УДК 004.777:[811.161.2:801.66]
DOI: 10.18523/2617-3808.2025.8.213-224

Смиш О. Р., Швець Д. В.

АНАЛІЗУВАННЯ УКРАЇНСЬКОМОВНИХ ВІРШІВ ЗАСОБАМИ ОБРОБКИ ПРИРОДНОЇ МОВИ

У статті описано розроблення та застосування парсера для комплексного аналізування українськомовних віршів. Створено механізми визначення віршового розміру, рим і способів римування, що ґрунтуються на побудові схем наголошування. Парсер дає змогу виявляти входження 16 художніх засобів на трьох рівнях оброблення тексту. На фонологічному рівні застосовано rule-based підходи в аналізуванні тексту, на морфосинтаксичному — оброблення даних формату CoNLL-U, сформованих із результатів роботи трьох мовних аналізаторів, а для виявлення тропів на семантичному рівні — використано велику мовну модель.

Створено вебзастосунок для наочної взаємодії з функціями парсера. Розроблений інструмент застосовано для аналізування трьох українськомовних збірок віршів.

Ключові слова: обробка природної мови, парсер, мовний аналізатор, великі мовні моделі, вебзастосунок, українська мова, вірш, аналіз вірша, троп, художній засіб, віршовий розмір, рима, спосіб римування.

Вступ

У сучасних умовах інтенсивного зростання обсягів текстової інформації та стрімкого розвитку технологій обробка природної мови стає одним із провідних напрямів у галузі інформатики та комп'ютерної лінгвістики. Сьогодні близько 30 млн осіб знають українську мову і вважають її рідною [15], тож існує потреба в інструментах для оброблення та різнобічного аналізування українськомовних текстів.

Необхідною частиною освітнього процесу в Україні є вивчення художньої літератури, зокрема поезії. У програмі зовнішнього незалежного оцінювання (ЗНО) з української мови та літератури зазначено, що учні повинні знати та розуміти поняття «ліричний вірш» і «художні засоби» [3]. Тому необхідно, щоб учителі української літератури надавали учням достатню кількість прикладів аналізу віршів.

Перед студентами гуманітарних спеціальностей, філологами, літературознавцями й іншими фахівцями, що досліджують фольклор і різноманітні епохи та течії в українській поезії, виникає потреба в обробленні значного обсягу віршованих текстів, що вимагає часу та зусиль. Проте наразі немає готових рішень, які б уможливили програмне аналізування віршів, написаних українською мовою.

Розроблений парсер дає змогу здійснювати різнобічне аналізування українськомовних віршів, а саме:

- визначати віршовий розмір (метр) кожного рядка та всього вірша;
- визначати риму та способи римування рядків;
- виявляти входження 16 тропів: алітерація, анафора, асонанс, гіпербола, епітет, епіфора, інверсія, літота, метафора, оксиморон, паралелізм, персоніфікація, порівняння, рефрен, риторичне звертання, риторичне питання.

Основні терміни в аналізі віршів

Автоматизоване аналізування вірша потребує комплексного підходу, що охоплює звукову та смислову структуру поетичного тексту. Через метр і риму організовано ритміку тексту та його звукову впорядкованість. Натомість тропи унаочнюють змістову й естетичну наповненість тексту, художнє мислення автора. Ці компоненти можна визначити однозначно, на відміну від теми вірша, емоційного забарвлення слів тощо. Об'єднання їх дає змогу різнобічно характеризувати поетичний

© *Смиш О. Р., Швець Д. В.*, 2025

текст, уможливлюючи подальше порівнювання віршів, визначення закономірностей у межах поетичної збірки чи в усій творчості поета, глобальне аналізування поезії певного мистецького напряму чи епохи.

Стопа — визначальний метричний період, найкоротший відрізок співмірності певного віршового метра, сконцентрованого в групі складів із відносно незмінним наголосом [2, с. 435]. Із наголошених і ненаголошених складів сформовано стопи, а зі стоп — ритмічну схему, за якою можна визначити метр вірша. Наголошені склади позначають символом «–», а ненаголошені — «∪».

Таблиця. **Стопи**

Назва	Схема	Наголошено склад
Хорей	– ∪	перший із двох
Ямб	∪ –	другий із двох
Дактиль	– ∪ ∪	перший із трьох
Амфібрахій	∪ – ∪	другий із трьох
Анапест	∪ ∪ –	третій із трьох

Віршовий розмір, або метр — особливість стопи, покладеної в основу певної віршової структури, що фіксує відповідний тактометричний період [1, с. 192]. До основних розмірів належать двоскладові (хорей і ямб) і трискладові (дактиль, амфібрахій, анапест) стопи [2, с. 387]. Віршовий розмір складається з кількості повторення стопи та її назви: п'ятистопний хорей, тристопний анапест тощо.

Рима — композиційно-звуковий засіб суголосся закінчень, що має фонетичне та метричне значення, об'єднує останні слова суміжних або близько розташованих віршових рядків (починаючи з останнього наголошеного складу) [2, с. 322]. Рядки є римованими, якщо їхні останні наголошені склади містять однакові голосні звуки (або близькі за звучанням звуки [и] й [і]) та розташовані на однаковій відстані до кінця рядка.

Римування — особливість розташування рим у вірші, інтервал між ними. Суміжне римування означає, що римовані слова розташовані в сусідніх рядках, перехресне — через один рядок (у другому та четвертому, п'ятому та сьомому тощо). У кільцевому римуванні два римовані рядки розташовані через два рядки, які також між собою римовані [2, с. 324].

Художні засоби (тропи) — сукупність зображально-виражальних засобів, прийомів, способів діяльності письменника, за допомогою яких він творить нову художню дійсність [2, с. 565].

Визначення віршового розміру

Сформовано програмний механізм визначення віршового розміру:

1) розставити наголоси в тексті вірша;
2) створити список списків булевих значень, де True позначає наголошені склади, а False — ненаголошені;
3) визначити для кожного рядка, скільки разів схема якої стопи в ньому повторюється;
4) визначити найчастішу стопу серед рядків і перевірити для рядків, у яких визначена інакша стопа (або не визначено жодної), чи підходить найчастіша стопа їм;
5) скоригувати створений у п. 2 список, якщо деякі наголоси суперечать визначеній схемі.

Для розставлення наголосів у тексті вірша використано Python-пакет Ukrainian Word Stress [8]. У цьому модулі передбачено оброблення омографів, можна обрати стратегію наголошування слів із паралельними наголосами. Вирішено використовувати стратегію розставлення всіх можливих паралельних наголосів, оскільки на етапі коригування залишаються лише ті, що не суперечать визначеній схемі.

Для кожного булевого списку, що відповідає складам одного рядка вірша, застосовано метод, який перевіряє перші n елементів (складів), де або n кратне двом чи трьом (оскільки стопа може бути двоскладовою чи трискладовою), або останній з n наголосів є наголошеним. Ці n елементів мають бути рівними n елементам списку, сформованого відповідно до однієї зі схем стоп. Якщо це так, то кількість стоп дорівнює частці від цілочисельного ділення n на довжину стопи, інакше розмір є невизначеним.

У віршах деякі слова можуть мати наголос, відмінний від нормативного. Окрім того, емпірично виявлено, що модуль для наголошування не завжди правильно розставляє наголоси. Із цих причин реалізовано методи для перевірки віршового розміру та коригування наголосів.

Після формування схем наголошення безпосередньо з наголошеного тексту, програма визначає стопу, що відповідає найбільшій кількості рядків. Якщо для деяких рядків визначено іншу стопу або не визначено жодної, програма перевіряє, чи відповідає такий рядок найчастішій стопі.

Якщо принаймні половина схем стоп у рядку відповідає схемі найчастішої стопи, відбувається коригування наголосів у цьому рядку. Для двоскладових стоп відбувається вилучення наголосів, що суперечать схемі, через що можуть утворитися стопи без наголошених складів. Двоскладова стопа без наголошених складів — пірихій — не суперечить схемам з ямбом і хореєм [2, с. 216]. Проте в трискладових стопах пропуски наголошених складів недопустимі, оскільки так виникає по п'ять ненаголошених складів поспіль. Тому для коригування наголосів у рядках із трискладовими стопами програма перестворює їхні схеми наголошування.

Визначення рим і способів римування

Для кожного рядка програма створює його репрезентацію, що є кортежем (a, b, c, d), де a — від'ємний індекс останнього наголошеного складу (останнього значення True в булевому списку наголошених і ненаголошених складів), b — голосний звук останнього наголошеного складу, c — решта голосних звуків після останнього наголошеного, d — усі голосні звуки рядка.

Рими між рядками позначено списком пар їхніх індексів. Пошук рим за їхніми репрезентаціями відбувається ітеративно зі зменшенням строгості зіставлення репрезентацій. Значення строгості спадає на кожній ітерації на одиницю від 3 до 1. Якщо для більш ніж половини рядків не знайдено рим після пошуку зі строгістю 2, програма визначає цей вірш таким, що не є римованим, тобто є верлібром [1, с. 165].

При строгості зі значенням 3 програма визначає римованими лише ті рядки, у репрезентаціях яких збігаються a, b і c; зі значенням 2 — a і b. Строгість зі значенням 1 передбачена для випадків, коли в одному з двох римованих рядків програма не поставила наголос на склад, який насправді є останнім наголошеним. Тоді для кожної пари зі списку рядків, для яких не знайдено рими, програмний механізм знаходить найбільший індекс останніх наголошених складів (і далі до кінця рядка) та перевіряє, чи збігаються в них голосні за цим індексом. Якщо так, то ці рядки є римованими, елементам із відповідним індексом у булевих списках римування цих рядків надається значення True.

Щоби визначити види римування для груп римованих рядків, використано словник, у який програма додає пари індексів римованих рядків відповідно до їхнього розташування у вірші.

Виявлення художніх засобів

Для аналізування парсером обрано тропи, що входять до списку художніх засобів у програмі ЗНО з української мови та літератури [3]: алітерація, анафора, асонанс, гіпербола, епітет, епіфора, інверсія, літота, метафора, оксиморон, паралелізм, персоніфікація, порівняння, рефрен, риторичне звертання, риторичне питання.

Наведені 16 тропів розділено на три групи, що відповідають певним рівням NLP [10], на яких можна визначити тропи відповідної групи. Алітерацію, анафору, асонанс, епіфору, рефрен і риторичне питання можна виявити на фонологічному рівні аналізування тексту. На морфосинтаксичному рівні можна визначити епітет, інверсію, паралелізм, порівняння та риторичне звертання. Решта тропів — гіпербола, літота, метафора, оксиморон і персоніфікація — стосуються семантичного рівня.

Щоб уніфікувати вигляд результатів аналізування тропів і точно вказувати їхні розташування в тексті вірша, знайдені входження тропів вирішено представляти як діапазони (a, b), де a — індекс символу в тексті вірша, де починається входження тропа (охопно), b — індекс символу, де закінчується входження тропа (не охопно). Тобто якщо перший символ входження має індекс 7, а останній — 14, то діапазоном є (7, 15).

Тропи на фонологічному рівні

На фонологічному рівні аналізування тексту існують три види правил: фонетичні, фонемні та просодичні [10], тобто на цьому рівні достатньо застосовувати правилозалежні (rule-based) підходи.

Алітерація [1, с. 53] і асонанс [1, с. 101] є стилістичними прийомами, що полягають у повторенні однакових звуків: алітерація — приголосних, асонанс — голосних. Для виявлення цих тропів засто-

совано однаковий підхід, що базується на підрахунку кількості входжень відповідних літер у тексті вірша, де йотовані голосні замінені відповідними голосними звуками (я → а, ю → у тощо). Якщо кількість однакових приголосних у рядку не менша за 3, то програма визначає такі символи як входження асонансу. Аналогічно з алітераціями, проте для них мінімальною кількістю повторів є 4, оскільки кількість різних голосних звуків в українській мові менша за кількість приголосних.

Рефрен є композиційним прийомом, що полягає в повторенні групи слів, рядка або кількох віршових рядків у строфах [2, с. 318]. Метод виявлення рефренів реалізує механізм, оснований на пошуку слів або їхніх послідовностей, які повторюються в тексті.

Аналізування починається з токенізації — виокремлення слів, для кожного з яких програма фіксує діапазони розташування у вихідному тексті. Це дає змогу не лише встановити факт повторення, а й точно визначити позицію кожного входження. Наступним кроком є побудова словника, де кожному унікальному слову відповідає список індексів його входжень. Після цього програма аналізує лише ті слова, які входять у текст більше ніж один раз.

Для кожного з повторів допоміжна функція визначає підмножину індексів, що позначає дубльовані слова, сусідні слова яких також повторюються. Ідея полягає в тому, щоби знайти ті входження слова, у яких найближче сусіднє слово (зліва або справа) збігається з аналогічним елементом в інших входженнях того самого слова. Такий підхід дає змогу вилучити дублікати одного слова без повтору принаймні одного сусіднього.

Анафора [1, с. 66] і епіфора [1, с. 342] — це стилістичні фігури, що полягають у повторенні однакових слів або зворотів на початку (анафора) чи наприкінці (епіфора) рядків. Застосовано метод, який визначає позиції перших і останніх літер кожного рядка відповідно. Далі використано метод виявлення рефренів, у який передано список індексів перших (для анафор) або останніх (для епіфор) літер у рядках. Це змінює поведінку методу визначення рефренів так, що він повертає входження повторів, які розташовані на початку чи в кінці рядків відповідно.

Риторичне питання — це стилістична фігура, виражена питанням, яке не потребує відповіді [2, с. 329]. Риторичне питання має бути словами автора (не належати героєві твору), тобто бути поза прямою мовою.

Питальні речення в українській мові позначають знаком питання, тож використано підхід знаходження таких речень за умови, що речення не починається з тире та не розташоване між парою лапок.

Тропи на морфосинтаксичному рівні

На морфосинтаксичному рівні аналізування тексту основну увагу зосереджено на морфологічних ознаках слів, граматичних зв'язках між ними та синтаксичній структурі речень. Для виявлення тропів на цьому рівні використано залежні від синтаксичних зв'язків правила, що спираються на розбір тексту у форматі CoNLL-U.

CoNLL-U (або Universal Dependencies CoNLL format) — це формат представлення лінгвістично розмічених корпусів, який використано в рамках проєкту UD для зберігання синтаксичних дерев і морфологічної інформації про слова. Він є частиною ініціативи CoNLL (Conference on Computational Natural Language Learning) для забезпечення уніфікованого підходу до оброблення мовних даних [5].

Дані CoNLL-U про вірш вирішено формувати на основі результатів оброблення тексту вірша трьома мовними аналізаторами: UDPipe [14], Stanza [13] та spaCy [12]. Для кожного слова програма обирає ті варіанти значень стовпців, які збігаються принаймні для двох аналізаторів, щоби запобігти ситуації, де один аналізатор неправильно визначає певні ознаки слів.

Епітет — це троп, виражений переважно прикметниками, що образно наголошує на характерній ознаці, одиничній якості певного предмета або явища. У ролі епітета може вживатися також іменник [1, с. 342]. Для його виявлення парсер шукає слова, які є прикметниками (*ADJ*) або прикладками (залежність *appos*).

Інверсія — стилістична фігура художнього мовлення, що полягає в порушенні прямого порядку слів у реченні [1, с. 419]. Для виявлення інверсій програма шукає інверсійні розміщення слів (див. рис. 1).

Паралелізм — аналогія, уподібнення, спільність ознак або дій, художній і композиційний прийом, що полягає в зіставленні або протиставленні двох чи кількох подій. Зазвичай паралелізм виражений однорідними синтаксичними конструкціями [2, с. 182]: присудками в межах однієї граматич-

ної основи або сурядними частинами. Метод визначення паралелізмів шукає однорідні конструкції (залежність *parataxis*), де послідовні речення або частини речення містять присудки з однаковим способом дії (*Mood*): дійсний, умовний або наказовий. Програма доповнює виявлені групи зв'язаними частками, якщо такі наявні для охоплення всього входження тропа.

```
if (tok[F.deprel] in ('nsubj', 'amod', 'det',
                      'advmod:det', 'det:numgov', 'det:nummod', 'nummod', 'nummod:gov')
        and tok[F.head] < tok[F.id]):
    inversions.append((head_of(tok, sent)[F.misc], tok[F.misc]))
elif (tok[F.deprel] in ('obj', 'iobj', 'obl', 'nmod', 'acl:relcl',
                        'xcomp', 'xcomp:pred')
      and tok[F.upos] in ('NOUN', 'PROPN', 'PRON')
      and tok[F.head] > tok[F.id]):
    adp = [t[F.misc] for t in sent[:tok[F.id]]
           if t and t[F.deprel] == 'case' and t[F.head] == tok[F.id]]
    if adp:
        inversions.append((adp[-1], tok[F.misc], head_of(tok, sent)[F.misc]))
    else:
        inversions.append((tok[F.misc], head_of(tok, sent)[F.misc]))
```

Рис. 1. Фрагмент методу виявлення інверсій

Порівняння — троп, що полягає в поясненні одного предмета через подібний до нього інший, зіставленні їх за допомогою компаративної зв'язки, тобто єднальних сполучників: *як*, *наче*, *неначе*, *мов*, *немов*, *мовби*, *немовби*, *ніби*, *нібито* [2, с. 248]. Програма шукає такі сполучники, а також залежність *advcl*, що в CoNLL-U позначає розгорнуті обставини, до яких належать порівняльні звороти, після чого поєднує пов'язані слова з відповідним сполучником.

Риторичне звертання — стилістична фігура, прийом звертання в якій ужитий для надання мовленню необхідної інтонації [2, с. 329]. Парсер знаходить цей троп як іменник у кличному відмінку (ознака *Case* має значення *Voc*) або за залежністю *vocative*, що в CoNLL-U позначає звертання.

Тропи на семантичному рівні

Семантичний рівень аналізування передбачає інтерпретацію значення тексту та виявлення смислових зв'язків. На цьому рівні реалізовано виявлення тропів, значення яких неможливо встановити за формальними ознаками, а лише через осмислення контексту.

```
У поданому тексті знайди всі входження вказаних тропів, тільки якщо такі наявні, і дай відповідь
чітко у форматі JSON без додаткової інформації:
{results: {metaphor: [метафори], personification: [персоніфікації], hyperbole: [гіперболи],
litotes: [літоти], oxymoron: [оксиморони]}}
Метафора — троп, що полягає в перенесенні значення одного слова (словосполучення) на
інше, розкрити сутності одних явищ чи предметів
через інші за схожістю і контрастом.
Персоніфікація — троп, що полягає в наданні предметам чи явищам рис живих істот.
Гіпербола — троп, для якого характерне надмірне перебільшення особливостей чи ознак предмета,
явища або дії.
Літота — троп, що полягає в художньому зменшенні величини, сили, значення зображуваного предмета
чи явища.
Оксиморон — троп, особливістю якого є поєднання контрастних, протилежних за значенням слів,
внаслідок чого утворюється парадоксальна семантична сполука.
Кожне входження має бути взяте з тексту без розривання тексту.

Текст:
Синиця в шибку вдарила крильми.
Годинник став. Сіріють німо стіни.
Над сизим смутком ранньої зими
Принишкли хмари, мов копиці сіна.
Пливе печаль. Біліють смолоскипи
Грайливо пофарбованих ялин —
Вони стоять, немов у червні липи,
Забрівши в сивий і густий полин.
Полин снігів повзе до видноколу,
Лоскоче обрій запахом гірким.
Лапаті, білі і колючі бджоли
Неквапно кружеляють понад ним...
```

Рис. 2. Приклад сформованого запиту для виявлення тропів у вірші В. Симоненка «З вікна»

На відміну від попередніх рівнів, тут застосовано велику мовну модель Claude 3.5 Sonnet, яка здатна інтерпретувати художні засоби, спираючись на попереднє тренування на великих корпусах текстів. Для цього програма формує запит (prompt) до LLM, у якому зазначено текст вірша та перелік тропів, які потрібно знайти. До вказаних у запиті тропів програма додає їхні визначення: метафора [1, с. 35], персоніфікація [2, с. 208], гіпербола [1, с. 227], літота [1, с. 581], оксиморон [2, с. 149]. Модель має надати відповідь строго у форматі JSON, де для кожного вказаного тропа наведено список його входжень. Приклад сформованого запиту наведено на рис. 2. У запиті також указано, що кожне входження має бути взято з тексту, не розриваючи його.

Для під'єднання до LLM використано Python-модуль g4f. Він містить засоби для взаємодії з переліком LLM [9]. Щоби запобігти галюцинаціям, кожен запит надсилається в окремому чаті.

Отриману відповідь програма перетворює на словник, після чого шукає вказані входження в тексті та замінює їх відповідними діапазонами індексів. Це потрібно для однакового представлення входжень усіх тропів, які виявляє парсер, а також для перевірки, що надані моделлю входження дійсно наявні в тексті вірша.

Створення вебзастосунку

Розроблення вебзастосунку для автоматичного аналізування українськомовних віршів зумовлено необхідністю створення інтерфейсу доступу до функцій парсера.

Серверна частина вебзастосунку реалізована з використанням вебфреймворку FastAPI, що забезпечує асинхронне оброблення HTTP-запитів і підтримання типізованого обміну даними через моделі Pydantic [6]. Ця частина відповідає за отримання тексту користувача, виклик відповідних модулів аналізування вірша та формування структурованої відповіді. Реалізовано два основні маршрути.

Перший маршрут — «/tropes» із методом POST — приймає в тілі запиту текст вірша та список тропів, які потрібно виявити. Згідно з рівнем аналізування, програма обирає відповідний клас для кожного тропа, створює по одному екземпляру класу для однієї категорії тропів і викликає потрібні методи. Після завершення аналізування тексту та виявлення тропів, програма об'єднує результати в один об'єкт і повертає його клієнтові у форматі JSON.

Другий маршрут — «/metre-rhyme» із методом POST — приймає в тілі запиту текст вірша та повертає результати аналізування віршового розміру, рими та способів римування. У відповіді клієнт отримує:

- бінарну схему наголошування (де 1 — наголошений склад, 0 — ненаголошений);
- представлення віршового розміру у форматі словника, де ключами на верхньому рівні є скорочені назви стоп, їхніми значеннями — об'єкти з ключами, що відповідають кількості стопи, та значеннями, представленими як списки індексів рядків (індекс відповідає номерові рядка в тексті);
- представлення способів римування у форматі словника, де ключами є назви способів римування, а їхніми значеннями — списки з парами індексів римованих рядків. Якщо деякі рими не відповідають жодному з трьох способів римування, то вони розміщені в списку за ключем «rest».

Клієнтська частина вебзастосунку реалізована з використанням бібліотеки React, що дає змогу створювати динамічний односторінковий інтерфейс із реактивною зміною станів. Основне завдання клієнтської частини полягає в збиранні вхідних даних від користувача, формуванні HTTP-запитів до серверної частини й інтерактивному відображенні результатів аналізування тропів, віршового розміру та римування.

Інтерфейс поділено на дві основні вкладки: «Тропи» та «Метр і рима». У першій вкладці користувач має змогу (див. рис. 3):

- увести текст вірша;
- обрати один або кілька тропів для виявлення;
- запустити виявлення тропів;
- переглядати результати через візуально виділені фрагменти тексту — кожен троп марковано відповідним кольором;
- перемикатися між кнопками тропів для фокусування на кожному з них (сірі кнопки позначають тропи, яких не виявлено в тексті).

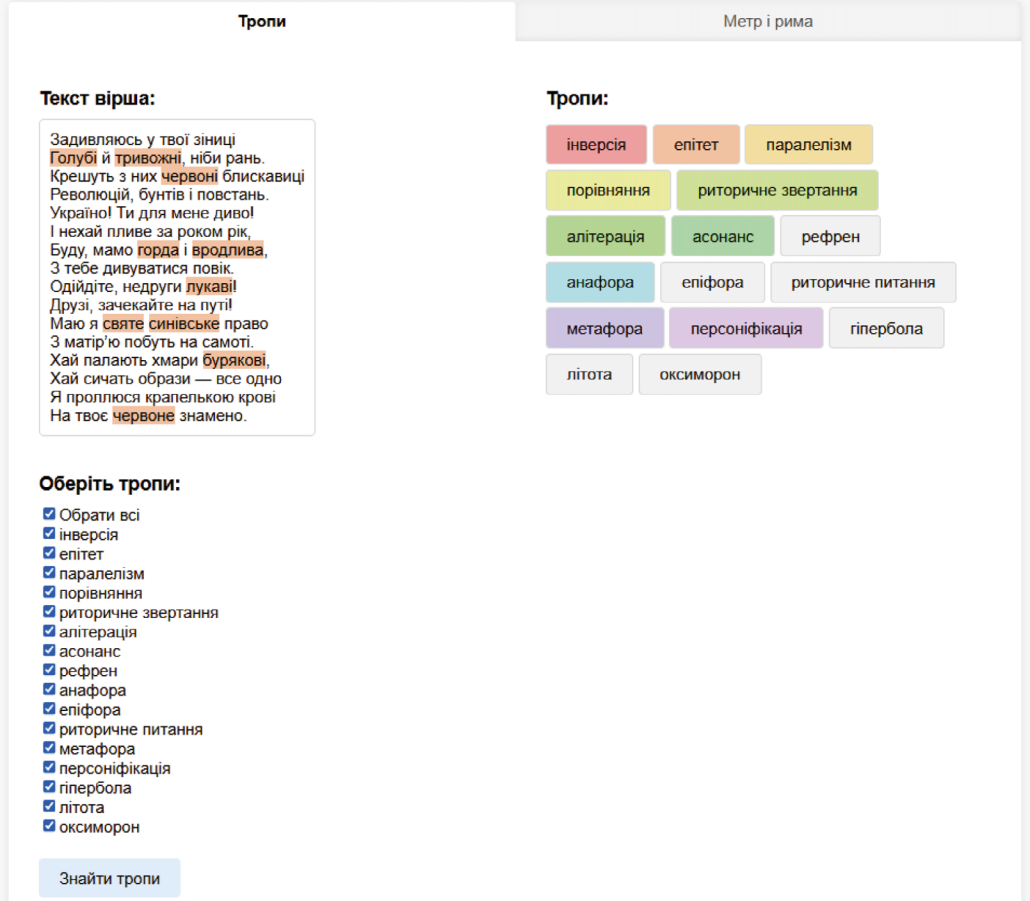

Рис. 3. Вкладка «Тропи» (вірш «Задивляюсь у твої зіниці...» В. Симоненка)

У вкладці «Метр і рима» відображаються результати аналізування віршового розміру та римування (див. рис. 4). Користувач може ввести текст вірша та натиснути відповідну кнопку для аналізування. На лівій половині вікна, праворуч від тексту розміщені кольорові позначки римованих рядків (між собою римовані ті рядки, що мають однакові позначки). Під текстом вірша — знайдені види римування відповідних кольорів. На правій половині — аналіз метра: схема наголошування, метр кожного рядка, основний віршовий розмір — тобто такий, що відповідає більшості рядків вірша.

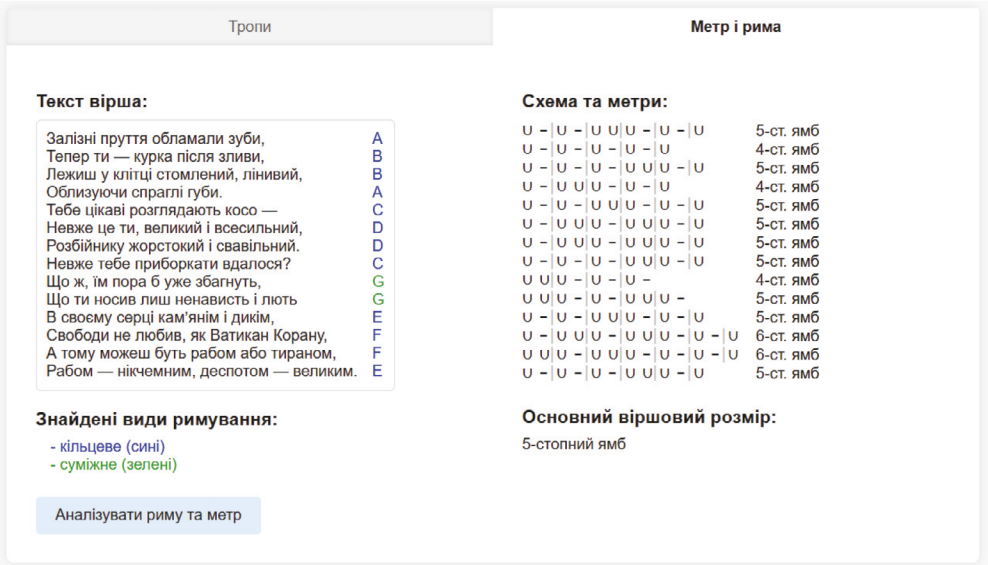

Рис. 4. Вкладка «Метр і рима» (вірш «Лев у клітці» В. Симоненка)

Точність аналізування віршів парсером і великими мовними моделями

Щоби надати оцінку точності визначення віршового розміру (метра) та рими парсером, сформовано збірку тестів, представлену як таблицю, де стовпцями є: текст вірша; назва стопи; кількість стоп; список пар індексів римованих рядків.

Зі 107 текстів віршів парсер правильно визначив метр (тобто назву стопи та її кількість одночасно) для 101 тексту, а риму — для 103. Тому точність визначення метра парсером становить 94,39 %, а рими — 96,26 %.

Щоб оцінити точність виявлення тропів парсером, створено збірку тестів у форматі таблиці, де стовпцями є: текст вірша або його фрагмент; назва тропа, який потрібно виявити; слово чи набір слів, що є входженням тропа в тексті. Збірка містить загалом 1808 таких тестів. Результати тестування наведено на рис. 5.

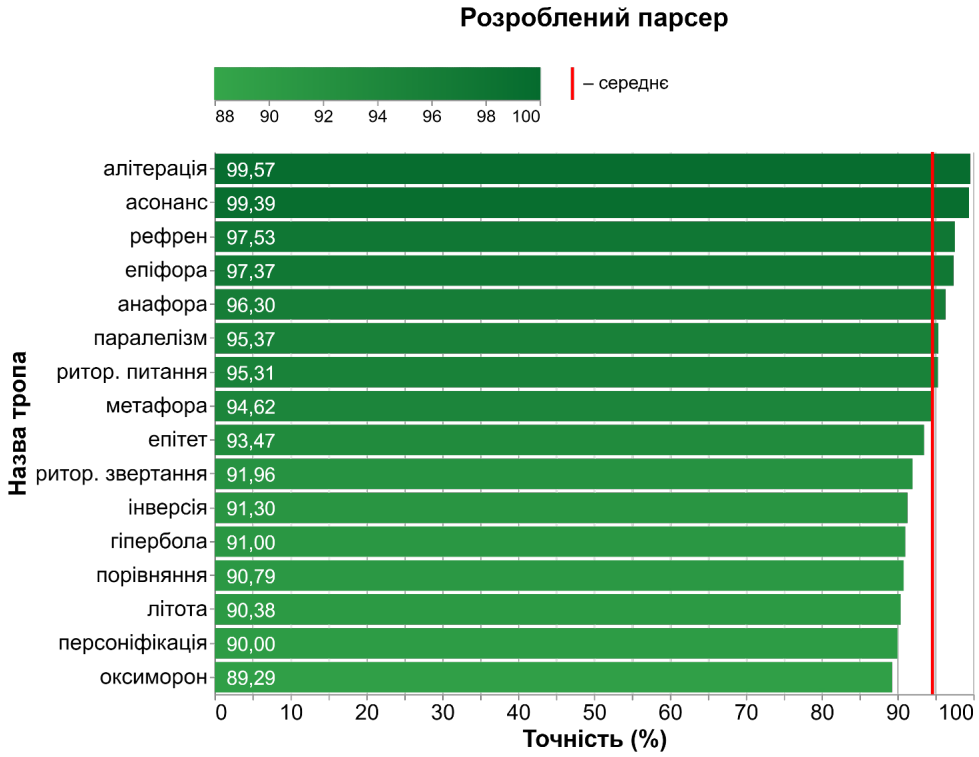

Рис. 5. Точність виявлення тропів розробленим парсером

З огляду на широкі можливості LLM, вирішено оцінити їхню здатність визначати метр і риму, а також виявляти художні засоби (тропи) в українськомовних віршах. У рамках дослідження протестовано три мовні моделі: ChatGPT-4 [11], Claude 3.5 Sonnet [4] і Gemini 2.0 Flash [7].

На основі результатів тестування великих мовних моделей і розробленого парсера на точність виявлення тропів, визначення метра та рими, складено графіки порівняння точностей, зображені на рис. 6.

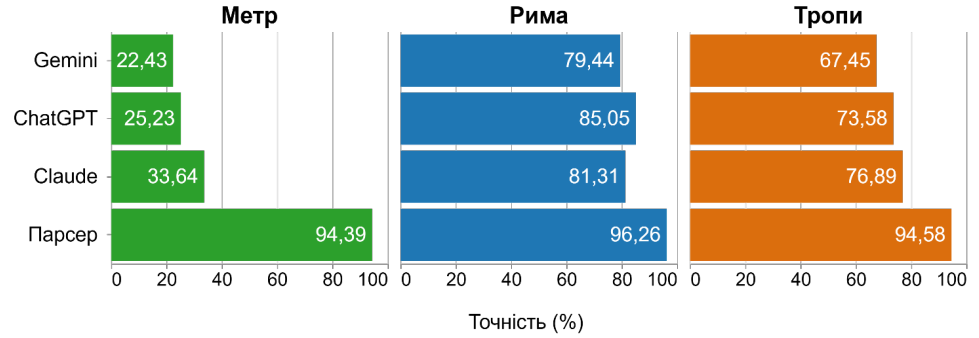

Рис. 6. Графіки точностей визначення метра, рими і тропів великими мовними моделями та розробленим парсером

На рисунку можна побачити відсоткові значення точностей. Розроблений парсер має точність визначення метра вірша на 67,26 % краще за середнє значення для LLM, рими — на 14,33 %, а тропів — на 21,94 %.

Застосування парсера для аналізування збірок віршів

Щоби наочно продемонструвати можливості розробленого парсера, з його використанням проаналізовано українськомовні вірші з трьох поетичних збірок: «Кобзар» Т. Шевченка — 66 віршів, «Відгуки» Лесі Українки — 25 віршів, «Земне тяжіння» В. Симоненка — 52 вірші.

На рисунку 7 зображено розподіл віршів зі збірки «Кобзар» за роками написання з урахуванням метричних стоп. Можна бачити, що до 1850 р. поет використовував різні стопи: переважно двоскладові (ямб і хорей), проте наявні також два вірші, написані анапестом, і один — амфібрахієм. Після перерви, з 1857 р. основною стопою у віршах Шевченка є лише ямб.

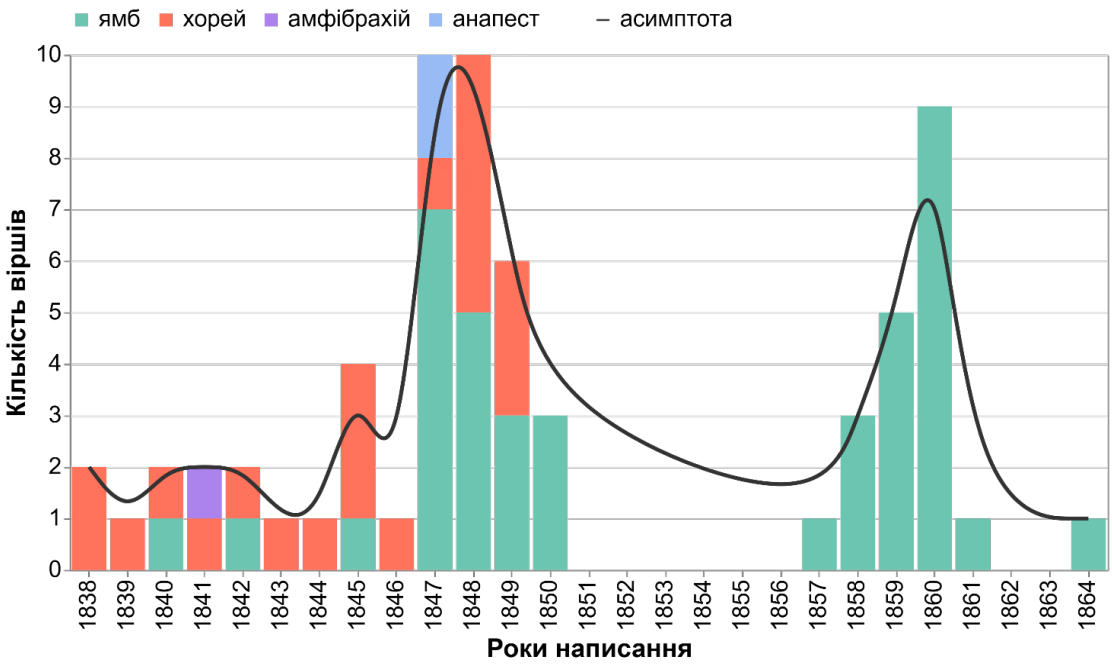

Рис. 7. Розподіл віршів зі збірки «Кобзар» за роками написання з урахуванням стоп

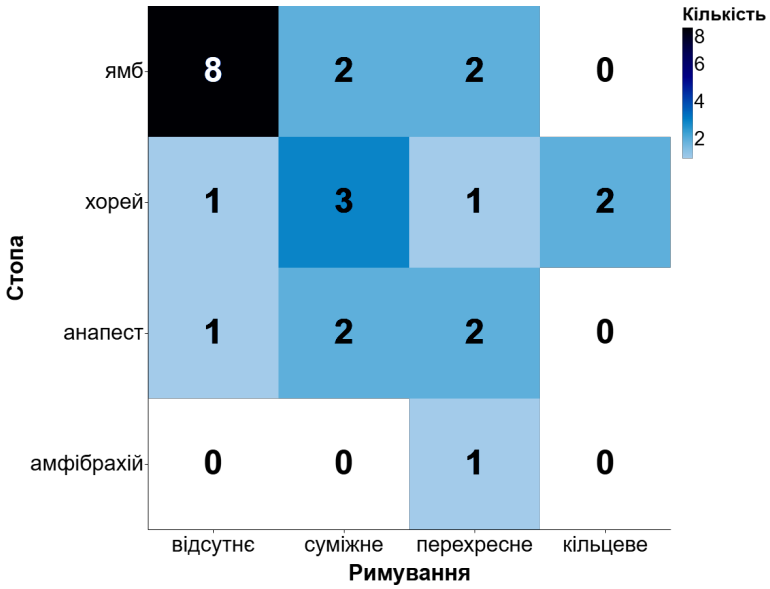

Рис. 8. Згруповані за використаною стопою та способом римування вірші зі збірки «Відгуки»

Наступною візуалізацією, поданою на рис. 8, є теплова мапа, де відображено кількості віршів зі збірки «Відгуки», згруповані за використаною стопою та способом римування. Найбільшу кількість віршів у збірці (8) написано ямбом без римування.

Для кожної з трьох проаналізованих збірок створено двовимірні KDE-графіки щільності розподілу тропів.

Для кожного вірша збірки, за допомогою парсера, визначено індекси символів, що відповідають входженням тропів, після чого координати цих ділянок нормалізовано від 0 до 100 у двох вимірах. Горизонтальна вісь відповідає позиції тропа в межах кожного рядка, а вертикальна — положенню рядка у вірші. У результаті для всієї сукупності віршів відповідної збірки сформовано список точок, що вказують розміщення тропів у тексті. На основі таких списків побудовано графіки, що відображають щільність розподілу тропів на площині, утвореній нашаруванням віршів збірки. Мапа щільності дає змогу виявити тенденції в композиційному розміщенні художніх засобів.

Першим на рис. 9 зображено графік для збірки «Кобзар». Можна побачити, що в правій верхній частині графіка є ділянка, де щільність точок найвища. Це свідчить про те, що в збірці в більшості віршів значна частина тропів розташована в кінці перших рядків.

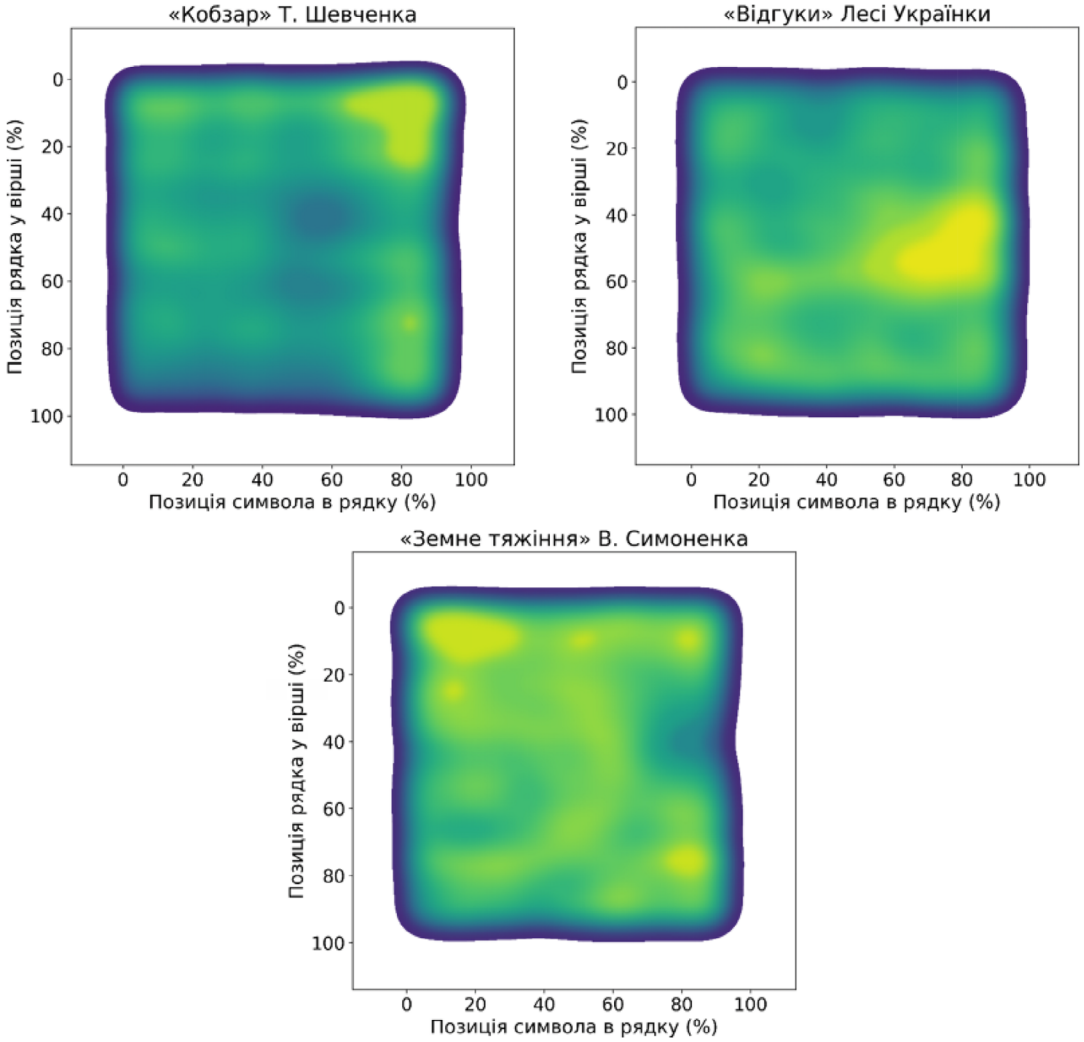

Рис. 9. KDE-графіки щільності розподілу тропів у проаналізованих збірках

На графіку для збірки «Відгуки» Лесі Українки видно, що найбільше тропів зосереджено в кінці середніх рядків віршів.

Якщо подивитися на KDE-графік для збірки «Земне тяжіння», можна помітити, що тропи у віршах збірки розташовані більш рівномірно, ніж у двох інших, а також те, що поет використовував тропи переважно на початку перших рядків вірша.

Висновок

У статті описано розроблення та застосування парсера для визначення метра, рими, способів римування та виявлення художніх засобів (тропів) в українськомовних віршах.

Запропоновано механізми визначення віршового розміру, рим і способів римування, що ґрунтуються на побудові схем наголошування. Передбачено методи коригування цих схем. Розроблено методи виявлення тропів на трьох рівнях оброблення тексту: фонологічному, морфосинтаксичному та семантичному. На фонологічному рівні застосовано правилозалежні підходи в аналізуванні тексту, на морфосинтаксичному — оброблення даних формату CoNLL-U, сформованих із результатів роботи трьох мовних аналізаторів, а для виявлення тропів на семантичному рівні — використано велику мовну модель. Реалізовано вебзастосунок, що уможливлює візуалізацію результатів роботи парсера, з використанням FastAPI та React.

Здійснено оцінювання точності парсера на основі сформованих збірок тестів. За результатами тестування встановлено точності компонентів парсера: 94,39 % — визначення метра, 96,26 % — визначення рими, 94,58 % — виявлення тропів. Порівняння з результатами роботи LLM показало вищу точність створеного інструмента. Також у статті оглянуто застосування парсера для аналізування віршів із трьох поетичних збірок та наведено візуалізації, що унаочнюють отримані результати.

Отже, розроблений парсер уможливлює комплексне аналізування українськомовних віршів, охоплюючи визначення віршового розміру, рим, способів римування та виявлення художніх засобів.

Список літератури

1. Ковалів Ю. І. Літературознавча енциклопедія : у двох томах. Т. 1 / Ю. І. Ковалів. — Київ : ВЦ «Академія», 2007. — 608 с.
2. Ковалів Ю. І. Літературознавча енциклопедія : у двох томах. Т. 2 / Ю. І. Ковалів. — Київ : ВЦ «Академія», 2007. — 624 с.
3. Програма ЗНО з української мови і літератури 2025 року [Електронний ресурс] // Освіта.UA. — Режим доступу: https://osvita.ua/test/program_zno/946.
4. Claude [Electronic resource] // Anthropic. — Mode of access: https://claude.ai.
5. CoNLL-U Format [Electronic resource] // Universal Dependencies. — Mode of access: https://universaldependencies.org/format.html.
6. FastAPI [Electronic resource] // FastAPI. — Mode of access: https://fastapi.tiangolo.com.
7. Gemini [Electronic resource] // Google AI. — Mode of access: https://gemini.google.com.
8. GitHub - lang-uk/ukrainian-word-stress: Adds word stress to Ukrainian texts [Electronic resource] // GitHub. — Mode of access: https://github.com/lang-uk/ukrainian-word-stress.
9. GitHub - techwithanirudh/g4f: The official gpt4free repository // GitHub. — Mode of access: https://github.com/techwithanirudh/g4f.
10. Liddy E. Natural Language Processing / E. Liddy // Library and Information Science. — 2nd ed. — NY, 2001.
11. OpenAI. GPT-4 Technical Report [Electronic resource] / OpenAI // Computation and Language. — 2023. — Mode of access: https://doi.org/10.48550/arXiv.2303.08774.
12. SpaCy · Industrial-strength Natural Language Processing in Python [Electronic resource] // SpaCy. — Mode of access: https://spacy.io.
13. Stanza Overview [Electronic resource] // Stanza. — Mode of access: https://stanfordnlp.github.io/stanza.
14. UDPipe [Electronic resource] // LINDAT/CLARIAH-CZ. — Mode of access: https://lindat.mff.cuni.cz/services/udpipe.
15. Ukrainian speaking countries [Electronic resource] // Worlddata.info. — Mode of access: https://www.worlddata.info/languages/ukrainian.php.

References

Claude. Anthropic. https://claude.ai.

CoNLL-U Format. Universal Dependencies. https://universaldependencies.org/format.html.

FastAPI. https://fastapi.tiangolo.com.

Gemini. Google AI. https://gemini.google.com.

GitHub - lang-word-stress: uk/ukrainianAdds word stress to Ukrainian texts. GitHub. https://github.com/lang-uk/ukrainian-word-stress.

GitHub - techwithanirudh/gpt4free: The official gpt4free. GitHub. https://github.com/techwithanirudh/gpt4free.

Kovaliv, Yu. I. (2007a). *Literaturoznavcha entsyklopediia* (T. 1). VTs “Akademiia” [in Ukrainian].

Kovaliv, Yu. I. (2007b). *Literaturoznavcha entsyklopediia* (T. 1). VTs “Akademiia” [in Ukrainian].

Liddy, E. (2001). Natural Language Processing. *Library and Information Science* (2nd ed.). Marcel Decker, Inc, NY, USA.

OpenAI. (2023). GPT-4 Technical Report. *Computation and Language*. https://doi.org/10.48550/arXiv.2303.08774.

Prohrama ZNO z ukrainskoi movy i literatury 2025 roku. (2025). Osvita.UA. https://osvita.ua/test/program_zno/946 [in Ukrainian].

SpaCy · Industrial-strength Natural Language Processing in Python. SpaCy. https://spacy.io.

Stanza Overview. Stanza. https://stanfordnlp.github.io/stanza.

UDPipe. CZ. LINDAT / CLARIAH. https://lindat.mff.cuni.cz/services/udpipe.

Ukrainian speaking countries. Worlddata.info. https://www.worlddata.info/languages/ukrainian.php.

O. Smysh, D. Shvets

UKRAINIAN POETRY ANALYSIS USING NLP METHODS

This paper presents the development and usage of a parser designed for the comprehensive analysis of Ukrainian-language poetry. The parser integrates techniques from natural language processing to determine metrical patterns, identify rhymes and rhyme schemes, and detect the presence of 16 tropes across three levels of linguistic processing: phonological, morphosyntactic, and semantic.

At the phonological level, rule-based methods are employed to detect stylistic devices such as alliteration, assonance, anaphora, epiphora, refrain, and rhetorical questions. For morphosyntactic analysis, the parser processes CoNLL-U formatted data generated by three language analysers (UDPipe, Stanza, spaCy) to ensure robustness in identifying epithets, inversions, parallelisms, similes, and rhetorical appeals. Semantic-level analysis is conducted using a large language model (LLM), Claude 3.5 Sonnet, which interprets metaphors, hyperbole, litotes, personification, and oxymora.

The parser includes a metrical analysis module that identifies stressed syllables and constructs rhythmic patterns to classify each poetic line into one of the canonical meters (iamb, trochee, dactyl, amphibrach, or anapest). It also identifies rhyme pairs and classifies rhyme schemes as tail, ring, or internal.

A web application was developed using FastAPI and React, enabling users to input poems and interactively visualize analytical results. This tool has been applied to three canonical Ukrainian poetry collections — by Taras Shevchenko, Lesia Ukrainka, and Vasyl Symonenko — demonstrating its capacity to uncover stylistic patterns across different periods and poetic styles.

Evaluation results show high accuracy: 94.39% for metre detection, 96.26% for rhyme identification, and 94.58% for trope recognition, outperforming three reviewed LLMs. The parser enables nuanced analysis of Ukrainian poetic texts, supporting educational, philological, and literary research.

Keywords: natural language processing, parser, language analyser, large language models, web application, Ukrainian language, poem, poem analysis, trope, metre, rhyme, rhyme scheme.

Матеріал надійшов 19.06.2025

Creative Commons Attribution 4.0 International License (CC BY 4.0)