

АНДРІЙЧУК Ярослав Сергійович

**Програмний модуль для виявлення фейкових
новин із використанням TensorFlow / Software
module for detecting fake news using TensorFlow**

спеціальність: 122 - Комп'ютерні науки
освітньо-професійна програма - Комп'ютерні науки

Кваліфікаційна робота

Виконав студент групи КН-41
Я. С. Андрійчук

Науковий керівник:
к.т.н., Д. І. Загородня

Кваліфікаційну роботу
допущено до захисту:

" ____ " _____ 20 ____ р.

Завідувач кафедри
_____ **М. П. Комар**

ТЕРНОПІЛЬ - 2024

Факультет комп'ютерних інформаційних технологій
Кафедра інформаційно-обчислювальних систем і управління
Освітній ступінь «бакалавр»
спеціальність: 122 «Комп'ютерні науки»
освітньо-професійна програма – «Комп'ютерні науки»

ЗАТВЕРДЖУЮ
Завідувач кафедри
_____ М.П. Комар
« ____ » _____ 2023 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТУ
АНДРІЙЧУК Ярослав Сергійович
(прізвище, ім'я, по батькові)

1. Тема кваліфікаційної роботи (*укр./англ. мовами*): Програмний модуль для виявлення фейкових новин із використанням TensorFlow / Software module for detecting fake news using TensorFlow

керівник роботи к.т.н., доцент Д.І. Загородня

керівник роботи Загородня Діана Іванівна, к.т.н., доцент

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом по університету від 12 грудня 2023 р. № 753.

2. Строк подання студентом закінченого проекту 15 травня 2024 р.

3. Вихідні дані до кваліфікаційної роботи: завдання на кваліфікаційну роботу студента, наукові статті, технічна література.

4. Основні питання, які потрібно розробити:

- провести аналіз предметної області;
- провести аналіз існуючих аналогів виявлення фейкових новин;
- зробити постановку задач дослідження;
- розробити структуру модуля;
- розробити алгоритмічне забезпечення модуля;
- провести реалізацію програмного забезпечення;
- натренувати модель;
- провести тестування програмного модуля.

5. Перелік графічного матеріалу у роботі

- існуючі аналоги виявлення фейкових новин;
- структура програмного модуля;
- алгоритм роботи програмного модуля.
- результатами експериментальних досліджень

6. Консультанти розділів проекту

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		Завдання видав	Завдання прийняв
Н. контроль	к.т.н., доцент Д.І. Загородня		

7. Дата видачі завдання 12 грудня 2023 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів кваліфікаційної роботи	Примітка
1	Затвердження теми кваліфікаційної роботи, ознайомлення з літературними джерелами та складання плану роботи.	до 01.01. 2024 р.	
2	Написання 1 розділу кваліфікаційної роботи	до 01.03. 2024 р.	
3	Написання 2 розділу кваліфікаційної роботи	до 01.04.2024 р.	
4	Написання 3 розділу кваліфікаційної роботи	до 01.05. 2024 р.	
5	Представлення попереднього варіанту кваліфікаційної роботи, перевірка та внесення змін керівником	до 15.05.2024 р.	
6	Опрацювання зауважень та представлення завершеного варіанту кваліфікаційної роботи. Підготовка супроводжуючих документів.	до 20.05.2024 р.	
7	Перевірка кваліфікаційної роботи на оригінальність тексту у системі «Unichек».	до 10.06.2024 р.	
8	Оформлення кваліфікаційної роботи та отримання допуску до захисту	до 14.06.2024 р.	
9	Подання кваліфікаційної роботи до захисту на засіданні атестаційної комісії.	до 14.06. 2024 р.	

Студент _____ Я.С. Андрійчук
(підпис) (прізвище та ініціали)

Керівник кваліфікаційної роботи _____ Д.І. Загородня
(підпис) (прізвище та ініціали)

АНОТАЦІЯ

Кваліфікаційна робота на тему «Програмний модуль для виявлення фейкових новин із використанням TensorFlow» на здобуття освітнього ступеня «бакалавр» зі спеціальності 122 «Комп'ютерні науки» освітньої програми «Комп'ютерні науки» написана обсягом в 51 сторінок і містить 21 ілюстрацій, 1 таблицю та 27 використаних джерел.

Метою роботи є дослідження підходів до розроблення програмного модуля виявлення фейкових новин.

Методами розроблення обрано метод аналізу для дослідження існуючих підходів до виявлення фейкових новин, метод синтезу (для поєднання переваг існуючих методів виявлення), методи моделювання (для представлення та дослідження процесів обробки тексту), метод порівняльного аналізу (для оцінювання точності моделей виявлення фейкових новин).

Внаслідок виконання роботи обґрунтовано структурно-орієнтований підхід до розроблення програмного модуля виявлення фейкових новин, який дає змогу перевіряти новини на хибність.

Результати дослідження можуть бути використані редакціями ЗМІ, медіаорганізаціями, дослідниками, аналітиками, а також широкою громадськістю.

Ключові слова: ПРОГРАМНИЙ МОДУЛЬ, НЕЙРОННІ МЕРЕЖІ, МАШИННЕ НАВЧАННЯ, TENSORFLOW, ВИЯВЛЕННЯ ФЕЙКОВИХ НОВИН

ANNOTATION

Qualification Thesis on the topic "Software Module for Fake News Detection Using TensorFlow" for the attainment of the Bachelor's degree in the specialty 122 "Computer Science" of the educational program "Computer Science" is written with a volume of 51 pages and includes 21 figures, 1 table and 27 sources.

The purpose of the work is to explore approaches to developing a software module for detecting fake news.

The development methods selected include the method of analysis (for studying existing approaches to detecting fake news), the method of synthesis (for combining the advantages of existing methods), modeling methods (for representing and investigating text processing processes), and the method of comparative analysis (for assess the accuracy of detecting fake news models).

As a result of the work, a structurally-oriented approach to developing a software module for fake news detection has been substantiated, which allows for the verification of news for falseness. The research results can be used by media editors, media organizations, researchers, analysts, and the general public.

Keywords: SOFTWARE MODULE, NEURAL NETWORKS, MACHINE LEARNING, TENSORFLOW, FAKE NEWS DETECTION.

ЗМІСТ

ВСТУП.....	6
1 АНАЛІЗ ВИЯВЛЕННЯ ФЕЙКОВИХ НОВИН.....	8
1.1 Опис предметної області виявлення фейкових новин	8
1.2 Огляд і аналіз існуючих аналогів виявлення фейкових новин.....	10
1.3 Постановка задачі дослідження.....	18
2 АЛГОРИТМІЧНЕ ТА ІНФОРМАЦІЙНЕ ЗАБЕЗПЕЧЕННЯ МОДУЛЯ.....	20
2.1 Загальна структура проєктованого модуля	20
2.2 Алгоритмічне забезпечення	23
3 ПРОГРАМНО-ТЕХНОЛОГІЧНЕ ЗАБЕЗПЕЧЕННЯ МОДУЛЯ.....	30
3.1 Реалізація програмного забезпечення.....	30
3.2 Тренування моделі	36
3.3 Тестування розробленої програми	38
ВИСНОВКИ.....	40
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	42
Додаток А Копія публікації.....	45
Додаток Б Код програмного модуля	50

ВСТУП

Сьогодні інформаційний простір став більш заплутаний і доступніший завдяки швидкому розвитку технологій та соціальних мереж. Завдяки цьому, доступ до інформації став миттєвим, а користувачі можуть легко звертатися до новин та думок різних джерел. Однак, разом з цим зростає проблема розповсюдження фейкових новин та дезінформації, що створює загрозу для суспільства та демократичних процесів.

Термін "фейкові новини" став особливо актуальним та політизованим під час президентських виборів у США 2016 року. Він став частково змінювати своє первинне значення та використовувався в різних контекстах. Спочатку він означав недостовірні новини, що зазвичай створюються з метою обману, імітуючи зміст традиційних новинних ЗМІ[1,2]. Однак згодом термін поширився і став застосовуватися до різних категорій контенту, таких як сатира, пародія, маніпуляція та пропаганда.

У цьому проекті термін "фейкові новини" буде використовуватися для опису цілком сфабрикованої інформації, яка намагається здаватися легітимною, з особливим акцентом на її дезінформаційний аспект[3].

Фейкові новини зазвичай мають різні мотивації, часто фінансові або ідеологічні. Перша мета полягає в тому, що велика кількість трафіку, отримана від вірусного поширення фейкових новин, може приносити великий прибуток від реклами. Друга причина зазвичай більш складна і залежить від мотивацій автора. Наприклад, фейкові новини можуть створюватися з метою дискредитації політичного опонента або підвищення рейтингу іншої сторони[4].

Хоча вплив фейкових новин на голосування залишається під сумнівом, вони можуть мати негативний вплив на суспільство та становити реальну небезпеку. Наприклад, це може сприяти поширенню недовіри до вакцинації дітей або навіть призвести до міжнародної напруги.

Фейкові новини, або недостовірна інформація, можуть мати серйозні наслідки, включаючи вплив на громадську думку, політичні процеси, фінансові ринки, та навіть загрозу національній безпеці. У зв'язку з цим, виникає необхідність

в розробці ефективних інструментів для виявлення та боротьби з фейковими новинами.

В рамках даного дослідження планується розробити програмний модуль для виявлення фейкових новин з використанням сучасних методів машинного навчання (Machine Learning). Проект спрямований на розробку модуля, який забезпечить автоматизоване виявлення фейкових новин на ранній стадії їх поширення, що дозволить ефективно протидіяти поширенню дезінформації та зберегти довіру до інформаційного середовища.

Мета кваліфікаційної роботи полягає в розробці та реалізації програмного модуля, який може аналізувати тексти новин та оцінювати їхню достовірність з використанням методів машинного навчання. Робота спрямована на створення інструменту, який може допомогти користувачам розпізнавати фейкові новини та приймати обґрунтовані рішення щодо їх довіри.

Об'єкт дослідження – модуль виявлення фейкових новин.

Предметна область охоплює методи та алгоритми машинного навчання, які можна використовувати для виявлення фейкових новин, а також аналіз поточних рішень у цій сфері.

Програмний модуль для розпізнавання фейкових новин із використанням TensorFlow може знайти застосування в різних сферах: медіа та Інформаційні технології, соціальні мережі та онлайн-платформи. фінансова сфера, дослідження громадської думки. освітня сфера, корпоративна безпека та репутаційний менеджмент.

1 АНАЛІЗ ВИЯВЛЕННЯ ФЕЙКОВИХ НОВИН

1.1 Опис предметної області виявлення фейкових новин

Фейкові новини стали значущою проблемою в сучасному світі. Вони можуть виникати в різних сферах, включаючи політику, соціальні мережі, фінанси та інші. Швидкість поширення і вплив фейкових новин можуть мати серйозні наслідки для громадської думки та соціально-політичного середовища.

Існують різноманітні технологічні рішення для виявлення фейкових новин, включаючи методи машинного навчання та обробки природної мови. TensorFlow, як потужна бібліотека машинного навчання, може бути ефективним інструментом для розробки програмного модуля для виявлення фейкових новин.

Виявлення фейкових новин є складною задачею через їхню різноманітність та складність. Ці новини можуть містити неправдиві факти, маніпулювати інформацією та викликати певні емоційні реакції у читачів. Тому ефективні методи аналізу та класифікації є важливими для точного виявлення фейкових новин.

Підвищення рівня медіаграмотності серед користувачів стає все більш важливим в контексті боротьби з фейковими новинами. Розробка програмних засобів для виявлення фейкових новин може сприяти усвідомленню громадськості про проблему фейкових новин та їхні наслідки.

Швидкий розвиток технологій штучного інтелекту та машинного навчання відкриває нові можливості для виявлення фейкових новин. Програмні модулі, які використовують інтелектуальні алгоритми для аналізу текстів та виявлення маніпуляцій, можуть стати ефективними інструментами для боротьби з цією проблемою.

Виявлення фейкових новин має на меті автоматичну ідентифікацію фейкових новин. Існуючі традиційні методи FND на основі машинного навчання вимагають розробки функцій. Відповідно до ознак, які використовувалися в моделях, ці методи можна умовно розділити на три категорії: лінгвістичні особливості, часово-структурні особливості та гібридні ознаки.[5]

Підходи, засновані на лінгвістичних ознаках, виявляють фейкові новини на основі текстового контенту (Castillo et al., 2011). Наприклад, Castillo et al. (2011)

використовували різноманітні лінгвістичні особливості, такі як спеціальні символи, символи-смайлики, сентиментальні слова тощо[6]. Popat (2017) досліджував особливості мовного стилю, такі як асертивні дієслова та дієслова [7]. Крім того, деякі методи досліджують тимчасово-структурну особливість (наприклад, особливість по-ширення) на основі соціальних мереж для виявлення фейкових новин (Jin et al., 2013)[8]. Наприклад, Wu et al. (2015) запропонували SVM для вивчення моделей поширення високого порядку[9]. Sampson et al. (2016) використовували неявні зв'язки між фрагментами розмови, щоб правильно класифікувати екстрені розмови [10].

Також пропонуються гібридні підходи, які поєднують різні типи функцій для виявлення фейкових новин. Наприклад, Sun et al. (2013) поєднали новинний контент, користувацькі та мультимедійні функції [11]. Ma et al. (2015) об'єднали часові варіації функцій на основі контенту, користувачів та дифузії в міру розвитку поширення новин [12]. Незважаючи на те, що ці традиційні методи машинного навчання досягають багатообіцяючих результатів, вони значною мірою покладаються на трудомістку розробку функцій.

По-перше, глибоке навчання може уникнути фіч-інжинірингу та повною мірою використовувати переваги своєї потужної експресивної сили для моделювання особливостей вхідних новин. Наприклад, Ma et al. (2016) змоделювали послідовний зв'язок між новинними публікаціями за допомогою рекурентних нейронних мереж[13]. Yu et al. (2017) використовували згорткові нейронні мережі для представлення високорівневих семантичних зв'язків між новинними публікаціями[14]. Bian et al. (2020) використовували графові нейронні мережі (GCN) із спрямованим як зверху, так і знизу вгору графом поширення чуток, щоб вивчити закономірності його поширення та дисперсії[15]. Khattar et al. (2019) запропонували мультимодальний варіаційний автокодер (MVAE) для вилучення прихованих мультимодальних представлень мультимедійних новин[16].

1.2 Огляд і аналіз існуючих аналогів виявлення фейкових новин

Цей аналіз включає огляд функціональності, ефективності та обмежень різних сервісів, програм та технологій, що застосовуються для виявлення фейкових новин. Його метою є визначення найкращих практик та інструментів для боротьби з дезінформацією та забезпеченням достовірності та цілісності інформації в медіа-середовищі.

Засоби визначення фейкових новин - це інструменти, програми або сервіси, які призначені для виявлення, аналізу та відстеження фейкової інформації та дезінформації в Інтернеті та медіа-просторі. Задача таких сервісів полягає у забезпеченні інформаційної безпеки та боротьбі з дезінформацією. Вони допомагають користувачам визначати недостовірну або маніпулятивну інформацію та надають засоби для перевірки достовірності новин та джерел їх походження. Шляхом аналізу тексту, візуальних елементів, контексту та інших параметрів, такі сервіси намагаються ідентифікувати фейкові новини та запобігти їх поширенню.

Задача таких сервісів – забезпечення інформаційної безпеки та боротьба з дезінформацією.

Насамперед потрібно визначити основні завдання засобів визначення фейкових новин:

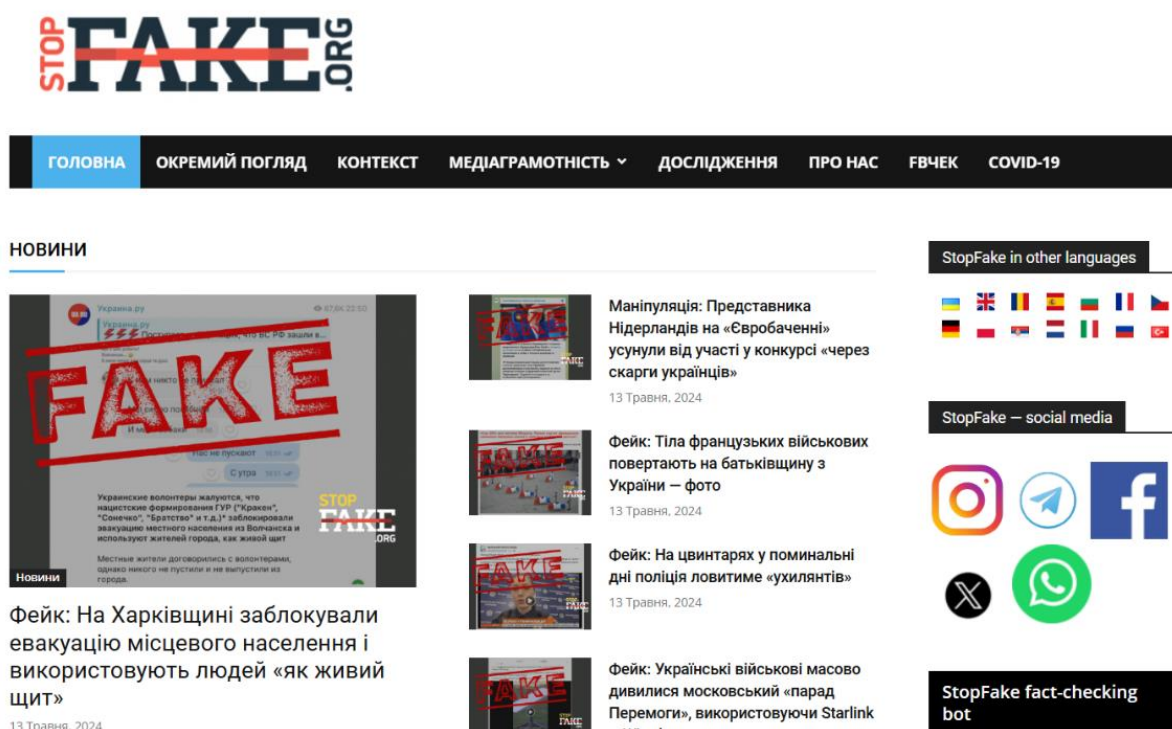
Виявленні фейкової інформації. Вони намагаються ідентифікувати недостовірну або маніпулятивну інформацію шляхом аналізу тексту, візуальних елементів, контексту та інших параметрів.

Перевірка достовірності. Засоби визначення фейкових новин допомагають користувачам перевіряти правдивість новин та достовірність джерел їх походження.

Запобігання поширенню дезінформації. Шляхом виявлення фейкових новин та надання користувачам інструментів для перевірки інформації, ці сервіси допомагають запобігти її подальшому поширенню та зменшити вплив дезінформації на суспільство.

Тепер розглянемо характеристику наявних засобів виявлення фейкових новин в Україні та за її межами.

1. StopFake.org (рисуюнок 1.1) - це незалежний медійний проект, який був заснований у 2014 році в Україні в реакції на поширення фейкових новин та дезінформації, особливо під час конфлікту на сході України та анексії Криму. Головною метою проекту є виявлення, аналіз та спростування фейкових новин та маніпуляцій у медійному просторі.



Рисуюнок 1.1 – Головна сторінка StopFake.org

StopFake.org використовує методи журналістського розслідування та перевірки фактів для виявлення дезінформації (рисуюнок 1.2). Вони аналізують медійні матеріали, перевіряють їх на достовірність, шукають джерела та факти, що підтверджують або спростовують інформацію.

Крім того, StopFake.org також надає освітні матеріали про фейкові новини та методи їх виявлення для широкої громадськості (рисуюнок 1.3). Вони співпрацюють з медіа, активістськими групами та міжнародними організаціями для боротьби з дезінформацією.[17]

2. VoxCheck, що знаходиться на сайті voxukraine.org (рисунок 1.4), – це український медійний проєкт, спрямований на фактчекінг та перевірку інформації, що циркулює в українських ЗМІ та політичному просторі.

VoxCheck зосереджений на виявленні та висвітленні міфів, фейкових новин та недостовірної інформації, яка може впливати на суспільство та політичний діалог в Україні.

Основним методом перевірки інформації є рецензування. Аналітики та волонтери вивчають виступи та заяви політиків, піддаючи їх моніторингу, а потім виділяють з них ті висловлювання, які потребують перевірки.

Такий підхід дозволяє перевіряти достовірність інформації та забезпечує широкому загалу доступ до перевірених фактів.[18]

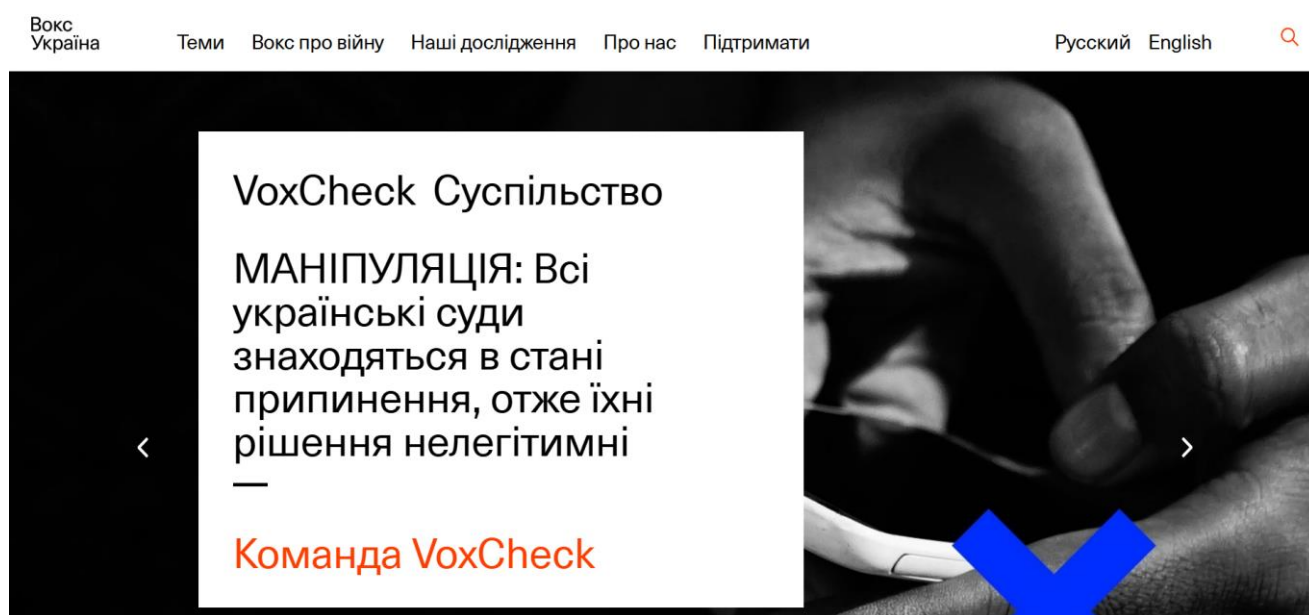


Рисунок 1.4 – Головна сторінка voxukraine.org

3. Snopes.com – це відомий медійний ресурс (рисунок 1.5), який спеціалізується на фактчекінгу та перевірці правдивості різних тверджень, міфів, чуток та фейкових новин, що поширюються в Інтернеті. Заснований у 1994 році, сайт надає достовірну інформацію та контекст щодо різноманітних тем, щоб допомогти користувачам розрізняти правдиву інформацію від міфів та маніпуляцій.[19]



Рисунок 1.5 – Головна сторінка Snopes.com

4. FactCheck.org – це незалежний фактчекінговий веб-сайт (рисуюнок 1.6), що заснований у 2003 році.

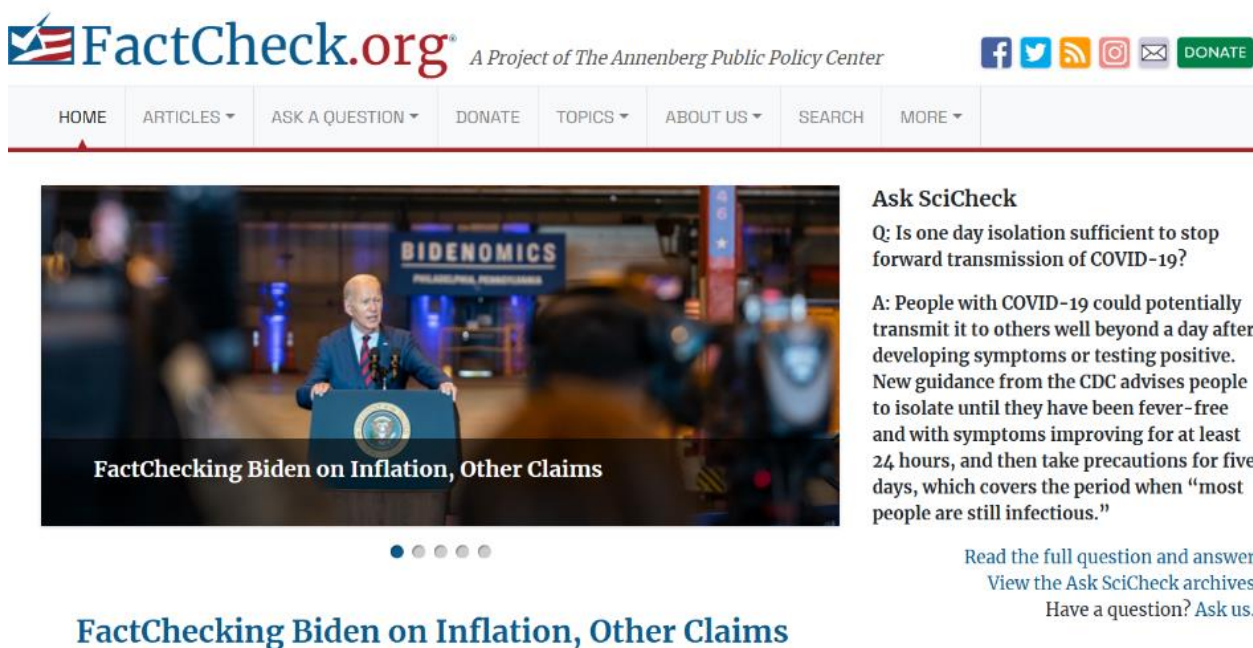


Рисунок 1.6 – головна сторінка FactCheck.org

Основною метою сайту є перевірка правдивості тверджень та стверджень, що з'являються у політичних дебатах, рекламних кампаніях та засобах масової інформації. Команда аналітиків та журналістів перевіряє факти, використовуючи надійні джерела і дані, і публікує свої висновки для сприяння об'єктивному освітленню громадськості.

Далі подано автоматизовані системи для перевірки фактів, що ґрунтуються на глибокому машинному навчанні і не потребують значного обсягу людської праці, як і вищезазначені сервіси.[20]

1. Fabula AI - це компанія, що спеціалізується на розробці технологій для виявлення та боротьби з дезінформацією в мережі Інтернет. Вони використовують штучний інтелект і аналіз даних для ідентифікації шаблонів розповсюдження фейкових новин та маніпуляційної інформації. Fabula AI намагається створити алгоритми, які можуть розпізнавати та фільтрувати маніпулятивний контент у реальному часі, щоб попередити його поширення та зменшити вплив дезінформації на громадську думку. [21]

Візуалізація моделі розповсюдження фейкових новин проти реальних подана на Рисунку 1.7.

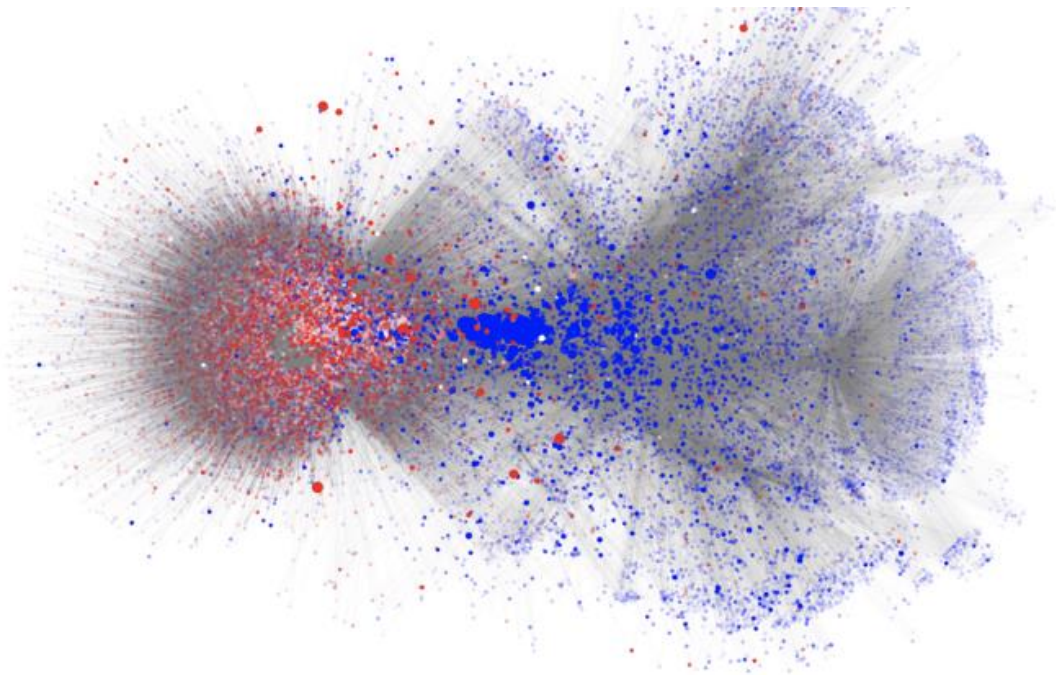


Рисунок 1.7 – Візуалізація Fabula моделі розповсюдження фейкових новин проти реальних [22]

Користувачі, які переважно поширюють фейкові новини, пофарбовані в червоний колір, а користувачі, які взагалі не поширюють фейкові новини, пофарбовані в синій колір, що, за словами Фабули, свідчить про чіткий поділ на окремі групи та «миттєво впізнавану різницю в моделях розповсюдження».

У своїх власних тестах Fabula каже, що її алгоритми змогли ідентифікувати 93 % «фейкових новин» протягом кількох годин після розповсюдження.[22]

2. Fakebox – це інструмент аналізу та виявлення фейкових новин, який використовує технології штучного інтелекту (рисунок 1.8). Ця система аналізує великі обсяги текстової інформації, використовуючи алгоритми машинного навчання та обробку природної мови

The screenshot displays the 'Newsbox' interface for analyzing news content. It includes input fields for URL, Title, and Content, followed by an 'Analyze' button. The results are presented in three sections: Content keywords, Title entities, and Content entities. A table shows the 'Decision' and 'Score' for the Domain, Title, and Content. The 'JSON' section displays the raw analysis data.

Content keywords

- senate majority leader mitch mcconnell
- debt ceiling
- senate majority leader
- republicans whose support
- representatives speaker paul
- president donald trump
- trump
- washington
- u.s. credit rating
- party leader

Title entities

- Trump person
- Republicans group

Content entities

- WASHINGTON person
- Reuters organization
- Donald Trump person
- Thursday date
- Republicans group

Field	Decision	Score
Domain	trusted	
Title	Impartial	0.688883364200592
Content	Impartial	0.894371509552002

JSON

```
{
  "success": true,
  "title": {
    "decision": "impartial",
    "score": 0.688883364200592,
    "entities": [
      {
        "text": "Trump",
        "start": 0,
        "end": 4,
        "type": "person"
      },
      {
        "text": "Republicans",
        "start": 24
```

Рисунок 1.8 – Невелика демонстрація роботи Fakebox [23]

Fakebox шукає специфічні ознаки або патерни, що характеризують фейкові новини, та надає звіт про ймовірність того, що конкретний матеріал є дезінформацією. Цей інструмент може бути використаний для автоматичної перевірки новин та визначення їх достовірності. [24]

Для кращого розуміння функціональних можливостей та відмінностей доступних інструментів для виявлення фейкових новин, у таблиці 1.1 наведено порівняльну характеристику деяких із них.

Таблиця 1.1 – Аналіз існуючих сервісів визначення фейкових новин

Сервіс	StopFake.org	VoxCheck	Snopes.com	FactCheck.org	Fabula AI	Fakebox
Тип	Веб-сайт	Веб-сайт	Веб-сайт	Веб-сайт	AI	API
Метод Виявлення	Аналіз новин, інтерв'ю, відео, тверджень	Аналіз новин, подій	Аналіз новин, відео, чутки	Аналіз інтерв'ю, відео, реклами, звіти	Аналіз тверджень	Аналіз новин, статей, заголовків, тверджень
Тип Виявлення	Ручна перевірка	Ручна перевірка	Ручна перевірка	Ручна перевірка	Автоматична перевірка	Автоматична перевірка
Мови	uk, ru, en, es, de, pl	uk, ru, en	en	en, es	Усі мови	en
Вартість	Безкоштовний	Безкоштовний	Безкоштовний	Безкоштовний	Закритий доступ	Безкоштовний
Сфери	Політика	Політика	Політика Релігія	Політика, соціальні сфери	Усі сфери	Усі сфери
Створено	Україна	Україна	США	США	Велика Британія	За кордоном

Провівши аналіз існуючих сервісів перевірки достовірності новин, було виведено наступні висновки:

- більшість з цих проектів потребують значної людської праці для перевірки інформації, що ускладнює їх підтримку та збільшує витрати з фінансової точки зору.
- швидкість реакції існуючих рішень не відповідає потребам медіа-гігантів, які примушені залучати модераторів для оперативного аналізу контенту у соціальних мережах та реагування на фейкову інформацію.

- на даний момент відсутній сервіс автоматичного визначення фейкових новин з підтримкою української мови.
- масштабованість цих сервісів обмежена.
- ефективність таких сервісів у обробці інформації значно відстає від обсягів контенту, який генерується у соціальних мережах та ЗМІ.

1.3 Постановка задачі дослідження

Провівши аналіз всіх зазначених даних, можна зробити висновок, що фейкові новини стали серйозною проблемою у наш час, а збитки від них лише зростають.

При існуванні різноманітних методів виявлення фейкових новин, більшість з них базуються на використанні людських ресурсів, що ускладнює процес та дає зловмисникам перевагу у поширенні неправдивої інформації.

Застосування методів аналізу природної мови, особливо шляхом використання штучного інтелекту, стає все більш популярним, зокрема за допомогою нейронних мереж, які є ефективними інструментами для класифікації.

У контексті розробки інформаційної технології для виявлення фейкових новин, виконання таких завдань передбачає:

- провести огляд системи для розробки програмного модуля
- провести аналіз методів для виявлення фейкових новин
- збір та аналіз вибірки фейкових новин. Першим кроком є ретельний аналіз сучасного стану інформаційного простору, виявлення проблем фейкових новин та визначення потреб користувачів у програмному засобі для їх виявлення та фільтрації.
- моделювання, включаючи обробку даних для навчання, огляд та порівняння архітектур нейронних мереж та класифікаторів, навчання моделей та тестування їх ефективності.
- аналіз поданих новин на наявність фейкової інформації.
- реалізація програмного модуля за допомогою мови програмування Python. На основі розробленої моделі машинного навчання створюється

програмний модуль, який може приймати вхідні дані у вигляді новинних статей або зображень та визначати їх автентичність.

- провести тестування розробленого програмного модуля.

Ця робота відрізняється від інших досліджень виявлення фейкових новин тим, що вона:

Використовує комплексну модель інформаційних факторів: Дослідження буде охоплювати широкий спектр інформаційних факторів, які не завжди враховуються в інших методах виявлення.

Застосовує передові методи аналізу даних: Використовуватимуться передові методи аналізу даних, такі як машинне навчання та обробка природної мови, для виявлення закономірностей та зв'язків між інформаційними факторами і достовірністю новин.

Фокусується на інтерпретації та оцінці результатів: Інтерпретувати їхні результати, оцінювати їхню точність.

В результаті виконання перелічених етапів буде створена інформаційна технологія для виявлення фейкових новин, що дозволить ефективно виявляти неправдиву інформацію у новинах.

2 АЛГОРИТМІЧНЕ ТА ІНФОРМАЦІЙНЕ ЗАБЕЗПЕЧЕННЯ МОДУЛЯ

2.1 Загальна структура проєктованого модуля

Розробка модуля базується на використанні глибоких нейронних мереж, створених за допомогою TensorFlow. Головна ідея полягає в аналізі текстового вмісту новинних статей і прогнозуванні їхньої достовірності. Для досягнення цієї мети необхідно описати функціональну структуру системи, включаючи перелік основних функцій та їх характеристики. Важливо також визначити інформаційні зв'язки між компонентами системи, а також зв'язки з іншими системами та зовнішніми об'єктами, які є джерелами або споживачами інформації.

Функціональна структура розроблюваного модуля включає декілька ключових компонентів: модуль збору та попередньої обробки даних, модуль інженерії ознак, модуль навчання моделі, модуль оцінки моделі та модуль прогнозування. Кожен з цих компонентів виконує специфічні функції, що сприяють ефективному аналізу та класифікації новин.

Інформаційні зв'язки між компонентами системи забезпечують безперебійну передачу даних на всіх етапах обробки. Наприклад, модуль збору даних передає очищені дані до модуля інженерії ознак, який у свою чергу передає згенеровані ознаки до модуля навчання моделі. Модуль навчання моделі передає результати до модуля оцінки, а модуль прогнозування використовує результати оцінки для аналізу нових статей.

Зв'язки з іншими системами та зовнішніми об'єктами включають взаємодію з джерелами даних, такими як новинні агентства та бази даних новин, а також з користувачами, які отримують результати аналізу.

Для реалізації модуля, що здатен виявляти фейкові новини із використанням технології TensorFlow, було розроблено структуру відношення між даними. З DFD, діаграми потоків даних (рисунок 2.1), випливає, що першим етапом у структурі є взаємодія з користувачем, який вводить текст новинної статті через інтерфейс користувача. Це забезпечує початкове введення даних до системи. Після введення статті, система зберігає текст у внутрішній базі даних для подальшої обробки.

Другим етапом є передобробка даних. Введений текст статті піддається ретельній обробці, яка включає видалення зайвих символів, нормалізацію тексту, токенизацію, лематизацію та інші методи обробки, необхідні для підготовки тексту до аналізу. Оброблений текст потім передається до моделі TensorFlow для подальшого аналізу.

Третім етапом є безпосередній аналіз тексту за допомогою моделі TensorFlow. Навчена модель отримує оброблений текст і виконує класифікацію, визначаючи, чи є новина фейковою, чи справжньою. Результати цього аналізу формуються у вигляді прогнозу.

Четвертим етапом є збереження результатів прогнозу. Після того як модель генерує прогноз, результат зберігається в системі для подальшого використання. Цей результат включає інформацію про те, якою є новина і готується для відображення користувачеві.

П'ятим і останнім етапом в DFD є відображення результату користувачеві. Підготовлений результат аналізу виводиться через інтерфейс користувача, де користувач може побачити результат класифікації новинної статті. Це завершальний етап, який забезпечує зворотний зв'язок з користувачем та надає інформацію про результат аналізу.

Інформаційні зв'язки між елементами модуля є важливою складовою ефективного функціонування системи. Користувач взаємодіє з модулем через інтерфейс, вводячи текст новинної статті, яка потім передається до етапу передобробки даних. Такі дані надходять до моделі TensorFlow, яка виконує аналіз і генерує прогноз. Результат прогнозу зберігається в системі та виводиться користувачеві через інтерфейс.

Зв'язки з модулями та зовнішніми об'єктами також мають важливе значення. Інтерфейс користувача забезпечує введення новинної статті та відображення результатів аналізу. БД використовується для збереження новинних статей та результатів аналізу. Модуль передобробки даних отримує текст новинної статті та передає оброблені дані моделі TensorFlow. Модуль машинного навчання виконує

аналіз тексту та генерує прогноз, а модуль звітності формує та відображає звіт для користувача.

Ця структура забезпечує послідовне та ефективне виконання завдань з виявлення фейкових новин, використовуючи методи машинного навчання та аналізу тексту. Функціональна структура модуля включає всі необхідні етапи від введення тексту користувачем до надання результатів аналізу, що дозволяє створити надійну та ефективну систему для виявлення фейкових новин.

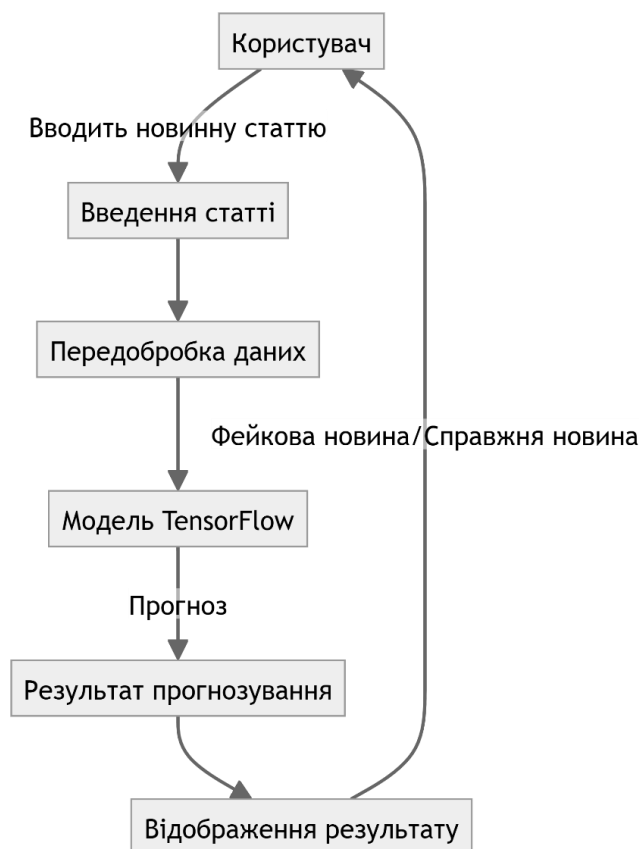


Рисунок 2.1 – Діаграма потоків даних (DFD)

Стосовно поведінки системи залежно від часу, метою цього модуля є ідентифікація фейкових статей новин на основі їх текстового вмісту.

Це досягається за допомогою спеціального типу моделі штучного інтелекту, відомого як нейронна мережа, тобто відсутня залежність від часу. Він безпосередньо не враховує фактор часу під час аналізу статей новин. Це означає, що він явно не враховує, коли була написана новинна стаття.

Хоча код явно не враховує залежність від часу, на модель може впливати опосередковано час. Це може статися, якщо навчальні дані, використовуватимуться для навчання моделі, міститимуть статті новин з різних моментів часу.

Модель може навчитися розпізнавати шаблони в стилях написання або темах, які частіше зустрічаються у фейкових новинних статтях певного періоду. Непрямий вплив часу не гарантується та залежить від якості та різноманітності тренувальних даних.

2.2 Алгоритмічне забезпечення

Щоб структурованіше розглянути алгоритмічне забезпечення була розроблена блок-схема, подана на рисунку 2.2, яка ілюструє послідовність етапів обробки даних, створення та навчання моделі. Схема складається з декількох ключових блоків, кожен з яких відповідає певному етапу модуля .

На початку проводиться завантаження необхідних даних, які будуть використані для тренування та оцінки моделі. Це буде датасет новин, який включає тексти новин та їхні відповідні мітки (фейкові або правдиві). Дані збережені у файлі та завантажені в програму для подальшої обробки. Після завантаження датасет перевіряється на наявність пустих значень. Якщо пусті значення виявлено, вони видаляються, і відповідні записи усуваються з подальшого аналізу. Якщо пустих значень немає, етап видалення пропускається.

Після успішного завантаження даних система переходить до блоку Попередня обробка даних, що свідчить про їх готовність до подальшого опрацювання. На цьому етапі дані піддаються очищенню, яке включає очищення тексту від зайвих символів, приведення тексту до нижнього регістру, видалення стоп-слів (таких як прийменники, займенники тощо) та лематизацію (приведення слів до їхньої базової форми). Цей етап є критично важливим для забезпечення якості та точності моделі. Це допомагає зменшити розмір словника та покращити якість подальшого аналізу.

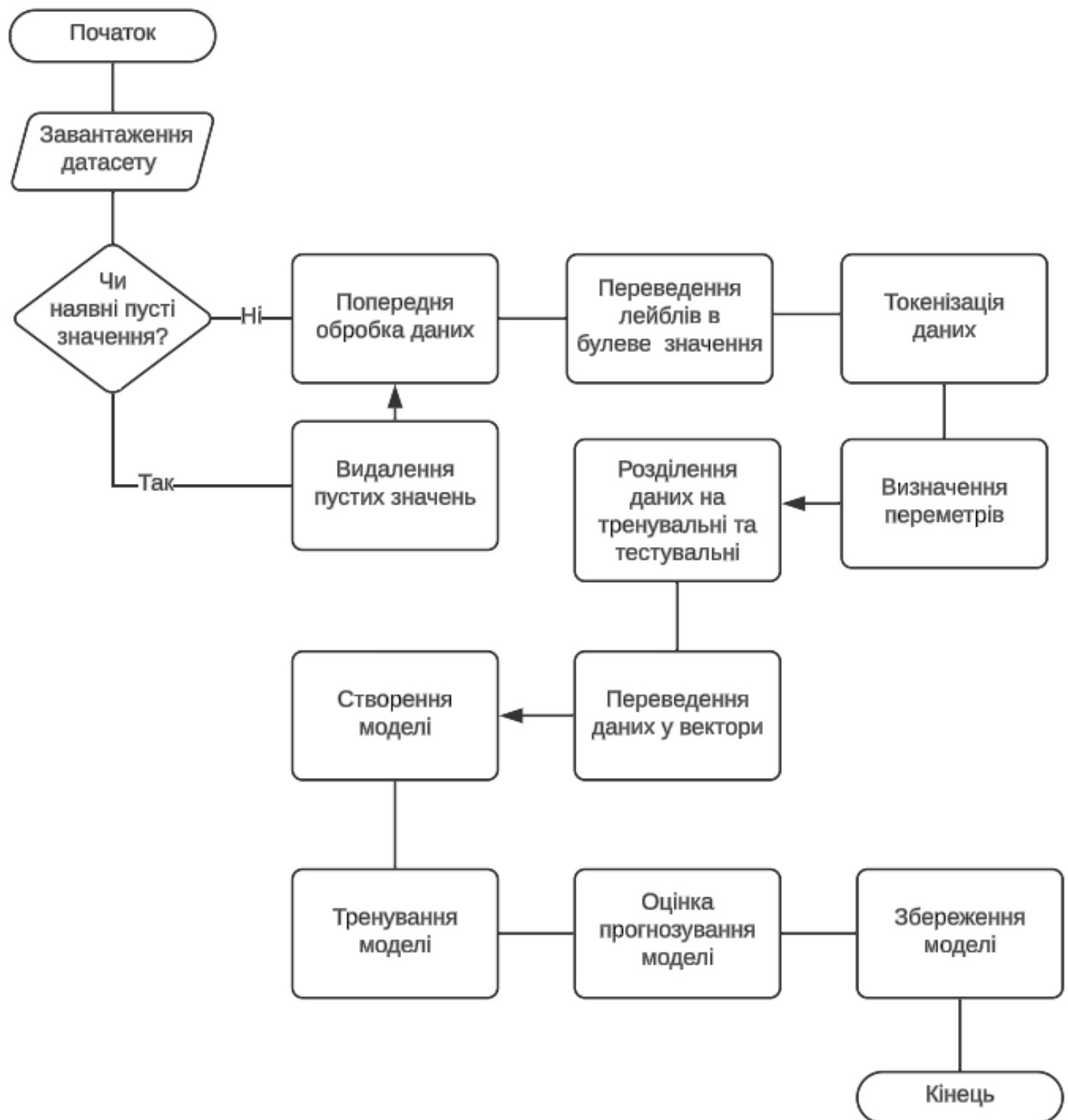


Рисунок 2.2 – Блок-схема алгоритму створення моделі

Наступним етапом є переведення лейблів у булеве значення, що спрощує подальший аналіз. Лейбли, які представляють категорії новин, кодуються у формат, зручний для використання у моделях машинного навчання. Після цього відбувається токенізація даних, що передбачає розбиття тексту новин на окремі слова або токени. Це дозволяє моделі краще розуміти структуру і зміст тексту.

Після токенізації дані розділяються на тренувальну та тестову вибірки. На тренувальних даних навчається модель машинного навчання. Тренувальний набір використовується для навчання моделі, а тестувальний - для оцінки її ефективності.

Дані також переводяться у вектори, що представляють собою числові представлення текстових даних. Це необхідно для того, щоб модель могла працювати з текстом у числовій формі.

Після підготовки даних здійснюється створення моделі. При використанні нейронних мереж, створюється архітектура мережі з вхідними, прихованими та вихідним шарами. Потім модель тренується за допомогою алгоритму оптимізації, такого як Adam, з використанням функції втрат, яка вимірює різницю між передбаченими та фактичними значеннями.

Далі слідує оцінка моделі. У цьому блоці проводиться оцінка якості натренованої моделі з використанням тестових даних. Процес оцінювання використовує метрики, такі як точність, повнота, оцінку F1 та інші, щоб перевірити, наскільки добре модель класифікує новини як фейкові або правдиві.

Точність алгоритму залежить від якості та розміру набору даних, що використовується для навчання. Зазвичай, нейронні мережі, навчені на великих наборах даних, можуть досягати високої точності (понад 90%).

Достовірність обчислень гарантується використанням перевірених та надійних бібліотек TensorFlow та Python. Алгоритм ретельно тестується, щоб уникнути помилок та збоїв.

На заключному етапі здійснюється збереження моделі. Це дозволяє використовувати її для подальшого аналізу новин без необхідності повторного навчання. Збережена модель може бути інтегрована у виробничу систему для автоматичного виявлення фейкових новин.

Таким чином, представлена блок-схема відображає комплексний підхід до розробки моделі машинного навчання для виявлення фейкових новин. Ця структура забезпечує послідовне виконання всіх необхідних етапів, починаючи від завантаження даних до збереження моделі аналізу новин. Кожен стан і перехід між ними ретельно сплановані, щоб забезпечити точність та ефективність роботи модуля.

Далі розглянемо алгоритм виявлення фейкових новин на основі даних введених користувачем (рисунок 2.3).



Рисунок 2.3 – Блок-схема алгоритму виявлення фейкових новин на основі даних користувача

Дана блок-схема, відображає основні етапи процесу, починаючи від введення новини до виведення результату. На початку користувач вводить новину, яку необхідно перевірити на правдивість. Цей етап є відправною точкою для подальшого аналізу.

Після введення новини здійснюється її токенизація. Токенизація передбачає розбиття тексту новини на окремі слова або токени. Це дозволяє моделі машинного навчання краще розуміти структуру і зміст тексту. Наступним етапом є

завантаження моделі, яка була попередньо навчена на великому датасеті новин для розпізнавання фейкових повідомлень.

Завантажена модель використовується для передбачення фейковості новини. На цьому етапі модель аналізує токенизований текст новини і визначає ймовірність того, що новина є фейковою. Результат аналізу передається на наступний етап, де здійснюється вивід результату для користувача.

Кінцевий етап алгоритму передбачає виведення результату для користувача. Користувач отримує інформацію про те, чи є введена ним новина правдивою чи фейковою. Цей результат дозволяє користувачу прийняти обґрунтоване рішення щодо достовірності новини.

Отже, блок-схема алгоритму виявлення фейкових новин на основі даних, введених користувачем, включає такі основні етапи: введення новини користувачем, токенизація новини, завантаження моделі, передбачення фейковості та вивід результату для користувача. Кожен з цих етапів забезпечує послідовну обробку новини від введення до аналізу і виведення результату, що дозволяє ефективно визначати достовірність новин на основі попередньо навченої моделі машинного навчання.

2.3 Інформаційне забезпечення

Щоб відобразити як дані організовані та як вони взаємопов'язані було створено ERD, діаграма «суть-зв'язок» (рисунок 2.4). Діаграма складається з трьох основних сутностей: `ARTICLE`, `PROCESSED_ARTICLE`, `MODEL` та `PREDICTION`, а також зв'язків між ними, що відображають взаємодію цих сутностей у контексті роботи модуля.

Основною сутністю в системі є "`ARTICLE`", яка представляє собою вихідні новини, що підлягають аналізу на предмет їх достовірності. Кожна стаття має унікальний ідентифікатор, що забезпечує однозначну ідентифікацію в базі даних. Крім того, для кожної статті зберігається її заголовок та зміст. Заголовок статті допомагає швидко зрозуміти основну тему матеріалу, а зміст містить повний текст статті, який аналізується на предмет фейковості.

Далі, статті обробляються і результати цієї обробки зберігаються в сутності "PROCESSED_ARTICLE". Ця сутність містить оброблений текст статті та числові характеристики, отримані в результаті попередньої обробки тексту. Оброблений текст представляє собою зміст статті після застосування різних методів обробки природної мови, таких як токенізація, видалення стоп-слів та інші техніки, що полегшують подальший аналіз. Числові характеристики, або фічі, є векторами, які використовуються моделями машинного навчання для аналізу тексту. Кожна оброблена стаття має унікальний ідентифікатор та посилання на оригінальну статтю, що дозволяє зберігати зв'язок між вихідними та обробленими даними.

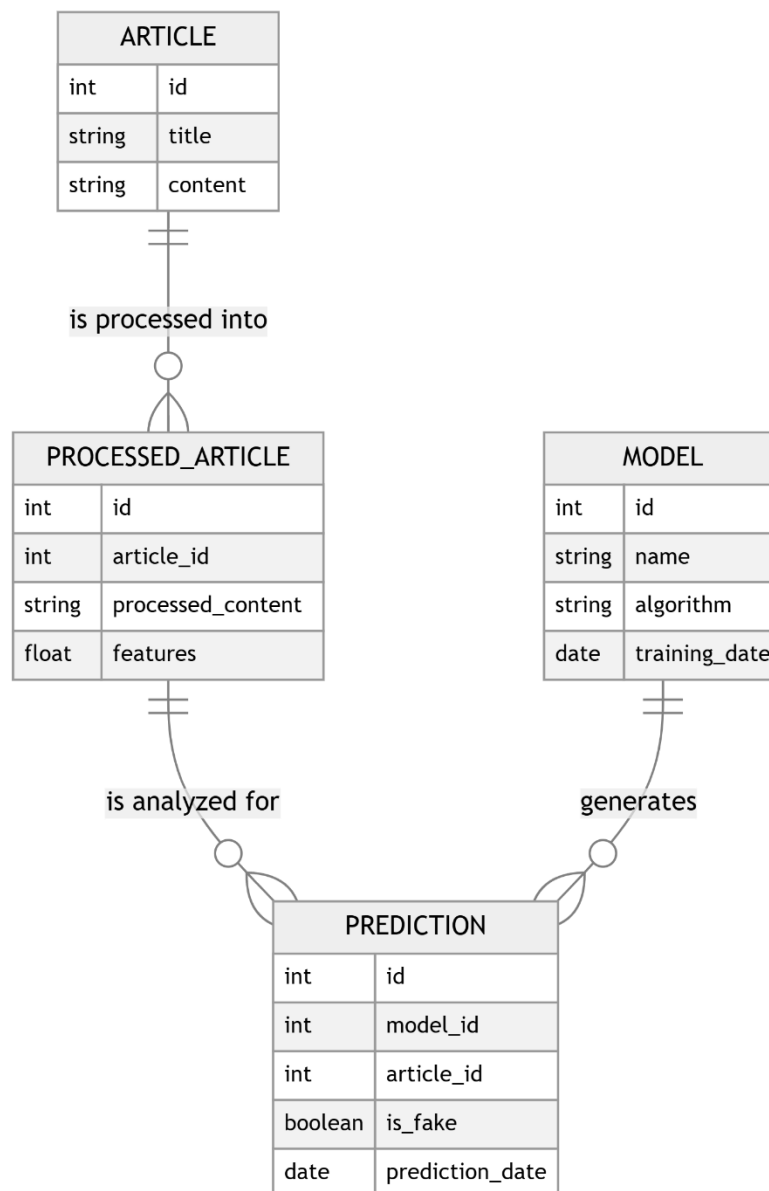


Рисунок 2.4 – Діаграма «суть-зв'язок» (ERD)

Для аналізу новин використовуються моделі машинного навчання, які представлені сутністю "MODEL". Кожна модель має унікальний ідентифікатор, назву, алгоритм, на основі якого вона побудована, та дату навчання. Назва моделі допомагає ідентифікувати її серед інших моделей, алгоритм описує метод машинного навчання, який використовується (наприклад, нейронні мережі, методи опорних векторів тощо), а дата навчання вказує на те, коли модель була навчена на тренувальному наборі даних. Це дозволяє відстежувати актуальність моделей і проводити їх оновлення за необхідності.

Результати аналізу зберігаються в сутності "PREDICTION", яка містить інформацію про те, чи є новина фейковою, а також дату прогнозу. Кожен запис про прогноз містить унікальний ідентифікатор, ідентифікатор моделі, що використовувалася для аналізу, ідентифікатор обробленої статті та логічний показник, що вказує на фейковість новини. Це дозволяє системі зберігати історію прогнозів і відстежувати ефективність моделей машинного навчання.

Взаємозв'язки між сутностями в ERD визначають їх взаємодію в системі. Вихідні статті з сутності "ARTICLE" обробляються і результати обробки зберігаються в сутності "PROCESSED_ARTICLE". Оброблені статті аналізуються за допомогою моделей, представлених у сутності "MODEL", для отримання прогнозів, які зберігаються в сутності "PREDICTION". Модель генерує прогнози на основі аналізу оброблених статей, що дозволяє визначити, чи є новина фейковою.

Таким чином, ERD описує логічну структуру системи, необхідну для ефективного виявлення фейкових новин з використанням моделі TensorFlow. Ця структура забезпечує зручне зберігання та обробку даних, що дозволяє здійснювати точний аналіз і прогнозування новин на предмет їх достовірності. Взаємозв'язки між сутностями гарантують цілісність даних і забезпечують коректне функціонування системи в цілому.

3 ПРОГРАМНО-ТЕХНОЛОГІЧНЕ ЗАБЕЗПЕЧЕННЯ МОДУЛЯ

3.1 Реалізація програмного забезпечення

Розглянемо загальну структуру системи, основні компоненти, їх взаємодію та принципи роботи відповідно до поставленої задачі. Розробка великих програм передбачає поділ функціоналу на окремі модулі. З таким підходом забезпечується більша гнучкість у налагодженні модуля, дозволяючи ефективно розміщувати бібліотеки, що сприяє економії ресурсів та часу.

Спершу було сформовано для розроблюваного модуля UML-діаграму класів (рисунок 3.1), що реалізує основну бізнес-логіку програмної системи, та її короткий опис, де зазначають призначення кожного класу.

Наведена діаграма відображає основну бізнес-логіку програмної системи моделі даних для розроблюваного модуля виявлення фейкових новин із використанням TensorFlow. Діаграма включає такі класи: NewsDetector, News, FeatureExtractor, Feature, Model та TrainingData, кожен з яких відіграє свою важливу роль у функціонуванні системи.

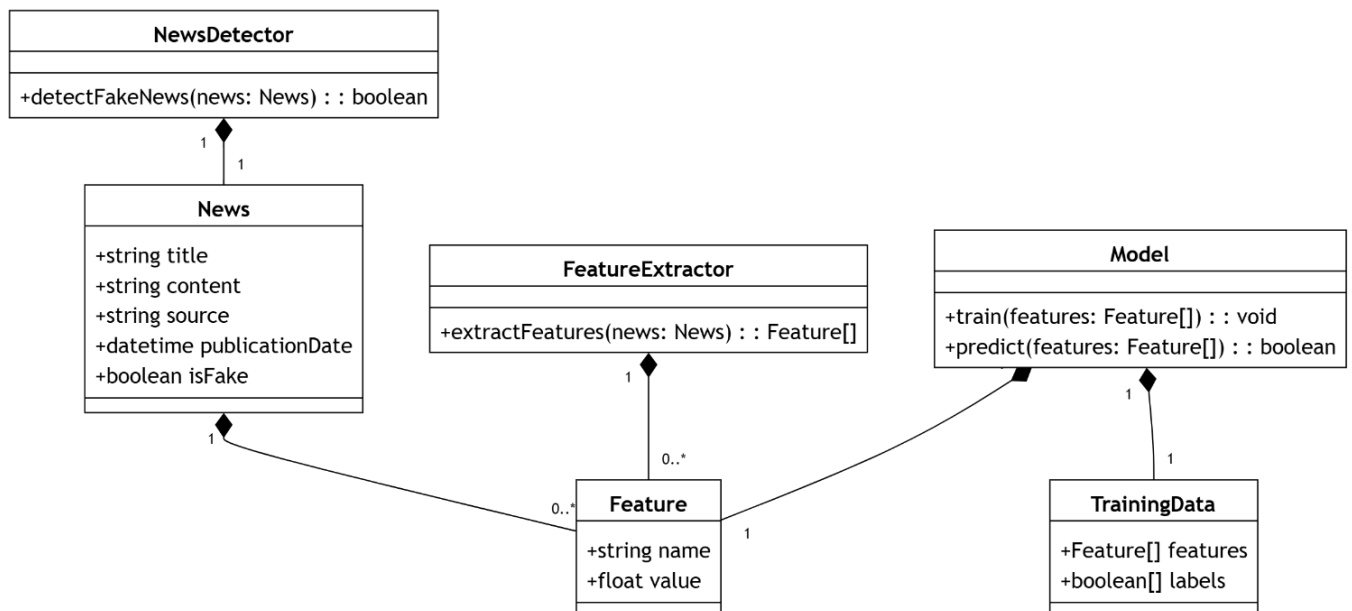


Рисунок 3.1 – UML-діаграма класів

Клас `NewsDetector` є центральним компонентом системи, який відповідає за виявлення фейкових новин. Основним методом цього класу є `detectFakeNews`, який приймає об'єкт класу `News` і повертає булеве значення, що вказує, чи є новина фейковою. Клас `NewsDetector` взаємодіє з іншими класами для виконання своєї основної функції.

Клас `News` представляє новинну статтю і містить такі атрибути: `title` (заголовок), `content` (зміст), `source` (джерело), `publicationDate` (дата публікації) та `isFake` (позначка, чи є новина фейковою). Цей клас забезпечує структуру для зберігання та обробки інформації про новини.

Клас `FeatureExtractor` відповідає за витягування ознак із новинного тексту. Метод `extractFeatures` приймає об'єкт класу `News` і повертає масив об'єктів класу `Feature`. Клас `FeatureExtractor` аналізує текст новини і виділяє значущі ознаки, які будуть використані для класифікації.

Клас `Feature` представляє окрему ознаку, яка використовується для аналізу новин. Він містить два атрибути: `name` (назва ознаки) та `value` (значення ознаки). Ці ознаки є ключовими елементами для визначення ймовірності того, що новина є фейковою.

Клас `Model` реалізує модель машинного навчання, побудовану за допомогою `TensorFlow`. Він містить два основні методи: `train` та `predict`.

Метод `train` приймає масив об'єктів класу `Feature` і навчає модель на основі переданих ознак та міток. Метод `predict` приймає масив ознак і повертає булеве значення, яке вказує на ймовірність того, що новина є фейковою.

Клас `TrainingData` зберігає дані для навчання моделі. Він містить два атрибути: `features` (масив об'єктів класу `Feature`) та `labels` (масив булевих значень, що вказують на те, чи є новина фейковою). Цей клас забезпечує необхідні дані для тренування та валідації моделі.

Таким чином, наведена UML-діаграма класів демонструє взаємодію між компонентами системи, що забезпечує процес виявлення фейкових новин. Кожен клас виконує свою специфічну функцію, сприяючи ефективному та точному аналізу текстових даних та визначенню їх достовірності.

Для реалізації модуля була обрана мова Python, оскільки є найкращою мовою для роботи з нейронними мережами та текстами, оскільки містить безліч бібліотек для обробки природної мови, текстових документів, вилучення вмісту веб-сторінок та забезпечує всі необхідні засоби для створення та навчання штучних мереж.

Python є об'єктно-орієнтованою мовою, що пропонує динамічну типізацію, автоматичне керування пам'яттю, зручні високорівневі структури даних, такі як списки, кортежі та словники, а також підтримує класи, модулі, пакети, обробку винятків та багатопотокові обчислення.

Код на Python може виконуватись у будь-якій операційній системі, де встановлений інтерпретатор Python.

Крім того, код на Python часто набагато компактніший порівняно з аналогічними програмами, написаними на Java.

Сам програмний модуль складається з наступних компонентів:

- модуль обробки набору даних;
- модуль попередньої обробки даних;
- модуль побудови нейронної мережі;
- модуль навчання нейронної мережі;
- модуль виявлення фейкових новин.

Тепер детальніше розглянемо певні модулі для списку для їх глибшого розуміння.

Модуль обробки набору даних відіграє важливу роль у забезпеченні системи необхідною інформацією для аналізу та подальшого навчання моделі. Його основні функції включають очищення, нормалізацію та попередню обробку текстових даних (рисунки 3.2).

Спершу текст новин очищується від зайвих пробілів, спецсимволів та приводиться до нижнього регістру, що забезпечує уніфікацію даних.

Наступним кроком є токенизація тексту, тобто розділення його на окремі слова, які далі лематизуються — зводяться до основної форми. Заключний етап обробки включає перетворення слів на числові вектори за допомогою методів

кодування тексту. Це робить текст готовим для подальшого аналізу нейронною мережею.

```

– Kaydee King (@KaydeeKing) November 9, 2016 The lesson from tonight's Dem
losses: Time for Democrats to start listening to the voters. Stop running the
same establishment candidates. – People For Bernie (@People4Bernie) November 9,
2016 If Dems didn't want a tight race they shouldn't have worked against Bernie.
– Walker Bragman (@WalkerBragman) November 9, 2016 New York Times columnist
Paul Krugman, who was one of Hillary Clinton's most outspoken surrogates during
the contentious Democratic primary, blamed Clinton's poor performance on Green
Party candidate Jill Stein, who has so far received a negligible number of votes
nationally, saying Stein was the Ralph Nader of 2016 in preventing a Clinton
victory. The account @BerniesTeachers threw Krugman's analysis back in his face.
Your candidate was the issue. Take responsibility. https://t.co/KHYOuUSrFS
– Teachers for Bernie (@BerniesTeachers) November 9, 2016 Ana Navarro, a
Republican who recently endorsed Hillary Clinton, summed up the preposterous
nature of the 2016 presidential election in this tweet: GOP nominated the only
damn candidate who could lose to Hillary Clinton. Democrats nominated the only
damn candidate who could lose to Trump – Ana Navarro (@ananavarro) November 9,
2016 Popular left-wing Facebook page The Other 98%, which was pro-Sanders
during the primary, responded to Trump's surge by simply posting a meme of
Sanders' face with the text "All this could've been avoided. Thanks for nothing,
DNC!" The meme has been shared almost 15,000 times in less than an hour: Posted
by The Other 98% on Tuesday, November 8, 2016 While Bernie Sanders endorsed
Hillary Clinton just before the Democratic National Convention in July, many of
his supporters remained adamant in their refusal to support the DNC-anointed
candidate, pointing to Wikileaks' revelations that top officials at the DNC had
been working behind the scenes to tip the scales in Clinton's favor by
...
remain too close to call. Clinton has 197 electoral votes to Trump's 187. Zach
Cartwright is an activist and author from Richmond, Virginia. He enjoys writing
about politics, government, and the media. Send him an email at [email
protected]
Output is truncated. View as a scrollable element or open in a text editor. Adjust cell output settings...

```

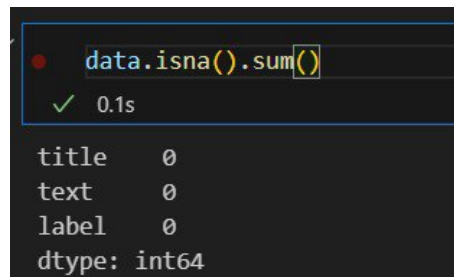
Рисунок 3.2 – Приклад новини з датасету

Ці операції будуть виконуватися механізмом попередньої обробки. При розробці модуля буде використовуватися `fake_news_dataset` (рисунок 3.3), який містить текст новини та відповідну мітку (FAKE або REAL).

Unnamed: 0		title	text	label
0	8476	You Can Smell Hillary's Fear	Daniel Greenfield, a Shillman Journalism Fello...	FAKE
1	10294	Watch The Exact Moment Paul Ryan Committed Pol...	Google Pinterest Digg Linkedin Reddit Stumbleu...	FAKE
2	3608	Kerry to go to Paris in gesture of sympathy	U.S. Secretary of State John F. Kerry said Mon...	REAL
3	10142	Bernie supporters on Twitter erupt in anger ag...	— Kaydee King (@KaydeeKing) November 9, 2016 T...	FAKE
4	875	The Battle of New York: Why This Primary Matters	It's primary day in New York and front-runners...	REAL

Рисунок 3.3 – БД фейкових новин

Також проведено перевірку на наявність пустих клітинок, які погано впливали б на модель при навчанні, що відображено на рисунку 3.4, де «title» - це заголовок новини, «text» - сам основний текст, а «label» - маркер, справжні (REAL) чи фейкові (FAKE) данні.

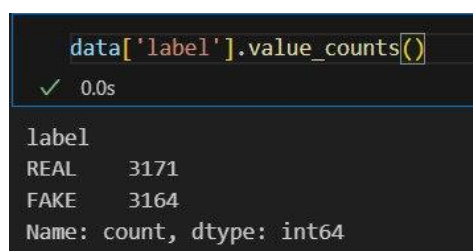


```
data.isna().sum()
✓ 0.1s
title    0
text     0
label    0
dtype: int64
```

Рисунок 3.4 – Перевірка на кількість пустих клітинок в датасеті

Модуль підготовки даних відповідає за підготовку даних для навчання моделі, розділяючи їх на тренувальний та тестовий набори. Вхідні числові вектори слів перетворюються на послідовності фіксованої довжини, що є необхідним для роботи з LSTM-мережами (Long Short-Term Memory), які використовуються у модулі побудови нейронної мережі. Для тренувального набору створюються мітки: 1 відповідає правдивій новині, а 0 — фейковій. Це дозволяє моделі навчатися на класифікації новин, що в подальшому підвищує її точність у виявленні фейкових новин.

Для забезпечення якісного навчання кількість різних маркерів в датасеті має бути хоча б приблизно однакова (рисунки 3.5), щоб модель мала однакову кількість даних як фейкових, так і справжніх новин, для змоги якісно навчатися на збалансованому наборі даних і не бути схильною до переваги одного класу над іншим, що сприяє підвищенню точності та надійності класифікації.



```
data['label'].value_counts()
✓ 0.0s
label
REAL    3171
FAKE    3164
Name: count, dtype: int64
```

Рисунок 3.5 – Кількість даних фейкових та справжніх новин

Модуль побудови нейронної мережі реалізує архітектуру нейронної мережі за допомогою платформи TensorFlow. Тут використовується техніка вбудовування TensorFlow з шаром вбудовування Keras, де відображаються вихідні та вхідні дані в деякий набір реальних розмірностей. Нейронна мережа складається з кількох ключових компонентів. Вхідний шар приймає числові вектори слів як вхідні дані. Наступними є шари LSTM, які здатні обробляти довгі послідовності даних і враховувати контекст слів у реченні. Далі йдуть шари Dense, які зменшують розмірність даних і роблять прогнози[25]. Вихідний шар використовує функцію активації сигмоїди, щоб генерувати ймовірність того, що текст є фейковим. Будова моделі було подано на рисунку 3.6.

Model: "sequential_1"

Layer (type)	Output Shape	Param #
embedding_1 (Embedding)	(None, 54, 50)	377600
dropout_1 (Dropout)	(None, 54, 50)	0
conv1d_1 (Conv1D)	(None, 50, 64)	16064
max_pooling1d_1 (MaxPooling 1D)	(None, 12, 64)	0
lstm_1 (LSTM)	(None, 64)	33024
dense_1 (Dense)	(None, 1)	65

Total params: 426,753
 Trainable params: 49,153
 Non-trainable params: 377,600

Рисунок 3.6 – Будова моделі

Відносно переваг використання вбудовування TensorFlow з шаром вбудовування Keras, можна віднести:

Простота використання. Використання вбудовування TensorFlow з шаром вбудовування Keras дозволяє легко створювати та навчати моделі для обробки текстових даних.

Ефективність. Вбудовувальні шари Keras оптимізовані для ефективної роботи з векторними представленнями слів, що полегшує роботу з текстовими даними.

Гнучкість. Вбудовувальні шари можуть бути налаштовані залежно від вимог конкретної задачі та можуть бути інтегровані в різні типи моделей.

Інформаційні зв'язки та взаємодія. Ефективна робота системи виявлення фейкових новин залежить від чіткої взаємодії між її модулями. Модуль збору та обробки даних надає модулю підготовки даних очищений та нормалізований текст у вигляді числових векторів. Модуль підготовки даних розділяє ці дані на тренувальний та тестовий набори, створюючи мітки для тренувального набору. Ці підготовлені дані передаються модулю побудови нейронної мережі, який створює та навчає модель на основі отриманих даних. Навчена модель передається модулю виявлення фейкових новин, який використовує її для аналізу нових текстів та визначення їх автентичності. Завдяки такій чіткій взаємодії модулів система забезпечує високий рівень точності у виявленні фейкових новин.

3.2 Тренування моделі

Тренування моделі є важливим етапом у розробці системи виявлення фейкових новин, оскільки від нього залежить здатність моделі точно класифікувати новини як правдиві чи фейкові. Модуль навчання відіграє ключову роль у цьому процесі, забезпечуючи оптимізацію моделі та її підготовку до реальної роботи.

Модуль навчання нейронної мережі відповідає за оптимізацію моделі, використовуючи алгоритм Adam (Adaptive Moment Estimation), який є одним з найпопулярніших алгоритмів градієнтного спуску для навчання нейронних мереж. Алгоритм Adam поєднує переваги двох інших методів - AdaGrad та RMSProp - забезпечуючи ефективне та швидке зниження функції втрат. Функція втрат, яку зазвичай використовують у задачах бінарної класифікації, як у нашому випадку, є бінарною перехресною ентропією. Ця функція вимірює різницю між передбаченими ймовірностями та фактичними мітками класів, прагнучи мінімізувати цю різницю.

Процес тренування моделі починається з розділення даних на тренувальний та тестовий набори. Тренувальний набір даних використовується для безпосереднього навчання моделі, тоді як тестовий набір застосовується для оцінки її продуктивності на нових, невідомих даних. Під час тренування модель навчається на тренувальному наборі даних протягом визначеної кількості епох. Епоха - це один повний цикл проходження через весь тренувальний набір даних.

Після кожної епохи модуль навчання оцінює точність моделі на тестовому наборі даних (Рисунок 3.7). Це дозволяє відслідковувати прогрес навчання та визначати, наскільки добре модель узагальнює інформацію на нових даних. Оцінка моделі включає такі показники, як функція втрат (loss) та точність (accuracy) як на тренувальному, так і на тестовому наборах. Функція втрат відображає, наскільки добре модель передбачає мітки новин, тоді як точність вимірює частку правильно класифікованих новин від загальної кількості новин.

```
Train on 2700 samples, validate on 300 samples
Epoch 1/50
2700/2700 - 1s - loss: 0.6423 - acc: 0.6181 - val_loss: 0.5621 - val_acc: 0.6833
Epoch 2/50
2700/2700 - 1s - loss: 0.5745 - acc: 0.7011 - val_loss: 0.5566 - val_acc: 0.6900
Epoch 3/50
2700/2700 - 1s - loss: 0.5303 - acc: 0.7337 - val_loss: 0.5272 - val_acc: 0.7300
Epoch 4/50
2700/2700 - 1s - loss: 0.4901 - acc: 0.7633 - val_loss: 0.5016 - val_acc: 0.7333
Epoch 5/50
2700/2700 - 1s - loss: 0.4423 - acc: 0.7963 - val_loss: 0.5064 - val_acc: 0.7367
Epoch 6/50
2700/2700 - 1s - loss: 0.4158 - acc: 0.8096 - val_loss: 0.4941 - val_acc: 0.7633
Epoch 7/50
2700/2700 - 1s - loss: 0.3726 - acc: 0.8352 - val_loss: 0.5763 - val_acc: 0.7500
Epoch 8/50
2700/2700 - 1s - loss: 0.3381 - acc: 0.8489 - val_loss: 0.5272 - val_acc: 0.7300
Epoch 9/50
2700/2700 - 1s - loss: 0.2979 - acc: 0.8767 - val_loss: 0.6131 - val_acc: 0.7400
Epoch 10/50
2700/2700 - 1s - loss: 0.2657 - acc: 0.8919 - val_loss: 0.5966 - val_acc: 0.7367
Epoch 11/50
2700/2700 - 1s - loss: 0.2689 - acc: 0.8878 - val_loss: 0.5400 - val_acc: 0.7700
Epoch 12/50
2700/2700 - 1s - loss: 0.2132 - acc: 0.9122 - val_loss: 0.5671 - val_acc: 0.7400
...
Epoch 49/50
2700/2700 - 2s - loss: 0.0523 - acc: 0.9781 - val_loss: 1.0049 - val_acc: 0.7433
Epoch 50/50
2700/2700 - 2s - loss: 0.0844 - acc: 0.9685 - val_loss: 0.8575 - val_acc: 0.7200
```

Рисунок 3.7 – Результати тренування нейронної мережі

Оцінювання моделі після кожної епохи також дозволяє вчасно коригувати гіперпараметри моделі. Гіперпараметри - це параметри, які налаштовуються до процесу навчання, такі як швидкість навчання, кількість нейронів у кожному шарі, кількість шарів у нейронній мережі, розмір міні-батчів та кількість епох. Оптимізація гіперпараметрів є важливою для досягнення високої точності моделі та її здатності до узагальнення.

Таким чином, модуль навчання виконує ключову функцію в процесі розробки моделі виявлення фейкових новин, забезпечуючи її ефективне та точне навчання. Після завершення процесу тренування, модель готова до використання в реальних умовах для класифікації нових новин як правдивих чи фейкових.

3.3 Тестування розробленої програми

Тестування є критично важливим етапом розробки програмного забезпечення, оскільки дозволяє оцінити продуктивність і надійність створеної системи. У цьому розділі подаються детальні процедури тестування розробленої програми для виявлення фейкових новин, а також аналізуються отримані результати.

Модуль виявлення фейкових новин використовує навчену нейронну мережу для аналізу нових текстів і визначення їх автентичності. Новий текст перетворюється на числові вектори слів та послідовності фіксованої довжини. Потім ці послідовності подаються на вхід нейронної мережі, яка генерує ймовірність того, що текст є фейковим. Якщо ймовірність перевищує визначений поріг, текст класифікується як фейковий.

Для перевірки правильності роботи модуля було проведено тестування на двох новинах: одна з них була фейковою, а друга — достовірною (Рисунок 3.8)

Результати тестування показали, що модуль успішно розпізнав фейкову новину, класифікувавши її як таку, що перевищує встановлений поріг ймовірності. Точність і надійність модуля підтверджуються цими тестовими прикладами, що свідчить про його ефективність у визначенні фейкових новин.

```

# sample text to check if fake or not
X = "Scientist: Elephant is the best home pet"

# detection
sequences = tokenizer1.texts_to_sequences([X])[0]
sequences = pad_sequences([sequences], maxlen=54,
                          padding=padding_type,
                          truncating=trunc_type)
if(model.predict(sequences, verbose=0)[0][0] >= 0.5):
    print("This news is True")
else:
    print("This news is false")

✓ 0.0s
This news is false

```

a)

```

# sample text to check if fake or not
X = "Karry to go to France in gesture of sympathy"

# detection
sequences = tokenizer1.texts_to_sequences([X])[0]
sequences = pad_sequences([sequences], maxlen=54,
                          padding=padding_type,
                          truncating=trunc_type)
if(model.predict(sequences, verbose=0)[0][0] >= 0.5):
    print("This news is True")
else:
    print("This news is false")

✓ 0.0s
This news is True

```

б)

Рисунок 3.8 – Перевірка а) фейкової та б) достовірної новин

ВИСНОВКИ

Розробка програмного модуля для виявлення фейкових новин має велике значення в сучасному світі, у боротьбі з поширенням дезінформації в сучасному інформаційному просторі. У світі, де доступ до інформації легко перетворюється на потік неперевіреної та маніпульованої інформації, виявлення фейкових новин стає надзвичайно важливим завданням для збереження довіри до ЗМІ та інформаційної безпеки.

Використання бібліотеки TensorFlow дозволило розробити потужні алгоритми машинного навчання, які здатні виявляти фейкові новини на основі аналізу текстових даних. Ця технологія надає можливість створити модель, яка виявлятиме навіть найбільш упереджені та хитрі форми дезінформації.

Успішна реалізація проекту передбачила комплексний підхід, який включає в себе збір та аналіз великого обсягу даних, тренування нейронної мережі на цих даних, імплементацію алгоритмів виявлення фейкових новин та постійне вдосконалення моделі на основі нових даних та використання оновлених методів машинного навчання.

Важливим етапом розробки програмного модуля є тестування та валідація його ефективності. Після впровадження програмного забезпечення його необхідно ретельно перевірити на реальних даних, щоб переконатися в його надійності та точності виявлення фейкових новин.

Використання вбудовування TensorFlow з шаром вбудовування Keras є потужним інструментом для обробки текстових даних у задачах машинного навчання. Ця техніка дозволяє легко створити та навчити модель для виявлення фейкових новин з використанням нейронних мереж.

Основними користувачами такого програмного модуля можуть бути редакції ЗМІ, медіаорганізації, дослідники, аналітики, а також широка громадськість. Виявлення фейкових новин стане ефективним інструментом для підвищення рівня довіри до інформації та попередження поширення маніпуляційних матеріалів.

Програмний модуль для виявлення фейкових новин є актуальним та перспективним напрямком розвитку в сфері інформаційної безпеки та машинного

навчання. Його впровадження сприятиме покращенню якості інформаційного простору та забезпеченню об'єктивності та достовірності інформації, що доходить до громадськості.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Quandt, T., Frischlich, L., Boberg, S. & Schatto-Eckrodt, T. Fake News, 1–6 (American Cancer Society, 2019). <https://onlinelibrary.wiley.com/doi/pdf/10.1002/9781118841570.iejs0128>. / (дата звернення 20.02.2024)
2. Lazer, D. M. J. et al. The science of fake news. *Science* 359, 1094–1096. <https://doi.org/10.1126/science.aao2998> (дата звернення 20.02.2024).
3. Tandoc, E. C., Lim, Z. W. & Ling, R. Defining fake news. *Digital J.* 6, 137–153. <https://doi.org/10.1080/21670811.2017.1360143> (дата звернення 20.02.2024).
4. Allcott, H. & Gentzkow, M. Social media and fake news in the 2016 election. *J. Econ. Perspect.* 31, 211–36. <https://doi.org/10.1257/jep.31.2.211> (дата звернення 20.02.2024).
5. Shu, K., Sliva, A., Wang, S., Tang, J., & Liu, H. (2017). Fake News Detection on Social Media: A Data Mining Perspective. *ACM SIGKDD Explorations Newsletter*, 19(1), 22-36.
6. Castillo, C., Mendoza, M., & Poblete, B. (2011). Information credibility on Twitter. *Proceedings of the 20th International Conference on World Wide Web*, 675-684.
7. Popat, K., Mukherjee, S., Yates, A., & Weikum, G. (2017). Where the Truth Lies: Explaining the Credibility of Emerging Claims on the Web and Social Media. *Proceedings of the 26th International Conference on World Wide Web*, 1003-1012.
8. Jin, Z., Cao, J., Guo, H., Zhang, Y., & Luo, J. (2013). Detection and Analysis of 2016 Election Hoaxes and Fake News: A Case Study. *Proceedings of the 2013 International Conference on Social Computing*, 2013, 173-179.
9. Wu, K., Yang, S., & Zhu, K. Q. (2015). False rumors detection on Sina Weibo by propagation structures. *Proceedings of the 2015 IEEE 31st International Conference on Data Engineering*, 651-662.
10. Sampson, J., Morstatter, F., Wu, L., & Liu, H. (2016). Leveraging the Implicit Structure within Social Media for Emergent Rumor Detection. *Proceedings of the 25th International Conference on World Wide Web*, 2016, 695-705.

11. Sun, L., Luo, Y., Chen, C., & Wang, C. (2013). A novel hybrid approach to detect fake reviews. Proceedings of the 22nd ACM International Conference on Information & Knowledge Management, 1601-1604.
12. Ma, J., Gao, W., Mitra, P., Kwon, S., Jansen, B. J., Wong, K. F., & Cha, M. (2015). Detecting Rumors from Microblogs with Recurrent Neural Networks. Proceedings of the 25th International Joint Conference on Artificial Intelligence, 3818-3824.
13. Ma, J., Gao, W., & Wong, K.-F. (2016). Detect Rumors in Microblog Posts Using Propagation Structure via Kernel Learning. Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL 2016), 708-717.
14. Yu, F., Liu, Q., Wu, S., Wang, L., & Tan, T. (2017). A Convolutional Approach for Misinformation Identification. Proceedings of the 26th International Joint Conference on Artificial Intelligence (IJCAI 2017), 3901-3907
15. Bian, T., Xiao, X., Xu, T., Zhao, P., Huang, W., & Rong, Y. (2020). Rumor Detection on Social Media with Bi-Directional Graph Convolutional Networks. Proceedings of the AAAI Conference on Artificial Intelligence, 34(01), 549-556.
16. Khattar, D., Goud, J. S., Gupta, M., & Varma, V. (2019). MVAE: Multimodal Variational Autoencoder for Fake News Detection. The World Wide Web Conference (WWW 2019), 2915-2921. <https://sci-hub.se/10.1145/3308558.3313552> (дата звернення 18.03.2024)
17. Про нас | Боротьба з неправдивою інформацією про події в Україні. URL: <https://www.stopfake.org/uk/pro-nas> / (дата звернення 20.02.2024).
18. Про нас | VoxUkraine. URL: <https://voxukraine.org/about-us> / (дата звернення 20.02.2024).
19. Snopes is the internet's definitive fact-checking resource. URL: <https://www.snopes.com/about-snopes> / (дата звернення 20.02.2024)
20. Our Mission - FactCheck.org. URL: <https://www.factcheck.org/about/our-mission> / (дата звернення 21.02.2024).

21. Twitter acquires Fabula AI, a machine learning startup that helps spot fake news VentureBeat. URL: <https://venturebeat.com/2019/06/03/twitter-acquires-fabula-ai-a-machine-learning-startup-that-helps-spot-fake-news> / (дата звернення: 25.02.2024).

22. Natasha Lomas, Fabula AI is using social spread to spot 'fake news' URL: https://techcrunch.com/2019/02/06/fabula-ai-is-using-social-spread-to-spot-fake-news/?guccounter=1&guce_referrer=aHR0cHM6Ly93d3cuZ29vZ2xlLnNvbS8&guce_referrer_sig=AQAAAIbUUE0epM41zfGbApnwqkQOzvYfU_Bhe_SNno6zjCjH8OX2J83jqeqSUKXDrleJgN0RnoaG9AqEyMXoG2VEm6TxPpcF9MxASWZHOA3KQY51GhIGmUV-RrDZB26fPBCgIEdNXPAV5-ErhhCvEhFEe3hxfadBUAKywmhvUIri7ur / (дата звернення: 25.02.2024).

23. Introducing Fakebox: Detect fake news with Machine Learning, URL: <https://medium.com/@machineboxio/introducing-fakebox-detect-fake-news-with-machine-learning-f602c39aad04> (дата звернення 26.02.2024).

24. Fakebox · Docs · Machine Box · Machine learning in a box. URL: <https://machinebox.io/docs/fakebox> / (дата звернення 27.02.2024).

25. Fake News Detection: A Deep Learning Approach. URL: <https://scholar.smu.edu/cgi/viewcontent.cgi?article=1036&context=datasciencereview> (дата звернення 15.05.2024).

26. Комар М.П., Саченко А.О., Васильків Н.М., Гладій Г.М., Коваль В.С., Ліп'яніна-Гончаренко Х.В. Методичні рекомендації до виконання кваліфікаційної роботи з освітньо-професійної програми «Комп'ютерні науки» спеціальності 122 «Комп'ютерні науки» за першим (бакалаврським) рівнем вищої освіти. Тернопіль: ЗУНУ, 2024. 52 с.

27. Андрійчук Я.С. Програмний модуль для виявлення фейкових новин із використанням TensorFlow. *Перспективні напрямки розвитку економіки, обліку, управління та права: теорія і практика*: тези доп. Міжнарод. науково-практичної конф. (Ізмаїл, 18 травня 2024 р.). Ізмаїл: ЦФЕНД, 2024. С. 24-25. [Додаток А]

Додаток А
Копія публікації



ЦЕНТР
ФІНАНСОВО-
ЕКОНОМІЧНИХ
НАУКОВИХ
ДОСЛІДЖЕНЬ

МІЖНАРОДНА НАУКОВО-ПРАКТИЧНА КОНФЕРЕНЦІЯ
INTERNATIONAL SCIENTIFIC-PRACTICAL CONFERENCE

ПЕРСПЕКТИВНІ НАПРЯМКИ РОЗВИТКУ ЕКОНОМІКИ,
ОБЛІКУ, УПРАВЛІННЯ ТА ПРАВА: ТЕОРІЯ І ПРАКТИКА

PERSPECTIVE DIRECTIONS FOR THE DEVELOPMENT OF ECONOMICS,
ACCOUNTING, MANAGEMENT AND LAW: THEORY AND PRACTICE

Збірник тез доповідей
Book of abstracts



18 травня 2024 р.
May 18, 2024

м. Ізмаїл, Україна
Izmail, Ukraine



СЕКЦІЯ 6. МАТЕМАТИЧНІ МЕТОДИ, МОДЕЛІ ТА ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ В ЕКОНОМІЦІ	
SECTION 6. MATHEMATICAL METHODS, MODELS, AND INFORMATIONAL TECHNOLOGIES IN ECONOMICS	24
<i>Андрійчук Я. С.</i> ПРОГРАМНИЙ МОДУЛЬ ДЛЯ ВИЯВЛЕННЯ ФЕЙКОВИХ НОВИН ІЗ ВИКОРИСТАННЯМ TENSORFLOW	24
<i>Гордовський О. А.</i> ОГЛЯД ВЕБ-БАЗОВАНИХ СИСТЕМ УПРАВЛІННЯ ОСОБИСТИМ БЮДЖЕТОМ	25
<i>Гурняк В. І., Гриньків А. М.</i> ПРОГРАМНА СИСТЕМА ВИЯВЛЕННЯ ЗЛОВЖИВАНЬ ПІД ЧАС ОНЛАЙН-ТЕСТУВАНЬ ЗАСОБАМИ МАШИННОГО НАВЧАННЯ	27
<i>Завойовська О. М.</i> ПРОГРАМНИЙ МОДУЛЬ ДЛЯ ВИЯВЛЕННЯ ОБРАЗЛИВИХ СЛІВ ТА НЕНАВИСТІ В ТЕКСТОВИХ ПОВІДОМЛЕННЯХ	28
<i>Кучар О. М.</i> ПРОГРАМНИЙ МОДУЛЬ НАВЧАННЯ ГРИ В ПОКЕР	29
<i>Мельник В. А.</i> ВЕБ-ОРІЄНТОВАНА СИСТЕМА УПРАВЛІННЯ ОСОБИСТИМИ ФІНАНСАМИ	30
<i>Sidh H., Kovalchuk Y.</i> IOT WEATHER MONITORING SYSTEM WITH DATA PROCESSING AND VISUALIZATION	32
<i>Старих О. В.</i> ВЕБ-БАЗОВАНА СИСТЕМА РЕКОМЕНДАЦІЙ РОБОЧИХ ВАКАНСІЙ	33
<i>Штинда В. В., Вітенко В. В.</i> ВЕБ-БАЗОВАНА СИСТЕМА РОЗПІЗНАВАННЯ ЇЖИ ПО ФОТО ЗАСОБАМИ КОМП'ЮТЕРНОГО ЗОРУ	34
СЕКЦІЯ 7. МЕНЕДЖМЕНТ	36
SECTION 7. MANAGEMENT	36
<i>Камал С.</i> ЗОВНІШНЬОЕКОНОМІЧНІ РИЗИКИ В БАНКІВСЬКИХ УСТАНОВАХ	36
<i>Пронін О. С.</i> РОЗРОБКА СТРАТЕГІЇ МІЖНАРОДНОГО РОЗВИТКУ ПІДПРИЄМСТВА БАНКІВСЬКОЇ СФЕРИ	38
СЕКЦІЯ 8. ІСТОРІЯ ТА ТЕОРІЯ ДЕРЖАВИ ТА ПРАВА, ФІЛОСОФІЯ ПРАВА	
SECTION 8. HISTORY AND THEORY OF STATE AND LAW, PHILOSOPHY OF LAW	40
<i>Бедрій М. М.</i> ЛОГІЧНІ ПРИЙОМИ (МЕТОДИ) ДОСЛІДЖЕННЯ УКРАЇНСЬКОГО ЗВИЧАЄВОГО ПРАВА	40

СЕКЦІЯ 6

SECTION 6

МАТЕМАТИЧНІ МЕТОДИ, МОДЕЛІ ТА
ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ В ЕКОНОМІЦІ
MATHEMATICAL METHODS, MODELS, AND
INFORMATIONAL TECHNOLOGIES IN ECONOMICS

УДК 004.932

Андрійчук Я. С.

студент 4 курсу факультету
комп'ютерних інформаційних технологій,
Західноукраїнський національний університет

**ПРОГРАМНИЙ МОДУЛЬ ДЛЯ ВИЯВЛЕННЯ
ФЕЙКОВИХ НОВИН ІЗ ВИКОРИСТАННЯМ TENSORFLOW**

Розвиток технологій та соціальних мереж призвів до зростання обсягу інформації та швидкості її поширення. Це створює сприятливі умови для розповсюдження фейкових новин, що є загрозою для суспільства та демократичних процесів [1].

Метою даного проекту є розробка програмного модуля для аналізу новинних текстів та оцінки їх достовірності з використанням методів машинного навчання. Це дозволить автоматично виявляти фейкові новини на ранній стадії їх поширення, що сприятиме боротьбі з дезінформацією та збереженню довіри до інформаційного середовища. У цьому проекті під фейковими новинами мається на увазі цілком сфабрикована інформація, яка намагається здаватися легітимною та має дезінформаційний характер [2].

Використання TensorFlow надає потужні інструменти для розробки алгоритмів машинного навчання, здатних виявляти фейкові новини на основі аналізу текстових та візуальних даних. Особливо корисним є використання вбудованих шарів TensorFlow і Keras для обробки текстових даних у задачах машинного навчання. Ця техніка дозволяє легко створювати та навчати моделі для виявлення фейкових новин за допомогою нейронних мереж [3].

Для реалізації даного проекту потрібно здійснити збір та аналіз великого обсягу даних, тренування нейронної мережі на зібраних даних, імплементацію алгоритмів виявлення фейкових новин та постійне вдосконалення моделі на основі нових даних та оновлених методів машинного навчання.

Розроблений програмний модуль може використовуватися в різних сферах. У медіа для автоматизованого виявлення фейкових новин серед великого обсягу інформації. На платформах для спілкування для фільтрації та виявлення фейкових новин серед контенту. У фінансовому секторі для аналізу та оцінки новин, що впливають на фінансові ринки. У академічній сфері для підвищення

медіаграмотності серед широкої аудиторії. У корпораціях для моніторингу та захисту своєї репутації.

Розробка програмного модуля для виявлення фейкових новин є актуальним та перспективним напрямком для сфери інформаційної безпеки та машинного навчання. Його впровадження сприятиме покращенню якості інформаційного простору, забезпечуючи об'єктивність та достовірність інформації. Основними користувачами такого програмного модуля можуть бути редакції ЗМІ, медіаорганізації, дослідники, аналітики, а також широка громадськість. Виявлення фейкових новин стане ефективним інструментом для підвищення рівня довіри до інформації та попередження поширення маніпуляційних матеріалів.

Список літератури

1. Tandoc, E. C., Lim, Z. W. & Ling, R. Defining fake news. *Digital J.* 6, 137–153. <https://doi.org/10.1080/21670811.2017.1360143> (2018).
2. Lazer, D. M. J. et al. The science of fake news. *Science* 359, 1094–1096. <https://doi.org/10.1126/science.aao2998> (2018).
3. Ciampaglia, G. L. Fighting fake news: A role for computational social science in the fight against digital misinformation. *J. Comput. Soc. Sci.* 1, 147–153. <https://doi.org/10.1007/s42001-017-0005-6> (2018).

УДК 004.738.5

Гордовський О. А.

студент 4 курсу,

Західноукраїнський національний університет

ОГЛЯД ВЕБ-БАЗОВАНИХ СИСТЕМ УПРАВЛІННЯ ОСОБИСТИМ БЮДЖЕТОМ

Веб-базована система управління особистим бюджетом — це сучасний інструмент, що дозволяє користувачам ефективно контролювати свої фінанси через інтернет. Такі системи включають функції обліку доходів і витрат, аналізу фінансових даних, планування бюджету та прогнозування майбутніх витрат. У сучасному світі, де фінансова грамотність є важливим елементом успішного життя, такі інструменти стають надзвичайно актуальними. Веб-базовані системи допомагають людям легко відстежувати свої витрати, аналізувати фінансові звички та приймати обґрунтовані рішення щодо витрат та заощаджень [1].

Основні функції веб-базованих систем управління особистим бюджетом включають облік доходів та витрат, автоматизацію фінансових операцій, створення звітів та графіків, налаштування фінансових цілей та інтеграцію з

Додаток Б

Код програмного модуля

Імпорт бібліотек і наборів даних

```
import numpy as np
import pandas as pd
import json
import csv
import random

from tensorflow.keras.preprocessing.text import Tokenizer
from tensorflow.keras.preprocessing.sequence import pad_sequences
from tensorflow.keras.utils import to_categorical
from tensorflow.keras import regularizers

import pprint
import tensorflow.compat.v1 as tf
from tensorflow.python.framework import ops
from sklearn.model_selection import train_test_split
from sklearn import preprocessing
tf.disable_eager_execution()

# Reading the data
data = pd.read_csv("news.csv")
data.head()
```

Токенізація і її застосування

```
title = []
text = []
labels = []
for x in range(training_size):
    title.append(data['title'][x])
    text.append(data['text'][x])
    labels.append(data['label'][x])
tokenizer1 = Tokenizer()
tokenizer1.fit_on_texts(title)
word_index1 = tokenizer1.word_index
vocab_size1 = len(word_index1)
sequences1 = tokenizer1.texts_to_sequences(title)
padded1 = pad_sequences(
    sequences1, padding=padding_type, truncating=trunc_type)
split = int(test_portion * training_size)
training_sequences1 = padded1[split:training_size]
test_sequences1 = padded1[0:split]
test_labels = labels[0:split]
training_labels = labels[split:training_size]
```

Створення Word Embedding

```
embeddings_index = {}
with open('glove.6B.50d.txt') as f:
    for line in f:
        values = line.split()
        word = values[0]
        coefs = np.asarray(values[1:], dtype='float32')
        embeddings_index[word] = coefs

# Generating embeddings
embeddings_matrix = np.zeros((vocab_size+1, embedding_dim))
for word, i in word_index.items():
    embedding_vector = embeddings_index.get(word)
    if embedding_vector is not None:
        embeddings_matrix[i] = embedding_vector
```

Створення архітектури моделі

```
model = tf.keras.Sequential([
    tf.keras.layers.Embedding(vocab_size+1, embedding_dim,
                              input_length=max_length, weights=[
                                  embeddings_matrix],
                              trainable=False),
    tf.keras.layers.Dropout(0.2),
    tf.keras.layers.Conv1D(64, 5, activation='relu'),
    tf.keras.layers.MaxPooling1D(pool_size=4),
    tf.keras.layers.LSTM(64),
    tf.keras.layers.Dense(1, activation='sigmoid')
])
model.compile(loss='binary_crossentropy',
              optimizer='adam', metrics=['accuracy'])
model.summary()
```