Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп'ютерно-інформаційних систем та програмної інженерії
(повна назва факультету)
Кафедра програмної інженерії
(повна назва кафедри)

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня

магістр
(назва освітнього ступеня)

на тему: Система автоматичного формування блоку метаданих наукових метаданих наукових документів з використанням відкритих баз даних

Виконав(ла): студент(ка) 6 курсу, групи СПм-61
спеціальності 121 Інженерія програмного забезпечення
(шифр і назва спеціальності)

(підпис) Сучков С. С. (прізвище та ініціали)

Керівник (підпис) Бойко І. В. (прізвище та ініціали)

Нормоконтроль (підпис) Стоянов Ю. М. (прізвище та ініціали)

Завідувач кафедри (підпис) Петрик М. Р. (прізвище та ініціали)

Рецензент (підпис) (прізвище та ініціали)

Тернопіль
2024

РЕФЕРАТ

Кваліфікаційна робота магістра містить: 69 сторінок, 26 рисунків, 5 таблиць, 49 джерел та 2 додатки.

ВИДОБУТОК ДАНИХ, ДОПОВНЕННЯ МЕТАДАНИХ, МАШИННИЙ ПЕРЕКЛАД, МЕТАДАНІ, НАУКОМЕТРИЧНІ БАЗИ

Пропонований сервіс спрощує процес видобування метаданих з наукових документів і автоматизовує їх доповнення з відкритих джерел, підвищивши точність та повноту інформації. Сервіс дозволяє завантажувати PDF документи та видобувати різні метадані про авторів наукових документів, як то їх імена, адреси електронної пошти, афіліації, ORCID ідентифікатори тощо. Відсутні метадані будуть отримані з Google Scholar, WikiData та ORCID, а потім додані до вихідного файлу. Проект реалізований з використанням мови програмування Kotlin, а інтерфейс користувача розроблений за допомогою Jetpack Compose Multiplatform. Сервіс буде корисним інструментом для дослідників, науковців та інших фахівців, які працюють із науковими документами.

ABSTRACT

The qualifying thesis contains: 69 pages, 26 figures, 5 tables, 49 sources and 2 appends.

DATA MINING, METADATA SUPPLEMENT, MACHINE TRANSLATION, METADATA, SCIENCEMETRIC BASES

The proposed service simplifies the process of extracting metadata from scientific documents and automates their addition from open sources, increasing the accuracy and completeness of information. The service allows you to download PDF documents and extract various metadata about the authors of scientific documents, such as their names, e-mail addresses, affiliations, ORCID identifiers, etc. Missing metadata will be retrieved from Google Scholar, WikiData and ORCID and then added to the output file. The project is implemented using the Kotlin programming language, and the user interface is developed using Jetpack Compose Multiplatform. The service will be a useful tool for researchers, scientists and other professionals who work with scientific documents.

ПЕРЕЛІК СКОРОЧЕНЬ І ТЕРМІНІВ

API (Application Programming Interface) – опис способів взаємодії двох або більше програмних забезпечень.

CRF (Conditional Random Field) - сукупність методів статистичного моделювання, що активно застосовуються в галузі машинного навчання для реалізації різних завдань, наприклад, розпізнавання мови та образів, структурованого прогнозування, обробки текстової інформації.

GROBID (GeneRation Of Bibliographic Data) – бібліотека з відкритим вихідним кодом на основі машинного навчання, що дозволяє отримувати та перетворювати метадані з наукових статей.

JATS (Journal Article Tag Suite) – технічний стандарт, що визначає набір XML елементів та атрибутів для маркування заголовків наукових статей та описує їх моделі.

JSON (JavaScript Object Notation) - текстовий формат, призначений для передачі інформації.

REST (Representational State Transfer) – архітектурний стиль взаємодії компонентів програми у мережі.

XSL (extensible Stylesheet Language) – сімейство рекомендацій консорціуму W3C, що описує мови перетворення та візуалізації XML -документів.

XML (Extensible Markup Language) – це мова розмітки та формат файлів для зберігання, передачі та відновлення довільних даних

БД – база даних

Доповнення метаданих - процес пошуку відсутніх у науковому документі блоків метаданих та подальше їх додавання до результату при знаходженні у відкритих базах даних

Метадані – загальне поняття, яке відноситься до даних, що описують інші дані. Термін включає різні аспекти інформації, такі як зміст, формат, структуру і контекст. Метадані використовуються для оптимізації процесів організації,

пошуку, управління та інтерпретації інформації.

ОС – операційна система.

ПЗ – програмне забезпечення.

Семантична мережа – концептуальна модель, що представляє структуроване відображення знань чи інформації, що визначає зв'язки між поняттями та сутностями. Застосовується у сферах штучного інтелекту, когнітивної науки та інформаційного пошуку.

Уточнення метаданих – перевірка коректності видобутих метаданих за їх наявності у науковому документі та заміна на актуальне значення у разі виявлення помилки у сформованому блоці метаданих.

Цифрова математична колекція – це комплекс інструментів, призначених для полегшення пошуку необхідного матеріалу та забезпечення зручності його використання.

ЗМІСТ

ВСТУП

Актуальність теми дослідження. Щороку у світі відбувається закономірний приріст обсягу знань, з'являються нові відкриття та наукові здобутки. За даними Scopus, на кінець 2023 кількість наукових публікацій тільки в Україні є більшою за 302 тисячі. При збільшенні обсягу інформації зростає потреба у її обробці та зберіганні. Саме для цього і потрібні структуровані описи цих даних – метадані. Крім автоматизованих систем, що працюють з обробкою даних, вони потрібні також для створення користувачами запитів, аналізу даних та інтерпретації їхнього вмісту. У деяких джерелах також згадується їхня ключова роль у забезпеченні технічних стандартів та правил генерації записів, що значно спрощує процес роботи з даними.

Проте метадані можуть містити помилки і неточності з різних причин (наприклад, у випадках, коли автори мають одне й те саме ім'я). Ручне формування блоку метаданих також потребує деяких витрат часу.

В рамках поточного дослідження для вирішення згаданих проблем реалізується система автоматичного формування блоку метаданих наукових документів на основі отриманих даних за допомогою Google Scholar SERP API, WikiData API, ORCID API та Google Translate API для уточнення вихідних метаданих та доповнення недостатньої інформації. Система, що розглядається, дозволяє витягувати метадані на основі завантажених у систему наукових публікацій і формувати вихідний файл XML у форматі NISO JATS V1.0. Дані, які не були вказан, можуть бути доповнені за допомогою сервісів Google Scholar SERP API, WikiData API, ORCID API та Google Translate API.

Метою роботи є проектування та розробка системи видобування та уточнення метаданих з наукових документів, а також доповнення відсутніх фрагментів з відкритих БД.

Для досягнення мети потрібно вирішити наступні завдання:

– аналіз існуючих рішень, які отримують метадані наукових документів;

– аналіз відкритих БД для уточнення та доповнення метаданих;

– аналіз сервісів машинного перекладу ключових слів та анотацій;

– розробка системи запитів для пошуку в семантичній мережі інформації про статтю;

– реалізація методу уточнення та доповнення метаданих наукових документів;

– тестування методу із використанням колекції наукових документів;

– розробка програмного продукту як десктопного застосунку.

Об’єкт дослідження: автоматизація процесу формування блоку метаданих наукових документів.

Предмет дослідження: аналіз структури наукових документів, видобуток вихідних метаданих та обробка запитів до семантичних мереж.

Методи дослідження: наукові роботи українських та зарубіжних вчених за темою дослідження, фундаментальні положення програмної інженерії; методи - аналітичний, порівняльний, системного аналізу, індукції та проектування.

Наукова новизна роботи:

- удосконалено механізм видобування та доповнення метаданих наукової публікації;

- отримав подальшого розвитку алгоритм перекладу анотацій та ключових слів.

Практичне значення одержаних результатів. Розроблене програмне рішення забезпечить зниження складності процесів видобутку, уточнення і доповнення метаданих у наукових текстах., що дасть змогу зменшити вірогідність появи у них помилок.

Апробація. Окремі результати дослідження доповідалися на XII науково-технічній конференції «Інформаційні моделі, системи та технології» як опубліковані тези.

1 АНАЛІЗ ТА ПОСТАНОВКА ЗАВДАННЯ

1.1 Вимоги до програмного рішення

Система, що розробляється, повинна бути сумісна з ОС Windows, а також закладена можливість подальшого розширення до підтримки таких десктопних платформ, як macOS і Linux. Інструмент повинен обробляти вхідні дані у форматі PDF, що є науковим документом, і відображати видобуті метадані у вигляді XML -файлу. Крім того, після видобування метаданих користувачу повинна бути доступна функція доповнення та уточнення відсутніх фрагментів метаданих з відкритих джерел, таких як Google Scholar (https://scholar.google.com/), WikiData (https://www.wikidata.org/) та ORCID (https://orcid.org/). Також при доповненні метаданих має бути можливість здійснення перекладу ключових слів та анотації за допомогою інструменту машинного перекладу тексту англійською або українською мовами залежно від мови написання вихідної статті.

Результат роботи обробки даних на всіх етапах повинен бути XML- файлом у форматі NISO JATS V1.0 [7]. Ця вимога необхідна для збирання колекцій метаданих для відповідності міжнародним стандартам з метою включення до Всесвітньої цифрової математичної бібліотеки (World Digital Mathematical Library - WDML; передісторію цього рішення пояснив Peter J. Olver, див. [1]]) [15].

Необхідний процес роботи системи представлений на рис. 1.1.

Процес роботи системи, що розробляється, відбувається в 4 етапи:

1. Підвантаження до системи наукової статті. Потрібно вибрати документ у форматі .PDF або вказати шлях до файлу на пристрої для подальшої обробки.

2. Видобування метаданих. Після передачі вхідного файлу PDF виконується процедура видобутку метаданих, що містяться у файлі наукової статті, а саме: інформації про авторів, їх ролі, афіліації, електронні пошти, ключові слова та анотації.

3. Уточнення та доповнення раніше видобутих метаданих наукового

документа. Цей процес включає відправлення системи запитів до відкритих джерел даних таким, як WikiData, Google Scholar і ORCID. Також відбувається взаємодія з Google Translate для перекладу ключових слів та анотації англійською або українською мовами за відсутності відповідних блоків у вихідній статті.

4. Відображення сформованих метаданих у вигляді файлу XML відформатованого за стандартом NISO JATS V1.0.

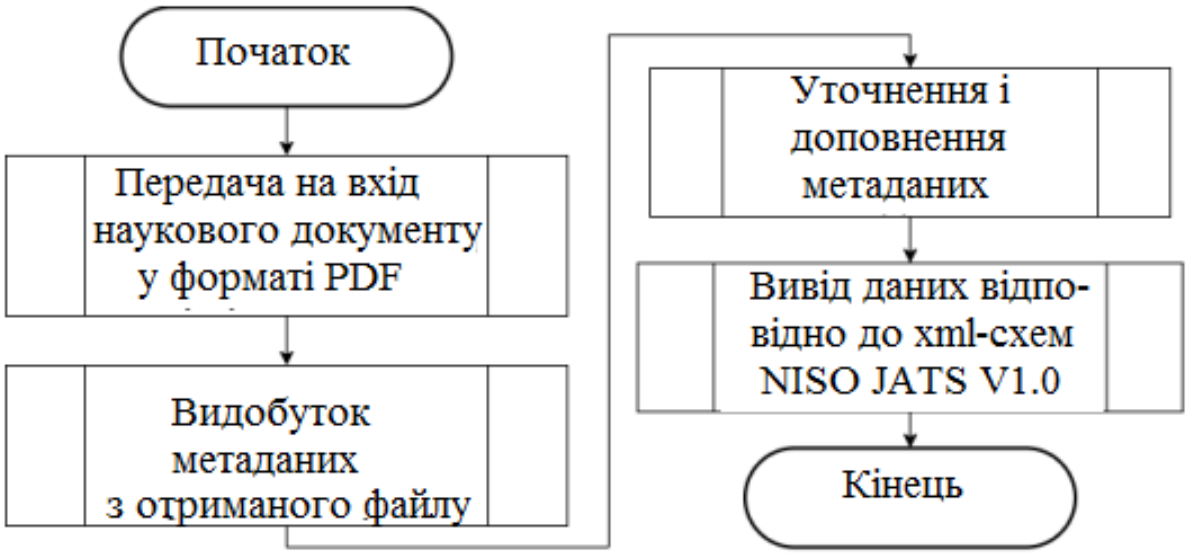


Рисунок 1.1 – Основні кроки роботи системи видобутку, уточнення та доповнення метаданих наукових праць

Виходячи з перерахованих вище вимог, як завантажені документи для тестування системи було вирішено використовувати наукові статті цифрової математичної бібліотеки. На виході буде представлений результат роботи системи у вигляді сформованих метаданих у вигляді файлу XML , відформатованого згідно NISO JATS.

Таким чином, планується скорочення часу на створення електронної колекції математичних документів за рахунок автоматизації процесів видобування, уточнення та доповнення метаданих наукових статей. Також підвищиться точність метаданих через звернення до кількох відкритих джерел інформації.

1.2 Інструменти розробки

Нижче наведено список інструментів розробки, які були використані для реалізації програмного рішення автоматичного формування блоку метаданих наукових документів:

- IntelliJ IDEA – інтегроване середовище розробки ПЗ на Java та інших мовах JVM, таких як Kotlin, Scala та Groovy. Включає велику кількість зручних інструментів для програмування, завдяки чому є одним з найпопулярніших інструментів [16].
- Kotlin – об'єктно-орієнтована мова програмування зі статичною типізацією. Однією з переваг є сумісність із Java, оскільки Kotlin працює на віртуальній машині Java (JVM) [17];
- Gradle – система складання, яка дозволяє описувати конфігурацію проекту мовами Groovy та Kotlin. Гнучкість конфігурацій та підтримка інкрементальних складання Gradle прискорює розгортання програмних проектів і скорочує час складання [18];
- Compose Multiplatform - декларативний мультиплатформний фреймворк для створення інтерфейсів користувача на Kotlin. Даний інструмент дозволяє реалізувати єдиний інтерфейс користувача для декількох платформ [19];
- GROBID - бібліотека на основі машинного навчання для видобування, аналізу та структурування необроблених документів формату PDF у структуровані XML файли, розмічені згідно з Text Encoding Initiative (TEI) [20];
- SPARQL – стандартизована мова запитів та протокол для роботи з

даними формату .RDF (Resource Description Framework). Використовується для формування та передачі запитів до семантичної мережі Wikidata;

- Wikidata Toolkit (https://www.mediawiki.org/wiki/Wikidata_Toolkit) - це Java -бібліотека з відкритим вихідним кодом для використання даних з Wikidata та інших сайтів Wikibase. Її основна мета - полегшити зовнішнім розробникам можливість використовувати ці дані у власних додатках;
- Google Scholar Serp API [3] - інструмент, призначений для пошуку та збирання даних зі сторінок результатів пошуку Google Scholar;
- Docker - ПЗ, створене для автоматизації розгортання та керування програмами з використанням контейнерних технологій;
- Kotlin Coroutines – співпрограми, які призначені для виконання асинхронних та паралельних операцій [21];
- Ktor – фреймворк, створений JetBrains для створення веб-додатків у Kotlin, що використовує Kotlin Coroutines для високої масштабованості та пропонує простий у використанні API [22];
- ORCID API [5] - програмний інтерфейс програми, наданий ORCID (Open Researcher and Contributor ID), який дозволяє інтегрувати та автоматизувати обмін інформацією між системами дослідників та різними платформами наукової спільноти. API ORCID використовується для програмного доступу до унікальних ідентифікаторів дослідників та пов'язаних з ними даних, що забезпечує більш ефективне управління науковими профілями та публікаціями;
- Google Translate API [6] - програмний інтерфейс програми, який надає компанія Google, який дозволяє інтегрувати функції машинного перекладу в різні сервіси. API підтримує безліч мов і забезпечує можливість перекладу тексту,

визначення мови вихідного тексту та отримання пропозицій щодо виправлення помилок у тексті.

1.3 Склад набору метаданих

Для ефективної інтеграції сервісів усередині цифрових бібліотек, а також для забезпечення їх сумісності із зовнішніми бібліотечними системами та БД критично важливо приділяти увагу узгодженості форматів метаданих, що використовуються в цих інформаційних ресурсах. Цифрові математичні бібліотеки DML-CZ (Czech Digital Mathematics Library, https://dml.cz/), Numdam (http://www.numdam.org/) та EuDML (The European Digital Mathematics Library, https://initiative). eudml.org/) використовують XML- схеми NISO JATS V1.0 для опису публікацій з математичних видань згідно з міжнародними стандартами, прийнятими для формування WDML. Як наслідок, колекції метаданих повинні відповідати міжнародним стандартам з метою інтеграції у Всесвітню цифрову математичну бібліотеку.

Стандарт NISO JATS версії 1.0 є набір тегів для структурування та опису наукових статей у форматі XML. Даний стандарт включає множину полів, які забезпечують детальний опис статті, інформацію про конкретні теги можна знайти [23]. Нижче наведений фрагмент XML -коду ілюструє компонування наукової статті відповідно до цього стандарту (рис. 1.2).

Набори даних, згідно з розробленими стандартами EuDML [24, 25], розрізняються за ступенем необхідності їх включення в XML -документ:

- обов'язкові (Mandatory). Це елементи, які мають бути присутніми у кожному JATS -документі. Вони необхідні для того, щоб документ відповідав мінімальним вимогам стандарту. Приклади обов'язкових елементів включають <article-id> (унікальний ідентифікатор статті), <article-title> (назва мовою

оригіналу) та <contrib-group> (список авторів);

- фундаментальні (Fundamental). Ці елементи вважаються основними для структури та змісту наукової статті, але вони не завжди є обов'язковими для кожної статті. Вони включають такі елементи, як <abstract> (анотація) та <kwd-group> (ключові слова), які надають основну інформацію про статтю та її зміст;
- додаткові (Supplemental). Елементи, які можуть бути додані в JATS -документ для збагачення інформації, але не є необхідними для відповідності стандарту. Приклади включають <ref-list> (список літератури), <funding-group> (дані про фінансування дослідження), <ext-link> (різні посилання з додатковою інформацією) та інші, які можуть надавати додаткові дані про статтю або її авторів.

```
<article>
    <front>
        <article-meta>
            <title-group>
                <article-tiь1e>Назва статті</article-title>
            </title-group>
            <contrib-group>
                <contrib contrib-type="author">
<name>
                        <surname>Сучков</surname>
                        <given-names>С.С.</given-names>
                    </name>
                    <xref ref-type="aff" rid="affO"/>
                </contrib>
                <!— Інші автори -->
            </contrib-group>
            <!-- Аффіляція -->
            <aff id="aff1">
                <institution>Назва університету</institution>
                <addr-line>Адреса університету</addr-line>
                <country>Країна</country>
            </aff>
            <abstract>
                <p>Анотація статті...</p>
            </abstract>
            <kwd-group>
                <kwd>Ключове слово 1</kwd>
                <kwd>ККлючове слово 2</kwd>
                <!— Додаткові ключові слова --> </kwd-group>
        </article-meta>
    </front>
</article>
```

Рисунок 1.2 – Приклад структури файлу XML згідно стандарту NISO JATS V1.0.

Дані розмежування блоків метаданих також актуальні і для системи, котра розробляється.

1.4 Огляд існуючих рішень

У цьому підрозділі проводиться огляд існуючих інструментів, які дозволяють видобувати та перекладати метадані з PDF -документів. Спочатку документ обробляється, після чого одержуються необхідні відомості - імена авторів, їх електронні адреси та ролі, ORCID ідентифікатори та афіліації.

Інструменти перекладу в роботі використовуються для формування блоку

додаткових метаданих у вигляді перекладу анотації та ключових слів наукової статті, що обробляється.

1.4.1 Інструменти для видобутку метаданих з наукових документів

GROBID - це бібліотека на основі машинного навчання для отримання, аналізу та структурування необроблених документів формату PDF у структуровані XML файли, розмічені згідно з Text Encoding Initiative (TEI). Принцип дії ґрунтується на моделі умовних випадкових полів (CRF), які були навчені на великій кількості наукових документів [26]. Різні блоки тексту певного типу за допомогою регулярних виразів витягуються на основі припущень про структуру та зміст блоків. Спочатку виділяються основні типи вмісту (наприклад, заголовок, титульна сторінка, колонтитули, основний текст, бібліографічні посилання), потім основних типів визначаються більш точні (наприклад заголовок, анотація).

GROBID знаходиться у відкритому доступі і написаний мовою Java, що дозволяє при необхідності додавати різні покращення до програмного коду. Інструмент активно підтримується і має чимало користувачів, що забезпечує його актуальність і сумісність з останніми технологіями. За результатами досліджень має одне з найвищих середніх значень точності інструменту – 74.9% при видобуванні метаданих статті та 78.9% при видобуванні метаданих бібліографічних посилань [27].

PdfAct є одним інструментом з високою точністю отримання біографічної інформації. Це програмний засіб для вибіркового видобування текстових блоків із PDF файлів. Однією з ключових особливостей PdfAct є видобування головних частин тексту (таких як заголовок, назва розділів та абзаци основного тексту) з певним ступенем деталізації (літери, слова, рядки тексту або параграфи).

Крім перерахованого бібліотека має додаткові можливості, такі як:

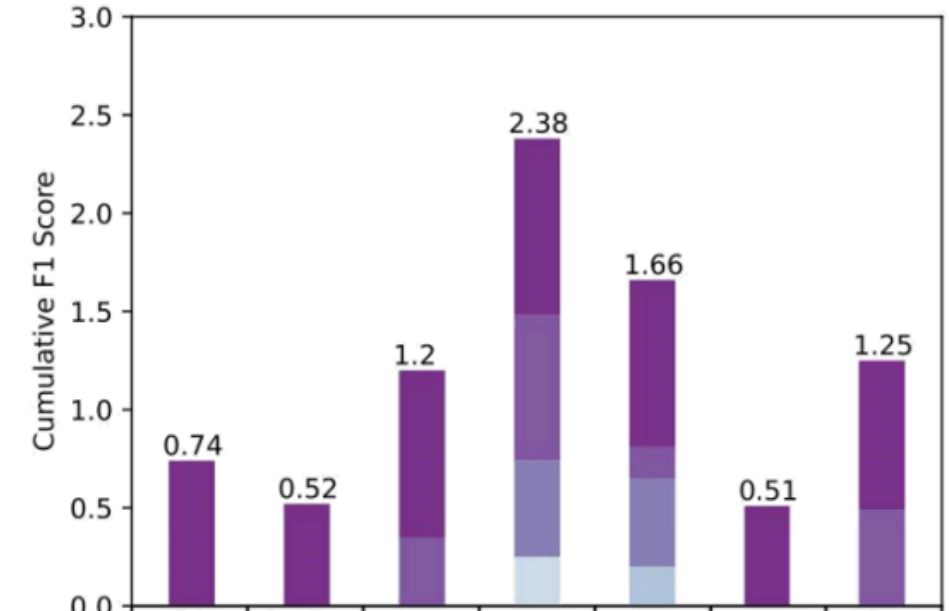
- видобування інформації, пов'язаної з версткою (наприклад, назва шрифту та колір тексту);
- визначення меж абзаців;
- встановлення черговості читання текстових блоків;
- переклад лігатур та символів з діакритичними знаками;
- виявлення семантичного значення текстових блоків;
- поєднання слів, розділених дефісом.

За результатами досліджень [28] у сервісів GROBID та PdfAct найвищі показники точності видобування даних. Середній показник F-балу складає більше 0,9 при отриманні метаданих із повних текстів статей. GROBID пропонує великий спектр функцій для отримання метаданих наукових праць. Крім стандартних бібліографічних метаданих, він також витягує структурні метадані. У порівнянні з PdfAct, GROBID має більшу точність у визначенні авторства, але з іншого боку - обробка великих колекцій документів займає більше часу. Основний мінус PdfAct – інструмент знаходиться на етапі розробки та все ще має велику кількість багів, які ускладнюють використання інструменту на багатьох наукових документах. Для отримання найкращих результатів дані повинні бути ретельно підготовлені та структуровані, що не підходить у рамках даної роботи, оскільки метадані наукових публікацій здебільшого не піддаються суворій шаблонізації.

На рис. 1.3 та 1.4 зображено графік та порівняльна таблиця, в яких представлені значення вимірювання продуктивності різних інструментів.

Проаналізувавши графік і таблицю з рис. 1.3 можна бачити, що підсумкові значення за одним і тим самим параметрам мають кардинальні відмінності залежно від типу даних. Тому досить важко вибрати єдиний інструмент, який би відмінно справлявся відразу з усіма типами даних.

Чимало елементів контенту, видобуток яких відбувається неякісно або зовсім не підтримується, вказує на високий потенціал для доробок у майбутньому.



	Adobe Extract	Apache Tika	CERMINE	GROBID	PdfAct	PyMuPDF	Science Parse
Paragraph	0.74	0.52	0.85	0.90	0.85	0.51	0.76
Section	0.00	0.00	0.35	0.74	0.16	0.00	0.49
Caption	0.00	0.00	0.00	0.49	0.45	0.00	0.00
List	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Footer	0.00	0.00	0.00	0.00	0.20	0.00	0.00
Equation	0.00	0.00	0.00	0.25	0.00	0.00	0.00

Рисунок 1.3 – Значення продуктивності існуючих рішень для видобування метаданих із PDF документів [28]

За результатами експерименту [28] можна зробити такі висновки:

- GROBID та PdfAct — унікальні інструменти, здатні до часткового видобування підписів;
- GROBID отримав найвищий бал F1, рівний 0,74, завдання видобування розділів;
- тільки GROBID здатний отримувати рівняння, проте якість досить низькому рівні, оскільки F1 = 0,25;

тільки PdfAct здатний видобувати нижні колонтитули, та його ефективність оцінюється низьким балом F1, рівним 0,20.

Tool[1]	Label	# Detected	# Processed[2]	*Acc*	F_1	*P*	*R*
Adobe Extract	Table	1,635	736	**0.52**	**0.47**	**0.45**	**0.49**
	Paragraph	3,985	3,088	0.85	0.74	0.72	0.76
Apache Tika	Paragraph	339,603	258,582	0.55	0.52	0.43	0.65
Camelot	Table	16,289	**11,628**	0.27	0.30	0.23	0.44
CERMINE	Title	16,196	14,501	0.84	0.81	0.81	0.81
	Author	**19,788**	14,797	0.43	0.44	0.44	0.46
	Abstract	19,342	16,716	0.71	0.72	0.68	0.76
	Reference	**40,333**	35,193	0.80	0.74	0.71	0.77
	Paragraph	361,273	348,160	0.89	0.85	0.83	0.87
	Section	**163,077**	139,921	0.40	0.35	0.32	0.38
GROBID	Title	16,196	16,018	**0.92**	**0.91**	**0.91**	**0.92**
	Author	**19,788**	**19,563**	**0.54**	**0.52**	**0.52**	**0.53**
	Abstract	19,342	**18,714**	0.82	**0.82**	**0.81**	0.83
	Reference	**40,333**	**36,020**	**0.82**	**0.79**	**0.79**	**0.80**
	Paragraph	361,273	**358,730**	**0.90**	**0.90**	**0.89**	**0.91**
	Section	**163,077**	**163,037**	**0.77**	**0.74**	**0.73**	**0.76**
	Caption	**90,606**	**62,445**	**0.57**	**0.49**	**0.47**	0.51
	Table	**16,740**	8,633	0.24	0.23	0.23	0.23
	Equation	**142,736**	**96,560**	**0.26**	**0.25**	**0.20**	**0.32**
PdfAct	Title	**17,670**	**16,834**	0.85	0.85	0.85	0.86
	Author	13,110	2,187	0.14	0.13	0.12	0.18
	Abstract	**21,470**	4,683	0.17	0.16	0.15	0.20
	Reference	30,263	12,705	0.19	0.15	0.17	0.20
	Paragraph	**361,318**	357,905	0.85	0.85	0.80	0.89
	Section	129,361	87,605	0.21	0.16	0.12	0.25
	Caption	83,435	53,314	0.45	0.45	0.40	**0.52**
	Footer	**32,457**	**26,252**	**0.23**	**0.20**	**0.25**	**0.16**
PyMuPDF	Paragraph	339,650	258,383	0.55	0.51	0.41	0.65
RefExtract	Reference	**40,333**	38,405	0.55	0.49	0.44	0.55
Science Parse	Title	11,696	11,687	0.79	0.70	0.70	0.70
	Author	471	471	**0.54**	**0.52**	**0.52**	**0.53**
	Abstract	14,150	14,149	**0.83**	0.81	0.73	**0.90**
	Reference	**40,333**	35,200	0.55	0.49	0.49	0.50
	Paragraph	**361,318**	355,529	0.79	0.76	0.76	0.76
	Section	**163,077**	158,556	0.54	0.49	0.49	0.50
Tabula	Table	10,361	9,456	0.29	0.28	0.20	0.46

Рисунок 1.4 – Розбивка бенчмарків інструментів видобування метаданих з PDF документів [28]

Для задачі в рамках цієї роботи GROBID демонструє перевагу над іншими подібними інструментами (рис. 1.4). Такий результат досягається шляхом використання різних типів сегментації, які застосовуються до різних форматів інформації.

Ці методи дозволяють ідентифікувати ключові сфери документа на основі унікальних характеристик його структури. В результаті, елементи тексту, які часто зустрічаються, не перешкоджають процесу видобутку елементів, що рідко

зустрічаються, з блоків, що не входять в головну частину тексту. Каскадні моделі, які активно використовуються в GROBID, також сприяють гнучкості налаштування кожної моделі.

Виходячи з результатів проведеного аналізу, GROBID показав найбільш ефективні результати, що спричинило вибір даного інструменту для видобування метаданих з наукових статей.

1.4.2 Вибір сервісу для машинного перекладу ключових слів та анотації

Існує кілька найбільш популярних сервісів для перекладу текстів:

- DeepL Translator (https://www.deepl.com/translator) - використовує нейронні мережі глибокого навчання для обробки природної мови та перекладу текстів між різними мовами. Сервіс спирається на великі мовні дані та алгоритми для моделювання мовних структур та контексту, що дозволяє досягати високої точності та природності перекладу;
- Google Translate (https://translate.google.com/) - використовує статистичні та нейронні методи обробки природної мови для перекладу слів, фраз та веб-сторінок між різними мовами. Сервіс інтегрує технології машинного навчання та великі обсяги даних для покращення якості перекладу;
- Microsoft Translator (https://translator.microsoft.com/) - надає функції перекладу тексту та мовлення в реальному часі. Також використовує нейронні мережі та методи машинного навчання для забезпечення перекладу з підтримкою багатьох мов, а також пропонує інтеграцію з іншими продуктами та сервісами Microsoft;
- Baidu Translate (https://fanyi.baidu.com) - розроблений китайським

гігантом Baidu. Застосовує комбінацію статистичних методів та нейронних мереж для перекладу текстів та веб-сторінок різними мовами, прагнучи забезпечити високу точність та розуміння контексту перекладених матеріалів.

Для порівняльного аналізу чотирьох сервісів перекладу, які можна інтегрувати в систему, було розглянуто такі ключові аспекти: підтримка мов, функціональність і доступність. Результати порівняльного аналізу представлені у табл. 1.1.

Таблиця 1.1 – Порівняння сервісів машинного перекладу тексту з метою інтеграції в систему, що розробляється

Критерій	DeepL Translator	Google Translate	Microsoft Translator	Baidu Translate
Підтримка мов	близько 30	понад 100	понад 70	понад 100
Наявність API	DeepL API	Google Cloud Translation API	Microsoft Translator Text API	Baidu Translate API
Вартість	Є версія безкоштовна	Є версія безкоштовна	Є версія безкоштовна	Є версія безкоштовна
Доступність API	Сервіс доступний	Сервіс доступний	Сервіс доступний	Сервіс доступний обмежено

Серед розглянутих варіантів, основними критеріями вибору сервісу для здійснення перекладу є можливість інтеграції в систему, що розробляється, доступність і точність перекладу.

Відповідно до дослідження, результати якого представлені на рис. 1.5 та 1.6, всі розглянуті послуги машинного перекладу, немає істотних відмінностей у якісних показниках [29].

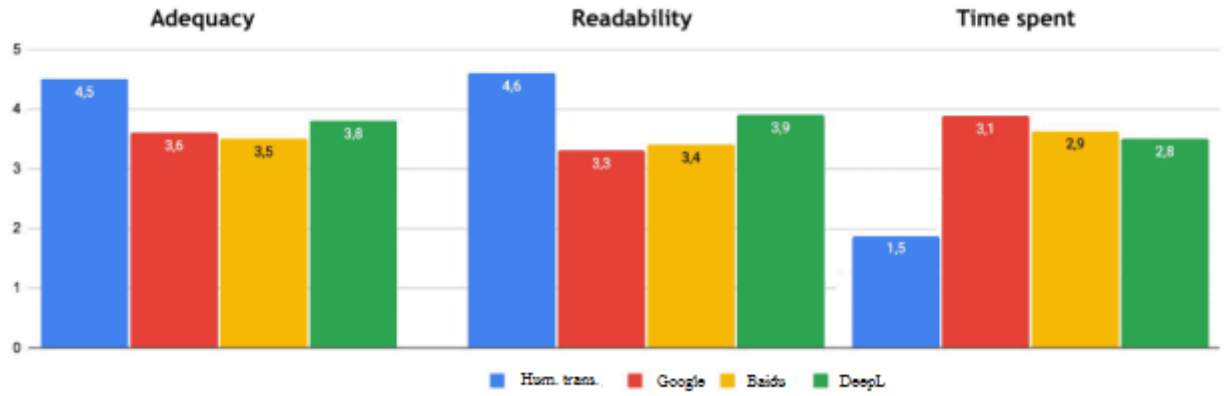


Рисунок 1.5 – Графіки показників сприйняття перекладеного тексту ручним та машинним способами з української на англійську [29]

Рисунок 1.6 – Графіки показників сприйняття перекладеного тексту ручним та машинним способами з англійської на українську [29]

На зображених графіках усі показники були оцінені за шкалою від 1 до 5:

- адекватність (Adequacy) показує, наскільки коректно передано зміст тексту, і чи є якісь зміни між початковим та підсумковим текстами. Оцінка за шкалою знаходиться в діапазоні від 1 до 5, де 1 говорить про серйозні зміни в сенсі тексту, а 5 свідчить про повну відповідність сенсу оригіналу;
- читабельність (Readability) пов'язана з простотою читання та сприйняття перекладеного тексту. Оцінка 1 вказує на те, що переклад не

піддається розумінню, тоді як оцінка 5 свідчить про те, що переклад легко сприймається, а сенс зрозумілий і не викликає двозначності;

- записи часу редагування (Time spent) фіксує рівень зусиль, необхідних постредагування тексту перед його використанням. Оцінка 1 вказує на мінімальні зусилля, тоді як 5 вимагає повного переписування перекладу.

Важливим критерієм є доступність сервісів, тому серед розглянутих найкращим вибором є Google Translate API.

У випадку системи, що розробляється, було вирішено, що сервіс Google Translate якнайкраще підходить для реалізації прототипу системи.

1.5 Висновок до першого розділу

У першому розділі встановлюються вимоги до системи, що розробляється, визначаються інструменти розробки та склад набору метаданих. Також проведено огляд існуючих рішень для видобутку метаданих із наукових документів та для перекладу наукового тексту.

За результатами проведеного аналізу і порівняння можливосей вибрано GROBID як інструмент для видобування метаданих та Google Translator для проведення перекладу наукових статей.

2 ПРОЕКТУВАННЯ СИСТЕМИ

2.1 Огляд відкритих баз даних наукових публікацій

Існують кілька найбільш популярних відкритих БД із науковими публікаціями. Варто навести основні з них:

- Scopus (https://www.scopus.com/) - єдина бібліографічна та реферативна БД наукової літератури, що рецензується. Доступ до Scopus здійснюється за інституційною підпискою. Більша кількість робіт досягається шляхом менш суворих правил для публікацій. Для ручного вивантаження даних є ліміт на 2000 публікацій, але є безкоштовний API, де обмеження на розвантаження даних відсутні [30];
- ORCID (https://orcid.org/) - міжнародна система персональної ідентифікації авторів наукових публікацій. Особливість проекту полягає у видачі авторам персональних кодів, що дозволяють виключити різночитання у їхніх іменах. Є ORCID API для вивантаження даних [32];
- Google Scholar (https://scholar.google.com/) - безкоштовний сервіс пошуку наукових статей. Крім наукових публікацій містить також препринти, постпринти, технічну документацію і навіть блоги - частково саме тому містить найбільша кількість публікацій серед інших баз [33]. Є сервіс для пошуку статей та отримання метаданих із наукових праць;
- WikiData (https://www.wikidata.org/) - відкрита та вільна база знань, доступна для редагування та використання як людьми, так і автоматизованими системами. Вона функціонує як єдине сховище структурованої інформації для проектів Вікімедіа та зовнішніх ресурсів. Вміст платформи поширюється за умовами відкритої ліцензії, доступні функції експортування до стандартних форматів та інтеграції з іншими відкритими даними [34].

У табл. 2.1 наведено порівняння відкритих БД, які підходять для подальшого доповнення метаданих.

Таблиця 2.1 – Порівняння відкритих БД наукових публікацій

Назва	Наявність API	Вільний доступ	Кількість наукових праць
Scopus	+	-	77 млн
ORCID	+	- /+	понад 14 млн акаунтів
Google Scholar	+	+	понад 390 млн
WikiData	+	+	близько 2 млн

Виходячи з порівняння за трьома критеріями, вибір для початку розробки впав на Google Scholar, WikiData і ORCID. В інших ресурсів або відсутні API для зручного пошуку даних, або є обмеження в отриманні доступів [35]. Що стосується ORCID, є API з обмеженим функціоналом та кількістю даних, чого буде достатньо для реалізації прототипу.

2.2 Модель роботи програмного рішення

Розроблений програмний продукт є десктопною програмою для ОС Windows, яка автоматизує процес формування блоків метаданих наукових документів шляхом видобування метаданих з наукової роботи, поданої на вхід, і доповнення відсутніх блоків з відкритих БД.

Процес роботи програми можна розбити окремі етапи, де кожен наступний етап виконується з урахуванням результату попереднього. Принцип роботи

системи в загальному вигляді представлений на рис. 2.1 у вигляді UML-діаграми видів діяльності.

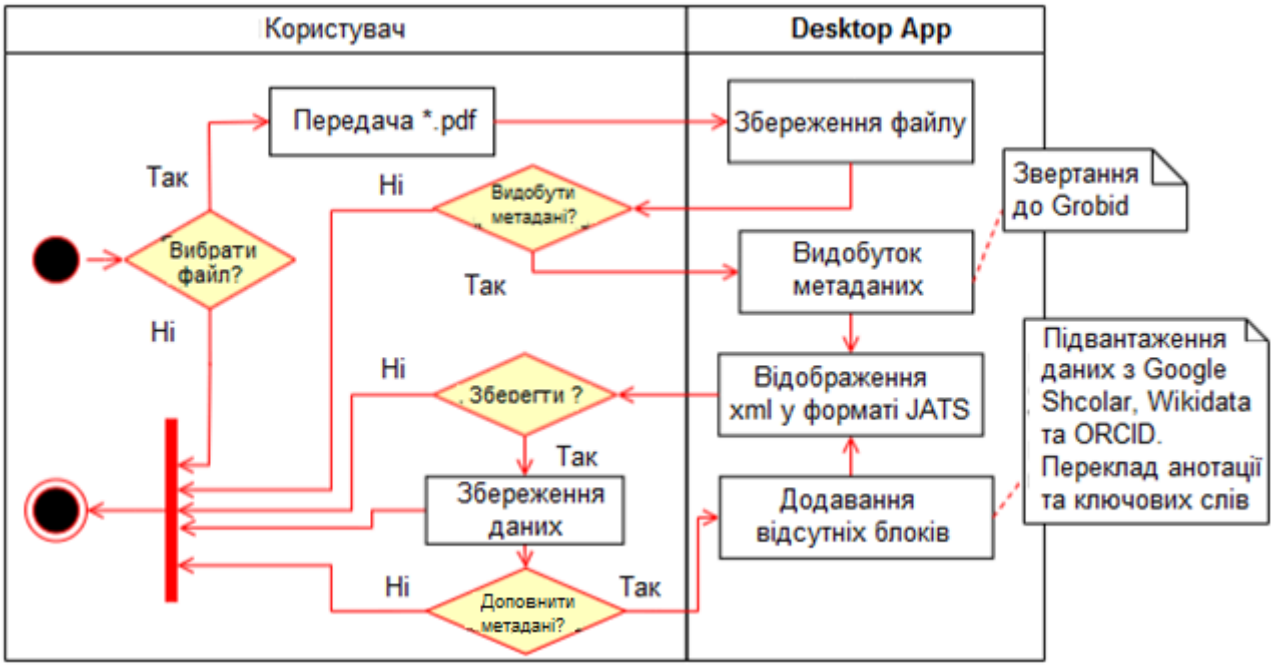


Рисунок 2.1 – Модель роботи системи у вигляді UML-діаграми видів діяльності

Загалом роботу системи можна описати так: користувач подає на вхід науковий документ у вигляді .pdf -файлу, далі користувач може скористатися функцією видобування метаданих, яка проаналізує документ, виділить блоки метаданих за допомогою сервісу Grobid і виведе дані у форматі NISO JATS. Наступним кроком користувачеві доступна функція доповнення метаданих, яка проаналізує раніше отриманий xml файл і здійснить пошук даних у Google Scholar, WikiData та ORCID, а також здійснить переклад анотації та ключових слів у Google Translate. Отримані результати будуть оформлені в єдиний файл XML і виведені на екран. На кожному етапі роботи користувач може зберегти результат.

2.3 Функціональні можливості системи

Система формування блоку метаданих реалізована як десктопний застосунок для ОС Windows.

Функціональні можливості застосування у загальному вигляді описані на діаграмі використання, наведеній на рис. 2.2. На діаграмі відображено основні варіанти використання для актора – користувач.

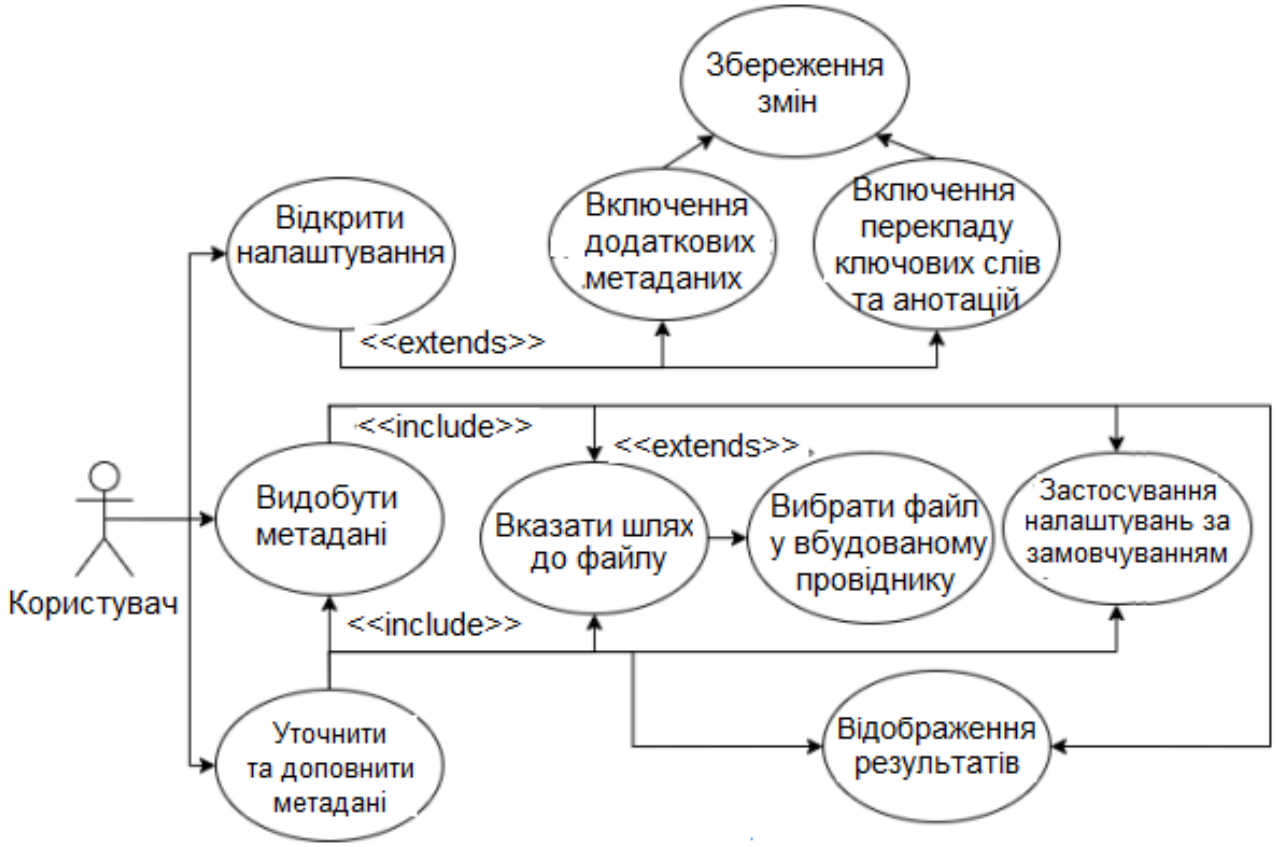


Рисунок 2.2 – Діаграма прецедентів розробленого сервісу

Далі докладніше розглядається кожен із етапів.

Після запуску десктопного застосунку користувач потрапляє на головний екран (рис. 2. 3), в якому відображені наступні елементи інтерфейсу користувача:

- рядок для введення та відображення шляху до PDF-файлу наукового документа, за яким необхідно сформувати блок метаданих;
- компонент "Вибрати файл", при натисканні на який відкриється

діалогове вікно для вибору PDF-файлу для обробки (рис. 2.4). У цьому вікні здійснено фільтрацію даних і для користувача відображаються лише папки та PDF файли;

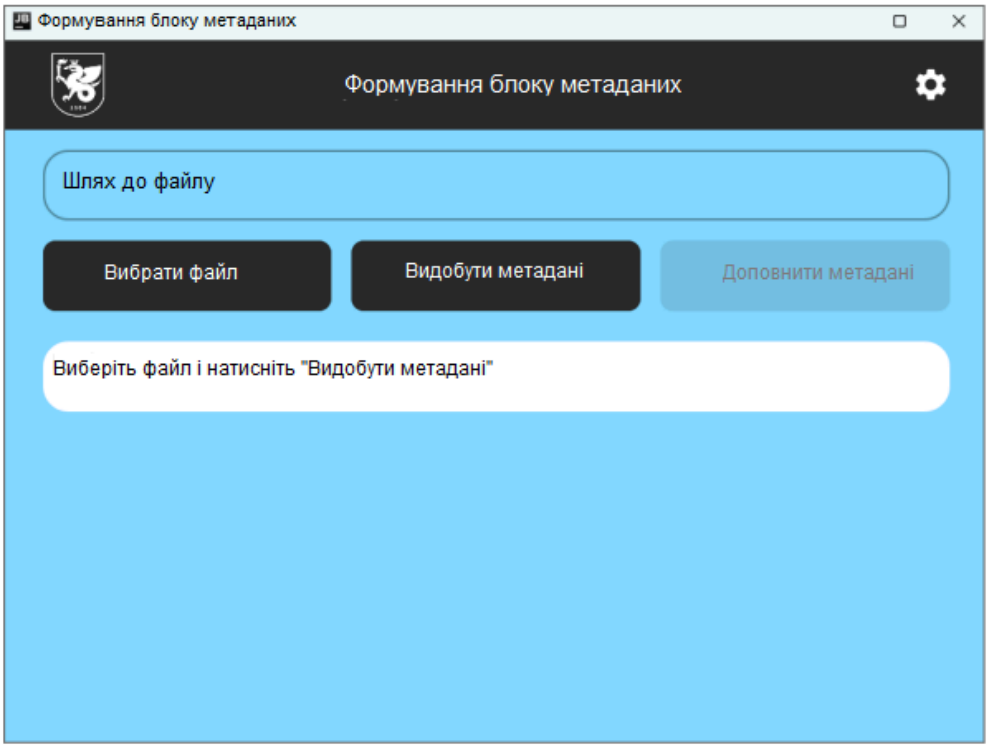


Рисунок 2.3 – Головний екран програми -десктопа після запуску

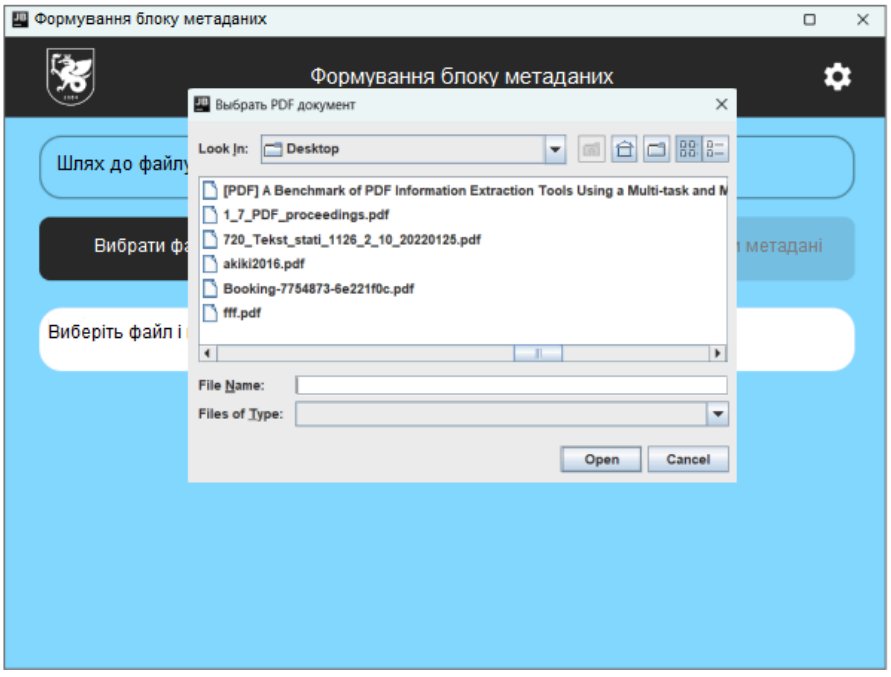


Рисунок 2.4 – Діалогове вікно вибору PDF-файлу. Дані відфільтровані та

користувачу відображено лише папки та PDF-файли

- компонент «Видобути метадані», при натисканні на який запуститься процес видобутку метаданих із вибраного документа. У разі некоректності шляху до файлу або неможливості видобутку даних з'явиться відповідна помилка (рис. 2.5);
- компонент «Доповнити метадані» стає активним після завершення процесу видобутку метаданих. При натисканні на нього проводиться уточнення та доповнення даних з відкритих джерел інформації;
- компонент під кнопками дій служить для відображення стану завантаження, підказок, помилок та результатів роботи системи користувача;
- компонент "Налаштування" у вигляді шестерні на панелі інструментів, необхідний для відкриття діалогового вікна з налаштуваннями складу метаданих.

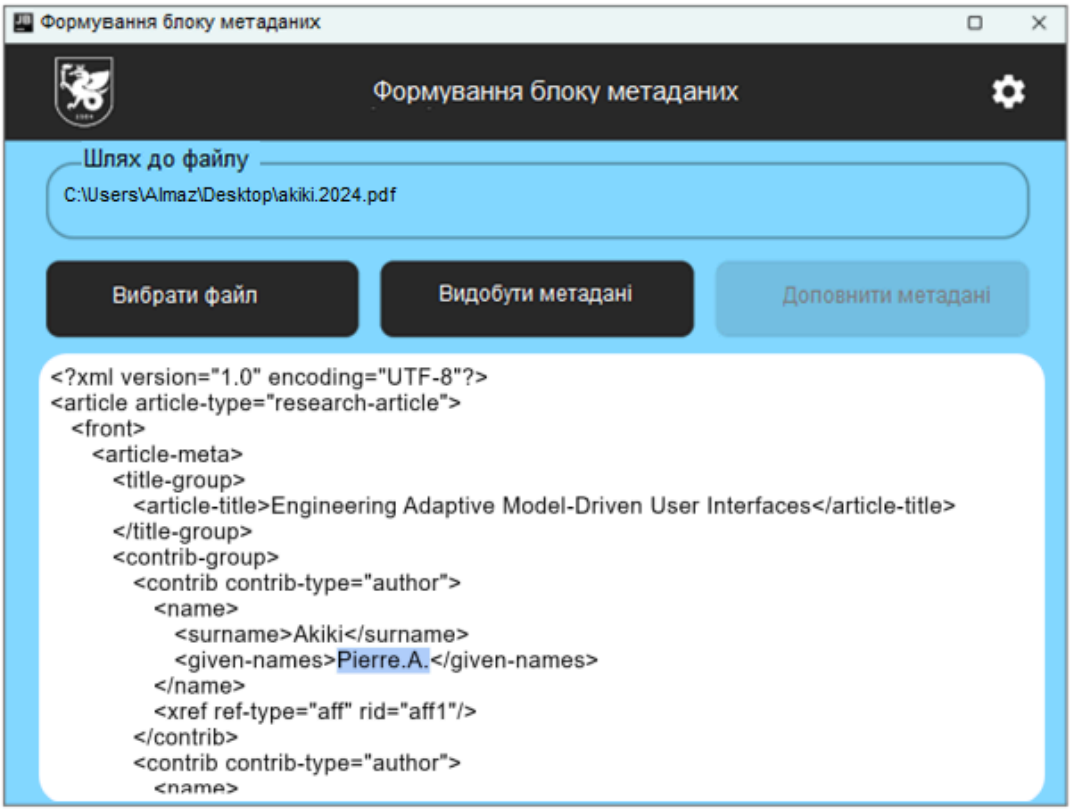


Рисунок 2.5 – Відображення результату видобування метаданих із файлу, який знаходиться на пристрої шляхом «C:\ Users\Almaz\Desktop\akiki2024.pdf"

Система показує обов'язкові та фундаментальні метадані, якщо є необхідність включення до кінцевого результату додаткових метаданих, користувач повинен проставити відповідний прапорець у меню налаштувань (рис. 2. 6), яке можна відкрити при натисканні на шестірню зліва у верхньому кутку.

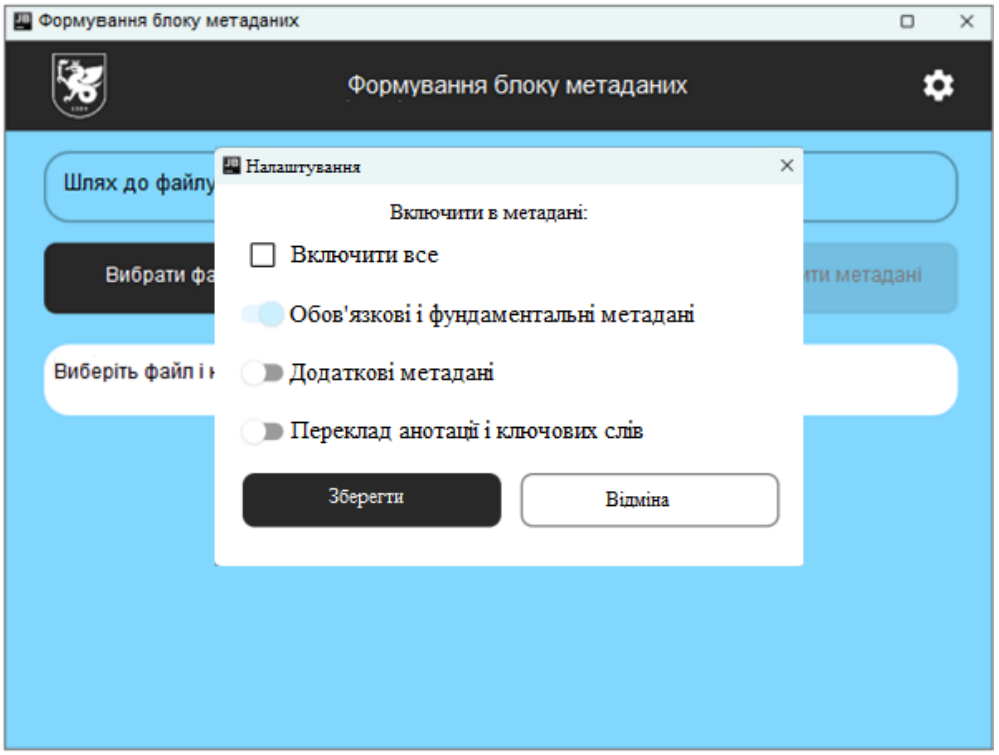


Рисунок 2.6 – Діалогове вікно налаштувань складу метаданих. Пункт «Обов'язкові та фундаментальні метадані» завжди в активному стані та недоступний для зміни

При зміні параметрів автоматично виконуються останні процеси, котрі здійснені користувачем. Наприклад, якщо користувач вийняв метадані з документа, а потім встановив прапорець "Додаткові метадані", то з наукового документа в автоматичному режимі також видобуватися блоки додаткових метаданих. У разі зняття галочки додаткові метадані будуть виключені з результату сформованих метаданих.

Інтерфейс користувача десктопної програми був розроблений за допомогою фреймворку Compose Multiplatform і в подальшому при розширенні на інші платформи може бути перевикористаний.

2.4 Архітектура проекту

За рахунок дотримання принципів clean architecture [37] при побудові

системи та запозичення ідей з різних архітектурних патернів вдалося домогтися поділу шару даних, бізнес-логіки та шару представлення, що дозволяє надалі оновити, замінити або перевикористовувати ту чи іншу частину коду, не вносячи значних змін до інших шарів.

Архітектура розробленої системи представлена у вигляді діаграми компонентів, яка описує зв'язки всередині програмного рішення (рис. 2.7).

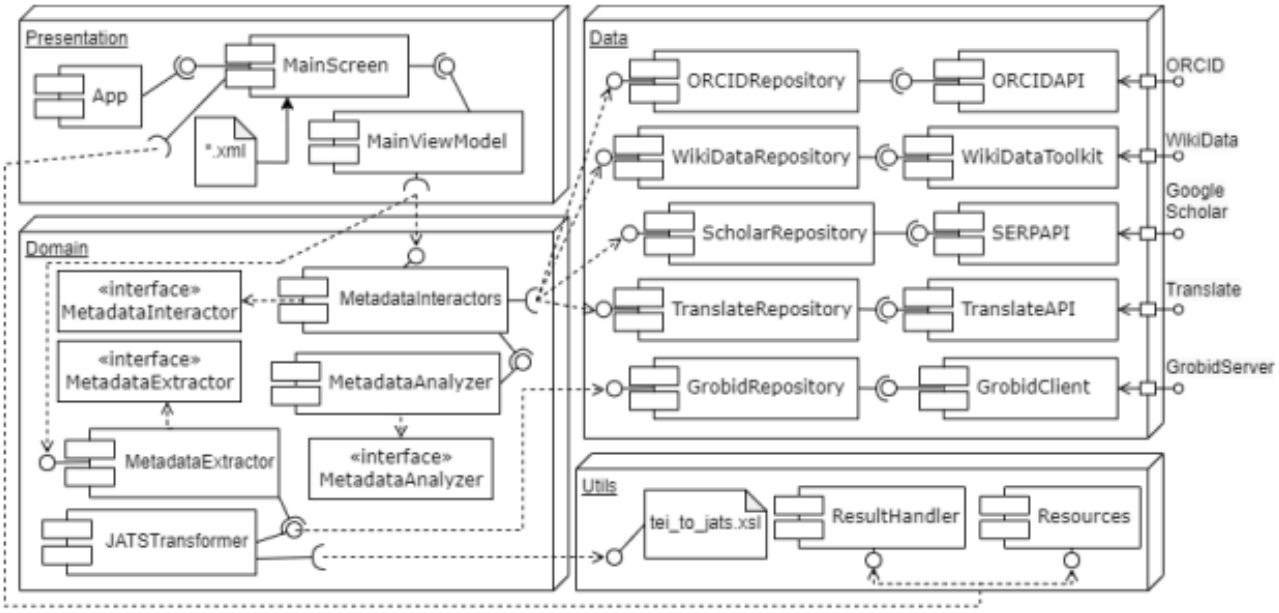


Рисунок 2.7 – Діаграма компонентів розробленої системи

Як показано на діаграмі компонентів проекту, було вирішено сформувати архітектуру проекту таким чином:

- шар даних (Data) є різноманітними сервісами взаємодії із зовнішніми

джерелами даних. Наприклад, для уточнення та доповнення метаданих із Google Scholar використовуються компоненти SERPAPI та ScholarRepository. У цих компонентах реалізовано всю логіку для взаємодії з семантичною мережею Google Scholar. За аналогічною архітектурою реалізовано клієнт-серверну взаємодію з іншими джерелами даних. Таке архітектурне рішення прибирає зв'язаність між модулями, що в свою чергу дозволяє надалі інтегрувати додаткові джерела даних,

для збільшення повноти і точності метаданих, що формуються;

- шар домену (Domain) містить у собі компоненти для видобування, уточнення та доповнення метаданих. MetadataExtractor відповідає за логіку видобування метаданих з наукових праць. MetadataInteractors, які відповідають за логіку обробки даних та використовують MetadataAnalyzer, який доповнює відсутні пункти вихідної моделі та уточнює ті частини даних, які не збігаються у різних джерелах, тобто можуть бути помилковими;
- на шарі представлення (Presentation) реалізований інтерфейс користувача, логіка оновлення стану та відображення результатів. Інтерфейс користувача спроектований згідно з правилами та рекомендаціями з [38, 39], також згідно з правилами Material Design [40] для кращої взаємодії користувача з програмним рішенням;
- допоміжний модуль (Utils) містить різні інструменти у вигляді функцій розширень, допоміжних компонентів для обробки станів та ресурсів проекту, куди входить і файл XSL .

2.5 Видобування та формування метаданих

Робота розробленого алгоритму видобування метаданих з наукової роботи за допомогою GROBID представлена на рис. 2.8 та складається з послідовності кроків.

Крок 1 Вказати шлях до файлу PDF, вибір параметрів видобування у вікні налаштувань і запуск процесу видобування.

Крок 2. Збереження файлу у системі подальшої передачі.

Крок 3. Створення та надсилання запиту до сервера GROBID для обробки

файлу згідно з налаштуваннями за допомогою спеціального http клієнта Ktor.

Крок 4. Перетворення отриманих даних з формату TEI в NISO JATS V1.0 за допомогою спеціального файлу XSL .

Крок 5. Передача отриманих метаданих у вихідний XML -файл для відображення користувачу.

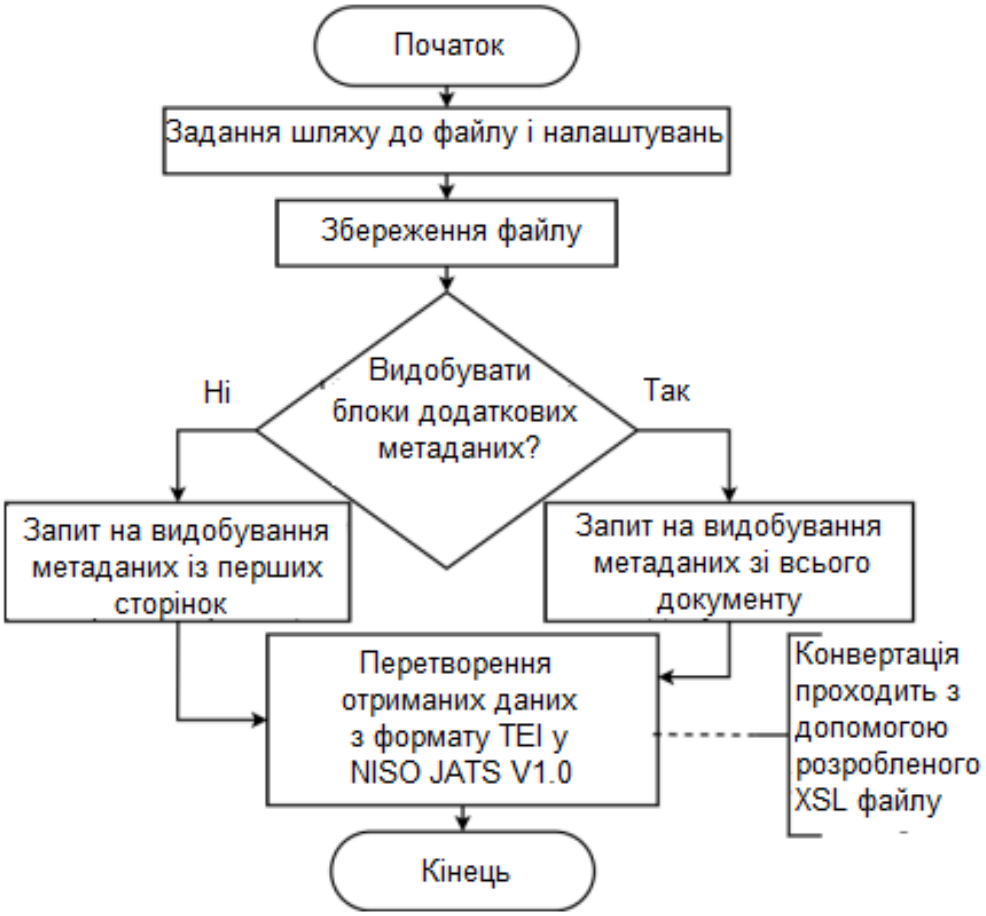


Рисунок 2.8 – Блок-схема алгоритму видобування метаданих з наукових документів

На рис. 2.9 представлений процес видобування метаданих з наукового документа у вигляді діаграми послідовності.

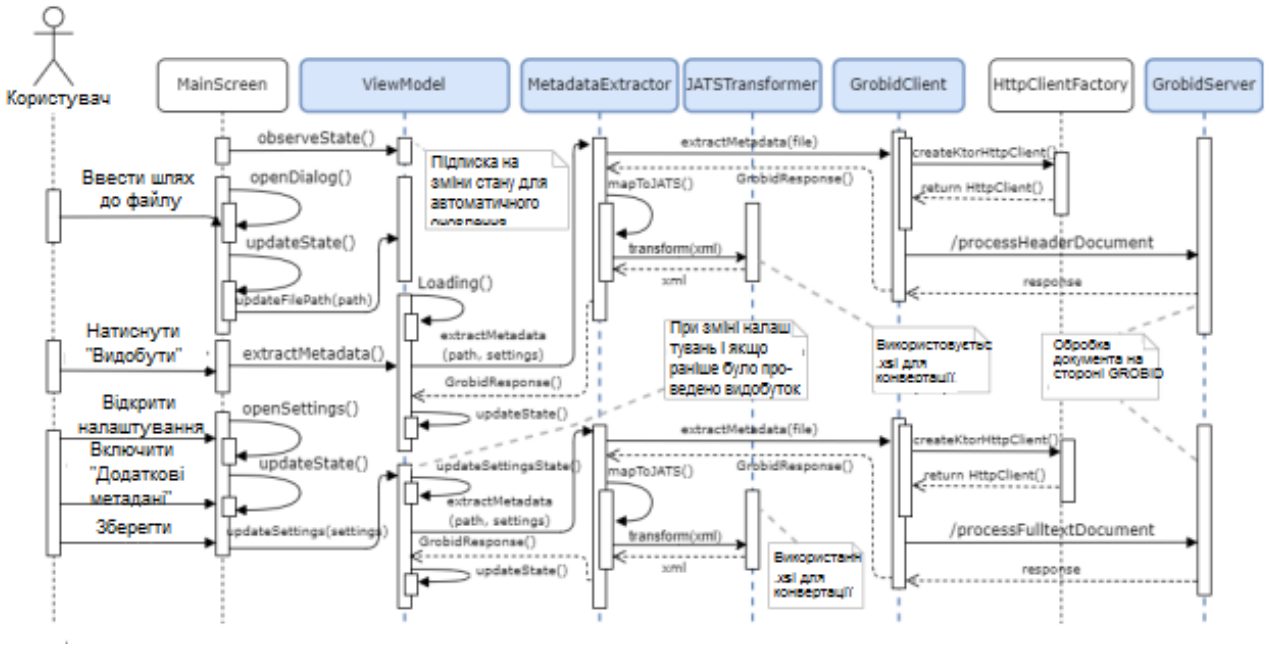


Рисунок 2.9 – Діаграма послідовності видобування метаданих з наукового документа з налаштуванням необхідності формування та включення до додаткових метаданих

Діаграма на рис. 2.9 описує роботу системи при наступному сценарії користувача:

- вказує шлях до файлу. Система зберігає шлях до файлу і підписується на зміни стану даних для автоматичного оновлення інтерфейсу користувача;
- запуск процесу видобування. За замовчуванням додаткові метадані не видобуваються з наукового документа, у зв'язку з чим надсилається відповідний запит GROBID для отримання базових і фундаментальних блоків. Відповіддю від GROBID є XML у форматі TEI, який перетворюється на формат NISO JATS V1.0 згідно з розробленим XSL файлом (рис. 2.10);
- увімкнення параметрів «Додаткових метаданих». При зміні налаштувань, а також, якщо на попередньому кроці вже було видобуто метадані, система автоматично запустить процес видобування метаданих. При увімкненому прапорці «Додаткові метадані» обробляється весь документ і можуть бути

отримані такі дані як: список літератури, дані про спонсорів, раніше не виявлені ORCID авторів, додаток до афіліацій тощо.

```
<article-meta>
    <title-group>
        <article-title>
            <xsl:value-of select="tei:fileDesc/tei:titleStmt/tei:title"/>
        </article-title>
    </title-group>

    <contrib-group>
        <xsl:for-each select="tei:fileDesc/tei:sourceDesc/tei:biblStruct/tei:analytic/tei:author">
            <xsl:choose>
                <xsl:when test="tei:persName/tei:surname != ''">
                    <contrib contrib-type="author">
                        <name>
                            <surname>
                                <xsl:value-of select="tei:persName/tei:surname"/>
                            </surname>
                            <given-names>
                                <xsl:for-each select="tei:persName/tei:forename">
                                    <xsl:value-of select="string(.)"/>
                                </xsl:for-each>
                            </given-names>
                        </name>
                    </contrib>
                </xsl:when>
```

Рисунок 2.10 – Фрагмент розробленого XSL файлу для перетворення XML-файлу у форматі TEI у NISO JATS V1.0.

2.6 Доповнення та уточнення метаданих

Розглянемо діаграму компонентів, що описує зв'язки всередині процесу доповнення метаданих (рис. 2.11). Важливою особливістю такого архітектурного рішення є незалежність компонентів для роботи з зовнішніми сервісами, що дозволить у подальшому оновити логіку взаємодії з тим чи іншим джерелом, наприклад, при появі офіційного API для отримання даних із джерела, або ж інтегрувати взаємодію з новою БД , не вносячи значних змін до інших шарів.

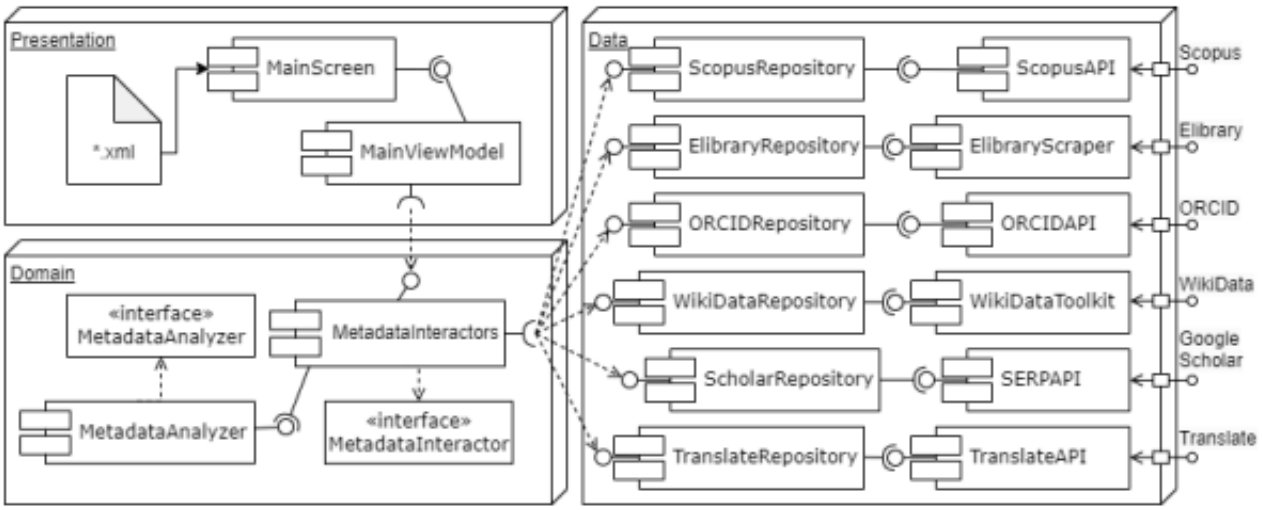


Рисунок 2.11 – Діаграма компонентів процесу доповнення метаданих із зазначенням додаткових модулів з метою показу можливих розширень системи, що розробляється

Загальну схему роботи процесу доповнення та уточнення блоків метаданих можна представити через виконання кроків.

Крок 1. Отримання даних із видобутої назви та списку авторів наукової статті в Google Scholar.

Крок 2. Отримання даних по видобутій назві та списку авторів наукової статті у Wikidata.

Крок 3. Отримання даних із видобутої назви в ORCID. У разі виявлення ORCID авторів під час видобування метаданих з документа або на попередніх кроках у Google Scholar та Wikidata, звернення до ORCID для уточнення та доповнення даних про авторів.

Крок 4. Переклад ключових слів та анотації англійською чи українською

мовами у разі потреби (за відсутності перекладу та при включенні прапорця в налаштуваннях користувачем).

Крок 5. Аналіз та збір блоків метаданих у єдиний файл XML згідно зі схемою NISO JATS.

2.7 Висновки до другого розділу

Було проведено огляд відкритих баз даних наукових публікацій (Scopus, Google Scholar, WikiData і ORCID) на предмет наявності API, вільного доступу та кількості робіт.

Запропонована модель системи, описано її функціональні можливості. Розроблена архітектура рішення, яка показана як діаграма компонентів, що описує зв'язки усередині розробки.

Наведено розроблені алгоритми видобування та формування метаданих із наукових публікацій. Запропоновано покроковий механізм доповнення та уточнення таких метаданих.

3 ПРАКТИЧНА ЧАСТИНА

3.1 Отримання даних із Google Scholar

Одним із джерел інформації для уточнення та доповнення метаданих, котрі видобуті із наукової статті, було обрано Google Scholar. Для взаємодії з цим сервісом використовується Google Scholar SERP API, який містить кілька конфігурованих запитів для отримання даних залежно від вимог.

Запишемо роботу алгоритму доповнення та уточнення метаданих з Google Scholar, представленого на рис. 3.1, у вигляді послідовності кроків.

Крок 1 Отримання даних із видобутої назви наукової статті.

Крок 2. Пошук даних по кожному автору відповідно до видобутого списку ПІБ та списку ПІБ та id профілів авторів, отриманого на попередньому кроці.

Крок 3. Отримання даних для цитування за унікальним ідентифікатором статті, отриманим на першому кроці.

Крок 4. Об'єднання метаданих у єдину модель. Фільтрування даних згідно з налаштуваннями користувача (увімкнення/виключення додаткових метаданих).

Крок 5. Передача даних для аналізу та включення отриманих результатів у вихідний XML- файл згідно з NISO JATS.

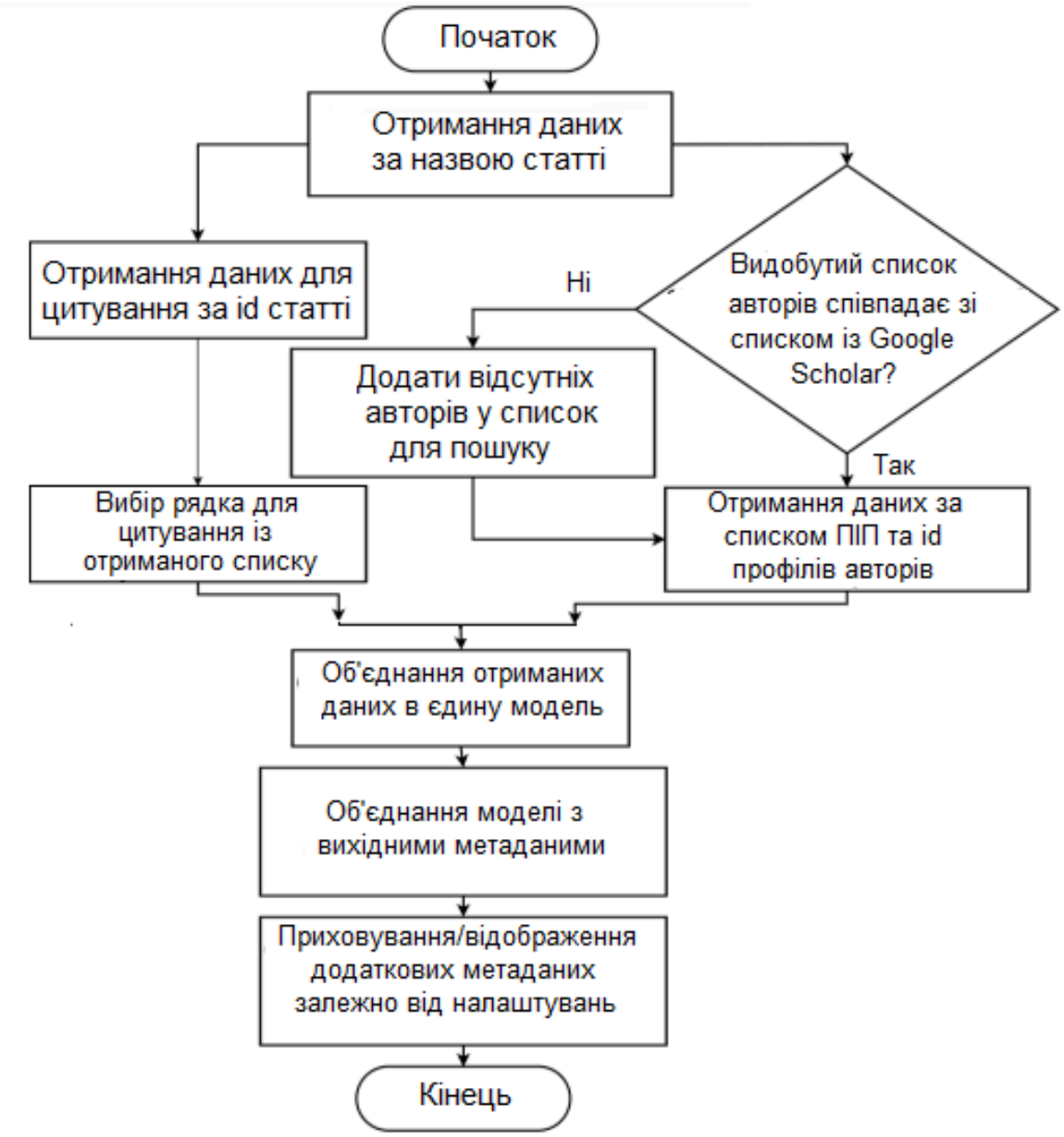


Рисунок 3.1 – Блок-схема алгоритму уточнення та доповнення метаданих з Google Scholar

Розглянемо загальні вимоги щодо запитів при взаємодії з SERP API. Програма звертається до сервера Google Scholar SERP API по мережевому протоколу HTTPS, виконуючи запити GET. Доменне ім'я «serpapi.com», шлях - «/search» Запит до API повинен містити два обов'язкові параметри:

- engine - ключ API, який позначає тип даних, що запитуються (профілі наукових діячів, цитування, статті і т.д.);

- api_key – приватний ключ для доступу до API, який необхідно згенерувати після реєстрації на сайті у консолі розробника.

Для процесу уточнення та доповнення метаданих використовується три підтипи запитів:

- одержання даних за назвою статті;
- отримання даних про автора за видобутим раніше ПІБ;
- одержання даних для цитування статті.

Для отримання даних з Google Scholar запит ПІБ автора використовується запит serpapi.com/search з наступними параметрами:

- engme=”google_scholar_profiles” - для пошуку профілю автора статті у Google Scholar;
- mauthors=”ПІБ” - видобуті ПІБ з наукового документа.

Нижче на рис. 3.2 наведено приклад фрагмента відповіді від Google Scholar SERP API у форматі JSON.

Згідно з цими даними можна отримати афіліації, фото та список основних ключових тематик робіт автора, який надалі може допомогти вирішити проблему відмінності повних тезок.

Для отримання даних з Google Scholar запит на назву наукової статті використовується запит serpapi.com/search з наступними параметрами:

- engine-'google_scholar” - для пошуку статті у Google Scholar;
- q -"Назва статті” - назва статті.

Для отримання додаткової інформації необхідно запитати дані для цитування цієї статті за значенням поля result_id – унікальний ідентифікатор

статті. Для отримання даних для цитування надсилається запит serpapi.com/search з наступними параметрами:

- engine-'google_scholar_cite” - для формування даних для цитування відповідно до загальноприйнятих стандартів;
- q=”result_id” – унікальний ідентифікатор статті.


```
{
    "profiles”: [
    {
    "name": "mykhaylo petryk",
    "author id": "xzok1v8AAAAJ",
    "affiliations": "Ternopil Ivan Puluj National Technical
    University",
    "email": " petrykmr@gmail.com ",
    "cited by": 1078,
    "interests": [
    {
    "title": "математичне моделювання"
    },
    {
    "title": "нанопористі структури"
    },
    {
    "title": "програмна інженерія"
    }
      ],
  "thumbnail":"https://scholar.googleusercontent.com/citations?
view_op=small_photo&user=xzok1v8AAAAJ&citpid=2"
    }
  ]
}
```


Рисунок 3.2 – Приклад фрагмента відповіді від Google Scholar SERP API при пошуку ПІБ автора "Mykhaylo Petryk".

Відповідно до цих даних можна отримати список авторів із зазначенням їх профілів у Google Scholar, для подальшого уточнення та доповнення інформації про авторів. Також розширити блок додаткових метаданих посиланням на публікацію, назвою журналу та видавничої компанії, роком публікації, окремим посиланням на завантаження статті та анотацією.

3.2 Отримання даних з WikiData

Як додаткове джерело даних для уточнення та доповнення метаданих, вилучених з наукових статей, була інтегрована БД Wikidata. Доступ до даної інформаційної системи здійснюється через SPARQL -запити за допомогою бібліотеки Wikidata Toolkit, яка спрощує реалізацію клієнт-серверної взаємодії з Wikidata. Нижче наведено алгоритм дій для доповнення та уточнення метаданих з використанням SPARQL -запитів до Wikidata, викладеного у вигляді послідовності операцій на рис. 3.3.

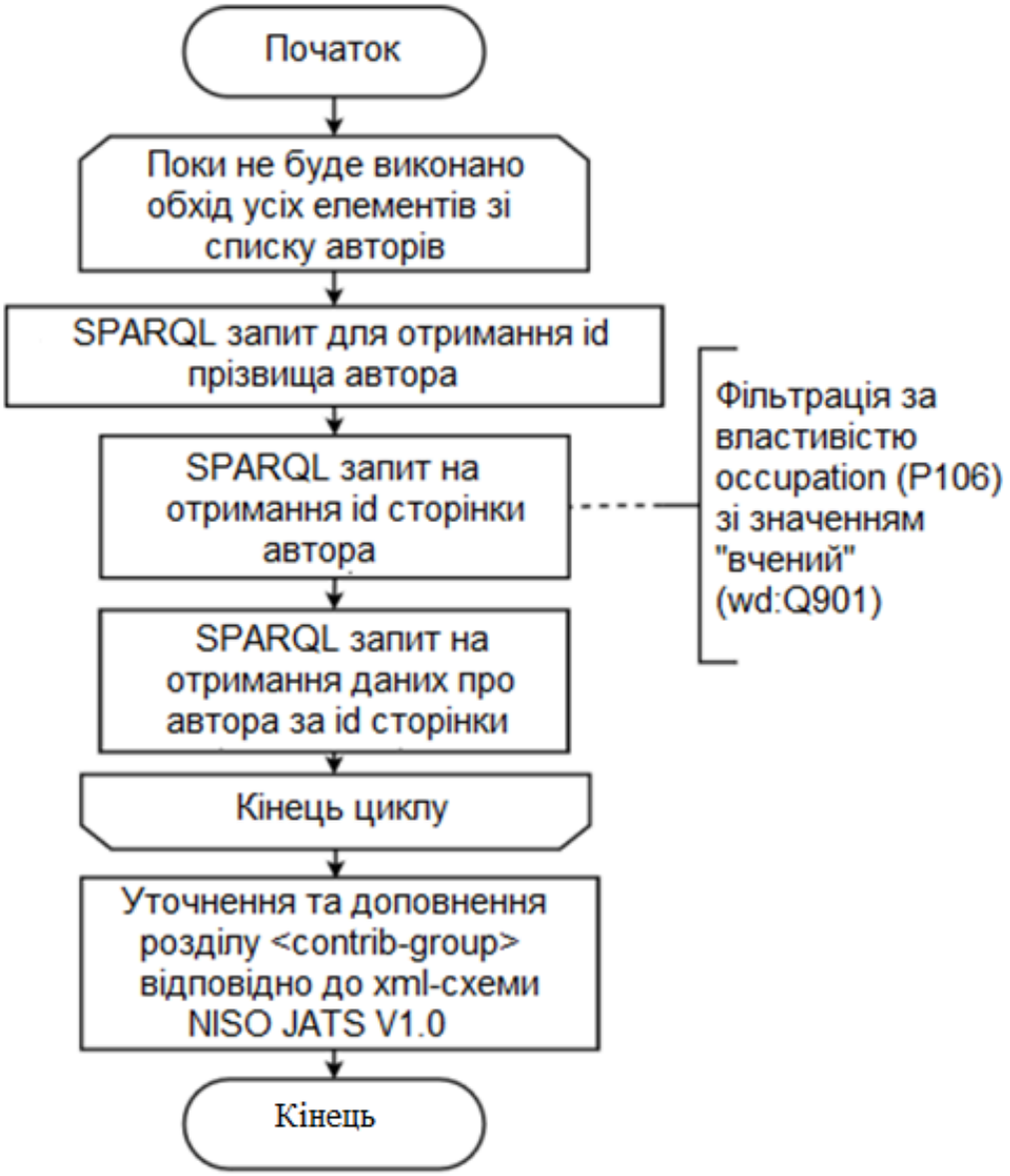


Рисунок 3.3 – Блок-схема алгоритму уточнення та доповнення метаданих з Wikidata

Крок 1. Надсилання запиту до Wikidata для отримання ID прізвища автора (рис. 3.4).

Крок 2. Надсилання запиту до Wikidata для отримання id сторінки автора у Wikidata за раніше отриманим id ПІБ автора (рис. 3.5).

Крок 3. Надсилання запиту до Wikidata для отримання даних про автора відповідно до раніше знайденого id сторінки (рис. 3.6).

Крок 4. Аналіз отриманих даних та доповнення метаданих про авторів на основі отриманої інформації.

Пошук відбувається за допомогою бібліотеки Wikidata Toolkit, яка дозволяє викликати Wikidata за допомогою SPARQL і служить заміною Ktor при клієнт-серверній взаємодії з Wikidata. Для збору інформації було виявлено властивості (Property), дані з яких використовувалися для уточнення та створення додаткових метаданих (табл. 3.1).

Таблиця 3.1 – Основні характеристики Wikidata, що застосовуються в алгоритмі уточнення та доповнення метаданих

Property	ID властивості	Приклад	Jats тег
Прізвище	P734	Petryk	<surname>
Ім'я	P735	Mykhailo	<given-name>
Ім'я рідною мовою	P1559	Михайло Романович Петрик	<string-name>
Установа в афіляції	P108	Ternopil Ivan Puluj National Technical University	<institution>
Країна в афіляції	P17	Україна	<country>
Вчений ступінь	P512	Doctor of Physics and Mathematics	<degree>

У табл. 3.1 перелічені елементи, які видобуваються за допомогою SPARQL запитів_до сервісу WikiData. Усі елементи пов'язані з відповідними JATS тегами та ідентифікаторами у семантичній мережі.

Нижче наведено деякі запити, які використовуються у розробленому алгоритмі. Стандартний запит пошуку у Wikidata додаткової інформації за автором статті представлений на рис. 3.4.

```
SELECT * WHERE {
  ?item rdfs:label "$name"@ua.
  ?item wdt:P31 wd:Q101352.
}
```

Рисунок 3.4 – Запит на отримання унікального ідентифікатора на прізвище автора, де змінна " name " набуває значення прізвища автора

У контексті формулювання SPARQL -запитів до БД Wikidata, властивість occupation (P106) відіграє ключову роль. Ця властивість дозволяє ефективно фільтрувати набори даних, виключаючи з результатів все, крім тих записів, котрі містять інформацію про суб'єктів, асоційованих з академічною та науковою сферами (рис. 3.5).

```
SELECT DISTINCT ?item ?itemLabel WHERE {
  SERVICE wikibase:label { bd:serviceParam wikibase:language "ua". }
  {
    SELECT DISTINCT ?item WHERE {
      ?item p:P734 ?statement0.
      ?statement0 (ps:P734/(wdt:P279*)) wd:$code.
      ?item p:P106 ?statement1.
      ?statement1 (ps:P106/(wdt:P279*)) wd:Q901.
  }
   LIMIT 100
  }
}
```

Рисунок 3.5 – Запит на отримання id сторінки автора, де замість змінної code проставляється унікальний ідентифікатор прізвища автора у Wikidata.

На рис. 3.5 фільтрування за властивістю occupation (P106) зі значенням ”вчений” (wd: Q901). Далі на рис. 3.6 представлено сформований запит згідно з отриманим унікальним ідентифікатором сторінки автора для пошуку метаданих, властивості яких описані в табл. 3.1. Також у запиті проставлено фільтрацію мови

- англійську та українську.

```
SELECT * WHERE {
 wd:$code wdt:P734 ?family_name_id.
     ?family_name_id rdfs:label
     ?family_name FILTER(LANG(?family_name) = 'ua' || LANG(?family_name) = 'en')
 wd:$code wdt:P735 ?given_name_id.
     ?given_name_id rdfs:label
     ?given_name FILTER(LANG(?given_name) = 'ua' || LANG(?given_name) = 'en')
 wd:$code wdt:P1559 ?name_in_native_language.
 wd:$code wdt:P106 ?occupation_id.
     ?occupation_id rdfs:label
     ?occupation FILTER(LANG(?occupation) = 'ua' || LANG(?occupation) = 'en')
 wd:$code wdt:P108 ?employer_id.
     ?employer_id rdfs:label
     ?employer FILTER(LANG(?employer) = 'ua' || LANG(?employer) = 'en')
 wd:$code wdt:P17 ?country_id.
     ?country_id rdfs:label
     ?country FILTER(LANG(?country) = 'ua' || LANG(?country) = 'en')
 wd:$code wdt:P512 ?academic_degree_id.
     ?academic_degree_id rdfs:label
     ?academic_degree FILTER(LANG(?academic_degree) = 'ua' || LANG(?academic_degree) = 'en')
}
```

Рисунок 3.6 – Запит на отримання даних про автора згідно з полями з табл. 3.1, де замість змінної code є унікальний ідентифікатор сторінки автора у Wikidata

Завершальним кроком алгоритму є інтеграція отриманих метаданих у підсумковий файл xml згідно зі схемою NISO JATS V1.0.

Таким чином, вказані в табл. 3.1 блоки метаданих можуть бути уточнені і доповнені за допомогою Wikidata.

3.3 Отримання даних із ORCID

У ході наукового дослідження для уточнення та доповнення метаданих, які були видобуті з наукових документів, було обрано БД ORCID. Обмін інформацією із цим джерелом здійснюється за допомогою відкритого API ORCID, що надає

системам підключення до реєстру ORCID. Сервіс ORCID суворо дотримується принципів конфіденційності та захисту даних [41]. Оскільки для доступу до інформації користувача через API необхідно отримати явну згоду від користувача, було вирішено використовувати частину сервісу для отримання загальнодоступної інформації про користувача через ORCID API, яке можна здійснити без його згоди. Далі наведено алгоритм роботи з доповнення та уточнення метаданих з використанням ORCID API, описаний у вигляді послідовності кроків (рис. 3.7).

Крок 1. Надсилання запиту до ORCID API для отримання ORCID авторів, згідно з раніше одержаною назвою статті.

Крок 2. Формування єдиної множини знайдених ORCID авторів для подальшого пошуку даних.

Крок 3. Надсилання запиту до ORCID API для отримання загальнодоступної інформації про авторів, згідно з раніше видобутим або знайденим ORCID.

Крок 4. Аналіз отриманих даних на предмет відповідності вихідним метаданим статті та доповнення метаданих на основі верифікованої інформації з ORCID.

Даний процес дозволяє підвищити точність і повноту метаданих, що сприяє поліпшенню якості формованих блоків метаданих.

Для взаємодії з ORCID API використовуються наступні запити REST :

- отримання ORCID авторів за назвою статті;

- отримання даних про автора по ORCID.

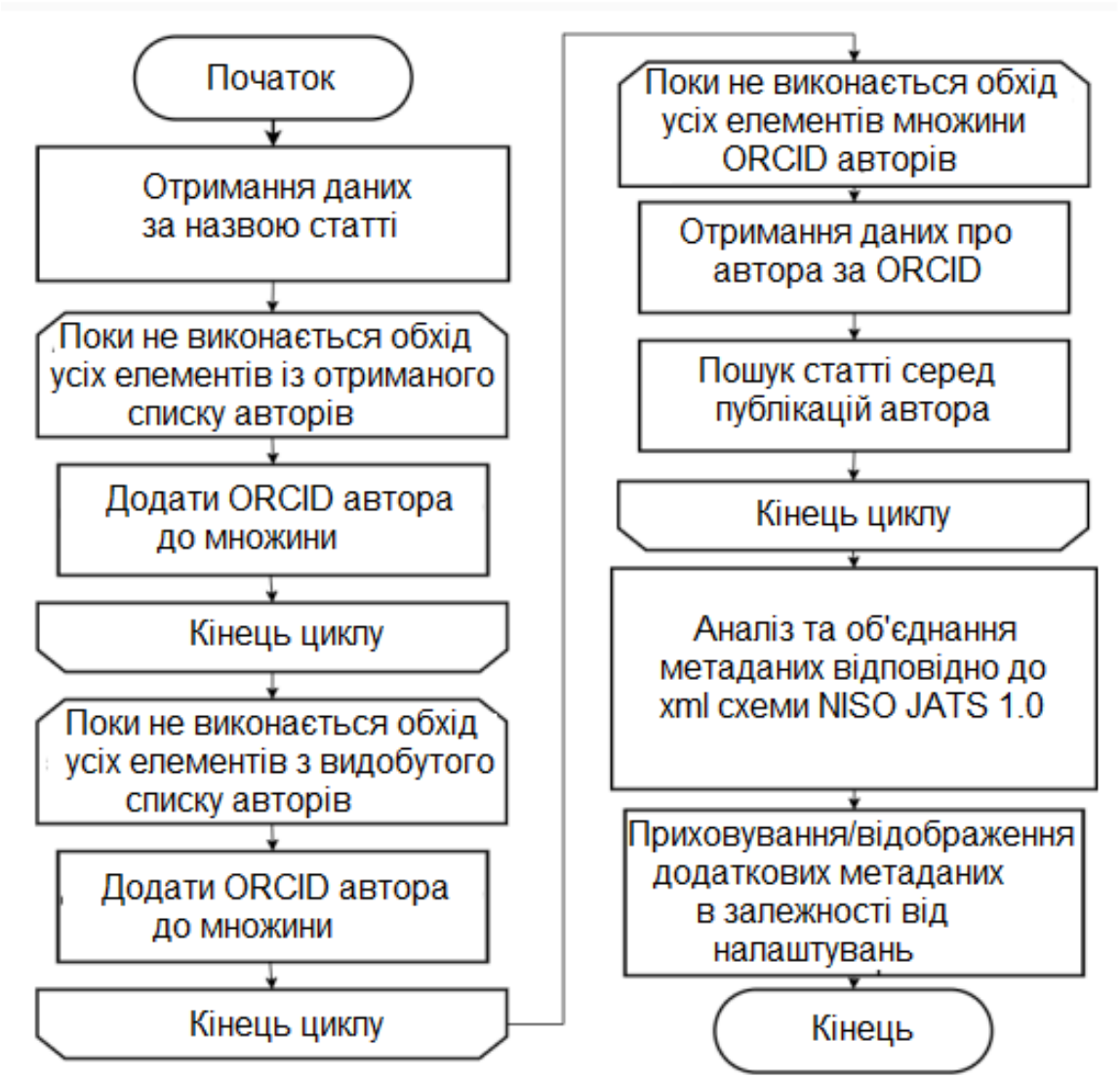


Рисунок 3.7 – Блок-схема алгоритму уточнення та доповнення метаданих з ORCID

Для отримання даних із ORCID за назвою статті формується наступний запит https://[HOST]/v3.0/search, де:

- [HOST] – pub.orcid.org. Для загальнодоступного API у реєстрі ORCID (тільки для загальнодоступної області /read-public) [39];
- параметр «q» зі значенням, що дорівнює назві статті. Назву статті необхідно вказувати в лапках для отримання точних авторів за цією статтею.

Для отримання даних з ORCID за ORCID автора формується наступний запит https://[HOST]/v3.0/ [ORCID]/record, де:

- [HOST] – pub.orcid.org. Для загальнодоступного API у реєстрі ORCID (тільки для загальнодоступної області /read-public) [42];
- [ORCID] - раніше отриманий ORCID автора.

На рис. 3.8 наведено приклад фрагмента відповіді від ORCID API у форматі JSON.


```
            "source-name": {
              "value": "mykhaylo petryk"
            }
          },
          "email": "petrykmr@gmail.com",
          "visibility": "public"
        }
      ]
    }
  },
  "activities-summary": {
    "educations": {
      "affiliation-group": [{
          "summaries": [
            {
              "education-summary": {
                "department-name": "Software Engineering",
                "role-title": "Doctor of Physics and
                               Mathematics",
                "start-date": {...},
                "end-date": {...},
                "organization": {
                  "name": "Ternopil Ivan Puluj National
                          Technical University",
                  "address": {
                    "city": "Ternopil",
                    "region": "Ternopil",
                    "country": "UA"
                  },
```


Рисунок 3.8 – Фрагмент відповіді від сервісу ORCID API при запиті даних ORCID автора «0000-0001-6612-7213»

Отримання даних про автора дозволить отримати не лише їх, а й дані про

поточну статтю, якщо вони присутні в ORCID. Внаслідок чого виконується пошук статті серед списку публікацій автора, отриманий на попередньому кроці разом із даними про автора. Подібний підхід дозволить знайти блок додаткових метаданих за їх наявності, таких як DOI статті, переклад заголовка статті, посилання на публікацію в журналі, дату публікації та назву журналу.

Згідно з результатами дослідження, завдяки взаємодії з ORCID можуть бути уточнені та доповнені афіліації, ПІБ, електронні пошти, ORCID авторів, а також додаткові метадані у вигляді посилань на профілі авторів в інших наукових сервісах, місця роботи та навчання авторів, список наукових статей, список ключових слів спрямованості наукових праць. Багато залежить від відкритості профілю авторів та заповненості інформації облікового запису на сайті ORCID. Робота з цим сервісом є цінною, тому що в деяких випадках подібна інформація дозволить не просто уточнити та доповнити метадані, а й розширити коло пошуку. Наприклад, отримати посилання на публікації та профілі Scopus для подальшого уточнення та доповнення інформації.

3.4 Модуль перекладу анотацій та ключових слів

Для перекладу анотації та ключових слів статті з української на англійську та навпаки було реалізовано модуль перекладу за допомогою сервісу Google Translate API. Розроблений алгоритм перекладу представлений як псевдокод на рис. 3.9 і складається з набору кроків.

Крок 1 Надсилання запиту з назвою статті Google Translate API для визначення мови тексту.

Крок 2 Надсилання запиту до Google Translate API з текстом анотації.

Крок 3 . Надсилання запиту до Google Translate API зі списком ключових слів.

Крок 4 . Інтеграція отриманих даних із позначенням мови у вихідний файл

XML відповідно до стандарту NISO JATS V1.0.

```
# Розбір XML -дерева та отримання назви статті
load article.title = input.xml
# Розбір XML -дерева та отримання тексту анотації
load abstract = input.xml
# Розбір XML -дерева та отримання списку ключових слів
load kwds = input.xml
# Визначення мови за назвою статті
article_lang=google_api.detect(article_title)
# Переклад англійською, якщо текст написаний українською
if article.lang =='ua':
    # Конфігурація та надсилання запиту перекладу англійською
    if abstract !="":
        abstract_translated=translate_en(abstract)
    for kwd in kwds:
        if kwd !=  "":
            kwds_translated=translate_en(kwds)
# Переклад українською, якщо текст написаний англійською
if article.lang =='en':
    # Конфігурація та відправка запиту перекладу українською
    if abstract !=  "":
        abstract_translated=translate_ua(abstract)
    for kwd in kwds:
        if kwd !=  "":
            kwds_translated=translate_ua(kwds)
# Інтеграція результатів з мовною міткою в xml файл
add.translation(article.lang, abstract.translated, kwd.translated)
```

Рисунок 3.9 – Псевдокод алгоритму перекладу анотації та списку ключових слів за їх наявності з української на англійську та навпаки залежно від мови вихідної статті

Для взаємодії з Google Translate API використовуються наступні запити REST:

- визначення мови, якою написана стаття;

- переклад тексту (анотації та ключових слів).

Загальні вимоги до запитів щодо взаємодії з Google Translate API. Програма звертається до сервера Google Translate API по мережному протоколу HTTPS, виконуючи запити POST.

Для визначення мови використовується запит, поданий в табл. 3.2.

Таблиця 3.2 – Опис запиту визначення мови відповідно до документації Google Translate API [43]

URL	/detect	додатковий шлях до url для запиту визначення мови
Тіло запиту	text	текст, на основі якого визначається мова
	languageCodeHints []	список ймовірних мов, котрим треба надати пріоритет

Для перекладу анотації та ключових слів використовується запит, поданий у табл. 3.3.

Таблиця 3.3 – Опис запит для перекладу анотації та ключових слів згідно з документацією Google Translate API [43]

URL	/translate	додатковий шлях до url для запиту на переклад тексту
Тіло запиту	sourceLanguageCode	мова, якою написано вихідний текст
	targetLanguageCode	мова, якою перекладається текст
	texts[]	масив рядків для перекладу

3.5 Висновок до третього розділу

У цьому розділі наведені програмні особливості отримання наукових даних із різних наукометричних баз - Google Scholar, WikiData, ORCID. Для кожного випадку представлено покрокові алгоритми доповнення та уточнення метаданих. Додатковою інформацією може бути афіліація автора, його електронна пошта, його профілі у інших базах наукових публікацій.

Розроблено для впровадження у систему модуль, який здійснюватиме переклад анотацій та ключових слів наукових публікацій.

4 ОХОРОНА ПРАЦІ ТА БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ

4.1 Охорона праці

Метою кваліфікаційної роботи магістра є розробка ТОС, що пропонує ОР на базі концепції ТО, яка повинна контролювати, налаштовувати та надавати туманні ОР в автоматичному режимі. Оскільки, проведення робіт з розробки та використання ТОС передбачає застосування комп’ютерної техніки, зокрема ПК та периферійних пристроїв, то обов’язковим є дотримання вимог з охорони праці і техніки безпеки.

Для ефективної і безпечної роботи колективу працівників з розробки ПЗ комп’ютерних систем, в тому числі і фахівців з підвищення ефективності контролю доступу в приміщення, необхідно організувати безпечні умови праці. При цьому керівник організації несе безпосередню відповідальність за порушення нормативно-правових актів з охорони праці [48]. Окрім цього, на робочих місцях працівників необхідно забезпечити дотримання вимог, затверджених Наказом Мінсоцполітики від 14.02.2018 за № 207 «Про затвердження Вимог щодо безпеки та захисту здоров’я працівників під час роботи з екранними пристроями». Згідно Вимог приміщення, де розміщені робочі місця операторів, крім приміщень, у яких розміщені робочі місця операторів великих ЕОМ загального призначення (сервер), мають бути оснащені системою автоматичної пожежної сигналізації відповідно до цих вимог;

– переліку однотипних за призначенням об'єктів, які підлягають обладнанню автоматичними установками пожежогасіння та пожежної сигналізації, затвердженого наказом Міністерства України з питань надзвичайних ситуацій та у справах захисту населення від наслідків Чорнобильської катастрофи від 22.08.2005 N 161, зареєстрованого в Міністерстві юстиції України 05.09.2005 за N 990/11270 (НАПБ Б.06.004-2005);

– Державних будівельних норм "Інженерне обладнання будинків і споруд. Пожежна автоматика будинків і споруд", затверджених наказом Держбуду України

від 28.10.98 N 247 (далі - ДБН В.2.5-56:2014, з димовими пожежними сповіщувачами та переносними вуглекислотними вогнегасниками.

В інших приміщеннях допускається встановлювати теплові пожежні сповіщувачі. Приміщення, де розміщені робочі місця операторів, мають бути оснащені вогнегасниками, кількість яких визначається згідно з вимогами ДСТУ 4297:2004 «Пожежна техніка. Технічне обслуговування вогнегасників». Загальні технічні вимоги і з урахуванням граничнодопустимих концентрацій вогнегасної рідини відповідно до вимог НАПБ А.01.001-2014. Приміщення, в яких розміщуються робочі місця операторів сервера загального призначення, обладнуються системою автоматичної пожежної сигналізації та засобами пожежогасіння відповідно до вимог ДБН В.2.5-56:2014, НАПБ А.01.001-2014 і вимог нормативно-технічної та експлуатаційної документації виробника. Проходи до засобів пожежогасіння мають бути вільними.

Лінія електромережі для живлення комп’ютера та периферійних пристроїв повинні бути виконаними як окрема групова трипровідна мережа шляхом прокладання фазового, нульового робочого та нульового захисного провідників. Нульовий захисний провідник використовується для заземлення (занулення) електроприймачів. Не допускається використовувати нульовий робочий провідник як нульовий захисний провідник. Нульовий захисний провідник прокладається від стійки групового розподільного щита, розподільного пункту до розеток електроживлення. Не допускається підключати на щиті до одного контактного затискача нульовий робочий та нульовий захисний провідники.

Площа перерізу нульового робочого та нульового захисного провідника в груповій трипровідній мережі має бути не менше площі перерізу фазового провідника. Усі провідники мають відповідати номінальним параметрам мережі та навантаження, умовам навколишнього середовища, умовам розподілу провідників, температурному режиму та типам апаратури захисту, вимогам НПАОП 40.1-1.01-97.

У приміщенні, де одночасно експлуатуються понад п’ять комп’ютерів, на помітному, доступному місці встановлюється аварійний резервний вимикач, який

може повністю вимкнути електричне живлення приміщення, крім освітлення. Комп’ютери повинні підключатися до електромережі тільки за допомогою справних штепсельних з'єднань і електророзеток заводського виготовлення.

У штепсельних з'єднаннях та електророзетках, крім контактів фазового та нульового робочого провідників, мають бути спеціальні контакти для підключення нульового захисного провідника. Їхня конструкція має бути такою, щоб приєднання нульового захисного провідника відбувалося раніше, ніж приєднання фазового та нульового робочого провідників. Порядок роз’єднання при відключенні має бути зворотним. Не допускається підключати комп’ютери до звичайної двопровідної електромережі, в тому числі – з використанням перехідних пристроїв. Електромережі штепсельних з'єднань та електророзеток для живлення комп’ютерної техніки повинні бути виконаними за магістральною схемою, по 3-6 з'єднань або електророзеток в одному колі. Штепсельні з'єднання та електророзетки для напруги 12 В та 42 В за своєю конструкцією мають відрізнятися від штепсельних з'єднань для напруги 127 В та 220 В. Штепсельні з'єднання та електророзетки, розраховані на напругу 12 В та 42 В, мають візуально (за кольором) відрізнятися від кольору штепсельних з'єднань, розрахованих на напругу 127 В та 220 В.

При підвищенні ефективності контролю доступу в приміщення, де для забезпечення безпеки мешканців, співробітників і збереження майна використовуються ДС, важливим, з точки зору охорони праці, є забезпечення достатньої величини природного та штучного освітлення, які визначені у НПАОП 0.00-7.15-18. Організація робочого місця фахівця із дослідження методів та програмно-апаратних засобів оптимізаційних процесів на основі ГА повинна забезпечувати відповідність усіх елементів робочого місця та їх розташування ергономічним вимогам ДСТУ 8604:2015 «Дизайн і ергономіка. Робоче місце для виконання робіт у положенні сидячи. Загальні ергономічні вимоги». Відстань від екрана до ока фахівців, які працюють за комп’ютером визначається згідно з вимогами ДСанПіН 3.3.2.007-98.

Розміщення принтера або іншого пристрою введення-виведення інформації

на робочому місці має забезпечувати добру видимість екрана комп’ютера, зручність ручного керування пристроєм введення-виведення інформації в зоні досяжності моторного поля згідно з вимогами ДСанПіН 3.3.2.007-98.

Таким чином, у результаті аналізу вимог щодо охорони праці користувачів комп’ютерів, визначено особливості організації робочих місць, вимог з електробезпеки, природного та штучного освітлення для ефективної і безпечної роботи фахівців з розробка ТОС, що пропонує ОР на базі концепції ТО, яка повинна контролювати, налаштовувати та надавати туманні ОР в автоматичному режимі.

4.2. Планування та порядок проведення евакуації населення з районів наслідків впливу НС техногенного та природного характеру

В умовах неповного забезпечення захисними спорудами в містах та інших населених пунктах, що мають об'єкти підвищеної небезпеки, основним засобом захисту населення є евакуація і розміщення його у зонах, які є безпечними для проживання людей [49].

Евакуації підлягає населення, яке проживає в населених пунктах, що знаходяться у зонах можливого катастрофічного затоплення, можливого небезпечного радіоактивного забруднення, хімічного ураження, в районах виникнення стихійного лиха, аварій і катастроф (якщо виникає безпосередня загроза життю та здоров'ю людей). Залежно від обставин, які склалися на час надзвичайної ситуації, може бути проведено загальну або часткову евакуацію населення тимчасового або безповоротного характеру. Загальна евакуація проводиться за рішенням Кабінету Міністрів України для всіх категорій населення і планується на випадок:

– можливого небезпечного радіоактивного забруднення територій навколо атомних електростанцій (якщо виникає безпосередня загроза життю та здоров'ю людей, які проживають в зоні ураження);

– виникнення загрози катастрофічного затоплення місцевості з чотиригодинним добіганням проривної хвилі.

Часткова евакуація проводиться за рішенням Кабінету Міністрів України у разі загрози або виникнення надзвичайної ситуації техногенного та природного характеру.

Під час проведення часткової евакуації завчасно вивозиться не зайняте у сферах виробництва та обслуговування населення: діти, учні навчальних закладів, вихованці дитячих будинків, разом з викладачами та вихователями, студенти, пенсіонери та інваліди, які утримуються у будинках для осіб похилого віку, разом з обслуговуючим персоналом і членами їх сімей [49].

У сфері захисту населення і територій від надзвичайних ситуацій техногенного та природного характеру евакуація населення планується на випадок:

– аварії на атомній електростанції з можливим забрудненням території; усіх видів аварій з викидом сильнодіючих отруйних речовин; загрози катастрофічного забруднення місцевості :

– лісових і торф'яних пожеж, землетрусів, зсувів, інших геофізичних і гідрометеорологічних явищ з тяжкими наслідкам, що загрожують населеним пунктам.

Загальна евакуація проводиться шляхом вивезення основної частини населення з міст і небезпечних районів усіма видами наявних транспортних засобів на відповідній адміністративній території та виведення найбільш витривалої його частини пішки. Часткова евакуація проводиться з використанням транспортних засобів, що експлуатуються за діючим графіком. На органи виконавчої влади, органи місцевого самоврядування та керівників об'єктів, які проводять евакуацію населення, покладається:

– планування і проведення евакуації працівників та членів їх сімей;

– подання до відповідних транспортних органів розрахунків потреби у транспортних засобах для вивезення працівників і членів їх сімей до безпечних районів;

– контроль за плануванням, підготовкою і проведенням евакуаційних заходів підвідомчими об'єктами;

– визначення та підготовка безпечного району для розміщення евакуйованих працівників і членів їх сімей.

Інші заходи та порядок проведення евакуації викладено у постанові Кабінету Міністрів від 26 жовтня 2001р. № 1432 про затвердження Положення про порядок проведення евакуації населення у разі загрози або виникнення надзвичайних ситуацій техногенного та природного характеру.

У плані евакуації, складовою частиною якого є карта (схема), зазначаються:

– висновки з оцінки обстановки у разі виникнення надзвичайної ситуації;

– порядок оповіщення населення про початок евакуації;

– кількість населення, яке підлягає евакуації, за віковими категоріями;

– терміни проведення евакуації;

– склад евакуаційних органів і терміни приведення їх у готовність;

– кількість населення, яке вивозиться різними видами транспортних засобів окремо і виводиться пішки;

– розподілення об'єктів за збірними евакуаційними пунктами, пунктами посадки, районами (пунктами) розміщення та евакуаційними напрямками; маршрути евакуації;

– райони (пункти) розміщення евакуйованого населення; пункти посадки на транспортні засоби, пункти висадки у безпечному районі, порядок доставки населення з пунктів висадки до районів (пунктів) розміщення;

– заходи щодо організації приймання, розміщення, захисту та життєзабезпечення евакуйованого населення у безпечному районі;

– порядок організації управління і зв'язку.

План приймання і розміщення евакуйованого населення включає також розділ з транспортного забезпечення евакуації, в якому зазначається [49]:

– кількість транспортних засобів кожного виду і термін їх подачі до пунктів посадки;

– кількість населення, яке підлягає евакуації;

– терміни відправлення евакуйованого населення у безпечні райони;

– терміни прибуття евакуйованого населення до пунктів посадки;

– маршрути руху транспортних засобів;

– кількість рейсів.

На всіх громадян, які підлягають евакуації, завчасно складаються списки за об'єктами і житлово-експлуатаційними організаціями у трьох примірниках, один з яких залишається на об'єкті або в житлово-експлуатаційній організації, другий (у разі одержання рішення про проведення евакуації) після уточнення списків надсилається на збірний евакуаційний пункт, третій - до евакуаційної комісії району (пункту) розміщення.

З отриманням рішення (сигналу) про проведення евакуації евакуаційні комісії уточнюють завдання керівникам об'єктів щодо проведення евакуаційних заходів, контролюють стан оповіщення населення, його збору, формування колон (через начальників маршрутів), забезпечують переміщення їх до пунктів евакуації, а також разом з транспортними службами - готовність транспортних засобів до перевезень, уточнюють порядок їх використання, підтримують постійний зв'язок з начальниками маршрутів та з органами виконавчої влади безпечних районів, інформують їх про хід евакуації.

У райони розміщення евакуаційних органів та населення, яке підлягає евакуації, направляються представники евакуаційних комісій для вирішення питань приймання, розміщення і життєзабезпечення евакуйованого населення.

Керівник органу виконавчої влади і евакуаційна комісія безпечного району, організовують підготовку пунктів висадки, розгортають приймальний евакуаційний пункт, уточнюють кількість прибулих і порядок подачі транспортних засобів для їх вивезення з пунктів висадки, а також з проміжних

пунктів евакуації до пунктів розміщення, контролюють роботу керівників об'єктів безпечних районів з прийому і розміщення евакуйованого населення.

У разі оголошення евакуації громадяни самостійно на міських транспортних засобах, які у цей період працюють цілодобово, прибувають на збірні евакуаційні пункти. Працівники цих пунктів розподіляють громадян, які підлягають евакуації, за транспортними засобами, інструктують їх і забезпечують посадку на транспортні засоби.

Евакуйовані громадяни повинні мати при собі паспорт, військовий квиток, документ про освіту, трудову книжку або пенсійне посвідчення, свідоцтво про народження, гроші і цінності, продукти харчування і воду на 3 доби, постільну білизну, необхідний одяг і взуття загальною вагою не більш як 50 кілограмів на кожного члена сім'ї. Дітям дошкільного віку вкладається у кишеню або пришивається до одягу записка, де зазначається прізвище, ім'я та по батькові, домашня адреса, а також ім'я та по батькові матері і батька.

4.3. Висновки до розділу

В цьому розділі проаналізовано важливі питання охорони праці та безпеки в надзвичайних ситуаціях, висвітлено питання планування та порядку проведення евакуації населення з районів наслідків впливу НС техногенного та природного характеру.

ВИСНОВОК

Результатом виконання кваліфікаційної роботи є система, яка видобуває, уточнює та доповнює метадані наукових документів на основі запитів до семантичних мереж у вигляді десктопної програми для ОС Windows. Реалізоване рішення забезпечує автоматизацію процесу видобування, уточнення та доповнення метаданих наукових публікацій з метою подальшого формування електронної колекції наукових документів відповідно до прийнятих міжнародних стандартів.

Для досягнення мети було виконано такі завдання:

- досліджено існуючі рішення вилучення метаданих із наукових документів;
- проаналізовано доступні БД для уточнення та доповнення метаданих;
- проаналізовано сервіси машинного перекладу ключових слів та анотацій;
- розроблено систему запитів для пошуку в семантичній мережі інформації про статтю;
- реалізовано метод доповнення метаданих наукових документів;
- протестовано метод із використанням колекції наукових документів;
- розроблено програмний продукт у вигляді десктопної програми.

Таким чином, розроблене програмне рішення забезпечить зниження трудомісткості процесів видобування, уточнення та доповнення метаданих та зменшить ймовірність виникнення помилок за рахунок верифікації інформації у кількох відкритих БД наукових публікацій.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Scimago Journal & Country Rank. URL: https://www.scimagojr.com/countryrank.php. (дата звертання: 15.11.2024)

2. Що таке метадані статті? Як вони впливають на її індексацію? URL: https://nim.media/articles/shcho-take-metadani-statti-yak-voni-vplivayut-na-yiyi-indeksatsiyu/ (дата звертання 20.11.2024).

3. Google Scholar API. URL: https://serpapi.com/google-scholar-api. (дата звертання: 15.05.2024)

4. Wikidata: REST API . URL: https://www.wikidata.org/wiki/Wikidata:REST_API. (дата звертання: 15.11.2024)

5. ORCID. Public API .URL: https://info.orcid.org/documentation/features/public-api. (дата звертання: 15.11.2024)

6. Cloud Translation API. Translation . URL: https://cloud.google.com/translate/ docs/reference/rest (дата звертання 04.12.2024)

7. Journal Article Tag Suite. URL: https://jats.nlm.nih.gov/1.0 (дата звертання: 22.11.2024)

8. Introduction to Algorithms, Fourth Edition. / Cormen, Thomas H., Leiserson, Charles E., Rivest, Ronald L., Stein, Clifford. MIT Press, 2022.

9. Метадані наукової статті – наріжний камінь роботи, що будуть цитувати. URL: https://nim.media/articles/nazva-anotatsiya-klyuchovi-slova-narizhny-kamin-statti-shcho-budut-tsituvati/ (дата звертання 20.11.2024)

10. Як якісно оформити метадані наукового дослідження. URL: https://dntb.gov.ua/wp-content/uploads/2023/11/08.11.2023.pdf/ (дата звертання 20.11.2024).

11. Інформаційна технологія формування метаданих для систем автоматизованого документообігу : монографія / І. А. Кравчук, О. В. Бісікало. Вінниця : ВНТУ, 2016. 164 с.

12. Методи та засоби інженерії даних та знань: навч. посіб. для студ. вищ. навч. закл. / В. В.Литвин: Л. Магнолія 2006, 2012. 248 с

13. Світова цифрова бібліотека (WDL). URL: https://ube.nlu.org.ua/article/Світова_цифрова_бібліотека(WDL) (дата звертання: 23.11.2024)

14. Peter J. Olver What's happening with the World Digital Mathematics Library? URL: http://www-users.math.umn.edu/~olver/t_/wdmlb.pdf. (дата звертання: 23.11.2024)

15. World Digital Mathematics Library (WDML). URL: https://www.mathunion.org/ceic/library/world-digital-mathematics-library-w dml. (дата звертання: 25.11.2024)

16. Krochmalski J. IntelliJ IDEA Essentials. Packt Publishing Ltd, 2014.

17. Kotlin programming language. URL: https://kotlinlang.org/. (дата звертання: 25.11.2024)

18. Davis A. L., Davis A. L. Gradle. Learning Groovy 3: Java-Based Dynamic Scripting. 2019. pp. 105-114.

19. Compose Multiplatform.URL: https://www.jetbrains.com/lp/compose-multiplatform/. (дата звертання: 25.11.2024)

20. Romary L., Lopez P. Grobid-information extraction from scientific publications // ERCIM News. 2015. V. 100.

21. Elizarov R. et al. Kotlin coroutines: design and implementation // Proceedings of the 2021 ACM SIGPLAN International Symposium on New Ideas, New Paradigms, and Reflections on Programming and Software. 2021. pp. 68-84.

22. Glantz I., Hurtig H. Express.js and Ktor web server performance: A comparative study. 2022.

23. Journal Archiving and Interchange Tag Library. NISO JATS version 1.0. / National Center for Biotechnology Information (NCBI). National Library of Medicine (NLM). 2012. URL https://jats.nlm.nih.gov/archiving/tag-library/1.0/. (дата звертання: 27.11.2024).

24. EuDML metadata schema specification (v2.0–final). URL: https://initiative.eudml.org/eudml-metadata-schema-specification-v20-final. (дата звертання: 27.11.2024)

25. The EuDML metadata schema. Revision: 1.6 as of 15th December 2010. / Jost M., Bouche T., Goutorbe C., Jorda J.P. URL: http://www.mathdoc.fr/ publis/d3.2-v1.6.pdf.

26. Вилучення метаданих. URL: https:// https://hackyourmom.com/kibervijna/vyluchennya-metadanyh/ (дата звертання 20.11.2024)

27. Lopez, Patrice. GROBID: Combining Automatic Bibliographic Data Recognition and Term Extraction for Scholarship Publications. // Research and Advanced Technology for Digital Libraries, 13th European Conference, ECDL 2009, Corfu, Greece. 2009. p. 473-474.

28. Meuschke N. et al. A Benchmark of PDF Information Extraction Tools Using a Multi-task and Multi-domain Evaluation Framework for Academic Documents // Information for a Better World: Normality, Virtuality, Physicality, Inclusivity: 18th International Conference, iConference 2023, Virtual Event, March 13–17, 2023, Proceedings, Part II. Cham: Springer Nature Switzerland, 2023. p. 383-405.

29. Rodionov K. Testing stock models of machine translation from Google and DeepL in working with technical texts. 2022. №24 (103). С. 26-37.

30. Elsevier Developer Portal Interactive Scopus APIs . URL: https://dev.elsevier.com/scopus.html#!/Affiliation_Search/AffiliationSearch. (дата звертання: 28.11.2024)/

31. Stefanyshyn, I. , Pastukh, O., Stefanyshyn, V. , Baran, I. , Boyko, I. Robustness of AI algorithms for neurocomputer interfaces based on software and hardware technologies CEUR Workshop Proceedings, 2024, 3742, pp. 137–149.

32. 32.ORCID API Tutorials. URL: https://info.orcid.org/documentation/ api-tutorials/ (дата звертання: 30.11.2024)

33. Gusenbauer M. Google Scholar to overshadow them all? Comparing the sizes of 12 academic search engines and bibliographic databases // Scientometrics. 2019. Т. 118. №. 1. p. 177-214.

34. Wikidata. .URL: https://www.wikidata.org/wiki/Wikidata:Main_Page (дата звертання: 30.11.2024)

35. Сучков В.В. Сервіси для машинного перекладу ключових слів та анотації наукових текстів // Інформаційні моделі, системи та технології: Праці XII наук.-техн. конф. Тернопіль, 2024. С. 96.

36. Petryk, M. , Mykhalyk, D. , Petryk, M. , Boyko, I. , Mudryk, I. Modeling of adsorption and desorption of hydrocarbons in nanoporous catalyst zeolite using nonlinear langmuir's isotherm. CEUR Workshop Proceedings, 2018, 2300, pp. 42–45.

37. Lano K., Tehrani S. Y. Introduction to Software Architecture: Innovative Design using Clean Architecture and Model-Driven Engineering. // Springer Nature, 2023.

38. Cooper A. et al. About face: the essentials of interaction design. // John Wiley & Sons, 2014.

39. Bodker S. Through the interface: A human activity approach to user interface design. // CRC Press, 2021.

40. Material Design.URL: https://material.io. (дата звертання 02.12.2024)

41. ORCID. Privacy policy .URL: https://info.orcid.org/privacy-policy. (дата звертання 03.12.2024)

42. ORCID API v3.0 Guide. URL: https://github.com/ORCID/orcid-model/blob/master/src/main/resources/record_3.0/README.md. (дата звертання 03.12.2024)

43. How to get a Google Translate API key in 5 minutes? URL: https://dev-opencart.com/nastrojk-opt-m-zac-ja/kak-poluch-t-klyuch-ot-google-translate-api-za-5-m-nut. (дата звертання 03.12.2024)

44. Boyko, I. , Petryk, M. , Mudryk, I. , Stoianov, Y., Tsupryk, H. Mathematical Model of the Capacitor Based on Zeolite Material. Proceedings -

International Conference on Advanced Computer Information Technologies, ACIT, 2021, pp. 45–48.

45. Petryk, O. , Boyko, I. , Stoianov, Y. , Balaban, S. , Nestor, J. Mathematical modeling of diffusion transfer for charged particles in the layered composite medium CEUR Workshop Proceedings, 2022, 3309, pp. 436–446

46. Stefanyshyn, I. , Pastukh, O., Stefanyshyn, V. , Baran, I. , Boyko, I. Robustness of AI algorithms for neurocomputer interfaces based on software and hardware technologies CEUR Workshop Proceedings, 2024, 3742, pp. 137–149.

47. Stefanyshyn, V. , Stefanyshyn, I. , Pastukh, O. , Yatsyshyn, V. , Yakymenko. Accuracy of software and hardware of computer systems for human-machine interaction, I. CEUR Workshop Proceedings, 2024, 3842, pp. 178–183.

48. Заікіна Д., Глива В. Основи охорони праці та безпека життєдіяльності. 2019. URL: https://doi.org/10.31435/rsglobal/001 (дата звернення: 14.12.2024).

49. Безпека в надзвичайних ситуаціях. Методичний посібник для здобувачів освітнього ступеня «магістр» всіх спеціальностей денної та заочної (дистанційної) форм навчання / укл.: Стручок В. С. Тернопіль: ФОП Паляниця В. А., 2022. 156 с.

ДОДАТКИ

ДОДАТОК А

Тези конференції

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ТЕРНОПІЛЬСЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
імені ІВАНА ПУЛЮЯ
НАУКОВЕ ТОВАРИСТВО ім. ШЕВЧЕНКА

М А Т Е Р І А Л И

XII НАУКОВО-ТЕХНІЧНОЇ КОНФЕРЕНЦІЇ

«ІНФОРМАЦІЙНІ МОДЕЛІ, СИСТЕМИ ТА ТЕХНОЛОГІЇ»

18-19 грудня 2024 року

ТЕРНОПІЛЬ
2024

УДК 004.5

С. С. Сучков
Тернопільський національний технічний університет імені Івана Пулюя, Україна

СЕРВІСИ ДЛЯ МАШИННОГО ПЕРЕКЛАДУ КЛЮЧОВИХ СЛІВ ТА АНОТАЦІЇ НАУКОВИХ ТЕКСТІВ

S. Suchkov

SERVICES FOR MACHINE TRANSLATION OF KEYWORDS AND ANNOTATION OF SCIENTIFIC TEXTS

Існує кілька найбільш популярних сервісів для перекладу текстів:

– DeepL Translator (https://www.deepl.com/translator) - використовує нейронні мережі глибокого навчання для обробки природної мови та перекладу текстів між різними мовами. Сервіс спирається на великі мовні дані та алгоритми для моделювання мовних структур та контексту, що дозволяє досягати високої точності та природності перекладу;

– Google Translate (https://translate.google.com/) - використовує статистичні та нейронні методи обробки природної мови для перекладу слів, фраз та веб-сторінок між різними мовами. Сервіс інтегрує технології машинного навчання та великі обсяги даних для покращення якості перекладу;

– Microsoft Translator (https://translator.microsoft.com/) - надає функції перекладу тексту та мовлення в реальному часі. Також використовує нейронні мережі та методи машинного навчання для забезпечення перекладу з підтримкою багатьох мов, а також пропонує інтеграцію з іншими продуктами та сервісами Microsoft;

– Baidu Translate (https://fanyi.baidu.com) - розроблений китайським гігантом Baidu. Застосовує комбінацію статистичних методів та нейронних мереж для перекладу текстів та веб-сторінок різними мовами, прагнучи забезпечити високу точність та розуміння контексту перекладених матеріалів.

Для порівняльного аналізу чотирьох сервісів перекладу, які можна інтегрувати в систему, було розглянуто такі ключові аспекти: підтримка мов, функціональність і доступність. Результати порівняльного аналізу представлені у табл. 1.

Таблиця 1 – Порівняння сервісів машинного перекладу тексту з метою інтеграції в систему, що розробляється

Критерій	DeepL Translator	Google Translate	Microsoft Translator	Baidu Translate
Підтримка мов	близько 30	понад 100	понад 70	понад 100
Наявність API	DeepL API	Google Cloud Translation API	Microsoft Translator Text API	Baidu Translate API
Вартість	С версія безкоштовна	С версія безкоштовна	С версія безкоштовна	С версія безкоштовна
Доступність API	Сервіс доступний	Сервіс доступний	Сервіс доступний	Сервіс доступний обмежено

Важливим критерієм є доступність сервісів, тому серед розглянутих найкращим вибором є Google Translate API. У випадку системи, що розробляється, було вирішено, що сервіс Google Translate якнайкраще підходить для реалізації прототипу системи.

Додаток В

Диск з роботою