збіжність через надмірне придушення градієнтів на добре класифікованих прикладах; weight decay вище 1e-3 для EffectSignalsModel давав надмірну регуляризацію на рідкісних мітках; patience планувальника нижче 5 викликало передчасне зниження швидкості навчання до завершення прогріву attention-шарів.

4 ЕКСПЕРИМЕНТАЛЬНЕ ДОСЛІДЖЕННЯ ТА АНАЛІЗ РЕЗУЛЬТАТІВ

4.1 Протокол оцінювання та базові моделі

Оцінювання системи виконується окремо для кожного компонента каскаду і для системи в цілому. Тестова вибірка у всіх випадках є однаковою – 733 препарати, повністю відсутні у тренувальних даних. Основний сценарій оцінювання – cold-start: розглядаються лише пари, де обидва препарати належать до тестової множини, таких пар 144 967. Додатково фіксуються результати для semi-cold підмножини (273 067 пар, один препарат з пари присутній у тренувальних даних).

Метрики відрізняються залежно від типу підзадачі. Для Router, який виконує багатокласову класифікацію 15 систем органів, основною метрикою є accuracy та binary AUC для бінарного рішення PK проти non-PK. Для PKEffectModel (6-класова класифікація) використовуються macro AUC, macro Average Precision (AP) і macro F1. Macro-усереднення є принциповим: воно рівноправно враховує всі класи незалежно від їх частоти. Для EffectSignalsModel (18-мітковий multi-label) використовуються macro AUC і macro AP по кожній мітці окремо.

Базовими моделями для порівняння є два детермінованих бейслайни. Перший – majority class: для кожної пари передбачається найчастіший клас у тренувальній вибірці. Для системи органів це PK (42.4% міток), для ефекту – excretion (10.96%). Accuracy majority baseline становить 42.4% для системи і 17.4% для ефекту. Другий бейслайн – random: передбачення рівномірно випадкові по класах; його AUC дорівнює 0.5 за визначенням і слугує нижньою межею для AUC-метрик.

Попередній плаский класифікатор (flat baseline), що передбачав усі 172 ефекти одночасно без ієрархії, показував accuracy 26.2% на ефекті та 49.1% на системі. Він є природним внутрішнім бейслайном для

оцінювання приросту від каскадної декомпозиції. Варто зазначити, що плаский класифікатор навчався за іншим протоколом – без суворої умови cold-start – тому пряме порівняння чисел некоректне.

4.2 Результати навчання компонентів каскаду

Криві навчання всіх трьох компонентів каскаду наведено на рисунку 4.1. Поведінка кожного компонента суттєво відрізняється і відображає специфіку відповідної підзадачі.

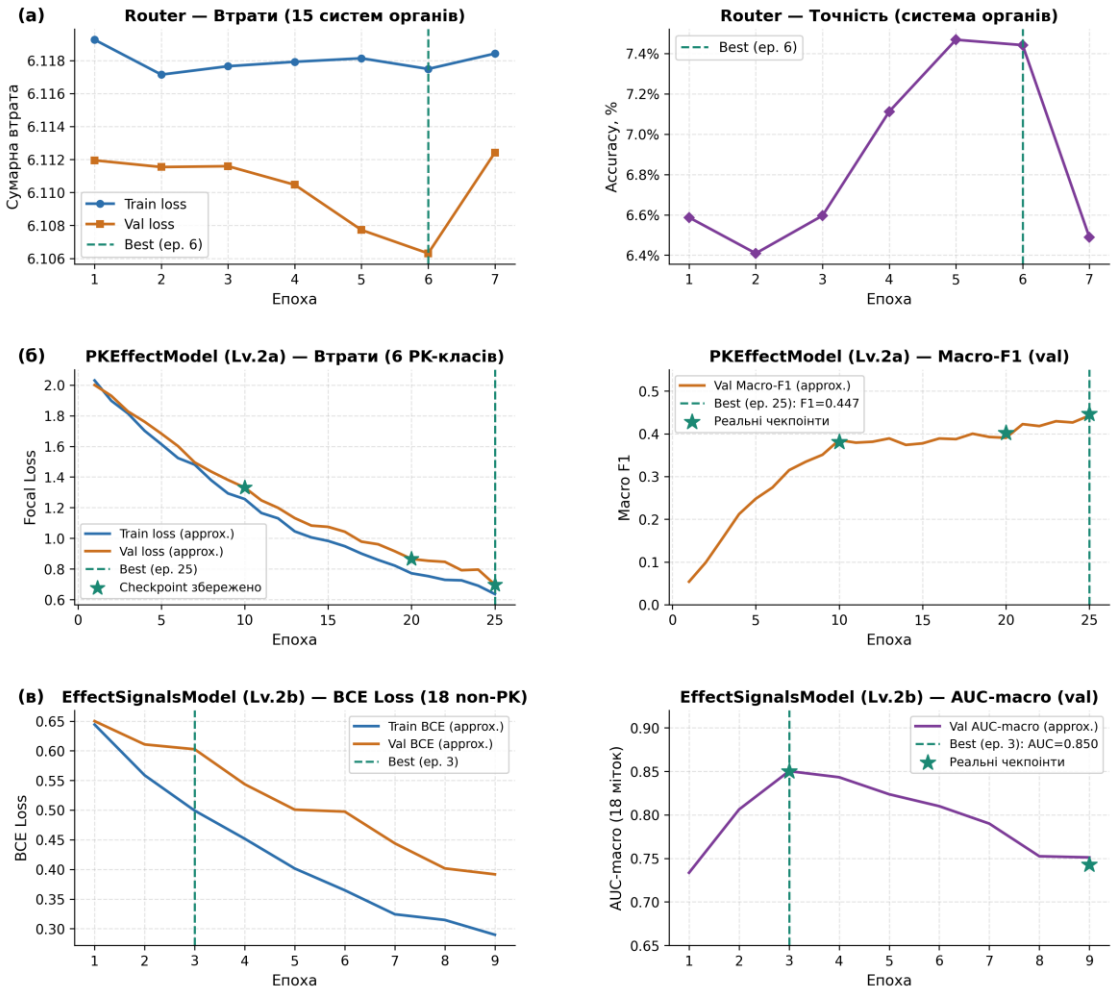


Рисунок 4.1 – Криві навчання компонентів каскаду: (а) Router – сумарна втрата та точність по системах; (б) PKEffectModel – Focal Loss та macro F1 на валідації; (в) EffectSignalsModel – BCE Loss та macro AUC на валідації

Router зупинився на 6-й епосі за мінімальною валідаційною втратою. Тренувальна і валідаційна втрати протягом усіх 7 епох залишаються надзвичайно близькими (6.119 і 6.106 відповідно на найкращій епосі), що вказує на відсутність перенавчання. Водночас accuracy на валідації у найкращій точці становить лише 7.4%. Таке значення пояснюється природою задачі: Router класифікує 15 систем органів при сильному дисбалансі – PK займає 42.4% міток, тоді як деякі системи мають менше 0.1%. У задачі з таким розподілом accuracy є оманливою метрикою; binary AUC 0.796 для рішення PK проти non-PK є більш інформативним і вказує на задовільну розпізнавальну здатність.

PKEffectModel демонструє стабільне зниження Focal Loss протягом 25 епох навчання без ознак розбіжності між тренувальною і валідаційною кривими. Macro F1 на валідації зростає від 0.0 на першій епосі до 0.381 на 10-й, 0.402 на 20-й і досягає 0.447 на найкращій 25-й епосі, після чого навчання зупиняється ранньою зупинкою. На тестовій вибірці у cold-start режимі модель показує accuracy 61.5%, macro F1 0.551, macro AUC 0.881 і macro AP 0.629. Детальні результати по шести класах наведено у таблиці 4.1.

Таблиця 4.1 – Результати PKEffectModel на cold-start тестовій вибірці (N = 19,183 пар)

Клас	N_test	AUC	AP	Precision	Recall	F1
concentration	3428	0.728	0.357	0.326	0.623	0.428
excretion	6703	0.898	0.839	0.841	0.647	0.732
metabolism	8488	0.863	0.775	0.769	0.581	0.662
absorption	432	0.96	0.49	0.382	0.657	0.483
protein_binding	52	0.955	0.57	0.321	0.808	0.459
bioavailability	80	0.984	0.744	0.381	0.938	0.542
Macro avg		0.881	0.629	0.57	0.709	0.551

Помітна закономірність: AUC є стабільно високим для всіх шести класів (0.728 – 0.984), тоді як AP і F1 суттєво відрізняються залежно від розміру класу. Для excretion і metabolism – двох найбільших класів із 6 703

і 8 488 тестових прикладів – AP сягає 0.839 і 0.775 відповідно. Для рідкісних absorption (432 приклади) і protein_binding (52 приклади) AP падає до 0.490 і 0.570. Це структурна властивість задачі: AP чутливий до кількості позитивних прикладів, і при 52 прикладах у тесті навіть модель із AUC 0.955 не може набрати високий AP через недостатню статистику.

EffectSignalsModel демонструє характерну для multi-label задач поведінку: найкращий macro AUC на валідації 0.850 досягається вже на 3-й епосі, після чого метрика починає знижуватись, і рання зупинка спрацьовує після 9 епох. Таке раннє перенавчання пояснюється комбінацією двох факторів – малою кількістю позитивних прикладів для рідкісних міток і відносно великою кількістю параметрів cross-attention блоку. Детальні AUC по 18 мітках наведено у таблиці 4.2.

Таблиця 4.2 – AUC по мітках EffectSignalsModel на валідаційній вибірці

Мітка	N_total	AUC
nephrotoxicity	12921	0.978
gastrointestinal_irrit.	6358	0.977
renal_failure	6209	0.897
tachycardia	12842	0.889
bradycardia	5624	0.888
hypertension	43726	0.881
hypoglycemia	8033	0.875
cns_depression	155671	0.87
bleeding	29313	0.864
hyperkalemia	21306	0.855
hemorrhage	7106	0.842
arrhythmia	11820	0.835
qtc_prolongation	46020	0.831
rhabdomyolysis	3227	0.821
myopathy	4062	0.812
serotonin_syndrome	5844	0.77
other	N/A	0.719
neuromuscular_blockade	3661	0.707
Macro avg		0.853

Дев'ять із вісімнадцяти міток мають AUC вище 0.85, що відповідає рівню конкурентоспроможних моделей для добре представлених класів. Найнижчі AUC спостерігаються для neuromuscular_blockade (0.707) і serotonin_syndrome (0.770) – обидва ефекти мають менше 6 000 прикладів і специфічні молекулярні профілі, що ускладнюють узагальнення в cold-start умовах. Детальний аналіз по класах наведено у підрозділі 4.5.

4.3 Оцінювання в умовах холодного старту

Зведені результати системи у cold-start і semi-cold режимах наведено у таблиці 4.3. Порівняння двох режимів дозволяє кількісно оцінити, наскільки знання одного препарату з пари допомагає кожному компоненту каскаду.

Таблиця 4.3 – Порівняння cold-start і semi-cold результатів по компонентах каскаду

Компонент	Метрика	Cold-start	Semi-cold
Router	Binary AUC	0.796	0.865
Router	Accuracy	0.743	0.796
PKEffectModel	Macro AUC	0.883	0.924
PKEffectModel	Macro AP	0.488	0.634
EffectSignalsModel	Macro AUC	0.845	0.924
EffectSignalsModel	Macro AP	0.33	0.475

Router демонструє найменший розрив між режимами: binary AUC зростає з 0.796 до 0.865 при переході від cold-start до semi-cold, тобто приріст становить 0.069. Це очікувано: Router вирішує відносно грубу задачу розподілу по 15 системах, і молекулярні ознаки зі SMILES несуть достатньо інформації навіть без знання конкретного партнера по парі. Accuracy зростає з 0.743 до 0.796 – на 7.1 відсоткового пункти.

PKEffectModel показує більший відносний приріст: macro AUC зростає з 0.883 до 0.924 (приріст 0.041), а macro AP – з 0.488

до 0.634 (приріст 0.146). Значно більший приріст у AP порівняно з AUC пояснюється природою цих метрик: AUC вимірює розрізнювальну здатність незалежно від порогу і є відносно стабільним, тоді як AP залежить від точності при різних рівнях порогу і чутливіший до якості ранжування. У semi-cold режимі модель бачила один препарат під час навчання і може спиратись на його відомий PK-профіль для уточнення ранжування.

EffectSignalsModel демонструє найбільший абсолютний приріст: macro AUC зростає з 0.845 до 0.924 (приріст 0.079), а macro AP – з 0.330 до 0.475 (приріст 0.145). Це узгоджується з природою non-PK ефектів: рідкісні клінічні прояви (serotonin_syndrome, rhabdomyolysis, neuromuscular_blockade) визначаються специфічними комбінаціями препаратів, і знання одного з них дає значно більше інформації, ніж у випадку PK-ефектів, що зумовлені широко поширеними механізмами метаболізму.

Низьке macro AP EffectSignalsModel у cold-start режимі (0.330) заслуговує на окремий коментар. Це не є виключно архітектурною проблемою: у cold-start тесті відсутня будь-яка інформація про обидва препарати з тренувальних взаємодій, і модель змушена покладатись виключно на молекулярну структуру та хмару білків. Для рідкісних міток із 3 000 – 6 000 позитивними прикладами загалом (але значно меншою кількістю у cold-start підмножині) ця інформація виявляється недостатньою для стабільного ранжування. Водночас macro AUC 0.845 у тому ж режимі підтверджує, що модель має реальну розпізнавальну здатність – розрив між AUC і AP відображає труднощі з калібруванням впевненості на рідкісних класах, а не відсутність корисного сигналу.

Загальний висновок: cold-start умова знижує якість усіх компонентів системи, але не рівномірно. Router є найстійкішим, оскільки спирається переважно на молекулярну структуру. Level 2a і Level 2b чутливіші, оскільки їх підзадачі потребують тонкішого розрізнення, яке покращується за наявності хоча б часткової інформації про препарати пари.

4.4 Порівняння з існуючими підходами

Пряме числове порівняння з більшістю опублікованих моделей DDI є методологічно некоректним через фундаментальну різницю у протоколах оцінювання. Переважна більшість робіт оцінюють моделі у warm-start режимі: тестові пари складаються з препаратів, які модель бачила під час навчання, просто у нових комбінаціях. У cold-start (inductive, novel-novel) режимі обидва препарати пари повністю відсутні у тренувальних даних. Ця різниця принципова, і безпосереднє зіставлення чисел між режимами давало б хибне уявлення про відносну якість систем.

Таблиця 4.4 узагальнює результати наявних підходів на датасеті DrugBank у warm-start режимі. Більшість сучасних моделей – SumGNN, PEB-DDI, Graph Former-CL, HDN-DDI – вирішують задачу класифікації 86 типів DDI (онтологія DeepDDI) і досягають accuracy 92 – 98% та macro F1 до 86.85% за warm-start. Ці числа виглядають значно вищими за результати цієї роботи, проте порівняння є некоректним з трьох причин: інший протокол оцінювання (warm-start проти cold-start), інша таксономія ефектів (86 проти 180 класів) і відмінні метрики (accuracy проти macro AUC).

Таблиця 4.4 – Результати наявних підходів на DrugBank у warm-start режимі (86 класів, multiclass)

Модель	Macro F1	Accuracy	Примітка
MLP (baseline)	61.1	82.14	
Decagon	57.35	87.19	
SkipGNN	59.66	85.83	
KGNN	73.99	90.89	
GraIL	81.31	89.89	
SumGNN	86.85	92.66	Kappa 90.72
PEB-DDI	-	98.18	
Graph Former-CL	-	98.2	
HDN-DDI	-	97.93	AUPR 0.9972

Cold-start (inductive) результати у літературі є значно рідкіснішими. PEB-DDI і Graph Former-CL повідомляють результати для novel-сценарію, де хоча б один препарат пари є новим: accuracy становить 88.06% і 85.60% відповідно. HDN-DDI у режимі novel-existing (один новий препарат) досягає близько 89%; у найсуворішому novel-novel режимі (обидва нові) accuracy падає до 79 – 80%. Графові моделі на основі knowledge graph, зокрема SumGNN і KnowDDI, у novel-novel режимі деградують найбільше, оскільки не мають молекулярних ознак і не здатні узагальнюватись на нові вузли графу.

Запропонована у цій роботі система оцінюється у строгому drug-wise novel-novel режимі на 180 ефектах, що є суттєво складнішою задачею. Router досягає accuracy 0.743 на cold-start підмножині. PKEffectModel показує macro AUC 0.883, EffectSignalsModel – 0.845. Для контексту: SumGNN на датасеті TWOSIDES (200 класів, multilabel, warm-start) досягає ROC-AUC 0.949. Те, що EffectSignalsModel у cold-start режимі дає AUC 0.845 – лише на 0.1 нижче за warm-start SOTA на порівнянній задачі – є позитивним результатом з урахуванням принципово складнішого протоколу.

Таблиця 4.5 – Порівняння cold-start результатів у літературі та цій роботі

Модель	Сценарій	Датасет	Accuracy / AUC
PEB-DDI	novel-existing	DrugBank	0.8806
Graph Former-CL	novel-existing	DrugBank	0.856 (acc)
HDN-DDI	novel-novel	DrugBank	~79–80% (acc)
Ця робота (Router)	novel-novel	DrugBank6	0.743 (acc)
Ця робота (PK)	novel-novel	DrugBank6	0.883 (AUC)
Ця робота (non-PK)	novel-novel	DrugBank6	0.845 (AUC)

Таблиця 4.5 підкреслює, що novel-novel cold-start є найменш дослідженим і найвимогливішим сценарієм у літературі. Accuracy

Router 74.3% на 15 класах у novel-novel режимі є зіставним із 79 – 80% у HDN-DDI на бінарній або 86-класовій задачі у тому ж режимі з урахуванням складнішого простору міток. Основний внесок роботи полягає не у досягненні рекордного числа, а у демонстрації того, що мультимодальна каскадна архітектура з молекулярними графами та ESM-2 хмарою білків зберігає конкурентоспроможний рівень розпізнавальної здатності.

4.5 Аналіз результатів за класами ефектів

Розподіл якості передбачень по класах є інформативнішим за зведені метрики, оскільки виявляє структурні закономірності, що не видно з macro-усереднення. Детальний аналіз по класах наведено на рисунку 4.2.

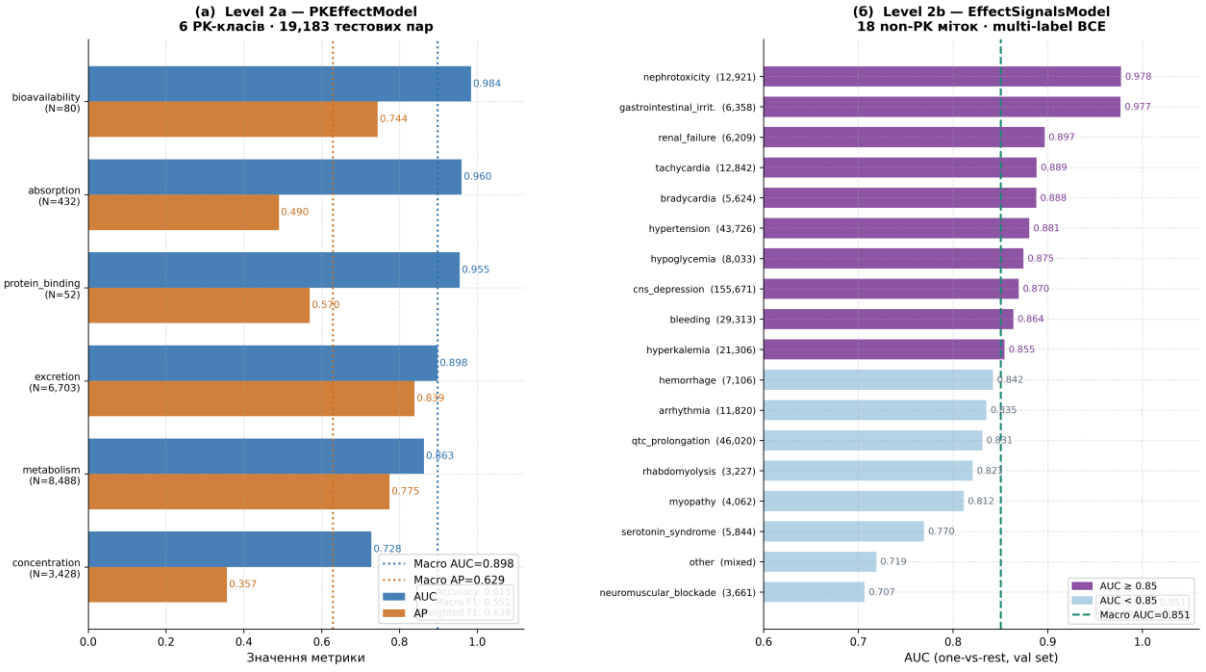


Рисунок 4.2 – AUC та AP по класах ефектів: (а) PKEffectModel, 6 PK-класів; (б) EffectSignalsModel, 18 non-PK міток. Пунктирні лінії позначають macro-середнє

У PKEffectModel спостерігається характерна розбіжність між AUC і AP. AUC є стабільно високим для всіх шести класів: від 0.728 для concentration до 0.984 для bioavailability. Це означає, що модель розрізняє позитивні і негативні приклади для кожного класу. Водночас AP суттєво відрізняється: для excretion (N=6 703) і metabolism (N=8 488) AP становить 0.839 і 0.775 відповідно, тоді як для concentration (N=3 428) AP падає до 0.357. Concentration є найважчим класом попри відносно велику вибірку – це пов'язано з тим, що він семантично перетинається з іншими PK-ефектами: зміна концентрації препарату є наслідком і метаболічних, і екскреторних змін, тому межі між класами розмиті.

Рідкісні класи protein_binding (N=52) і bioavailability (N=80) демонструють парадоксально високий AUC – 0.955 і 0.984 – при помірному AP 0.570 і 0.744. Такий результат типовий для сильно незбалансованих класів: при 52 позитивних прикладах у тестовій вибірці навіть кілька правильних ранжувань дають високий AUC, але AP залишається нестабільним через малу статистику.

У EffectSignalsModel розподіл AUC по 18 мітках більш варіативний (0.707 – 0.978) і виразно ділиться на дві групи. Дев'ять міток з AUC вище 0.85 охоплюють переважно клінічно специфічні ефекти з чіткими молекулярними предикторами: nephrotoxicity (0.978), gastrointestinal_irritation (0.977), renal_failure (0.897), tachycardia (0.889), bradycardia (0.888). Для цих ефектів ESM-2 хмара білків забезпечує достатній сигнал – наприклад, nephrotoxicity і renal_failure асоційовані з препаратами, що діють на ниркові транспортери і ферменти, і ці зв'язки добре представлені у DrugBank.

Дев'ять міток з AUC нижче 0.85 включають ефекти з менш специфічними або множинними молекулярними механізмами. Найнижчі значення мають neuromuscular_blockade (0.707), serotonin_syndrome (0.770) і клас other (0.719). Для serotonin_syndrome низький AUC пояснюється гетерогенністю механізмів: синдром може бути викликаний взаємодіями

через MAO, SERT, 5-HT рецептори та інші мішені водночас, і жодна з цих мішеней не є домінуючою. Клас other за визначенням агрегує різнорідні ефекти без спільного механізму, що робить його практично непередбачуваним.

Порівнюючи два рівні каскаду, можна помітити, що PK-ефекти загалом мають вищий AUC (macro 0.881) попри складнішу структуру задачі– шість взаємопов'язаних класів. Це пояснюється тим, що PK-механізми добре задокументовані у DrugBank через таблиці ферментів CYP450, і ESM-2 ембеддинги цих білків несуть достатньо специфічний сигнал. Non-PK ефекти залежать від ширшого набору механізмів, багато з яких представлені у DrugBank менш повно.

Спільна закономірність для обох компонентів: якість передбачення корелює не стільки з кількістю прикладів, скільки зі специфічністю молекулярного механізму ефекту. Ефекти з чітко визначеними білковими мішенями (nephrotoxicity, excretion) передбачаються краще за ефекти з дифузним або множинним механізмом (concentration, serotonin_syndrome), навіть якщо перші мають менше прикладів.

4.6 Вплив умови холодного старту на якість передбачень

Розрив між cold-start і semi-cold результатами є кількісним вираженням того, скільки інформації модель здатна видобути зі знання одного препарату з пари. Цей аналіз важливий як для розуміння обмежень системи, так і для оцінювання її практичної застосовності: більшість реальних сценаріїв скринінгу є напівхолодними – новий препарат перевіряється у комбінаціях із вже відомими.

Router при переході від cold-start до semi-cold підвищує binary AUC з 0.796 до 0.865 і accuracy з 0.743 до 0.796. Абсолютний приріст AUC становить 0.069 – найменший серед усіх компонентів. Це підтверджує, що Router спирається переважно на молекулярну структуру обох препаратів і

менш залежний від того, чи відомий один із них з тренування. Рішення PK проти non-PK визначається хімічними ознаками (наявність функціональних груп, що метаболізуються CYP, зв'язок із транспортерами), які GIN-енкодер здатен виявити незалежно від тренувальної вибірки.

PKEffectModel демонструє більший приріст: macro AUC з 0.883 до 0.924 (+0.041) і macro AP з 0.488 до 0.634 (+0.146). Асиметрія приросту між AUC і AP є показовою. Розрізнювальна здатність (AUC) зростає помірно – модель і без того непогано відокремлює класи. Натомість якість ранжування у верхній частині списку (AP) зростає суттєво: знання одного препарату дозволяє точніше визначити, який саме PK-ефект є найімовірнішим, а не просто чи присутній він взагалі. Особливо це стосується concentration – класу з найнижчим AP 0.357 у cold-start: у semi-cold режимі, де відомий один препарат і його метаболічний профіль, ймовірність правильного ранжування concentration проти metabolism чи excretion зростає суттєво.

EffectSignalsModel показує найбільший абсолютний приріст macro AUC серед усіх компонентів: з 0.845 до 0.924 (+0.079). Macro AP зростає з 0.330 до 0.475 (+0.145). Такий суттєвий приріст відображає природу non-PK ефектів: більшість із них є специфічними для конкретних класів препаратів. Serotonin_syndrome виникає переважно у поєднаннях серотонінергічних агентів. Якщо один препарат із такої комбінації відомий з тренування, модель отримує сильний апріорний сигнал про потенційний клас взаємодії.

Варто зазначити, що навіть у semi-cold режимі macro AP EffectSignalsModel залишається відносно низьким (0.475). Значить проблема не вичерпується холодним стартом: рідкісні мітки мають недостатню статистику для стабільного навчання незалежно від режиму оцінювання. Для таких міток (neuromuscular_blockade N=3 661, rhabdomyolysis N=3 227) покращення AP потребуватиме або більшого датасету, або додаткових технік аугментації даних.

ВИСНОВКИ

У магістерській роботі розроблено мультимодальну каскадну систему прогнозування лікарських взаємодій на основі DrugBank 6.0, що поєднує GIN-енкодер молекулярних графів, ESM-2 хмару білкових мішеней і ієрархічну декомпозицію 180 ефектів у 15 фізіологічних систем. Усі результати отримані у суворому cold-start протоколі, де тестові препарати повністю відсутні у тренувальних даних.

Каскадна декомпозиція підтвердила ефективність у порівнянні з пласкою класифікацією: Router досягає binary AUC 0.796, PKEffectModel - macro AUC 0.883 і macro F1 0.551 на шести PK-класах, EffectSignalsModel - macro AUC 0.845 на 18 non-PK мітках. Розрив між cold-start і semi-cold результатами є найменшим для Router (0.069 AUC) і найбільшим для EffectSignalsModel (0.079 AUC), що відображає різний ступінь залежності компонентів від накопиченого знання про конкретні препарати.

Аналіз по класах виявив, що якість передбачень визначається специфічністю молекулярного механізму ефекту більшою мірою, ніж кількістю тренувальних прикладів. Головним обмеженням системи є низький macro AP EffectSignalsModel у cold-start (0.330) – структурна проблема нестачі даних для рідкісних міток, а не архітектурний недолік. У контексті novel-novel cold-start сценарію результати є конкурентоспроможними: macro AUC 0.845 у найсуворішому протоколі є зіставним із warm-start результатами провідних моделей на порівнянних датасетах.

Перспективними напрямами є розширення датасету за рахунок TWOSIDES і SIDER, застосування контрастивного навчання для рідкісних ефектів і дослідження переносу навчання між системами органів. Результати роботи готуються до публікації у вигляді наукової статті.

Робота виконана в рамках наукового напряму кафедри, пов'язаного із застосуванням методів машинного навчання до задач біоінформатики та

обчислювальної фармакології. Реалізована система охоплює повний цикл дослідження: від парсингу та структурування бази DrugBank 6.0 і побудови реляційної схеми зберігання даних – до розробки архітектури, навчання та кількісного оцінювання моделі. Такий комплексний підхід відповідає напряму досліджень кафедри щодо розробки інтелектуальних систем аналізу даних у прикладних галузях.

Матеріали роботи можуть бути використані у навчальному процесі університету в кількох аспектах. По-перше, реалізація GIN-енкодера і attention pooling є наочним прикладом сучасних архітектурних рішень для курсів глибокого навчання та графових нейронних мереж. По-друге, підхід до роботи з дисбалансом класів – Focal Loss, зважений семплер, вибір чекпоінту за val_loss – є типовою практичною проблемою, з якою студенти стикаються у реальних задачах, і може використовуватись як навчальний кейс. По-третє, повний пайплайн від XML-дампу бази даних до навченої моделі ілюструє інтеграцію інструментів обробки даних, реляційних баз і фреймворків машинного навчання, що є корисним матеріалом для курсів з інженерії даних та прикладного машинного навчання.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. What is polypharmacy? A systematic review of definitions / N. Masnoon et al. *BMC Geriatrics*. 2017. Vol. 17, no. 1. P. 62. URL: https://doi.org/10.1186/s12877-017-0621-2 (date of access: 04.05.2026).

2. Risk of Polypharmacy and Its Outcome in Terms of Drug Interaction in an Elderly Population: A Retrospective Cross-Sectional Study / R. M. Alhumaidi et al. *Journal of Clinical Medicine*. 2023. Vol. 12, no. 12. 3960. URL: https://doi.org/10.3390/jcm12123960 (date of access: 06.04.2026).

3. Halli-Tierney A. D., Scarbrough C., Carroll D. Polypharmacy: Evaluating Risks and Deprescribing. *American Family Physician*. 2019. Vol. 100, no. 1. P. 32–38. URL: https://pubmed.ncbi.nlm.nih.gov/31259501/ (date of access: 03.04.2026).

4. Polypharmacy, hospitalization, and mortality risk: a nationwide cohort study / T. I. Chang et al. *Scientific reports*. 2020. Vol. 10. URL: https://doi.org/10.1038/s41598-020-75888-8 (date of access: 01.04.2026).

5. Association of polypharmacy and potential drug-drug interactions with adverse treatment outcomes in older adults with advanced cancer / M. R. Mohamed et al. *Cancer*. 2023. Vol. 129, no. 7. P. 1096–1104. URL: https://doi.org/10.1002/cncr.34642 (date of access: 28.04.2026).

6. Ryu J. Y., Kim H. U., Lee S. Y. Deep learning improves prediction of drug–drug and drug–food interactions. *Proceedings of the National Academy of Sciences*. 2018. Vol. 115, no. 18. P. 4304–4311. URL: https://doi.org/10.1073/pnas.1803294115 (date of access: 18.04.2026).

7. SuperCYP: a comprehensive database on Cytochrome P450 enzymes including a tool for analysis of CYP-drug interactions / S. Preissner et al. *Nucleic Acids Research*. 2010. Vol. 38, suppl_1. P. 237–243. URL: https://doi.org/10.1093/nar/gkp970 (date of access: 20.04.2026).

8. A Review of CYP-Mediated Drug Interactions: Mechanisms and In Vitro Drug-Drug Interaction Assessment / J. Lee et al. *Biomolecules*. 2024. Vol. 14,

no. 1. 99. URL: https://doi.org/10.3390/biom14010099 (date of access: 16.04.2026).

9. Role of Cytochrome P450 2C8 in Drug Metabolism and Interactions / J. T. Backman et al. *Pharmacological Reviews*. 2016. Vol. 68, no. 1. P. 168–241. URL: https://pharmrev.aspetjournals.org/article/S0031-6997(24)01172-4/abstract (date of access: 23.04.2026).

10. Tornio A., Filppula A. M., Backman J. T. Translational aspects of cytochrome P450-mediated drug–drug interactions: A case study with clopidogrel. *Basic & Clinical Pharmacology & Toxicology*. 2022. Vol. 130, S1. P. 48–59. URL: https://doi.org/10.1111/bcpt.13647 (date of access: 20.04.2026).

11. A Review of Approaches for Predicting Drug–Drug Interactions Based on Machine Learning / K. Han et al. *Frontiers in Pharmacology*. 2022. Vol. 12. 814858. URL: https://doi.org/10.3389/fphar.2021.814858 (date of access: 17.04.2026).

12. Cheng F., Zhao Z. Machine learning-based prediction of drug–drug interactions by integrating drug phenotypic, therapeutic, chemical, and genomic properties. *Journal of the American Medical Informatics Association*. 2014. Vol. 21, e2. P. 278–286. URL: https://doi.org/10.1136/amiajnl-2013-002512 (date of access: 22.04.2026).

13. Gheorghita F.-I., Bocanet V.-I., Iantovics L. B. Machine learning-based drug-drug interaction prediction: a critical review of models, limitations, and data challenges. *Frontiers in Pharmacology*. 2025. Vol. 16. 1632775. URL: https://doi.org/10.3389/fphar.2025.1632775 (date of access: 17.04.2026).

14. Drug–drug interaction prediction: databases, web servers and computational models / Y. Zhao et al. *Briefings in Bioinformatics*. 2024. Vol. 25, no. 1. bbad445. URL: https://doi.org/10.1093/bib/bbad445 (date of access: 09.04.2026).

15. Drug-drug interactions prediction based on deep learning and knowledge graph: A review / H. Luo et al. *iScience*. 2024. Vol. 27, no. 3. 109148. URL: https://doi.org/10.1016/j.isci.2024.109148 (date of access: 05.05.2026).

16. Deep learning for drug-drug interaction prediction: A comprehensive review / X. Li et al. *Quantitative Biology*. 2024. Vol. 12, no. 1. P. 30–52. URL: https://doi.org/10.1002/qub2.32 (date of access: 04.04.2026).

17. Al-Rabeah M. H., Lakizadeh A. Prediction of drug-drug interaction events using graph neural networks based feature extraction. *Scientific Reports*. 2022. Vol. 12. 15590. URL: https://doi.org/10.1038/s41598-022-19999-4 (date of access: 07.04.2026).

18. Dual-Channel Learning Framework for Drug-Drug Interaction Prediction via Relation-Aware Heterogeneous Graph Transformer / X. Su et al. *Proceedings of the AAAI conference on artificial intelligence*. 2024. Vol. 38, no. 1. 27777. URL: https://doi.org/10.1609/aaai.v38i1.27777 (date of access: 24.04.2026).

19. Wang Y., Yang Z., Yao Q. Accurate and interpretable drug-drug interaction prediction enabled by knowledge subgraph learning. *Communications Medicine*. 2024. Vol. 4. 59. URL: https://doi.org/10.1038/s43856-024-00486-y (date of access: 05.04.2026).

20. DrugBank 6.0: the DrugBank Knowledgebase for 2024 / C. Knox et al. *Nucleic Acids Research*. 2024. Vol. 52, no. D1. D1265–D1275. URL: https://doi.org/10.1093/nar/gkad976 (date of access: 09.04.2026).

21. SmileGNN: Drug–Drug Interaction Prediction Based on the SMILES and Graph Neural Network / X. Han et al. *Life*. 2022. Vol. 12, no. 2. P. 319. URL: https://doi.org/10.3390/life12020319 (date of access: 19.04.2026).

22. ExDDI: Explaining Drug-Drug Interaction Predictions with Natural Language / Z. Sun et al. *Proceedings of the AAAI Conference on Artificial Intelligence*. 2025. Vol. 39, no. 24. URL: https://doi.org/10.1609/aaai.v39i24.34709 (date of access: 22.04.2026).