

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет Автоматики та комп'ютеризованих технологій
(повна назва)

Кафедра Комп'ютерно-інтегрованих технологій, автоматизації та
робототехніки
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА Пояснювальна записка

рівень вищої освіти перший (бакалаврський)

Розробка автоматизованої системи підтримки академічної доброчесності
(тема)

Виконав:

студент IV курсу, групи АКТСІ-20-2
Левенець І. О.
(прізвище, ініціали)

Спеціальність 151 Автоматизація та
комп'ютерно-інтегровані технології
(код і повна назва спеціальності)

Тип програми освітньо-професійна
Освітня програма Системна інженерія
(повна назва освітньої програми)

Керівник проф. Сезонова І. К.
(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри _____
(підпис)

Невлюдов І. Ш.
(прізвище, ініціали)

2024 р.

ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ РАДІОЕЛЕКТРОНІКИ

Факультет Автоматики та комп'ютеризованих технологій
Кафедра КІТАР
Рівень вищої освіти перший (бакалаврський)
Спеціальність 151 Автоматизація та комп'ютерно-інтегровані технології
Тип програми Освітньо-професійна
Освітня програма Системна інженерія
(шифр і назва)

ЗАТВЕРДЖУЮ:
Зав. кафедри КІТАР _____
(підпис)
« ____ » _____ 2024 р.

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

студентові Левенцю Іллі Олександровичу
(прізвище, ім'я, по батькові)

1. Тема роботи Розробка автоматизованої системи підтримки академічної доброчесності

Затверджена наказом по університету від 03.06.2024 р. № 544 Ст _____

2. Термін подання студентом роботи до екзаменаційної комісії 20.06.2024 р.

3. Вихідні дані до роботи експертна система, база даних, програма на мові програмування C++, керівництво користувача

4. Перелік питань, що потрібно опрацювати в роботі _____

4.1 Вступ;

4.2 Аналіз технічного завдання;

4.3 Огляд експертних систем в освітній діяльності;

4.4 Розробка лінгвістичних алгоритмів обробки речень;

4.5 Розробка програми для підтримки прийняття рішень;

4.6 Розробка керівництва користувача;

4.7 Охорона праці;

4.8 Висновки.

Я, Левенець Ілля Олександрович, як студент ХНУРЕ, розумію і підтримую політику закладу із академічної доброчесності. Я не надавав і не одержував недозволену допомогу під час підготовки кваліфікаційної роботи. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело.

15.06.2024



І. О. Левенець

РЕФЕРАТ

Пояснювальна записка: 104 с., 21 рис., 4 дод., 25 джерел.

ЕКСПЕРТНА СИСТЕМА, ЛІНГВІСТИЧНИЙ АНАЛІЗ, АКАДЕМІЧНА ДОБРОЧЕСНІСТЬ, ПЛАГІАТ, БАЗА ДАНИХ, БАЗА ЗНАНЬ, АЛГОРИТМ ВИСНОВКУ

Мета роботи – підвищення рівня академічної доброчесності в роботі освітньо-наукового підрозділу за рахунок впровадження комп'ютерної системи аналізу назв кваліфікаційних робіт.

Предмет – програмний продукт, структура експертної системи, база знань та методики лінгвістичного контролю для дотримання правил академічної доброчесності в діяльності освітньо-наукового підрозділу (кафедри).

Об'єкт – автоматизація управління університетською кафедрою щодо дотримання академічної доброчесності за допомогою комп'ютерної експертної системи.

В роботі здійснено аналіз існуючих експертних систем для перевірки текстів на наявність плагіату. Розроблено модель експертної системи, придатну для автоматизації перевірки на плагіат тем кваліфікаційних робіт кафедри. Створено програмний продукт для підтримки прийняття рішень щодо наявності плагіату в темах кваліфікаційних робіт.

В кваліфікаційній роботі розглянуто актуальні питання за темою, запропоновано новий підхід до аналізу текстової інформації визначеної тематики.

ABSTRACT

Explanatory note: 104 p., 21 fig., 4 adj., 25 sources

EXPERT SYSTEM, LINGUISTIC ANALYSIS, ACADEMIC INTEGRITY, PLAGIARISM, DATABASE, KNOWLEDGE BASE, INFERENCE ALGORITHM

The purpose of the work – design and development of a computer system to support the decision-making of academic integrity in the work of the educational and scientific unit.

Subject – software product, structure of expert system, knowledge base and methods of linguistic control for compliance with the rules of academic integrity in the activities of the educational and scientific unit (department).

Object – automation of the management of the university department regarding compliance with academic integrity with the help of a computer expert system.

The paper analyzes existing expert systems for checking texts for plagiarism. A model of an expert system, suitable for automating the plagiarism check of topics of the department's qualification papers, has been developed. A software product was created to support decision-making regarding the presence of plagiarism in the topics of qualification papers.

In the qualification work, topical questions on the topic were considered, a new approach to the analysis of textual information on a specific topic was proposed.

ЗМІСТ

Перелік скорочень	9
Вступ	6
1 Аналіз технічного завдання	13
1.1 Автоматизовані експертні системи перевірки на академічний плагіат.....	13
1.2 Структура та типологія експертних систем	14
1.3 Аналіз експертних систем в освітньо-науковій діяльності.....	17
1.3.1 Освітні експертні системи.....	17
1.3.2 Експертні системи перевірки на плагіат.....	20
1.4 Постановка завдання на розробку експертної системи	23
2 Розробка експертної системи академічної доброчесності.....	26
2.1 Алгоритм роботи експертної системи виявлення плагіату.....	26
2.2 Логіко-математичне моделювання текстової інформації для задачі виявлення плагіату.....	27
2.3 Моделі синтаксичного аналізу речень.....	30
2.4 Методи синтаксичного аналізу текстової інформації.....	35
2.5 Вибір програмного інструментарію для створення експертної системи...	37
3 Програмна реалізація системи академічної доброчесності.....	39
3.1 Середовище розробки та стандартні модулі.....	39
3.2 Опис програмного продукту.....	42
3.3 Керівництво користувача.....	45
4 Охорона праці.....	49
4.1 Промислова безпека та аналіз умов праці.....	49
4.2 Виробнича санітарія.....	49
Висновки	52
Перелік джерел посилання	53

Додаток А Вид вхідного документа.....	57
Додаток Б Код програми.....	63
Додаток В Демонстраційний матеріал.....	101

ПЕРЕЛІК СКОРОЧЕНЬ

- АПС – автоматизовані інформаційно-пошукові системи;
- АОТ – автоматична обробка тексту;
- БД – база даних;
- БЗ – база знань;
- ВНЗ – вищі навчальні заклади;
- ЗВО – заклади вищої освіти;
- ПМ – природньо-мовний;
- ПЗ – програмне забезпечення;
- ІІ – інформаційний процес;
- ІС – інформаційна система;
- ІТ – інформаційна технологія;
- СМП – система машинного перекладу;
- СУБД – система управління базами даних;
- СГЗ – системи, що ґрунтуються на знаннях;
- ТМЦ – товарно-матеріальні цінності;
- УкрІНТЕІ – Український інститут науково-технічної експертизи та інформації;
- ЕС – експертна система;
- ІІІ – штучний інтелект;
- АРА (American Psychological Association) – стиль цитування і оформлення бібліографічних описів, розроблюваний Американською психологічною асоціацією;
- AIWBES (Adaptive and intelligent Web-based educational systems) – адаптивні та інтелектуальні навчальні веб-системи;
- VS – інтегроване середовище розробки Visual Studio від Майкрософт;
- SAIUP (Strengthening Academic Integrity in Ukraine Project) – проект сприяння академічній доброчесності в Україні;

LMS (Learning Management System) – система управління навчанням;

IQ (Quality Initiative) – якість освіти;

MLA (Modern Language Association) – стиль цитування Асоціації сучасних

мов.

ВСТУП

Однією з ключових складових внутрішньої системи забезпечення якості навчання у закладах вищої освіти є система забезпечення академічної доброчесності.

Академічна доброчесність – це сукупність етичних принципів та визначених законом правил, якими мають керуватися учасники освітнього процесу під час навчання, викладання та провадження наукової (творчої) діяльності з метою забезпечення довіри до результатів навчання або наукових (творчих) досягнень.

Проект сприяння академічній доброчесності в Україні (Strengthening Academic Integrity in Ukraine Project – SAIUP) реалізується Американськими Радами з міжнародної освіти за сприяння Міністерства освіти і науки України за підтримки Посольства США в Україні. Проект має на меті застосовувати спільний досвід США і України для розробки та втілення плану заходів, зміст яких полягає у орієнтації та навчанні студентів, викладачів та адміністраторів навчальних закладів України практичної цінності і важливості академічної доброчесності, наданні ресурсів і плану дій задля їхнього тісного залучення до зміцнення академічної доброчесності в освітньому середовищі [5].

Учасниками Проекту сприяння академічній доброчесності в Україні стали 10 українських ВНЗ з різних регіонів України, в тому числі Харківський національний університет радіоелектроніки.

Проект реалізовувався протягом 4 років (2016-2019), що дозволило охопити ціле покоління студентів – бакалаврів, починаючи з їх першого року навчання в університеті і закінчуючи отриманням дипломів бакалавра. Саме це покоління випускників повинно стати носієм нових цінностей у вищій освіті, в науці та в суспільстві загалом.

Зараз діє Проект «Ініціатива академічної доброчесності та якості освіти» (Academic Integrity and Quality Initiative – Academic IQ), що реалізується

Американськими Радами з міжнародної освіти за сприяння Посольства США в Україні (Відділу преси, освіти та культури), Міністерства освіти і науки України та Національного агентства із забезпечення якості вищої освіти, має на меті об'єднати професійну спільноту освітян для обміну досвідом та співпраці задля інституціоналізації принципів академічної доброчесності в системі середньої та вищої освіти України.

Мета роботи – підвищення рівня академічної доброчесності в роботі освітньо-наукового підрозділу за рахунок впровадження комп'ютерної системи аналізу назв кваліфікаційних робіт.

Предмет – програмний продукт, структура експертної системи, база знань та методики лінгвістичного контролю для дотримання правил академічної доброчесності в діяльності освітньо-наукового підрозділу (кафедри).

Об'єкт – автоматизація управління університетською кафедрою щодо дотримання академічної доброчесності за допомогою комп'ютерної експертної системи.

Для досягнення мети необхідно вирішити наступні задачі:

- здійснити аналіз предметної області;
- здійснити аналіз структури і завдань експертних систем;
- розробити модель експертної системи, придатну для автоматизації перевірки на плагіат тем кваліфікаційних робіт кафедри;
- створити проект бази даних та програмний продукт для прийняття рішень щодо наявності плагіату в темах кваліфікаційних робіт.

Результати кваліфікаційної роботи сприяють Цілям сталого розвитку, зокрема цілі 4 Забезпечення всеохоплюючої і справедливої якісної освіти та заохочення можливості навчання впродовж усього життя для всіх, п. 4.4. Вирішення задачі дотримання академічної доброчесності підвищує якість вищої освіти та забезпечує її тісний зв'язок з наукою [1].

Пояснювальна записка виконана згідно [2,3].

1 АНАЛІЗ ТЕХНІЧНОГО ЗАВДАННЯ

1.1 Автоматизовані експертні системи перевірки на академічний плагіат

Академічна доброчесність – це сукупність етичних принципів та визначених законом правил, якими мають керуватися учасники освітнього процесу під час навчання, викладання та провадження наукової (творчої) діяльності з метою забезпечення довіри до результатів навчання та/або наукових (творчих) досягнень. Порухення правил академічної доброчесності не лише паплюжить репутацію університету, а й зменшує цінність результатів освіти, здобутих в університеті. Відповідальність за дотримання принципів академічної доброчесності є водночас як особистим обов'язком кожного, так і спільною справою університетської громади загалом. Кожен член університетської академічної спільноти, зокрема здобувач відповідальний за дотримання принципів академічної чесності. Під час виконання кваліфікаційних робіт, найпоширенішою формою порушення академічної доброчесності є академічний плагіат, тобто привласнення авторства на чужу академічну працю, а також використання у своїй роботі чужого твору без посилання на автора.

Плагіат – серйозне порушення академічних норм та правил і може бути виражений із застосуванням різних форм – від дослівного копіювання до перефразування іншої роботи. Загальнодоступна електронна форма висвітлення матеріалів дає змогу не лише багаторазово використовувати і розповсюджувати інформацію, а й сприяє академічному плагіату. В студентському середовищі плагіат найчастіше виражений у несамотійному виконанні студентських навчальних робіт або в запозиченні фрагментів чужих робіт без зазначення джерела запозичення [7].

Контроль щодо дотримання вищезазначених правил є складною задачею навидь для висококваліфікованих фахівців та потребує впровадження автоматизованої системи. Її проектування, програмування та впровадження є актуальною задачею сьогодення.

Сьогодні плагіат урізноманітнівся і набув самих різних форм, які доволі складно виявити, якщо не користуватись досягненнями сучасних програмних продуктів. Досить розповсюдженою в українських вузах є практика, коли кваліфікаційні роботи студентів не допускаються до захисту поки не отримають результати перевірки на плагіат. Для зазначених цілей використовуються різноманітні програмні продукти: Unicheck, eTXT Антиплагіат, Advego Plagiatus. Серед зарубіжних систем перевірки найбільш відомими є такі: TurnItIn, SafeAssign, CopyScare, WriteCheck, iThenticate. Одним з головних критеріїв перевірки у зазначених системах є можливість доступу до різних видів документів, що можуть виступати джерелами запозичень. Як правило, в основу такого виду програмного забезпечення покладено використання або власної бази документів, або ж перевірка відбувається на основі формування спеціальних запитів для пошукових систем та подальшого використання отриманих результатів. Однак виявити всі види плагіату за допомогою лише програмних продуктів не можливо, оскільки комп'ютерна програма обмежується виявленням лише технічного плагіату - прямих запозичень. Відносно інших видів запозичень, таких як рерайт та інші, то, як зазначають фахівці в галузі інформаційних технологій: «...не існує надійних методів достовірного виявлення всіх випадків запозичень, включаючи такі складні випадки, як виявлення неявно виражених запозичень» [8, с. 144].

Саме тому наступним етапом виявлення академічного плагіату повинна бути змістовна перевірка на плагіат, яку здійснює експерт, що має значний досвід у відповідній галузі науки. Фахівець, керуючись своїми уявленнями про зміст документів та засоби його вираження, а також спираючись на власний професійний досвід та знання може встановити наявність або відсутність такого факту як плагіат. Програмні засоби розвиваються, й можливо у недалекому майбутньому, програма зможе виявляти також і неявні запозичення. Але вже сьогодні потрібно вживати заходів, не тільки для того, щоб виявити плагіаторів, а й для попередження такого виду порушень.

Деталізує поняття «академічної доброчесності» стаття 42 Закону України «Про вищу освіту». Слід зазначити, що у визначенні академічної доброчесності законодавець серед її порушень першим наводить саме академічний плагіат. Також

у списку порушень зазначені: самоплагіат, фабрикація, фальсифікація, списування, обман, хабарництво та необ'єктивне оцінювання [8,9].

Користуючись Великим тлумачним словником сучасної української мови, плагіат визначається як привласнення авторства на чужий твір або на чуже відкриття, винахід чи раціоналізаторську пропозицію, а також використання у своїх працях чужого твору без посилання на автора [10].

У США плагіат визначається як акт присвоєння літературного твору іншого автора, уривків, ідей або їхніх фрагментів, і видача цього матеріалу як власного твору [11].

Сайт plagiarism.org наводить схоже визначення плагіату: «.. відтворення або привласнення чужої роботи без посилання на джерело; видавання чужої роботи за власну» [12]. Зазначений сайт знайомить відвідувачів із класифікацією плагіату, запропонованою Міріам Вебстер (Merriam-Webster), яку побудовано виходячи з таких неправомірних дій, як-то:

- викрадення та видання за власні ідей та слів іншого;
- використання чужих матеріалів без посилань на джерело;
- здійснення літературної крадіжки;
- представлення як нових та оригінальних ідей чи продуктів, що було запозичено з джерела, що вже існувало.

У підсумку зазначено, що плагіат являє собою акт шахрайства, який охоплює крадіжку чужої роботи та введення в оману інших щодо авторства цієї роботи [12].

Наведена класифікація дозволяє розширити уявлення про плагіат як небезпечне явище, але не дає уявлення про всі його прояви. Найбільш повною вважаємо класифікацію проявів плагіату, яка пропонується на веб-сайті ithenticate.com [13].

Плагіат вторинних джерел інформації – Secondary Source, суть якого полягає в тому, що дослідник використовує вторинні джерела, але цитує лише первинні, що вказані у ньому. Вважаємо, що такий вид плагіату доволі складно виявити, але можливо. Для цієї мети потрібно враховувати дату публікації.

Другий вид Invalid Source – приписування недостовірних джерел інформації, яке відбувається тоді, коли автор посилається на джерела, які не використовував,

або яких взагалі не існує. Така спроба маскує некоректно проведене дослідження. Така практика не дуже поширена, але вважається більш серйозним порушенням ніж попереднє.

Третій вид, який отримав назву Duplication – дублювання відбувається в тому випадку, коли автор знов використовує власні дослідження без відповідних посилань на них. В Україні такий вид плагіату отримав назву «само плагіат».

Четвертий вид з назвою Paraphrasing – коли відбувається переписування чужого тексту, вираження ідеї іншими словами. В мережі навіть існує вид такої творчості, що отримав назву рерайт, суть якого полягає в запозиченні фрагментів, а іноді навіть цілого твору шляхом заміни його оригінального авторського тексту. Відбувається зміна форми (структурної або лексичної) зі збереженням змісту першоджерела, яке не відображується.

Repetitive Research – наступний вид, який дублює інформацію або текст схожого дослідження, що має відповідну методологію, але без належних посилань на джерела. Тобто автор використовує не сам текст, а систему методів першоджерела.

Replication – копіювання або подання наукової роботи більш ніж один раз, коли автор запевняє, що дослідження нове, в той час як воно вже було опубліковано. Таке порушення етики є досить поширеним та стосується не лише наукової сфери. Реплікацією також називають дублювання певної інформації на різних сайтах – тиражування контенту без дозволу автора.

Misleading Attribution – введення в оману приписуванням. Про цей вид плагіату йдеться коли відбувається існує некоректний список внесків авторів в певне дослідження. Може мати дві форми: коли авторів не вказують, або навпаки додають тих, хто не мав до нього відношення.

Восьмий вид з назвою Unethical Collaboration – неетичне співробітництво, яке стосується порушення етики у співавторстві. Відбувається публікація без вказання всіх співавторів твору.

Verbatim Plagiarism – дослівний плагіат або копіювання чужих текстів або робіт без відповідних посилань. Має два прояви: по-перше, можуть присвоюватися ідеї або тексти без жодного посилання; по-друге, порушник посилається на

джерело, але не вказує, що це пряма цитата.

Complete Plagiarizm – повний плагіат, коли дослідник використовує роботу іншого дослідника та подає її як власну [11].

Слід зазначити, що наведений вище перелік охоплює найбільшу кількість видів неправомірних запозичень. Всі перераховані види можна поділити на дві великі групи: явні та неявні. Перші досить легко виявити, користуючись програмними засобами. Щодо другої групи неявно вираженого плагіату слід зазначити, що найбільш ефективною вважаємо експертну перевірку.

Програмний продукт, який дозволяє виявити явні ознаки плагіату має відноситись до класу експертних систем та потребує нетрадиційного підходу до його проектування за рахунок специфіки предметної області, її низького рівня формалізації.

1.2 Структура та типологія експертних систем

Експертні системи (ЕС) з'явилися внаслідок розвитку систем зі штучним інтелектом (ШІ). Вирішення завдань за допомогою логічного висновку можливе лише якщо база знань ЕС містить високоякісні знання про предметну область, тоді механізм логічного висновку міститиме інформацію про те, як ці знання ефективно використовувати.

В даний час виявлено великий інтерес до ЕС у всіх галузях сучасної науки та людської діяльності. ЕС використовуються для вирішення різних типів завдань у самих різноманітних проблемних галузях, таких, як високотехнологічна промисловість, фінанси, енергетика, транспорт, фармацевтичне виробництво, медицина, освіта, телекомунікації, зв'язок тощо.

З іншого боку, використання ЕС обмежено та недостатньо розроблено. Широкому впровадженню даних систем заважає велика кількість невирішених теоретичних проблем та практики використання. Кожна працююча ЕС є унікальним програмним продуктом, яка потребувала окремого вивчення предметної області, формалізації знань в цій області та створення унікального програмного продукту.

В ІТ-галузі ЕС розглядаються спільно з базами знань як моделі поведінки експертів у певній галузі знань з використанням процедур логічного висновку та прийняття рішень, а бази знань – як сукупність фактів та правил логічного висновку у вибраній предметній області.

Під ЕС розуміють складні програмні комплекси, які акумулюють знання фахівців у конкретних предметних галузях і тиражують цей емпіричний досвід для консультацій менш кваліфікованих користувачів.

Системи, що ґрунтуються на знаннях (СГЗ) є програмним забезпеченням, основними структурними елементами яких є база знань та механізм логічних висновків [13].

Серед СГЗ можна виділити:

- інтелектуальні інформаційно-пошукові системи;
- експертні системи.

Базова структура ЕС наведена на рисунку 1.1.



Рисунок 1.1 – Структура експертної системи

Структурні елементи, що становлять експертну систему: база знань (БЗ), модуль логічних висновків, модуль набуття знань, модуль порад та пояснень, користувацький інтерфейс, база даних (БД).

Розглянемо функції, які виконують структурні елементи ЕС.

БЗ містить факти та правила, за якими залежно від вхідної інформації приймається те чи інше рішення. Знання у БЗ представлені у конкретній формі, і організація БЗ дозволяє їх легко визначати, модифікувати та поповнювати. БЗ реалізує функції формалізації знань у конкретній предметній області та управління ними.

Модуль логічних висновків виконує логічні висновки із знань, наявних у БЗ. Механізмом логічного висновку називають загальні знання про процес прийняття рішення. Він виконує дві основні функції:

- доповнення, зміна БЗ на основі аналізу БЗ та вихідної інформації;
- управління порядком обробки правил у БЗ.

Існує дві основні стратегії логічного висновку:

- прямий ланцюжок міркувань;
- зворотній ланцюжок міркувань.

Інтерфейс користувача необхідний для правильної передачі відповідей користувачу, інакше користуватися системою буде вкрай незручно.

Модуль набуття знань необхідний для отримання знань від експерта, підтримки БЗ та доповнення її за необхідності.

Модуль додаткових пояснень формує заключення експертної системи і представляє особисті коментарі, що додаються до висновку, а також пояснення щодо отриманих висновків.

Перелічені структурні елементи є найбільш характерними, хоча в реальних експертних системах їх функції можуть бути відповідним чином змінені чи розширені.

У загальному випадку, всі системи, побудовані на знаннях, можна розділити на системи, що вирішують задачі аналізу та системи, які вирішують задачі синтезу.

Основна відмінність завдань аналізу від задач синтезу полягає в тому, що в задачах аналізу множину рішень можна перерахувати та включити до системи, то в задачах синтезу безліч рішень потенційно не обмежена.

Завданнями аналізу є інтерпретація даних, діагностика, підтримка прийняття рішення. До завдань синтезу відноситься проектування, планування, управління.

Комбіновані завдання спрямовані на навчання, моніторинг, прогнозування [5].

На сьогоднішній день створено вже велику кількість експертних систем, які мають одну велику відмінність від інших систем штучного інтелекту: вони не призначені для вирішення будь-яких універсальних задач. Вони призначені для якісного рішення завдань у визначеній розробниками області, в окремих випадках – областях. І незважаючи на всі обмеження та недоліки, ЕС вже доказують свою цінність і значимість у багатьох областях [14].

1.3 Аналіз експертних систем в освітньо-науковій діяльності

1.3.1 Освітні експертні системи

Перш за все використання ЕС пов'язане з інтерпретуванням знань, передбаченням та діагностикою явищ. Проектування та реалізація ЕС передбачає наявність чітких закономірностей та правил у досліджуваній області. Тим не менш, сучасні ЕС успішно створюються та експлуатуються на веб-платформах.

У світі вже існують інтерактивні помічники та адаптивні програми і для навчання, які дають змогу за допомогою штучного інтелекту індивідуалізувати освітній процес і автоматизувати певні технічні завдання.

Одним із прикладів успішної реалізації ЕС в онлайн-просторі є системи AIWBES (від. англ. Adaptive and intelligent Web-based educational systems) – веб-система інтелектуального навчання та підтримки [16]. Першими AIWBES-системами, які одними з перших реалізували програмування навчальних курсів, були ELM-ART та KBS-Hyperbook. У ELM-ART програмування реалізоване у формі рекомендованих посилань і адаптивної кнопки «Далі». У KBS-Hyperbook це реалізується як рекомендований навчальний шлях.

Наступні розробки AIWBES-систем мають модулі інтелектуального аналізу рішень, які пов'язані із розв'язанням студентами навчальних задач (які можуть змінюватись від простих запитань до комплексних програмних завдань). На відміну від неінтелектуальних перевіряючих інструментів, які здатні лише повідомляти про вірне розв'язання завдання, інтелектуальні аналізатори можуть повідомити, що саме є невірним або що розв'язано неповністю, і які пропущені чи

невірні знання можуть відповідати за помилку. Інтелектуальні аналізатори здатні забезпечити студента потужною технікою зворотного зв'язку для опрацювання помилок.

Детальний аналіз інтелектуальних освітніх платформ, які використовують технологію ЕС проведено Арханом Хатрі Дас (Archana Khatri Das) [17].

Smart Sparrow – це платформа з веб-орієнтованим набором навчальних курсів. Навчальні матеріали надаються учню на підставі аналізу його відповідей та аналітики навчальних курсів, залежно від результатів виконаних завдань учнем. Адаптивні функції та визначені алгоритми в Smart Sparrow не закладаються заздалегідь, викладачі мають змогу самостійно створювати адаптивні функції на основі аналізу таких динамічних даних, як час перебування студента на веб-сторінці чи кількість спроб відповідей на запитання тощо. Багатофункціональність освітньої платформи ScootPad робить її популярною в освітян. ScootPad – це платформа, на якій використовуються адаптивні алгоритми з прогнозуванням аналітики, з візуалізацією даних для забезпечення кожному учню диференційованого навчання з індивідуальними навчальними траєкторіями.

Новаторська технологія навчальної платформи ALEKS розроблена на основі досліджень Нью-Йоркського та Каліфорнійського університетів. Характерною особливістю цієї навчальної платформи є використання штучного інтелекту з теорією простору знань, щоб швидко та точно визначити, які саме знання учень уже опанував і вказати на теми, які учень готовий опанувати надалі з періодичним оцінюванням його навчальної діяльності. Теорія простору знань базується на поняттях комбінаторики та стохастичних процесів, моделювання та емпіричного опису окремих галузей знань.

SmartUp – це навчальна платформа, яка дозволяє користувачам створювати спільноти для ведення бізнесу з можливістю формувати зміст навчання відповідно до власних уподобань. Ця платформа використовується для навчання бізнес-школами та персоналу банків.

Також потребує уваги технологія Domoscio [18], що використовує штучний інтелект для вивчення особистих даних учня та використання їх на генерованих LMS-платформах (системи управління навчанням) з метою забезпечення кожному

учню оптимального шляху до досягнення власних цілей у навчанні. Адаптивність навчання забезпечується: механізмом рекомендацій, що надають учню найбільш адаптовані навчальні ресурси відповідно до його профілю; інструменти консолідації навчання, які надають учню персоналізований план навчальної діяльності відповідно до його потреб; інструменти оповіщення, за допомогою яких різним користувачам (учню, викладачу, тренеру та ін.) у потрібний час надсилаються потрібні повідомлення та інформаційні джерела.

Створення адаптивної навчальної системи визначається специфікою освітнього процесу, яка повинна враховувати: індивідуально-типологічні особливості учнів; структуру та рівень початкової підготовки учня; рівень розвитку здібностей учня; статистику використання учнем навчального середовища; поточні результати підготовки учня; соціальні умови учня й інше та перспективний стан розвитку цифрових технологій Суспільства 5.0. Ці технології передбачають опрацювання значних масивів даних у кіберпросторі за допомогою штучного інтелекту і надання опрацьованих даних у вигляді нових рішень.

Визначимо суттєві аспекти побудови моделі адаптивної навчальної системи .

До складу адаптивної моделі входять такі компоненти:

- модуль збору даних, в якому збір та структуризація даних відбувається в реальному часі;
- модуль штучного інтелекту, який включає експертні системи, бази знань, блок коригування. Завдяки вдосконаленню технологій штучного інтелекту індивідуальні характеристики учня частково враховуються в інтелектуальних системах навчання (Intelligent Tutoring Systems) і в недалекому майбутньому будуть застосовуватись абсолютно на всіх етапах навчання;
- модуль підготовки до навчання (мотиваційний блок, контент, технології навчання), який на основі зібраних даних формує контент предметних дисциплін та надає технології для засвоєння контенту;
- модуль навчання (опрацювання та засвоєння контенту, контроль засвоєння знань у процесі навчання) формує цілісний процес навчання, що оптимізує розумовий розвиток учня;
- модуль контролю знань (рис. 1.2).

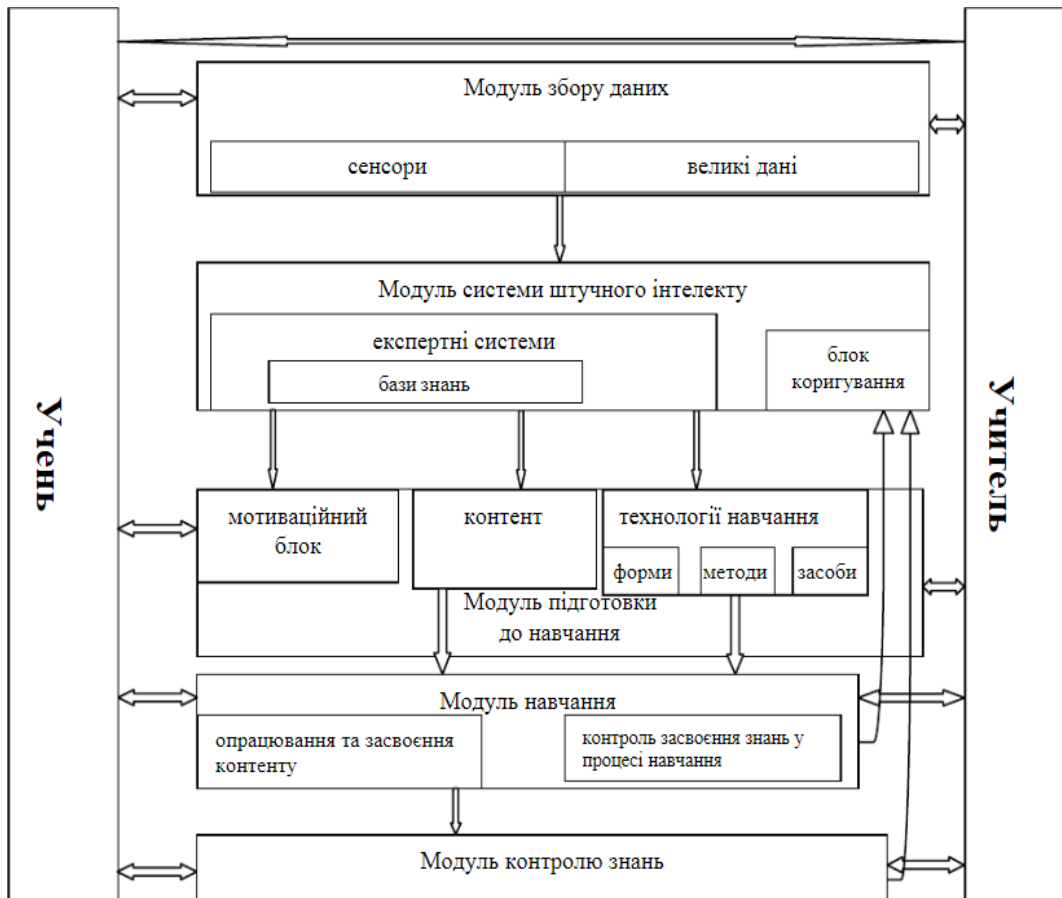


Рисунок 1.2 – Модель адаптивної навчальної системи з модулем ШІ[19]

Останні тенденції в освітніх процесах показують актуалізацію технологій адаптивного навчання та необхідність змін у підході до навчання. Для розв'язання такої задачі, враховуючи її складність та багатогранність, доцільно створювати адаптивні навчальні системи з використанням штучного інтелекту.

1.3.2 Експертні системи перевірки на плагіат

Основними програмами, які використовують українські ЗВО є Unicheck та StrikePlagiarism [20]. Вони мають великі бази даних студентських робіт та схожі принципи роботи. Кожен ЗВО, який користується програмою, завантажує студентські роботи для перевірки унікальності, перевіряючи плагіат не лише з Інтернету, а й із внутрішньої бази.

Функціонал програми Unicheck:

– візуально виділяє всі збіги тексту, цитати та посилання, допомагаючи

швидко знайти неоригінальний текст;

– видає точний відсоток схожості, дозволяє автоматичне видалення джерел, цитат (в стилях цитування APA, MLA, Chicago/Turabian, Harvard), списку використаної літератури та отримує точний відсоток схожості [21];

– швидка навігація допоможе швидко переглядати документ, знаходити цитати, збіги та посилання;

– показує текстові збіги, які варті вашої уваги, та не включає часто вживані вислови до результатів перевірки;

– порівнює перевірений текст з джерелом, в якому є схожі або такі ж самі формулювання, в одному вікні;

– звіти Unicheck завжди залишаються в доступі і можуть бути завантажені у форматі .pdf для офлайн-використання;

– програма генерує список працюючих посилань на джерела з текстовими збігами. Небезпечні посилання мають спеціальну позначку-попередження;

– якщо Unicheck помітить підозріле форматування тексту або інші форми академічного шахраювання, він повідомить вас про це у вкладці Modifind.

Система StrikePlagiarism виявляє фрагменти аналізованого документа, ідентичні текстам, розміщеним в порівняльній базі даних, включно з текстами, котрі відмічені лапками, виносками і фрагментами, які не захищені авторським правом (наприклад, правові акти). Система не вказує на вміст плагіату в документі!

Система Strikeplagiarism.com порівнює всі завантажені документи з джерелами в наступних базах даних :

– документи в домашній базі даних університету (документи «додані в базу даних»);

– документи з бази даних інших університетів (програма обміну базами даних). Щоб забезпечити перевірку по базах даних інших університетів, необхідно підписати угоду про обмін базами даних;

– інтернет-джерела;

– база даних RefBooks. Дана база створена компанією Plagiat.pl Sp.o.o. (Польща) на основі книг, статей, а також інших наукових і спеціалізованих

публікацій, наданих авторами і видавцями виключно для цілей антиплагіатного аналізу. В даний час вона містить більше 3 мільйонів текстів, захищених авторським правом, в основному на польській та англійській мовах [20];

- юридична база даних;
- база даних УкрІНТЕІ. Завдяки співпраці StrikePlagiarism.com з УкрІНТЕІ надає доступ перевірки у своїй базі даних.

Основним недоліком цих програм є те, що вони працюють в режимі надання послуг на комерційній основі. В залежності від режимів перевірки ціна послуги може наблизитись до еквіваленту 100 доларів США.

Тому відслідковувати наявність запозичень і ознак плагіату в роботах працівників, студентів, аспірантів невеликих освітньо-наукових підрозділів (наприклад, кафедрах, факультетах тощо) за допомогою наявних на ринку програм перевірки на плагіат вбачається недоцільним.

Підтримка роботи освітньо-наукового підрозділу із забезпечення академічної доброчесності потребує розробки спеціальної програми, яка відноситься до класу експертних систем, база даних якої зберігає документи підрозділу.

1.4 Постановка завдання на розробку експертної системи

Виконання завдання кваліфікаційної роботи полягає в розробці комп'ютерної експертної системи локального (кафедрального) рівня, яка слугує перевірці тем кваліфікаційних робіт кафедри з метою дотримання правил академічної доброчесності.

Проект передбачає створення програмного засобу, з використанням якого можна створити експертну систему для пошуку схожих за назвою документів (кваліфікаційних, наукових робіт). Метою даного проекту є створення простого у експлуатації інструменту для підтримки прийняття рішень з питань академічної доброчесності.

Для виконання мети проекту необхідно здійснити наступні кроки:

- розробити модель ЕС;
- забезпечити роботу з даними за допомогою зручного інтерфейсу;

- розробити критерії пошуку та механізми висновків;
- розробити модель бази даних та заповнити її;
- розробити лексичний, синтаксичний, семантичний аналізатори.

2 РОЗРОБКА ЕКСПЕРТНОЇ СИСТЕМИ АКАДЕМІЧНОЇ ДОБРОЧЕСНОСТІ

2.1 Алгоритм роботи експертної системи виявлення плагіату

Головним недоліком більшості систем антиплагіату є відсутність можливості детального лінгвістичного аналізу текстів природною мовою, що включає морфологічний, синтаксичний, семантичний та прагматичний види аналізу.

Одним із головних завдань експертної системи є вирішення проблеми якісної автоматизованої обробки природної мови. Це вимагає залучення цілої низки наукових дисциплін та їх методів, у тому числі методів комп'ютерної, когнітивної, математичної лінгвістики, теорії штучного інтелекту, семантичних мереж, нейрокібернетики, логіки тощо.

Комплексне застосування засобів, моделей та методів наукових галузей в принципі дозволяє створити ефективний інструментарій обробки природно-мовної текстової інформації. Але оскільки в роботі йдеться про систему оцінювання ступеня подібності назв наукових робіт, тобто речень визначеної природи та подібних один до одного стилістично, то на перший план висувається синтаксична та семантична складова лінгвістичного аналізу

Лінгвістичний аналіз текстів природною мовою в системах аналізу документів на наявність плагіату пропонується виконувати в такій послідовності.

На першому етапі необхідно здійснити нормалізацію тексту із застосуванням графематичного аналізу (токенізації), що передбачає виконання наступних процедур:

- видалення службової інформації;
- видалення нетекстових символів;
- поділ тексту на речення;
- виділення слів (лексем);
- виділення цифр, скорочень та інших елементів тексту.

Розглянувши структуру вхідного документа (Додаток А) до службової

інформації потрібно віднести текст, який передує таблиці та зазначений наприкінці таблиці. Також попередня обробка тексту включає видалення зайвих стовбців, які не будуть включені до лінгвістичного аналізу. Це стовбці Місце проходження практики, Керівник практики та Примітка.

Підлягає попередній обробці стовпчик № п/п. Інформація, яка зберігається у цьому стовпчику, буде доповнена шифром, який визначає рік, спеціальність, кваліфікаційний рівень.

Наприклад, 1 перетворюється на 1.2024.СІ.Б. Отриманий шифр означає: номер 1, 2024 рік, освітня програма СІ, рівень кваліфікаційної роботи «бакалавр».

Аналіз одиниць лексичного рівня (лексем), тобто слів, є одним із найважливіших.

До інструментарію цього виду аналізу відносять різні словники (таблиці) синонімів, які містять словозмінні парадигми, таблиці (правила) побудови відмінків лексем. Також можна використовувати позасловникові методи, які дозволяють здійснювати розбиття речення на фрази за заданим алгоритмом та забезпечувати порівняння речень через порівняння фраз.

Узагальнений алгоритм роботи експертної системи виявлення плагіату зображено на рис. 2.1.

Алгоритм роботи експертної системи базується на базі даних, яка має визначену структуру записів. Тому необхідна попередня обробка текстових даних, яка є окремою задачею проектувальника.

Також в залежності від типу аналізу проектується структура синтаксичної таблиці, зміст якої є основою роботи бази знань. Синтаксична таблиця може бути замінена переліком лексем або словосполучень, які будуть основою пошуку запозичень.

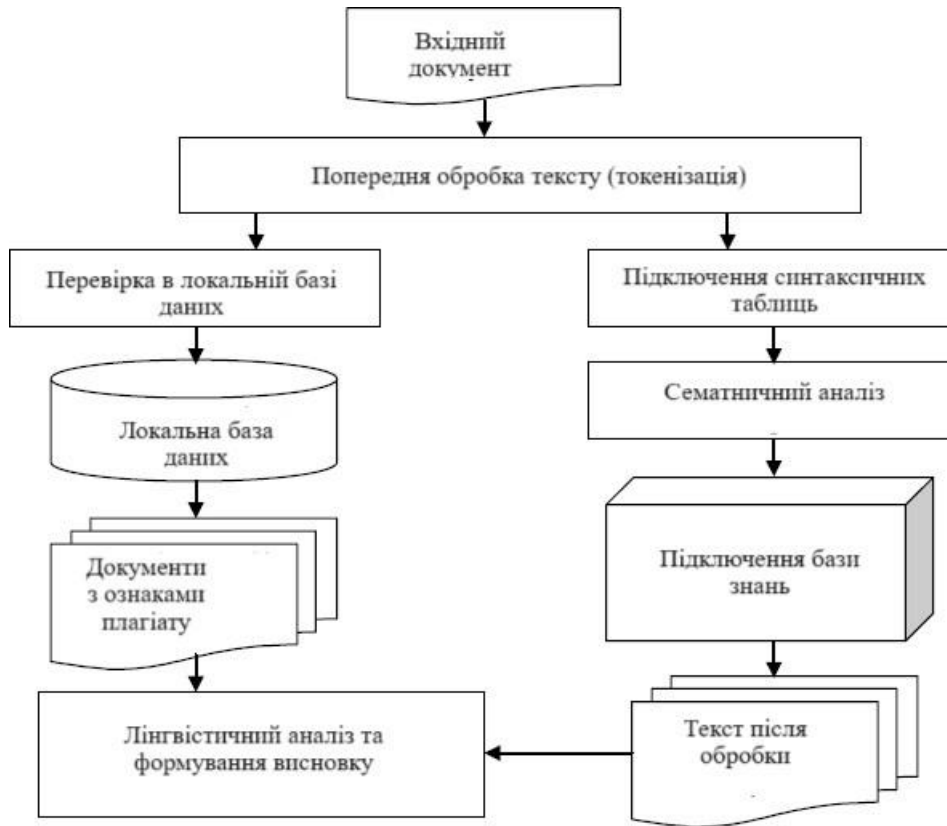


Рисунок 2.1 – Узагальнений алгоритм роботи експертної системи виявлення плагіату

2.2 Логіко-математичне моделювання текстової інформації для задачі виявлення плагіату

На формальному рівні текст природною мовою представляється ставиться у вигляді нескінченної послідовності символів з певного (також кінцевого) набору, що називається узагальненим алфавітом. У цей алфавіт включені не лише літери певної мови, але й інші символи, які можуть бути в тексті (розділові знаки, цифри, спеціальні символи тощо). Отже, найзагальніше формальне уявлення тексту виглядає так:

$$T = x_1 * x_2 * \dots * x_k, \quad x_i \in A, \quad A = \{a_1, a_2, \dots, a_N\}, \quad (3.1)$$

де * – операція конкатенації;

A – узагальнений алфавіт.

Формалізація текстової структури вхідного тексту передбачає розробку методів та лінгвістичних алгоритмів (процедур), що дозволяють виділити з такої послідовності лінгвістично значущі об'єкти – морфемного, лексичного, синтаксичного, семантичного та інших мовних рівнів, які містяться у формальній послідовності символів із кінцевого алфавіту А. Зазначені лінгвістичні процедури включають операції графематичного, морфологічного, передсинтаксичного, синтаксичного та інших видів аналізу, в процесі застосування яких вдається деталізувати лінгвістичну структуру об'єкта, що підлягає формалізації.

Процедура передсинтаксичного аналізу потрібна для виділення одиниць тексту, подальшого морфологічного аналізу цих елементів, підготовки даних для синтаксичного аналізу.

Здійснюється це в такий спосіб. На вхід системи надходить текст. Насамперед потрібно виділити одиниці цього тексту: речення, окремі слова та розділові знаки. Для вирішення цього завдання застосовується графематичний аналіз.

Дещо іншою є ситуація із завданням виділення речень. Зазвичай наприкінці речення стоїть крапка, знак питання, знак оклику, багатокрапка. Однак деякі із цих знаків застосовують ще й з іншою метою, наприклад, точку часто застосовують у скороченнях, знаки питання і оклику – для позначення виразності в тексті. В реченнях предметної області, що розглядається, крапка застосовується у скороченнях і не застосовується наприкінці речення, яке є основним об'єктом перевірки, тобто назвою кваліфікаційної роботи. Як бачимо, розділовий знак не завжди є ознакою кінця речення.

Окрему проблему складають також тире та дефіс, завдання ідентифікації яких частково вирішується на рівні синтаксичного аналізу. Таким чином, спочатку графематичний аналіз за певними критеріями виділяє речення. Далі відбувається процедура виділення лексем. Якщо рядок складається з цифр та букв (наприклад, шифр), то його позначають частиною мови «числова» і відправляють у проміжний масив. Для ідентифікації значення точки у реченні потрібно запровадити правила аналізу скорочень. Після елімінації неоднозначності точок і встановлення маркерів про можливий кінець речення, можна вважати, що всі крапки, знаки питання,

знаки оклику або маркери, після яких стоїть слово з великої літери, означають закінчення речення.

Розроблений алгоритм семантичного порівняння нечіткої текстової інформації (вхідний текст, представлений для перевірки природною мовою у довільній формі, з варіантами текстів, що є в локальних базах даних) передбачає автоматичну конвертацію вхідного тексту природною мовою до внутрішньосистемного вигляду. Застосування розробленого алгоритму дозволяє усувати помилки, які можуть бути у вхідному тексті (неправильні закінчення, нестандартні скорочення тощо), визначати належність вхідного тексту до певної предметної підобласті (категорії тем).

Алгоритм нечіткого семантичного аналізу тексту за змістом речень подано на рис. 2.2.

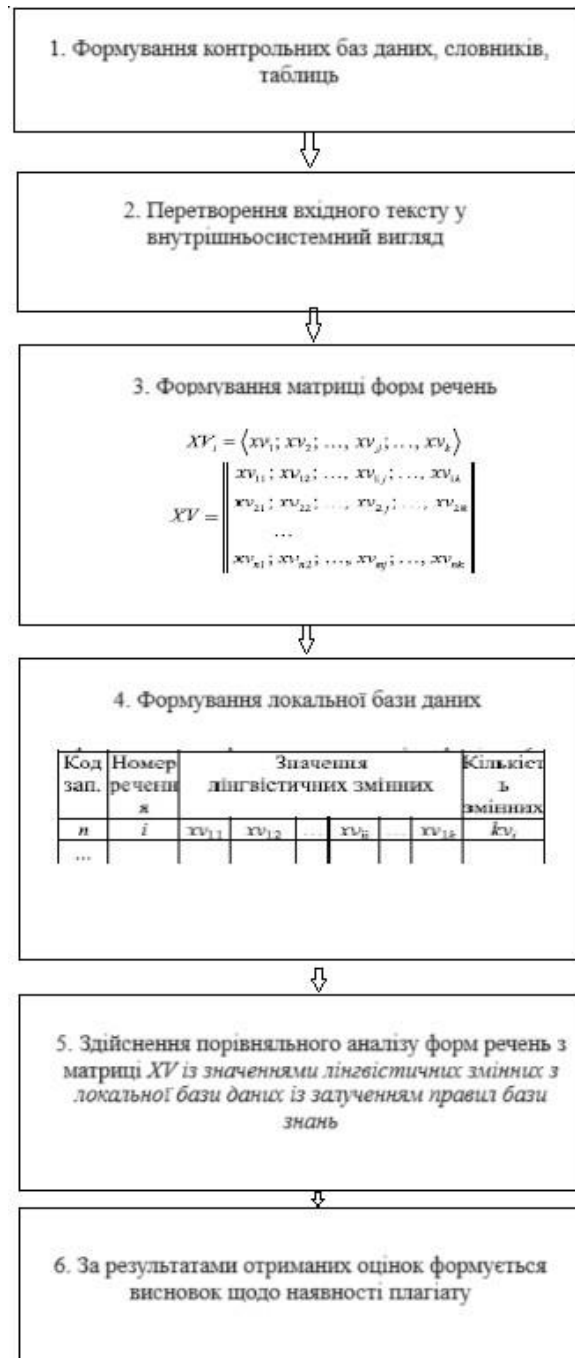


Рисунок 2.2 – Алгоритм семантичного аналізу тексту за змістом речень

2.3 Моделі синтаксичного аналізу речень

Автоматизований синтаксичний аналіз є одним з найбільш складних та актуальних напрямів теорії комп'ютерної лінгвістики.

Синтаксичні аналізатори широко застосовуються в таких областях як створення компіляторів, проектування інтерфейсів баз даних, штучний інтелект

(Ш), автоматична обробка текстів (АОТ), у тому числі, для автоматизованих інформаційно-пошукових систем (АПС), систем машинного перекладу (МП), перевірки правопису тощо.

За кілька десятків років, протягом яких розроблялися різні алгоритми автоматичного синтаксичного аналізу, не було отримано жодного алгоритму, який би задовільно вирішував питання відтворення синтаксично-семантичної структури тексту у формальних моделях для української мови.

Зважаючи на стрімке зростання обсягів текстової інформації та складної структурованості природньомовних (ПМ) текстів, аналіз текстів є актуальною проблемою.

Основною метою автоматичного синтаксичного аналізу є отримання за допомогою алгоритмів у явному вигляді синтаксичної структури речення, простого та складного (складеного). Під реченням розуміється найчастіше просте речення, в якій відсутні як складові інші прості речення, а складне речення називають фразою [22].

В даний час все частіше використовується ще одне визначення синтаксичного аналізу – це процес зіставлення лінійної послідовності лексем (слів) мови з його формальною граматикою.

Результатом є дерево розбору або синтаксичне дерево. Такий процес прийнято називати парсингом.

Розглянемо графічні моделі синтаксичної структури речення.

Існують різні способи зображення синтаксичної структури речення, серед яких можна виділити три основні та найбільш поширені: дужковий запис; зображення залежностей у вигляді стрілок, спрямованих від керуючого слова до керованого; зображення синтаксичної структури у вигляді дерева.

Розглянемо всі три способи на прикладі речення Синтаксичний аналіз – необхідний елемент штучного інтелекту.

Дужковий запис структури цього речення матиме вигляд:

((Синтаксичний аналіз) – (необхідний елемент)(штучного інтелекту))

Рисунок 2.3 – Синтаксична структура речення у вигляді дужок

В дужках розташовані слова, безпосередньо пов'язані один з одним залежністю синтаксичного характеру. Зображення синтаксичної структури у вигляді стрілок від керуючого слова до керованого можна представити так:



Рисунок 2.4 – Синтаксична структура речення у вигляді стрілок

Запис у вигляді дерева передбачає зображення структури речення у вигляді певного графу:

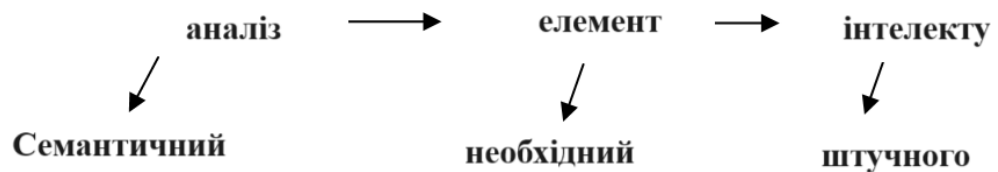


Рисунок 2.5 – Синтаксична структура речення у вигляді дерева

В українській та інших мовах дерева підпорядкування речень ділового стилю будуються, як правило, за законом проєктивності (рис. 3.4). Він полягає в тому, що всі стрілки можна провести над прямою, на якій записано речення, таким чином, що жодна з них не перетнеться і корінь не лежатиме ні під якою стрілкою [23]. У мові художньої літератури, особливо у поезії, відхилення від закону проєктивності допустимі і найчастіше слугують створенню певного художнього ефекту (рис.2.6).



Рисунок 2.6 – Порушення закону проєктивності [23]

Методи подання синтаксичної структури речення.

В даний час можна виділити два основних метода подання синтаксичної структури речення. Всі інші методи можна вважати варіаціями цих двох.

Якщо використовувати термін граматики для позначення системи зв'язків, то два найбільш популярні способи називаються граматикую безпосередньо складових (constituent grammars) та граматикую залежностей (dependency grammars).

Constituent grammars виділяє у реченні найбільш пов'язані між собою за змістом слова (рис. 2.7).

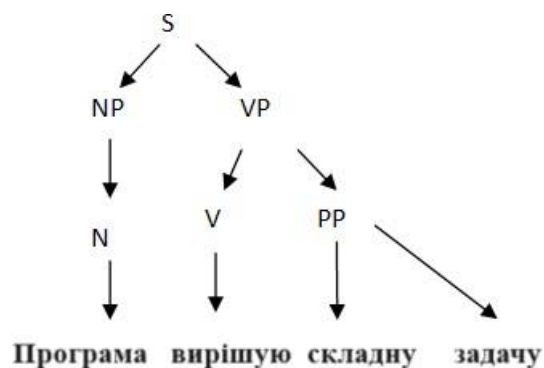


Рисунок 2.7 – Подання синтаксичної структури речення у вигляді структури складових (constituent grammars)

Dependency grammars базується на уявленні про управління, і її основним інструментом є стрілка, як спосіб позначення зв'язків між лексемами.

Constituent grammars побудована на факті просторового слідування лексем, з яких складається речення (іменника, дієслова, прикметника тощо). Тут можливі різні види зв'язків. Речення загалом розглядається як сукупність лексем, що зводяться до двох головних підгруп: підгрупа іменника та підгрупа дієслова.

Constituent grammars використовується переважно при моделюванні текстів з фіксованим порядком слів, а dependency grammars – в текстах з вільним порядком слів.

Предметна область, що розглядається, містить речення визначеної структури. Тобто перевагу щодо побудови моделі речень системи, яка виконує аналіз з точки зору текстових і смислових збігів, потрібно надати моделі constituent grammars.

2.4 Методи синтаксичного аналізу текстової інформації

Методи, засновані на вищеописаних граматиках, використовують дві основні стратегії для проведення синтаксичного аналізу:

– розбір зверху вниз (top-down parsing), коли речення як початковий елемент аналізу розкладають на лексеми та реалізують правило висновку;

– розбір знизу вгору (bottom-up parsing), коли аналізують лексеми даного речення та реалізують правило висновку.

Таким чином, розрізняють низхідні та висхідні аналізатори.

Відмінна риса низхідного аналізу – це цілеспрямованість. На кожному кроці низхідного аналізу розпізнавач формує мету – знайти висновок, що починається з деякого нетермінального символ і породжує частину вхідного рядка.

Розпізнавач намагається досягти цієї мети шляхом спрямованого перебору різноманітних можливостей.

Основна ідея низхідного аналізу наступна: починаючи процес аналізу вхідного речення (в розроблюваній системі – це назва наукової роботи) , розпізнавач виходить із припущення, що модель цього речення можна представити як constituent grammars.

Звідси витікає головна мета аналізу – побудувати дерево лексем.

При невдалому виборі стратегії мета може виявитися недосяжною, тоді потрібно спробувати застосувати інше правило.

Можливі випадки, коли для будь-якої мети всі правила призводять до невдачі.

Опис процесу завершується, коли знайдено кінцевий результат (ланцюжок схожих речень) або коли встановлено, що цього результату немає, тобто вхідне речення не є повтором речень, які вже зберігаються в системі.

Низхідний розпізнавач переглядає лексеми вхідного речення та символи правої частини, застосовуючи правило зліва-направо. Такі розпізнавачі називають лівосторонніми.

Для здійснення аналізу текстової інформації необхідно також враховувати правила граматики. Одним із способів їх врахування при аналізі лексем – це застосування синтаксичних таблиць, які будуть зберігатись в пам'яті комп'ютера.

Методи висхідного аналізу знайшли широке застосування в програмах-трансляторах. Загальна ідея висхідного аналізу для текстових речень полягає в наступному: вхідне речення розглядається як рядок лексем. Розпізнавач по кроках додає до першої лексеми наступні. Так створюються фрази. Тобто частини речення, які ми називаємо фразами і приймають участь в аналізі збігів. У більшості висхідних розпізнавачів знаходиться фраза (найчастіше, декілько фраз), яку називають основою. Основа – це фраза (сполучення лексем), яка частіше всіх з'являється першою при розборі вхідного речення. Правила формальної граматики при висхідному аналізі застосовуються не до лексем, а до фраз.

Можна застосовувати комбінацію низхідних та висхідних методів синтаксичного аналізу. Наприклад, низхідний аналіз виділяє відносно великі синтаксичні конструкції (розбір речення на фрази), кожна з яких потім аналізується докладніше методами висхідного аналізу.

2.5 Вибір програмного інструментарію для створення експертної системи

C++ є однією з найпопулярніших мов програмування, вона використовується для розробки різноманітних програм та додатків. Ця мова має свої переваги та недоліки, які варто враховувати при роботі з нею.

Однією з переваг C++ є його широке застосування. Ця мова програмування може бути використана для розробки програмного забезпечення, вбудованих систем, графічних додатків та ігор. Вона надає багато можливостей для створення ефективного та потужного коду, що робить її привабливою для розробників.

Ще одним плюсом C++ є його швидкодія. Мова володіє низькорівневими можливостями та дозволяє безпосередньо керувати пам'яттю, що дає можливість

оптимізувати роботу програми та забезпечити високу продуктивність. Це особливо важливо для додатків, які потребують швидкого виконання, наприклад, графічних програм або операцій з великими обсягами даних.

Крім того, C++ має потужну стандартну бібліотеку, яка включає в себе багато корисних функцій та класів. Вона надає готові інструменти для роботи зі строками, контейнерами, потоками введення-виведення та багатьма іншими завданнями. Використання стандартної бібліотеки спрощує розробку програм і дозволяє зосередитися на самому завданні.

Для розробки проекту кваліфікаційної роботи доцільно використовувати стандартні модулі мови C++, такі як Vector, Algorithm тощо.

Проте, C++ має й свої недоліки.

Недоліком C++ є нестабільність деяких функцій і бібліотек. Від версії до версії можуть змінюватися деякі особливості мови, що може призвести до проблем з сумісністю коду. Крім того, існує ризик використання застарілих або вразливих функцій, які можуть призвести до помилок або безпекових проблем.

Нарешті, C++ вимагає більшої кількості коду порівняно з іншими мовами програмування. Це може призвести до більшої складності розробки та підтримки коду. Крім того, недостатня обережність при використанні покажчиків та керуванні пам'яттю може призвести до втрати даних або інших проблем з продуктивністю.

Незважаючи на недоліки, C++ залишається потужною і гнучкою мовою програмування, яка може бути дуже ефективною для великих проектів та задач, які вимагають високої продуктивності.

3 ПРОГРАМНА РЕАЛІЗАЦІЯ СИСТЕМИ АКАДЕМІЧНОЇ ДОБРОЧЕСНОСТІ

3.1 Середовище розробки та стандартні модулі

Інтегроване середовище розробки (Integrated Development Environment) – це програмний продукт, що поєднує текстовий редактор, компілятор, налагоджувач та довідкову систему. Для створення проекту кваліфікаційної роботи використовувалось Інтегроване середовище розробки Microsoft Visual Studio.

Будь-яка програма, створювана серед Microsoft Visual Studio, завжди оформляється як окремий проект (project). Проект – набір взаємопов'язаних вихідних файлів, компіляція та компоновання яких дозволяють створити програму, що виконується. Зовнішній вигляд робочої області проекту (projectworkspace) представлений на рис. 3.1.

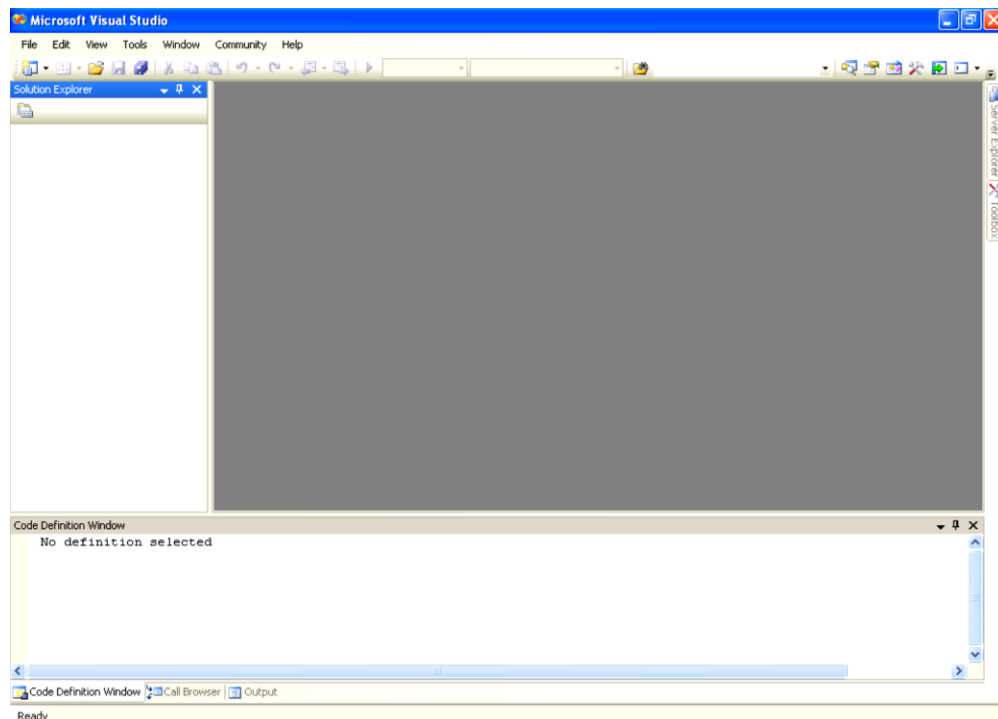


Рисунок 3.1 – Головне вікно програми Visual C++.

Робоча область проекту може містити будь-яку кількість різних проектів, згрупованих разом для узгодженої розробки: від окремої програми до бібліотеки функцій або цілого програмного продукту.

Робочий стіл VisualC++ включає три вікна:

– вікно Project Workspace (Вікно робочої області), призначене для надання допомоги при описі та супроводі великих багатофайлових програм.

– вікно Editor (вікно редактора) використовується для введення та перевірки вихідного коду.

– вікно Output (Вікно виводу) слугує для виведення повідомлень про хід компіляції, складання та виконання програми. Зокрема, повідомлення про помилки з'являються саме в цьому вікні (рис. 3.1).

Інтегроване середовище Visual C++ дозволяє будувати проекти різних типів, орієнтовані на різні сфери застосування, також передбачена робота з консольними додатками. При запуску консольної програми операційна система створює так зване консольне вікно, через яке йде все введення-виведення програми. Зовні це нагадує роботу в командному процесорі операційної системи Windows.

При створенні проекту використовувались стандартні бібліотеки мови C++.

Природньо – це бібліотека `iostream`. У ній реалізована підтримка файлового введення/виведення даних вбудованих типів. Крім того, розробники класів можуть розширювати цю бібліотеку для читання та запису нових типів даних.

Для використання бібліотеки `iostream` у програмі необхідно включити заголовний файл `iostream`.

Операції введення/виведення виконуються за допомогою класів `istream` (потокове введення) та `ostream` (потокове виведення). Третій клас, `iostream`, є похідним від них і підтримує двонаправлене введення/виведення. Для зручності в бібліотеці визначено три стандартні об'єкти-потоки:

– `cin`: об'єкт класу `istream`, що відповідає стандартному введенню. Загалом він дозволяє читати дані з терміналу користувача;

– `cout`: об'єкт класу `ostream`, який відповідає стандартному виведенню. Загалом він дозволяє виводити дані на термінал користувача;

– `cerr`: об'єкт класу `ostream`, який відповідає стандартному виведенню помилок. У цей потік ми надсилаємо повідомлення про помилки програми.

Виведення здійснюється, як правило, за допомогою перевантаженого оператора зсуву вліво (`<<`), а введення – за допомогою оператора зсуву вправо (`>>`).

Крім читання та запису з терміналу, бібліотека `iostream` підтримує читання та запис у файли. Для цього призначені такі класи:

– `ifstream`, похідний від `istream`, пов'язує введення даних в програмі із файлом;

– `ofstream`, похідний від `ostream`, пов'язує виведення даних з файлом;

– `fstream`, похідний від `iostream`, пов'язує як введення, так і виведення даних з файлом.

Щоб використовувати частину бібліотеки `iostream`, пов'язану з файловим введенням/виводом, необхідно включити до програми заголовковий файл `fstream`.

Файлове введення/виведення підтримується тими ж операторами.

Вектор (`std::vector`) та рядок (`std::string`) – це важливі базові контейнери стандартної бібліотеки C++. Вони зберігають свої елементи у безперервному фрагменті пам'яті. Обидва ці контейнери надають доступ до елементів по індексу і дозволяють ефективно додавати нові елементи в кінець.

Контейнер `std::vector` представляє собою динамічний масив, що забезпечує швидке додавання нових елементів у кінець і змінює свій розмір за потреби. Вектор гарантує відсутність витоку пам'яті. Для роботи з вектором потрібно підключити файл заголовка `vector`.

Елементи вектору мають бути однакового типу і задаються у кутових дужках після `std::vector`.

Наприклад, `std::vector<std::string>` – вектор рядків, який використовувався в розробленому проекті.

Заголовний файл `conio.h` використовується для створення текстового інтерфейсу користувача. Він не є частиною мови програмування C. Цей заголовний файл оголошує кілька бібліотечних функцій для роботи з консольним введенням та виведенням програми.

Для реалізації функцій зі змінним числом аргументів у проекті підключено заголовний файл `stdarg.h`.

В проекті використовувався також стандартний клас `map`, який підключається за допомогою однойменного модуля `map`.

Клас `map` у стандартній бібліотеці C++ має наступні властивості:

- як контейнер змінного розміру фактично витягує значення елементів на основі зв'язаних значень ключів;
- реверсивний, оскільки він надає двонаправлені ітератори для отримання доступу до його елементів;
- сортується, оскільки його елементи впорядковані за значеннями ключів відповідно до заданої функції порівняння;
- є унікальним, оскільки кожен його елемент повинен мати унікальний ключ;
- є контейнером асоціативної пари, оскільки його значення елементів залежать від його значень ключів;
- є шаблоном класу, оскільки функціональність, що надається, є універсальною та незалежною від типу елемента або ключа. Типи даних, що використовуються для елементів та ключів, визначаються як параметри у шаблоні класу разом із функцією та розподільником порівняння.

Заголовні модулі `windows.h` та `local` використовувались для організації інтерфейсу на українській мові.

3.2 Опис програмного продукту

Програмний продукт має назву LB2-master і представляє собою багатофайловий проект, розроблений в середовищі Visual Studio Community 2022.

До проекту входять наступні файли:

- main.h, файл заголовків;
- main.cpp, файл з описом головної функції main;
- ui.h, файл-помічник для відображення елементів консолі;
- ui.cpp, файл для роботи з головним меню програми;
- file.h, файл оголошення класу FileManager для читання та запису файлів;
- file.cpp, файл для роботи з записами бази даних;
- locale.h та locale.cpp, файли для підтримки українського інтерфейсу;
- models/QualifyingWork.h, файл для відображення записів із бази даних кваліфікаційних робіт;
- models/Keyword.h, файл трансформує модель у формат пари ключ-значення для відображення та запису;
- models/AnalyzeResult.h, файл для аналізу бази даних кваліфікаційних робіт;
- QualifyingManager.h та QualifyingManager.cpp, файли контролю бази даних кваліфікаційних робіт щодо додавання записів, перезапису файлу;
- KeywordsManager.h та KeywordsManager.cpp, файли для роботи з ключовими словами бази знань;
- navigation.h та navigation.cpp, файли для розташування елементів інтерфейсу;
- table.h та table.cpp, файли для організації друку в консолі бази даних кваліфікаційних робіт у вигляді таблиці;
- helpers.h та helpers.cpp, файли для підтримки кольору для відображення інформації при роботі з програмою.

Повний текст проекту надано в додатку Б.

Файл main.h підключає до проекту файли з розширенням .h, а саме ui.h, file.h, locale.h, models/QualifyingWork.h, models/Keyword.h, models/AnalyzeResult.h, QualifyingManager.h, KeywordsManager.h. Він починається з директиви #pragma once. Це нестандартна, але часто вживана препроцесорна директива, яка контролює те, щоб конкретний вихідний файл під час компіляції підключався тільки один раз (рис.3.2).

```
#pragma once

#include "ui.h"
#include "file.h"
#include "locale.h"
#include "models/QualifyingWork.h"
#include "models/Keyword.h"
#include "models/AnalyzeResult.h"
#include "QualifyingManager.h"
#include "KeywordsManager.h"

using namespace std;

int main();
```

Рисунок 3.2 – Зміст файлу main.h

Файли main.h та main.cpp є основними керуючими модулями проекту, які відповідають за підключення файлу бази даних кваліфікаційних робіт db.txt та файлу бази знань keywords.txt, де зберігаються

```
#include "main.h"

int main(){
    setLanguageSupport();

    UI::Controller ui;
    Controllers::QualifyingManager qfManager("db.txt");
    Controllers::KeywordsManager keywords("keywords.txt");
```

Рисунок 3.3 – Фрагмент файлу main.cpp

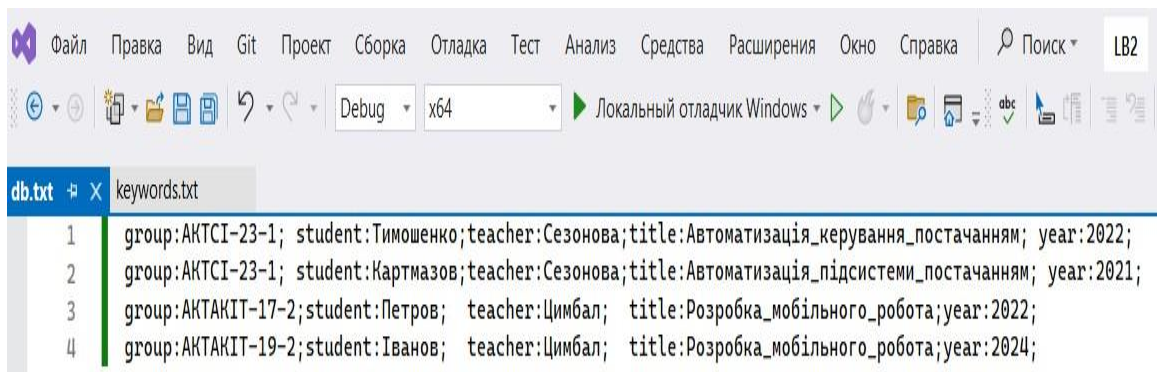
Файли `ui.h` та `ui.cpp` відповідають за користувацький інтерфейс, який реалізовано через меню.

Пункти меню, які реалізують всі можливості програмного продукту, наступні:

- аналіз назви кваліфікаційної роботи;
- поповнити базу даних;
- перегляд бази даних кафедри;
- додати ключові слова;
- пошук робіт за прізвищем автора;
- пошук робіт за прізвищем керівника.

Файл `ui.cpp` забезпечує підключення функціоналу, який організує роботу кожного пункту меню. За допомогою оператора вибору `switch` відбувається вибір дій в залежності від обраного користувачем пункту меню.

Файлами, на основі яких працює програмний продукт `LB2-master` – це текстові файли `db.txt` та `keywords.txt`. Файл `db.txt` складається із записів, які поділені на поля (рис. 3.4).



```

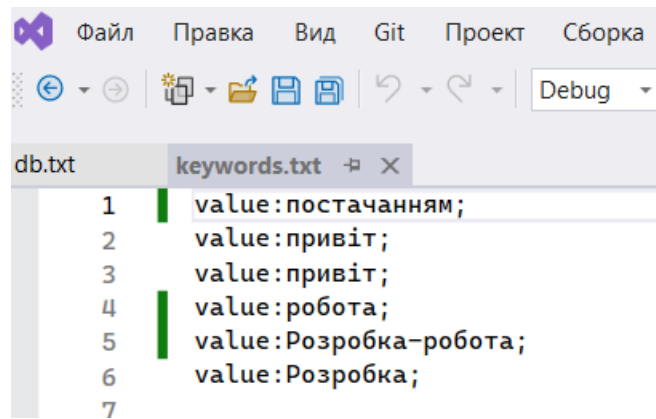
1 group:АКТСИ-23-1; student:Тимошенко;teacher:Сезонова;title:Автоматизация_керування_постачанням; year:2022;
2 group:АКТСИ-23-1; student:Картмазов;teacher:Сезонова;title:Автоматизация_підсистеми_постачанням; year:2021;
3 group:АКТАКІТ-17-2;student:Петров; teacher:Цимбал; title:Розробка_мобільного_робота;year:2022;
4 group:АКТАКІТ-19-2;student:Іванов; teacher:Цимбал; title:Розробка_мобільного_робота;year:2024;

```

Рисунок 3.4 – Записи у файлі `db.txt`

Шифр академічної групи записується після ключового слова `group` та відділяється від значення поля двокрапкою. За аналогічним принципом заповнюються поля `student`, `teacher`, `title`, `year` відповідно прізвищем студента, прізвищем наукового керівника, назвою кваліфікаційної роботи та роком її захисту.

Робота алгоритмів аналізу базується на переліку слів та словосполучень, які є основою проектованої бази знань. Приклад заповнення файлу keywords.txt, який зберігає всі слова та словосполучення, за якими проводиться аналіз файлу db.txt, зображено на рис. 3.5.



```
1 value:постачання;  
2 value:привіт;  
3 value:привіт;  
4 value:робота;  
5 value:Розробка-робота;  
6 value:Розробка;  
7
```

Рисунок 3.5 – Приклад файлу keywords.txt

В файлі keywords.txt ключові слова записуються після ключового слова value та двокрапки. Якщо потрібно записати словосполучення, використовується роздільник тире між окремими словами словосполучення.

3.3 Керівництво користувача

Проект LB2-master представляє собою консольний додаток, розроблений в середовище VS.

Після запуску проекту на консолі з'являється меню, вигляд якого відображено на рис. 3.6.

За допомогою пункту меню Аналіз назви кваліфікаційної роботи можна здійснити аналіз назв кваліфікаційних робіт, які зберігаються в файлі db.txt за ключовими словами та словосполученнями.

Для переходу до пунктів меню потрібно натиснути клавішу Enter. Для навігації по меню використовуються клавіші зі стрілочками вгору та вниз.

Активний в даний момент часу пункт меню підсвічується червоним кольором, що додає певної зручності для використання програми.

```

Головне меню
[Стрілки вгору - вниз] - Навігація по меню [Enter] - Перехід до розділу меню

|Аналіз назви кваліфікаційної роботи
|Поповнити базу даних
|Перегляд бази даних кафедри
|Додати ключові слова
|Пошук робіт за прізвищем автора
|Пошук робіт за прізвищем керівника
  
```

Рисунок 3.6 – Головне меню програми LB2-master

Пункт меню Поповнити базу даних слугує додаванню до бази даних назв кваліфікаційних робіт нову назву роботи з консолі (рис. 3.7). Для повернення до головного меню потрібно натиснути клавішу ESC. При потребі додавання з консолі ще одного запису до бази даних використовується клавіша Enter. Кількість таких маніпуляцій не обмежена.

```

Поповнити базу даних

Вкажіть ПІБ керівника: Бронніков
Вкажіть ПІБ здобувача: Забора
Вкажіть академічну групу: АКТСІ-18-1
Вкажіть тему кваліфікаційної роботи: Розробка_функціональної_моделі_маніпулятора
Вкажіть рік: 2022

Запис створено!
Натисність [Enter] - щоб створити новий запис, або [ESC] - щоб повернутись в головне меню
  
```

Рисунок 3.7 – Діалогове вікно для додавання в базу даних нового запису з консолі

Для перегляду вмісту бази назв кваліфікаційних робіт слугує пункт меню Перегляд бази даних кафедри. Формат, в якому відображається інформація про наявні в базі даних записи, відображений на рис. 3.8.

Перегляд бази даних кафедри
 [ESC] – Повернутись до головного меню

ПІБ керівника	ПІБ автора	Тема роботи	Академічна група	Рік
Сезонова	Тимошенко	Автоматизація_керування_постачанням	АКТСІ-23-1	2022
Сезонова	Левенець	Автоматизація_підсистеми_постачанням	АКТСІ-23-1	2021
Цимбал	Петров	Розробка_мобільного_робота	АКТАКІТ-17-2	2022
Цимбал	Іванов	Розробка_мобільного_робота	АКТАКІТ-19-2	2024
Бронніков	Забора	Розробка_функціональної_моделі_маніпулятора	АКТСІ-18-1	2022

Рисунок 3.8 – Результат виклику пункту меню Перегляд бази даних кафедри

Після натискання пункту меню Додати ключові слова користувач може поповнити файл keywords.txt з консолі. Приклад додавання до переліку ключових слів слова маніпулятор відображено на рис. 3.9.

```

Додати ключові слова

Вкажіть ключове слово яке потрібно додати: маніпулятор

Запис створено!
Натисніть [Enter] – щоб створити новий запис, або [ESC] – щоб повернутись в головне меню
  
```

Рисунок 3.9 – Діалогове вікно для поповнення бази знань ключовими словами або словосполученнями

Підказки щодо навігації між пунктами меню присутні на консолі постійно. Для повернення до головного меню натискається клавіша ESC. Програма надає можливість надавати з консолі будь-яку кількість ключових слів або словосполучень, що забезпечує гнучкість та багатоваріантність пошуку подібних назв кваліфікаційних робіт. Кожний новий запис створюється після натискання клавіші Enter.

Після виклику пункту меню Пошук робіт за прізвищем автора відкривається діалогове вікно, в якому потрібно натиснути клавішу Enter та ввести з консолі прізвище студента, роботи якого потрібно знайти (рис.3.10).

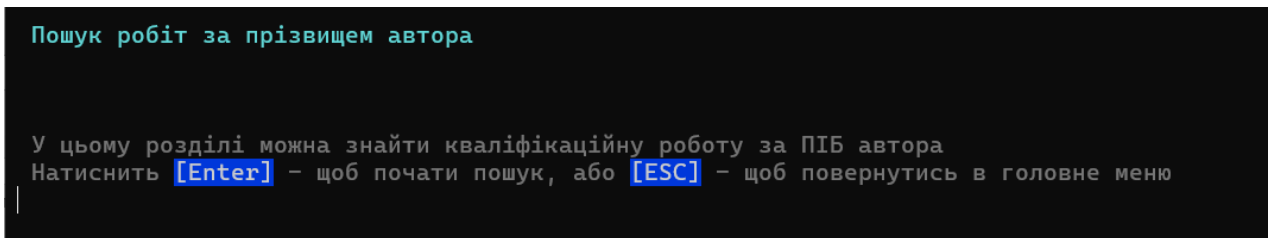


Рисунок 3.10 – Діалогове вікно пункту Пошук робіт за прізвищем автора

Після введення прізвища студента у відповідне поле після запрошення та натискання клавіши Enter на консоль буде виведено всі роботи із введеним прізвищем, які зберігаються в базі даних (рис. 3.11).

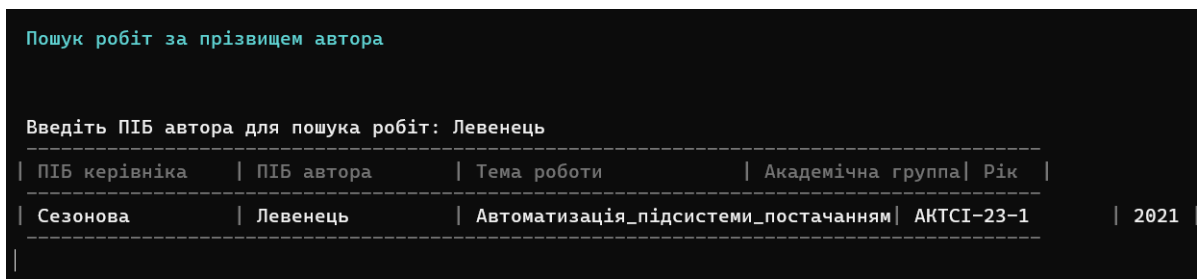


Рисунок 3.11 – Результат пошуку записів щодо кваліфікаційних робіт по прізвищу студента

Аналогічним чином працює пункт головного меню Пошук робіт за прізвищем керівника. Приклад результату роботи цього пункту меню після введення прізвища керівника відображено на рис. 3.12.

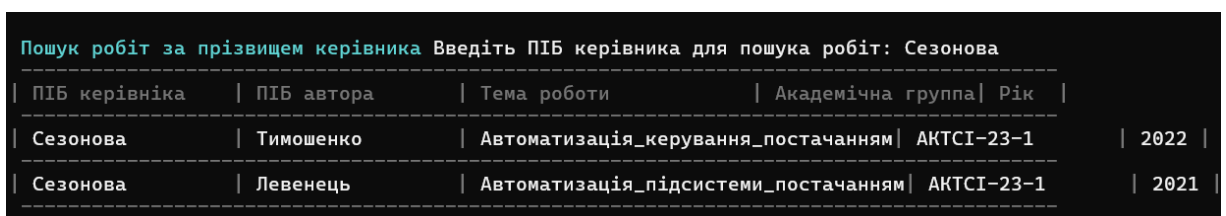


Рисунок 3.12 – Результат роботи пункту меню Пошук робіт за прізвищем керівника

Загальний аналіз тем кваліфікаційних робіт кафедри з підключення бази знань здійснюється за допомогою пункту меню Аналіз назви кваліфікаційної

роботи. Результат аналізу тем кваліфікаційних робіт (тестовий варіант) відображено на рис. 3.13. В таблиці висновку щодо аналізу по ключовим словам буде зазначено теми, результат пошуку (Знайдено або Не знайдено), прізвища студента та керівника. В останньому стовпчику буде зазначене ключове слово пошуку.

```

Аналіз назви кваліфікаційної роботи
[ESC] - Повернутись до головного меню
  
```

Тема роботи	Результат	ПІБ керівника	ПІБ автора	Ключове слово
Автоматизація_керування_постачанням	Знайдено	Сезонова	Тимошенко	постачанням
Автоматизація_підсистеми_постачанням	Знайдено	Сезонова	Левенець	постачанням
Розробка_мобільного_робота	Знайдено	Цимбал	Петров	Розробка
Розробка_мобільного_робота	Знайдено	Цимбал	Іванов	Розробка
Розробка_функціональної_моделі_маніпулятора	Знайдено	Бронніков	Забора	маніпулятор

Рисунок 3.13 – Результат роботи алгоритму пошуку проекту LB2-master

Розроблений проект є унікальним в класі програм, які призначені для виявлення плагіату.

В ньому вбачаються певні недоліки, а саме:

- відсутність модуля для автоматичного перетворення вхідних даних у вигляді наказу (Додаток А) в придатний для поповнення бази даних проекту формат;

- відсутність програмного аналізу ключових слів та словосполучень.

Сьогодні реалізовано тільки введення ключових слів з клавіатури або безпосередньо заповнення текстового файлу.

4 ОХОРОНА ПРАЦІ

4.1 Промислова безпека та аналіз умов праці

Розміри лабораторії, в якій виконувалась робота, складають 5 м × 6 м. Робоче місце складається із стола, стільця і персонального комп'ютера. У приміщенні працює 4 людини. Площа приміщення 30 м², об'єм – 90 м³. Згідно ДСанПиН 3.3.2.007-98 площа на одне робоче місце має становити не менше 6 м², а об'єм – 20 м³ [24]. Для даного приміщення робоча площа і об'єм на одного людини відповідає нормам, так як в нашому випадку площа на одне робоче місце становить 10 м², а об'єм – 30 м³.

Живлення комп'ютерів здійснюється від трифазної чотирьох провідної електричної мережі змінного струму з глухо-заземленою нейтраллю і напругою 220 В, частотою 50 Гц.

Згідно НПАОП 40.1-1.21-98 лабораторію можна віднести до категорії без підвищеної небезпеки, так як в приміщенні відсутні чинники, які викликають підвищену або особливу небезпеку [24].

4.2 Виробнича санітарія

Робота в лабораторії проводиться сидячи і не вимагає фізичної напруги. Тому вона відноситься до категорії Ia (легкі фізичні роботи, енерговитрати до 120 ккал/год). З метою забезпечити комфортні умови для працівників та відповідно до ДСН 3.3.6.042-99 у відвідуванні встановлені наступні метеорологічні параметри:

а) для холодного періоду:

- температура повітря від 22 °С до 24 °С;
- вологість повітря від 40 % до 60 %;
- швидкість руху повітря оптимальна до 0,1 м/с;

б) для теплого періоду року:

- температура повітря від 23 °С до 25 °С;
- вологість повітря від 40 % до 60 %;
- швидкість руху повітря оптимальна до 0,1 м/с [25].

Для освітлення робочих місць і приміщення в цілому застосовується як природне бічне освітлення, так і штучне освітлення.

Приміщення з ПК повинні мати природне і штучне освітлення відповідно до ДБН В.25-28-2006 «Природне і штучне освітлення». Природне світло повинно проникати через бічні світлові прорізи, зорієнтовані, як правило, на північ або північний схід, і забезпечувати коефіцієнт природної освітленості (КПО) не нижче 1,5 %: $e^{IV} = 1,35$, де $e^{IV}_{\text{норм}}$ – нормоване значення КПО для 4-го поясу світлового клімату України [25].

Згідно ДСН 3.3.6.037-99 рівень шуму в лабораторії не перевищує 50 дБ.

Загальний рівень штучного освітлення приміщення можна перевірити за допомогою методу питомої потужності.

Розрахункова формула методу:

$$W = \frac{W_{\Sigma}}{S}, \quad (4.1)$$

де W – питома потужність, Вт/м²;

S – площа приміщення, м²;

W_{Σ} – загальна потужність освітлювальної установки, Вт, яка розраховується за формулою

$$W_{\Sigma} = W_{cb} \cdot n_{cb}, \quad (4.2)$$

де W_{cb} – потужність одного світильника, $W_{cb} = 80$ Вт;

n_{cb} – кількість світильників у приміщенні, $n_{cb} = 4$.

Дане приміщення має площу 30 м², в якому розташовано шість світильників потужністю 80 Вт.

$$W_{\Sigma} = 4 \cdot 80 = 320 \text{ (Вт)}, \quad (4.3)$$

$$W = \frac{320}{30} = 11 \text{ (Вт/м}^2\text{)}.$$

Табличне значення для отриманого результат освітленість складе 200 лк, коли відповідно до стандарту ДБН В.2.5-28-2006. в лабораторії освітленість повинна бути від 300 лк до 500 лк. Для отримання освітленості в 400лк необхідна питома потужність 21 Вт/м² [25].

Для поліпшення умов роботи в лабораторії необхідно в денний час застосовувати додаткове освітлення.

Приміщення для роботи з ПК потребують виконання звичайних ергономічних вимог до робочих місць та підтримки мікроклімату без наявних особливостей.

ВИСНОВКИ

В ході виконання роботи було проведено аналіз наукової літератури за темою кваліфікаційної роботи, проведено аналіз існуючих програмних засобів перевірки наукових та кваліфікаційних робіт на наявність плагіату, які використовуються в освітньо-науковій сфері діяльності. Зокрема зроблено аналіз програм перевірки на наявність плагіату в кваліфікаційних роботах, які подаються до захисту у вищих навчальних закладах.

Були виконані наступні завдання:

– розглянуто формалізми, моделі та методи, що застосовуються для синтаксичного та семантичного аналізу при перевірці текстів. Виділено серед них найбільш перспективні;

– обрано методи та моделі для реалізації проекту;

– проаналізована особливості структури текстових документів, які подаються на перевірку;

– розроблено алгоритми лінгвістичного аналізу, придатні до автоматизації пошуку збігів в реченнях;

– здійснено вибір засобу для програмної реалізації проекту.

В результаті виконання кваліфікаційної роботи вирішені наступні задачі:

– розроблено формат та створено базу даних тем кваліфікаційних робіт шляхом обробки вхідних документів;

– створено програмний продукт для прийняття рішень щодо наявності академічного плагіату.

ПЕРЕЛІК ПОСИЛАНЬ

1. Цілі сталого розвитку [Електронний ресурс]. Режим доступу: <https://business.diia.gov.ua/handbook/sustainable-development-goals/cili-stalogo-rozvitku>
2. ДСТУ 3008-15. Документація. Звіти у сфері науки та техніки. структура та правила оформлення. Введ. 2015-06-22. К. Держстандарт України, 2017. – 29 с.
3. Методичні вказівки з підготовки кваліфікаційної роботи бакалавра для студентів усіх форм навчання спеціальності 151 «Автоматизація та комп'ютерно-інтегровані технології» освітньої програми «Автоматизація та комп'ютерно-інтегровані технології» / Упоряд.: І.Ш. Невлюдов, А.О. Андрусевич, О.В. Токарева, С.П. Новоселов, О.В Сичова. Харків: ХНУРЕ, 2022. – 55 с.
4. Харківський національний університет радіоелектроніки. [Електронний ресурс]. Режим доступу: <https://nure.ua/department/kafedra-kompyuterno-integrovanih-tehnologiy-avtomatizatsiyi-ta-mehatroniki-kitam>.
5. SAIUP (Srtengthening Academic Integrity in Ukraine Project). [Електронний ресурс]. Режим доступу: <https://saiup.org.ua>
6. Вергун А. Академічна доброчесність з позицій антиплагіатної експертизи: ключові принципи та їх імплементація / А. Вергун, С. Ягело // Український педагогічний журнал. – 2021. – № 1. – С. 100–108
7. Розширений глосарій термінів та понять ст. 42 «Академічна доброчесність» Закону України «Про освіту» від 05.09.2017/ Проект сприяння академічної доброчесності в Україні – SAIUP // Міністерство освіти і науки України. [Електронний ресурс]. Режим доступу: <https://bit.ly/30Aamj3>
8. Про вищу освіту: Закон України від 01.07.2014 № 1556-VII [Електронний ресурс]. Режим доступу: <http://zakon2.rada.gov.ua/laws/show/1556-18>

9. Великий тлумачний словник сучасної української мови. – К.: ВТФ «Перун», 2005. – 977 с.
10. Plagiarism [Електронний ресурс]. Режим доступу: <https://legal-dictionary.thefreedictionary.com/plagiarism>
11. What is Plagiarism? [Електронний ресурс]. Режим доступу: <http://plagiarism.org/article/what-is-plagiarism>
12. Researcher Insights into the Types of Plagiarism & Attribution Issues [Електронний ресурс]. Режим доступу: <http://www.ithenticate.com/resources/papers/decoding-plagiarism>
13. Федорчук Є.Н. Програмування систем штучного інтелекту. Експертні системи. – Львів: Вид-во Львівської політехніки, 2022. – 168 с.
14. Баклан І.В. Експертні системи. Курс лекцій /Навчальний посібник. – К.: НАУ, 2022. – 132с.
15. Ali Alkhatlan and Jugal K. Kalita. 2018. Intelligent Tutoring Systems: A Comprehensive Historical Survey with Recent Developments. 1, 1 (December 2018), 31 pages. [Електронний ресурс]. Режим доступу: <https://arxiv.org/pdf/1812.09628>
16. A. K. Das, «Best Adaptive Learning Platforms for Professional Development». [Електронний ресурс]. Режим доступу: <https://indvstrvs.com/best-adaptive-learning-platforms-2018/>
17. Чотири тенденції в сучасному навчанні. [Електронний ресурс]. Режим доступу: <http://arenanews.com.ua/novosti-partnerov/6610-4-tendencyi-v-suchasnomu-navchann.html>
18. Дем'яненко В. М. Модель адаптивної навчальної системи інформаційного простору відкритої освіти // Інформаційні технології і засоби навчання. 2020. – Т. 77. – №3. – С.27-36
19. Інструкція для адміністратора по використанню системи Strikeplagiarism.com. [Електронний ресурс]. Режим доступу: https://strikeplagiarism.com/uk/assets/files/_Admin_manuals_Strikeplagiarism_com_UA_.pdf
20. Рекомендації з оформлення посилань в наукових роботах.

[Електронний ресурс]. Режим доступу:
<https://ela.kpi.ua/server/api/core/bitstreams/96056837-129b-4845-88d0-49a7d963e6b7/content>

21. Бондаренко М. Ф., Осика А. Ф. Автоматична обробка інформації на природній мові. - К.:УМК ВО,1991.-142 с.

22. Bondarenko M.F., Terziyan V. Reasoning with Metasemantic Network// Proc. of the First All-Ukrainian Conf. on Signal-Image Processing and Pattern Recognition. – Kiev (Ukraine). – 1992. – P.11.

23. Сезонова І., Левенець І. Розробка автоматизованої система підтримки академічної доброчесності / XXVIII Міжнародна науково-практична конференція PROSPECTS OF INNOVATIVE DEVELOPMENT IN SCIENCE AND TECHNOLOGY, 19-21 червня 2024, Гетеборг, Швеція. Режим доступу: <https://isu-conference.com/prospects-of-innovative-development-in-science-and-technology/>

24. ДСН 3.3.6.042-99. Санітарні норми мікроклімату виробничих приміщень / Постанова Міністерства охорони здоров'я України від 01.12.1999 № 42 [Електронний ресурс]. Режим доступу: <https://zakon.rada.gov.ua/rada/card/va042282-99>.

25. Комплекс навчально-методичного забезпечення навчальної дисципліни «Організація керування умовами праці» підготовки освітнього рівня бакалавр усіх спеціальностей та усіх напрямів університету / ХНУРЕ; розроб.: Т. Є. Стиценко, Г. В. Пронюк, Н. М. Сердюк. – Харків, 2017. – 108 с. [Електронний ресурс]. Режим доступу: <https://catalogue.nure.ua/document=218933>