КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

ІМЕНІ ТАРАСА ШЕВЧЕНКА

НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ВИСОКИХ ТЕХНОЛОГІЙ

Завідувач кафедри супрамолекулярної хімії

д.х.н., проф. Рябухін Сергій Вікторович

Протокол №____ засідання кафедри

від “____” _____20___ р

МЕТОДИ ПОШУКУ ТА АНАЛІЗУ ХІМІЧНИХ РЕАКЦІЙ У ВЕЛИКИХ БАЗАХ ДАНИХ

Випускна кваліфікаційна робота бакалавра

студента спеціальності 102 хімія

ОП «високі технології (хімія та наноматеріали)»

Савіцького Данила Андрійовича

Науковий керівник

Доктор хімічних наук

Волочнюк Дмитро Михайлович

Оцінка захисту роботи

_____

Київ − 2024 р

АНОТАЦІЯ

Савіцький Д.А. Методи пошуку та аналізу хімічних реакцій у великих базах даних. – Випускна кваліфікаційна робота магістра за спеціальністю 102 хімія (високі технології) ОП «високі технології (хімія та наноматеріали)»

У роботі створено алгоритм та основу для ПЗ, мета якого пошук реакцій в базах даних з врахуванням атом-атом мапінгу. Для цього було використано конденсований граф реакції, який є одним з методів репрезентації реакції через реакційний центр. Випробовування методу було проведено на базі Open Reaction Database (з англ. – відкрита база даних реакцій). Внаслідок пошуку роботи були сформовані датасети для реакцій деоксифторування та амідного сполучення

Ключові слова: атом-атом мапінг; конденсований граф реакції; пошук реакцій; реакційна інформатика.

ЗМІСТ

ВСТУП

Розвиток комп'ютерних технологій вплинув на всі сфери дослідження, від фізики до біології та навіть соціальних наук. Хімія, як наука та сфера дослідження здатна продукувати величезні обсяги інформації. В органічній хімія, кількість можливих сполук та шляхів до їх синтезу може сягати астрономічних масштабів. В 2014 році CAS (Chemical Abstracts Service, Хімічна реферативна служба) Registry анонсувало що їхня база даних містить понад 100 мільйонів сполук. У 2014 році було додано більше сполук, ніж з 1965 по 1990 рік. Тільки завдяки розвитку комп'ютерних технологій та появи хемоінформатичних методів дослідження ми можемо аналізувати, класифікувати та отримувати нові знання з цих даних.

Важливою гілкою хемоінформатики є реакційна інформатика (з англ. reaction informatics) яка розробляє інструменти та методи для аналізу великих обсягів реакцій та прагне оптимізувати і трансформувати сферу органічного синтезу. Синтез бажаної хімічної сполуки є одним із головних завдань синтетичної органічної хімії тому пошук оптимального синтетичного маршруту, вибір умов реакції, оцінка вибірковість реакції також є важливим завданням цієї гілки хемоінформатики. Традиційно, вирішення таких завдань вимагає хорошого досвіду, глибоких знань про природу реакції та високої кваліфікації від хіміка. Тому задля полегшення цього завдання розробляються моделі на основі машинного навчання та ведеться аналіз даних у пошуках закономірностей.

Проте видобуток та аналіз даних на великих масштабах стає нетривіальною задачею в рамках будь якої бази даних, оскільки в них завжди присутні помилки, недостовірні записи й т.д.. Візуалізація отриманих результатів також непростим завданням, тому що сама природа хімічного

перетворення є комплексна та може описуватися великою кількістю параметрів.

Об'єктом даного дослідження є бази даних хімічних реакцій та видобуток з них необхідних реакцій.

Предметом є написання програми та розробка алгоритму для ефективного методу пошуку в базах даних та аналізу отриманих даних.

Метою даної роботи:

1) Розробка алгоритму який здатний у ефективно та швидко відшукувати задані реакції за допомогою хімічного перетворення.

2) Валідація методу пошуку та видобуток структур реагентів, субстратів, та розчинників для реакції.

3) Формування підмножини реакцій певного типу та їх аналіз, та підготовка цих даних для машинного навчання.

Актуальність роботи полягає в застосуванні отриманих даних в для покращення існуючих методів органічного синтезу та комп'ютерних моделей.

Особистий внесок: літературний огляд, написання програмного коду та аналіз отриманих підмножин були виконані здобувачем освіти особисто. Розробка алгоритму для пошуку реакцій була здійсненна у співпраці з студенткою Erasmus Mundus Joint Master ChEMoinformaticsPlus Бойко Іриною Богданівною.

1. ЛІТЕРАТУРНИЙ ОГЛЯД

Хімічна реакція є більш складним об'єктом, ніж окремі молекули, оскільки вона включає три типи молекул: реагенти, продукти та інші добавки (реагенти, каталізатори або розчинники). Властивості реакції мають складну інваріантну симетрію. Властивості реакції (результат, швидкість тощо) є інваріантними щодо порядку молекул, поданих як реагенти чи продукти. Результат реакції залежить від умов реакції, включаючи фізичне (температура, тиск) та хімічне (розчинники, каталізатори тощо) середовище. Часто порядок і швидкість додавання реагентів до реакційної суміші, а також концентрація реагентів можуть впливати на результат. Проте варто звернути увагу, що дані про реакції в публічних базах даних такі як USPTO (United States Patent and Trademark Office з англ. – Відомство по патентам та товарним знакам США) чи ORD[1], [2] (Open Reaction Database з англ. - Відкрита база даних реакцій) зазвичай неповні: деякі умови можуть бути не зазначені. Більше того, більшість реакцій є незбалансованими, тобто деякі реагенти або продукти можуть бути пропущені в рівнянні реакції. Бази даних майже ніколи не містять інформації про негативні реакції, оскільки базуються на наукових публікаціях[3].

IUPAC визначає[4] реактант як "речовину, що споживається в процесі хімічної реакції", а продукт як "речовину, що утворюється під час хімічної реакції". Реагенти - це молекули такі як розчинники та каталізатори, які не вносять атом в продукти, але є необхідними для здійснення реакції[5]. Атоми реактантів, які змінюють свої зв’язки і/або формальний заряд під час перетворення, утворюють центр реакції. Для ідентифікації центру реакції необхідно встановити одноразову відповідність між атомами реагентів і продуктів, що називається атомно-атомним мапінгом (AAM). AAM (Рисунок 1 містить приклад процесу (адаптований з статті [6]) встановлює

відповідність між атомами реагенту і продукту: відповідні атоми повинні мати однакові номери. Разом із реагентами та продуктами можуть бути зазначені інші хімічні сполуки, що називаються реагентами, добавками або агентами. Зазвичай це стосується каталізаторів або каталітичних систем, реагентів, розчинників, каталітичних отрут або активаторів, комплексоутворюючих агентів, редокс-агентів, кислот або основ Льюїса чи Бренстеда. Реагенти або добавки разом із умовами зазвичай вказуються під стрілкою або в текстовому описі реакції. Таким чином, деякі реагенти, котрі в органічні хімії мають назву реагент (наприклад, реагент Вітіга), в хемоінформатиці насправді є реактантами в хімічних рівняннях, що створює проблеми в дослідженні певних типів реакцій. Наприклад, реакції деоксифторування, такі реагенти як DAST будуть записані як реактанти, що ускладнює процедуру їх ідентифікації.

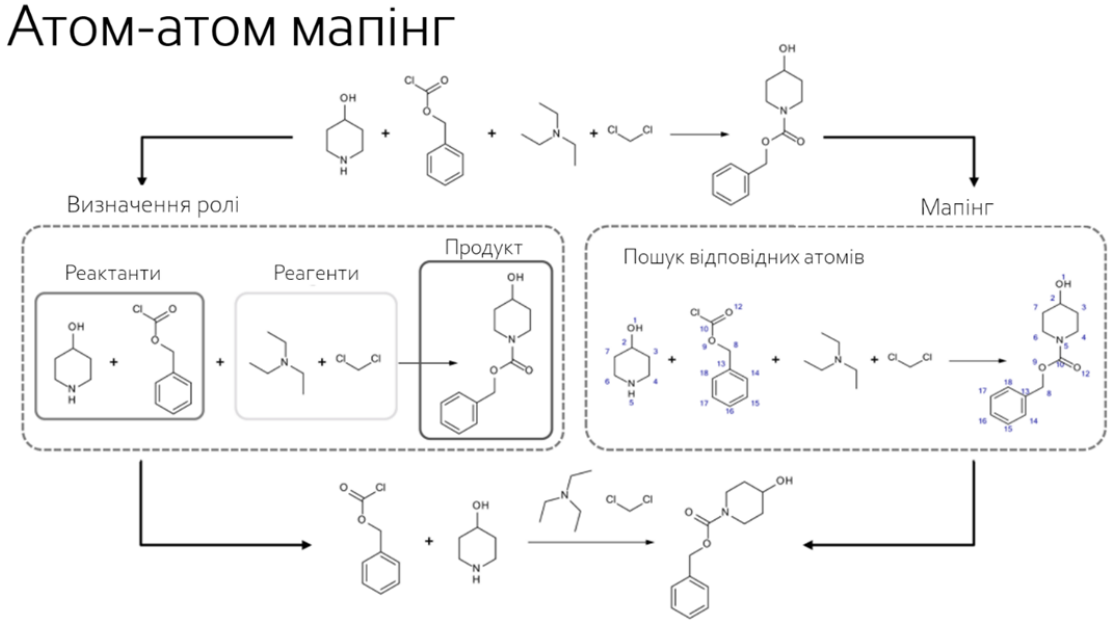

Рисунок 1 - схема AAM

Проте щодо самого запису реакції можна виділити три основні підходи подання реакцій, які кодують реакцію як:

(1) Набір реагентів та продуктів,

(2) Різниця між продуктами та реагентами

(3) Характеристики реакційного центру та його оточення.

Перший підхід є досить інтуїтивний, адже ми його зустрічаємо постійно в органічні хімії. Він дозволяє нам записати в рівняння та зробити припущення чому реакція проходить та які важливі структурні елементи субстрату відіграють у перебігу реакції. Також він є досить узагальненим, щоб записати будь-яке хімічне перетворення. В поєднанні з атомним мапінгом цей метод найбільш використовуваний в хемоінформатиці. Така репрезентація часто записується за допомогою SMARTS (SMiles ARbitrary Target Specification), розширенням SMILES[7](simplified molecular-input line-entry system з англ. спрощена система молекулярного введення) для реакцій. Але окрім цього методу, можливо запис через, RInChI1 (розширення для InChI (International Chemical Identifier з англ. - міжнародний хімічний ідентифікатор)

Ідея запису різниці між продуктами та реагентами як спосіб подання реакції зародилась ще з початків реакційної інформатики, а саме з кінця 1930 років, задовго до появи перший комп’ютерів. Навіть на той час, органічним хімікам було важко працювати з усім обсягом накопиченої інформації, який потребував систематизації та класифікацій типів реакцій. Тому 1938 році була запропонована[8] перша систему класифікації органічних реакцій Конрадом Вейгандом з Лейпцігського університету (Німеччина). Класифікація базувалася на сформованих та зруйнованих зв’язках, й навіть наразі така концепція, модифікована, має застосування. Пізніше, на основі минулої системи, Вільям Тілгеймер розробив свою[9], де розглядав чотири основні класи хімічних перетворень, а саме: (1) приєднання, (2) елімінація, (3) перегрупування та (4) заміщення. Реакції були записані текстовим

рядком, котрий перераховував атоми сформованих зв’язків перед стрілкою вгору та атоми зруйнованих навпаки стрілкою вниз. З розвитком комп’ютерних технологій були запропоновані нові ідей як зберігати хімічне перетворення в пам’яті комп’ютера таким методом, наприклад, за допомогою R-матриці[10].

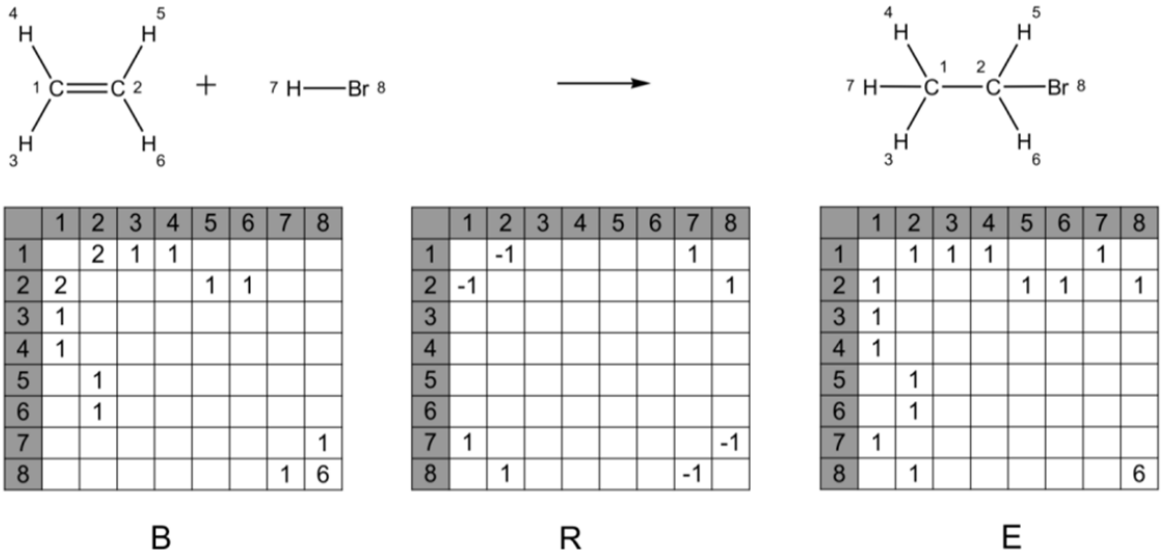

Рисунок 2 - Угі-Дугунджі (R-матриця) представлення реакції гідробромування етилену. Нулі опущені для зручності

В середині 1970, Угі та Дугунджі запропонували використовувати такі матриці для опису реакцій. Цей метод представляє молекулу субстрата (B) та продукта (E) в формі матриць зв'язків. Для цього потрібно, щоб реагенти та продукти мали рівну кількість складових атомів. Це також вимагає правильного картографування атомів. Потім R-матриця визначається як різниця між матрицями E і B: R = E – B. Позитивні значення діагоналі отриманої матриці відображають збільшення порядку зв’язку або його формування. Негативні значення відображають зменшення порядку зв’язку або його розрив. Діагональні значення відображають зміни кількості

незв’язаних пар електронів (Рисунок 2). Такий метод має недоліки, а саме сильна залежність від якості атом-атом мапінгу.

Третій спосіб запису реакції через реакційний центр, де під центром реакцій розуміють набір атомів, які змінюють принцип за яким вони зв’язані або змінюються зв’язки, які суміжні до них в ході реакції. В перше така ідея була запропонована Владутцом [11], а саме: представляти центр реакції графіком скелета реакції в яких накладені на однин одного реагенти та атоми продукту, і межі такої суперпозиції характеризують трансформації в ході реакції. Проте ідея стала популярна після публікації Фуджіти[12], в котрий запропонував репрезентацію реакції одним графом, котрий би зображував «уявний» перехідний стан. Ідея такого «уявного» перехідного стану лягли в основу Condensed Graph of Reaction (CGR з англ. стислий граф реакції) завдяки низки праць[13], [14]. Приклад такої репрезентації можна спостерігати на Рисунку 3. В праці Варнека[15] запропоновані розглядати CGR як псевдо-молекулярний граф, для якого можна створити перелік молекулярних дескрипторів, котрі, по суті, будуть дескрипторами для реакції. CGR можна використовувати для пошуку реакції по типу перетворення, оскільки він підтримує підструктурний пошук. Він також залежить від якості AAM, але на відміну від, наприклад, R-матриці не вимагає однакової кількості атомів в продуктів та реактантів. Проте, він не зберігає інформацію про реагенти (в класичному розумінні), оскільки сама реакція записана як один перехідний стан

Конденсовані графи реакцій використовувались для зберігання, пошуку, аналізу, візуалізації реакцій та побудови моделей з використанням машинного навчання. CGRs можуть бути побудовані за допомогою CGRTools[16] бібліотеки у мові програмування Python.

Наприклад, Деланне та Ніклаус[17] використали CGRs для розробки ReactionCode: інструменту для кодування та декодування реакцій в багатошаровий машино зчитуваний код. Цей формат є гнучким і може використовуватися в контексті пошуку реакції за подібністю та класифікації.

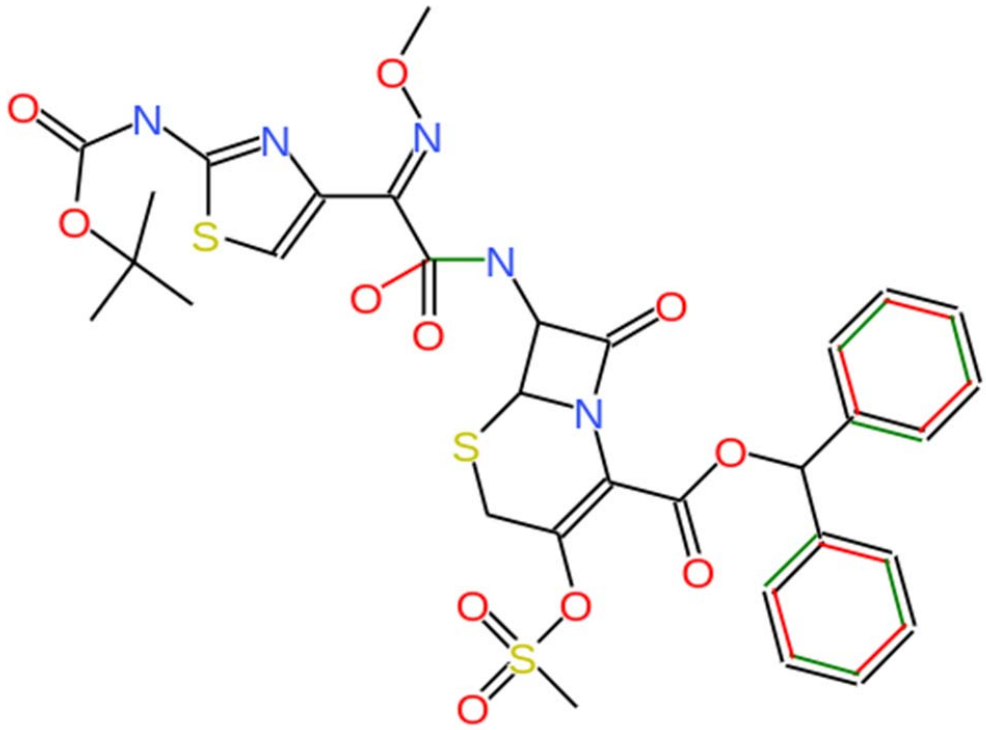

Рисунок 3 - Приклад реакції амідного каплінгу та CGR який йому відповідає(нумерація атомів була забрана для картинки). В CGR розірванні зв'язки зображенні як [->.](червоний колір) та створений C-C зв'язок [.>-](зелений)

Недоліком Конденсованих графів реакції можна вважати їх залежність від атом-атом мапінгу. Порівняльний аналіз інструментів для AAM показав відносну неефективність багатьох традиційних маперів, що базуються на вручну введених правилах[18]. Проте, в останні роки, з розвитком машинного навчання, з'явилось кілька інструментів заснованих на ML-моделях [19]. Вони показують кращий результат — близько 84% коректно

оброблених реакцій для RXNMapper[20] в даній публікації. Швидкість обробки даних також грає роль в таких завданнях.

Існує чимало публікацій на предмет дослідження певних типів реакції та їх параметрів ефективності, котрі, мають на меті знайти приховані закономірності, або віднайти найкращі методики. Серед них існує низка публікацій[21], [22], котрі проводять вручну огляд літератури для пошуку необхідних даних оскільки такі дані можуть гарантувати точність, та повноту записів. Проте такі дослідження вимагають багато зусиль та вручну опрацьованих публікацій. Інший тип публікацій застосовують комп'ютерні методи для виконання таких завдань, аналізуючи величезні масиви інформації. Мета таких публікацій це зазвичай оптимізація реакційних умов, та підбір каталізатора чи реагенту[23], [24] [25]. В останні роки набуває популярності спроби[26], [27] залучити машинне навчання або нейронні мережі для QSRR (quantitative structure-reactivity relationship, з англ. – кількісне структура-реактивність співвідношення). Мета QSRR це передбачення параметрів, реакцій та властивості. Частину таких параметрів, такі як кінетичні, та термодинамічні чудово передбачаються квантовохімічними обчисленнями, навіть існують вже інструменти, котрі можуть такі параметри системи обчислювати на потоці [28], [29], [30]. Але якщо передбачити чи реакція теоретично буде проходити не є важким завданням для квантовохімічних методів, то зрозуміти наскільки (в сенсі, наприклад, такого параметру як вихід) це вже є не тривіальним завданням. Тому набрав популярності так званий «data-driven» підхід (з англ. – направлений даними). В його основі це використання великих датасетів для навчання моделей машинного навчання, або для того щоб аналізувати дані "en masse" для розуміння інших комплексних параметрів. Прикладом, такого комплексного параметру, можна вважати реагент, котрий використовується в

реакції або каталізатор[31].Такий підхід до погляду на дані з точки зору реагентів отримав назву reagent-driven[5] (з англ. – направлений реагентами). Взяти до прикладу, реакцію амідного сполучення, для котрої існує чимало можливих варіантів реагентів, від TBTU до HATU. Проте сучасне визначення реагенту, котре використовується в базах даних, ускладнює нам пошук конкретних типів регентів, оскільки відносить їх до реактантів [6] (наприклад, реагенти для деоксифторування). Й хоч існують інакші методи віднесення ролей в реакції, як от fingerprint-based[6] (з англ. – на основі "відбитку"), він демонструє змішенні результати.

Хоч на разі «data-driven» підхід є популярним, він має значну проблему на даний момент[31], [32]. Зокрема, це наявність даних, їх якість та загальна кількість записів. Чимало реакцій мало репрезентовані в базах даних або дані не є варіативними, а скоріш одноманітними. Якщо, наприклад, 80 відсотків записів стосується субстратів, котрі мають малу різницю в своїй будові або є гомологами, такі дані вважають малоінформативними. Інша проблема, це доступність даних, у вільному доступі. Наявно дуже малий відсоток в порівнянні з такими базами як CASREACT[33] чи Reaxys[34]. Такий стан речей якраз зумовив появу ORD[1], [2]. Це датабаза є доступною для всіх, та має в собі приблизно 2 мільйона реакції з USPTO (період з 1976 по 2016) та певну частину реакцій які були надані добровільно дослідниками. Їх репозиторій загалом складається з 2.2 мільйона реакцій, а сама база даних розрахована бути оптимальною для машинного навчання. Вони мають свій формат запису даних (Рисунок 4), але абсолютна більшість записів на даний момент не є повними, й містять тільки дані про вихід. Також сама база даних пропонує дуже базовий спосіб пошуку записів в собі, який повертає не всі записи.

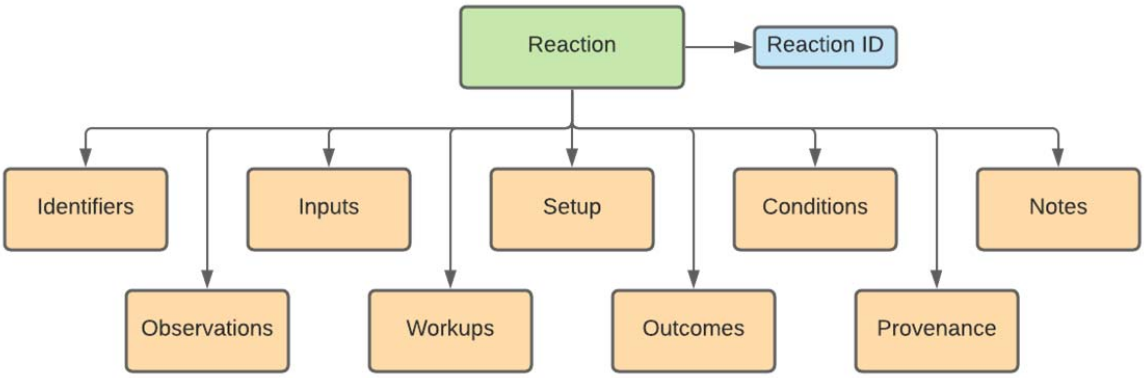

Рисунок 4 - Схема запису даних в ORD

ORD має також вже декілька сформованих датасетів, в котрий містяться вже готові дані для навчання алгоритмів, але їх розміри обмежені. Зазвичай використовується певний тип реакції для формування датасетів для машинного навчання, тому необхідно мати метод, котрий:

1) Здатний відфільтрувати некоректні записи
2) Віднайти певний тип реакції який ми маємо задати
3) Підготувати датасет, віднести складові реакції на субстрат та реагент, та розрахувати хімічні дескриптори для субстрату.

2. РЕЗУЛЬТАТИ ТА ОБГОВОРЕННЯ

2.1 Процес AAM та формування чистої бази даних

Дані для роботи були взяті з Open Reaction Database, репозиторію якого можна отримати на сайті GitHub. ORD містить по більшій частині дані з USPTO, але має ще додаткові записи з інших джерел. Поверхневий аналіз самих даних та ознайомлення з літературою підтвердило, що AAM реакцій не є якісним в ORD, й в переважній більшості записів не можливо було віднайти за допомогою пошуку самим же інструментом від розробників ORD. Для вирішення цієї проблеми ми застосували методику з наукової праці, яка дозволяє очистити початковий мапінг та створити новий з використанням RXNutils. Після застосування цієї методики ми покращили якість AAM, а також очистили частину хибних записів, що надзвичайно важливо для конденсованих графів реакції.

Використання вже існуючого функціоналу ORD для роботи над даними не надає нам жодних переваг. Ми не використовуємо схему запису даних ORD (Рисунок 4), а сам пошук в базі відбувається без врахування атом-атом мапінгу.

2.2 Розробка алгоритму

Якщо сформувати завдання по пунктах, ми хочемо знайти реакції певного типу в датабазі, та зробити це ефективно. Класичний спосіб це використання бібліотеки RDKit для мови програмування Python для того, щоб перетворити SMARTS у вбудований клас реакції, котрий має вже готові методи для розбиття реакції на реактанти, реагенти та продукти, або дозволяє робити підструктурний пошук для перетворення. Загалом, цей метод є простим, але він не враховує AAM під час пошуку підструктури. Й хоч RDKit є відносно добре документованим та на даний момент чудово оптимізований для великих баз даних, алгоритм повністю створений на його основі буде повертати багато хибних результатів. Також метод ускладнений тим, що наш підстуктурний пошук має бути записаний в SMARTS, який не інтуїтивний, й є досить комплексним, якщо ми хочемо передати складні перетворення чи специфічні субстрати. Він також не є швидким, на жаль, оскільки використовує декілька методів одразу, щоб розбивати реакцію на продукти та реактанти, що стає більш помітно при пошуку у великих об’ємах.

Більш кращим для цього буде використання конденсованого графу реакції, котрі одразу відфільтрують реакції з неправильним атом-атом мапінгом та підструктруний пошук по яким досить точний, в додачу сам метод пошуку є досить швидким. Єдиний недолік це довгий й затратний в плані пам’яті процес конвертації реакції з SMARTS в конденсований граф реакції.

Тому було прийнято до розгляду такий алгоритм, котрий базується на тому, що ми маємо спростити собі пошук, зменшивши його обсяг, щоб найбільш повільний етап займав якомога менше часу. Принципова схема на Рисунку 5

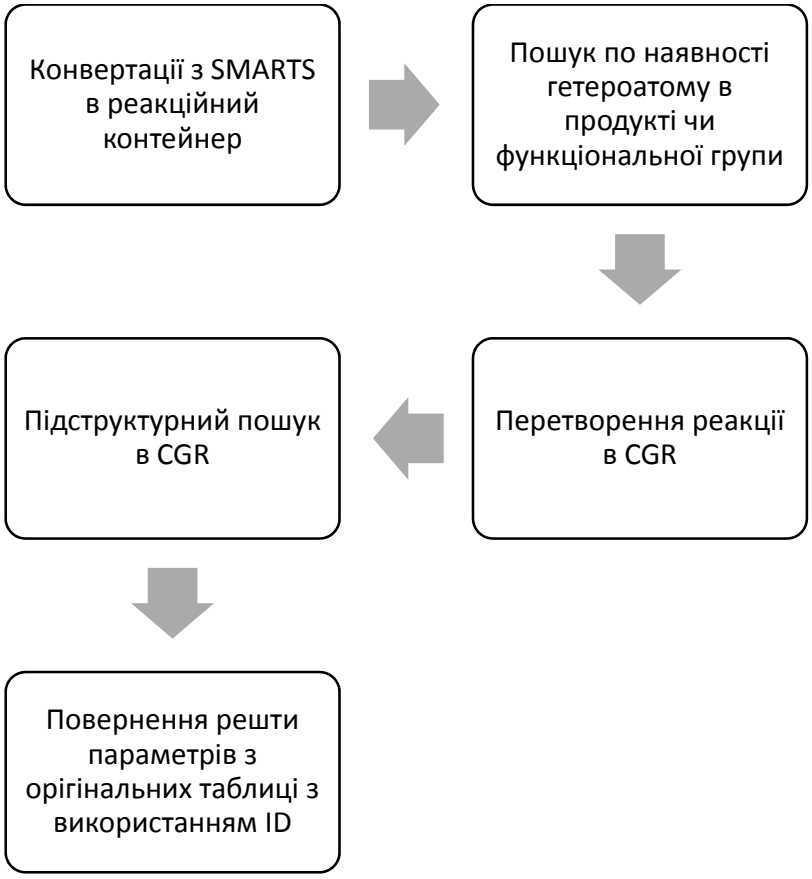

Рисунок 5 - Принципова схема алгоритму

Також варто мати на увазі, обсяги пам’яті, котрі будуть використовуватися в процесі. Ми не можемо одразу взяти та провести конвертацію всіх записів у реакційний контейнер чи конденсовані графи реакції, оскільки не вистачить пам’яті для операційної системи та система просто «зависне». Також потрібно мати увазі, що мова програмування Python повертає пам’ять яку було використано у фрагментованому стані, котрий, ускладнює використання цієї пам’яті операційною системою. Частково, ми можемо уникнути цю проблему за допомогою кешування змінних у пам’яті комп’ютера, але це також має свій ліміт. Саме тому, було використано класичний метод розбивання датасету на декілька менших, рівних за обсягом. Це разом з кешуванням дозволяє нам уникнути будь-яких проблем з

пам’яттю. Емпірично було з’ясовано що оптимально для мого персонального комп’ютера (див. Додатки для системних параметрів) була кількість в 10 порції. Ці порції формуються з початкового датасету, звідки беруться тільки ID та SMARTS для реакції. ID потрібно, щоб потім повернути інформацію про вихід та рік.

Далі відбувається перетворення кожної з цих порцій в реакційний контейнер за допомогою RDKit. Це робиться, щоб за допомогою підструктрурного пошуку знайти певні елементи, котрі ми очікуємо в продукті. Хоч CGRtools має схожий функціонал, але він менш оптимізований, тому повільніший в процесі перетворення, й більш ефективно для цього використати RDKit, в додачу, нам не є важливим в цей момент AAM. Проте насправді цей крок опціональний, оскільки, ми не завжди можемо виділити елемент, котрий буде в продукті. Візьмем для прикладу деоксифторування та реакції утворення зв’язку Карбон-Карбон: в першому випадку ми можемо перевірити, чи в продукту є атом Фтору, (й хоч це не завжди означає, що він там присутній внаслідок реакції, але це набагато менший обсяг реакцій) проте в другому випадку не можливо виділити якийсь такий патерн. Тому це опціональний крок, але для більшості хімічних перетворень він працює.

Далі після цього відбувається конвертація в конденсований граф реакції, й цей процес займає найдовше часу, близько 40% процентів часу для опрацювання однієї порції, тим не менш, на далі підструктурний пошук надалі відбувається набагато швидше. Також важливим є момент, котрий вже згадувався вище, що ми відокремлюємо реакції з неправильним AAM, котрі ми можемо потім проаналізувати за їх маючи їх ID.

Підструктурний пошук відбувається за допомогою створення підстуктури конденсованого графу реакції, який після ініціалізації ми наповнюємо

атомами та прописуємо зв’язки між ними. В своїй суті, такий граф можна розуміти як «уявний» перехідний стан, котрий описується однією молекулою. В разі необхідності, можна прописати такі параметри як гібридизація, чи кількість сусідів(дивись методи для більш детального огляду). На даний момент, робота містить функції, котрі містять вже заздалегідь написані правила для таких підструктур, але надалі планується створити функціонал, який буде дозволятиме користувачу уводити свої підструктури для пошуку. Приклади конденсованих графів реакції на Рисунку 6.

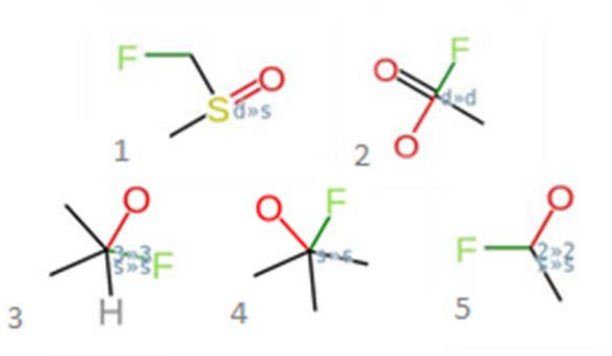

Рисунок 6 - Приклади конденсованих графів реакції для підструктурного пошуку деоксифторуючих реакцій 1) Для Сульфоксиду 2) Для утворення ацилфлуориду 3) Для вторинного спирту 4) Для третинного сприту 5) Для первинного спирту

Останній етап необхідний нам щоб доповнити решту інформації з початкової бази даних (вихід, рік), та виділення субстратів з реакції. Виділення субстрату легко можна провести за допомогою розділення конденсованого графу реакції. Але у випадку, якщо він складається з субстрату та реагенту(в цьому випадку, доречніше сказати реактанту) то тоді ми перевіряємо який з них має більшу масу та число атомів. Це примітивний

метод, але в більшості випадків повертає правильний результат. Альтернативний спосіб, котрий є більш надійний, це сформувати наперед список реагентів, проте він вимагає створення окремої бази даних реагентів. Й хоч цей варіант гарантує абсолютне віднесення, створення такої бази вимагає додатковий час та ресурсів, й планується в майбутньому. Далі для субстратів ми розраховуємо 14 дескрипторів, котрі, ми додаємо в наш датасет. Ці дескриптори були вибрані таким чином, щоб демонструвати структуру, її «складність» субстрату та топологічну складність графу, та правило Ліпінського (точний перелік в дескрипторів в методах). Отриманий хімічний простір можна візуалізувати за допомогою T-розподілення вкладення стохастичної близькості, котре часто використовують для візуалізації даних, коли параметрів системи багато, й необхідно знизити розмірність системи. На Рисунку 7 можна побачити приклад такого графіку

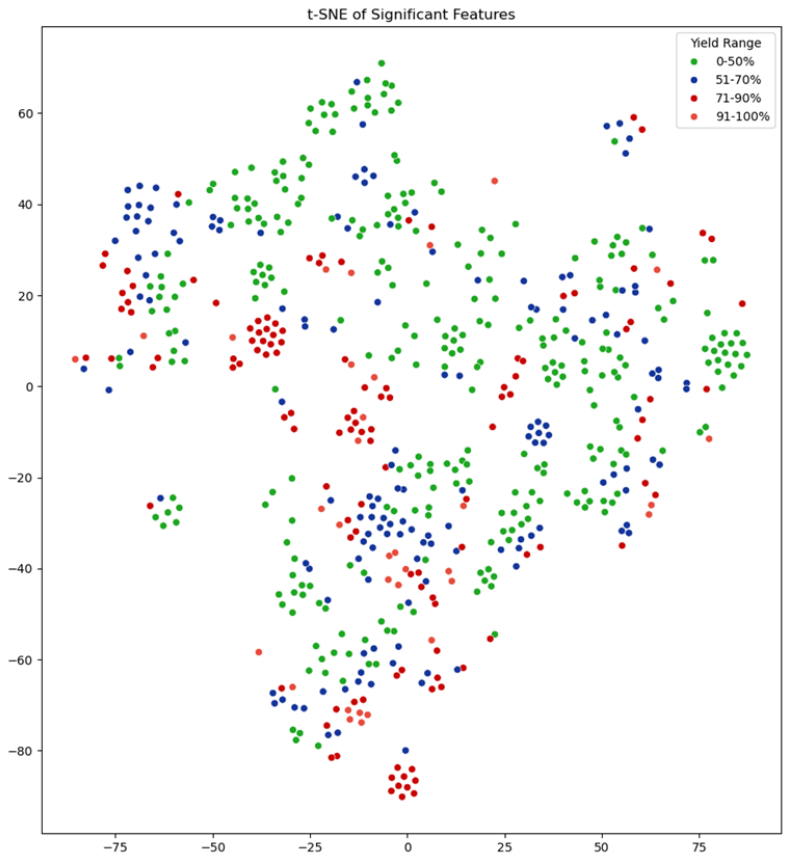

Рисунок 7 - Т-розподілення вкладення стохастичної близькості для реакцій деоксифторування.

Таке представлення дозволяє нам оцінити різноманіття даних, та схожість субстратів за будовою, тому що кожній точці відповідає певна структура в базі. На Рисунку 8 ми бачимо утворення кластерів, з сполук, котрі дуже схожі за будовою, й завдяки різним кольорам точок відповідно їх вихід у реакції.

2.3 Валідація методу

Після створення алгоритму та його реалізації у програмі, необхідно зрозуміти наскільки він ефективний. Спершу було створено невеликі за обсягом датасети з реакції деоксифторування. Він був сформований з даних USPTO, та складається з результатів пошуку з використанням RDKit. Й хоч він не підтримує пошук реакцій за допомогою AAM, ми можемо обійти цю проблемо. Замість шукати необхідне нам хімічне перетворення, ми можемо шукати реагент котрий необхідний для цієї реакції. У випадку деоксифторування, наприклад, можемо взяти такі популярні реагенти як DAST чи суміш триетиламоній:фторидна кислота.

Після реалізації такого пошуку, було повернуто відносно небагато реакцій (563 записів) з всього датасету, та процес зайняв приблизно 20 хвилин для всієї бази даних. Після було сформовано запит з використанням розробленого алгоритму та в результаті отримано 504 реакції. Після детального огляду реакцій, які не були отримані, стало зрозуміло, що 59 реакцій містять помилки. На Рисунку 8 можна спостерігати 2 приклади таких помилок.

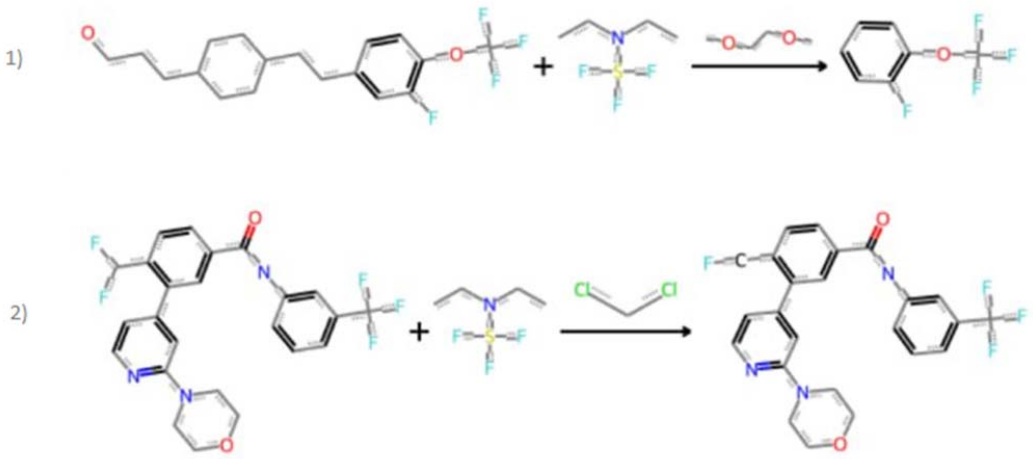

Рисунок 8 -Приклад помилкових записів в тестових датасетах

Після цієї перевірки, було запущено пошук з використанням алгоритму по всій базі даних. Конденсовані графи реакції котрі взяті для цього пошуку присутні на Рисунку 7. Процес зайняв приблизно 11 хв та повернув таку кількість реакцій (для відповідних субстратів):

- Первинний спирт: 290 реакцій
- Вторинний: 292 реакцій
- Третинний: 187 реакцій
- Кетон:10 реакцій
- Карбоксильна кислота:35 реакцій
- Сульфоксид: 8 реакцій

Як можна спостерігати, використання конденсованих графів реакції є досить ефективним методом пошуку хімічного перетворення не тільки зору точності, але й швидкості.

Аналогічно було протестовано пошук для реакцій амідного сполучення. Спершу реалізували пошук з використанням тільки RDKit для таких реагентів як HBTU, EDC, та оксалілхлорид. Отримано 10263 реакцій, з яких алгоритм з використанням конденсованих графів реакції повернув 7163 реакцій, решта 3100 реакцій або не є реакціями амідного сполучення (88 відсотків) решта має помилку (11%) в записі або не формували конденсований граф реакції (1%). З них:

- Реакції за участі EDC: 5626 записів
- Реакції за участі оксалілхлориду: 598 записів
- Реакції за участі HBTU: 939 записів

2.4 Аналіз даних

Було виведено гістограму для кількість записів відповідно до року, коли було зроблено запис (Рисунок 9). З нього чітко видно як експоненційно зростає кількість записів, які в основному, з патентів USPTO. Це відповідає загальному тренду в всіх базах даних, в яких загальна кількість записів зростає таким же темпом.

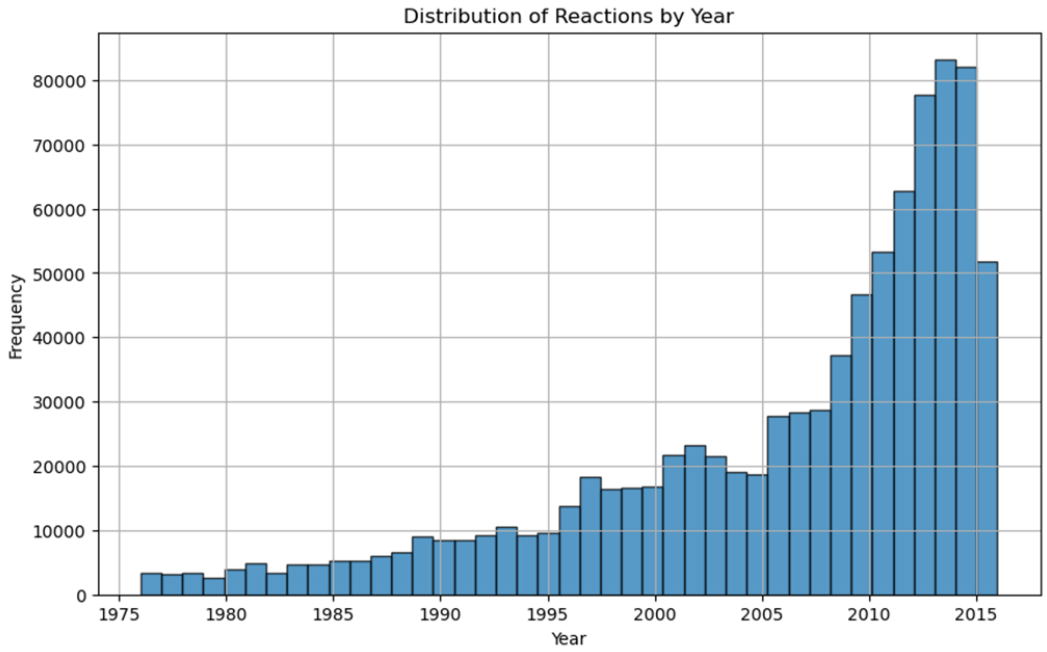

Рисунок 9 - динаміка кількості записів з 1976 – 2016 рік

Розглядаючи дані, котрі були отриманні в результаті пошуку, ми можемо спостерігати таку ж тенденцію на Рисунку 11.

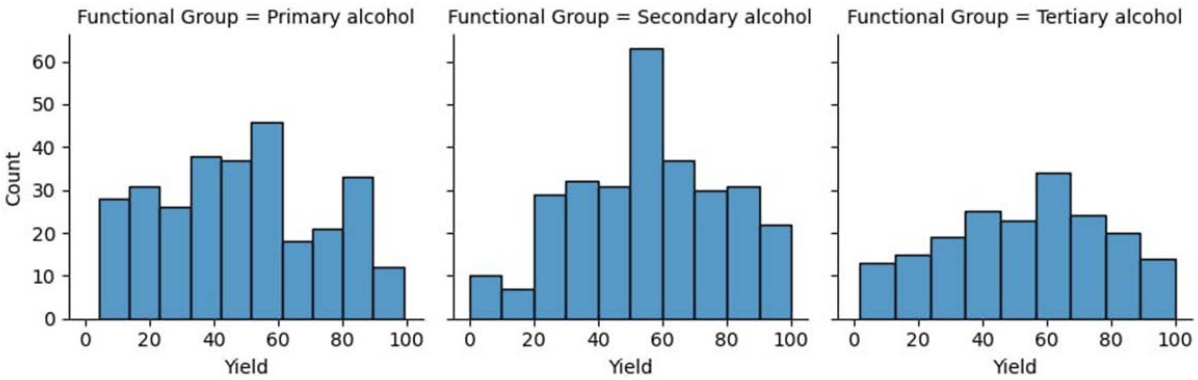

Рисунок 10 - Розподіл виходів залежно від типу субстрату, з ліва на право : первинні, вторинні, третинні

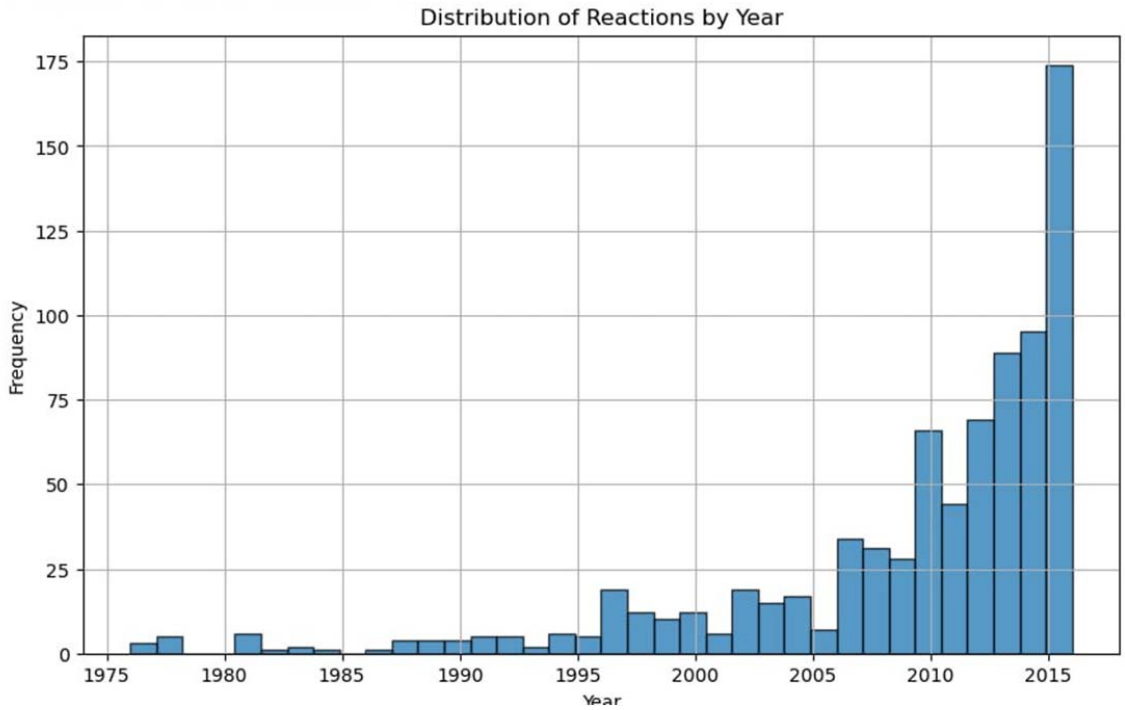

Рисунок 11 - Динаміка кількості записів з 1976 – 2016 рік для реакцій деоксифторування

Для реакцій деоксифторування спирту (768 реакцій) найпопулярнішим реагентом виявився DAST (всього 568 реакцій) далі суміш триетиламонію та фторидної кислоти (98 реакцій) й третє місце тетраамоній фторид (43 реакції). Решту 60 реакцій містять багато різноманітних реагентів, проте які

важко виділити за їх кількістю. На Рисунку 10 можна побачити розподіл виходів залежно від субстрату.

Пошук по всій базі даних для амідного сполучання повертає нам 32590 записів, проте лиш 9209 реакцій для них містять реагент, тому решту ми відкидаємо.

Розподіл по реагентам для реакцій амідного сполучання можна спостерігати на Рисунку 12. Варто відмітити, що найпопулярніший реагент є EDC та HBTU, та також якщо об’єднати тіоніл хлорид разом з оксалілхлоридом, то ми отримаємо 13 відсотків.

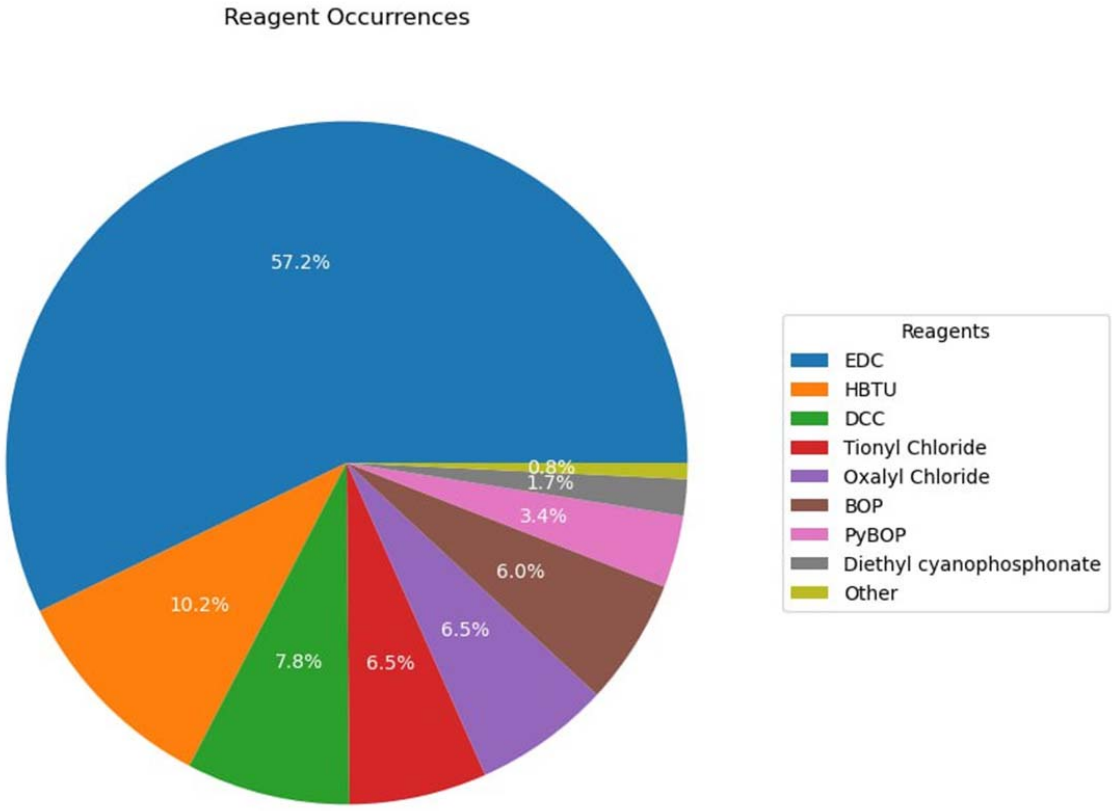

Рисунок 12 - Розподіл реагентів для реакцій амідного сполучення

TSTU: 65.58%	Тіоніл Хлорид: 62.10%	PyBOP: 56.59 %
BOP: 60.10%	EDC: 60.11%	Oxyma: 62.55%
COMU: 49.06%	HBTU: 56.42%	PyAOP: 48.87%
DCC: 58.22%	HCTU: 45.37%	PyBOP: 56.59
Диетилціанофосфонат : 63.02%	Оксалілхлорид: 56.28%	_

Таблиця 1 - Середній вихід для статистично значущих даних

Середній вихід для реакцій є цікавим, лиш 7 реагентів з 13 представлених мають вихід вище 50 % (Таблиця 1). Сформовані датасети надалі будуть надісланні в ORD для розгляду, щоб вони були опубліковані в цій базі даних

.

3. МЕТОДИ І МАТЕРІАЛИ

3.1. Походження даних

Реакції для подальшого аналізу було отримано з Open Reaction Database[1], [2]. Більшість записів походять із USPTO – бази даних реакцій з патентів США, отриманих з праці [35]. Початково, було отримано 2.2 мільйона реакцій. Для отримання більш якісних даних було використано бібліотеку ReactionUtils (rxnutils), що пропонує дані в очищеному вигляді, прибираючи частину записів з помилками в AAM та перероблює AAM (детальніше про метод AAM надалі). Після застосування цієї методики ми покращили якість AAM, а також очистили 10,856 хибних записів, що дозволило отримати більш точні та надійні результати для подальшого аналізу. Після цієї процедури, було прибрано дані котрі не містять ніякої інформації про вихід, та реакції котрі містять вихід понад 100 відсотків. В результаті отримуємо очищену базу даних з приблизно 903 727 записів, з якими можна вести роботу. Результат ми додатково обробляємо, забравши зайві стовбці, та додали стовбець року, коли було додано запис (ми можемо отримати це з номеру датасету в ORD). Вирішено було зберегти такі колонки, як «ID», «ReactionSmiles», «ReactionSmilesClean», «Year» та «Yield». «ReactionSmiles» містить реакцію, котра записана за допомогою SMARTS та містить AAM(«ReactionSmilesClean» містить те саме тільки без AAM).

3.2. Атом-атом мапінг

Атом-атомний мапінг було виконано за допомогою RXNMapper[36]за алгоритмом, описаним в документації ReactionUtils. Даний інструмент був тренований визначати атом-атомний мапінг з використанням ШІ-моделі ALBERT (A lite Bidirectional Encoder Representations from Transformers, двоспрямовані кодувальні представлення з трансформерів[37]), навченої некеровано на великому наборі даних хімічних реакцій.

3.3. Алгоритм пошуку

1. Перетворення реакцій зі SMIRKS у реакційні контейнери

Перетворення відбувається за допомогою бібліотеки RDKit, модуля Allchem (AllChem.ReactionFromSmarts(smarts). Для того щоб краще використовувати ОП було застосовано кешування з бібліотеки joblib для мови програмування Python.

2. Ізоляція підмножин реакцій за типом

«QueryCGRContainer()» є методом з бібліотеки CGRtools, котрий дозволяє нам проводити пошук за підструктурою конденсованого графу реакції. Нижче наведено приклад створення такого контейнеру для реакції деоксифторування первинних спиртів.

```
rule_primary = QueryCGRContainer()
rule_primary.add_atom("O", 1)
rule_primary.add_atom("C", 2, hybridization=1, p_hybridization=1, neighbors=2, p_neighbors=2)
rule_primary.add_atom("F", 3)
rule_primary.add_atom("C", 4)
rule_primary.add_bond(1, 2, DynamicBond(1, None))
rule_primary.add_bond(2, 3, DynamicBond(None, 1))
rule_primary.add_bond(2, 4, 1)
```

3. Підструктурний пошук за атомами або функціональною групою

Відбувається за допомогою RDKit та є опціональним етапом. Конвертація необхідного фрагементу в молекулярний контейнер проводиться методом Chem.MolFromSmiles(). У випадку використання функціональних груп береться SMARTS патерн для відповідної групи.

3. Перетворення реакцій на Конденсовані Графи

Відбувається за допомогою методу .compose() з бібліотеки CGRtools. Реакції котрі не можуть бути конвертовані видають помилку, але не зупиняють процес. ID проблемної реакції зберігається для майбутнього огляду.

4. ХІМІЧНІ ДЕСКРИПТОРИ ДЛЯ ПІДМНОЖИН ТА АНЛІЗ ДАНИХ

Були взяті з бібліотеки RDKit, нижче наведений список та пояснення для кожного х них

1. TPSA (Topological Polar Surface Area)

Обчислює площу полярної поверхні молекули, що є сумою поверхонь всіх атомів, які мають частковий заряд. Використовується для оцінки здатності молекул проникати через клітинні мембрани.

2. MaxAbsPartialCharge

Максимальний абсолютний значення часткового заряду на будь-якому атомі в молекулі. Це важливо для розуміння електронного розподілу.

3. MaxPartialCharge

Максимальний частковий заряд на атомі в молекулі. Допомагає визначити найбільш електронегативні атоми.

4. MinAbsPartialCharge

Мінімальний абсолютний значення часткового заряду на будь-якому атомі в молекулі. Використовується для оцінки розподілу зарядів у молекулі.

5. MinPartialCharge

Мінімальний частковий заряд на атомі в молекулі. Дозволяє визначити найбільш електропозитивні атоми.

6. FractionCSP3

Відсоток sp3-гібридизованих вуглецевих атомів у молекулі. Вказує на рівень насиченості та складність структури.

7. NumAliphaticCarbocycles

Кількість аліфатичних карбоциклів у молекулі. Аліфатичні карбоцикли є некон'югованими кільцевими структурами, що складаються тільки з вуглецевих атомів.

8. NumAliphaticHeterocycles

Кількість аліфатичних гетероциклів у молекулі. Аліфатичні гетероцикли містять принаймні один гетероатом (не вуглець) у кільцевій структурі.

9. NumAromaticCarbocycles

Кількість ароматичних карбоциклів у молекулі. Ароматичні карбоцикли є кон'югованими кільцевими структурами, що складаються тільки з вуглецевих атомів і мають ароматичні властивості.

10. NumAromaticHeterocycles

Кількість ароматичних гетероциклів у молекулі. Ароматичні гетероцикли містять принаймні один гетероатом у кільцевій структурі і мають ароматичні властивості.

11. NumHAcceptors

Кількість атомів у молекулі, які можуть приймати водневі зв'язки. Важливо для визначення водневих зв'язків і розчинності.

12. NumHDonors

Кількість атомів у молекулі, які можуть віддавати водневі зв'язки. Також впливає на здатність молекули утворювати водневі зв'язки і розчинність.

13. NumHeteroatoms

Кількість гетероатомів (атомів, які не є вуглецем або воднем) у молекулі. Важливо для оцінки хімічних і фізичних властивостей молекули.

14. NumRotatableBonds

Кількість обертових зв'язків у молекулі. Використовується для оцінки гнучкості молекули.

15. MolLogP

Розрахований логарифм коефіцієнта розподілу (октанол/вода). Використовується для оцінки гідрофобності молекули.

16. Chi0n

Константа валентного зв'язку Чи (0-го порядку) для несистематичних атомів. Використовується для оцінки структури молекули.

17. Chi0v

Константа валентного зв'язку Чи (0-го порядку) для систематичних атомів. Використовується для оцінки структури молекули.

18. Chi1

Константа валентного зв'язку Чи (1-го порядку). Використовується для оцінки топології молекули.

5 ВІЗУАЛІЗАЦІЯ ХІМІЧНОГО ПРОСТОРУ ЗА ДОПОМОГОЮ Т-РОЗПОДІЛЕНОГО ВКЛАДЕННЯ СТОХАСТИЧНОЇ БЛИЗЬКОСТІ

t-SNE (t-Distributed Stochastic Neighbor Embedding) - це метод нелінійного зниження розмірності, який особливо корисний для візуалізації багатовимірних даних. Основна ідея методу полягає в тому, щоб зменшити кількість вимірів даних, зберігаючи при цьому їх структуру та взаємозв'язки. Це досягається шляхом моделювання кожної високовимірної точки в низьковимірному просторі таким чином, щоб схожі точки залишались близькими одна до одної, а віддалені точки залишались далекими.

Процес t-SNE включає такі етапи: 1) Обчислення ймовірностей сусідства: У високовимірному просторі обчислюються ймовірності того, що точки є сусідами одна з одною.2) Проєкція на низьковимірний простір: Створюється низьковимірна проєкція точок, де також обчислюються ймовірності сусідства. 3) Мінімізація розбіжностей: Метод намагається мінімізувати розбіжності між ймовірностями сусідства у високовимірному та низьковимірному просторах за допомогою алгоритму градієнтного спуску.

Для проведення аналізу з використанням t-SNE було обрано 2 компоненти, що дозволяє візуалізувати дані у двовимірному просторі. Це робить результати легкими для інтерпретації та аналізу. У Python для цього можна використовувати бібліотеку sklearn.manifold, яка надає простий у використанні інтерфейс для застосування t-SNE.

ВИСНОВКИ

1) Було розроблено алгоритм для знаходження реакцій та очищення даних від неповних записів та хибних. Алгоритм було реалізовано в ПЗ, котре демонструє швидший результат аніж схожі аналоги та враховує атом-атом мапінг реакцій.
2) Створені декілька підмножин датабази ORD, котрі після розгляду будуть уведену в базу даних

ДОДАТКОВА ІНФОРМАЦІЯ

Програма була реалізована на програмній мові Python останьої версії 3.11. Більшість тестування відбулося з використанням Jupiter Notebook. Дані, сформовані датасети та код викладані на ресурісі GitHub під ліцензію Apache Software License версії 2 . Посилання https://github.com/Dude6626/Search-AAM-reaction-with-the-use-of-CGR- . Останні версії будуть розташовані під цим же посиланням та також там де буде список змін та покращень.

ДЖЕРЕЛА

[1] S. M. Kearnes *et al.*, “The Open Reaction Database,” *J Am Chem Soc*, vol. 143, no. 45, pp. 18820–18826, Nov. 2021, doi: 10.1021/jacs.1c09820.

[2] “Open reaction database.” Accessed: May 20, 2024. [Online]. Available: https://docs.open-reaction-database.org

[3] M. P. Maloney *et al.*, “Negative Data in Data Sets for Machine Learning Training,” *J Org Chem*, vol. 88, no. 9, pp. 5239–5241, May 2023, doi: 10.1021/acs.joc.3c00844.

[4] G. Book, “IUPAC Gold Book.”

[5] M. Andronov, N. Andronova, M. Wand, J. Schmidhuber, and D.-A. Clevert, “A reagent-driven visual method for analyzing chemical reaction data”, doi: 10.26434/chemrxiv-2024-q9tc4.

[6] N. Schneider, N. Stiefl, and G. A. Landrum, “What’s What: The (Nearly) Definitive Guide to Reaction Role Assignment,” *J Chem Inf Model*, vol. 56, no. 12, pp. 2336–2346, Dec. 2016, doi: 10.1021/acs.jcim.6b00564.

[7] D. Weininger, “SMILES, a chemical language and information system. 1. Introduction to methodology and encoding rules,” *J Chem Inf Comput Sci*, vol. 28, no. 1, pp. 31–36, Feb. 1988, doi: 10.1021/ci00057a005.

[8] G. Hilgetag, A. Martini, and et al, “Weygand/Hilgetag Preparative Organic Chemistry.”

[9] R. K. Hill, “Organic Chemistry: *Organic Syntheses: Collective Volume* IV (a revised edition of annual volumes 30-39). Norman Rabjohn, Ed. Wiley, New York, 1963. xiv + 1036 pp. Illus. $ 16.50.; *Synthetic Methods of Organic Chemistry* . William Theilheimer. Karger, Basel, Switzerland, 1963. xvi +

507 pp. Illus. $ 38.50.,” *Science (1979)*, vol. 142, no. 3589, pp. 221–221, Oct. 1963, doi: 10.1126/science.142.3589.221.b.

[10] J. Dugundji and I. Ugi, “An algebraic model of constitutional chemistry as a basis for chemical computer programs,” in *Computers in Chemistry*, Berlin/Heidelberg: Springer-Verlag, pp. 19–64. doi: 10.1007/BFb0051317.

[11] O. N. Temkin, A. V. Zeigarnik, and D. Bonchev, *Chemical Reaction Networks*. CRC Press, 2020. doi: 10.1201/9781003067887.

[12] S. Fujita, “Description of organic reactions based on imaginary transition structures. 1. Introduction of new concepts,” *J Chem Inf Comput Sci*, vol. 26, no. 4, pp. 205–212, Nov. 1986, doi: 10.1021/ci00052a009.

[13] P. Jauffret, T. Hanser, C. Tonnelier, and G. Kaufmann, “Machine learning of generic reactions: 1. Scope of the project; the GRAMS program,” *Tetrahedron Computer Methodology*, vol. 3, no. 6, pp. 323–333, 1990, doi: 10.1016/0898-5529(90)90059-H.

[14] P. Jauffret, C. Tonnelier, T. Hanser, G. Kaufmann, and R. Wolff, “Machine learning of generic reactions: 2. toward an advanced computer representation of chemical reactions,” *Tetrahedron Computer Methodology*, vol. 3, no. 6, pp. 335–349, 1990, doi: 10.1016/0898-5529(90)90060-L.

[15] A. Varnek, D. Fourches, F. Hoonakker, and V. P. Solov’ev, “Substructural fragments: an universal language to encode reactions, molecular and supramolecular structures,” *J Comput Aided Mol Des*, vol. 19, no. 9–10, pp. 693–703, Sep. 2005, doi: 10.1007/s10822-005-9008-0.

[16] R. Nugmanov *et al.*, “stsouko/CGRtools: Final release of 4.1.” Zenodo, Jul. 2021. doi: 10.5281/zenodo.5141977.

[17] V. Delannée and M. C. Nicklaus, “ReactionCode: format for reaction searching, analysis, classification, transform, and encoding/decoding,” *J Cheminform*, vol. 12, no. 1, p. 72, Dec. 2020, doi: 10.1186/s13321-020-00476-x.

[18] A. Lin *et al.*, “Atom□to□atom Mapping: A Benchmarking Study of Popular Mapping Algorithms and Consensus Strategies,” *Mol Inform*, vol. 41, no. 4, Apr. 2022, doi: 10.1002/minf.202100138.

[19] P. Schwaller, B. Hoover, J.-L. Reymond, H. Strobelt, and T. Laino, “Unsupervised attention-guided atom-mapping.”

[20] C. Kannas and S. Genheden, “rxnutils-A Cheminformatics Python Library for Manipulating Chemical Reaction Data.” [Online]. Available: https://github.com/MolecularAI/rxnutils

[21] J. Magano and J. R. Dunetz, “Large-scale applications of transition metal-catalyzed couplings for the synthesis of pharmaceuticals,” *Chemical Reviews*, vol. 111, no. 3. pp. 2177–2250, Mar. 09, 2011. doi: 10.1021/cr100346g.

[22] J. R. Dunetz, J. Magano, and G. A. Weisenburger, “Large-Scale Applications of Amide Coupling Reagents for the Synthesis of Pharmaceuticals,” *Organic Process Research and Development*, vol. 20, no. 2. American Chemical Society, pp. 140–177, Feb. 19, 2016. doi: 10.1021/op500305s.

[23] M. S. Sigman, K. C. Harper, E. N. Bess, and A. Milo, “The Development of Multidimensional Analysis Tools for Asymmetric Catalysis and Beyond,” *Acc Chem Res*, vol. 49, no. 6, pp. 1292–1301, Jun. 2016, doi: 10.1021/acs.accounts.6b00194.

[24] A. Evers, G. Hessler, L. Wang, S. Werrel, P. Monecke, and H. Matter, "CROSS: An Efficient Workflow for Reaction-Driven Rescaffolding and Side-Chain Optimization Using Robust Chemical Reactions and Available Reagents," *J Med Chem*, vol. 56, no. 11, pp. 4656–4670, Jun. 2013, doi: 10.1021/jm400404v.

[25] J. Boström, N. Falk, and C. Tyrchan, "Exploiting personalized information for reagent selection in drug design," *Drug Discov Today*, vol. 16, no. 5–6, pp. 181–187, Mar. 2011, doi: 10.1016/j.drudis.2011.01.006.

[26] J. N. Wei, D. Duvenaud, and A. Aspuru-Guzik, "Neural networks for the prediction of organic chemistry reactions," *ACS Cent Sci*, vol. 2, no. 10, pp. 725–732, Oct. 2016, doi: 10.1021/acscentsci.6b00219.

[27] C. Tonnelier, P. Jauffret, T. Hanser, and G. Kaufmann, "Machine learning of generic reactions: 3. an efficient algorithm for maximal common substructure determination," *Tetrahedron Computer Methodology*, vol. 3, no. 6, pp. 351–358, 1990, doi: 10.1016/0898-5529(90)90061-C.

[28] J. S. Smith, O. Isayev, and A. E. Roitberg, "ANI-1: an extensible neural network potential with DFT accuracy at force field computational cost," *Chem Sci*, vol. 8, no. 4, pp. 3192–3203, 2017, doi: 10.1039/C6SC05720A.

[29] A. Toniato *et al.*, "Quantum chemical data generation as fill-in for reliability enhancement of machine-learning reaction and retrosynthesis planning," *Digital Discovery*, vol. 2, no. 3, pp. 663–673, 2023, doi: 10.1039/D3DD00006K.

[30] S. Maeda, Y. Harabuchi, M. Takagi, T. Taketsugu, and K. Morokuma, "Artificial Force Induced Reaction (AFIR) Method for Exploring Quantum

Chemical Potential Energy Surfaces," *The Chemical Record*, vol. 16, no. 5, pp. 2232–2248, Oct. 2016, doi: 10.1002/tcr.201600043.

[31] S. H. Newman-Stonebraker, J. Y. Wang, P. D. Jeffrey, and A. G. Doyle, "Structure–Reactivity Relationships of Buchwald-Type Phosphines in Nickel-Catalyzed Cross-Couplings," *J Am Chem Soc*, vol. 144, no. 42, pp. 19635–19648, Oct. 2022, doi: 10.1021/jacs.2c09840.

[32] S. Johansson *et al.*, "AI-assisted synthesis prediction," *Drug Discov Today Technol*, vol. 32–33, pp. 65–72, Dec. 2019, doi: 10.1016/j.ddtec.2020.06.002.

[33] "Casreact website." Accessed: May 20, 2024. [Online]. Available: https://www.cas.org/support/documentation/ reactions

[34] "Reaxys database." Accessed: May 20, 2024. [Online]. Available: https://www.reaxys.com

[35] D. M. Lowe, "Extraction of chemical structures and reactions from the literature," 2012.

[36] P. Schwaller, B. Hoover, J.-L. Reymond, H. Strobelt, and T. Laino, "Extraction of organic chemistry grammar from unsupervised learning of chemical reactions," *Sci Adv*, vol. 7, no. 15, Apr. 2021, doi: 10.1126/sciadv.abe4166.

[37] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, and R. Soricut, "ALBERT: A Lite BERT for Self-supervised Learning of Language Representations," Sep. 2019.