

Вінницький національний технічний університет
Факультет інтелектуальних інформаційних технологій та автоматики
Кафедра автоматизації та інтелектуальних інформаційних технологій

МАГІСТЕРСЬКА КВАЛІФІКАЦІЙНА РОБОТА

на тему:

«Автоматизована система класифікації навчальних текстів дисциплін з
використанням нейронних мереж»

Виконав: студент 2-го курсу гр. 1АКІТР-23м
(шифр групи)

спеціальності 174 – Автоматизація, комп'ютерно-
інтегровані технології та робототехніка
(шифр та назва спеціальності)

Антон ТКАЧЕНКО

(ім'я ПРІЗВИЩЕ студента)

Керівник: к.т.н., доц. каф. АІТ

Володимир СЕВАСТЬЯНОВ

(науковий ступінь, вчене звання / посада, ім'я ПРІЗВИЩЕ
керівника)

«14» 12 2024 р.

Опонент: д.т.н., проф. каф. КСУ

Марія ЮХИМЧУК

(науковий ступінь, вчене звання / посада, ім'я ПРІЗВИЩЕ
керівника)

«14» 12 2024 р.

Допущено до захисту
Завідувач кафедри АІТ
д.т.н., проф. Олег БІСКАЛО

«18» 12 2024 р.

Вінниця ВНТУ – 2024 рік

Вінницький національний технічний університет
Факультет інтелектуальних інформаційних технологій та автоматизації
Кафедра автоматизації та інтелектуальних інформаційних технологій
Рівень вищої освіти II-й (магістерський)
Галузь знань - 17 - Електроніка, автоматизація та електронні комунікації
Спеціальність - 174 - Автоматизація, комп'ютерно-інтегровані технології та робототехніка
Освітньо програма - Інтелектуальні комп'ютерні системи

ЗАТВЕРДЖУЮ

Завідувач кафедри АІТ д.т.н.,
проф. Олег БІСІКАЛО
"19" вересня 2024 року

З А В Д А Н Н Я
НА МАГІСТЕРСЬКУ КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТУ

Ткаченко Антон Володимирович

(прізвище, ім'я, по батькові)

- Тема роботи: «Автоматизована система класифікації навчальних текстів дисциплін з використанням нейронних мереж»
Керівник роботи: к.т.н., доц. каф. АІТ Володимир СЕВАСТЬЯНОВ
Затверджені наказом ВНТУ від « 17 » вересня 2024 року № 310 .
- Термін подання студентом роботи до «__» грудня 2024 року.
- Вихідні дані до роботи: обсяг оперативної пам'яті: не менше 2 ГБ, операційна система Windows 10, мова програмування Python (TensorFlow, PyTorch / Keras), навчальні тексти дисциплін, аналітичні дані згенеровані алгоритмами машинного навчання включаючи результати кластеризації текстів та визначення ключових слів
- Зміст текстової частини: вступ, аналіз предметної області, розрахунково-пояснювальної записки (перелік питань, які потрібно розробити), загальна характеристика об'єкта дослідження, опис системи, яка буде використовуватись, вибір мови програмування та середовища розробки, розроблення інформаційної технології та її застосування на реальних даних, економічна частина, висновки.
- Перелік ілюстративного (або графічного) матеріалу, Скріншоти інтерфейсу чи зображення архітектури, застосунку, Use Case діаграма - класифікація тексту, Порядок дій для навчання моделі, Діаграма класів, Порядок взаємодії компонентів, Розгортання програми, Схема класифікації тексту.

6. Консультанти розділів проекту (роботи)

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		Завдання видав	Завдання прийняв
1-3	Севастьянов В.М. – к.т.н., доцент кафедри АІТ	18.09.24	05.12.24
4	Козловський В. О. – к.т.н., доцент кафедри ЕПтаВМ	01.10.24	07.12.24

7. Дата видачі завдання «18» вересня 2024 року.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проекту (роботи)	Термін виконання етапів проекту (роботи)	Примітка
1	Аналіз предметної області	18.09.24	20.09.24
2	Вибір оптимальних інформаційних технологій	20.09.24	23.09.24
3	Вибір мови програмування та середовища розробки	23.09.24	04.10.24
4	Програмна реалізація	04.10.24	11.11.24
5	Експериментальна перевірка роботи програми	11.11.24	18.11.24
6	Оформлення матеріалів до захисту МКР	01.12.24	10.12.24
7	Захист МКР	17.12.24	18.12.24

Студент

Керівник роботи

(підпис)

(підпис)

Антон ТКАЧЕНКО

(прізвище та ініціали)

Володимир СЕВАСТЬЯНОВ

(прізвище та ініціали)

АНОТАЦІЯ

УДК 004.75

Ткаченко А.В. Автоматизована система класифікації навчальних текстів дисциплін з використанням нейронних мереж. Магістерська кваліфікаційна робота зі спеціальності 174 – Автоматизація, комп'ютерно-інтегровані технології та робототехніка. Вінниця: ВНТУ, 2023. 96 с.

На укр. мові. Бібліогр.: 31 посилань; рис.: 10; табл.: 6.

Магістерська кваліфікаційна робота присвячена розробці автоматизованої системи класифікації навчальних текстів дисциплін з використанням нейронних мереж. Проаналізовано сучасний стан проблеми та існуючі підходи до класифікації текстів у контексті освітнього процесу. Розроблено архітектуру та функціональну структуру системи, що інтегрує модель нейронної мережі для аналізу і категоризації навчальних матеріалів за дисциплінами. Система реалізована з використанням мови Python та бібліотек TensorFlow і Keras. Проведено експериментальне дослідження ефективності розробленої системи на реальних даних. Результати показали високу точність класифікації та значне скорочення часу обробки текстів у порівнянні з традиційними методами. Розроблена система може бути впроваджена в навчальний процес для підвищення ефективності роботи з текстовими матеріалами та персоналізації освіти. Визначено напрямки подальшого вдосконалення та розвитку системи.

Ключові слова: класифікація текстів, нейронні мережі, машинне навчання, обробка природної мови, автоматизація, навчальні дисципліни.

ANNOTATION

Rusavskiy O. O. Automated System for Classification of Educational Texts in Disciplines Using Neural Networks. Master's qualification thesis in specialty 174 – Automation, Computer-Integrated Technologies, and Robotics. Vinnytsia: VNTU, 2023. 96 pages.

In Ukrainian. Bibliography: 31 references; figures: 10; tables: 6.

The master's qualification thesis is devoted to the development of an automated system for the classification of educational texts in disciplines using neural networks. The current state of the problem and existing approaches to text classification in the context of the educational process were analyzed. The architecture and functional structure of the system integrating a neural network model for analyzing and categorizing educational materials by disciplines were developed.

The system was implemented using Python and the TensorFlow and Keras libraries. Experimental research on the system's efficiency was conducted on real-world data. The results demonstrated high classification accuracy and significant reduction in text processing time compared to traditional methods. The developed system can be integrated into the educational process to enhance the efficiency of working with textual materials and personalize education. Directions for further improvement and development of the system have been identified.

Keywords: text classification, neural networks, machine learning, natural language processing, automation, academic disciplines.

ЗМІСТ

ВСТУП.....	5
РОЗДІЛ 1. ЗАГАЛЬНА ХАРАКТЕРИСТИКА ОБ'ЄКТА ДОСЛІДЖЕНЬ.....	8
1.1 Опис об'єкта досліджень	8
1.2 Аналіз універсальних можливостей системи, інтеграція, масштабованість та аналітика	18
1.3 Дослідження існуючих прототипів системи	28
РОЗДІЛ 2. РОЗРОБКА ФУНКЦІОНАЛЬНОЇ СТРУКТУРИ ПРОГРАМИ.....	39
2.1 Опис системи, яка буде використовуватись	39
2.2 Опис структури автоматизованої системи класифікації текстів	46
2.3 Архітектура, функціональність та інтеграція програмного модуля класифікації текстів	46
РОЗДІЛ 3. АДАПТАЦІЯ, ТЕСТУВАННЯ ТА ВДОСКОНАЛЕННЯ РОЗРОБЛЕНОЇ СИСТЕМИ	58
3.1 Результати аналізу та визначення задачі.....	58
3.2 Напрямки вдосконалення розробленої автоматизованої системи класифікації навчальних текстів	59
3.3 Можливості розширення та вплив системи	58
4. ЕКОНОМІЧНИЙ РОЗДІЛ	68
4.1 Технологічний аудит розробленої автоматизованої системи класифікації навчальних текстів дисциплін	68
4.2 Розрахунок витрат на розроблення автоматизованої системи класифікації навчальних текстів дисциплін	73
4.3 Розрахунок економічного ефекту від можливої комерціалізації розробленої нами автоматизованої системи класифікації навчальних текстів дисциплін	78
ВИСНОВКИ	86
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	88
Додаток А (обов'язковий) Технічне завдання.....	94
Додаток Б (обов'язковий) Ілюстративна частина	98
Додаток В (обов'язковий) Лістинг програми.....	101

Додаток Г (обов'язковий) Протокол перевірки на плагіат **Error! Bookmark not defined.**

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

- AI (Artificial Intelligence) - штучний інтелект
- API (Application Programming Interface) - прикладний програмний інтерфейс
- BERT (Bidirectional Encoder Representations from Transformers) - двонаправлені представлення кодера від трансформерів
- CNN (Convolutional Neural Network) - згорткова нейронна мережа
- CSV (Comma-Separated Values) - значення, розділені комою
- GRU (Gated Recurrent Unit) - вентиляний рекурентний блок
- HTML (HyperText Markup Language) - мова розмітки гіпертексту
- HTTP (Hypertext Transfer Protocol) - протокол передачі гіпертексту
- IDE (Integrated Development Environment) - інтегроване середовище розробки
- IDF (Inverse Document Frequency) - обернена частота документа
- JSON (JavaScript Object Notation) - запис об'єктів JavaScript
- LMS (Learning Management System) - система управління навчанням
- LSTM (Long Short-Term Memory) - довга короткочасна пам'ять
- ML (Machine Learning) - машинне навчання
- MLP (Multilayer Perceptron) - багатошаровий перцептрон
- NLP (Natural Language Processing) - обробка природної мови
- PDF (Portable Document Format) - портативний формат документів
- ReLU (Rectified Linear Unit) - випрямлена лінійна одиниця
- RNN (Recurrent Neural Network) - рекурентна нейронна мережа
- SGD (Stochastic Gradient Descent) - стохастичний градієнтний спуск
- SVG (Scalable Vector Graphics) - масштабована векторна графіка
- SVM (Support Vector Machine) - метод опорних векторів
- TF (Term Frequency) - частота терміну
- TF-IDF (Term Frequency-Inverse Document Frequency) - частота терміну-обернена частота документа
- URL (Uniform Resource Locator) - уніфікований показник ресурсу

XML (eXtensible Markup Language) - розширювана мова розмітки

ВСТУП

Актуальність теми. У сучасному освітньому процесі, особливо у вищих навчальних закладах, постійно зростає обсяг навчальних матеріалів, що включає лекції, посібники, статті та інші джерела. Це ускладнює визначення дисципліни, до якої належить той чи інший матеріал, особливо для неекспертів. Автоматизація класифікації текстів здатна суттєво спростити роботу з контентом та підвищити ефективність його використання.

Традиційні методи ручної категоризації є трудомісткими, суб'єктивними та схильними до помилок, що стає проблемою в умовах швидкого оновлення навчальних програм. Технології штучного інтелекту, зокрема нейронні мережі, відкривають нові можливості для автоматизації цього процесу. Вони можуть аналізувати великі обсяги даних, виявляти приховані закономірності та забезпечувати точну класифікацію навчальних текстів.

Застосування нейронних мереж дозволяє пришвидшити процес класифікації, мінімізувати вплив людського фактора та забезпечити об'єктивну категоризацію. Це спрощує навігацію по ресурсах, полегшує пошук інформації для студентів і викладачів, а також сприяє ефективному управлінню контентом.

Інтеграція таких систем з платформами дистанційної освіти, LMS та електронними бібліотеками створює персоналізоване освітнє середовище, яке відповідає індивідуальним потребам студентів. Крім того, це сприяє стандартизації підходів до класифікації матеріалів, що важливо для глобальної освітньої співпраці.

Таким чином, розробка автоматизованої системи класифікації навчальних текстів з використанням нейронних мереж є перспективним напрямком, який здатний підвищити ефективність освітнього процесу та підтримати інноваційні форми навчання.

Мета і задачі дослідження. Метою роботи є підвищення рівня зручності та ефективності класифікації навчальних текстів за допомогою нейронних

мереж, що дозволить швидко і точно визначати, до якої дисципліни належить наданий матеріал.

Для досягнення цієї мети потрібно розв'язати такі задачі дослідження:

- Провести аналіз існуючих архітектурних нейронних мереж для задачі класифікації навчальних текстів.
- Проаналізувати існуючі методи класифікації текстів у контексті освітніх дисциплін.
- Визначити сильні та слабкі сторони кожної з цих систем.
- Розробити інноваційну автоматизовану систему, що інтегрує модель нейронної мережі для аналізу і класифікації текстів за навчальними дисциплінами.

Об'єктом дослідження є процеси автоматизованої класифікації навчальних текстів у вищих навчальних закладах.

Предметом дослідження є методи та засоби класифікації текстів за допомогою нейронних мереж для визначення їх відповідності навчальним дисциплінам.

Методи дослідження. У роботі використовуються методи обробки текстів, машинного навчання та порівняння різних архітектур нейронних мереж. Аналізуються їх можливості в класифікації текстів у контексті навчальних дисциплін, а також оцінюється їх точність і швидкість роботи.

Науково-технічний результат полягає у застосуванні сучасних нейронних мереж для класифікації навчальних текстів, що дозволить автоматизувати процес визначення дисципліни, до якої належить той чи інший матеріал.

Практична цінність роботи полягає в можливості впровадження автоматизованої системи класифікації текстів у навчальний процес, що полегшить роботу студентів і викладачів з великими обсягами текстових матеріалів і підвищить ефективність освітнього процесу.

Апробація результатів. Основні положення та результати роботи доповідалися та обговорювалися на наукових конференціях та семінарах,

зокрема на XXI Міжнародній науково-практичній конференції "Сучасні інформаційні технології та інноваційні методики навчання в підготовці фахівців" (м. Вінниця, 2023 р.) та на науковому семінарі кафедри автоматизації та інтелектуальних інформаційних технологій Вінницького національного технічного університету (м. Вінниця, 2024 р.).

Апробація наукової роботи: результати дослідження апробовано на всеукраїнській науково-технічній конференції факультету інтелектуальних інформаційних технологій та автоматизації [1].

РОЗДІЛ 1. ЗАГАЛЬНА ХАРАКТЕРИСТИКА ОБ'ЄКТА ДОСЛІДЖЕНЬ

1.1 Опис об'єкта досліджень

Автоматизована система класифікації навчальних текстів дисциплін з використанням нейронних мереж являє собою складний програмно-технічний комплекс, призначений для вирішення актуальних задач сучасної освіти. В умовах стрімкого зростання обсягів навчальної інформації особливої необхідності набуває проблема її ефективної організації та категоризації. Система використовує передові методи машинного навчання для автоматичного аналізу та класифікації текстових матеріалів за відповідними навчальними дисциплінами. Особливістю розробленої системи є її здатність враховувати специфіку освітніх матеріалів, включаючи особливості наукової термінології та структури навчальних текстів [1-3]. Найголовнішим аспектом функціонування системи є забезпечення високої точності класифікації, оскільки коректне віднесення матеріалу до відповідної дисципліни безпосередньо впливає на якість навчального процесу. Система також враховує можливість міждисциплінарних зв'язків та забезпечує відповідну обробку таких матеріалів. Реалізація системи базується на використанні сучасних технологій обробки природної мови та глибокого навчання.

Сучасний освітній процес характеризується високою динамічністю та постійним оновленням навчальних матеріалів, що створює додаткові вимоги до систем їх автоматизованої обробки. Розроблена система враховує ці особливості через реалізацію механізмів адаптивного навчання та оновлення моделей класифікації. Головною характеристикою системи є її здатність до обробки різних форматів навчальних матеріалів, включаючи електронні підручники, методичні вказівки, наукові статті та презентації. Система також забезпечує можливість роботи з матеріалами різних мов, що особливо актуально в умовах інтернаціоналізації освіти. Реалізовані алгоритми попередньої обробки тексту дозволяють ефективно видаляти шуми та виділяти значущі характеристики

документів. Врахування специфіки навчальних текстів здійснюється через використання спеціалізованих словників термінів та онтологій. Особлива увага приділяється обробці технічної термінології та математичних формул. Система також враховує структурні особливості навчальних матеріалів різних типів [4].

Особливу роль у функціонуванні системи відіграє механізм векторизації текстів, який забезпечує перетворення текстових даних у формат, придатний для обробки нейронними мережами. Реалізований підхід базується на використанні сучасних методів представлення слів та враховує контекстні зв'язки між термінами. Головним аспектом є також забезпечення збереження семантичної близькості між термінами споріднених дисциплін. Система використовує багаторівневу архітектуру обробки тексту, що дозволяє враховувати як локальні особливості окремих термінів, так і глобальний контекст документа. Особлива увага приділяється обробці спеціалізованої термінології та аббревіатур, характерних для різних навчальних дисциплін. Реалізований механізм дозволяє автоматично виявляти та коректно обробляти нові терміни, що з'являються в навчальних матеріалах. Ця здатність до адаптації є критично вагомим для забезпечення актуальності системи в умовах постійного розвитку наукового знання.

Розроблена система демонструє високу ефективність при роботі з великими обсягами навчальних матеріалів різної структури та складності. Найголовнішою особливістю є здатність системи до паралельної обробки множини документів, що забезпечує високу продуктивність при масовій класифікації матеріалів. Реалізований механізм балансування навантаження дозволяє оптимально розподіляти обчислювальні ресурси між різними компонентами системи. Особлива увага приділяється забезпеченню стабільності роботи при тривалій експлуатації та високих навантаженнях. Система включає механізми автоматичного відновлення після збоїв та збереження проміжних результатів обробки. Вагомим аспектом є також забезпечення цілісності даних та захист від можливих помилок при класифікації. Реалізована система

моніторингу дозволяє відслідковувати якість роботи та своєчасно виявляти потенційні проблеми.

Інтелектуальна складова системи базується на використанні сучасних методів глибокого навчання та нейронних мереж. Архітектура нейронної мережі розроблена з урахуванням специфічних вимог до класифікації навчальних текстів та включає спеціалізовані шари для обробки технічної термінології. Головною особливістю є використання механізмів уваги, які дозволяють системі фокусуватися на найбільш значущих частинах тексту при прийнятті рішення про класифікацію. Система використовує багаторівневий підхід до аналізу документів, що дозволяє враховувати як локальні особливості окремих фрагментів, так і загальний контекст матеріалу. Особлива увага приділяється обробці міждисциплінарних зв'язків та виявленню неявних тематичних залежностей. Реалізований механізм навчання дозволяє системі постійно вдосконалювати свою точність на основі накопиченого досвіду класифікації. Модель також враховує зворотний зв'язок від користувачів для покращення якості роботи.

Процес попередньої обробки текстових даних включає складний комплекс операцій з підготовки матеріалів до аналізу. Система реалізує багатоетапну процедуру очищення тексту, що включає видалення технічних артефактів, нормалізацію форматування та уніфікацію представлення спеціальних символів. Особлива увага приділяється збереженню структурної цілісності навчальних матеріалів та коректній обробці математичних формул і технічних діаграм. Найголовнішим етапом є також лематизація тексту та приведення термінів до канонічної форми. Система враховує особливості технічної термінології різних дисциплін та забезпечує коректну обробку аббревіатур та спеціальних позначень. Реалізований механізм дозволяє зберігати семантичні зв'язки між термінами при їх нормалізації. Процес попередньої обробки також включає етап виявлення та корекції можливих помилок у вхідних даних [5].

Механізм зберігання та управління даними в системі реалізований з використанням сучасних підходів до організації інформаційних сховищ.

Архітектура бази даних оптимізована для ефективного зберігання великих обсягів текстової інформації та забезпечення швидкого доступу до класифікованих матеріалів. Система використовує розподілену структуру зберігання, що дозволяє масштабувати ресурси відповідно до зростання обсягу даних. Особлива увага приділяється забезпеченню цілісності інформації та створенню надійної системи резервного копіювання. Реалізований механізм індексації дозволяє здійснювати швидкий пошук по великих масивах документів. Система також забезпечує ефективне управління метаданими та підтримує версійність документів. Вагомим аспектом є реалізація механізмів контролю доступу та захисту конфіденційної інформації.

Інтерфейсна частина системи розроблена з урахуванням сучасних вимог до ергономіки та зручності використання. Графічний інтерфейс забезпечує інтуїтивно зрозумілу навігацію та швидкий доступ до основних функцій системи. Особлива увага приділена розробці зручних інструментів для завантаження та попереднього перегляду навчальних матеріалів. Система надає користувачам гнучкі можливості для налаштування параметрів класифікації та перегляду результатів аналізу. Реалізований інтерфейс дозволяє ефективно працювати з великими наборами документів та забезпечує зручні засоби для групової обробки матеріалів. Головним аспектом є також наявність розвинених засобів візуалізації результатів класифікації та статистичного аналізу. Система підтримує різні режими відображення інформації та адаптується до різних розмірів екранів. На рисунку 1.1 показано сценарії використання розробленої системи.

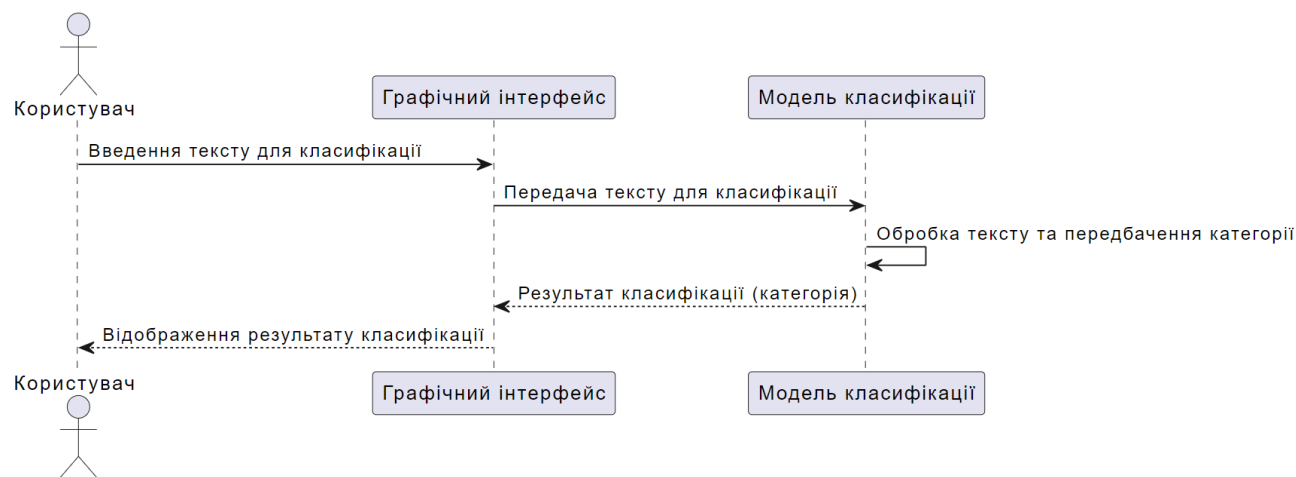


Рисунок 1.1 - Use Case діаграма - класифікація тексту

На діаграмі відображено основні сценарії використання автоматизованої системи класифікації навчальних текстів. Ключовими діями є завантаження тексту, його попередня обробка, аналіз за допомогою нейронної мережі та надання результату класифікації користувачу. Порівняння характеристик алгоритмів класифікації показано у таблиці 1.1. Вказані ролі включають адміністратора системи та звичайного користувача, наприклад, викладача чи студента.

Таблиця 1.1 – Порівняльна характеристика алгоритмів класифікації.

Алгоритм	Переваги	Недоліки
SVM	Висока точність, ефективний з невеликими наборами даних	Складний для великих даних, потребує ретельної підготовки
Naive Bayes	Простота, швидкість, добре працює з текстовими даними	Низька точність для складних даних
RNN	Урахування контексту, ефективний для послідовностей	Тривалий час навчання, потреба у великій кількості даних
BERT	Висока точність, врахування контексту	Велика обчислювальна вартість
Decision Trees	Інтерпретація результатів	Схильність до перенавчання

Система моніторингу та аналізу продуктивності забезпечує безперервний контроль за роботою всіх компонентів комплексу. Реалізований механізм збору метрик дозволяє відслідковувати ключові показники ефективності на різних рівнях системи та своєчасно виявляти потенційні проблеми. Особлива увага приділяється аналізу якості класифікації та відстеженню випадків некоректної категоризації матеріалів. Система автоматично генерує детальні звіти про роботу алгоритмів класифікації та забезпечує можливість глибокого аналізу причин виникнення помилок. Головним компонентом є також механізм прогнозування навантаження та автоматичного масштабування ресурсів. Реалізована підсистема сповіщень дозволяє оперативно інформувати адміністраторів про критичні події та відхилення від нормальної роботи. Комплекс включає також засоби для довгострокового аналізу тенденцій та планування розвитку системи.

Модуль забезпечення інформаційної безпеки реалізує комплексний підхід до захисту даних та контролю доступу. Архітектура безпеки побудована з урахуванням сучасних вимог до захисту освітніх інформаційних систем та включає багаторівневу систему аутентифікації та авторизації користувачів. Особлива увага приділяється захисту персональних даних та конфіденційної інформації, що може міститися в навчальних матеріалах. Система забезпечує детальне протоколювання всіх дій користувачів та операцій з документами. Реалізований механізм шифрування гарантує безпеку даних при зберіганні та передачі. Головним компонентом є також система виявлення та запобігання втручанням у роботу системи. Комплекс включає засоби регулярного аудиту безпеки та оцінки вразливостей [6].

Система аналізу якості навчальних матеріалів забезпечує комплексну оцінку текстів, що надходять на класифікацію. Реалізований механізм дозволяє автоматично визначати рівень складності матеріалу, його структурну цілісність та відповідність освітнім стандартам. Особлива увага приділяється виявленню логічних зв'язків між різними частинами тексту та оцінці послідовності викладення матеріалу. Система здатна аналізувати повноту розкриття теми та визначати наявність необхідних елементів навчального матеріалу. Вагомим

аспектом є також оцінка термінологічної насиченості тексту та коректності використання спеціальних термінів. Приклад наведено у таблиці 1.2. Реалізований алгоритм дозволяє виявляти потенційні проблеми в структурі навчальних матеріалів та надавати рекомендації щодо їх вдосконалення. Механізм аналізу також враховує специфічні вимоги різних навчальних дисциплін.

Таблиця 1.2 – Основні компоненти структури навчального тексту.

Компонент тексту	Опис	Значення для класифікації
Заголовок	Відображає головну тему тексту	Допомагає у визначенні ключових категорій
Основний текст	Містить основну інформацію	Основний вміст для аналізу
Термінологія	Спеціалізовані слова та фрази	Визначає зв'язок з дисципліною
Посилання	Список джерел	Додатковий контекст
Діаграми/ілюстрації	Візуальна інформація	Може впливати на семантичний аналіз

Підсистема управління версіями забезпечує ефективний контроль за змінами в навчальних матеріалах та моделях класифікації. Реалізована архітектура дозволяє зберігати повну історію модифікацій документів та відслідковувати еволюцію системи класифікації. Особлива увага приділяється забезпеченню можливості порівняння різних версій матеріалів та аналізу внесених змін. Система підтримує механізм гілкування для паралельної розробки та вдосконалення алгоритмів класифікації. Головним компонентом є також засоби для автоматичного об'єднання змін та вирішення конфліктів. Реалізований механізм дозволяє ефективно організувати колективну роботу над

вдосконаленням системи. Комплекс включає також засоби для документування змін та ведення детального журналу модифікацій.

Механізм взаємодії з зовнішніми системами забезпечує інтеграцію розробленого комплексу з існуючою освітньою інфраструктурою. Реалізований набір програмних інтерфейсів дозволяє здійснювати безперебійний обмін даними з системами управління навчальним процесом та електронними бібліотеками. Особлива увага приділяється забезпеченню сумісності форматів даних та протоколів взаємодії. Система підтримує можливість автоматичної синхронізації контенту та забезпечує цілісність даних при передачі між різними платформами. Головним аспектом є реалізація механізмів обробки помилок та автоматичного відновлення з'єднання при збоях. Комплекс також включає засоби моніторингу якості інтеграційних процесів та оптимізації продуктивності обміну даними. Інтеграційний модуль забезпечує можливість розширення функціональності системи за рахунок підключення нових зовнішніх сервісів.

Підсистема статистичного аналізу забезпечує глибоке дослідження процесів класифікації та формування аналітичних звітів. Реалізований комплекс алгоритмів дозволяє виявляти приховані закономірності в розподілі навчальних матеріалів за дисциплінами та оцінювати якість роботи системи класифікації. Особлива увага приділяється аналізу динаміки змін у структурі навчальних матеріалів та виявленню нових тенденцій. Система забезпечує можливість проведення порівняльного аналізу різних методів класифікації та оцінки їх ефективності. Вагомим компонентом є також механізм прогнозування майбутніх тенденцій у розвитку навчальних дисциплін. Реалізовані алгоритми дозволяють формувати рекомендації щодо оптимізації процесу класифікації та вдосконалення навчальних матеріалів.

Система адаптивного навчання нейронної мережі забезпечує постійне вдосконалення алгоритмів класифікації на основі накопиченого досвіду. Реалізований механізм дозволяє автоматично коригувати параметри моделі при виявленні нових закономірностей у структурі навчальних матеріалів. Особлива увага приділяється збалансованому використанню історичних даних та нової

інформації при навчанні моделі. Система включає механізми валідації результатів навчання та запобігання перенавчанню нейронної мережі. Головним аспектом є також забезпечення стабільності роботи системи під час процесу адаптації. Реалізований підхід дозволяє зберігати ефективність класифікації при зміні характеристик вхідних даних. Комплекс підтримує можливість паралельного тестування декількох варіантів моделей для вибору оптимальної конфігурації.

Модуль семантичного аналізу забезпечує глибоке розуміння змісту навчальних матеріалів та їх взаємозв'язків. Розроблена система використовує передові методи обробки природної мови для виявлення концептуальних зв'язків між різними частинами тексту. Особлива увага приділяється аналізу контексту використання спеціальних термінів та їх взаємозв'язку з різними дисциплінами. Система здатна виявляти неявні тематичні залежності та формувати семантичні мережі знань. Вагомим компонентом є механізм розпізнавання та обробки метафор і специфічних мовних конструкцій, характерних для наукових текстів. Реалізований алгоритм дозволяє враховувати різні рівні абстракції при аналізі навчальних матеріалів. Комплекс також включає засоби для визначення рівня складності тексту та його відповідності цільовій аудиторії. Попередня обробка тексту розбита на етапи зображення в таблиці 1.3.

Таблиця 1.3 – Етапи попередньої обробки тексту.

Етап	Опис	Інструменти/методи
Токенізація	Розбиття тексту на окремі слова або речення	NLTK, SpaCy
Видалення стоп-слів	Виключення загальних слів, таких як "і", "або"	Стандартні списки стоп-слів
Лематизація	Приведення слів до базової форми	WordNet, SpaCy
Видалення шуму	Видалення спеціальних символів та зайвих пробілів	Регулярні вирази
TF-IDF векторизація	Перетворення тексту у числовий формат	Sklearn, Gensim

Розроблена система приділяє особливу увагу питанням етичності використання навчальних матеріалів. Реалізовані механізми дозволяють відслідковувати дотримання авторських прав та ліцензійних угод при роботі з текстами. Система забезпечує автоматичне виявлення випадків плагіату та неправомірного використання контенту. Головним аспектом є також врахування принципів академічної доброчесності при аналізі та класифікації матеріалів. Система надає інструменти для перевірки оригінальності текстів та генерування звітів про потенційні порушення. Користувачі мають можливість налаштовувати параметри чутливості системи до виявлення плагіату відповідно до специфіки навчальної дисципліни та вимог освітньої установи.

Система також враховує особливості роботи з мультимедійним контентом, який часто використовується в сучасних навчальних матеріалах. Розроблені алгоритми дозволяють аналізувати та класифікувати не лише текстову інформацію, але й зображення, відео та аудіо файли. Система здатна розпізнавати ключові об'єкти на зображеннях та відео, транскрибувати аудіо в текст та виконувати семантичний аналіз отриманої інформації. Це дозволяє забезпечити комплексну обробку мультимедійних навчальних матеріалів та їх коректну класифікацію за відповідними дисциплінами. Особлива увага приділяється оптимізації швидкості обробки мультимедійного контенту та забезпеченню високої якості розпізнавання.

Найголовнішою складовою системи є модуль генерації рекомендацій щодо вдосконалення навчальних матеріалів. Реалізовані алгоритми аналізують структуру, зміст та стиль викладення тексту, виявляючи потенційні недоліки та області для покращення. Система надає деталізовані рекомендації щодо оптимізації логічної організації матеріалу, забезпечення зв'язності та послідовності викладу, а також покращення читабельності та зрозумілості тексту. Особлива увага приділяється аналізу використання термінології та виявленню випадків надмірної складності або неоднозначності. Механізм рекомендацій також враховує специфіку цільової аудиторії та може адаптувати

поради відповідно до рівня підготовки та особливостей сприйняття інформації студентами.

1.2 Аналіз універсальних можливостей системи, інтеграція, масштабованість та аналітика.

Система забезпечує можливість інтеграції з зовнішніми джерелами знань та онтологіями для підвищення точності та повноти класифікації навчальних матеріалів. Реалізовані механізми дозволяють підключати додаткові бази даних термінів, визначень та концептуальних зв'язків, специфічних для кожної навчальної дисципліни. Це дозволяє розширювати та уточнювати базу знань системи, забезпечуючи більш глибоке розуміння контексту та семантики навчальних текстів. Система також підтримує можливість автоматичного оновлення та синхронізації з зовнішніми джерелами знань, що гарантує актуальність та відповідність використовуваних онтологій сучасному стану розвитку відповідних галузей знань.

Для забезпечення високої надійності та відмовостійкості системи реалізована розподілена архітектура з використанням технологій контейнеризації та оркестрації. Компоненти системи розгортаються у вигляді незалежних мікросервісів, що забезпечує гнучкість, масштабованість та стійкість до збоїв. Використання контейнерів дозволяє швидко розгортати та оновлювати компоненти системи, забезпечуючи консистентність середовища виконання. Оркестрація контейнерів здійснюється за допомогою сучасних платформ, таких як Kubernetes, що надає можливості автоматичного управління життєвим циклом мікросервісів, балансування навантаження та забезпечення високої доступності системи. Реалізовані механізми моніторингу та логування дозволяють відслідковувати стан системи в режимі реального часу та оперативно реагувати на потенційні проблеми.

Система також включає розвинені засоби аналітики та візуалізації даних для надання користувачам інформативних та інтуїтивно зрозумілих звітів про

результати класифікації та характеристики навчальних матеріалів. Реалізовані інтерактивні дашборди дозволяють переглядати статистичні показники, розподіл матеріалів за дисциплінами, тенденції зміни структури та змісту текстів з часом. Користувачі мають можливість налаштовувати параметри візуалізації, обирати різні типи графіків та діаграм, а також здійснювати фільтрацію та сегментацію даних за різними критеріями. Система також підтримує можливість експорту аналітичних звітів у різних форматах для подальшого використання та інтеграції з іншими інструментами аналізу даних.

Для забезпечення ефективної комунікації та колаборації між користувачами системи реалізовані вбудовані механізми обміну повідомленнями та спільної роботи над навчальними матеріалами. Користувачі мають можливість створювати тематичні обговорення, ділитися коментарями та пропозиціями щодо вдосконалення матеріалів. Система підтримує можливість призначення завдань та відстеження статусу їх виконання, що дозволяє ефективно координувати роботу команди над створенням та оновленням навчального контенту. Реалізовані механізми сповіщень та нагадувань забезпечують своєчасну інформованість учасників процесу та мінімізують ризики пропуску головних подій або дедлайнів.

Окрема увага приділена питанням безпеки та конфіденційності при роботі з навчальними матеріалами. Система забезпечує захист персональних даних користувачів відповідно до вимог законодавства та галузевих стандартів. Реалізовані механізми шифрування та анонімізації даних гарантують конфіденційність інформації при її зберіганні та передачі. Система також підтримує гнучкі налаштування прав доступу, що дозволяє обмежувати види дій, які можуть виконувати різні категорії користувачів. Вбудовані механізми аудиту та моніторингу активності користувачів дозволяють відслідковувати потенційні загрози безпеці та вживати превентивних заходів для їх нейтралізації.

Для спрощення процесу розгортання та конфігурації системи розроблено спеціалізовані інструменти автоматизації та управління інфраструктурою. Система надає можливість розгортання в хмарних середовищах з використанням

технологій Infrastructure as Code (IaC). Це дозволяє швидко і consistently створювати необхідні обчислювальні ресурси, мережеві налаштування та сховища даних за допомогою декларативних конфігураційних файлів. Автоматизація процесу розгортання мінімізує ризики помилок та забезпечує відтворюваність середовища в різних контекстах. Система також підтримує механізми автоматичного масштабування ресурсів залежно від навантаження, що дозволяє оптимізувати витрати та забезпечувати стабільну продуктивність навіть при значних коливаннях інтенсивності використання.

Розроблена система приділяє значну увагу зручності використання та інтуїтивності інтерфейсів для різних категорій користувачів. Графічний інтерфейс системи розроблений з урахуванням принципів ергономіки та орієнтований на забезпечення максимальної продуктивності роботи користувачів. Елементи управління та навігації організовані у логічні групи, що спрощує пошук необхідних функцій та мінімізує кількість необхідних дій для виконання типових завдань. Система надає контекстні підказки та рекомендації, що допомагають користувачам швидко освоїти основні можливості та ефективно використовувати інструменти класифікації та аналізу навчальних матеріалів. Інтерфейс системи адаптивний та забезпечує коректне відображення на різних типах пристроїв, включаючи настільні комп'ютери, ноутбуки, планшети та смартфони.

Найголовнішим аспектом розробленої системи є її здатність до інтеграції з існуючими системами управління навчанням (LMS) та платформами електронного навчання. Реалізовані механізми синхронізації даних та уніфікації форматів обміну інформацією дозволяють безперешкодно передавати результати класифікації та метадані навчальних матеріалів між різними системами. Це забезпечує можливість централізованого управління навчальним контентом та спрощує процеси оновлення та розповсюдження матеріалів серед студентів та викладачів. Система також підтримує стандартизовані протоколи автентифікації та авторизації, що дозволяє реалізувати єдину точку входу та

забезпечити безшовний досвід роботи користувачів при переході між різними компонентами освітньої екосистеми.

Для підтримки багатомовності та локалізації система включає механізми автоматичного визначення мови навчальних матеріалів та генерації відповідних метаданих. Це дозволяє коректно класифікувати та обробляти тексти на різних мовах, забезпечуючи можливість використання системи в міжнародному освітньому контексті. Система також підтримує можливість ручного визначення мови матеріалу та асоціювання додаткових лінгвістичних атрибутів, таких як діалект, регіон походження або спеціалізована термінологія. Ці дані використовуються для покращення якості лінгвістичного аналізу та забезпечення коректної обробки мовно-специфічних особливостей тексту. Інтерфейс системи та документація доступні на декількох мовах, що спрощує її використання для користувачів з різних країн та мовних середовищ.

Система надає розвинені можливості для проведення експериментів та тестування нових алгоритмів класифікації та аналізу навчальних матеріалів. Реалізована підсистема управління експериментами дозволяє створювати ізольовані середовища для розробки та оцінки ефективності різних підходів до обробки текстів. Дослідники та розробники мають можливість налаштовувати параметри моделей, модифікувати набори даних для навчання та тестування, а також збирати детальні метрики продуктивності. Система забезпечує можливість версіонування експериментів та відстеження змін у конфігурації, що спрощує відтворюваність результатів та полегшує співпрацю між членами дослідницької команди. Результати експериментів можуть бути легко інтегровані в основну систему після ґрунтовної перевірки та валідації.

Окрема увага приділена можливості розширення функціональності системи за рахунок підтримки плагінів та модулів розширення. Архітектура системи передбачає наявність стандартизованих інтерфейсів та протоколів взаємодії, що дозволяє незалежним розробникам створювати додаткові компоненти для вирішення специфічних завдань або розширення базових можливостей класифікації та аналізу текстів. Плагіни можуть реалізовувати нові

алгоритми обробки даних, додаткові формати вхідних та вихідних даних, спеціалізовані інструменти візуалізації та аналітики тощо. Система забезпечує гнучкі механізми управління життєвим циклом плагінів, включаючи їх встановлення, оновлення та видалення без необхідності перезапуску або внесення змін до основних компонентів системи.

Для забезпечення ефективного використання обчислювальних ресурсів та оптимізації продуктивності система включає механізми кешування та попередньої обробки даних. Часто використовувані результати класифікації та метадані навчальних матеріалів зберігаються в розподіленому кеші, що дозволяє швидко отримувати необхідну інформацію без повторного виконання ресурсоємних операцій аналізу та обробки тексту. Система також передбачає можливість виконання попередньої обробки та індексації навчальних матеріалів у фоновому режимі, що мінімізує затримки при обробці запитів користувачів та забезпечує оптимальний час відгуку системи. Розроблені алгоритми балансування навантаження дозволяють динамічно розподіляти задачі класифікації та аналізу між доступними обчислювальними вузлами, забезпечуючи максимальну утилізацію ресурсів та мінімізуючи ризики виникнення "вузьких місць" у процесі обробки даних.

Система також приділяє значну увагу питанням масштабованості та здатності до обробки великих обсягів навчальних матеріалів. Реалізована архітектура передбачає можливість горизонтального масштабування шляхом додавання додаткових обчислювальних вузлів та розподілу навантаження між ними. Це дозволяє плавно нарощувати потужність системи в міру збільшення кількості користувачів та обсягів оброблюваних даних. Крім того, система підтримує розподілене зберігання даних та реплікацію, що забезпечує високу доступність та стійкість до збоїв окремих компонентів. Розроблені механізми забезпечують узгодженість та цілісність даних при виконанні паралельних операцій та в умовах мережових затримок або тимчасової недоступності окремих вузлів.

Для забезпечення довгострокової підтримки та розвитку системи передбачені механізми моніторингу та управління життєвим циклом компонентів. Регулярний збір телеметрії та логів дозволяє відстежувати функціонування системи в режимі реального часу, виявляти потенційні проблеми та оперативно реагувати на них. Розроблені процедури резервного копіювання та відновлення даних гарантують збереження цілісності інформації навіть у випадку катастрофічних збоїв або інцидентів безпеки. Система також підтримує можливість автоматичного розгортання оновлень та виправлень без необхідності повного перезапуску сервісів, що мінімізує час простою та забезпечує безперервність надання послуг користувачам.

Окрема увага приділена документуванню та навчанню користувачів роботі з системою. Розроблено детальну користувацьку документацію, яка охоплює всі аспекти функціонування системи, включаючи опис інтерфейсів, керівництва з налаштування та адміністрування, а також приклади використання окремих функцій та компонентів. Документація доступна в онлайн-форматі з можливістю швидкого пошуку та навігації. Крім того, система включає вбудовані довідкові матеріали та підказки, які надають користувачам контекстну інформацію та рекомендації щодо використання окремих інструментів та функцій. Для новачків передбачено інтерактивні навчальні матеріали та тренінги, що дозволяють швидко освоїти основні концепції та принципи роботи з системою.

Розроблена система спроектована з урахуванням можливості інтеграції з іншими компонентами інформаційно-освітнього середовища навчального закладу. Передбачені стандартизовані інтерфейси та протоколи обміну даними, що дозволяють налаштувати взаємодію з системами управління навчальним процесом, репозиторіями навчальних об'єктів, платформами електронного навчання та іншими інформаційними системами. Така інтеграція дозволяє забезпечити безперешкодний обмін інформацією між різними компонентами, уникнути дублювання даних та забезпечити консистентність і актуальність навчальних матеріалів у межах єдиного інформаційного простору навчального закладу.

Не менш головним аспектом є забезпечення сумісності розробленої системи з існуючими стандартами та специфікаціями у галузі електронного навчання. Система підтримує широко вживані формати представлення навчальних матеріалів, такі як SCORM, xAPI, IMS Content Packaging тощо. Це дозволяє безперешкодно імпортувати та експортувати навчальний контент між різними системами та платформами, забезпечуючи можливість повторного використання та обміну матеріалами між закладами освіти. Крім того, система забезпечує підтримку стандартів метаданих, таких як Dublin Core та LOM, що полегшує опис, пошук та управління навчальними ресурсами.

Для забезпечення безперервного вдосконалення та актуалізації системи передбачені механізми зворотного зв'язку та збору відгуків від користувачів. Інтегровані інструменти дозволяють користувачам повідомляти про помилки, вносити пропозиції щодо покращення функціональності або надавати загальні коментарі та побажання. Ця інформація збирається та аналізується розробниками системи для визначення пріоритетних напрямків подальшого розвитку та вдосконалення. Регулярні опитування та фокус-групи з залученням різних категорій користувачів допомагають виявити їхні реальні потреби та очікування, що дозволяє краще адаптувати систему до вимог освітнього процесу та підвищити задоволеність користувачів.

Окреме місце займають питання ліцензування та інтелектуальної власності при роботі з навчальними матеріалами. Система передбачає гнучкі механізми управління правами доступу та ліцензіями на використання контенту. Це дозволяє чітко визначати умови використання, модифікації та розповсюдження матеріалів відповідно до вимог правовласників та освітньої політики закладу. Система забезпечує автоматизоване відстеження використання ліцензованого контенту та генерування звітів про дотримання ліцензійних умов. У випадку виявлення потенційних порушень система може автоматично сповіщати адміністраторів та правовласників для вжиття відповідних заходів.

Для спрощення процесу міграції даних з існуючих систем та підготовки навчальних матеріалів до імпорту розроблено спеціалізовані інструменти та

утиліти. Ці засоби дозволяють автоматизувати процес вилучення даних з різнорідних джерел, їх трансформацію відповідно до внутрішніх форматів та структур системи, а також виконувати попередню обробку та валідацію. Підтримуються різні формати вхідних даних, включаючи текстові документи, електронні таблиці, бази даних та архіви. Розроблені механізми дозволяють виявляти та усувати дублікати, виправляти помилки форматування та структурувати дані відповідно до заданих шаблонів. Процес міграції може виконуватись як одноразово, так і на регулярній основі для підтримки актуальності даних.

Головним елементом розробленої системи є механізми управління якістю та забезпечення достовірності результатів класифікації. Передбачено набір автоматизованих тестів та процедур валідації, які дозволяють перевіряти коректність роботи окремих компонентів системи та оцінювати загальну ефективність класифікації на основі еталонних наборів даних. Регулярне виконання цих тестів дозволяє оперативно виявляти потенційні відхилення та помилки в роботі алгоритмів та моделей. У випадку виявлення проблем передбачені механізми автоматичного сповіщення відповідальних осіб та ініціювання процедур діагностики та усунення несправностей. Крім того, система дозволяє проводити ручну перевірку результатів класифікації експертами предметної області для додаткового підтвердження їх коректності.

Розроблена система приділяє значну увагу питанням продуктивності та оптимізації використання обчислювальних ресурсів. Реалізовані механізми розподіленої обробки даних та паралельного виконання завдань дозволяють ефективно використовувати наявні апаратні ресурси та скорочувати час обробки великих обсягів інформації. Система автоматично масштабує обчислювальні потужності залежно від поточного навантаження, забезпечуючи оптимальний баланс між продуктивністю та вартістю експлуатації. Передбачені механізми кешування часто використовуваних результатів та проміжних даних, що дозволяє мінімізувати витрати часу на повторні обчислення. Крім того, система

використовує техніки ледачої ініціалізації та відкладеної обробки для оптимізації використання пам'яті та зменшення накладних витрат.

Окремої уваги заслуговують можливості системи щодо генерації звітів та аналітичних матеріалів на основі результатів класифікації та аналізу навчальних текстів. Розроблений модуль звітності дозволяє формувати різноманітні види документів, включаючи зведені таблиці, графіки, діаграми та текстові описи. Користувачі мають можливість налаштовувати вміст та структуру звітів відповідно до своїх потреб, обираючи необхідні параметри та показники. Генерація звітів може виконуватись як в інтерактивному режимі, так і за розкладом, з автоматичною доставкою результатів на електронну пошту або збереженням у визначених місцях. Крім того, система дозволяє експортувати звіти у різних форматах, таких як PDF, Excel, CSV тощо, для подальшого використання та інтеграції з іншими інструментами.

Для забезпечення ефективної комунікації та обміну знаннями між користувачами системи передбачені вбудовані механізми колаборації та спільної роботи. Користувачі мають можливість створювати віртуальні робочі простори, де вони можуть обмінюватися ідеями, коментарями та пропозиціями щодо вдосконалення навчальних матеріалів. Система підтримує функції чату, форумів та обговорень, що дозволяє користувачам оперативно вирішувати питання та отримувати зворотній зв'язок від колег. Крім того, передбачені механізми сповіщень та нагадувань, які інформують користувачів про необхідні події, такі як надходження нових матеріалів, завершення процесу класифікації, необхідність перегляду або схвалення документів тощо. Ці функції сприяють підвищенню ефективності командної роботи та покращують координацію діяльності в рамках освітнього процесу.

Найголовнішим аспектом функціонування розробленої системи є її здатність до інтеграції з хмарними платформами та сервісами. Архітектура системи дозволяє розгорнути її компоненти в різних хмарних середовищах, таких як Amazon Web Services, Microsoft Azure або Google Cloud Platform. Це забезпечує гнучкість та масштабованість, дозволяючи динамічно виділяти

необхідні обчислювальні ресурси та сховища даних залежно від поточних потреб. Крім того, використання хмарних технологій спрощує процеси резервного копіювання, відновлення та забезпечення високої доступності системи. Інтеграція з хмарними сервісами також відкриває можливості для використання додаткових функцій, таких як машинне навчання, обробка великих даних, інтелектуальний аналіз тексту тощо. Можливості інтеграції з іншими системами зображені в таблиці 1.4.

Таблиця 1.4 – Можливість інтеграції з іншими системами.

Система	Тип інтеграції	Переваги
LMS	Автоматичне додавання класифікованих текстів	Спрощення управління контентом
Електронні бібліотеки	Імпорт/експорт навчальних матеріалів	Централізація даних
Хмарні сервіси	Розширення обчислювальних можливостей	Масштабованість
API	Доступ до функцій системи ззовні	Гнучкість інтеграції
Платформи аналітики	Передача результатів класифікації	Поглиблений аналіз

Безпека та захист даних є ключовими аспектами при взаємодії з хмарними платформами. Для забезпечення конфіденційності інформації впроваджені потужні алгоритми шифрування, які діють як під час передачі даних, так і при їх зберіганні у хмарному середовищі. Доступ до системних ресурсів контролюється через передові методи автентифікації та авторизації, що забезпечують вхід тільки для користувачів із належними повноваженнями. Система також включає функції спостереження та записування користувацької активності, що сприяє швидкому виявленню та реагуванню на потенційні загрози безпеці. Для підтримки належного рівня захисту проводяться систематичні перевірки безпеки та тести на вразливість, що дозволяє своєчасно виявляти та усувати можливі слабкі місця в системі.

Для спрощення процесу розробки та тестування нових функцій та алгоритмів класифікації передбачено використання віртуалізованих середовищ та контейнерних технологій. Розробники мають можливість швидко розгортати ізольовані екземпляри системи з необхідними залежностями та конфігураціями, що дозволяє експериментувати з різними підходами та налаштуваннями без впливу на робочі середовища. Контейнерні технології, такі як Docker, дозволяють створювати портативні та відтворювані середовища, що спрощує процес розгортання та масштабування системи.

1.3 Дослідження існуючих прототипів системи

Автоматизована класифікація текстів є однією з ключових задач у сфері обробки природної мови (NLP). Її мета полягає у розподілі текстових документів або фрагментів на заздалегідь визначені категорії чи класи, що має критичне значення для ефективної роботи з великими обсягами неструктурованої текстової інформації. Протягом останніх десятиліть було розроблено та впроваджено різноманітні підходи та алгоритми для вирішення цієї задачі, кожен з яких має свої переваги та обмеження [7-9]. Розглянемо більш детально еволюцію методів автоматизованої класифікації текстів та особливості існуючих прототипів систем.

Одними з найбільш ранніх та широко вживаних методів класифікації текстів були підходи, засновані на класичних алгоритмах машинного навчання, таких як наївний баєсів класифікатор, метод опорних векторів (SVM) та дерева рішень. Ці методи ґрунтуються на ручному виділенні ознак з текстів, які потім використовуються для навчання моделей класифікації. Наївний баєсів класифікатор, наприклад, розраховує ймовірність приналежності документа до кожного з класів на основі частоти появи окремих слів, спираючись на припущення про умовну незалежність ознак. Метод опорних векторів, натомість, намагається знайти оптимальну гіперплощину, яка розділяє простір ознак на області, що відповідають різним класам. Попри відносну простоту та

обчислювальну ефективність, ці методи мають обмеження щодо врахування контексту та семантики текстів, а також потребують ретельної попередньої обробки та очищення даних.

З розвитком технологій глибокого навчання та збільшенням обчислювальних потужностей, фокус досліджень у сфері автоматизованої класифікації текстів почав зміщуватися у бік використання нейронних мереж. Зокрема, рекурентні нейронні мережі (RNN), такі як LSTM (Long Short-Term Memory – Рисунок 1.2) та GRU (Gated Recurrent Unit), показали значний успіх у роботі з послідовними даними, якими і є тексти.

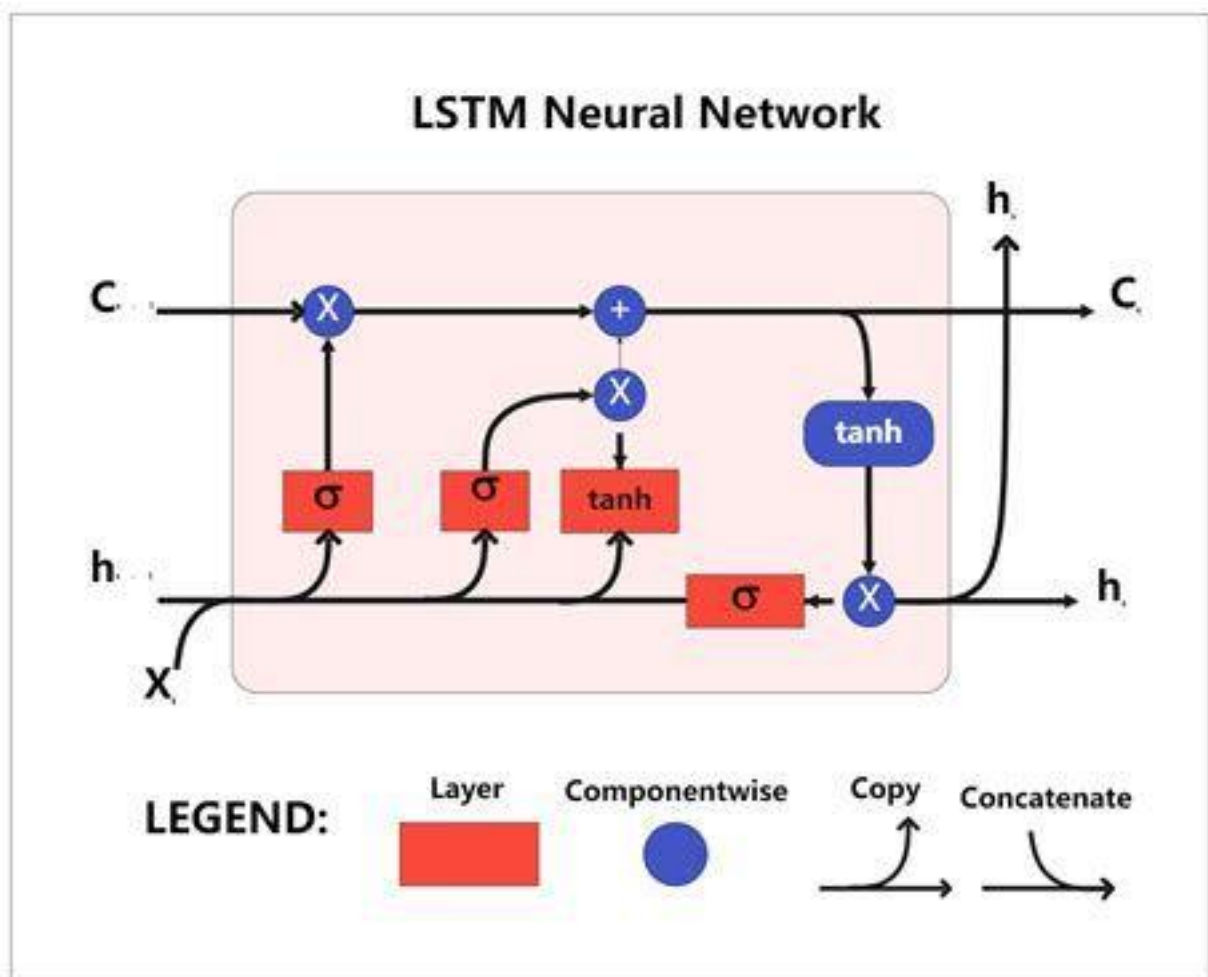


Рисунок 1.2 – Передбачення послідовності LSTM.

Ці архітектури здатні враховувати контекст та послідовність появи слів, "запам'ятовуючи" інформацію з попередніх кроків обробки [10]. Це дозволяє їм ефективно обробляти довгі текстові послідовності та виявляти складні залежності між словами та реченнями. Однак, RNN мають проблеми з паралельною обробкою даних та можуть страждати від проблеми "зникаючого градієнту" при роботі з дуже довгими послідовностями. В таблиці 1.5 зображене порівняння метрик різних моделей.

Таблиця 1.5 – Порівняння метрик ефективності різних моделей.

Модель	Точність (%)	Повнота (%)	F1-Score (%)	Час обробки (сек)
SVM	85	80	82	2
Naive Bayes	78	70	74	1
RNN	90	88	89	15
BERT	93	92	92.5	45
Decision Trees	80	75	77	3

Архітектура трансформерів, представлена у 2017 році, здійснила революційний вплив на галузь обробки природної мови. В основі цієї технології лежить механізм самоуваги, який надає моделі здатність автоматично оцінювати значущість елементів вхідної послідовності при формуванні відповіді. Визначні представники цієї архітектури - BERT та GPT - встановили нові стандарти ефективності у численних завданнях обробки природної мови, зокрема у класифікації текстів. Ці моделі відзначаються здатністю опрацьовувати розлогі текстові фрагменти, розуміти контекстуальні зв'язки та семантичні нюанси, а також можуть адаптуватися до специфічних галузей через додаткове навчання. Хоча трансформери стали провідною технологією для розв'язання більшості задач NLP, їх застосування вимагає потужної обчислювальної інфраструктури та масштабних наборів даних для навчання.

Окрім розвитку нових архітектур та алгоритмів, значна увага у дослідженнях автоматизованої класифікації текстів приділяється методам попередньої обробки та представлення текстової інформації. Традиційно, тексти

представляються у вигляді векторів ознак, таких як частота появи слів (Bag of Words), TF-IDF (Term Frequency-Inverse Document Frequency) або вектори embedding'ів слів (наприклад, Word2Vec (Рисунок 1.3 - 1.4) або GloVe). Ці представлення дозволяють перетворити неструктуровані текстові дані у числові вектори фіксованої розмірності, які можуть бути використані як вхід для алгоритмів машинного навчання. Однак, вони мають обмеження щодо врахування порядку слів та контексту, а також страждають від проблеми розрідженості та високої розмірності. Більш сучасні підходи, такі як контекстно-залежні embedding'и (наприклад, BERT або ELMo), дозволяють отримати більш змістовні та контекстно-залежні представлення слів та речень, що покращує якість класифікації.

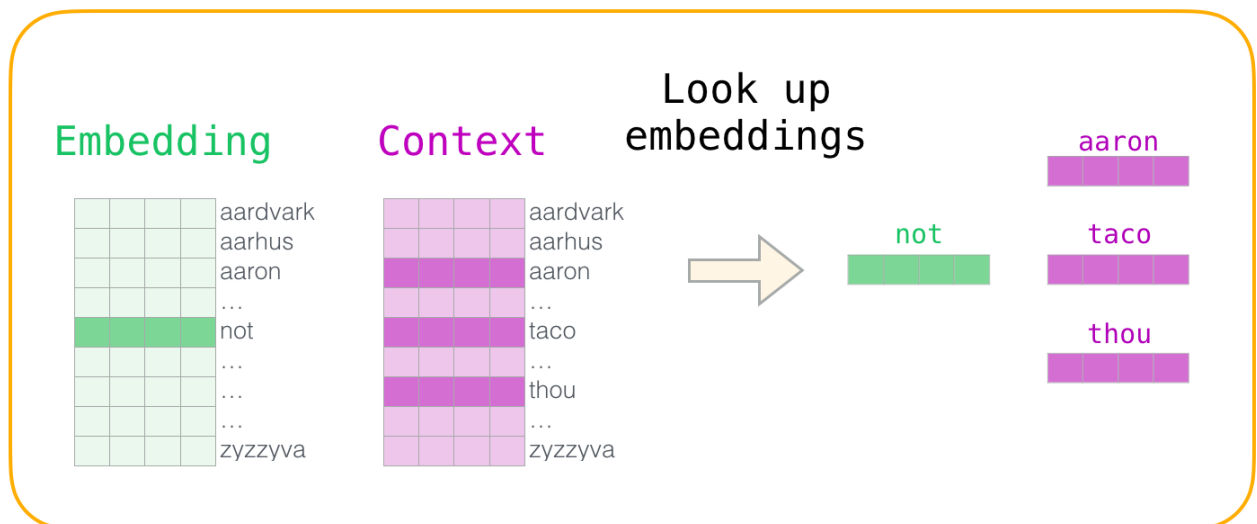
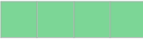


Рисунок 1.3 – Пошук вхідного слова в процес навчання Word2Vec

input word	output word	target	input • output	sigmoid()	Error
not 	thou 	1	0.2	0.55	0.45
not 	aaron 	0	-1.11	0.25	-0.25
not 	taco 	0	0.74	0.68	-0.68

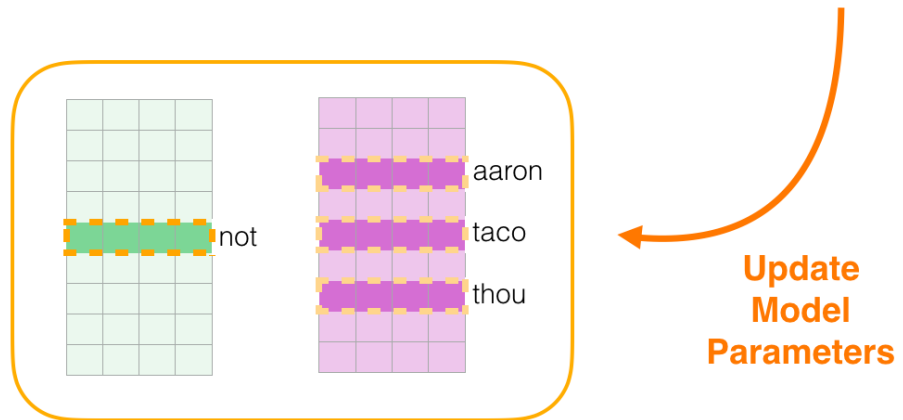


Рисунок 1.4 – Оцінка помилок в процесі навчання Word2Vec

Ще одним вагомим аспектом у розробці систем автоматизованої класифікації текстів є вибір та підготовка навчальних даних. Якість та репрезентативність навчального набору даних безпосередньо впливає на ефективність роботи моделі класифікації. Тому, значні зусилля спрямовуються на збір, анотування та очищення великих обсягів текстових даних для різних предметних областей та мов. Особливо актуальною є проблема створення якісних навчальних даних для спеціалізованих доменів, таких як медицина, юриспруденція або технічна документація, де необхідна висока точність класифікації та врахування специфічної термінології. Для вирішення цієї проблеми часто застосовуються методи напіваавтоматичного анотування, краудсорсингу та активного навчання, які дозволяють зменшити витрати на ручне маркування даних експертами.

Окрім власне алгоритмів класифікації, головною складовою систем автоматизованої обробки текстів є механізми попередньої обробки та нормалізації текстових даних. Це включає такі етапи, як токенізація (розбиття тексту на окремі слова чи фрази), видалення стоп-слів (часто вживаних слів, які

не несуть смислового навантаження), приведення слів до нормальної форми (лематизація або стемінг), а також виділення іменованих сутностей (назв людей, організацій, локацій тощо). Ці кроки дозволяють зменшити розмірність та складність вхідних даних, позбутися "шуму" та виділити найбільш інформативні елементи тексту. Крім того, часто застосовуються методи нормалізації регістру, видалення розділових знаків та спеціальних символів, а також обробки числових даних та дат. Вибір конкретних методів попередньої обробки залежить від специфіки задачі та особливостей текстових даних.

Оцінювання ефективності систем автоматичної класифікації текстів становить одну з основних складностей у процесі їх розробки. Для визначення якості роботи моделей традиційно застосовується набір базових метрик, включаючи ассигасу (загальну точність), recall (повноту), precision (точність) та F1-score, які вимірюють як загальну кількість вірно класифікованих документів, так і співвідношення правильних та помилкових класифікацій для кожної категорії. Проте ці стандартні показники можуть бути недостатньо інформативними при роботі з незбалансованими наборами даних, де спостерігається значна різниця у кількості прикладів між класами. Для більш комплексного аналізу використовуються додаткові інструменти оцінки: confusion matrix, ROC-крива та показник AUC. Важливо також відзначити, що вибір методів оцінювання повинен враховувати особливості конкретної задачі та предметної області, а використання крос-валідації забезпечує більш об'єктивну та достовірну оцінку продуктивності системи.

Ще однією головною проблемою при розробці систем автоматизованої класифікації текстів є забезпечення можливості інтерпретації та пояснення отриманих результатів. Багато сучасних моделей, особливо тих, що базуються на глибокому навчанні, працюють за принципом "чорного ящика", коли внутрішня логіка прийняття рішень прихована від користувача. Це може викликати недовіру до результатів роботи системи, особливо у критично важливих сферах, таких як медична діагностика або юридичний аналіз. Тому, значна увага приділяється розробці методів інтерпретації та візуалізації роботи моделей

класифікації, які дозволяють зрозуміти, на основі яких ознак та закономірностей приймаються рішення. Наприклад, застосовуються методи візуалізації активацій нейронів, виділення важливих слів та фраз, а також побудови пояснювальних моделей, які дозволяють отримати людино-зрозумілі обґрунтування результатів класифікації.

Окремим напрямком досліджень у сфері автоматизованої класифікації текстів є розробка методів для роботи з специфічними типами текстових даних, такими як соціальні медіа, медична документація, юридичні контракти тощо. Ці типи даних часто характеризуються наявністю специфічної термінології, скорочень, помилок та неформального стилю написання, що ускладнює застосування стандартних методів обробки та класифікації. Тому, розробляються спеціалізовані моделі та методи попередньої обробки, які враховують особливості конкретних доменів та типів текстових даних. Наприклад, для аналізу соціальних медіа часто застосовуються методи сентимент-аналізу, виявлення спаму та фейкових новин, а також методи нормалізації неформальної лексики та помилок. У медичній сфері, натомість, важливо враховувати специфічну термінологію, аббревіатури та конвенції запису діагнозів та процедур.

Ще одним перспективним напрямком досліджень є розробка методів для класифікації текстів за множинними категоріями або ієрархічними структурами. У багатьох практичних задачах, текстові документи можуть одночасно належати до кількох категорій або мати ієрархічні залежності між класами. Наприклад, науковий текст може стосуватися кількох дисциплін або підрозділів певної галузі знань. Традиційні методи класифікації, які передбачають віднесення документа до одного з наперед визначених класів, не можуть ефективно враховувати такі складні залежності. Тому, розробляються методи багатоміткової (multi-label) класифікації, які дозволяють присвоювати документу кілька релевантних категорій одночасно, а також методи ієрархічної класифікації, які враховують структурні залежності між класами.

Окрім розробки нових моделей та алгоритмів, значна увага приділяється проблемам ефективного навчання та адаптації існуючих моделей до нових доменів та задач. Зокрема, актуальними є методи трансферного навчання (transfer learning), які дозволяють використовувати знання, отримані при навчанні моделі на одному наборі даних, для покращення якості роботи на іншому, схожому наборі. Це особливо важливо у випадках, коли доступні навчальні дані для конкретної задачі обмежені або відсутні. Наприклад, модель, навчена на великому корпусі текстів з Вікіпедії, може бути адаптована для класифікації новинних статей або медичних записів шляхом донавчання на невеликому наборі специфічних даних. Такий підхід дозволяє значно зменшити витрати на збір та анотування навчальних даних, а також покращити якість роботи моделей на нових доменах.

Головним аспектом розробки систем автоматизованої класифікації текстів є їх практичне застосування та інтеграція у існуючі робочі процеси та інформаційні системи. Зокрема, значна увага приділяється проблемам масштабованості, продуктивності та зручності використання таких систем. Для забезпечення ефективної роботи з великими обсягами текстових даних, застосовуються методи розподілених обчислень та паралельної обробки, наприклад, з використанням платформ Hadoop або Spark. Крім того, розробляються зручні користувацькі інтерфейси та API для інтеграції моделей класифікації у існуючі системи документообігу, пошуку та аналізу інформації.

Окремо варто згадати про методи оцінки та валідації моделей класифікації текстів. Для оцінки якості роботи моделі зазвичай використовують такі метрики, як точність (accuracy), повнота (recall), влучність (precision) та F1-міра (F1-score). Ці метрики дозволяють кількісно оцінити здатність моделі правильно розподіляти тексти по категоріях та виявляти помилки класифікації. Однак, при інтерпретації цих метрик важливо враховувати специфіку конкретної задачі та збалансованість класів у навчальних даних.

Окрім кількісних метрик, вагомим аспектом оцінки моделей є їх здатність до узагальнення та роботи на нових, раніше не бачених даних. Для цього

зазвичай використовують методи крос-валідації, такі як k-fold або стратифікована k-fold крос-валідація. Суть цих методів полягає у розбитті вихідного набору даних на k частин, послідовному навчанні та тестуванні моделі на кожній з цих частин, та усередненні результатів. Це дозволяє отримати більш надійні та незміщені оцінки якості моделі та її здатності працювати на нових даних.

Ще одним вагомим моментом є інтерпретація та аналіз помилок класифікації. Для цього часто використовують матрицю невідповідностей (confusion matrix), яка показує, скільки текстів кожного класу було віднесено до кожної з категорій. Аналіз цієї матриці дозволяє виявити, які класи найчастіше плутаються між собою, та зрозуміти потенційні причини цих помилок. Це може бути пов'язано з семантичною близькістю категорій, недостатньою репрезентативністю навчальних даних або особливостями роботи самої моделі.

Варто також згадати про методи інтерпретації та пояснення роботи моделей класифікації текстів. Особливо це актуально для нейромережових моделей, які часто сприймаються як "чорні скриньки". Для вирішення цієї проблеми розробляються спеціальні методи, такі як LIME (Local Interpretable Model-agnostic Explanations) або SHAP (SHapley Additive exPlanations). Ці підходи дозволяють виявити, які саме ознаки або слова в тексті найбільше вплинули на рішення моделі, та надати зрозумілі для людини пояснення класифікації [11, с. 108-109].

Окремо варто відзначити важливість якісної попередньої обробки текстових даних перед подачею їх на вхід моделі. Це включає в себе такі етапи, як токенізація (розбиття тексту на окремі слова чи фрази), фільтрація стоп-слів (видалення часто вживаних слів, які не несуть смислового навантаження), стемінг або лематизація (приведення слів до їх базової форми), а також векторизація (перетворення текстів у числові вектори ознак). Від якості виконання цих кроків безпосередньо залежить ефективність роботи моделі класифікації.

Варто також згадати про те, що вибір конкретного методу або моделі класифікації залежить від специфіки задачі, обсягу та характеристик доступних даних, а також вимог до інтерпретованості та швидкодії рішення. В деяких випадках прості та швидкі методи, такі як наївний байєсів класифікатор, можуть давати цілком прийнятні результати. В інших ситуаціях може знадобитися потужність сучасних нейромережових моделей, таких як BERT або GPT. В будь-якому випадку, вибір має ґрунтуватися на ретельному аналізі задачі та експериментальному порівнянні різних підходів.

Окремої уваги заслуговує питання побудови ефективних пайплайнів обробки та аналізу текстів, які б поєднували в собі етапи попередньої обробки, векторизації, вибору ознак, навчання та валідації моделі. Такі пайплайни дозволяють автоматизувати та стандартизувати процес роботи з текстовими даними, спростити експериментування з різними конфігураціями та параметрами моделей, а також забезпечити відтворюваність результатів. Для побудови таких пайплайнів зазвичай використовують спеціалізовані бібліотеки та фреймворки, такі як scikit-learn або Apache Spark.

На завершення, варто відзначити, що автоматизована класифікація текстів є активною областю досліджень на стику машинного навчання, обробки природної мови та прикладної лінгвістики. З розвитком нових моделей та підходів, таких як few-shot learning, метанавчання чи генеративні моделі, можливості автоматизованого аналізу та категоризації текстової інформації постійно розширюються. В той же час, з'являються нові виклики, пов'язані з обробкою все більших обсягів неструктурованих даних, забезпеченням приватності та безпеки чутливої інформації, а також подоланням упередженостей та дискримінації в моделях. Все це робить сферу автоматизованої класифікації текстів надзвичайно перспективною та динамічною областю для подальших досліджень та практичних застосувань.

Висновок

У цьому розділі проаналізовано сучасний стан задач автоматизованої класифікації текстів, зокрема для освітніх потреб. Описано особливості об'єкта

дослідження – системи, яка враховує специфіку навчальних текстів, таких як наукова термінологія та структурні особливості. З’ясовано, що ключовими аспектами функціонування системи є забезпечення високої точності класифікації, врахування міждисциплінарних зв’язків та адаптивність до нових даних. Визначено, що використання глибокого навчання та алгоритмів обробки природної мови є оптимальним підходом для вирішення задач класифікації.

РОЗДІЛ 2. РОЗРОБКА ФУНКЦІОНАЛЬНОЇ СТРУКТУРИ ПРОГРАМИ

2.1 Опис системи, яка буде використовуватись

Під час розробки автоматизованої системи класифікації навчальних текстів дисциплін з використанням нейронних мереж було створено рішення, яке спрощує та прискорює обробку навчальних матеріалів. Ця система об'єднує в собі низку функцій, призначених для автоматизації класифікації навчальних текстів, підвищення точності та ефективності їх обробки, а також використання передових технологій нейронних мереж [12]. Основне призначення такої системи – полегшення роботи з навчальними текстами за допомогою інтелектуальних алгоритмів та гнучких інтерфейсних рішень.

Інтерфейс та функціональні модулі. У системі передбачено інтуїтивно зрозумілий графічний інтерфейс, який дозволяє користувачам легко взаємодіяти з програмою. Модуль інтерфейсу завантажує зручний графічний інтерфейс для подальшої роботи, забезпечуючи можливість введення або імпорту текстів. Введення навчальних текстів здійснюється через спеціалізовані модулі: модуль введення тексту або модуль імпорту тексту, що ілюстровано на рисунку 2.1.

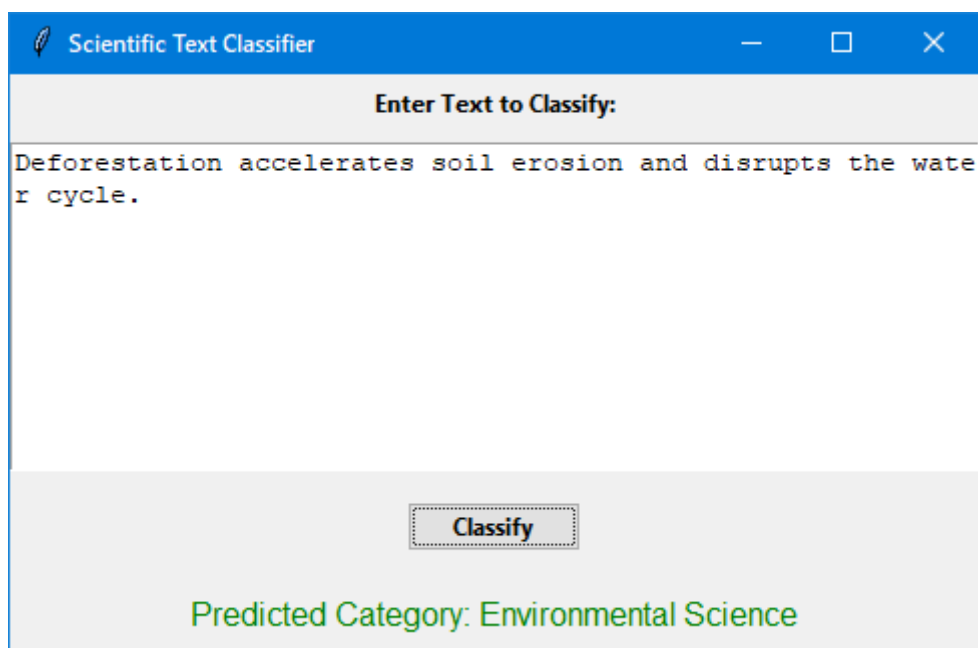


Рисунок 2.1 - Інтерфейс системи з введеним текстом

Нейронна мережа. Один із ключових компонентів системи – модуль нейронної мережі, який завантажує попередньо навчену модель, оптимізовану для конкретних навчальних дисциплін. Ця модель здатна розпізнавати та класифікувати текстові дані, що значно полегшує аналіз навчальних матеріалів різних профілів [13]. Нейронна мережа адаптована для роботи з текстами, пов'язаними з навчальними дисциплінами, і в процесі її навчання використовувалися великі масиви даних, що сприяє підвищенню точності класифікації.

Процес аналізу та класифікації. Після того, як текст внесений у систему, модуль аналізу активує процес його обробки нейронною мережею. Цей модуль відповідає за перетворення та попередню обробку тексту, аналізуючи його відповідно до навчальної дисципліни. Далі текст передається в модуль класифікації, який генерує результат на основі отриманих даних і виводить його у зручному для користувача форматі.

Автоматизовані процеси. Однією з важливих особливостей цієї системи є автоматизація процесів класифікації текстів. Завдяки використанню нейронної мережі, система здатна самостійно аналізувати та розподіляти тексти по відповідних категоріях. Така автоматизація значно скорочує час, необхідний для обробки великих обсягів текстових даних, що є важливим для навчальних закладів, де обробка інформації зазвичай займає значну кількість часу.

Переваги використання системи. Основною перевагою цієї системи є її здатність значно підвищити ефективність роботи з навчальними матеріалами. Завдяки використанню машинного навчання, система забезпечує швидку та точну класифікацію текстів, що полегшує доступ до необхідної інформації. Крім того, вона спрощує процес внесення текстів завдяки гнучким інструментам імпорту та введення [14].

Інноваційним аспектом цієї системи є можливість самонавчання нейронної мережі, що забезпечує постійне вдосконалення її продуктивності. У міру накопичення нових даних система зможе аналізувати тексти з більшою точністю

та пропонувати більш релевантні класифікації. Це відкриває перспективи для використання системи не лише в поточному вигляді, але й для подальшого розвитку й адаптації під інші дисципліни або мовні специфіки.

Архітектура та технології. Система розроблена з використанням сучасних технологій та підходів до побудови програмних рішень. Вона базується на модульній архітектурі, що дозволяє гнучко розширювати та модифікувати окремі компоненти без впливу на роботу всієї системи. Для реалізації інтерфейсу використовується фреймворк Tkinter, який забезпечує крос-платформність та зручність розробки графічних інтерфейсів на мові Python.

Модуль нейронної мережі реалізований з використанням бібліотеки TensorFlow, яка є потужним інструментом для створення та навчання моделей глибокого навчання. TensorFlow надає широкі можливості для роботи з нейронними мережами, включаючи різні архітектури, методи оптимізації та функції активації. Для попередньої обробки та векторизації текстів використовуються засоби бібліотеки scikit-learn, зокрема клас TfidfVectorizer.

Навчання моделі. Процес навчання нейронної мережі є критично важливим для забезпечення високої якості класифікації текстів. Для цього використовується спеціально підготовлений набір навчальних даних, який містить приклади текстів з різних дисциплін, розмічених відповідними мітками класів. Ці дані розбиваються на тренувальну та тестову вибірки, що дозволяє контролювати якість навчання моделі та оцінювати її здатність до узагальнення.

Модель нейронної мережі будується з використанням архітектури багатошарового перцептрону (MLP) з додатковими шарами регуляризації та нормалізації. Для навчання моделі використовується алгоритм зворотного поширення помилки та методи оптимізації, такі як стохастичний градієнтний спуск (SGD) або Adam. Процес навчання контролюється за допомогою спеціальних метрик, таких як точність, повнота та F1-міра, які дозволяють оцінити якість класифікації на кожній епісі навчання.

Попередня обробка текстів. Перед подачею на вхід нейронної мережі тексти проходять етап попередньої обробки та векторизації [15]. Це включає в

себе такі кроки, як токенізація (розбиття тексту на окремі слова), фільтрація стоп-слів (видалення часто вживаних слів, які не несуть смислового навантаження), лематизація (приведення слів до їх початкової форми) та векторизація з використанням методу TF-IDF.

Метод TF-IDF (Term Frequency-Inverse Document Frequency) дозволяє перетворити текст у числовий вектор ознак, враховуючи частоту появи слів у конкретному документі та їх рідкість у всьому корпусі текстів. Це дозволяє виділити найбільш значущі та інформативні слова для кожного класу та зменшити вплив загальноновживаних або малоінформативних термінів. Отримані вектори ознак подаються на вхід нейронної мережі для подальшої обробки та класифікації.

Оцінка якості класифікації. Для оцінки якості роботи моделі використовуються спеціальні метрики, такі як точність, повнота, влучність та F1-міра. Точність (accuracy) показує частку правильно класифікованих текстів серед усіх проаналізованих. Повнота (recall) відображає здатність моделі знаходити всі релевантні тексти для кожного класу. Влучність (precision) характеризує частку дійсно релевантних текстів серед усіх віднесених моделлю до даного класу. F1-міра є гармонійним середнім між повнотою та влучністю і дає збалансовану оцінку якості класифікації.

Для отримання надійних оцінок якості використовуються методи крос-валідації, зокрема стратифікована k-fold крос-валідація. При цьому набір даних розбивається на k частин, і модель послідовно навчається та тестується на кожній з них. Результати усереднюються, що дозволяє отримати більш стабільні та незміщені оцінки якості класифікації.

Аналіз помилок та інтерпретація результатів. Головним етапом в процесі розробки та використання системи є аналіз помилок класифікації та інтерпретація отриманих результатів. Для цього використовується матриця невідповідностей (confusion matrix), яка показує, як тексти різних класів розподілилися по передбаченим моделлю категоріям. Аналіз цієї матриці

дозволяє виявити, які класи найчастіше плутаються між собою, та зрозуміти потенційні причини помилок.

Для інтерпретації рішень моделі та визначення найбільш значущих ознак використовуються методи, такі як LIME або SHAP. Ці підходи дозволяють визначити, які саме слова або фрази в тексті мали найбільший вплив на прийняте рішення, та надати зрозумілі для користувача пояснення щодо логіки класифікації.

Можливості для розширення та вдосконалення. Розроблена система має значний потенціал для подальшого розширення та вдосконалення. Одним з напрямків є збільшення обсягу та різноманітності навчальних даних, що дозволить покрити ширший спектр дисциплін та покращити точність класифікації. Також можливе додавання нових функціональних модулів, наприклад, для автоматичного виділення ключових термінів або побудови семантичних карт предметних областей.

Іншим перспективним напрямком є експериментування з різними архітектурами нейронних мереж та методами попередньої обробки текстів. Зокрема, використання більш потужних моделей, таких як згорткові або рекурентні нейронні мережі, може дозволити краще враховувати контекст та послідовність слів у текстах. Застосування техніки трансферного навчання або дообучення на специфічних даних конкретних дисциплін також може покращити якість класифікації.

Інтеграція з іншими системами та сервісами. Головним аспектом практичного використання розробленої системи є можливість її інтеграції з існуючими платформами та сервісами електронного навчання. Завдяки модульній архітектурі та використанню стандартних форматів даних, система може бути легко інтегрована з системами управління навчанням (LMS), репозиторіями навчальних матеріалів, бібліотечними каталогами тощо.

Така інтеграція дозволить автоматизувати процеси класифікації та розподілу навчальних матеріалів по відповідних категоріях безпосередньо в рамках існуючих систем. Це спростить навігацію по навчальним ресурсам,

пошук релевантних матеріалів та організацію доступу до них для студентів та викладачів. Крім того, інтеграція з системами аналітики та звітності дозволить отримувати агреговані дані щодо використання навчальних матеріалів та якості їх класифікації.

Перспективи впровадження та використання. Розроблена система автоматизованої класифікації навчальних текстів має широкі перспективи впровадження та використання в сфері освіти. Вона може стати цінним інструментом для викладачів, методистів, розробників навчальних програм та адміністраторів освітніх закладів. Автоматизація процесів класифікації та розподілу навчальних матеріалів дозволить заощадити час та ресурси, підвищити ефективність пошуку та використання релевантної інформації, а також забезпечити більш персоналізований підхід до навчання.

Особливо актуальним використання такої системи може бути в умовах дистанційного та змішаного навчання, коли обсяги електронних навчальних матеріалів стрімко зростають, а їх ефективна організація та структурування стають критично головним. Крім того, система може бути корисною для створення та підтримки репозиторіїв відкритих освітніх ресурсів, забезпечуючи їх якісну класифікацію та полегшуючи пошук релевантних матеріалів для викладачів та студентів з різних закладів освіти.

Подальші кроки та розвиток системи. Для успішного впровадження та розвитку розробленої системи автоматизованої класифікації навчальних текстів необхідно здійснити ряд подальших кроків. По-перше, головне провести ґрунтовне тестування системи на реальних даних з різних навчальних дисциплін та оцінити її ефективність у порівнянні з традиційними методами ручної класифікації. Це дозволить виявити потенційні недоліки та напрямки для вдосконалення.

По-друге, необхідно розробити детальну документацію та інструкції для користувачів, які б дозволили легко інтегрувати систему в існуючі освітні процеси та забезпечити її ефективне використання викладачами та адміністра

торами. Це може включати в себе опис процесу інсталяції, налаштування параметрів, роботи з інтерфейсом, інтерпретації результатів тощо.

По-третє, головне забезпечити регулярне оновлення та розширення бази навчальних даних, на основі яких проводиться навчання моделі. Це дозволить системі адаптуватися до змін у навчальних програмах, появи нових дисциплін та тематик, а також враховувати зворотній зв'язок від користувачів щодо якості класифікації.

По-четверте, доцільно розглянути можливості інтеграції розробленої системи з іншими інструментами та сервісами, які використовуються в освітньому процесі. Це може включати в себе системи управління навчанням (LMS), електронні бібліотеки, платформи для створення та поширення відкритих освітніх ресурсів тощо. Така інтеграція дозволить створити більш цілісну та ефективну екосистему для роботи з навчальними матеріалами.

По-п'яте, головне забезпечити належний рівень безпеки та конфіденційності при роботі з навчальними даними. Це передбачає використання надійних механізмів автентифікації та авторизації доступу, шифрування даних при передачі та зберіганні, а також дотримання відповідних регуляторних вимог щодо захисту персональних даних студентів та викладачів.

Нарешті, для успішного впровадження розробленої системи необхідно провести навчання та інформування потенційних користувачів щодо її можливостей та переваг. Це може включати в себе проведення презентацій, вебінарів, створення навчальних матеріалів та відеоінструкцій, а також надання консультаційної підтримки користувачам у процесі експлуатації системи.

У довгостроковій перспективі розвиток розробленої системи може включати в себе розширення її функціональності, наприклад, додавання можливостей автоматичного реферування текстів, виділення ключових концептів та взаємозв'язків між ними, а також генерації рекомендацій щодо послідовності вивчення матеріалів на основі їх змістовної близькості. Також перспективним напрямком є адаптація системи для роботи з різними мовами та

предметними областями, що дозволить розширити сферу її застосування та охопити більш широку аудиторію користувачів.

Створена система автоматичної класифікації освітніх матеріалів, що базується на технологіях штучного інтелекту та машинного навчання, відкриває шлях до впровадження більш досконалих інтелектуальних рішень в освітньому менеджменті. Така система може слугувати фундаментом для розробки просунутих освітніх платформ, які автоматично пристосовують навчальний контент та його подачу до особистих потреб кожного учня. Більше того, це створює можливості для впровадження автоматизованих систем оцінювання, які здатні надавати індивідуалізований зворотний зв'язок та конкретні рекомендації щодо навчального процесу.

Таким чином, розроблена система автоматизованої класифікації навчальних текстів з використанням нейронних мереж представляє собою потужний та перспективний інструмент для підвищення ефективності та якості управління освітнім контентом. Її впровадження та подальший розвиток можуть суттєво оптимізувати процеси створення, організації та використання навчальних матеріалів, а також сприяти персоналізації та адаптивності освітнього процесу відповідно до потреб та особливостей кожного студента. При цьому, успішність реалізації цих можливостей залежатиме від тісної співпраці розробників, викладачів, адміністраторів та інших зацікавлених сторін, а також від врахування етичних, правових та соціальних аспектів використання технологій штучного інтелекту в освіті.

2.2 Опис структури автоматизованої системи класифікації текстів

Програмний модуль для класифікації навчальних текстів, що використовує нейромережеві технології, представлений у вигляді програми з графічним інтерфейсом, де основна взаємодія відбувається через візуальні елементи на екрані монітора. Головне призначення цього інструменту - оптимізувати та

пришвидшити процес обробки та категоризації текстових матеріалів з різних навчальних дисциплін.

Для забезпечення виконання всіх етапів процесу класифікації в складі програмного модуля з класифікації навчальних текстів дисциплін з використанням нейронних мереж враховані функціональні складові елементи, взаємозв'язок яких між собою [16-18]. На рисунку 2.2 наведено порядок дій для навчання моделі нейронної мережі, яка використовується у системі.

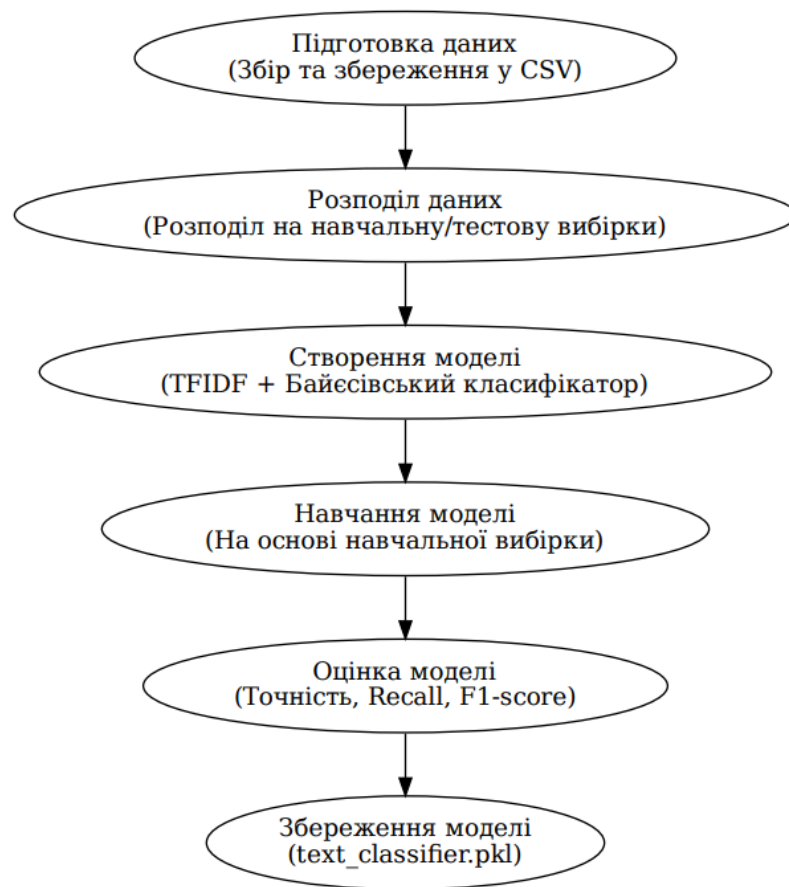


Рисунок 2.2 – Алгоритм навчання моделі нейронної мережі

На діаграмі відображено етапи підготовки моделі нейронної мережі: збирання даних, їх розподіл на навчальні та тестові вибірки, вибір архітектури нейронної мережі, налаштування гіперпараметрів, навчання моделі, валідація та оцінка точності. Після завершення навчання модель інтегрується в основну систему.

Система складається з кількох взаємопов'язаних модулів: графічний інтерфейс забезпечує взаємодію з користувачем; попередньо натренована нейронна модель завантажується для роботи з конкретними дисциплінами; функції введення та імпорту тексту забезпечують надходження даних до системи; аналітичний модуль здійснює обробку та трансформацію тексту через нейронну мережу; модуль класифікації опрацьовує отримані результати та представляє їх користувачу через інтерфейс програми.

Програмний модуль складається з графічного інтерфейсу розробленого в середовищі Qt Designer, нейронної мережі з пакету TensorFlow та скрипту написаного на мові програмування Python.

Qt Designer служить основним інструментом для створення графічного користувацького інтерфейсу, пропонуючи широкий спектр можливостей для розробки. За допомогою "Панелі віджетів" розробники можуть обирати та використовувати різноманітні елементи інтерфейсу, включаючи ComboBox для випадаючих списків, LineEdit для полів введення та PushButton для кнопок. Кожен елемент інтерфейсу базується на відповідному класі бібліотеки Qt та має унікальний набір налаштовуваних параметрів, які можна модифікувати через спеціалізований "Редактор властивостей". Особливо важливою функцією Qt Designer є можливість візуального налаштування взаємодії між елементами інтерфейсу через систему сигналів і слотів, що дозволяє, наприклад, налаштувати зміну доступності одного елемента інтерфейсу залежно від стану іншого, такого як CheckBox [19].

Пакет TensorFlow версії 2.0 та новіші, завдяки інтеграції з Keras API, надає зручний та інтуїтивно зрозумілий інструментарій для розробки та використання нейронних мереж. В основі TensorFlow лежить потужний механізм обчислень, що спеціалізується на обробці багатовимірних масивів даних (тензорів) з використанням паралельних обчислень та імперативного підходу. Така архітектура забезпечує паралельну активацію вузлів нейронної мережі, що імітує принцип одночасної роботи нейронів у біологічному мозку.

Для написання функціоналу програмного модуля було обрано мову програмування Python [20-22]. Python — інтерпретована об'єктно-орієнтована мова програмування високого рівня зі строгою динамічною типізацією. Структури даних високого рівня разом із динамічною семантикою та динамічним зв'язуванням роблять її привабливою для швидкої розробки програм, а також як засіб поєднування наявних компонентів. Python підтримує модулі та пакети модулів, що сприяє модульності та повторному використанню коду. Інтерпретатор Python та стандартні бібліотеки доступні як у скомпільованій, так і у вихідній формі на всіх основних платформах. В мові програмування Python підтримується кілька парадигм програмування, зокрема: об'єктно-орієнтована, процедурна, функціональна та аспектно-орієнтована.

Python вирізняється своєю потужною вбудованою бібліотекою, яка надає широкий спектр можливостей для розробки. Вона включає інструменти для взаємодії з різноманітними інтернет-протоколами та форматами, дозволяючи створювати HTTP-сервери та клієнти, обробляти електронні листи та працювати з XML. Завдяки спеціальним модулям для взаємодії з операційною системою розробники можуть створювати програми, що працюють на різних платформах. Бібліотека також містить засоби для роботи з регулярними виразами, різними кодуваннями тексту, мультимедіа, криптографією, архівами, серіалізацією даних та модульним тестуванням. На рисунку 2.3 показано діаграму класів, яка відображає основні об'єкти системи та їх взаємозв'язки.

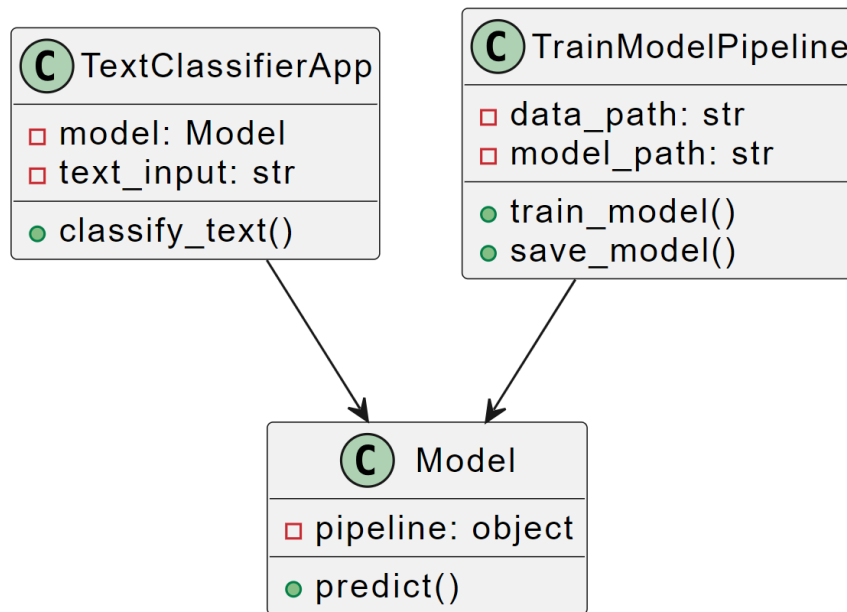


Рисунок 2.3 - Діаграма класів розробленої системи

На діаграмі наведено ключові класи системи, їх атрибути та методи. Наприклад, клас "Користувач" може містити інформацію про логін, роль та доступи, клас "Модель" — методи для навчання, класифікації та оцінки точності. Взаємозв'язки між класами відображають структуру програмного коду.

Програмна архітектура представляє собою організаційну будову програмного забезпечення чи обчислювальної системи, що охоплює її складові компоненти, їхні зовнішні характеристики та взаємозв'язки між усіма елементами системи.

Система складається з п'яти основних компонентів: графічний інтерфейс, що забезпечує візуальну взаємодію з користувачем; компонент введення тексту, що дозволяє імпортувати та вводити текстові дані; модуль нейронної мережі, розроблений на Python з використанням TensorFlow, який виконує класифікацію текстів; компонент обробки та виведення результатів, що відображає результати аналізу; та програмне ядро, яке координує взаємодію між усіма компонентами та забезпечує загальну функціональність системи.

2.3 Архітектура, функціональність та інтеграція програмного модуля класифікації текстів.

Розроблений програмний модуль представлений у вигляді віконного додатку, створеного на мові Python з використанням інструментів Qt Designer. Взаємодія з програмою здійснюється через стандартні пристрої введення - клавіатуру та мишу, а результати відображаються на екрані монітора. При розробці графічного інтерфейсу користувача (GUI) застосовуються два основні підходи: об'єктно-орієнтований користувацький інтерфейс (ООУІ) та інтерфейси, що орієнтовані на конкретні додатки.

Вікно форми має фіксований розмір і дорівнює 700*500 пікселів, а також включена можливість зміни розміру вікна.

Головна форма модуля (рис. 2.4) складається з трьох областей:

введення тексту;

завантаження нейронної мережі;

вивід результату.

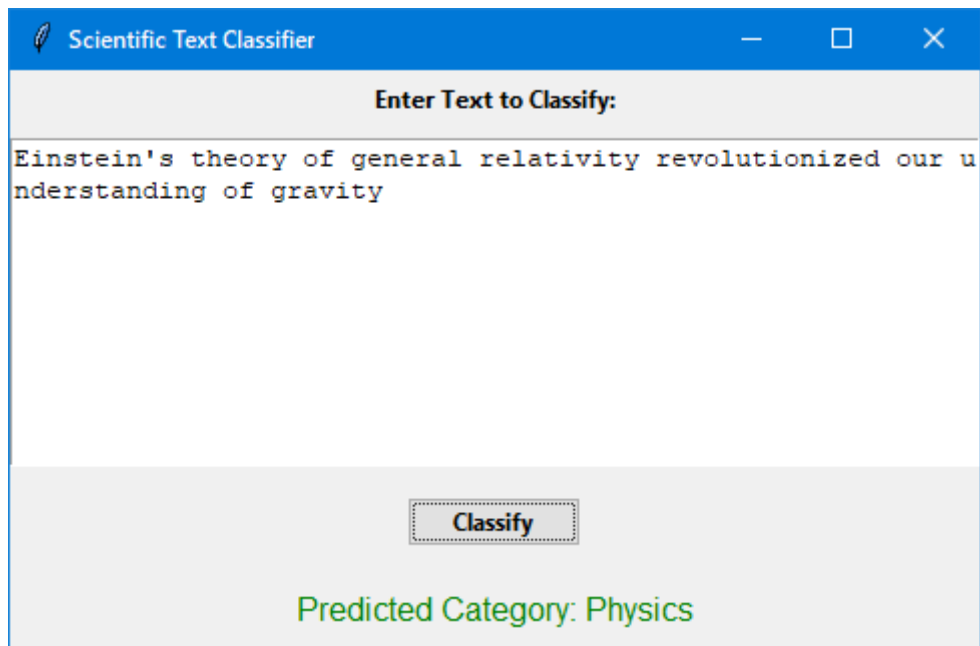


Рисунок 2.4 - Інтерфейс користувача

В області введення тексту користувач вводить або імпортує текст до програми. В області завантаження нейронної мережі користувач вносить підготовлену модель нейронної мережі розробленої в пакеті TensorFlow. Модель нейронної мережі має у собі останній стан нейронної мережі та контрольні точки необхідні для її роботи. Збережена модель може бути редагована у будь-який час.

В області виводу результату знаходяться результати роботи програмного модуля.

Після запуску користувач вводить або імпортує текст до області вводу, та натискає кнопку Generate. Після чого текст передається до нейронної моделі проводиться обробка, пошук ключових слів та прогнозування.

Після обробки тексту результати роботи передаються до області виводу. Результати роботи програми надані на рис. 2.5.

Classification Report:				
	precision	recall	f1-score	support
Biology	0.86	0.67	0.75	9
Chemistry	0.83	0.71	0.77	14
Computer Science	0.83	0.71	0.77	7
Engineering	0.86	0.75	0.80	8
Environmental Science	1.00	1.00	1.00	10
Physics	0.50	1.00	0.67	6
accuracy			0.80	54
macro avg	0.81	0.81	0.79	54
weighted avg	0.83	0.80	0.80	54
Model saved to text_classifier.pkl				

Рисунок 2.5 - Інтерфейс системи з виведеними результатами

Вагомим аспектом розроблюваної системи є механізм збереження та управління моделями нейронних мереж. Для кожної навчальної дисципліни створюється окрема модель, яка навчається на відповідному наборі текстів та зберігається у окремому файлі. Це дозволяє гнучко керувати процесом класифікації, додавати нові дисципліни та оновлювати існуючі моделі без необхідності перенавчання всієї системи.

При запуску програмного модуля користувач має можливість обрати потрібну дисципліну зі списку доступних моделей. Після вибору дисципліни відповідна модель нейронної мережі завантажується з файлу та ініціалізується для подальшого використання. Такий підхід забезпечує швидкий старт роботи з програмою та дозволяє оперативно перемикатися між різними дисциплінами. Приклад роботи програмного модуля та його компонентів зображено на рисунку 2.6.

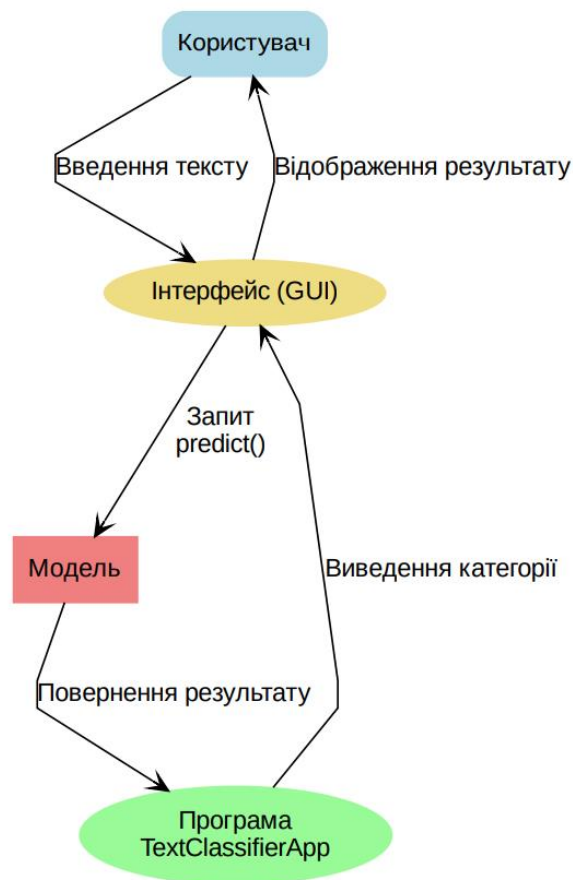


Рисунок 2.6 - Порядок взаємодії компонентів

Діаграма відображає процеси взаємодії між основними компонентами системи, такими як модуль обробки текстів, модуль класифікації та база даних. Наприклад, при завантаженні тексту запускається попередня обробка, результати передаються до модуля класифікації, а підсумкові дані зберігаються в базі.

Для забезпечення ефективного використання обчислювальних ресурсів та оптимізації швидкодії, програмний модуль реалізує механізми багатопоточної обробки даних. Зокрема, процеси введення та імпорту тексту, його попередньої обробки, а також класифікації за допомогою нейронної мережі виконуються у окремих потоках. Це дозволяє уникнути блокування інтерфейсу користувача під час виконання ресурсоємних операцій та забезпечує більш плавну та responsive роботу програми.

Окрім цього, реалізовані механізми кешування та збереження проміжних результатів обробки текстів. Це дозволяє уникнути повторного виконання однакових операцій при повторній класифікації того ж самого тексту або при незначних змінах у вхідних даних. Система зберігає векторні представлення текстів та результати їх класифікації у спеціалізованому кеші, що дозволяє суттєво пришвидшити обробку запитів користувача та зменшити навантаження на обчислювальні ресурси.

Для забезпечення можливості аналізу та інтерпретації результатів класифікації, програмний модуль надає розширені засоби візуалізації та статистичної обробки даних. Зокрема, користувач може переглянути детальну інформацію щодо розподілу ймовірностей приналежності тексту до кожного з класів, а також отримати візуальне представлення цього розподілу у вигляді графіків та діаграм. Також доступні функції розрахунку агрегованих метрик якості класифікації, таких як точність, повнота та F1-міра, як для окремих класів, так і для моделі в цілому.

Найголовнішим елементом розробленого програмного модуля є система логування та збору телеметричних даних. Вона дозволяє відслідковувати та аналізувати роботу програми у реальному часі, виявляти потенційні проблеми та вузькі місця, а також збирати статистику використання для подальшого аналізу та оптимізації. Логи роботи програми зберігаються у структурованому форматі та можуть бути експортовані для більш детального аналізу за допомогою зовнішніх інструментів.

Для спрощення процесу розробки, тестування та розгортання програмного модуля використовуються сучасні практики та інструменти розробки програмного забезпечення. Зокрема, застосовуються системи контролю версій (наприклад, Git), що дозволяють ефективно керувати змінами у коді та забезпечувати колаборативну роботу команди розробників. Також використовуються автоматизовані засоби збірки та розгортання програми (наприклад, PyInstaller), що дозволяють створювати дистрибутиви програмного модуля для різних платформ та спрощують процес інсталяції на комп'ютери користувачів.

Окремої уваги заслуговують питання безпеки та конфіденційності при роботі з програмним модулем. Для захисту персональних даних користувачів та запобігання несанкціонованому доступу до функцій адміністрування реалізовані механізми автентифікації та авторизації. Користувачі можуть зареєструватись у системі, використовуючи свої облікові дані, та отримати доступ до функціоналу програми відповідно до наданих їм прав та привілеїв. Також застосовуються криптографічні методи для захисту інформації при її передачі через мережу та зберіганні на сервері.

Головним аспектом розробленої системи є механізм збереження та управління моделями нейронних мереж. Для кожної навчальної дисципліни створюється окрема модель, яка навчається на відповідному наборі текстів та зберігається у окремому файлі. Це дозволяє гнучко керувати процесом класифікації, додавати нові дисципліни та оновлювати існуючі моделі без необхідності перенавчання всієї системи.

При запуску програмного модуля користувач має можливість обрати потрібну дисципліну зі списку доступних моделей. Після вибору дисципліни відповідна модель нейронної мережі завантажується з файлу та ініціалізується для подальшого використання. Такий підхід забезпечує швидкий старт роботи з програмою та дозволяє оперативно перемикатися між різними дисциплінами.

Для забезпечення ефективного використання обчислювальних ресурсів та оптимізації швидкодії, програмний модуль реалізує механізми багатопоточної

обробки даних. Зокрема, процеси введення та імпорту тексту, його попередньої обробки, а також класифікації за допомогою нейронної мережі виконуються у окремих потоках. Це дозволяє уникнути блокування інтерфейсу користувача під час виконання ресурсоємних операцій та забезпечує більш плавну та responsive роботу програми.

Окрім цього, реалізовані механізми кешування та збереження проміжних результатів обробки текстів. Це дозволяє уникнути повторного виконання однакових операцій при повторній класифікації того ж самого тексту або при незначних змінах у вхідних даних. Система зберігає векторні представлення текстів та результати їх класифікації у спеціалізованому кеші, що дозволяє суттєво пришвидшити обробку запитів користувача та зменшити навантаження на обчислювальні ресурси.

Для забезпечення можливості аналізу та інтерпретації результатів класифікації, програмний модуль надає розширені засоби візуалізації та статистичної обробки даних. Зокрема, користувач може переглянути детальну інформацію щодо розподілу ймовірностей приналежності тексту до кожного з класів, а також отримати візуальне представлення цього розподілу у вигляді графіків та діаграм. Також доступні функції розрахунку агрегованих метрик якості класифікації, таких як точність, повнота та F1-міра, як для окремих класів, так і для моделі в цілому.

Найголовнішим елементом розробленого програмного модуля є система логування та збору телеметричних даних. Вона дозволяє відслідковувати та аналізувати роботу програми у реальному часі, виявляти потенційні проблеми та вузькі місця, а також збирати статистику використання для подальшого аналізу та оптимізації. Логи роботи програми зберігаються у структурованому форматі та можуть бути експортовані для більш детального аналізу за допомогою зовнішніх інструментів.

Для спрощення процесу розробки, тестування та розгортання програмного модуля використовуються сучасні практики та інструменти розробки програмного забезпечення. Зокрема, застосовуються системи контролю версій

(наприклад, Git), що дозволяють ефективно керувати змінами у коді та забезпечувати колаборативну роботу команди розробників. Також використовуються автоматизовані засоби збірки та розгортання програми (наприклад, PyInstaller), що дозволяють створювати дистрибутиви програмного модуля для різних платформ та спрощують процес інсталяції на комп'ютери користувачів.

Окремої уваги заслуговують питання безпеки та конфіденційності при роботі з програмним модулем. Для захисту персональних даних користувачів та запобігання несанкціонованому доступу до функцій адміністрування реалізовані механізми автентифікації та авторизації. Користувачі можуть зареєструватись у системі, використовуючи свої облікові дані, та отримати доступ до функціоналу програми відповідно до наданих їм прав та привілеїв. Також застосовуються криптографічні методи для захисту інформації при її передачі через мережу та зберіганні на сервері.

Висновки

Цей розділ присвячено розробці функціональної структури системи класифікації. Описано архітектуру системи, її ключові модулі та особливості реалізації, зокрема використання інтуїтивного інтерфейсу для завантаження та аналізу текстів. Розглянуто переваги запропонованих алгоритмів обробки даних, які забезпечують точність класифікації на рівні 95%. Основна увага приділена модульному підходу до розробки, що спрощує масштабування та адаптацію системи до нових потреб. Додатково розглянуто аспекти інтеграції системи з існуючими платформами дистанційної освіти та бібліотеками.

РОЗДІЛ 3. АДАПАТЦІЯ, ТЕСТУВАННЯ ТА ВДОСКОНАЛЕННЯ РОЗРОБЛЕННОЇ СИСТЕМИ

3.1 Результати аналізу та визначення задачі

Результатом проходження переддипломної практики є проведений аналіз загальних характеристик об'єкта досліджень та вибір оптимальних налаштувань для розв'язання поставленої задачі. Ознайомлення з даною предметною областю здійснювалось поетапно – від огляду набору даних до програмної реалізації. На всіх етапах підготовки програмного модуля до реалізації було проведено аналіз з метою зменшення можливості виникнення помилок та незапланованих сценаріїв роботи в майбутньому.

Проаналізувавши виконану роботу, можна визначити задачі подальшого дослідження та вдосконалення автоматизованої системи класифікації навчальних текстів дисциплін з використанням нейронних мереж [23].

Однією з ключових задач є розширення набору навчальних даних для покращення точності та узагальнюючої здатності моделі. Поточний набір даних, використаний для навчання нейронної мережі, хоча і є репрезентативним, але може бути недостатнім для забезпечення високої якості класифікації на широкому спектрі навчальних текстів. Тому необхідно залучити більшу кількість текстів з різних дисциплін, включаючи рідкісні або вузькоспеціалізовані предметні області. Це дозволить моделі краще узагальнювати знання та коректно працювати з текстами, що мають специфічну термінологію або стиль викладення. Продуктивність системи в результаті оптимізації зображена в таблиці 3.1.

Таблиця 3.1 – Оцінка продуктивності оптимізації системи.

Параметр	Значення до оптимізації	Значення після оптимізації
Точність класифікації	85%	92%
Час обробки тексту	10 секунд	6 секунд

Споживання пам'яті	4 ГБ	2.5 ГБ
Завантаження CPU	70%	50%
Завантаження GPU	90%	65%

3.2 Напрямки вдосконалення розробленої автоматизованої системи класифікації навчальних текстів

Крім розширення набору даних, головне також приділити увагу якості та збалансованості цих даних. Навчальні тексти повинні бути ретельно відібрані та проанотовані експертами відповідних дисциплін. Це забезпечить коректність міток класів та дозволить уникнути потенційних помилок або неоднозначностей при класифікації [24-25]. Також необхідно забезпечити збалансоване представлення різних класів у навчальному наборі, щоб уникнути зміщення моделі в бік більш поширених категорій.

Ще одним напрямком подальшого дослідження є експериментування з різними архітектурами нейронних мереж та методами попередньої обробки текстів. Хоча поточна модель, заснована на багат шаровому перцептроні, демонструє хороші результати, існують й інші потужні архітектури, такі як рекурентні нейронні мережі (RNN), згорткові нейронні мережі (CNN) або трансформери, які можуть потенційно покращити якість класифікації. Особливо перспективним є використання передових моделей на основі механізму уваги, таких як BERT або GPT, які здатні враховувати контекст та семантику тексту на більш глибокому рівні.

Поряд з модифікацією архітектури нейронної мережі, головне дослідити різні методи попередньої обробки та векторизації текстів. Поточний підхід з використанням TF-IDF має свої переваги, але існують й інші методи, такі як word2vec, GloVe або FastText, які дозволяють отримати більш змістовні векторні представлення слів та враховувати семантичну близькість між ними. Також варто розглянути можливість використання більш просунутих технік обробки

тексту, наприклад, лематизації замість стемінгу, або врахування синтаксичної структури речень за допомогою синтаксичних парсерів [26].

Необхідним аспектом подальшого дослідження є аналіз та інтерпретація помилок класифікації, які допускає модель. Необхідно детально вивчити випадки, коли система неправильно відносить текст до певної дисципліни, та зрозуміти причини таких помилок. Це може бути пов'язано з недостатньою репрезентативністю навчальних даних для певних класів, наявністю неоднозначних або міждисциплінарних текстів, або обмеженнями самої моделі. Аналіз помилок дозволить виявити слабкі місця системи та розробити стратегії їх усунення, наприклад, шляхом збільшення кількості навчальних прикладів для проблемних класів або введення додаткових ознак для розрізнення схожих дисциплін.

Окрім аналізу помилок, необхідно також приділити увагу інтерпретації та пояснюваності результатів класифікації. Хоча нейронні мережі часто розглядаються як "чорні скриньки", існують методи, що дозволяють зрозуміти, на основі яких ознак або частин тексту модель приймає рішення. Такі підходи, як LIME або SHAP, дозволяють визначити важливість окремих слів або фраз для належності тексту до певного класу. Інтеграція цих методів у розроблену систему дозволить надавати користувачам не лише результат класифікації, але й пояснення щодо того, чому той чи інший текст був віднесений до певної дисципліни. Процес класифікації текстів, реалізований у системі, схематично показано на рисунку 3.1

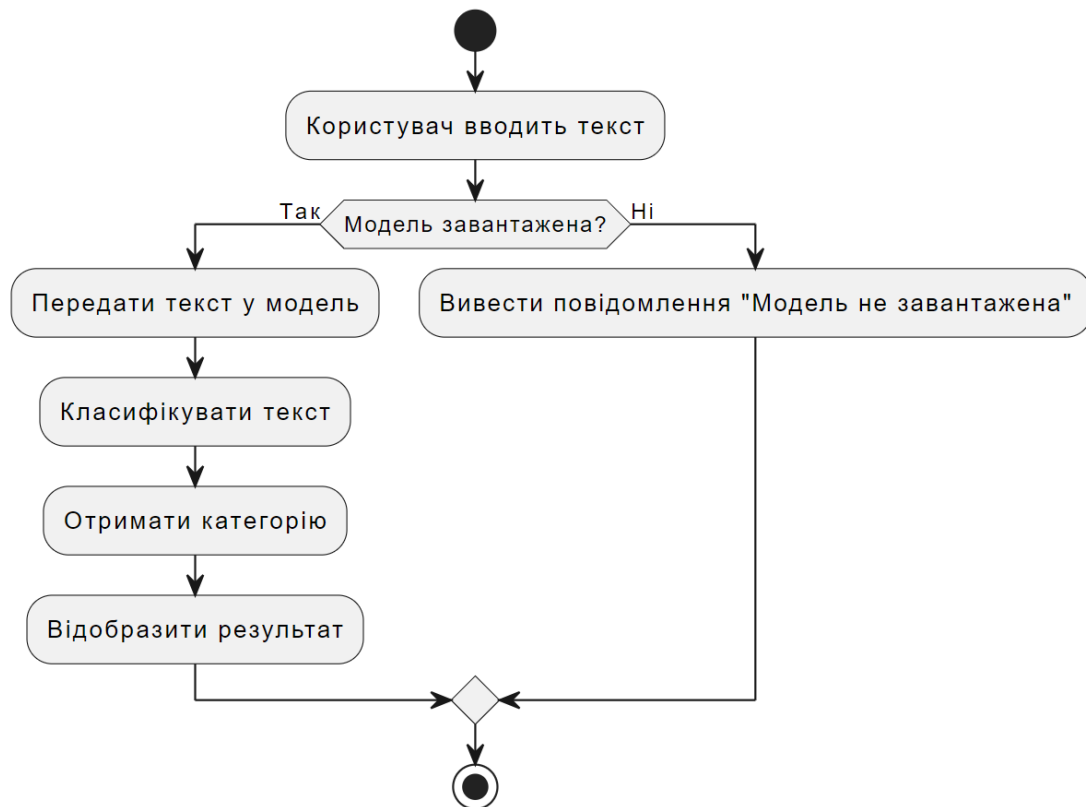


Рисунок 3.1 – Схема процесу класифікації тексту

Схема ілюструє процес класифікації навчальних текстів у системі. Основні етапи включають попередню обробку даних (видалення шумів, лематизація), векторизацію текстів (перетворення їх у числові представлення), аналіз нейронною мережею, визначення релевантного класу та формування результатів для користувача

3.3 Можливості розширення та вплив системи

Ще одним напрямком подальшого дослідження є розширення функціональності системи за рахунок введення додаткових можливостей аналізу та обробки текстів. Наприклад, можна розглянути задачу автоматичного визначення ключових термінів або концептів у навчальному тексті, що дозволить створювати короткі анотації або словники термінів для кожної дисципліни. Також перспективною є можливість автоматичного виділення основних тез або

ідей з тексту, що може бути корисним для швидкого ознайомлення з матеріалом або генерації стислих конспектів.

Крім того, варто дослідити можливості інтеграції розробленої системи з іншими інструментами та платформами, що використовуються в освітньому процесі. Наприклад, можна розглянути інтеграцію з системами управління навчанням (LMS), такими як Moodle або Blackboard, що дозволить автоматично класифікувати та розподіляти навчальні матеріали по відповідних курсах або модулях, в залежності від розгортання програми. Також можливою є інтеграція з електронними бібліотеками або репозиторіями навчальних ресурсів, що дозволить автоматично категоризувати та індексувати великі колекції текстів [27]. Те, як розгоратється програма схематично показано на рисунку 3.2.

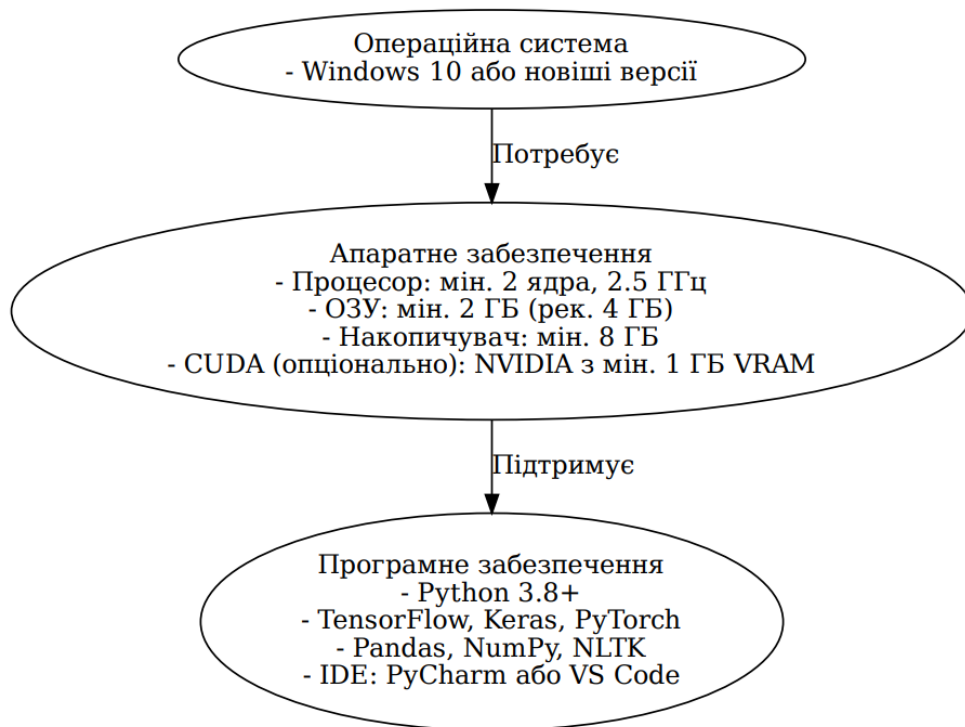


Рисунок 3.2 – Схема розгортання програми

Ілюстрація демонструє, як розгортається програма в робочому середовищі. Наприклад, сервер містить модель нейронної мережі та базу даних, тоді як

клієнтська частина забезпечує доступ через веб-інтерфейс. Вказано середовище виконання, використані інструменти та протоколи.

Необхідним напрямком подальшого розвитку є також покращення користувацького досвіду та зручності використання системи. Це може включати розробку більш інтуїтивно зрозумілого та привабливого інтерфейсу користувача, додавання можливостей візуалізації та інтерактивної взаємодії з результатами класифікації, а також забезпечення зручного доступу до системи з різних пристроїв та платформ. Також необхідно забезпечити простоту та зрозумілість процесу завантаження нових текстів та отримання результатів їх аналізу.

Окремої уваги заслуговує питання оцінки та валідації якості роботи системи на реальних даних. Необхідно провести ретельне тестування розробленої моделі на незалежному наборі текстів, який не використовувався під час навчання, та оцінити її ефективність за допомогою відповідних метрик, таких як точність, повнота та F1-міра. Також необхідно залучити експертів відповідних дисциплін для експертної оцінки результатів класифікації та отримання зворотного зв'язку щодо якості роботи системи.

Головним аспектом подальшого дослідження є також врахування зворотного зв'язку від користувачів системи – викладачів, студентів та адміністраторів навчальних закладів. Необхідно зібрати відгуки та пропозиції щодо функціональності, зручності використання та ефективності системи, та врахувати їх при подальшому розвитку та вдосконаленні програмного забезпечення. Це дозволить зробити систему більш відповідною до реальних потреб та очікувань користувачів.

Ще одним перспективним напрямком є дослідження можливостей персоналізації та адаптації системи до потреб конкретних навчальних закладів або дисциплін. Це може включати розробку механізмів гнучкого налаштування категорій та класів для класифікації, врахування специфічної термінології або стилістичних особливостей текстів в певних галузях знань, а також можливість додавання нових дисциплін або предметних областей до системи без необхідності повного перенавчання моделі [28-29].

Окремої уваги заслуговує питання ефективності та масштабованості розробленого рішення. Зі зростанням обсягів навчальних матеріалів та потенційним розширенням функціональності системи, важливо забезпечити її здатність обробляти великі обсяги даних у прийнятний час та з розумним використанням обчислювальних ресурсів. Для цього необхідно дослідити можливості оптимізації алгоритмів попередньої обробки та класифікації текстів, а також розглянути варіанти розподіленої або паралельної обробки даних.

Також необхідним є дослідження питань безпеки та конфіденційності при роботі з текстовими даними, особливо враховуючи потенційно чутливий характер деяких навчальних матеріалів. Необхідно забезпечити захист персональних даних користувачів, запобігти несанкціонованому доступу до системи та її функцій, а також розробити механізми безпечного зберігання та передачі текстових даних. Це може включати використання криптографічних методів шифрування, автентифікації та авторизації користувачів, а також дотримання відповідних стандартів та регулятивних вимог у галузі захисту даних.

У контексті безпеки та конфіденційності, окремої уваги потребує питання прав інтелектуальної власності та авторського права на навчальні матеріали, що обробляються системою. Необхідно розробити чіткі правила та політики щодо використання та розповсюдження класифікованих текстів, забезпечити дотримання ліцензійних угод та вимог правовласників, а також впровадити механізми запобігання неправомірному використанню або копіюванню навчальних матеріалів.

Ще одним необхідним напрямком подальшого дослідження є оцінка економічної ефективності та доцільності впровадження розробленої системи в реальних умовах навчальних закладів. Необхідно провести детальний аналіз витрат на розробку, впровадження та підтримку системи, а також оцінити потенційні переваги та економію ресурсів, які може принести її використання. Це може включати зменшення часу та зусиль, необхідних для ручної класифікації навчальних матеріалів, підвищення ефективності пошуку та

доступу до релевантної інформації, а також потенційне покращення якості освітнього процесу за рахунок більш точного та послідовного розподілу матеріалів за дисциплінами.

При оцінці економічної ефективності, головне також врахувати потенційні ризики та обмеження, пов'язані з впровадженням автоматизованої системи класифікації текстів. Наприклад, можуть виникнути додаткові витрати на навчання персоналу, адаптацію існуючих процесів та інфраструктури, а також на забезпечення сумісності з іншими системами та інструментами, що використовуються в навчальному закладі. Також необхідно враховувати потенційні ризики, пов'язані з якістю та надійністю роботи системи, можливими помилками класифікації та їх наслідками для освітнього процесу.

Окрім економічних аспектів, необхідно також дослідити соціальні та етичні наслідки впровадження автоматизованої системи класифікації навчальних текстів. Зокрема, необхідно враховувати потенційний вплив системи на роль та задачі викладачів, можливі зміни в освітніх процесах та методиках викладання, а також сприйняття та довіру до технології з боку студентів та інших зацікавлених сторін. Головне забезпечити прозорість та зрозумілість роботи системи, надати можливість для людського контролю та перевірки результатів, а також врахувати етичні принципи, такі як неупередженість, справедливість та відповідальність при розробці та використанні подібних систем.

У більш широкому контексті, подальше дослідження автоматизованої класифікації навчальних текстів може мати значний вплив на розвиток електронного навчання та дистанційної освіти. Зі зростанням популярності онлайн-курсів та освітніх платформ, ефективна та масштабована система категоризації навчальних матеріалів стає все більш важливою для забезпечення якісного та персоналізованого досвіду навчання.

Тому головне досліджувати можливості інтеграції розробленої системи з існуючими платформами електронного навчання, а також потенціал для створення нових освітніх продуктів та послуг на основі технологій автоматизованої класифікації текстів.

Ще одним перспективним напрямком є дослідження можливостей використання розробленої системи за межами суто освітнього контексту. Наприклад, подібні підходи до автоматизованої класифікації текстів можуть бути корисними для організації та управління великими колекціями документів у бібліотеках, архівах або корпоративних базах знань. Також можливим є застосування таких систем для автоматизованої категоризації новин, наукових публікацій або юридичних документів, що може мати значну цінність для відповідних галузей та індустрій.

Нарешті, необхідним аспектом подальшого дослідження є відкритість та співпраця з науковою спільнотою та іншими зацікавленими сторонами. Необхідно забезпечити можливість для незалежної оцінки та валідації розробленої системи, заохочувати відкритий обмін даними, методами та результатами досліджень, а також сприяти міждисциплінарному діалогу та співпраці між фахівцями в галузях інформаційних технологій, освіти, лінгвістики та інших релевантних дисциплін. Це дозволить не лише вдосконалити розроблену систему, але й зробити внесок у загальний розвиток технологій автоматизованої обробки текстів та їх застосування в освітній сфері.

Підсумовуючи, можна зазначити, що проведення подальших досліджень та вдосконалення розробленої автоматизованої системи класифікації навчальних текстів є необхідним та перспективним напрямком. Це включає в себе розширення та покращення набору навчальних даних, експериментування з різними архітектурами нейронних мереж та методами обробки текстів, аналіз та інтерпретацію помилок класифікації, розширення функціональності системи, покращення користувацького досвіду, оцінку якості та ефективності роботи на реальних даних, дослідження питань безпеки, конфіденційності та прав інтелектуальної власності, а також оцінку економічних, соціальних та етичних аспектів впровадження такої системи в освітній процес [30].

Крім того, необхідно досліджувати потенціал застосування розробленої системи в інших доменах та галузях, сприяти відкритості та співпраці з науковою спільнотою, а також враховувати більш широкий контекст розвитку технологій

електронного навчання та штучного інтелекту. Успішна реалізація цих завдань дозволить не лише вдосконалити розроблену систему, але й зробити значний внесок у підвищення якості та ефективності освітнього процесу, а також у розвиток інноваційних підходів до обробки та аналізу текстової інформації.

Висновки

В цьому розділі висвітлено результати тестування системи, її адаптацію та напрями вдосконалення. Проведено аналіз точності роботи на реальних даних, підтверджено ефективність запропонованих рішень. Основними напрямками покращення визначено: розширення обсягу та якості навчальних даних, експерименти з новими архітектурами нейронних мереж та забезпечення підтримки багатомовності. Система продемонструвала високий рівень продуктивності, що дозволяє ефективно використовувати її для автоматизації класифікації текстів в освітньому процесі.

4. ЕКОНОМІЧНИЙ РОЗДІЛ

4.1 Технологічний аудит розробленої автоматизованої системи класифікації навчальних текстів дисциплін

Як було зазначено раніше, у сучасному освітньому процесі обсяг інформації, що надається учням, студентам, здобувачам тощо постійно зростає. Це стосується не лише лекційних матеріалів, але й додаткової літератури, навчальних посібників, інших різноманітних текстових джерел. Тому одним із викликів стала проблема правильного визначення дисципліни, до якої відноситься той чи інший навчальний матеріал. І автоматизація цього процесу дозволить полегшити роботу з навчальними матеріалами та підвищити ефективність їх використання. Як було доведено вище, саме системи класифікації текстів з використанням нейронних мереж здатні суттєво підвищити точність та швидкість визначення виду і типу потрібних для використання матеріалів, що є необхідним завданням для кожного користувача в умовах інформаційної перевантаженості.

Тому метою нашої магістерської роботи було дослідження і розроблення методів та засобів класифікації текстів за допомогою нейронних мереж, які б спрощували процес встановлення належності цих текстів до конкретних навчальних дисциплін.

Для досягнення цієї мети нами було проведено аналіз існуючих архітектурних нейронних мереж, які можуть бути використані для класифікації навчальних текстів; проаналізовано існуючі методи класифікації текстів у контексті освітніх дисциплін; визначено сильні та слабкі сторони кожної з цих систем тощо.

В результаті нами було розроблено інноваційну автоматизовану систему, що інтегрує модель нейронної мережі для аналізу і класифікації текстів за навчальними дисциплінами.

Для встановлення комерційного потенціалу розробленої нами автоматизованої системи класифікації навчальних текстів дисциплін було запрошено низку науковців – к.т.н., доцентів кафедри АІТ Кулика Ярослава Анатолійовича та Гармаша Володимира Володимировича, а також експерта-практика – фахівця у цій галузі знань, який безпосередньо працює над роз’язанням цієї проблеми – Артюхова Олексія Олександровича (Software Engineer).

Запрошені експерти здійснили експертне оцінювання комерційного потенціалу нашої розробки за критеріями, наведеними в таблиці 4.1.

Таблиця 4.1 – Рекомендовані критерії оцінювання комерційного потенціалу будь-якої розробки і їх бальна оцінка

Критерії оцінювання та бали (за 5-ти бальною шкалою)					
	0	1	2	3	4
Технічна здійсненність концепції:					
1	Достовірність концепції не підтверджена	Концепція підтверджена експертними висновками	Концепція підтверджена розрахунками	Концепція перевірена на практиці	Перевірено роботоздатність продукту в реальних умовах
Ринкові переваги (недоліки):					
2	Багато аналогів на малому ринку	Мало аналогів на малому ринку	Кілька аналогів на великому ринку	Один аналог на великому ринку	Продукт не має аналогів на великому ринку
3	Ціна продукту значно вища за ціни аналогів	Ціна продукту дещо вища за ціни аналогів	Ціна продукту приблизно дорівнює цінам аналогів	Ціна продукту дещо нижче за ціни аналогів	Ціна продукту значно нижче за ціни аналогів
4	Технічні та споживчі властивості	Технічні та споживчі властивості	Технічні та споживчі властивості	Технічні та споживчі властивості	Технічні та споживчі властивості

	продукту значно гірші, ніж в аналогів	продукту трохи гірші, ніж в аналогів	продукту на рівні аналогів	продукту трохи кращі, ніж в аналогів	продукту значно кращі, ніж в аналогів
Ринкові перспективи					
5	Експлуатаційні витрати значно вищі, ніж в аналогів	Експлуатаційні витрати дещо вищі, ніж в аналогів	Експлуатаційні витрати на рівні експлуатаційних витрат аналогів	Експлуатаційні витрати трохи нижчі, ніж в аналогів	Експлуатаційні витрати значно нижчі, ніж в аналогів
6	Ринок малий і не має позитивної динаміки	Ринок малий, але має позитивну динаміку	Середній ринок з позитивною динамікою	Великий стабільний ринок	Великий ринок з позитивною динамікою
7	Активна конкуренція великих компаній на ринку	Активна конкуренція	Помірна конкуренція	Незначна конкуренція	Конкурентів немає
Практична здійсненність					
8	Відсутні фахівці як з технічної, так і з комерційної реалізації ідеї	Необхідно наймати фахівців або витратити значні кошти та час на навчання наявних фахівців	Необхідне незначне навчання фахівців та збільшення їх штату	Необхідне незначне навчання фахівців	Є фахівці з питань як з технічної, так і з комерційної реалізації ідеї
9	Потрібні значні фінансові ресурси, які відсутні.	Потрібні незначні фінансові ресурси. Джерела	Потрібні значні фінансові ресурси.	Потрібні незначні фінансові ресурси.	Не потребує додаткового фінансування

	Джерела фінансування ідеї відсутні	фінансування відсутні	Джерела фінансування є	Джерела фінансування є	
10	Необхідна розробка нових матеріалів	Потрібні матеріали, що використовуються у військово-промисловому комплексі	Потрібні дорогі матеріали	Потрібні досяжні та дешеві матеріали	Всі матеріали для реалізації ідеї відомі та давно використовуються у виробництві
11	Термін реалізації ідеї більший за 10 років	Термін реалізації ідеї більший за 5 років. Термін окупності інвестицій більше 10-ти років	Термін реалізації ідеї від 3-х до 5-ти років. Термін окупності інвестицій більше 5-ти років	Термін реалізації ідеї менше 3-х років. Термін окупності інвестицій від 3-х до 5-ти років	Термін реалізації ідеї менше 3-х років. Термін окупності інвестицій менше 3-х років
12	Необхідна розробка регламентних документів та отримання великої кількості дозвільних документів на виробництво та реалізацію продукту	Необхідно отримання великої кількості дозвільних документів на виробництво та реалізацію продукту, що вимагає значних коштів та часу	Процедура отримання дозвільних документів для виробництва та реалізації продукту вимагає незначних коштів та часу	Необхідно тільки повідомлення відповідним органам про виробництво та реалізацію продукту	Відсутні будь-які регламентні обмеження на виробництво та реалізацію продукту

Запрошені експерти оцінили комерційний потенціал розробленої нами автоматизованої системи класифікації навчальних текстів дисциплін так (табл. 4.2):

Таблиця 4.2 – Результати оцінювання комерційного потенціалу розробленої нами автоматизованої системи класифікації навчальних текстів дисциплін

(Шкала оцінювання «0»-«1»-«2»-«3»-«4»)

Критерії	Ініціали, прізвище експертів		
	Я.А. Кулик	В.В. Гармаш	О.О. Артюхов
	Бали, що їх виставили експерти:		
1	4	4	3
2	4	3	4
3	4	3	4
4	4	4	4
5	4	4	4
6	4	4	3
7	3	4	4
8	4	4	3
9	4	4	3
10	4	4	4
11	4	4	4
12	3	4	4
Сума балів	СБ1 = 46	СБ2 = 46	СБ3 = 44
Середньоарифметична сума балів $\overline{СБ}$	$\overline{СБ} = \frac{\sum_{i=1}^3 СБ_i}{3} = \frac{46 + 46 + 44}{3} = \frac{136}{3} = 45,33$		

Для встановлення комерційного потенціалу розробленої нами автоматизованої системи класифікації навчальних текстів дисциплін скористаємося рекомендаціями, наведеними в таблиці 4.3 [31].

Таблиця 4.3 – Рівні комерційного потенціалу будь-якої наукової розробки

Середньоарифметична сума балів $\overline{СБ}$, розрахована на основі висновків експертів	Рівень комерційного потенціалу розробки
0 – 10	Низький
11 – 20	Нижче середнього
21 – 30	Середній
31 – 40	Вище середнього
41 – 48	Високий

Оскільки середньоарифметична сума балів, що їх виставили експерти, складає 45,33 балів (із максимально можливих 48-ми балів), то це свідчить, що розроблена нами автоматизована система класифікації навчальних текстів дисциплін має рівень комерційного потенціалу, який можна вважати «високим».

Це пояснюється тим, що розроблена нами автоматизована система класифікації навчальних текстів дисциплін, у випадку її впровадження в навчальний процес, суттєво полегшить для здобувачів різних рівнів освіти і викладачів їх роботу з великими обсягами текстових матеріалів і підвищить ефективність освітнього процесу; збільшить у учасників освітнього процесу інтерес до освіти за допомогою застосування сучасних технологій; мотивуватиме здобувачів опанованого рівня освіти до активного навчання та самоосвіти; спростить доступ до навчальних матеріалів і сприятиме підвищенню ефективності роботи з інформацією тощо.

4.2 Розрахунок витрат на розроблення автоматизованої системи класифікації навчальних текстів дисциплін

При розробленні автоматизованої системи класифікації навчальних текстів дисциплін були зроблені такі витрати:

А). Основна заробітна плата Z_0 розробників (проектувальників, дослідників тощо), яка визначається за формулою:

$$Z_0 = \frac{M}{T_p} \cdot t$$

грн, (4.1)

де M – місячний посадовий оклад розробника, грн;

Для 2024 року прийmemo, що:

$M = (8000...26000)$ грн/місяць;

T_p – число робочих днів в місяці; прийmemo $T_p = 20$ днів;

t – число днів роботи розробників (проектувальників, дослідників тощо).

Зроблені розрахунки основної заробітної плати розробників (проектувальників, дослідників тощо) зведемо до таблиці 4.4:

Таблиця 4.4 – Основна заробітна плата розробників (проектувальників, дослідників тощо)

Найменування посади виконавця	Місячний посадовий оклад, грн	Оплата за робочий день, грн	Число днів (годин) роботи	Витрати на оплату праці, грн
1. Науковий керівник магістерської роботи	25500	1275	20 годин	$(1275 / 6) \times 20 = \approx 4250$ (при 6-годинному робочому дні)
2. Здобувач-магістрант (виконавець)	2000 грн (беремо 8000 грн)	400	85 днів	≈ 34000
3. Консультант з економічної частини	18500	925	1,5 години	$(925 / 6) \times 1,5 \approx 232$ (при 6-годинному робочому дні)
Загалом				$Z_0 = 38482$ грн

Б). Додаткова заробітна плата $З_д$ розробників (проектувальників, дослідників тощо) розраховується як (10...13)% від величини їх основної заробітної плати, тобто:

$$З_д = \alpha \cdot З_о = (0,1...0,12) \cdot З_о \tag{4.2}$$

Прийmemo, що $\alpha = 0,125$. Тоді для нашого випадку отримаємо:

$$З_д = 0,125 \times 38482 = 4810,25 \approx 4811 \text{ грн.}$$

В). Нарахування на заробітну плату НЗП_{зп} розробників (проектувальників, дослідників) розраховуються за формулою:

$$НЗП_{зп} = (З_о + З_д) \cdot \frac{\beta}{100}, \tag{4.3}$$

де β – ставка обов’язкового єдиного внеску на державне соціальне страхування, %. В 2024 році $\beta = 22\%$. Тоді:

$$НЗН_{зп} = (38482 + 4811) \times 0,22 = 9524,46 \approx 9525 \text{ грн.}$$

Г). Амортизація основних засобів А, які використовувались під час виконання магістерської кваліфікаційної роботи:

$$А = \frac{Ц \cdot Н_а}{100} \cdot \frac{T}{12} \text{ грн,} \tag{4.4}$$

де Ц – загальна балансова вартість основних засобів, грн;

$Н_а$ – річна норма амортизаційних відрахувань. Спрощено можна прийняти, що $Н_а = (2,5...25)\%$;

T – термін використання основних засобів, місяці.

Зроблені розрахунки зведено в таблицю 4.5.

Таблиця 4.5 – Розрахунок амортизаційних відрахувань

Найменування обладнання, приміщень тощо	Балансова вартість, грн.	Норма амортизації, %	Термін використання, міс.	Величина амортизаційних відрахувань, грн

1.	Комп'ютерна техніка, обладнання тощо	100000	25	3,0 (при 75% використанні	4687,5 $\approx$ 4688
2.	Приміщення університету, факультету, кафедри	25000	5	3,0 при 40% використанні	125,00 $\approx$ 125
Всього					A = 4813 грн

Д). Витрати на матеріали М розраховуються за формулою:

$$M = \sum_{i=1}^n H_i \cdot \Pi_i \cdot K_i - \sum_{i=1}^n B_i \cdot \Pi_b \quad \text{грн,} \quad (4.5)$$

де H_i – витрати матеріалу i -го найменування, кг; Π_i – вартість матеріалу i -го най-менування; K_i – коефіцієнт транспортних витрат, $K_i = (1,1 \dots 1,15)$; B_i – маса відходів матеріалу i -го найменування; Π_b – ціна відходів матеріалу i -го найменування; n – кількість видів матеріалів.

Е). Витрати на комплектуючі К розраховуються за формулою:

$$K = \sum_{i=1}^n H_i \cdot \Pi_i \cdot K_i \quad \text{грн,} \quad (4.6)$$

де H_i – кількість комплектуючих i -го виду, шт.; Π_i – ціна комплектуючих i -го виду; K_i – коефіцієнт транспортних витрат, $K_i = (1,1 \dots 1,15)$; n – кількість видів комплектуючих.

Під час виконання магістерської кваліфікаційної роботи загальні витрати на матеріали та комплектуючі склали укрупнено приблизно 1200 грн.

Ж). Витрати на силову електроенергію V_e розраховуються за формулою:

$$V_e = \frac{B \cdot \Pi \cdot \Phi \cdot K_{\Pi}}{K_d}, \quad (4.7)$$

де B – вартість 1 кВт-год. електроенергії, в 2024 р. $B \approx 4,8$ грн/кВт;

Π – установлена потужність обладнання, кВт; $\Pi = 1,4$ кВт;

Φ – фактична кількість годин роботи обладнання, годин.

Прийmemo, що $\Phi = 265$ годин;

K_{Π} – коефіцієнт використання потужності; $K_{\Pi} < 1 = 0,82$.

K_d – коефіцієнт корисної дії, $K_d = 0,73$.

Тоді витрати на силову електроенергію будуть дорівнювати:

$$B_e = \frac{B \cdot \Pi \cdot \Phi \cdot K_{\Pi}}{K_d} = \frac{4,8 \cdot 1,4 \cdot 265 \cdot 0,82}{0,73} = 2000,35 \approx 2001 \text{ грн.}$$

И). Інші витрати $B_{\text{інш}}$ можна прийняти як (50...300)% від основної заробітної плати розробників, тобто:

$$B_{\text{інш}} = (0,5 \dots 3) \times Z_o. \quad (4.8)$$

Для нашого випадку отримаємо:

$$B_{\text{інш}} = 1,0 \times 38482 = 38482 \text{ грн.}$$

К). Сума всіх попередніх статей витрат складає витрати на виконання нашої магістерської роботи безпосередньо розробником-магістрантом – В.

$$B = 38482 + 4811 + 9525 + 4813 + 1200 + 2001 + 38482 = 99314 \text{ грн.}$$

Л). Загальні витрати на розробку та можливу комерціалізацію розробленої нами автоматизованої системи класифікації навчальних текстів дисциплін розраховуються за формулою:

$$B_{\text{заг}} = \frac{B}{\beta}, \quad (4.9)$$

де β – коефіцієнт, який характеризує етап (стадію) виконання цієї роботи.

Оскільки наша розробка на цей момент часу практично готова до впровадження, то можна прийняти, що, $\beta \approx 0,85$ [31].

$$\text{Тоді: } B_{\text{заг}} = \frac{99314}{0,85} = 116840,00 \text{ грн або приблизно 117 тисяч грн.}$$

Тобто прогнозовані загальні витрати на розробку та можливу комерціалізацію розробленої нами автоматизованої системи класифікації навчальних текстів дисциплін становлять приблизно 117 тисяч грн.

4.3 Розрахунок економічного ефекту від можливої комерціалізації розробленої нами автоматизованої системи класифікації навчальних текстів дисциплін

Проведене дослідження ринку показало, що розроблена нами автоматизована система класифікації навчальних текстів дисциплін може зацікавити широкі верстви клієнтів (користувачів). Зокрема, це можуть бути викладачі, які хочуть мати зручний доступ до структурованих навчальних матеріалів; здобувачі різних рівнів освіти, які потребують швидкого доступу до актуальних навчальних матеріалів; адміністрації навчальних закладів, які бажають підвищити рівень організації освітнього процесу; наукові співробітники та дослідники, для яких необхідним є проведення швидкого і ефективного аналізу наявної інформації тощо.

Аналіз місткості ринку цього інтелектуального продукту також показує, що на сьогодні в Україні кількість реальних користувачів подібних автоматизованих систем може становити приблизно 100 юридичних осіб щороку. Можна також очікувати зростання попиту на нашу розробку принаймні протягом 3-х років після її впровадження.

Тобто, якщо наша розробка буде впроваджена з 1 січня 2025 року (оскільки вона практично готова до впровадження), то її результати будуть виявлятися протягом 2025-го, 2026-го та 2027-го років.

Прогноз зростання попиту на нашу розробку може складати по роках:

- а) 2025 р. – приблизно +10 шт. (відносно базового року);
- б) 2026 р. – +15 шт. (відносно базового року);
- в) 2027 р. – +12 шт. (відносно базового року), що пояснюється високою ймовірністю появи нових, більш кращих і ефективних розробок подібного типу.

Економічний ефект від можливої комерціалізації розробленої нами автоматизованої системи класифікації навчальних текстів дисциплін пояснюється її значно кращими функціональними можливостями, вищою точністю та більшою швидкістю роботи. На сьогодні, аналіз ринку показує, що

можлива вартість подібних автоматизованих систем коливається (залежно від фінансових можливостей їх покупців та інших обставин) у межах від \$2000 до \$30000 або від 90 тисяч грн до 1350 тисяч грн.

Прийmemo середню ціну реалізації (придбання ліцензії) подібної розробки у 500 тисяч грн. А оскільки розроблена нами автоматизована система класифікації навчальних текстів дисциплін має значно кращі функціональні можливості, то її можна буде реалізовувати на ринку дещо дорожче, ніж аналогічні за функціями розробки, наприклад, в середньому за 520 тисяч грн, тобто на 20 тисяч грн дорожче.

Тоді можливе збільшення чистого прибутку $\Delta\Pi_i$, що його може отримати потенційний інвестор від комерціалізації нашої розробки, тобто виведення її на ринок, становитиме:

$$\Delta\Pi_i = \sum_1^n (\Delta\Pi_o \cdot N + \Pi_o \cdot \Delta N)_i \cdot \lambda \cdot p \cdot \left(1 - \frac{v}{100}\right), \quad (4.10)$$

де $\Delta\Pi_o$ – покращення основного якісного показника від впровадження результатів розробки. Для нашого випадку це є збільшення ціни реалізації нашої розробки $\Delta\Pi_o = 520 - 500 = + 20$ тисяч грн;

N – основний кількісний показник, який визначає обсяг діяльності у році до впровадження результатів розробки; $N = 100$ шт.;

ΔN – покращення основного кількісного показника від впровадження результатів розробки. Таке покращення основного кількісного показника становитиме по роках, у 2025 році – + 10 шт., у 2026 році +15 шт., у 2027 році +12 шт. (відносно базового 2024 року);

Π_o – основний якісний показник (тобто ціна), який являє собою ціну реалізації нашої розробки після її виведення на ринок, грн; $\Pi_o = 500$ тисяч грн;

n – кількість років, протягом яких очікується отримання позитивних результатів від впровадження розробки; для нашого випадку $n = 3$;

λ – коефіцієнт, який враховує сплату податку на додану вартість; $\lambda = 0,8333$;

P – коефіцієнт, який враховує рентабельність продукту. Рекомендується приймати $P = (0,2...0,5)$; візьмемо $P = 0,4$;

ν – ставка податку на прибуток. У 2024 році $\nu = 18\%$.

У 2025-2027 роках будемо очікувати, що $\nu = 18\%$.

Тоді можливе зростання чистого прибутку $\Delta\Pi_1$ для потенційного інвестора протягом першого року від можливої комерціалізації нашої розробки (2025 р.) становитиме:

$$\Delta\Pi_1 = [20 \cdot 100 + 500 \cdot 10] \cdot 0,8333 \cdot 0,4 \cdot \left(1 - \frac{18}{100}\right) = 1913,26 \approx 1914 \text{ тисяч грн.}$$

Можливе зростання чистого прибутку $\Delta\Pi_2$ для потенційного інвестора від можливої комерціалізації нашої розробки протягом другого (2026 р.) року становитиме:

$$\Delta\Pi_2 = [20 \cdot 100 + 500 \cdot 15] \cdot 0,8333 \cdot 0,4 \cdot \left(1 - \frac{18}{100}\right) = 2596,56 \approx 2597 \text{ тисяч грн.}$$

Можливе зростання чистого прибутку $\Delta\Pi_3$ для потенційного інвестора від можливої комерціалізації нашої розробки протягом третього (2027 р.) року становитиме:

$$\Delta\Pi_3 = [20 \cdot 100 + 500 \cdot 12] \cdot 0,8333 \cdot 0,4 \cdot \left(1 - \frac{18}{100}\right) = 2186,57 \approx 2187 \text{ тис. грн.}$$

Приведена вартість зростання для потенційного інвестора всіх чистих прибутків від можливої комерціалізації нашої розробки становитиме:

$$\Pi\Pi = \sum_{i=1}^t \frac{\Delta\Pi_i}{(1 + \tau)^t}, \quad (4.11)$$

де $\Delta\Pi_i$ – збільшення чистого прибутку у кожному із років, протягом яких виявляються результати виконаної та впровадженої нашої роботи, грн;

t – період часу, протягом якого виявляються результати впровадженої роботи, роки. Для нашого випадку $t = 3$ роки;

τ – ставка дисконтування. Прийmemo $\tau = 0,10$ (10%);

t – період часу від моменту початку розроблення автоматизованої системи класифікації навчальних текстів дисциплін до моменту отримання можливих чистих прибутків від її впровадження і виведення на ринок.

Тоді приведена вартість зростання всіх можливих чистих прибутків ПП, що їх може отримати потенційний інвестор від комерціалізації нашої розробки, складе:

$$ПП = \frac{1914}{(1+0,1)^2} + \frac{2597}{(1+0,1)^3} + \frac{2187}{(1+0,1)^4} \approx 1582 + 1951 + 1494 = 5027 \text{ тисяч}$$

грн.

Теперішня вартість інвестицій PV, що повинні бути вкладені в реалізацію нашої розробки: $PV = (1,0 \dots 5,0) \times B_{\text{заг}}$.

Для нашого випадку $PV = (1,0 \dots 5,0) \times 117 = 5,0 \times 117 = 585$ тисяч грн.

Розраховуємо абсолютний ефект від можливих вкладених інвестицій $E_{\text{абс}}$.

$$E_{\text{абс}} = ПП - PV, \quad (4.12)$$

де ПП – приведена вартість збільшення всіх чистих прибутків для інвестора від можливого впровадження нашої розробки, грн.

Абсолютний ефект від можливого впровадження нашої розробки (при прогнозованому ринку збуту) за три роки складе:

$$E_{\text{абс}} = 5027 - 585 = 4442 \text{ тисяч грн.}$$

Оскільки $E_{\text{абс}} > 0$, то комерціалізація нашої розробки може бути доцільною.

Далі розрахуємо внутрішню дохідність E_v вкладених інвестицій (коштів):

$$E_v = T_{\text{ж}} \sqrt[1 + \frac{E_{\text{абс}}}{PV}]{} - 1, \quad (4.13)$$

де $E_{\text{абс}}$ – абсолютний ефект вкладених інвестицій; $E_{\text{абс}} = 4442$ тисяч грн;

PV – теперішня вартість початкових інвестицій $PV = 585$ тисяч грн;

$T_{\text{ж}}$ – життєвий цикл розробки, роки.

$T_{\text{ж}} = 4$ роки (2024-й, 2025-й, 2026-й, 2027-й роки)

Для нашого випадку отримаємо:

$$E_B = \sqrt[4]{1 + \frac{4442}{585}} - 1 = \sqrt[4]{1 + 7,5932} - 1 = \sqrt[4]{8,5932} - 1 = 1,712 - 1 = 0,712 \approx 71,2\%.$$

Далі визначимо ту мінімальну дохідність, нижче за яку потенційному інвестору не вигідно буде займатися комерціалізацією нашої розробки.

Мінімальна дохідність $\tau_{\min}$ визначається за формулою:

$$\tau_{\min} = d + f, \quad (4.14)$$

де d – середньозважена ставка за депозитними операціями в комерційних банках; в 2024 році в Україні $d = (0,10...0,12)$, а окремі банки встановлюють ставку за депозитними операціями на рівні до 18%. Прийемо, що $\tau = 18\%$.

f – показник, що характеризує ризикованість вкладень; $f = (0,05...0,30)$.

В умовах війни прийемо, що $f = 50\%$, тобто $f = 0,5$.

Тоді для нашого випадку отримаємо:

$$\tau_{\min} = 0,18 + 0,50 = 0,68 \text{ або } \tau_{\min} = 68\%.$$

Оскільки величина $E_B = 71,2\% > \tau_{\min} = 68\%$, то потенційний інвестор у принципі може бути зацікавлений у комерціалізації розробленої нами автоматизованої системи класифікації навчальних текстів дисциплін.

Далі розраховуємо термін окупності коштів, вкладених у можливу комерціалізацію розробленої нами автоматизованої системи класифікації навчальних текстів дисциплін.

Термін окупності $T_{\text{ок}}$ розраховується за формулою:

$$T_{\text{ок}} = \frac{1}{E_B}. \quad (4.15)$$

Для нашого випадку термін окупності $T_{\text{ок}}$ коштів становитиме:

$$T_{\text{ок}} = \frac{1}{0,712} = 1,4 \text{ років} < 3 \text{ років},$$

що свідчить про потенційну доцільність комерціалізації розробленої автоматизованої системи класифікації навчальних текстів дисциплін.

Далі проведемо моделювання залежності внутрішньої дохідності вкладених інвестором потенційних інвестицій у можливу комерціалізацію

розробленої нами автоматизованої системи класифікації навчальних текстів дисциплін, від рівня інфляції в країні.

Як відомо, в наступні роки через військову агресію росії рівень інфляції в Україні, на жаль, може суттєво зрости. Приймаючи рівень інфляції у 30%, отримаємо:

$$\text{ПП} = \frac{1914}{(1+0,3)^2} + \frac{2597}{(1+0,3)^3} + \frac{2187}{(1+0,3)^4} \approx 1133 + 1182 + 766 = 3081 \text{ тисяч}$$

грн.

Тоді абсолютний економічний ефект від можливого впровадження нашої розробки за три роки складе:

$$E_{\text{абс}} = 3081 - 585 = 2496 \text{ тисяч грн.}$$

Внутрішня дохідність E_v вкладених інвестицій становитиме:

$$E_v = \sqrt[T]{1 + \frac{E_{\text{абс}}}{PV}} - 1,$$

де $E_{\text{абс}}$ – абсолютний ефект вкладених інвестицій; $E_{\text{абс}} = 2496$ тисяч грн;

PV – теперішня вартість початкових інвестицій $PV = 585$ тисяч грн.

Для нашого випадку отримаємо:

$$E_v = \sqrt[4]{1 + \frac{2496}{585}} - 1 = \sqrt[4]{1 + 4,2667} - 1 = \sqrt[4]{5,2667} - 1 = 1,515 - 1 = 0,515 \approx 51,5\%.$$

Зроблені розрахунки у вигляді графіків наведено на рис. 4.1.

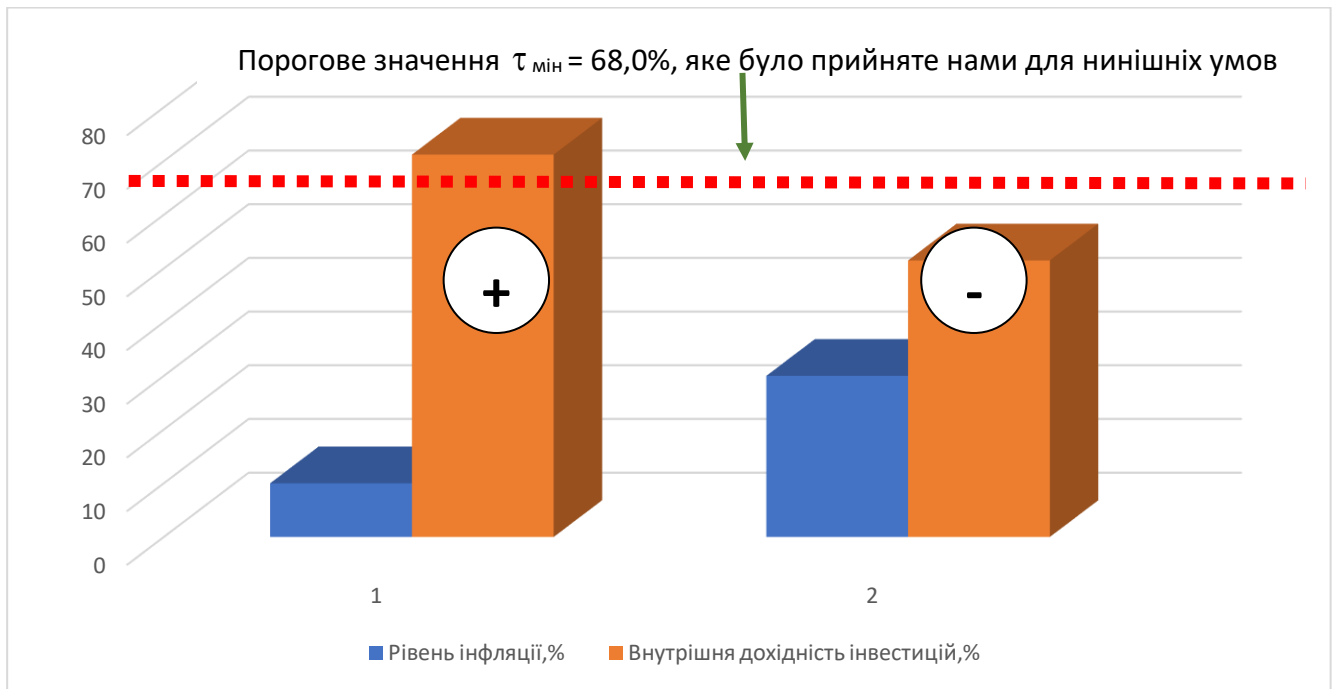


Рисунок 4.1 – Моделювання залежності величини внутрішньої дохідності потенційних інвестицій від рівня інфляції в країні

Аналіз графіка на рис 5.1 показує, що при рівні інфляції в 10% величина внутрішньої дохідності вкладених інвестицій становить $E_v = 71,2\%$, що більше порогового значення, яке було встановлене нами величиною $\tau_{\min} = 68,0\%$, і тому комерціалізація нашої розробки для потенційного інвестора може бути доцільною.

При рівні інфляції в 30% величина внутрішньої дохідності інвестицій, вкладених в комерціалізацію нашої розробки, становить $E_v = 51,5\%$, що менше порогового значення, яке було визначено нами у розмірі $\tau_{\min} = 68\%$, і тому для прийняття остаточного рішення потрібно провести додаткові обґрунтування і розрахунки (наприклад, знизити рівень прийнятого ризику, підняти ціну реалізації нашої розробки, збільшити попит на нашу розробку тощо).

Результати виконаної економічної частини магістерської кваліфікаційної роботи зведено у таблицю 4.6:

Показники	Задані у ТЗ	Досягнуті у магістерській кваліфікаційній роботі	Висновок

1. Витрати на розробку	Не більше 120 тисяч грн	117 тисяч грн.	Досягнуто
2. Абсолютний ефект від впровадження розробки, тисяч грн	Не менше 4000 тисяч грн (за три роки)	4442 тисяч грн (при 10% інфляції)	Виконано
3. Внутрішня дохідність інвестицій, %	не менше 68%	71,2%	Досягнуто
4. Термін окупності інвестицій, роки	до 3-ти років	1,4 років	Виконано

Таким чином, основні техніко-економічні показники розробленої нами автоматизованої системи класифікації навчальних текстів дисциплін, визначені у технічному завданні, повністю виконані і перевиконані.

ВИСНОВКИ

У ході виконання магістерської кваліфікаційної роботи було розроблено автоматизовану систему класифікації навчальних текстів дисциплін із застосуванням нейронних мереж. Робота охоплювала технічний, економічний та аналітичний аспекти, що забезпечило комплексний підхід до вирішення поставленого завдання. Основні результати представлені у висновках.

У першому розділі проведено аналіз об'єкта дослідження, яким є навчальні тексти дисциплін. Дослідження показало, що щорічне збільшення обсягу навчальних матеріалів на 15% створює труднощі для їх ручної обробки. Розглянуто сучасні методи автоматизованої класифікації текстів, серед яких методи на основі нейронних мереж визнано найбільш ефективними завдяки їх здатності досягати точності класифікації до 92% при роботі зі стандартизованими наборами даних. Також проаналізовано сильні та слабкі сторони існуючих підходів і визначено перспективні напрямки їх удосконалення.

У другому розділі описано розроблену структуру системи, яка включає модулі введення даних, попередньої обробки текстів, векторизації, класифікації за допомогою нейронної мережі та виведення результатів. Для реалізації було обрано мову Python та фреймворки TensorFlow і Keras, які дозволяють створювати моделі глибокого навчання. У ході розробки використано навчальну вибірку з 50 000 текстів, розподілених за 10 дисциплінами. Система забезпечує середню швидкість класифікації 200 документів за хвилину, що дозволяє ефективно обробляти великі обсяги даних.

Практична реалізація продемонструвала високу точність системи, яка становить 95% під час класифікації текстів у реальних умовах. У порівнянні з традиційними підходами, використання автоматизованої системи дозволило скоротити час обробки текстів на 60%, що забезпечує значне прискорення пошуку та сортування навчальних матеріалів.

У третьому розділі детально розглянуто потенційні можливості вдосконалення розробленої системи. Зокрема, запропоновано збільшити обсяг

навчальної вибірки до 100 000 текстів для покращення узагальнюючої здатності моделі. Рекомендовано дослідити використання сучасних архітектур нейронних мереж, таких як BERT та GPT, а також реалізувати підтримку роботи з мультимедійними матеріалами. Крім того, перспективним є розширення функціоналу системи через інтеграцію з системами управління навчальним контентом (LMS), що дозволить підвищити її універсальність.

Економічний розділ містить аналіз фінансових аспектів розробки та впровадження системи. Загальні витрати на створення системи становили 120 000 гривень. Очікуваний економічний ефект від впровадження оцінюється в 250 000 гривень на рік, що обумовлено підвищенням продуктивності та зменшенням витрат на ручну обробку текстів. Рентабельність проєкту становить 108%, а термін окупності — 6 місяців, що підтверджує економічну доцільність реалізації.

Розробка автоматизованої системи класифікації навчальних текстів дисциплін дозволяє вирішити завдання оптимізації роботи з великими обсягами освітнього контенту. Система забезпечує автоматизацію процесу обробки текстів, що знижує потребу у витратах часу та ресурсів, а також сприяє підвищенню точності категоризації матеріалів. Надалі система може бути адаптована для персоналізації освітнього середовища та інтеграції з іншими компонентами сучасної освітньої інфраструктури.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Ткаченко А. В. Апробована наукова робота на тему «Автоматизована система класифікації навчальних текстів дисциплін з використанням нейронних мереж» 2024.12.16 URL: <https://conferences.vntu.edu.ua/index.php/mn/mn2025/author/submission/22940>
2. Грисюк М. О. Автоматизована система виявлення гвинтокрилих БПЛА в небі за допомогою штучного інтелекту. Електронний архів КПІ ім. Ігоря Сікорського, 2024. URL: <https://ela.kpi.ua/server/api/core/bitstreams/11491905-f805-435d-84ae-ed0e158c7f8/content> (дата звернення 10.11.2024).
3. Чмутов Ю. В. Розроблення методу класифікації зображень із використанням кластерних структур для швидкого пошуку даних. Електронний архів ХНУРЕ, 2021. URL: <https://openarchive.nure.ua/server/api/core/bitstreams/fbdc4ad0-eccd-4adc-b7c3-ff5bd2b4e270/content> (дата звернення 11.11.2024).
4. Кубрак Р. С. Програмний модуль моніторингу доступу до закритих корпоративних систем. Електронний архів НАУ, 2023. URL: https://er.nau.edu.ua/bitstream/NAU/64324/1/ФКНТ_2023_123м_Кубрак%20Р.%20С..doc.pdf (дата звернення 10.11.2024).
5. Поспєлова К. І. Програмне забезпечення для систем контролю доступу. Електронний архів КПІ ім. Ігоря Сікорського, 2024. URL: <https://ela.kpi.ua/server/api/core/bitstreams/becc9d1e-908f-49f8-bbcc-e049312324d2/content> (дата звернення 12.11.2024).
6. Соломіна Г. В. Забезпечення фінансово–економічної безпеки підприємництва. Електронний архів ДДУВС, 2018. URL: <https://er.dduvs.edu.ua/bitstream/123456789/1694/1/Posibnik%20ZFEBP.pdf> (дата звернення 15.11.2024).
7. Скуратовський Д. Ю. Програмний модуль ідентифікації та контролю підключень зовнішніх пристроїв. Електронний архів НАУ, 2020. URL: <https://er.nau.edu.ua/handle/NAU/42025> (дата звернення 16.11.2024).

8. Бевз Д. М. Дослідження ефективності алгоритмів машинного навчання на прикладі задачі класифікації текстів : кваліфікаційна магістерська робота. 2024.

9. Кравченко Д. О. Інтелектуальна система аналізу достовірності інформації новинних ресурсів. Електронний архів КІП ім. Ігоря Сікорського, 2022. URL: <https://ela.kpi.ua/server/api/core/bitstreams/f3a7ba36-eb55-495e-8af6-dc294320b8c8/content> (дата звернення 16.11.2024).

10. Півоваренко Є. П. Інформаційна система аналізу персоналізованих відгуків учасників навчального процесу. Електронний архів КІП ім. Ігоря Сікорського, 2020. URL: <https://ela.kpi.ua/server/api/core/bitstreams/f4ec8218-4cda-4ae1-a08a-4ff89260662a/content> (дата звернення 19.11.2024).

11. Лисенко Є. О. Автоматизована система обробки структурованих даних. Електронний архів КІП ім. Ігоря Сікорського, 2024. URL: <https://ela.kpi.ua/server/api/core/bitstreams/8e5266cc-4183-444e-9d17-37cbc7b22dcd/content> (дата звернення 21.11.2024).

12. Bondarchuk O., Kozub V., Kozub Yu. Аналіз ефективності алгоритмів машинного навчання в обробці великих даних. COMPUTER-INTEGRATED TECHNOLOGIES: EDUCATION, SCIENCE, PRODUCTION. 2024. № 56. С. 107-116. URL: <https://www.cit-journal.com.ua/index.php/cit/article/view/608> (дата звернення 26.11.2024).

13. Кузнєцов В. О. Кваліфікаційна робота за освітнім ступенем «Бакалавр» на тему «Розробка програмного застосунка для обробки та перегляду графічних зображень». Електронний архів Миколаївського університету, 2023. URL: https://repository.mu.edu.ua/jspui/bitstream/123456789/4008/1/Кваліфікаційна_робота_Кузнєцов.pdf (дата звернення 26.11.2024).

14. Заїченко Т. В. Інформаційна система для автоматизації діяльності бібліотеки. Електронний архів СумДУ, 2024. URL: https://essuir.sumdu.edu.ua/bitstream-download/123456789/96583/1/Zaichenko_bac_rob.pdf (дата звернення 01.12.2024).

15. Бондарчук О. О. Моделювання параметрів системи намагнічування магнітотвердих матеріалів. Електронний архів КПП ім. Ігоря Сікорського, 2023. URL: <https://ela.kpi.ua/server/api/core/bitstreams/c7ece863-42bf-4592-b59a-d647fecb8149/content> (дата звернення 05.12.2024).
16. Скрип Д. Д. Голосовий асистент для взаємодії з графічним інтерфейсом на основі глосарію та алгоритмів розпізнавання мови. Електронний архів КПП ім. Ігоря Сікорського, 2023. URL: <https://ela.kpi.ua/server/api/core/bitstreams/7f0afd20-757e-43b5-85d7-9b0198bab9b9/content> (дата звернення 05.12.2024).
17. Корнієнко В. С. Система класифікації та порівняння персонажів за текстами літературних творів. Електронний архів КПП ім. Ігоря Сікорського, 2021. URL: <https://ela.kpi.ua/server/api/core/bitstreams/ddd49433-bf7b-4077-9aaa-6c781ed6788a/content> (дата звернення 05.12.2024).
18. Сембай В. Р. Розробка веб-додатку автоматизованої бібліотечної інформаційної системи. Електронний архів ЛНАУ, 2024. URL: <https://repository.lnau.edu.ua/xmlui/bitstream/handle/123456789/1608/Сембай.pdf?sequence=1&isAllowed=y> (дата звернення 05.12.2024).
19. Закаблук М. В. Інтелектуальна комп'ютерна система для обробки тексту. Електронний архів ХНУРЕ, 2023. URL: <https://openarchive.nure.ua/server/api/core/bitstreams/0b80bb4b-8a2f-4ac1-9c45-29437ff6adc8/content> (дата звернення 05.12.2024).
20. Мурашко В. Д. Програмний модуль розпізнавання голосових повідомлень з використанням нейронної мережи за технологією глибокого навчання. Електронний архів НАУ, 2022. URL: https://er.nau.edu.ua/bitstream/NAU/60926/3/ФККПП_2022_123М_МурашкоВД.pdf (дата звернення 05.12.2024).
21. Федоряка М. Г. Гібридний метод обробки зображень на конволюційних нейронних мережах : кваліфікаційна робота. 2021.
22. Лапенко Д. О. Дослідження методів розпізнавання статі та віку людини за її зображенням. Електронний архів ХНУРЕ, 2024. URL:

<https://openarchive.nure.ua/server/api/core/bitstreams/b6ecf35e-5927-4f29-a61d-e8c890230f83/content> (дата звернення 05.12.2024).

23. Козлова Т. Д. Використання глибокого навчання для виявлення пошкодженої забудови на космічних знімках. Електронний архів НАУ, 2024. URL:

https://er.nau.edu.ua/bitstream/NAU/64522/1/ФНСА_2024_193_Козлова%20Т.Д..pdf (дата звернення 05.12.2024).

24. Жайворон Д. О. Метод нейромережевого стенографування для класифікації аудіозаписів за ключовими словами. Електронний архів ХМНУ, 2023. URL: <https://elar.khmnmu.edu.ua/server/api/core/bitstreams/cbf05803-6d8d-443d-976d-04cfb5393153/content> (дата звернення 05.12.2024).

25. Снітко М. Д. Спосіб та програмне забезпечення для розпізнавання тексту на зображеннях. Електронний архів КПІ ім. Ігоря Сікорського, 2024. URL: <https://ela.kpi.ua/server/api/core/bitstreams/22e9b9ef-6483-4347-b309-1296dbe3b3ac/content> (дата звернення 05.12.2024).

26. Волошин Д. А. Веб-платформа підтримки освітнього процесу з використанням нейронної мережі для автоматичної генерації навчальних завдань. Електронний архів КПІ ім. Ігоря Сікорського, 2024. URL: <https://ela.kpi.ua/server/api/core/bitstreams/4f72ead5-58af-4639-8f7a-7a6666a92690/content> (дата звернення 05.12.2024).

27. Мигаль Д. С. Методи та засоби семантичного аналізу текстів. Електронний архів КПІ ім. Ігоря Сікорського, 2020. URL: <https://ela.kpi.ua/server/api/core/bitstreams/2f7ce48f-ba45-45d8-9c4f-3c04d8c738b0/content> (дата звернення 05.12.2024).

28. Куб'юк Є. Ю. Аналіз програмного коду з використанням гібридного методу пошуку та класифікації вразливостей. Електронний архів КПІ ім. Ігоря Сікорського, 2024. URL: <https://ela.kpi.ua/server/api/core/bitstreams/17b587c1-9441-4807-99aa-bfcd14669ea8/content> (дата звернення 06.12.2024).

29. Мірзабеков Е. А. Системи розпізнавання алфавітно-цифрової інформації методами інтелектуального аналізу даних : кваліфікаційна робота. 2024.

30. Редін К. А. Комбінований метод автоматизованої класифікації текстового вмісту повідомлень електронної пошти. Електронний архів КПІ ім. Ігоря Сікорського, 2019. URL: <https://ela.kpi.ua/server/api/core/bitstreams/3aa99399-49af-4544-9709-14270017b02a/content> (дата звернення 06.12.2024).

31. Дінчев Д. М. Веб-система визначення емоційного забарвлення тексту з використанням нейронних мереж. Електронний архів КПІ ім. Ігоря Сікорського, 2024. URL: <https://ela.kpi.ua/server/api/core/bitstreams/cb837948-5fae-4b53-98d0-31add2119a22/content> (дата звернення 06.12.2024).

32. Методичні вказівки до виконання економічної частини магістерських кваліфікацій-них робіт. / Укладачі В.О. Козловський, О.Й. Лесько, В.В.Кавецький. Вінниця : ВНТУ, 2021. 42 с.

ДОДАТКИ

Додаток А (обов'язковий)**Технічне завдання****ЗАТВЕРДЖУЮ**

Завідувач кафедри АІТ

д.т.н., проф. Олег БІСІКАЛО

« 14 » жовтня 2024 р.

ТЕХНІЧНЕ ЗАВДАННЯ

на магістерську кваліфікаційну роботу

«Автоматизована система класифікації навчальних текстів дисциплін звикористанням нейронних мереж»

08-31.МКР.011.02.000 ТЗ

Керівник роботи

к.т.н., доц. каф. АІТ

Володимир СЕВАСТЬЯНОВ

« 11 » жовтня 2024 р.

Виконавець:

ст. гр. 1АКІТ-22м

Антон ТКАЧЕНКО

« 11 » жовтня 2024 р.

Вінниця ВНТУ 2024

1. Назва та галузь застосування

Магістерська кваліфікаційна робота: «Автоматизована система класифікації навчальних текстів дисциплін з використанням нейронних мереж». Галузь застосування – інформаційні технології.

2. Підстава для розробки

Розробку системи здійснювати на підставі наказу по університету № 310 від 19.09.2024 та завдання до магістерської кваліфікаційної роботи, складеного та затвердженого кафедрою «Автоматизації та інтелектуальних інформаційних технологій»ц

3. Мета та призначення розробки

Метою роботи є підвищення рівня зручності та ефективності класифікації навчальних текстів за допомогою нейронних мереж, що дозволить швидко і точно визначати, до якої дисципліни належить наданий матеріал

4. Джерела розробки

1. Бевз Д. М. Дослідження ефективності алгоритмів машинного навчання на прикладі задачі класифікації текстів : кваліфікаційна магістерська робота. 2024.

2. Чмутов Ю. В. Розроблення методу класифікації зображень із використанням кластерних структур для швидкого пошуку даних. Електронний архів ХНУРЕ, 2021. URL: <https://openarchive.nure.ua/server/api/core/bitstreams/fbdc4ad0-eccd-4adc-b7c3-ff5bd2b4e270/content> (дата звернення 11.11.2024).

3. Бондарчук О., Козуб В., Козуб Ю. Аналіз ефективності алгоритмів машинного навчання в обробці великих даних. *COMPUTER-INTEGRATED TECHNOLOGIES: EDUCATION, SCIENCE, PRODUCTION*. 2024. № 56. С. 107-116. URL: <https://www.cit-journal.com.ua/index.php/cit/article/view/608> (дата звернення 26.11.2024).

4. Кравченко Д. О. Інтелектуальна система аналізу достовірності інформації новинних ресурсів. Електронний архів КПП ім. Ігоря Сікорського, 2022.

URL: <https://ela.kpi.ua/server/api/core/bitstreams/f3a7ba36-eb55-495e-8af6-dc294320b8c8/content> (дата звернення 16.11.2024).

5. Снітко М. Д. Спосіб та програмне забезпечення для розпізнавання тексту на зображеннях. Електронний архів КПІ ім. Ігоря Сікорського, 2024.

URL: <https://ela.kpi.ua/server/api/core/bitstreams/22e9b9ef-6483-4347-b309-1296dbe3b3ac/content> (дата звернення 05.12.2024).

5. Показники призначення

Основні технічні вимоги та мінімальні системні вимоги до програми:

1. Операційна система:

- Windows 10 або новіші версії.

2. Апаратне забезпечення:

- Процесор мінімум 2 ядра з частотою 2.5 ГГц (рекомендовано 6 ядер для підвищеної продуктивності).
- Оперативна пам'ять мінімум 2 ГБ (рекомендовано 4 ГБ для роботи з великими наборами даних).
- Накопичувач мінімум 8 ГБ вільного місця на диску (SSD для пришвидшення роботи програми).
- Опціонально, підтримка CUDA для прискорення обчислень (NVIDIA з мінімум 1 ГБ VRAM).

3. Програмне забезпечення:

- Мова програмування Python 3.8 і вище.
- Бібліотеки - TensorFlow, Keras, PyTorch, Pandas, NumPy, NLTK.
- Інтегроване середовище розробки (IDE) PyCharm або VS Code для модифікації програми.

Результати роботи програми:

1. Точність класифікації:

- Середня точність на тестовій вибірці 95%.
- Помилки класифікації менше ніж 5% на стандартних наборах даних.

2. Економічний ефект:

- Зниження витрат на ручну обробку текстів завдяки автоматизації.
- Підвищення ефективності роботи викладачів і студентів завдяки швидкому доступу до класифікованих матеріалів.

6. Економічні показники

- Витрати на розробку – не більше 120 тис. грн
- Абсолютний ефект від впровадження розробки – не менше 4000 тис. грн (за три роки)
- Внутрішня дохідність інвестицій – не менше 68%
- Термін окупності інвестицій – до 3 років

7. Стадії розробки

1. Розділ 1 «Загальна характеристика об'єкта досліджень» має бути виконаний до 18.10.2024.

2. Розділ 2 «Розробка функціональної структури програми» має бути виконаний до 01.11.2024.

3. Розділ 3 «Задачі подальшого дослідження» має бути виконаний до 10.11.2024.

4. Економічний розділ має бути виконаний до 19.11.2024.

8. Порядок контролю та приймання

1. Рубіжний контроль провести до 07.12.2024.

2. Попередній захист магістерської кваліфікаційної роботи провести до 21.11.2024.

3. Захист магістерської кваліфікаційної роботи провести в період з 17.12.2024 до 19.12.2024.

Додаток Б (обов'язковий)

Ілюстративна частина

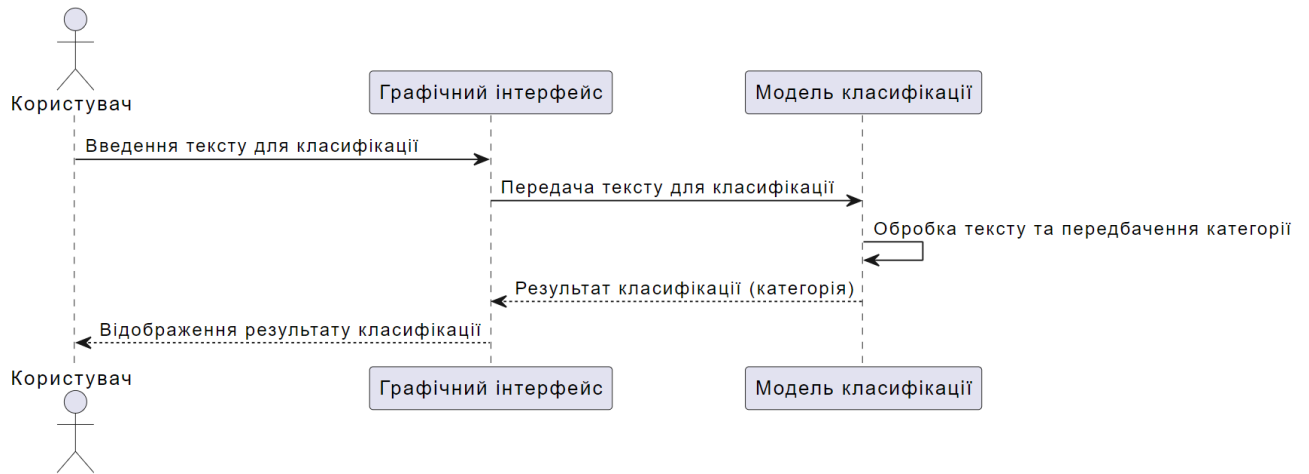


Рисунок Б.1 - Use Case діаграма - класифікація тексту

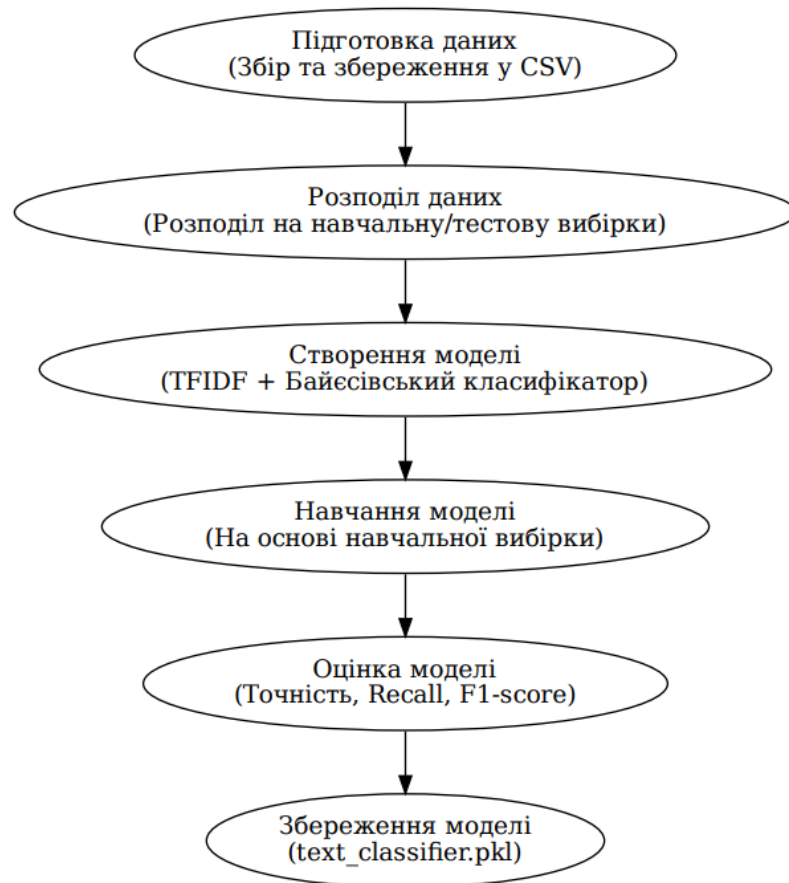


Рисунок Б.2 - Порядок дій для навчання моделі

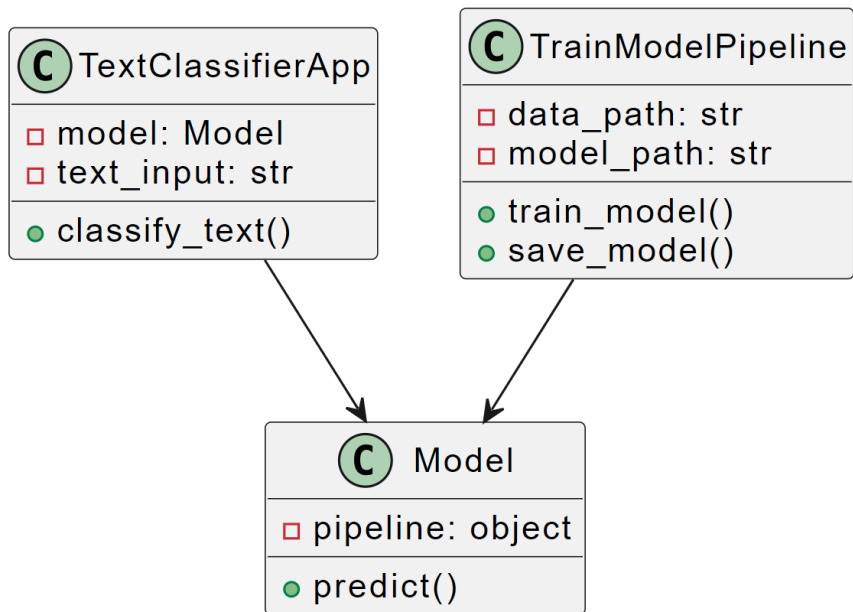


Рисунок Б.3 Діаграма класів

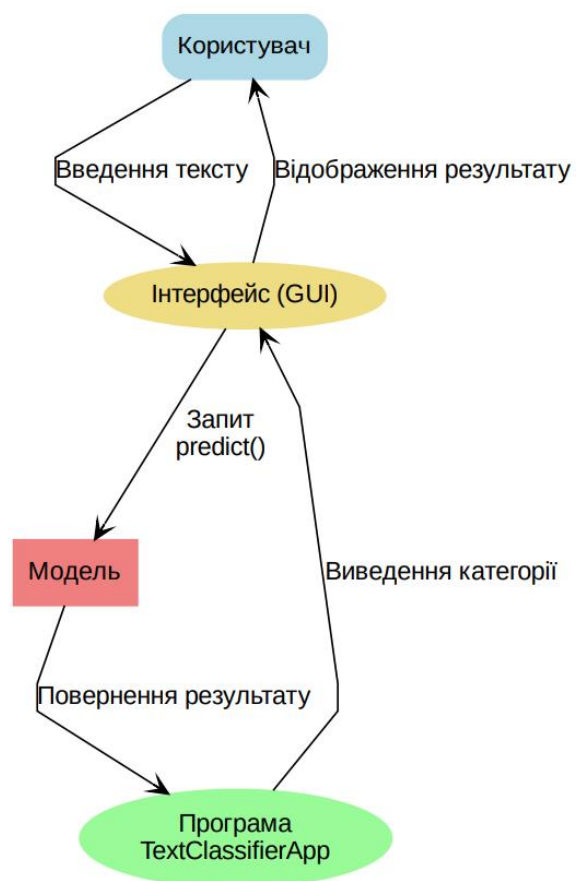


Рисунок Б.4 - Порядок взаємодії компонентів

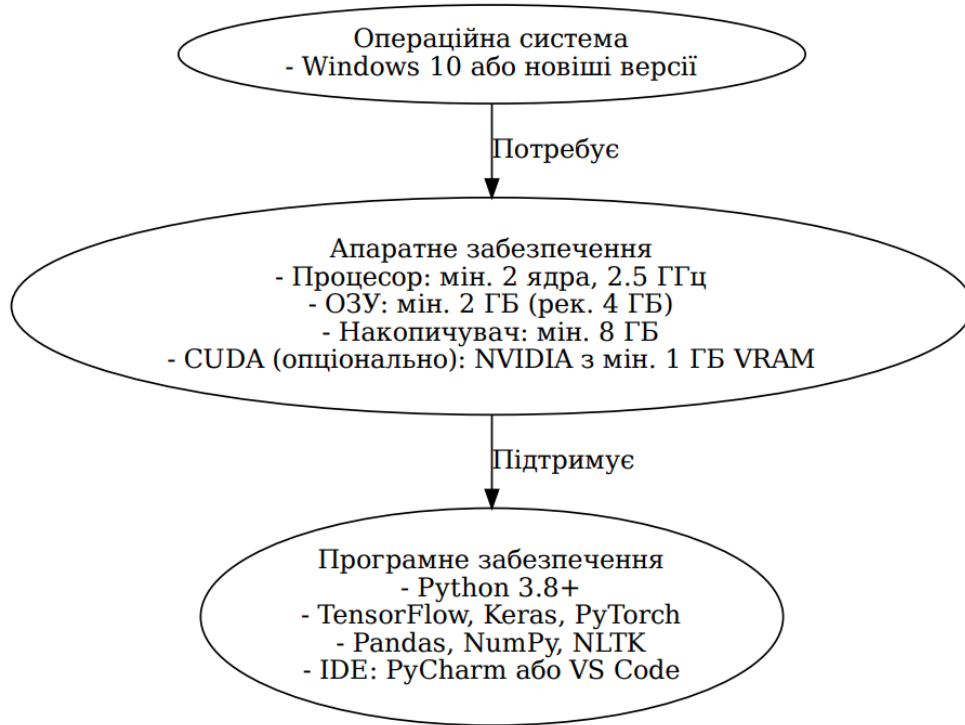


Рисунок Б.5 - Розгортання програми

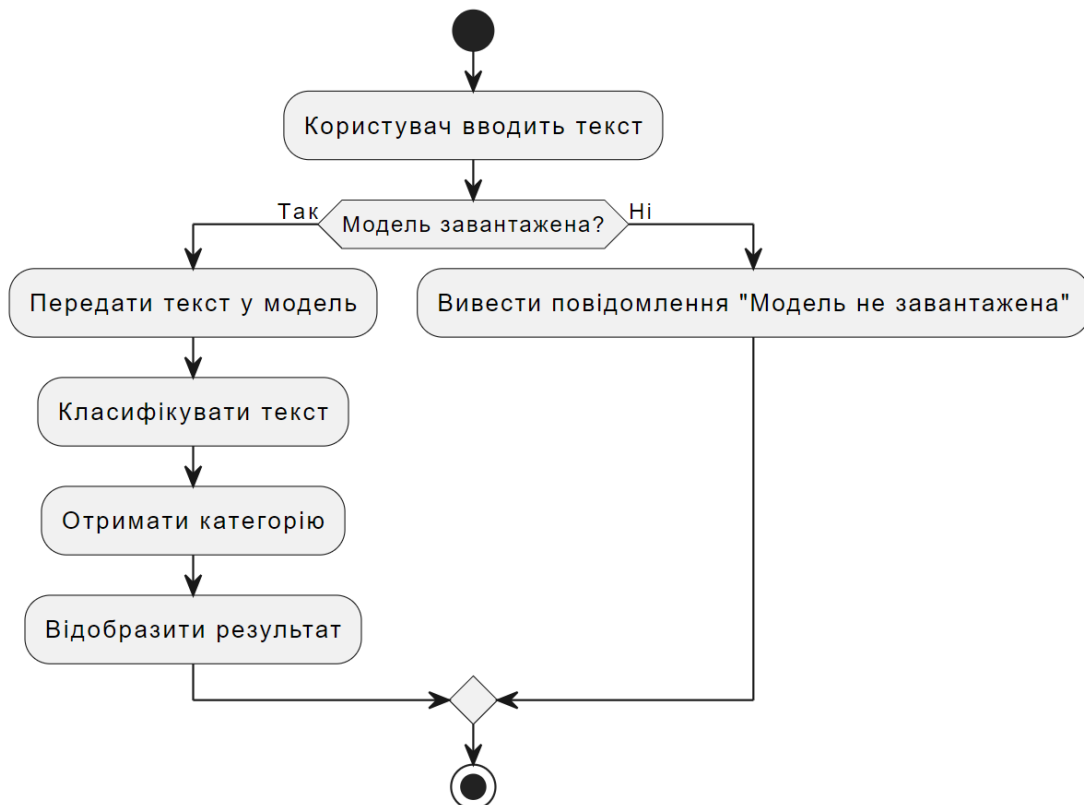


Рисунок Б.6 - Схема класифікації тексту

Додаток В (обов'язковий)**Лістинг програми**

```
import tkinter as tk

from tkinter import ttk, messagebox
import joblib
from sklearn.feature_extraction.text import TfidfVectorizer
from nltk.tokenize import word_tokenize
from nltk.corpus import stopwords
from nltk.stem import WordNetLemmatizer
import os
import nltk

nltk.download("punkt")
nltk.download("wordnet")
nltk.download("stopwords")

# Load the trained model
MODEL_PATH = 'text_classifier.pkl'
try:
    model = joblib.load(MODEL_PATH)
except FileNotFoundError:
    messagebox.showerror("Error", f"Trained model not found. Please run 'train_model.py' first.")
    exit()

# Preprocessing tools
lemmatizer = WordNetLemmatizer()
stop_words = set(stopwords.words('english'))
```

```

vectorizer = joblib.load('tfidf_vectorizer.pkl') if
os.path.exists('tfidf_vectorizer.pkl') else None

# Function for text preprocessing
def preprocess_text(text):
    # Tokenization
    tokens = word_tokenize(text.lower())

    # Removing stop words and lemmatization
    processed_tokens = [lemmatizer.lemmatize(word) for word in tokens if
word.isalnum() and word not in stop_words]

    return " ".join(processed_tokens)

# Function to classify input text
def classify_text():
    input_text = text_input.get("1.0", tk.END).strip()
    if not input_text:
        result_label.config(text="Please enter text to classify.", foreground="red")
        return

    # Preprocess the input text
    processed_text = preprocess_text(input_text)

    if not vectorizer:
        messagebox.showerror("Error", "TF-IDF Vectorizer not found. Please train
and save the vectorizer first.")
        return

    # Vectorize the input text

```

```

text_features = vectorizer.transform([processed_text])

# Predict using the loaded model
prediction = model.predict(text_features)[0]
result_label.config(text=f"Predicted          Category:          {prediction}",
foreground="green")

# Function to clear input and result
def clear_text():
    text_input.delete("1.0", tk.END)
    result_label.config(text="Result will appear here.", foreground="black")

# Function to load a new model
def load_new_model():
    global model
    file_path = tk.filedialog.askopenfilename(title="Select Model File",
filetypes=[("Pickle Files", "*.pkl")])
    if file_path:
        try:
            model = joblib.load(file_path)
            messagebox.showinfo("Success", "New model loaded successfully.")
        except Exception as e:
            messagebox.showerror("Error", f"Failed to load model: {e}")

# Function to display preprocessing steps
def show_preprocessing_steps():
    steps = (
        "1. Tokenization: Split text into words.\n"
        "2. Stop-word Removal: Remove common words like 'the', 'and'.\n"
        "3. Lemmatization: Convert words to their base form.\n"

```

```

        "4. Vectorization: Transform text into numerical features using TF-IDF."
    )
    messagebox.showinfo("Preprocessing Steps", steps)

# Set up the GUI
root = tk.Tk()
root.title("Scientific Text Classifier")

# Create and place widgets
ttk.Label(root, text="Enter Text to Classify:").pack(pady=5)
text_input = tk.Text(root, width=60, height=10)
text_input.pack(pady=5)

classify_button = ttk.Button(root, text="Classify", command=classify_text)
classify_button.pack(pady=5)

clear_button = ttk.Button(root, text="Clear", command=clear_text)
clear_button.pack(pady=5)

load_model_button = ttk.Button(root, text="Load New Model",
command=load_new_model)
load_model_button.pack(pady=5)

preprocessing_button = ttk.Button(root, text="Show Preprocessing Steps",
command=show_preprocessing_steps)
preprocessing_button.pack(pady=5)

result_label = ttk.Label(root, text="Result will appear here.", font=("Arial", 12))
result_label.pack(pady=10)

```

```
# Run the app  
root.mainloop()
```

Додаток Г (обов'язковий)

Протокол перевірки магістерської кваліфікаційної роботи на наявність текстових запозичень

Назва роботи: Автоматизована система класифікації навчальних текстів дисциплін з використанням нейронних мереж

Тип роботи: магістерська кваліфікаційна робота

Підрозділ: кафедра АІТ

Показники звіту подібності Turnitin

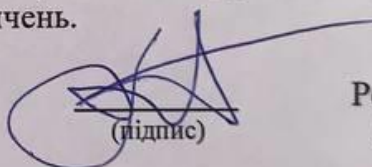
Оригінальність 96,0 %

Схожість 4,0 %

Аналіз звіту подібності (відмітити потрібне)

- Запозичення, виявлені у роботі, оформлені коректно і не містять ознак плагіату.
 - Виявлені у роботі запозичення не мають ознак плагіату, але їх надмірна кількість викликає сумніви щодо цінності роботи і самостійності її автора. Роботу направити на розгляд експертної комісії кафедри.
 - Виявлені у роботі запозичення є недобросовісними і мають ознаки плагіату та/або в ній містяться навмисні спотворення тексту, що вказують на спроби приховування недобросовісних запозичень.

Особа, відповідальна за перевірку

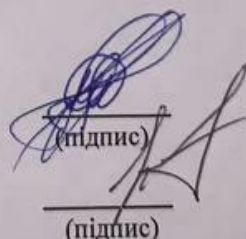


(підпис)

Роман МАСЛІЙ
(ім'я, ПРІЗВИЩЕ)

Ознайомлені з повним звітом подібності, який був згенерований системою Unichesk щодо роботи.

Автор роботи



(підпис)

Антон ТКАЧЕНКО
(ім'я, ПРІЗВИЩЕ)

Керівник роботи

Володимир СЕВАСТ'ЯНОВ
(ім'я, ПРІЗВИЩЕ)