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

Факультет комп'ютерно-штформаційних 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: Розробка сервера пошукового ядра Антиплагіат

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

	(підпис)	Зелений В. В. (прізвище та ініціали)
Керівник	(підпис)	Петрим М. Р. (прізвище та ініціали)
Нормоконтроль	(підпис)	Бойко І. В. (прізвище та ініціали)
Завідувач кафедри	(підпис)	Петрим М. Р (прізвище та ініціали)
Рецензент	(підпис)	Приймак М. В. (прізвище та ініціали)

Тернопіль
2020

РЕФЕРАТ / ABSTRACT

Атестаційна робота магістра містить: 41с., 7 рис.,1 табл., 16 джерел

C#, ASP .NET, VISUAL STUDIO, ВИЯВЛЕННЯ ПАГІАТУ, АЛГОРИМИ, REST.

Метою роботи є аналіз алгоритмів по виявленню плагіату та побудови на їх основі ядра сервера системи Атиплагіат.

Методи розробки базуються на інструментах розробки веб-застосувань на платформі C#, протоколу передачі даних HTTP, засобу автоматизованої роботи з програмними проектами Visual Studio для розробки веб-застосунків на платформі ASP.Net.

В результаті роботи було проведено дослідження для порівняння деяких існуючих алгоритмів для виявлення плагіату. На основі результату експериментів було обрано алгоритм для знаходження плагіату в текстах, і на основі нього розроблено програмний продукт.

C#, ASP .NET, VISUAL STUDIO, PLAGIARISM DETECTING, ALGORITHMS, REST.

The object of the work is to analyze the algorithms for detecting plagiarism and building on their basis the server core of the Atyplagiat system.

Development methods are based on tools for developing web applications on the S # platform, HTTP data transfer protocol, a tool for automated work with Visual Studio software projects for developing web applications on the ASP.Net platform.

As a result, a study was conducted to compare other existing algorithms for detecting plagiarism. Based on the results of the experiment, an algorithm was created to find plagiarism in the texts, as well as on the basis of the proposed software product.

Зміст

ВСТУП

З тих пір, як ми вступили в еру цифрового спілкування, простота обміну інформацією через Інтернет заохочувала пошук літератури в Інтернеті. З цим потенційний ризик зростання академічних порушень та інтелектуальної власності також зріс. У міру зростання занепокоєння щодо плагіату все більше уваги приділяється автоматичному виявленню плагіату. Це обчислювальний підхід, який допомагає людині при оцінці того, чи фрагменти в текстів є плагіатом. Однак більшість існуючих підходів до виявлення плагіату обмежуються лише поверхневими, грубими методами перевірки рядків. Якщо текст зазнав істотної семантичних та синтаксисчних зміни, підходи до узгодження рядків не працюють добре. З метою виявлення таких зміни, лінгвістичні прийоми, здатні провести глибший аналіз - необхідні. На сьогоднішній день на цю тему проведено дуже обмежені дослідження використання лінгвістичних методів при виявленні плагіату.

Обмеження доступу до знань та інформації неможливо. Таким чином для забезпечення академічної доброчесності та якість дослідницької роботи, ефективна система виявлення в цьому необхідна. Плагіат поділяється на текстовий плагіат та плагіат вихідного коду. Плагіат у вихідному коді або загалом називають програмним плагіатом, це такий де сегменти коду копіюються. Методи виявлення цих двох плагіатів абсолютно різні, оскільки програмного плагіату більше обмежений. Іншими словами, тут фокус зміщується на мова, що використовується, набір ключових слів, структура кодування тощо. У текстах, з іншого боку, плагіат поширюється на різні можливості і складність обробки і навіть може траплятися міжмовний плагіат. Ця робота зосереджена на вивченні та аналізі тексту плагіат.

Ця дипломна робота пропонує нові перспективи щодо виявлення плагіату. Гіпотеза полягає в тому, що оригінальні тексти та переписані тексти мають суттєві, але вимірювані відмінності, і що ці відмінності можна охопити за

допомогою статистичних та лінгвістичних показників. Щоб дослідити цю гіпотезу, визначено основні цілі дослідження.

По-перше, пропонується нова основа для виявлення плагіату. Він включає використання методів обробки природної мови, а не лише покладання на традиційні підходи до узгодження рядків. Завдання полягає у дослідженні та оцінці впливу попередньої обробки тексту, а також статистичної, поверхневої та глибокої лінгвістичної методики. Це досягається шляхом оцінки техніки в двох основних експериментальних умовах.

По-друге, досліджується перспектива застосування запропонованих рамок у масштабному сценарії. Завдання полягає у дослідженні масштабованості запропонованого алгоритми. Це досягається експериментами з великими масштабами. Перші два етапи базуються на більшій довжині тексту та заключний етап базується на фрагментах текстів.

Нарешті, розробимо додаток що допоможе розрізняти плагіат в текстах, використовуючи набуті знання. Статистична та лінгвістична ознаки досліджуються індивідуально або в різних поєднаннях. Завдання полягає у представлені нового погляду на традиційне порівняння з використанням грубої сили.

Висновки цього дослідження пропонують ідеї щодо подальших напрямків дослідження та потенційні програми для вирішення проблем, що стоять перед виявленням плагіату в текстах.

1. Огляд предметної області роботи

1.1. Визначення плагіату

Плагіат, це акт подачі чужих слів та ідей як власний, розглядається як моральний злочин, а часто і як правопорушення. Плагіат має давній корінь, оскільки саме слово походить від латинських слів “plagiaries”, що означає викрадач, і “plagiare”, що означає красти. Визначення плагіату у словнику: „ Дія чи практика прийняття чужої роботи, ідеї, тощо, і видаючи його як своє; літературна крадіжка ". Плагіат став основною проблемою з моменту встановлення оцінки освіти. З моменту вступу в епоху інтернету, швидкого та простого доступу до інформація, ця проблема ще більше загострилася.

Плагіат існує в багатьох різних сферах, і його часто важко довести або вирішити. З сучасної освітньої точки зору, поширення Інтернету як платформи обміну інформацією надала студентам більше способів доступу до електронних матеріалів.

Всупереч поширеній думці, не лише студенти стикаються з данною ретельністю. Окрім звинувачень в академічних проступках, плагіат може також спричинити фінансові втрати та втрати репутації. Був ряд скандалів, коли гучні автори потрапляли під плагіат у видавничій галузі, а інші, де навіть міністри уряду були спіймані на плагіаті під час кандидатських дисертацій. Є також були випадки, коли вчені повторно використовували великі частини тексту для пропозицій щодо фінансування. Для прикладу у 2011 році було виявлено, що докторська дисертація міністра оборони Німеччини складалася з великої кількості зплагіачених текстів. Протягом тижня його докторська ступінь була скасовано, і його було усунуто від своєї ролі.

У міру того, як дедалі більше інформації стає доступною в Інтернеті, сама кількість інформації для ручного розслідування стає непосильною. Отже, були введені обчислювальні методи, які допомагають в ідентифікації текстів що були використані повторно. Саме тут автоматичне виявлення плагіату почало

привертати увагу, оскільки воно може запропонувати дієві рішення за з меншими витратами, ніж використання людських ресурсів.

1.2. Ранні методи виявлення плагіату

У перші дні плагіат можна було виявити лише вручну, покладаючись на власні знання читачів. Як пізнання різниться від людини до людини, так і величезна кількість матеріалів неможливо досягти, процес виявлення плагіату в тексті може бути складним завданням. У більшості випадків плагіат виявляють за допомогою читання текст, який викликає “D´ej`a vu” у читачі, де читач розпізнав це. Очевидним недоліком ручного методу є те, що коли сума інформація збільшується, читач рідше зможе виявити подібність, оскільки мозок людини не функціонує як жорсткий диск комп'ютера, де інформація легко доступний за запитом.

Один із найперших методів виявлення плагіату був введений Bird (1927), який досліджував застосування статистичних методів при виявленні плагіату відповідей із множинним вибором. Пізніше, схожі методи розроблялися в 1960-х роках, вони були зосереджені на виявленні плагіату в тестах з множинним вибором. Системи раннього виявлення плагіату для письмових текстів почали з’являтися приблизно в 1990-х роках. Ці інструменти використовували статистичні методи для обчислення подібності між текстами, і більшість інструментів зосереджувались на плагіаті письмового тексту, тоді як деякі фокусувались лише на вихідному коді комп'ютера.

В останнє десятиліття комерційні системи процвітали. У 2000 році їх було створено лише п’ять системи, чотири з яких використовувались для виявлення плагіату письмового тексту та одну для виявлення плагіату вихідного коду (Lathrop and Foss, 2000). Десятиліття по тому, у 2010 р. було відзначено 47 систем. Це істотне зростання свідчить про те, що з плагіатом не вдалося ефективно

боротися, отже, багато інструментів були розроблені для задоволення збільшення ринкового попиту.

Використання систем виявлення плагіату стало стандартною практикою у багатьох вищих навчальних закладах. У Великобританії було багато університетів за рекомендацією Спільного комітету з інформаційних систем (JISC) прийняти онлайн-сервіс Turnitin. Він забезпечує перевірку схожості з власною базою даних, яка містить архіви всіх раніше поданих студентських робіт та доступ до Інтернет журналів та книг. Алгоритми виявлення подібності тексту, що використовуються в комерційних системах - це комерційна таємниця, але прості тестові приклади, що містять певний рівень перефразування та структурні зміни показали, що можна обійти виявлення.

Неадекватність існуючих систем спричинила дослідження щодо виявлення плагіату. Існують різні підходи до виявлення плагіату, і вони зазвичай складаються з трьох основних етапів: попередня обробка тексту, фільтрація та виявлення. Однак існуючі підходи здебільшого обмежуються точним порівнянням між підозрілими плагіатних текстів та потенційними вихідними текстами на рівні символів або рядків. Точність цих підходів ще не досягла задовільного рівня та плагіат продовжує впливати на багато областей, особливо у галузі освіти та видавничої справи.

Найбільшою проблемою в області виявлення плагіату є те, що більшість підходів є недостатніми для виявлення текстів із суттєвими семантичними та синтаксичними змінами. Для людини легко зрозуміти тексти, що мають навіть подібне значення коли вони переписуються з використанням різних слів і структур. Однак комп’ютери не можуть розуміти тексти подібним чином, особливо в автоматичному режимі, виявлення залежить від точної відповідності тексту. Можливе рішення цього виклику - у дослідницькій галузі обчислювальної лінгвістики, яка забезпечує методики для глибшого лінгвістичного аналізу. Застосування таких методів досі залишається недостатньо дослідженим в області виявлення плагіату. Для того, щоб пролити світло на існуючі підходи до

виявлення плагіату, ця робота пропонує використовувати лінгвістичних прийомів для дослідження глибшого значення тексту для виявлення плагіату.

1.3. Проблема плагіату в сучасному суспільстві

Плагіат не вважається проблемою в якій можна провести чітку межу, в цьому питанні виникає багато сірих зон. Дослідження констатували, що поняття плагіату є розмитим і дуже важко дати чітке визначення.

У сучасних термінах на визначення плагіату значною мірою впливає людська суб'єктивність, і вона іноді розмита іншими проблемами, такими як крадіжка інтелектуальної власності, порушення авторських прав та повторне використання тексту в таких сферах, як журналістика. У деяких випадках повторне використання власних матеріалів розцінюється як порушення авторських прав, який також відомий як самоплагіат.

Технічно, плагіат визначають як послідовність n-грамів слова з одного документа, що відображається в іншому документі як послідовні слова, або така ж послідовність слів замінені їх синонімами. Однак це визначення не охоплює випадків, коли порядки слів змінюються.

Це викликає необхідність відповісти на питання про те, “які атрибути відповідають плагіату? ". У нашому дослідницькому контексті, з метою пропонування підходів до виявлення плагіату, ми визначаємо випадок плагіату наступним чином:

- Плагіат має послідовність слів, також відому як словесні n-грами, які були або безпосередньо скопійовані, або перефразовані з одного джерела в інше.
- Плагіат може мати різну довжину; плагіат може існувати в цілому документі або в межах сегментів документа.

Оскільки фокус цієї роботи не полягає у визначенні та обґрунтуванні межі між плагіатом та інші вищезазначені проблемами, наведені вище визначення

повинні бути достатніми для встановлення специфікації експериментів, описаних в наступних розділах.

1.4. Типи плагіату

Плагіат буває у багатьох формах. Це може статися в будь-якій галузі, яка передбачає процес створення, що включає письмовий текст, комп'ютерний вихідний код, мистецтво та дизайн, і навіть музичні твори. Оскільки ця робота зосереджена лише на письмовому тексті, деталі про інші типи плагіату буде лише коротко згадуватися.

Типи плагіату, про які йшлося в попередніх дослідженнях головним чином:

- Тест з декількома варіантами вибору.
- Вихідний код написаний мовами програмування.
- Письмовий текст

Плагіат у тестах із декількома варіантами та вихідному коді сильно відрізняється від плагіату в письмовому тексті. Визначення плагіату в тестах з множинним вибором покладається на статистичні підходи, при яких кількість відповідних неправильних відповідей між двома тестами порівнюється із нормальним розподілом подібних неправильних відповідей у колекції. З іншого боку, для виявлення плагіату вихідного коду потрібні різні інструменти та показники, які фіксують статистичні особливості для визначення схожості між кодом. У цьому дослідженні основна увага приділяється письмовому тексту, оскільки він є глобальнішим завданням, а особливості лінгвістики можна досліджувати поряд із статистичними ознаками.

Що стосується плагіату письмового тексту, то найпоширеніші випадки зустрічаються в академічній галузі. У навчальних закладах зазвичай існує набір

правил, що перелічують те, що вважається плагіатом. Далі подано приклади того, які форми плагіату трапляються в наукових колах:

- Видання чужої роботи за свою
- Недостатня кількість посилань
- Пряме копіювання з одного або декількох джерел
- Перефразовування тексту

Вищезазначені випадки можуть мати місце у двох типах тексту:

- Одномовний (скопійована з однієї мови)
- Міжмовний (скопійована з другої мови, іноді її називають перекладеним плагіатом)

Щоб дотриматися обсягу дослідження, у решті даної дипломної роботи, термін "плагіат" стосується випадків, коли були написані одномовні тексти, скопійовані безпосередньо або перефразовані з одного або кількох оригінальних джерел.

1.5.Характеристики плагіату

Характеристики плагіату часто можна спостерігати із статистичних та лінгвістичних риси. Є кілька факторів, які можуть свідчити про наявність плагіату:

1) Лексичні зміни

Лексичні зміни передбачають додавання, видалення або заміну слів у тексті. Раптова зміна словникового запасу, наприклад надмірне використання нової термінології всередині документа, як правило, є гарним свідченням плагіату копіювання та вставлення. Інший прикладом може служити заміна слів на синонімами. Цей тип плагіату неможливо виявити за допомогою традиційного

підходу узгодження рядків. Виявлення такого плагіату вимагають аналізу лексичної інформації у всьому тексті.

2) Синтаксичні зміни

Зміни в синтаксичній інформації найкраще спостерігати внаслідок значної перебудови структури тексту. Приклади включають впорядкування слів / речень, пряма, непряма мова, тощо. Подібність у синтаксичних структурах може бути показником плагіату, але знову ж таки його неможливо виявити за допомогою традиційного збігу рядків, і виявлення вимагало б аналізу синтаксичної структури тексту.

3) Семантичні зміни

Це передбачає більш радикальні зміни в тексті, як правило, засновані на перефразуванні, яка може включати як лексичні, так і синтаксичні зміни. Виявлення цього типу змін потребуватиме аналізу семантичної інформації, щоб визначити, чи два тексти мають однакове значення. Знову ж таки, це неможливо виявити у традиційному підходи.

1.6. Методи виявлення плагіату

Цей розділ охоплює основні поняття про автоматичні підходи до виявлення плагіату, підходи для оцінки та різні типи методологій виявлення плагіату. Щоб визначити поняття «виявлення плагіату» для цієї роботи, спочатку необхідно визначити метод що буде застосовуватися. Існує два основних типи підходів до виявлення плагіату:

1) Внутрішній

Цей підхід стосується випадків, коли плагіат слід виявляти на основі єдиного тексту, який може містити як неплагіатні, так і плагіатні уривки. Завдання виявлення спрямоване на виявлення плагіатних частин у тексті без посилання на будь-який оригінальний текст

2) Зовнішній

Даний підхід відноситься до випадків, коли набори підозрілих плагіатних текстів та їх потенційні оригінальні тексти обидва доступні. Завдання виявлення має на меті виявити пари відповідних випадків підозрілого джерела, аналізуючи схожість кожної підозрілої справи з колекцією потенційних оригінальних текстів.

3) Змішаний

Це поєднання двох попередніх підходів. Зазвичай застосовується як поліпшення на стадії фільтрації, де зовнішнє виявлення використовується як стратегія фільтрування, а потім внутрішнє виявлення застосовується щоб визначити місце плагіату та навпаки.

Для зовнішнього та гібридного підходів можна розрізнити онлайн та офлайн підходи. Онлайн підхід проводить порівняння не тільки з локального набору даних, але також шукає в Інтернеті тексти, які можуть бути оригіналами документів. Офлайн підхід заснований на алгоритмах виявлення доказів плагіату в місцевій колекції текстів.

Також розрізняють методи виявлення плагіату за кількістю мов що обули використані:

1) Одномовне розпізнавання

Підхід одномовної розвідки розглядає підозрілі випадки та джерела лише однієї мови.

2) Виявлення між мовами

Необхідний підхід до виявлення між мовами коли підозрілі випадки походять від справ джерел різних мов. Цей підхід зазвичай вимагає узагальнення мови як частини етапу попередньої обробки.

У цій дипломній роботі основна увага приділяється зовнішньому виявленню одномовних текстів українською мовою, в режимі офлайн. Крім того, наш підхід до виявлення плагіату забезпечує зазначення потенційно плагіатних пар, замість точного визначення які частини тексту піддані плагіату.

1.7. Наявні дослідження та засоби для виявлення плагіату

Системи виявлення плагіату стартували як інструменти виявлення для тестів із множинним вибором та вихідним комп'ютерним кодом. Системи виявлення плагіату для текстів не були розроблені до 1990-х років.

У період з 1990 по 2000 р. Більшість розроблених систем були призначені для виявлення плагіату програмного коду, і лише небагато досліджень були зосереджені на виявлення плагіату для письмових текстів. Приклад цих ранніх досліджень виявлення був прототип COPS. Він був розроблений для виявлення повної або часткові копії цифрових документів. Подібність між документами вимірювали за допомогою відповідності на рівні речення. Послідовності речень у кожному документі співпадали з іншими послідовностями в документах у набору даних. Однак підхід, заснований на реченні, не був ефективним у виявленні часткове перефразованих.

Як продовження COPS, Шивакумар запропонував прототип SCAM. Підхід ввів вилучення займенників та часто повторюваних слів як етап попередньої обробки. Тексти порівнювали як накладання послідовності слів або речень, а пороги встановлювали для визначення трьох рівнів перекриття між текстами: точні копії, великі та деякі. Результати показали, що використання послідовностей слів як ознаки призвело до кращої точності, і видалення займенників пропонується як напрямок для подальшого вивчення. Встановлення порогу подібності є загальним підходом до фільтрації та виявлення, яким є також прийнято в цьому дослідженні.

Ще один ранній приклад дослідження - інструмент YAP3 для ідентифікації схожості в коді програмування, використовував алгоритм Running-Karp-Rabin Greedy-StringTiling (RKR-GST) як структуровану метричну систему виявлення подібності. Алгоритм RKR-GST є варіантом найдовшої загальної підпослідовності (LCS) алгоритм, який дозволяє максимально збігати разом із мінімальною довжиною збігу між текстами. Цей алгоритм був введений для

обробки випадків, коли послідовності тексту були впорядковані. Інструмент в основному тестувався на комп'ютерному вихідному коді та були запропоновані подальші експерименти для оцінки ефективності RKR-GST алгоритму у написаних текстах.

До кінця десятиліття у 2000 р. було лише декілька комерційних підходів для виявлення плагіату письмового тексту, для приклад EVE2 та iParadigms (рання версія Turnitin. Обидва підходи виконували виявлення в Інтернеті за допомогою пошуку подібних текстів шляхом порівняння текстів із власною базою даних.

1.8. Проблеми виявлення плагіату

У цьому розділі описані проблеми, з якими стикаються існуючі підходи до виявлення плагіату. Ці виклики можна згрупувати за двома основними напрямками: мовна складність та технічна складність.

По-перше, перекриття n-грами слів можуть бути дуже ефективними проти копій слово в слово, але випадки плагіату складніші, ніж дослівне копіювання та вставлення. Основними мовними проблемами є лексичні зміни, структурні зміни та перефразування.

1) Лексичні зміни. Це стосується використання синонімії або споріднених понять замінити оригінальні слова, які по суті містять два слова те саме значення, але з різними уявленнями. Наприклад

Джерело: Коли цей чоловік повернувся, він приніс мені лист від твого батька, в якому він сказав, що збирається зробити втечу, і що він ніколи більше вернеться в Італію.

Лексична зміна: Коли цей чоловік повернувся, він передав мені записку від вашого тато, в якому він сказав, що збирається влаштувати втечу і що він ніколи не буде повертатиметься в Італію.

2) Структурні зміни. Це стосується зміни в порядку слів, перевпорядкування компонентів речення при збереженні первісного значення.

3) Парафрази. Це стосується найскладнішої форми текстових операцій і поєднує лексичні та структурні зміни. Текст представлений за допомогою різні слова і структури, і, можливо, з різною довжиною, але значення залишається незмінним.

Одного лише перекриття n-граму недостатньо для виявлення подібності між парами тексту, але за допомогою лексичного узагальнення можна розпізнати синоніми та справитися з лексичними змінами. Синтаксичний та семантичний розбір може допомогти виявити структуру текстів, тоді як інші рівні обробки, такі як розпізнавання сутності може виділити важливі поняття в текстах. Наша гіпотеза полягає в тому, що ці методи та інші методи можуть допомогти визначити складні випадки плагіату, але вони мають свої проблеми.

Технічні труднощі також обмежують продуктивність системи. Основним обмеженням є обчислювальні ресурси. Виконуючи порівняння у великих колекціях документів необхідно оперувати значними ресурсами та обсягами пам'яті. Це є особливо проблематичним, оскільки плагіат може бути отриманий із багатьох джерел. А плагіат документу може містити текстові сегменти з більш ніж одного джерела, і важко визначити можливі сегменти джерела, якщо початковий рівень документа порівняння не змогло встановити документи- кандидати. Іншими словами, виявити плагіат із кількох джерел складніше, ніж з одного джерело, оскільки деякі показники виявлення відносять підозрілий документ лише до одного джерела документа.

Більше того, складність отримання реальної груп даних означає, що експерименти обмежуються використанням спеціально створених груп. Хоча такі групи містять дещо переписані вручну тексти, деякі з них - випадки плагіату, створені штучними засобами, що створює додатковий виклик, оскільки штучно сформовані випадки не є лінгвістично добре структурованими, а отже існуючі інструменти не можуть надійно їх обробити.

Ось деякі проблеми, з якими стикається виявлення плагіату сьогодні. Використання лише відповідності рядків буде недостатнім для вирішення цих проблем ефективно. Крім того, використання складних методів вимагає величезної кількості обчислювальні ресурси. Компроміс між обчислювальними швидкостями та надійністю виявлення потрібно враховувати при застосуванні алгоритмів.

У цій роботі проблемами, які будуть розглянуті, є лексичні зміни, структурні зміни та перефразування. Технічні проблеми полегшуються з використанням етапу фільтрації на основі простої обробки, як типового для виявлення плагіату підходу, але загалом ми зосереджені не на вирішенні цього типу викликів.

У цій главі описано існуючі підходи до виявлення плагіату. Це допомагає в досягнення першої мети шляхом ретельного розслідування поточної техніки та підходів, тим самим забезпечуючи фундаментальне розуміння, пропонуючи основу виявлення плагіату в наступному розділі. У цій главі зазначається, що більшість існуючих методів засновані на грубому поєднанні рядків алгоритмів. Існуючі методи використовують триступеневий підхід виявлення, який також є включеним у запропонованій структурі. У цій главі також описуються інші супутні дослідження щодо виявлення плагіату, виявлення міжмовного плагіату та інших виявлень подібності тексту.

2. Методи дослідження та аналізу точності алгоритмів пошуку плагіату

Цей розділ описує метрики подібності, які застосовуються після обробки тексту. Залежно від цього, обчислюються різні метрики подібності типів та рівень виконаної обробки. Застосування метрик подібності має важливе значення для генерації об’єктів, оскільки кожна функція складається з балів подібності, що генерується шляхом порівняння оброблених пар тексту та рівня подібності для кожної пара підозрілого джерела тексту.

2.1. Алгоритми виявлення плагіату

2.1.1. Мовна модель

Статистичне моделювання мови має на меті побудувати модель, яка може оцінити розподіл текстів мовою, враховуючи короткі послідовності до n слів. Прикладом набору інструментів, що дозволяє будувати такі моделі, є SRILM18. У контексті виявлення плагіату на основі моделі, побудованої з одного або декількох вихідні текстів, засоби моделювання мови корисні, для оцінки ймовірності нових послідовностей слів у підозрілому тексті за такою моделлю. Іншими словами, модель мови може розглядатися як міра того, наскільки подібні два тексти, шляхом порівняння їх n-грамового розподілу. Ми використовуємо стандартну n-грамову модель мови, яка обчислює ймовірність даного слова на основі послідовності попереднього n - 1 слово, на відміну від усіх попередніх слів у документі:

$$P(w_1^n) \approx \prod_{k=1}^{n} P(w_k|w_{k-1})$$

(2.1)

Ця модель буде обчислювати P (w | w - 1, w - 2 ..., w - n), де wi послідовності слів у підозрілому тексті від 1 до n. Ймовірності такі оцінюється з використанням частот у вихідних текстах.

Іншим методом є обчислення варіанту ймовірності мовної моделі, яке нормалізує показники ймовірності з мовних моделей відповідно до кількості слів у підозрілому тексті.

$$\frac{1}{m}\log_2 P(w_1^m) \tag{2.2}$$

Нарешті, коефіцієнт словесного запасу обчислюється шляхом підрахунку кількості слів у підозрілому тексті, якого не бачили у вихідних текстах.

2.1.2. Найпоширеніший підрядок

Іншою метрикою узгодження рядків, є алгоритм Longest Common Subsequence (LCS), який знаходить найдовшу послідовність збігів слів як у підозрілих, так і у вихідних текстах.

$$Sim_{LCS}(A, B) = \log_2 \left(1 + \frac{|LCS(A,B)|}{|B|}\right) \tag{2.3}$$

де A і B - це підозрілі та вихідні тексти відповідно. Набір LCS (A, B) - довжина найдовшого шматка тексту в A та B відповідно. Алгоритм LCS може бути реалізований шляхом порівняння пар текстів за допомогою парних порівнянь між усіма реченнями в обох текстах та поверненням найдовшої послідовності між парами речень у даному тексті. Алгоритм повертає наступне:

- Кількість відповідних слів у текстовій парі;
- Середня довжина відповідних слів у текстовій парі;
- Кількість відповідних слів у кожній парі речень;
- Середня довжина відповідних слів на пару речень;
- Загальна кількість слів і речень для кожного тексту.

Відомо, що алгоритм LCS є складним і дуже залежним від ресурсів. В цьому дослідженні використовується реалізація LCS на основі C#. Метою цього дослідження є не знайти найбільш ефективний алгоритм, а навпаки дослідити алгоритми, які може допомогти виявити плагіат.

Методи попередньої обробки тексту, генерують оцінки подібності для кожної пари тексту, і ці оцінки потім представляються як особливості на етапі класифікації. Наведені нижче показники подібності використовується на етапі генерації об'єкта результату, де порівнюються оброблені текстові пари.

2.1.3. Вилучення синтаксичної складової

Для обчислення подібності між синтаксичними складовими пар тексту кількість пересікаючих синтаксичних складових у парі тексту підозріло-джерело нормується на кількість синтаксичних складових у підозрілому тексті, використовуючи міра стримування де n = 1, що має кожну синтаксичну складову представлений як 1-грамовий.

$$Sim_{Constituent}(A, B) = \frac{|S(A,n) \cap S(B,n)|}{|S(A,n)|} \quad (2.5)$$

Нехай, наприклад, S (A, n) та S (B, n) - унікальні синтаксичні складові, що міститься у підозрілих та вихідних текстах відповідно. Перетин обох множин ділиться на кількість синтаксичних складових у підозрілому тексті S (A, n).

2.1.4. Вилучення залежності

Для обчислення подібності між відношеннями залежностей у парних текстах, залежності у підозрілому тексті порівнюються із відносинами в вихідному тексті для перевірки перекриття залежностей між двома текстами. Всього відповідність пар обчислюється з використанням коефіцієнта перекриття, де n = 1, маючи кожну залежність залежності, представлена як 1-грамова:

$$Sim_{Constituent}(A,B) = \frac{|S(A,n) \cap S(B,n)|}{\min(|S(A,n)|,|S(B,n)|)} \quad (2.6)$$

Нехай S (A, n) та S (B, n) - це унікальні відношення залежностей, що містяться у підозрілих та вихідних текстах відповідно. Кількість співвідношень, що перекриваються, нормується меншим набором S (A, n) або S (B, n).

2.1.5. Відстань Левенштейна

У теорії інформації, лінгвістиці та інформатиці відстань Левенштейна є рядковою метрикою для вимірювання різниці між двома послідовностями. Неофіційно відстань Левенштейна між двома словами - це мінімальна кількість односимвольних редагувань (вставок, видалень чи підстановок), необхідних для зміни одного слова на інше. Він названий на честь радянського математика Володимира Левенштейна, який враховував цю відстань у 1965 р.

Відстань Левенштейна також може називатися відстанню редагування, хоча цей термін може також позначати більшу сімейство метрик відстані, відомих спільно як відстань редагування. Це тісно пов'язано з попарними вирівнюваннями рядків.

Відстань Левенштейна між двома рядками A, B (довжиною | A | | B | відповідно) задається

$$lev(A,B) = \begin{cases} |A| & if|B| = 0, \\ |B| & if|A| = 0, \\ lev\big(tail(A), tail(B)\big) & if A[0] = B[0], \\ 1 + min \begin{cases} lev(tail(A), B) \\ lev(A, tail(B)) \\ lev(tail(A), tail(B)) \end{cases} & \text{інакше} \end{cases}$$

(2.7)

де $tail$ деякого рядка x - це рядок усіх, крім першого символу x, і x [n] - й символ рядка x, починаючи з символу 0.

2.1.6. Відстань Джаро – Вінклера

В інформатиці та статистиці відстань Джаро – Вінклера - це рядкова метрика, що вимірює відстань редагування між двома послідовностями. Це варіант, запропонований у 1990 р. Вільямом Е. Вінклером з метрики відстані Джаро (1989 р., Метью А. Джаро).

Чим менша відстань Джаро – Вінклера для двох струн, тим більше подібні струни. Оцінка нормується таким чином, що 0 означає точну відповідність, а 1 означає, що немає подібності. Подібність Джаро – Вінклера - інверсія, (1 - відстань Джаро – Вінклера).

Хоча її часто називають метрикою відстані, відстань Джаро – Вінклера не є метрикою в математичному розумінні цього терміна, оскільки вона не підпорядковується нерівності трикутника.

Відстань Джаро – Вінклера двох заданих рядків:

$$Sim_j = \begin{cases} 0 & if\ m = 0 \\ \frac{1}{3}\left(\frac{m}{|s_1|} + \frac{m}{|s_2|} + \frac{m-t}{m|}\right) \end{cases}$$

(2.8)

Де $|s_i|$ - довжина рядка s_i, m - кількість символів що збігається, t - половина кількості транспозицій.

2.1.7. Подібність косинусів

Подібність косинусів - це міра подібності двох ненульових векторів внутрішнього продуктового простору. Він визначається рівним косинусу кута між ними, який також збігається з внутрішнім добутком тих самих векторів, нормованих на обидва, що мають довжину 1. Косинус 0 ° дорівнює 1, і це менше 1 для будь-якого кута в інтервалі (0, π] радіанів. Таким чином, це судження про орієнтацію, а не за величиною: два вектори з однаковою орієнтацією мають косинусну подібність 1, два вектори, орієнтовані на 90 ° відносно один одного, мають подібність 0, і два діаметрально протилежні вектори мають подібність -1, незалежно від їх величини. Подібність косинусів особливо використовується в позитивному просторі, де результат чітко обмежений [0,1]. назва походить від терміна "напрямок косинуса": у цьому випадку одиничні вектори максимально "подібні", якщо вони паралельні, і максимально "несхожі", якщо вони ортогональні (перпендикулярні). Це аналогічно косинусу, який є одиницею (максимальне значення), коли сегменти подають нульовий кут і нуль (некорельований), коли сегменти перпендикулярні.

Враховуючи два вектори атрибутів, A і B, подібність косинуса, cos (θ), представляється за допомогою крапкового добутку та величини як:

$$Sim = \cos(\theta) = \frac{A*B}{\|A\|\|B\|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2}\sqrt{\sum_{i=1}^{n} B_i^2}} \quad (2.9)$$

де A_iта B_i — координати вектору A та B відповідно.

2.2. Методи оцінки

2.2.1. Коефіцієнт кореляції

Коефіцієнт Пірсона використовується для оцінки лінійної залежності між двома змінними:

$$r = \frac{1}{n-1}\sum_{i=1}^{n}\left(\frac{X_i - \bar{X}}{s_X}\right)\left(\frac{Y_i - \bar{Y}}{s_Y}\right)$$

(2.10)

Дві змінні X (підозрілі тексти) та Y (вихідні тексти) з відносною частотою n значень у X та Y, представлених Xn та Yn. Значення X та Y є представлені $\bar{X}$ та $\bar{Y}$. s_X та s_Y - стандартне відхилення X та Y.

Перевага використання коефіцієнта кореляції полягає в тому, що ознаки не потрібно нормалізувати, оскільки кореляція не залежить від ознак. Ознаки можна оцінювати індивідуально, прямолінійно.

2.2.2. F-оцінка точності

Для оцінки використовуються стандартні показники точності, віддачі, оцінки F та точності порівняно з результатами класифікації. Правильно класифіковані тексти плагіату (True Positives: TP), правильно класифіковані чисті тексти (True Negatives: TN), чисті тексти неправильно класифіковано як плагіат ((False Positives: FP), тексти плагіату, неправильно класифіковані як чисті (False

Negatives: FN), використовуються у стандартному розрахунку точності оцінки F наступним чином:

$$Precision = \ TP/(TP + FP) \tag{2.11}$$

Точність (Precision), обчислює кількість текстів, правильно визначених як такі, що належать до класу, нормованих на загальну кількість текстів як правильно, так і неправильно визначених такими що належать до цього класу.

$$Recall \ = TP/(TP + FN) \tag{2.12}$$

Віддача (Recall) обчислює кількість правильно визначених текстів, що належать до класу, нормується на загальну кількість правильно визначених текстів та тих, що маютьне були визначені як такі, що належали до цього класу, але не виначені.

$$F\ score \ = 2 * \ (P * R)/(P + R) \tag{2.13}$$

F-оцінка - це гармонічне середнє значення точності та віддачі.

2.2.3. Статистична значущість

Щоб оцінити, чи відображають отримані результати закономірності, а не просто виникають як похибка, статистична значимість обчислюється за допомогою двостороннього z-критерію. Z-тест використовується для даних із нормальним розподілом, де приклади не залежать один від одного. У цих рамках α = 0,05, де рівень довіри становить 95% або вище статистично значущий результат.

$$Z = \frac{\bar{x}_1 - \bar{x}_2 - \Delta}{\sqrt{\frac{\sigma_1^2}{n_1} + \frac{\sigma_2^2}{n_2}}} \quad (2.14)$$

де x_1 - $\bar{x}_2$ – наявна різниця, а Δ - очікувана різниця. Спостережувані та очікувані різниці нормуються стандартною похибкою для різниці, де σ_1 та σ_2 є стандартом відхилення двох популяцій, а n_1 та n_2 - це розміри двох зразків.

У цьому розділі описано загальні рамки запропонованого підходу до виявлення плагіату. Методи попередньої обробка тексту були пояснені. Методи натхненні відповідними дослідженнями та об’єднані для емпіричного аналізу. Опис методів супроводжувався переліком метрик подібності, які вимірюють схожість між текстами та генерують функції. Показники подібності включають давно встановлені алгоритми узгодження рядків, поряд із статистичною мовою моделі та найдовшим підрядком. Розділ завершився переліком загальноприйнятих показників оцінки які використовуються в цьому аналізі.

2.3. Результати експериментів

Для вибору методу пошуку плагіату, для розробки програми було вирішено провести ряд експериментів на текстах різних величин. Основною метою даних експериментів було вивчити наступні параметри кожного з алгоритмів: ресурсні затрати, швидкодія, точність та час обробки на різних групах текстів.

Щоб об’єктивно оцінити кожен алгоритм було розроблену міні програму на тій же мові що й основна програма для перевірки на плагіат. Було проведено 3 тести для кожного з алгоритмів. Для цього було розроблено три тексти різної довжини. Форматування текстів однакове. Перший тест проводився на тексті в один аркуш А4, другий на текст в 10 аркушів і третій – 100 аркушів.

Результати експерименту наведені в таблиці нижче :

Таблиця 1 – Результати експериментального дослідження

Алгоритм	Складність	Точність	Час обробки		
			1ст	10ст	100ст
Мовна модель	O(Log(m*n))	82%	91ms	1750ms	5123ms
Найпоширеніший підрядок	O(m)n	67%	82ms	1130ms	3611ms
Вилучення синтаксичної складової	O(m*n) 1	86%	65ms	1651ms	4008ms
Вилучення залежності	O(Log(m*n))	74%	40ms	2052ms	4421ms
Відстань Левенштейна	O(m*n) 1	84%	53ms	1655ms	4980ms
Відстань Джаро – Вінклера	O(m*n)	76%	84ms	1986ms	4506ms
Подібність косинусів	O(m+n)	78%	50ms	1396ms	4072ms

За результатами дослідження було вирішено застосувати алгоритм подібності косинусів, оскільки він показав одне з най крах відношень точності до часу обробки.

3. Розробка сервера пошукового ядра програми Антиплагіат

3.1.Вибір методології розробки

При виборі моделі розробки було обрано модель Scrum – гнучку методологію розробки. Scrum - одна з найпопулярніших методологій гнучкої розробки нової послуги чи продукту. Одна з причин популярності – простота. Методологія Scrum орієнтована на те, щоб швидко адаптуватися до нових завдань та задовільними потреби клієнта. Така адаптація досягається шляхом отримання зворотного зв'язку за результатами ітерації: маючи після кожної ітерації продукт, який вже можна використовувати, показувати і обговорювати, легше збирати інформацію і робити правильні коригування і змінювати пріоритети вимог.

Головні дійові особи — ScrumMaster, той хто опікується процесами, веде їх і працює як керівник проекту, Власник Продукту, людина, що представляє інтереси кінцевих користувачів та інших зацікавлених в продукті сторін, та команду, яка включає розробників.

Протягом кожного спринту, 15-30 денного періоду (тривалість визначається командою), працівники створюють функціональний ріст програмного забезпечення.

Набір можливостей, які імплементуються кожного спринту, приходять з етапу, що має назву product backlog (документація запитів на виконання робіт), який має найвищу пріоритетність за рівнем вимог до роботи, що повинна бути виконана. Запити на виконання робіт (backlog items), що визначені протягом наради з планування спринту (sprint planning meeting), переміщуються в етап спринту. Протягом цієї наради власник продукту інформує про завдання, які він хоче, аби були виконані. Тоді команда визначає, скільки з бажаного вони можуть зробити, щоб завершити необхідні частини протягом наступного спринту. Протягом спринту команда виконує визначений фіксований список завдань (т.з. backlog items). Впродовж цього періоду ніхто не має права змінювати перелік

запитів на виконання робіт, що слід розуміти, як заморожування вимог (requirements) протягом спринту.

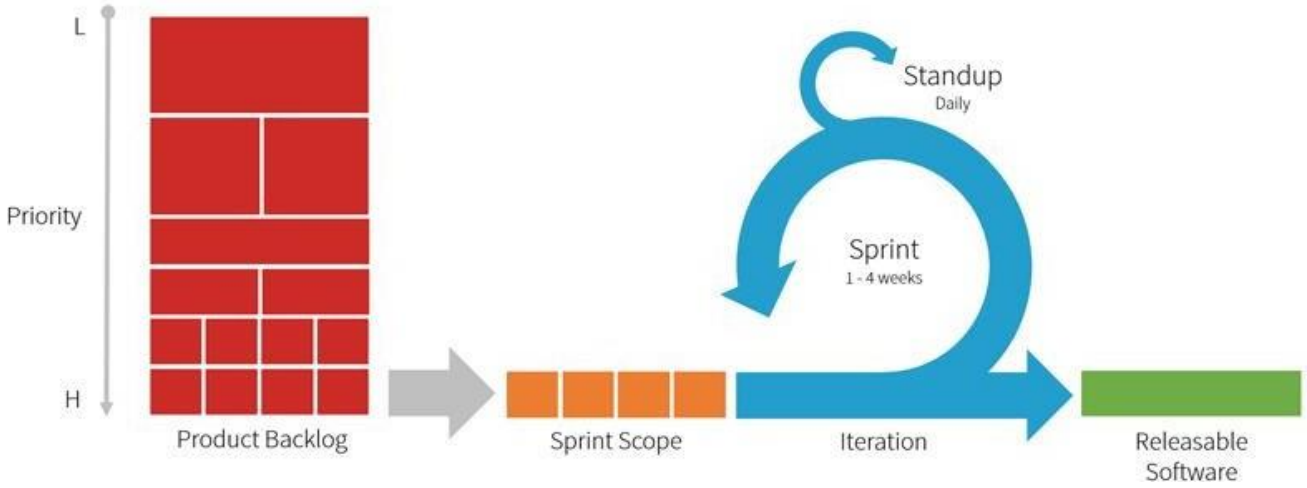

Рисунок 3.1 – Схема роботи по методології Scrum

Девіз Scrum - «аналізуй і адаптуй»: аналізуй те, що отримав, адаптуй те, що є, до реальної ситуації, а потім аналізуй знову. Таким чином виходить зациклення аж до виконання повного. Але це не означає, що формальних процесів не повинно бути зовсім, їх має бути достатньо для організації ефективної взаємодії та управління проектом.

3.2. Вибір мови програмування, IDE, СУБД

Мова програмування — один з найважливіших факторів, які впливають на якість програмного забезпечення. Для написання системи було обрано мову C# та технологію .NET. Перевагами даної технології є легкість використання, повна підтримка Майкрософтом, швидкодія роботи, гнучкість використання.

Провівши експерименти з дослідження алгоритмів для знаходження плагіату в попередніх розділах, було вияснено що більшість з них вимагають попередньої обробки даних для покращення точності. Дані після обробки займають велику кількість місця, якщо зберігати їх у стандартній реляційній базі

даних. Тому для даного продукту було обрано використовувати не реляційну базу даних.

Під нереляційними СУБД розуміються комп'ютерні системи, розроблені для зберігання, одержання та управління документо-орієнтованої чи слабоструктурованої інформації. Ці системи оперують абстрактним поняттям «Документ», в якому відбувається інкапсуляція та кодування інформації, що зберігається у стандартному форматі (XML, YAML, JSON, BSON, бінарні формати PDF, документи Microsoft Office та ін.). Документи усередині документно-орієнтованих БД деяким чином схожі на записи реляційних БД, але є більш гнучкими. Вони не вимагають наявності одних і тих же розділів, частин, ключів тощо. Документи адресуються у БД допомогою унікального ключа, який представляє конкретний документ. Часто роль ключа виконує звичайний рядок, шлях до файлу тощо. Зазвичай, документо-орієнтовані СУБД будують індекси за таким ключам, що дозволяє досить швидко отримати документ з бази даних. Прикладами документно-орієнтованих СУБД є MongoDB, IBM Lotus Notes, CouchDB, Oracle NoSQL та ін.

В даному випадку найкращим з кандидатів, який задовільняє критерії роботи, є база данних MongoDB. Для простоти роботи з даними і полегшення перегляду вмісту бази даних під час тестування було вибрано MongoDB Compas, для графічної візуалізації

3.3. Виявлення основних сутностей предметної області

Метою даної роботи було дослідження алгоритмів пошуку плагіату, для подальшого впровадження їх в розробку програми для автоматичного пошуку скопійованих робіт, та визначення відсотку унікальності. Основною метою даної програми є полегшення роботи з виявлення плагіату.

Для досягнення цілей поставлених перед програмою вона повинна мати наступні функції:

- Можливість перевірки двох документів на плагіат між собою,
- Перевірка тексту на плагіат з базою робіт
- Можливість заповнення бази даних
- Отримання звіту по унікальності проекту

В наступних розділах буде детально описано та змодельовано варіанти використання та можливості продукту для пошуку плагіату

3.4. Розробка архітектури бази даних

Для зберігання даних необхідних при перевірці текстів на плагіат було вирішено обрати не реляційну базу даних MongoDB. Так як в даній базі даних не має як такого представлення таблиць, вся інформація необхідна для програми зберігається в колекціях. Було обрано розділити данні на наступні колекції

- Колекція даних з інформацією про файл
- Колекція даних з файлами
- Колекція даних з векторами слів

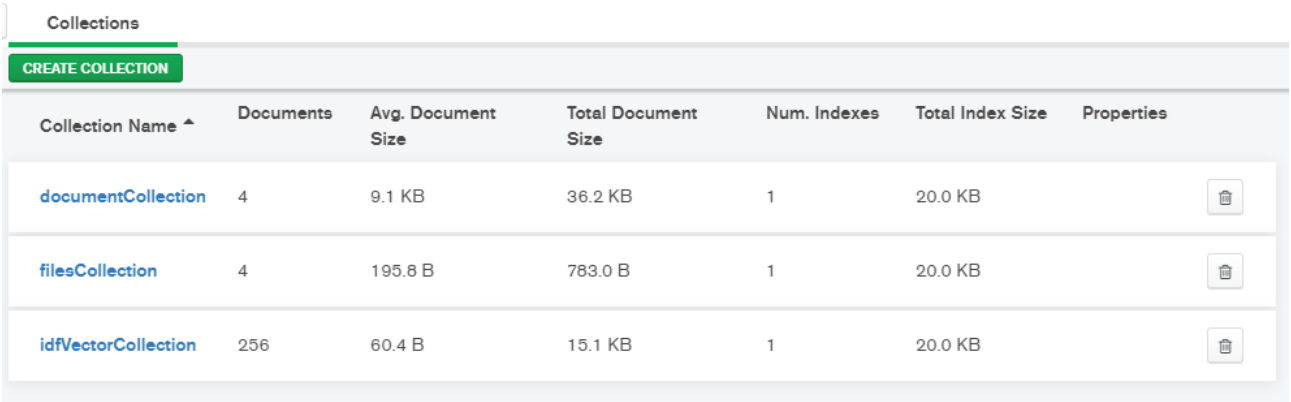

Рис 3.2 – Колекції даних

В результаті було отримано базу даних для зберігання інформації необхідної для перевірки та зберігання текстів.

3.5. Основи тестування програмного забезпечення

Тестування програмного забезпечення — техніка контролю якості, що перевіряє відповідність між реальною та очікуваною поведінкою системи завдяки кінцевому набору тестів, які обираються певним чином.

Тестування програмного забезпечення – це процес технічного дослідження, призначений для виявлення інформації про якість продукту відносно контексту, в якому він має використовуватись. Техніка тестування також включає як процес пошуку помилок або інших дефектів, так і випробування програмних складових з метою оцінки.

Оскільки число можливих тестів навіть для нескладних програмних компонент практично нескінченне, тому стратегія тестування полягає в тому, щоб провести всі можливі тести з урахуванням наявного часу та ресурсів. Як результат програмне забезпечення (ПЗ) тестується стандартним виконанням програми з метою виявлення багів (помилок або інших дефектів).

Також тестування програмного забезпечення можна проводити без відповідних тестів. Простим користувацьким тестуванням.

3.6. Тестування системи

Тестування даного програмного продукту буде проводитися за допомогою користувацього тестування. Тобто, ми будемо поетапно проводити тести функціоналу системи шляхом вводу даних.

Отже, почнемо тестування зі стартової сторінки.

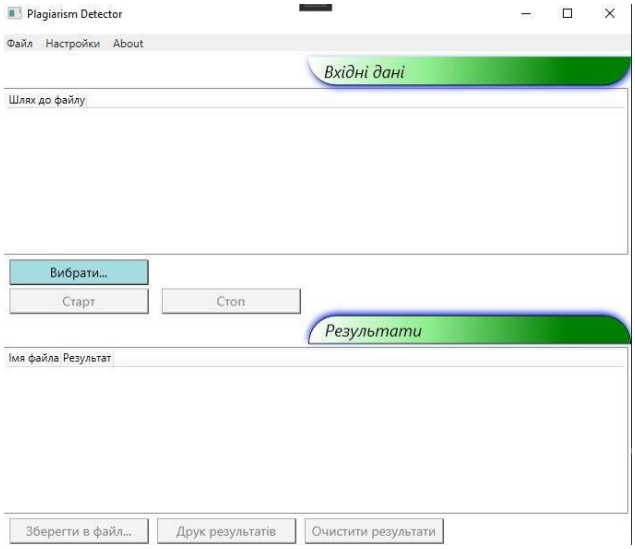

Рисунок 3.3 – Стартове вікно програми

В данному вікні відбувається завантаження файлів для перевірки та отримання результатів перевірок на плагіат.

Щоб початку роботи потрібно загрузити файл для перевірки. Це робиться за допомогую кнопки “Вибрати”, після чого можна обрати файл з поточного персонального компютера.

Для перевірки обраного файла необхідно натиснути на кнопку “Старт”, після чого програма почне початкову обробку тексту та перевірку на плагіат.

Результатами такої перевірки буде відсоткове представлення унікальності файлу, яке наведене нижче .

Рисунок 3.4 – Структура результатів перевірки

Також в даній програмі є можливість популяції бази даних, без перевірки на плагіат. Це потрібно для заповнення бази обробленими текстами для полегшення та збільшення точності наступних перевірок.

Для цього в вкладці налаштування потрібно спочатку вказати шлях до бази данних.

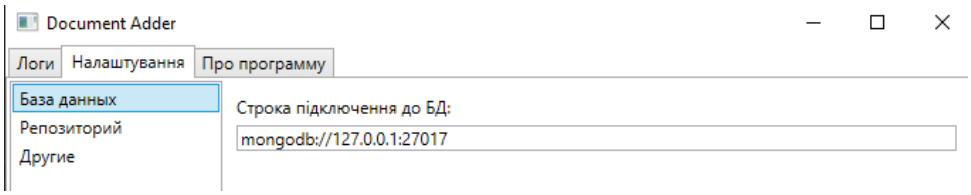

Рисунок 3.5 – Налаштування підключення до бази даних

Після цього в наступному розділі “Репозиторій” вказати папку з даними які потрібно перемістити в базу даних.

Рисунок 3.6 – Вибір репозиторію

По завершиню потрібно перейти в вкладку “Інше”, де після натиску на кнопку ”Старт” розпочнеться додавання файлів з папки у базу даних

По завершенню додавання інформації будуть виведені логи відповідно до дій які відбулися. Також якщо під час додавання файли не буде можливості завантажити через певні проблеми, дані логи також будуть відображатися в даному вікні.

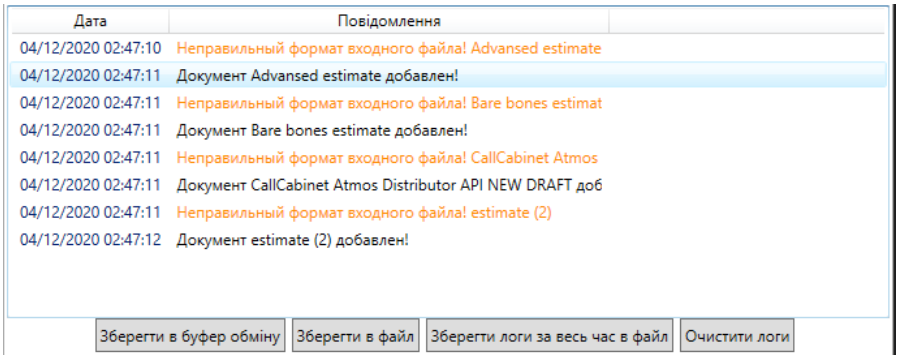

Рисунок 3.7 – Логи системи при додаванні даних

Отже, в даному розділі було проведено тестування можливостей продукту, та огляд варіантів використання. Для цього було проведене користувацьке тестування яке охоплювало весь спектр можливостей програми.

4. Охорона праці та безпеки в надзвичайних ситуаціях

4.1.Охорона праці

При роботі, пов'язаної з використанням ПК, для збереження здоров'я працюючих, запобігання професійним захворюванням І підтримки працездатності передбачаються внутрішньозмінні регламентовані перерви для відпочинку.

Внутрішньозмінні режими праці й відпочинку містять додаткові не тривалі перерви в періоди, що передують появі об'єктивних і суб'єктивних ознак стомлення й зниження працездатності.

При виконанні робіт, що належать до різних видів трудової діяльності, за основну роботу з ПК слід вважати таку, що займає не менше 50% робочого часу. Впродовж робочої зміни мають передбачатися:

- перерви для відпочинку і вживання їжі (обідні перерви);
- перерви для відпочинку і особистих потреб (згідно з трудовими нормами);
- додаткові перерви, що вводяться для окремих професій з урахуванням особливостей трудової діяльності.

За характером трудової діяльності розрізняють три професійні групи, згідно з діючим класифікатором професій:

1) розробники програм інженери-програмісти, виконують роботу переважно з відеотерміналом та документацією, при необхідності інтенсивного обміну Інформацією з ЕОМ і високою частотою прийняття рішень. Робота характеризується інтенсивною розумовою творчою працею з підвищеним напруженням зору, концентрацією уваги на фоні нервово-емоційного напруження, вимушеною робочою позою, загальною гіподинамією, періодичним навантаженням на кисті верхніх кінцівок. Робота виконується в режимі діалогу з ПК у вільному темпі з періодичним пошуком помилок в умовах дефіциту часу;

2) оператори електронно-обчислювальних машин, виконують роботу, пов'язану з обліком інформації, одержаної із ВДТ за попереднім запитом, або тієї, що надходить з нього, супроводжується перервами різної тривалості, пов'язана з

виконанням іншої роботи й характеризується напруженням зору, невеликими фізичними зусиллями, нервовим напруженням середнього ступеня та виконується у вільному темпі;

3) оператор комп'ютерного набору, виконує одноманітні за характером роботи з документацією та клавіатурою і нечастими, нетривалими переключеннями погляду на екран дисплея, з введенням даних з високою швидкістю. Робота характеризується як фізична праця з підвищеним навантаженням на кисті верхніх кінцівок на фоні загальної гіподинамії з напруженням зору (фіксація зору переважно на документи), нервово-емоційним напруженням.

Правилами встановлюються такі внутрішньозмінні режими праці та відпочинку при роботі з ПК при 8-годинній денній робочій зміні в залежності від характеру праці:

- для розробників програм із застосуванням ПК слід призначати регламентовану перерву для відпочинку тривалістю 15 хвилин через кожну годину роботи за ПК;
- для операторів із застосуванням ПК слід призначати регламентовані перерви для відпочинку тривалістю 15 хвилин через кожні дві години;
- для операторів комп'ютерного набору слід призначати регламентовані перерви для відпочинку тривалістю 10 хвилин після кожної години роботи за ПК.

У всіх випадках, коли виробничі обставини не дозволяють застосувати регламентовані перерви, тривалість безперервної роботи з ПК не повинна перевищувати 4 години.

При 12-годинній робочій зміні регламентовані перерви повинні встановлюватися в перші 8 годин робота аналогічно перервам при 8-годинній робочій зміні, а протягом останніх 4-х годин роботи, незалежно від характеру трудової діяльності, через кожну годину тривалістю 15 хвилин.

Для зниження нервово-емоційного напруження, втомлення зорового аналізатора, поліпшення мозкового кровообігу, подолання несприятливих

наслідків гіподинамії, запобігання втомі доцільно деякі перерви використовувати для виконання комплексу вправ, які наведені у Державних санітарних правилах і нормах роботи з візуальними дисплейними терміналами електронно-обчислювальних машин ДСаяПІН 3.3.2.007–98.

Працюючі з ПК підлягають обов'язковим медичним оглядам: попереднім – при влаштуванні на роботу і періодичним – протягом трудової діяльності, відповідно до наказу МЗ України N45 від 31.03.94 р.

Періодичні методичні огляди мають проводитися раз на два роки комісією в складі терапевта, невропатолога та офтальмолога.

Основними критеріями оцінки придатності до роботи з ПК мають бути показники стану органів зору: гострота зору, показники рефракції, акомодації, стану бінокулярного апарату ока тощо. При цьому необхідно враховувати також стан організму в цілому.

Виконання вимог, наведених вище, в комплексі з практичним здійсненням первинних та спеціальних заходів повинно стати нормою діяльності всіх фахівців, безпосередньо пов'язаних з роботою на ПК.

4.2. Безпека в надзвичайних ситуаціях

Практика показує, що завчасна підготовка людей і матеріально-технічних засобів до дій при виникненні надзвичайних ситуацій в значній мірі знижує ймовірність загибелі людей та втрати матеріальних засобів.

Розглянемо, які заходи необхідно вживати при виникненні надзвичайних ситуацій техногенного характеру.

В умовах хімічної аварії при надходженні сигналу "Увага всім!" необхідно включити радіоприймач і телевізор для отримання достовірної інформації про аварії та рекомендованих діях. Закрити вікна, вимкнути електропобутові прилади і газ. Надіти гумові чоботи, плащ, взяти документи, необхідні теплі речі,

тридобовий запас продукти, які не псуються, оповістити сусідів і швидко, але без паніки виходити із зони можливого зараження перпендикулярно до напрямку вітру, на відстань не менше 1,5 км від попереднього місця перебування. Для захисту органів дихання використовувати протигаз, а при його відсутності - ватно-марлеву пов'язку або підручні вироби з тканини, змочені у воді, 2-5%-ном розчині харчової соди (для захисту від хлору), 2%-ном розчині лимонної або оцтової кислоти (для захисту від аміаку).

При неможливості залишити зону зараження щільно закрити двері, вікна, вентиляційні отвори і димоходи. Наявні в них щілини заклеїти папером або скотчем. Виключити випадки знаходження на перших поверхах будинків, у підвалах і напівпідвалах.

При аваріях на залізничних і автомобільних магістралях, пов'язаних з транспортуванням АХІВ, небезпечна зона встановлюється в радіусі 200 м від місця аварії. Наближатися до цієї зони і заходити до неї категорично заборонено.

Після хімічної аварії при підозрі на ураження АХІВ виключіть будь-які фізичні навантаження, прийміть рясне питво (молоко, чай) і негайно зверніться до лікаря. Вхід до приміщення дозволяється тільки після контрольної перевірки вмісту в них ВИГУКІВ. Якщо Ви потрапили під безпосередній вплив АХІВ, при першій можливості прийміть душ. Заражену виперіть одяг, а при неможливості прання - викиньте. Проведіть ретельне вологе прибирання приміщення. Утримайтеся від вживання водопровідної (колодязної води, фруктів і овочів з городу, м'яса худоби та птиці, забитих після аварії, до офіційного висновку про їх безпеку.

При оповіщенні про радіаційної аварії, перебуваючи на вулиці, негайно захистіть органи дихання хусткою (шарфом) і поспішіть сховатися в приміщенні. Опинившись в укритті, зніміть верхній одяг і взуття, помістіть їх у пластиковий пакет і прийміть душ. Закрийте вікна та двері. Увімкніть телевізор і радіоприймач для отримання додаткової інформації про аварію і вказівок місцевої влади. Загерметизуйте вентиляційні отвори, щілини на вікнах (дверях) і не підходьте до них без необхідності. Зробіть запас води в герметичних ємностях. Відкриті

продукти загорніть в поліетиленову плівку і помістіть в холодильник (шафа). Для захисту органовдыхания використовуйте респіратор, ватно-марлеву пов'язку або підручні вироби з тканини, змочені водою для підвищення їх фільтруючих властивостей.

При одержанні вказівок через ЗМІ проведіть йодну профілактику, приймаючи протягом 7 днів по одній таблетці (0,125 г) йодистого калію, а для дітей до двох років - 1/4 частина таблетки (0,04 г). При відсутності йодистого калію використовуйте йодистий розчин: три-п'ять крапель 5%-ного розчину йоду на склянку води, дітям до двох років - одну-дві краплі.

Якщо ви опинилися в зоні радіоактивного забруднення місцевості, виходьте з приміщення тільки в разі необхідності і на короткий час, використовуючи при цьому респіратор, плащ, гумові чоботи і рукавички. На відкритій місцевості не роздягайтесь, не сідайте на землю, не паліть, виключіть купання в відкритих водоймах і збір лісових ягід, грибів. Територію біля будинку періодично зволожуйте, а в приміщенні щодня проводите ретельне вологе прибирання із застосуванням миючих засобів. Перед входом у приміщення вимийте взуття, витрусіть і почистіть вологою щіткою верхній одяг. Воду вживайте тільки з перевірених джерел, а продукти харчування - придбані в магазинах, ретельно мити перед їжею руки і полоскати рот 0,5%-вим розчином питної соди. Дотримання цих рекомендацій допоможе уникнути променевої хвороби.

У разі явної загрози життю населення відповідними органами може проводитися евакуація в безпечні зони.

Готуючись до евакуації, приготуйте засоби індивідуального захисту, у тому числі підручні (накидки, плащі із плівки, гумові чоботи, рукавички), складіть у валізу або рюкзак одяг і взуття по сезону, одноденний запас продуктів, нижню білизну, документи, гроші та інші необхідні речі. Оберніть валізу (рюкзак) поліетиленовою плівкою. Залишаючи при евакуації квартиру, вимкніть електро - та газові прилади, винесіть в сміттєзбірник швидко псуються продукти, а на двері прикріпіть оголошення "У квартирі № нікого немає". Під час посадки на транспорт або формуванні пішої колони зареєструватися у представника

эвакокомиссии. Прибувши в безпечний район, прийміть душ і змініть білизна і взуття на незаражені.

З метою недопущення або зниження негативного впливу надзвичайних ситуацій техногенного характеру органами державної влади всіх рівнів проводяться заходи у сфері природної та техногенної безпеки. До них відносяться:

1) перехід на нові принципи містобудування, що забезпечують реалізацію вимог комплексної безпеки на етапах проектування, будівництва та експлуатації будівель і споруд, а також виведення з території міст або зниження ступеня небезпеки вибухо-, хімічно-, пожежонебезпечних об'єктів і виробництв;

2) звільнення від забудови санітарно-захисних зон навколо небезпечних об'єктів, заборона і санкції проти їх подальшої забудови;

3) посилення порядку перевезення небезпечних вантажів всередині або поблизу населених пунктів;

4) облік при промислово-цивільному будівництві геологічних аномалій з ймовірними катастрофічними проявами.

Поряд з цим необхідно:

1) уживати заходів щодо скорочення застосування небезпечних речовин на об'єктах, що використовують їх у технологічному циклі;

2) здійснювати постійний радіаційно-екологічний моніторинг території регіонів і населених пунктів і радіаційне обстеження об'єктів, проведення радіаційно-аварійних робіт з дезактивації виявлених ділянок радіоактивного забруднення, збір, транспортування, переробку та кондиціонування радіоактивних відходів;

3) вести облік можливих аномальних природних явищ при розвитку комунально-енергетичних і транспортних структур, а також заборонити будівництво нових і розширення існуючих виробництв, що представляють потенційну небезпеку для міст;

4) здійснювати забезпечення функціонування загальної системи виклику екстрених оперативних служб, а також вдосконалення заходів з метою

прогнозування і профілактики надзвичайних ситуацій природного і техногенного характеру.

Таким чином, щоб підвищити рівень безпеки населення, об'єктів і інфраструктури, необхідно створити умови, що забезпечують можливість гідного життя громадян, динамічного розвитку економічної, соціальної та духовної сфер життя суспільства; створити в регіонах і містах ефективну систему забезпечення комплексної безпеки, здатну відбити існуючі та прогнозовані загрози, мінімізувати збиток від впливу деструктивних і негативних факторів, що генерують різного роду небезпеки і загрози.

ВИСНОВОК

У цій роботі представлений короткий огляд інструментів та техніки для виявлення плагіату тексту. Плагіат - явище складне. Для вирішення цієї проблеми було прийнято рішення про розробку засобу автоматизованого виявлення плагіату. На даний момент існуючі рішення за останні два десятиліття перетворилися із простих програм для узгодження тексту на потужні інструменти, здатні виявляти часткові та неперервні блоки "запозиченого" тексту. Однак вони досі не можуть виявити різні типи плагіати, починаючи від простої маніпуляції текстом, використовуючи слабкі сторони детекторів до великого переформулювання, перефразування, та перекладу вихідних документів.

Сучасні технології обробки мов можуть бути вдосконаленим за допомогою програмного забезпечення. Такі інструменти як синтаксичний та семантичний парсери, морфологічні аналізатори, моделювання тем, цитування відстеження та авторство мають потенціал стати наріжними каменями наступного покоління автоматизованих систем виявлення плагіату. Зростання якості комп'ютеризованих детекторів плагіату збільшує їх популярність, яке в свою чергу піднімає нетехнічні суперечки щодо правових та етичних питань, пов'язаних з використання таких інструментів.

Основним внеском даної роботи є порівняння наявних алгоритмів та методів виявлення плагіату. Результати експериментів показали, що один з найбільш перспективних методів - це порівняння векторів, заснований на лексичному узагальненні, залежності вилучення відносин та вилучення синтаксичних складових. Також вході дослідження було виявлено, що методи для пошуку плагіату з використанням штучного інтелекту є недостатньо дослідженими та перспективними, що може стати темою наступних робіт в даній галузі.

У цій роботі було протестовано та вивчено ряд новітніх моделей для знаходження плагіату в текстах. На основі отриманих даних було розроблено систему для розпізнавання плагіату з використанням мови програмування C#.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Офіційний сайт Microsoft Visual C# .NET. URL: https://www.visualstudio.com/ (дата звернення: 18.04.2020).
2. Шинкаренко В.І., Куропятник О.С. Система контролю плагіату в студенських роботах // Восточно-Европейский журнал передовых технологий. Харьков, 2012. № 4/2 (58). С.34–38.
3. Поповський О. І. Огляд програм порівняльного аналізу на збіг // Збірник тез доповідей 2-го Кіровоградського соціально-економічного форуму «Інформаційне суспільство і влада». Кіровоград. 2013. С. 99–100.
4. Мокін В. Б. Автоматизована система перевірки текстів на плагіат / В. Б. Мокін, В. В. Войтко, С. В. Бевз, О. В. Гавенко, І. А. Білоус // Вісник Вінницького політехнічного інституту. 2010. №5. С. 12–17
5. Автоматическая обработка текстов на естественном языке и компьютерная лингвистика : учеб. пособие / Большакова Е.И., Клышинский Э.С., Ландэ Д.В., Носков А.А., Пескова О.В., Ягунова Е.В. М.: МИЭМ, 2011. 272 с.
6. The Porter Stemming Algorithm. URL: https://tartarus.org/martin/PorterStemmer (дата звернення: 18.04.2020).
7. Lancaster, T. Effective and Efficient Plagiarism Detection : PhD thesis / Lancaster Thomas. – London, 2003. – 228 p.
8. Lancaster T. Classifications of Plagiarism Detection Engines [online] / T. Lancaster, F. Culwin // Innovation in Teaching and Learning in Information and Computer Sciences. 2005. – №2 (4). – 16 p. – Available from: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.184.2406&rep=rep1&type=pdf.
9. Marcial D. E. ICT Skills Enhancement Training in Teacher Education: The Case in Central Visayas, Philippines [online] / D. E. Marcial, M. S. Fortich, J. B. Rendal// Information Technologies and Learning Tools. – 2014. – № 1 (39).— P. 230-240 – Available from: http://journal.iitta.gov.ua/index.php/itlt/article/view/964/749.
10. Meyer zu Eissen S. Intrinsic Plagiarism Detection [online] / S. Meyer zu Eissen, B. Stein// Advances in Information Retrieva: Proceedings of the 28th European

Conference on IR Research, ECIR. – 2006. – P. 565-569 – Available from: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.110.5366&rep=rep1&type=pdf.

11. Shenoy M. Automatic Plagiarism Detection Using SimilarityAnalysis [online] / M. Shenoy, K. C.Shet, U. D. Acharya// Advanced Computing: An International Journal (ACIJ). – 2012. – № 3 (3).— P. 59-62 – Available from: http://airccse.org/journal/acij/papers/0512acij06.pdf.
12. Singh R. Duplicity Detection System for Digital Documents [online] / R. Singh, C. Dutta// International Journal of Soft Computing and Engineering (IJSCE). – 2012. – № 5 (2).— P. 24-28 – Available from: http://www.iaeng.org/publication/IMECS2011/IMECS2011_pp272-277.pdf.
13. Автоматизована система перевірки текстів на плагіат / В. Б. Мокін, В. В. Войтко, С. В. Бевз [та ін.] // Вісник Вінницького політехнічного інституту. – 2010. – № 5.— С. 12—17.
14. Квашина Ю. А. Методы поиска дубликатов скомпонованных текстов научной стилистики [Электронный ресурс] / Ю. А. Квашина // Технологический аудит и резервы производства. – 2013. – № 3 (3).— С. 16-20 – Режим доступа : http://journals.uran.ua/tarp/article/viewFile/14893/12698
15. Tschuggnall M. Detecting Plagiarism in Text Documents through Grammar-Analysis of Authors [online] / M. Tschuggnall, G. Specht// 15th GI-Symposium Database Systems for Business, Technology and Web, 11th March - 15th March, 2013. – 2013. — P. 241-259 – Available from: http://www.btw2013.de/proceedings/Detecting%20Plagiarism%20in%20Text%20Documents%20through%20Grammar Analysis%20of%20Authors.pdf.
16. М.Р. Петрик, Д.М. Михалик, О.Ю. Петрик, Г.Б. Цуприк. Методичні вказівки до виконання атестаційної роботи магістра за спеціальністю 121 – “Інженерія програмного забезпечення” для усіх форм навчання [Текст] – Тернопіль : Тернопільський національний технічний університет імені Івана Пулюя – 2020 – 27 с.

ДОДАТКИ

ДОДАТОК А

Міністерство освіти і науки України

Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп'ютерно-інформаційних систем і програмної інженерії

Кафедра програмної інженерії

ЗАТВЕРДЖУЮ

Завідувач кафедрою

програмної інженерії

"____" _____2020 р.

ТЕХНІЧНЕ ЗАВДАННЯ

на виконання кваліфікаційної роботи магістра

на тему: «Розробка сервера ядра системи Антиплагіат»

виконавець ст. гр. СПм-61

Зелений Віктор Васильович

_____

(підпис)

керівник роботи:

д.ф-м.н. Петрик Михайло Романович

_____

(підпис)

Тернопіль 2020

ЗМІСТ

1 ПІДСТАВИ ДО РОЗРОБКИ

Розробка проводиться у відповідності до графіку навчального плану на 2020 рік, та згідно наказу на виконання кваліфікаційної роботи студента-магістра.

Тема проекту: «Розробка сервера ядра системи Антиплагіат».

2 ПРИЗНАЧЕННЯ РОЗРОБКИ

Ця дипломна робота пропонує нові перспективи щодо виявлення плагіату. Гіпотеза полягає в тому, що оригінальні тексти та переписані тексти мають суттєві, але вимірювані відмінності, і що ці відмінності можна охопити за допомогою статистичних та лінгвістичних показників. Щоб дослідити цю гіпотезу, визначено основні цілі дослідження.

По-перше, пропонується нова основа для виявлення плагіату. Він включає використання методів обробки природної мови, а не лише покладання на традиційні підходи до узгодження рядків. Завдання полягає у дослідженні та оцінці впливу попередньої обробки тексту, а також статистичної, поверхневої та глибокої лінгвістичної методики. Це досягається шляхом оцінки техніки в двох основних експериментальних умовах.

По-друге, досліджується перспектива застосування запропонованих рамок у масштабному сценарії. Завдання полягає у дослідженні масштабованості запропонованого алгоритми. Це досягається експериментами з великими масштабами. Перші два етапи базуються на більшій довжині тексту та заключний етап базується на фрагментах текстів.

Нарешті, розробимо додаток що допоможе розрізняти плагіат в текстах, використовуючи набуті знання. Статистична та лінгвістична ознаки досліджуються індивідуально або в різних поєднаннях. Завдання полягає у представлені нового погляду на традиційне порівняння з використанням грубої сили.

3 ВИМОГИ ДО ПРОГРАМНОГО ПРОДУКТУ

3.1 Функціональні характеристики

Програмне забезпечення має виконувати наступні дії:

- Можливість заповнення бази текстів ;
- Перевірка текстів на пагіат;
- Обробка результатів та формування звіту;

3.2 Склад та параметри технічних засобів

1) ПК із 4096 Мб оперативної пам‘яті, встановленою операційною системою Windows Seven, 8, 8.1, 10. Не менше 1024 Мб вільного місця на жорсткому диску. Двоядерний процесор з тактовою частотою від 1.2 GHz і більше.
2) Наявність встановленого Entity Framework 4.5

3.3 Інформаційна та програмна сполучність

Програмний продукт повинен коректно функціонувати в операційних системах Windows Seven, 8, 8.1, 10, на яких доступний для встановлення Entity Framework 4.5. Розроблювана система повинна бути пристосована для автоматизованої ідентифікації плагіату та зереження результатів в базі даних.

4. СТАДІЇ РОЗРОБКИ

В ходів реалізації роботи проект повинен пройти крізь наступні стадії розробки:

- аналіз предметної області;
- аналіз маркетингових стратегій;
- аналіз доступних алгоритмів виявлення плагіату;
- проектування та розробка застосунку;
- оформлення супровідної документації;
- здача роботи.

5. ПРОГРАМНА ДОКУМЕНТАЦІЯ

Для програмного продукту повинні бути розроблені наступні документи:

- Пояснювальна записка;
- Технічне завдання;
- Презентаційний матеріал;
- Додатки.

6. ПОРЯДОК КОНТРОЛЮ ТА ПРИЙМАННЯ

Розроблений програмний продукт має виконувати всі вимоги, що складаються з перерахованих у п. 3.1 характеристик.

Приймання проводиться спеціально створеною екзаменаційною комісією в термін до:

“__” грудня 2020р.

ДОДАТОК Б

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ТЕРНОПІЛЬСЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
ІМЕНІ ІВАНА ПУЛЮЯ

М А Т Е Р І А Л И

VIII НАУКОВО-ТЕХНІЧНОЇ КОНФЕРЕНЦІЇ

«ІНФОРМАЦІЙНІ МОДЕЛІ, СИСТЕМИ ТА ТЕХНОЛОГІЇ»

9–10 грудня 2020 року

ТЕРНОПІЛЬ
2020

СЕКЦІЯ 4. **ПРОГРАМНА ІНЖЕНЕРІЯ ТА МОДЕЛЮВАННЯ СКЛАДНИХ РОЗПОДІЛЕНИХ СИСТЕМ**

УДК 004.4
В.В. Зелений
(Тернопільський національний технічний університет імені Івана Пулюя)

АНАЛІЗ АЛГОРИТМІВ ПОШУКУ ПЛАГІАТУ ЛЕКСЕМ

UDC 004.4
V.V. Zelenyi

ANALYSIS OF PLAGIARISM SEARCH ALGORITHMS

Щоб виявити плагіат, важливо мати широкі знання про його можливі форми та типи, а також існування різних засобів та систем для його виявлення. Плагіат може мати місце у статті чи будь-якому текстовому виданні. З роками було запроваджено чимало інструментів та прийомів для виявлення плагіату. У цій доповіді буде висвітлено кілька перспективних методів виявлення плагіату та проаналізовано складність цих алгоритмів.

1. Плагіат у сучасному суспільстві

Завдяки цифровій ері, обсяг цифрових ресурсів у Всесвітній павутині збільшується. При швидкому зростанні цих ресурсів, можливість порушення авторських прав та плагіат також зростають. Щоб вирішити цю проблему дослідники почали працювати над виявленням плагіату між різними мовами з 1990 р. Це було новаторським методом виявлення копій у цифрових документах[1].

2. Виявлення плагіату

Плагіат може відбуватися між двома однаковими або двома різними мовами. На основі мовної однорідності або неоднорідності текстових документів, що порівнюються, виявлення плагіату можна розділити на два основних типи[4].

1. Виявлення одномовного плагіату: цей тип виявлення стосується однорідних текстів плагіату, наприклад, українська-українська. Більшість методів виявлення відносяться до цієї категорії[2].

2. Виявлення міжмовного плагіату: цей підхід виявлення може виконуватись у неоднорідних текстах плагіату, українська-англійська. Є лише невелика кількість способів розпізнавання даного плагіату через труднощі у пошуку близькості між двома текстовими сегментами для різних мов.

2.1. Знаходження подібності для порівняння документів або сегментів тексту

Щоб виявити плагіат, нам слід виміряти подібність між двома документами. Для цього більшість дослідників використовують наступні два типи метрик подібності[3].

1. Показник подібності рядків (String Similarity Metric): це метрика, яка вимірює відстань між двома текстовими рядками для приблизної відповідності рядків.

2. Метрика векторної схожості (Vector Similarity or Cosine similarity Metric): коефіцієнт подібності двох не нульових векторів у предгільбертовому просторі, який обчислюється як косинус кута між ними.

2.2. Методи виявлення плагіату

Виявлення плагіату в текстовому документі, з високою точністю, є складним завданням. Два десятиліття дослідники повідомляють про велику кількість методів для вирішення цього завдання. Деякі відомі методи будуть висвітлені далі.

1. Відстань Левенштейна (Levenshtein distance): у теорії інформації і комп'ютерній лінгвістиці міра відмінності двох послідовностей символів (рядків). Обчислюється як мінімальна кількість операцій вставки, видалення і заміни, необхідних для перетворення одної послідовності в іншу.

2. Відстань Джаро-Вінклера (Jaro–Winkler distance): є рядковою метрикою, що вимірює відстань редагування між двома послідовностями. Це варіант, запропонований у 1990 р. Вільямом Е. Вінклером з метрики відстані Джаро

ДОДАТОК В