Національний університет ”Львівська політехніка”
Інститут комп’ютерних наук та інформаційних технологій
Кафедра систем штучного інтелекту
Спеціальність 122 «Комп’ютерні науки»

ПОЯСНЮВАЛЬНА ЗАПИСКА

до магістерської кваліфікаційної роботи на тему:

«Дослідження ефективності методів класифікації та резюмування новин для полегшення орієнтації в інформаційному просторі»

Студента(ки) групи КНСШ-23 Гірника Ю.В.

Керівник роботи _____ (___Бойко Н.І.___)
(підпис)

Консультанти _____ (_____)
(підпис)

_____ (_____)
(підпис)

Завідувач
кафедри СШІ _____ (__Шаховська Н. Б.___)

Львів - 2023

Національний університет “Львівська політехніка”

Інститут комп’ютерних наук та інформаційних технологій

Кафедра систем штучного інтелекту

Спеціальність 122 «Комп’ютерні науки»

ЗАТВЕРДЖУЮ:

Завідувач кафедри СШІ _____

“____” _____2023р.

ЗАВДАННЯ
НА МАГІСТЕРСЬКУ КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТОВІ

Гірнику Юрію Володимировичу

1. Тема роботи: «Дослідження ефективності методів класифікації та резюмування новин для полегшення орієнтації в інформаційному просторі» затверджена наказом університету № 4420-4-06 від 13.10.2023

2. Термін подання закінченої роботи: _01.12.2023 _____

3. Вихідні дані до (проекту) роботи: програмний продукт для класифікації та резюмування новин, порівняльна характеристика.

4. Зміст розрахунково-пояснювальної записки: (перелік питань, що належить розробити): аналітичний огляд літературних та інших джерел, системний аналіз та обґрунтування проблеми, методи та засоби вирішення проблеми, практична реалізація.

5. Перелік графічного матеріалу (з точним зазначенням обов’язкових креслень): блок-схема Prisma

6. Консультанти по роботі, із зазначенням розділів проекту, що стосується їх

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв
Консультант з іноземної мови			

7. Дата видачі завдання _____10.03.2023_____

Керівник роботи _____
(підпис)

Завдання прийняв до виконання _____
(підпис)

КАЛЕНДАРНИЙ ПЛАН

№ п/п	Назва етапів дипломної роботи	Термін виконання етапів роботи	Примітка
1	Огляд літератури	10.03.2023 – 03.04.2023	
2	Створення архітектури	03.04.2023 – 15.05.2023	
3	Розробка алгоритмічного та програмного забезпечення	15.05.2023 – 11.09.2023	
4	Експериментальні дослідження	11.09.2023 – 06.11.2023	
5	Оформлення записки	06.11.2023 – 01.12.2023	

Керівник роботи _____
(підпис)

Студент _____
(підпис)

АНОТАЦІЯ

Магістерська кваліфікаційна робота виконана студентом групи КНСШ-23 Гірником Юрієм Володимировичем. Тема “Дослідження ефективності методів класифікації та резюмування новин для полегшення орієнтації в інформаційному просторі”. Робота направлена на здобуття ступеня магістр за спеціальністю 122 «Комп’ютерні науки».

Об’єктом дослідження є процес споживання новин, його особливості і способи покращення.

Предметом дослідження є методи класифікації та резюмування новин на основі текстових даних.

Мета магістерської кваліфікаційної роботи полягає в дослідженні та можливості покращення ефективності методів класифікації та резюмування новин. Досягнення мети досягається шляхом дослідження сучасних методів вирішення завдань обробки природньої мови, особливостей їхньої архітектури та підходів. Також внаслідок проведення серії дослідів над розширеним набором даних проведено аналіз отриманих результатів.

У результаті виконання дипломної роботи було отримано опис методів класифікації та резюмування текстів, побудовано набір даних, а також проведена серія дослідів над ним і обраними методами. Отримані результати були проаналізовані та порівняні, що дає змогу обрати метод для прикладного застосування в сфері новин в подальшому.

Загальний обсяг роботи: 69 сторінок, 16 рисунків, 39 посилань.

Ключові слова: класифікація новин, резюмування новин, обробка природньої мови, машинне навчання.

ABSTRACT

Master paper executed by the student of group CSAI-23 Hirnyk Yurii Volodymyrovych. The topic is "Investigation of the effectiveness of methods of classification and summarization of news to facilitate orientation in the information space". The work is aimed at obtaining a master's degree in the specialty 122 "Computer Science".

The object of the research is the process of news consumption, its features and methods of improvement.

The subject of the research is methods of classification and summarization of news based on textual data.

The purpose of the master's thesis is to investigate and improve the effectiveness of news classification and summarization methods. The goal is achieved by researching modern methods of solving natural language processing tasks, features of their architecture and approaches. Also, as a result of conducting a series of experiments on an expanded data set, an analysis of the obtained results was carried out.

As a result of completing the master’s qualification work, a description of the methods of classifying and summarizing texts was obtained, a data set was built, and a series of experiments was conducted on it and the selected methods. The obtained results were analyzed and compared, which makes it possible to choose a method for application in the field of news in the future.

Today, many events take place every day, many news are published every second in thousands of news resources. The amount of information that each of us consumes is growing exponentially. This can be especially felt in the last year, when many of us are chained to our phones and monitor Telegram channels to understand what is happening. It can be concluded that the news has a significant influence on our personal activities, mood and thoughts. Among such an array of unorganized content, the task of selecting relevant articles that may be of interest to you becomes a complex task.

Several challenges arise with such conditions. This is primarily the processing and organization of these large volumes of data. News is one of the most popular and widespread sources of information, but with it comes information noise. This makes it

difficult for people who are looking for brevity, or who are interested in a particular topic. Like, people who are only interested in sports, would not like to see news from the world of art.

It is difficult to overestimate the importance of news in today's world. This or that news can play a big role in various aspects of human life. First of all, news helps to navigate what is happening around. It can be news about a change in the exchange rate, the introduction of electronic referral to a doctor, a natural or man-made disaster nearby or about a small event in the city. Each of these examples affects life differently, but being informed about each of them, a person draws conclusions and certain actions. In some cases, it can save lives.

The sources of news are several main communication channels: TV, newspaper, online news resource, social networks, and recently also Telegram channels. It is also worth noting that the number of readers of paper newspapers has been decreasing in recent years. The conducted research shows that the number of readers in the USA decreased by 45%, namely from 45 to 25 million, during the years 2010-2020. This trend forces large newspaper companies to digitize and develop other channels of communication with users. In addition, the importance of quality and systematization of online articles is increasing. In a fast and massive stream of new publications, it is important to enable the reader to quickly understand what the topic of the publication is (news classification) and whether it is interesting for him (news summarization). In addition, a review of the literature on the given topic shows the growing interest of the scientific community (Fig. 1).

This topic of the diploma thesis seeks to investigate methods of classification and summarization of news, which can help users navigate in a rich information space, and make the process of reading news more efficient and pleasant. This is what justifies the choice of this topic.

The work considers the solution of two NLP problems: text classification and text summarization (building a title or a short description based on a large text).

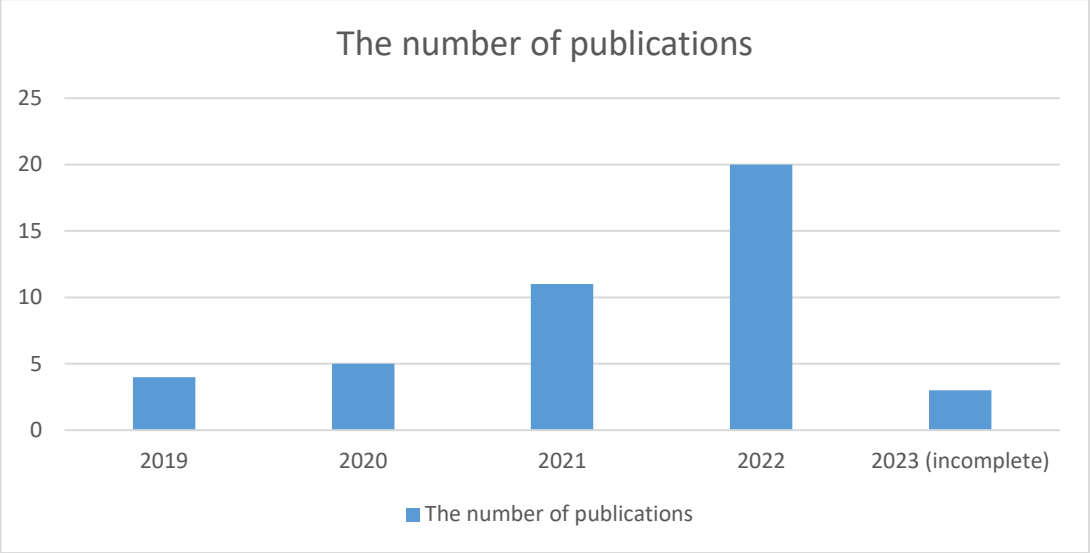

Fig. 1. The number of publications on the selected topic over the past 5 years, published in the Scopus scientometric database

Text classification is the process of determining the text category based on its content and content. It has become widespread in many areas and solves a number of applied problems (spam detection, text language detection, article tagging, etc.).

There are several approaches to building systems to solve the problem of text classification

- Rule-based systems. In this case, the assignment of the text to a specific category occurs according to predetermined rules [19]. This approach is simple to implement, but can be quite inaccurate and expensive, because the construction of linguistic rules for dictionaries requires a deep understanding of a class or category. Among the advantages of the method is no need for training data and ease of understanding.
- Systems based on machine learning. This method is based on the use of machine learning methods and models. Instead of manually building the rules for assigning text to a category, developers shift the responsibility to algorithms that can find certain connections between text and class. This approach requires pre-processing of the text and representation of the text in an understandable representation for the algorithm - usually a vector of numbers.
- Hybrid systems combine the use of both previous approaches: rule-based and

machine learning. But it immediately becomes obvious that such a system is more difficult to maintain and develop.

The work focuses on the construction of classification and summarization systems based on transformers - models presented in 2017 that challenged the most popular recurrent neural networks at the time. At the heart of RNN were several concepts that were reused and improved upon in transformers. Among them, it is worth highlighting 2 main ones: the encoder-decoder model and the attention mechanism. Instead, he offered something new to the transformers – a self-attention mechanism.

Self-attention is the main innovation of the transformer architecture. The mechanism of attention is assigning each element a certain coefficient, a certain rating. Self-attention means that the coefficients will be calculated for each hidden state of the encoder or decoder. This allows the network to separately compute context from a large amount of information without intermediate recurrent connections, which means the calculations can be parallelized.

There are two types of summarization, which will actually be considered in this work: extractive summarization – this is the process of extracting already existing sentences from the text. That is, the algorithm must evaluate the sentences based on a certain metric, sort by this score, and return some number of the most important sentences. The algorithm cannot generate a new word or sentence that was not mentioned in the input text. Second is abstract summarization. This method is more difficult to implement and differs in the ability to generate an output based on the facts in the text and meaningful meaning.

The Python and a set of libraries (pandas, request, nltk, etc.) were chosen to solve the task, namely the classification and summarization of text news. Experiments were conducted in the Google Colab environment with a PRO subscription, which provides the opportunity to use an increased amount of RAM and more powerful processors. Google Colab is a web application, so the main software requirements are Internet access and a browser.

Also, a data set was built for the experiments, which is an extension of the existing data set (Fig. 2). A large number of records (210 thousand) allows for detailed research,

and the presence of a link to the original source provides the possibility of using data extraction tools (web mining). Thus, this paper will expand the existing data set by hand.

	link	headline	category	short_description	authors	date
0	https://www.huffpost.com/entry/covid-boosters-...	Over 4 Million Americans Roll Up Sleeves For O...	U.S. NEWS	Health experts said it is too early to predict...	Carla K. Johnson, AP	2022-09-23
1	https://www.huffpost.com/entry/american-airlin...	American Airlines Flyer Charged, Banned For Li...	U.S. NEWS	He was subdued by passengers and crew when he ...	Mary Papenfuss	2022-09-23
2	https://www.huffpost.com/entry/funniest-tweets...	23 Of The Funniest Tweets About Cats And Dogs ...	COMEDY	"Until you have a dog you don't understand wha...	Elyse Wanshel	2022-09-23
3	https://www.huffpost.com/entry/funniest-parent...	The Funniest Tweets From Parents This Week (Se...	PARENTING	"Accidentally put grown-up toothpaste on my to...	Caroline Bologna	2022-09-23
4	https://www.huffpost.com/entry/amy-cooper-lose...	Woman Who Called Cops On Black Bird-Watcher Lo...	U.S. NEWS	Amy Cooper accused investment firm Franklin Te...	Nina Golgowski	2022-09-22

Fig. 2. Entries in the News Category Dataset

The next stage after building the data set is data preprocessing. In the case of solving the classification problem, the text must go through a number of changes before it is ready for use (Fig. 3):

- Tokenization: the process of dividing the text into tokens (sentences, words). At the same time, one large text becomes a set of lexemes, each of which can carry a separate information value.
- Normalization: translation of word forms into one format. There are several factors that need to be taken into account: bringing words to the same register (lower), lemmatization - bringing different morphological forms of the word to the initial form.
- Removal of stop words: filtering of small parts of speech that are not significant for solving the problem.
- Removal of special punctuation marks: filtering commas, periods, colons, etc.

Models based on various transformers, namely DistilBert, RoBERTa, AlBERT, were used for classification experiments. All of them are a modification of the original Bert model presented in a scientific paper in 2018 [8]. At the time of its appearance, it exceeded the most popular and modern NLP models in solving several tasks of varying complexity.

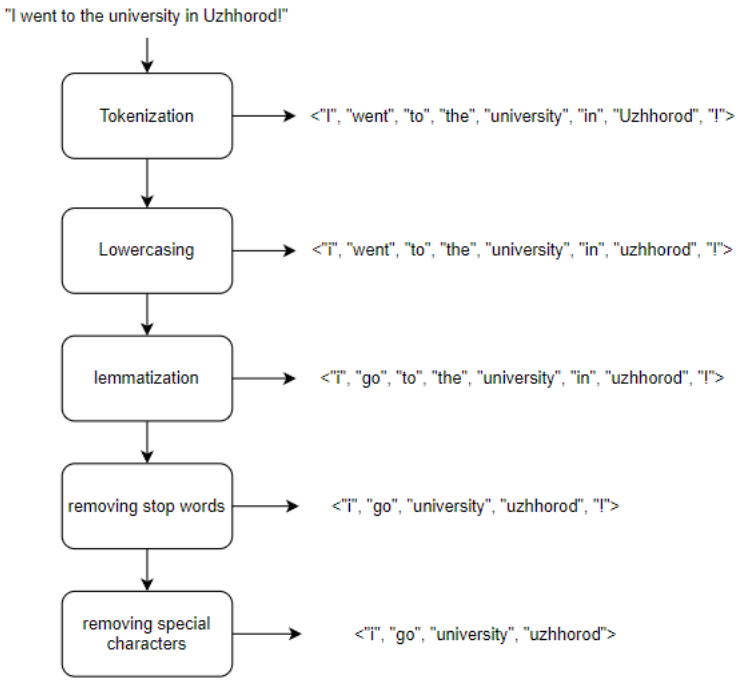

Fig. 3. Data preprocessing

The basis for the implementation of news classification is the construction and training of a neural network using the pre-trained transformers mentioned earlier. Instead, there will be several other layers in a deep neural network (Fig. 4):

- The first layer accepts as input an embedding of words with a length of 512, which are built by tokenizers that differ for each type of transformer. That is, the vector of text data is converted into the vector of numerical values needed by the transformer. The required result of the transformer is presented in the form of a CLS token, which represents the entire text received at the input and can be used for classification tasks.
- Dropout layer is a regularization method that helps to avoid overtraining by excluding neurons with some predetermined probability.
- A dense layer is a fully connected layer, that is, one that contains a certain number of neurons, each of which is connected to each neuron of the previous layer. The first layer is designed to reduce the dimensionality of the results obtained from the transformer, and can also serve as an additional regularization seed. The last layer is the output layer with 27 neurons and

softmax activation function. The number of neurons is determined by the number of classes, that is, categories of news, and the softmax function normalizes the received data, which makes it possible to represent it as probabilities of belonging to a certain class.

The Adam optimizer is used as an optimizer - a method of stochastic optimization based on the evaluation of first- and second-order derivatives.

The metric used to evaluate the multi-class classification is F1.

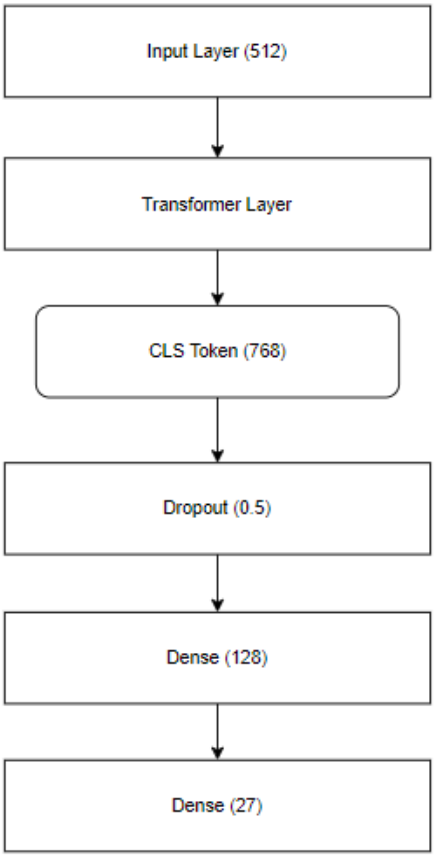

Fig. 4. The architecture of a neural network using a transformer

To implement summarization, two libraries were used - Summarizer and SimpleT5. The first provides access to a number of models (BERT, XLNet) for summarization based on pretrained transformers. Instead, SimpleT5 provides functionality to refine the T5 model based on input data.

The data obtained as a result of the classification experiments are organized in Table 1.

Table. 1. Results of training and testing classification models

Metric\Model	MultinomialNB	DistilBert	RoBERTa	Albert
Accuracy	0.617	0.711	0.694	0.700
Precision	0.699	0.637	0.610	0.635
Recall	0.400	0.617	0.596	0.596
F1 score	0.432	0.620	0.597	0.609
Training time, minutes	-	26,50	68,43	95,86
Testing time, minutes	-	43,15	88,70	90,23

Based on the obtained results, it is possible to note the dominance of the DistilBert model over others. This applies to almost all of the above indicators. The proportion of correct predictions is slightly higher (0.711 versus 0.694 and 0.700), however, the proportion of true positive predictions among all positive results is somewhat low in each of the models. From this we can conclude that the models need further research in order to improve the indicators.

For the summarization task, models based on three transformers were investigated: BERT, XLNet, and T5.

The conducted studies with different models of summaries are also systematized in the form of Table 2. The type of summarization (generalization), time of training and generation of the answer, as well as a set of Rouge metrics for qualitative assessment of the obtained results were selected as the characteristics of the comparison.

Table 2. Results of training and testing summarization models

Metric/Model	Bert	XLNet	T5
Summarization	Extractive	Extractive	Abstractive
Training time, minutes	-	-	78
Generation time of one summation, seconds	1.51	1.43	0.73
Rouge1	0.187	0.285	0.310
Rouge2	0.066	0.273	0.182
RougeL	0.139	0.285	0.280
RougeLsum	0.140	0.285	0.279

Before analyzing specific numerical results, it is worth mentioning a key difference between Bert/XLNet and T5. The first two models use extractive generalization, that is, one that selects already existing fragments of text at the input. Instead, T5 is a representative of abstract generalization - the generation of new text that may not be a fragment of the input.

The response generation time is the lowest for T5, which makes it attractive for large and busy applications. Also, Rouge's metrics for evaluating the quality of summarization indicate T5 dominance, which I attribute to the use of abstract generalization.

A description of existing approaches and methods for solving the task of classifying and summarizing texts was given. Special attention is paid to modern architectures for solving NLP tasks - transformers. In recent years, they have gained immense popularity and not for nothing - their effectiveness and flexibility in solving various tasks are really not unreasonable. The specifics of the work of transformers, their innovative solutions and architecture were given.

In the course of the research, a model was built using various transformers to solve the classification problem, as well as to indicate the use of transformers for summarizing. The key objective was to investigate the difference in the performance of different architectures and the possibilities of application in the field of news. The models showed a fairly high accuracy, and such characteristics as training and work time allow you to additionally compare them.

The obtained results proved the possibility of using transformers to solve the classification problem and their dominance over primitive algorithms. Thus, the F1 accuracy metric for the Bayesian classifier was 0.432, whereas for the architecture using the DistilBERT transformer it was 0.620. It is worth noting that most of the transformers were used already trained, which made it possible to use a large amount of efforts already made by the scientific community.

The topic of the master's thesis is relevant, because every day people are influenced more and more by information, and it becomes more difficult to navigate in the ocean of information. The obtained results make it possible to verify the possibility of applying

machine learning methods, in particular transformers, in the field of news (news sites, Telegram channels, etc.).

In addition, there are many more methods of classification and summarization than those presented in the paper, so their research can continue, which allows to continue the research activities presented here. It is also planned to develop a software product - a web resource that will use transformers for auto-marking of news with characteristic topics, as well as the formation of summaries of news articles in order to form daily digests for users. These technologies also make it possible to develop a system of recommendations, because usually users are most interested in several areas – auto-labeling will help with this.

Keywords: news classification, news summarization, natural language processing, machine learning.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

NLP – Natural Language Processing.

Rouge – Recall-Oriented Understudy for Gisting Evaluation.

RNN – recurrent neural network.

GPT – Generative Pretrained Transformer.

BERT – Bidirectional Encoder Representations from Transformers.

MLM – Masked Language Modeling.

ВСТУП

Сьогодні кожного дня відбуваються безліч подій, безліч новин щосекунди публікується в тисячах новинних ресурсів. Обсяги інформації, яку кожен з нас споживає, зростають в геометричній прогресії. Особливо це можна відчути в останній рік, коли багато хто з нас прикутий до свого телефону та моніторить телеграм-канали аби розуміти що відбувається. Можна зробити висновок, що новини мають значний вплив на нашу особисту діяльність, настрій та думки. Серед такого масиву неорганізованого контенту завдання вибрати відповідні статті, які можуть зацікавити саме вас, стає складною комплексною задачею.

З такими умовами виникають декілька викликів. Це перш за все обробка та організації цих великих обсягів даних. Новини – одне з найпопулярніших та поширеніших джерел інформації, але разом з ними виникає інформаційний шум. Через це виникає складність для людей, які шукають лаконічність, або цікавляться окремою темою. Як-от, люди які захоплюються лише спортом, не хотіли б бачити новини з світу мистецтва.

Дана тема дипломної роботи (“Дослідження ефективності методів класифікації та резюмування новин для полегшення орієнтації в інформаційному просторі”) прагне дослідити методи класифікації та резюмування новин, що може допомогти користувачам орієнтуватись в багатому інформаційному просторі, а сам процес читання новин зробити ефективнішим та приємнішим. Саме це й аргументує вибір даної теми.

Метою роботи є полягає в дослідженні та можливості покращення ефективності методів класифікації та резюмування новин.

Для досягнення мети необхідно вирішити ряд задач:

- дослідження предметної області;
- аналіз існуючих підходів та алгоритмів;
- пошук або побудова набору даних;
- тестування та порівняння методів класифікації та резюмування;
- аналіз отриманих результатів;

Об’єктом дослідження є процес споживання новин, його особливості та способи покращення.

Предметом дослідження є методи класифікації та резюмування новин на основі текстових даних.

Методами дослідження є алгоритми машинного навчання, які використовуються для роботи з текстами.

Новизною дослідження є програма здатна виконувати автоматичну класифікацію та резюмування новини на основі вхідних даних.

Практична цінність полягає у тому, що кількість новин, що генеруються, постійно зростає, а їхня категоризація та резюмування може полегшити орієнтацію читача в інформаційному просторі.

Особистий внесок магістранта полягає в розробці програми для обробки новин. Обрані моделі класифікації та резюмування були досліджені на вказаних даних, тож всі результати отримані випускником особисто.

Апробація результатів роботи. На конференції “Інформаційні технології і автоматизація – 2023” була опублікована однойменна наукова робота за темою магістерської кваліфікаційної роботи [22].

1. АНАЛІТИЧНИЙ РОЗДІЛ

1.1. Опис предметного середовища.

Задача класифікації чи резюмування новин перш за все стосуються такого напрямку штучного інтелекту як обробка природньої мови (Natural language processing). NLP фокусуються на задача комп'ютерного сприйняття людської мови та може вирішувати безліч дотичних завдань, як-от:

- Класифікація тексту. Приклад застосування класифікації тексту за певними параметрами можна спостерігати майже кожного дня в щоденному житті. Наприклад, лист, що приходить на вашу електронну скриньку, може не з'явитись у папці вхідних повідомлень, оскільки поштовий сервіс визначить його як спам і помістить у відповідну папку. Також яскравим прикладом класифікації текстів є визначення тональності мови (рис. 1.1) у додатку Grammarly [35]. Як сама компанія зазначає, ваш тон напряму впливає на те, що ви хочете донести до співбесідника.

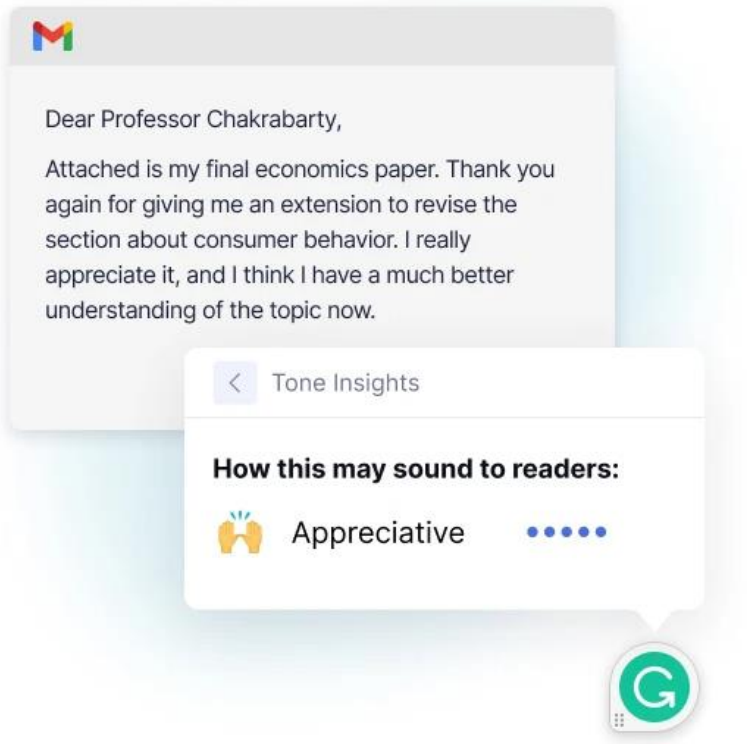

Рис. 1.1. Приклад класифікації тексту за тональністю

- Резюмування тексту – побудова заголовка або ж короткого опису на основі великого за обсягом тексту. Дана прикладна область може знайти себе в багатьох сферах життєдіяльності людини: побудова заголовків чи щоденних дайджестів новин, короткий опис викладеного матеріалу наукової статті, тощо.
- Машинний переклад – автоматичний переклад тексту з однієї мови на іншу. Яскравий приклад – застосунок Google Перекладач, який розвивається з часом та також вирішує іншу задачу обробки природньої мови, а саме – розпізнання та синтез мови. Тобто автоматичний перевід голосового вводу користувача в текст та навпаки.

Важко переоцінити важливість новин в сучасному світі. Та чи інша новина може відігравати велику роль в різних аспектах людського життя. Перш за все, новини допомагають орієнтуватись в тому, що відбувається навколо. Це може бути новина щодо зміни курсу валют, запровадження електронного направлення до лікаря, природньої чи техногенної катастрофи поруч чи про маленьку подію в місті. Кожен з цих прикладів по різному впливає на життя, але будучи проінформованим про кожну з них, людина робить висновки і певні дії. У деяких випадках це може зберегти життя.

Джерелами новин є декілька основних каналів зв’язку: телевізор, газета, новинний онлайн-ресурс, соціальні мережі, а віднедавна ще й телеграм-канали. Також варто зазначити, що протягом останніх років зменшується кількість читачів паперових газет [14]. Проведене дослідження показує, що кількість читачів в США зменшилась на 45%, а саме з 45 до 25 мільйонів, протягом 2010-2020 років. Така тенденція змушує великі газетні компанії оцифровуватись та розвивати інші канали зв’язку з користувачами. Крім того, збільшується важливість якості та систематизованості онлайн-статей. У швидкому та масовому потоці новий публікацій важливо дати змогу читачу швидко зрозуміти якої тематики публікація та чи є вона цікавою для нього.

1.2. Аналіз наявних аналогів.

Для пошуку наукових джерел я обрав наукометричні бази Scopus [34] та Google Scholar [28]. Вибір Scopus аргументований його авторитетністю та масштабністю. Крім того, статті проходять ретельне рецензування перед публікацією, що суттєво підвищує якість робіт. Натомість Google Scholar об’єднує та надає доступ до відкритий наукових джерел. Також ця система має інтуїтивно зрозумілий та простий інтерфейс, що полегшує користування.

Для початку пошуку потрібно сформувати відповідний пошуковий запит. Задля цього використаємо пошук за ключовими словами, які актуальні для обраної теми роботи. Також використаємо оператори OR та AND, задля комбінування різних ключових слів та їх варіацій. Початковий запит:

TITLE-ABS-KEY(("news summarization" or "news classification") and ("NLP" OR "Natural language processing" OR "Machine Learning" OR "Text Mining" OR "Information Retrieval")).

Результат цього запиту - 412 документів. Документи, відібрані цим запитом, стосуються або резюмування новин, або ж їх класифікації. Було зроблено таке рішення, оскільки рідко наукові роботи сконцентровані водночас на цих обох поняттях.

Однак, 412 документів все ще важко проаналізувати. Для відбору більш релевантних статей слід застосувати наступні критерії для фільтрування:

- Результат повинен бути саме статтею. Тобто потрібно відкинути роботи, як-от, доповіді з конференцій.
- Статті повинні бути опубліковані за останні 5 років (2019-2023).
- Мова написання – англійська.
- Сфера дослідження – комп’ютерні науки.
- Відкрий режим доступу.

Сформуємо оновлений пошуковий запит з урахуванням нових критеріїв: TITLE-ABS-KEY(("news summarization" or "news classification") and ("NLP" OR "Natural language processing" OR "Machine Learning" OR "Text Mining" OR "Information Retrieval")) AND PUBYEAR > 2018 AND (LIMIT-TO (OA,"all")) AND

(LIMIT-TO (DOCTYPE,"ar")) AND (LIMIT-TO (SUBJAREA,"COMP")) AND (LIMIT-TO (LANGUAGE,"English")).

Оновлений запит підвищив релевантність статей, та дозволив повернути 43 публікації в межах наукометричної бази Scopus. Також проведемо пошук в Google Scholar за подібним принципом: знайдемо статті за останні 5 років, що також містять ключові слова ("news summarization", "news classification"). Результат виклику пошукового запиту – 29 публікацій.

На основі отриманих результатів, проведено аналітичний огляд наукових джерел згідно з стандартизованою методологією PRISMA, що містить загальні рекомендації щодо огляду наукових та базових особливостей мета-аналізу.

При об’єднанні результатів з двох баз, було виявлено 1 дублікат, що звузило кількість результатів до 71 документа. На наступному етапі, шляхом аналізу заголовка/анотації, було виключено 41 статтю. На етапі відповідності було знайдено 3 статті, які були опубліковані раніше 2019 року, але були помилкового повернуті в результаті пошукового запиту. Також 9 статей не були включені в якісний аналіз після перегляду їх змісту. Таким чином кількість статей, над якими буде проведено критичний аналіз, скоротилась до 15. Кінцевий варіант схеми PRISMA відображений на рис. 1.2.

У [10] автори наводять короткий огляд існуючих моделей та підходів до класифікації тексту. У процесі вони висвітлюють не тільки методи, але й процес попередньої підготовки даних, надаючи йому таку ж важливість та актуальність, як і сам метод класифікації. Дана публікація наводить огляд як сучасних методів класифікації, так і більш уставлених в цій сфері. Ця робота корисна при виборі підходу до виконання дипломної роботи, але не надає достатньо глибокого аналізу якогось з перерахованих методів.

Дослідження [13] присвячено новій моделі класифікації текстів на основі багаторівневих семантичних ознак. Автори пропонують додати коефіцієнт кореляції категорій до TF-IDF, а також коефіцієнт концентрації частоти до CHI. Це відображає рівень ключових семантичних ознак. Інші 2 рівні це – локальні семантичні ознаки за допомогою TextCNN та глобальні семантичні ознаки за

допомогою BiLSTM. Об’єднання цих рівнів і складає ідею багаторівневих ознак, що є основною перевагою, адже дозволяє отримати більше інформації про текст. Даний метод був протестований на декількох відомих наборах даних та показав результати вище базової моделі для цих наборів даних. Однак, автори не показали використання цього підходу для проблеми, яку вони прагнули вирішити – класифікація новин китайською мовою.

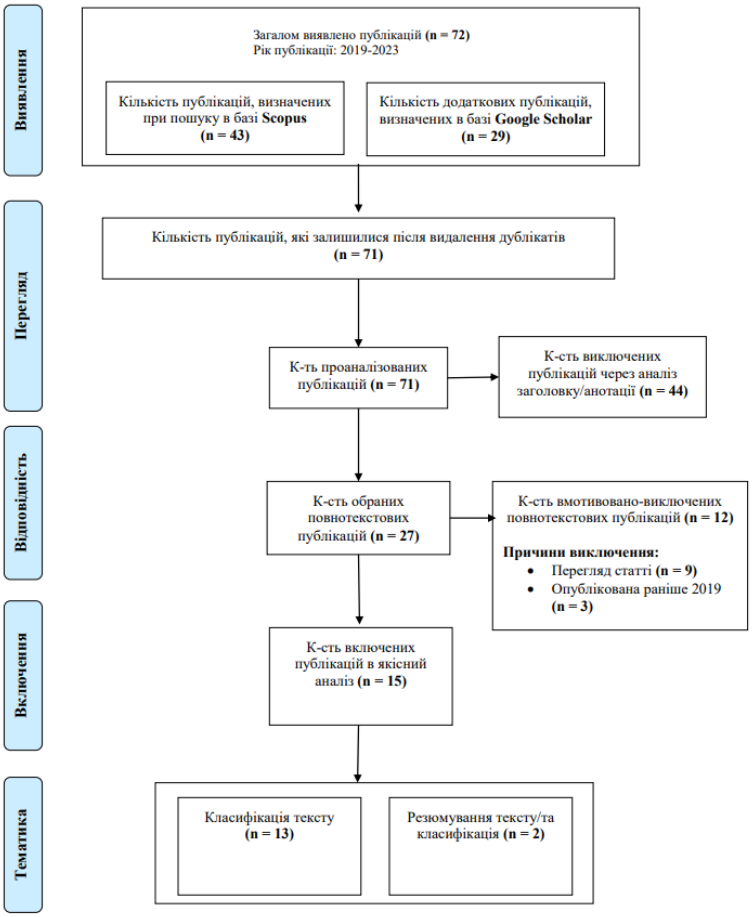

Рис. 1.2. Отримана схема PRISMA

Автори з [29] досліджували як покращити виявлення фейкових новин за допомогою автоматичного резюмування тексту. Пропонується покращена модель BERT, яка має назву CMTR-BERT. Дана модель відрізняється від звичної тим, що використовує додаткову контексту інформацію, як-от відео- чи фото-контент. Запропонований метод показав достатньо хороші результати на досліджуваних наборах даних, що доводить важливість додаткової контекстної інформації. Однак звідси виникає й недолік роботи – дані без додаткової інформації, яких все ще більшість у світі.

У роботі [6] автори порушують проблему класифікування коротких новин лише за заголовком, тобто доводять недоліки багатьох моделей NLP, які використовують великі масиви тексту без використання семантики. Натомість тут намагаються використати попередню обробку заголовків за допомогою word2vec з TF-IDF, а опісля застосувавши ансамблеві методи машинного навчання (KNN, наївний байєсівський класифікатор, SVM). Найкращий результат показав наївний байєсівський класифікатор. Серед недоліків роботи – дослідження відбувалось на невеликих наборах даних з трьома класами, та може показати зовсім інші результати на більших даних.

Робота [5] наводить тезу, що сучасні методи класифікації тексті найбільш розвинуті в декількох мовах (англійська, китайська) та ставить за мету розробити покращену модель для менш популярної мови – казахської. Автори наводять дослідження щодо побудови моделей, які використовують лише текстові (наївний байєсівський класифікатор), лише фото-дані (CNN), або ж комбінують їх (DNN). Останні, показали кращий результат, що доводить доцільність використання такого підходу для цієї мови. Однак, в цьому й знаходить основний недолік методу – він може бути не таким результативним для інших мов, тобто не є універсальним.

Дослідження [24] знову порушує проблему вирішення задач NLP для різноманітних мов. Автори наводять думку, що одні й ті ж методи NLP можуть давати зовсім різні результати для декількох мов. У даному випадку досліджується індонезійська. Пропонується використання згорткових нейронних мереж, які зазвичай притаманні вирішенням задач із зображеннями, до класифікації тексту. Розроблена модель показала непогані результати, однак дослідження проводились на доволі малому наборі даних (4 класи та 472 новини). Крім того, автори вказують, що потрібно провести подальше дослідження з комбінуванням різних шарів, а також функцій активації, що може суттєво вплинути на результати. До того ж цей метод може бути менш ефективним для інших мов.

У своїй роботі [21] автори досліджують можливості класифікації новин різними методами для бенгальської мови. Було розглянуто можливості DNN, а також популярних класифікаторів: KNN, наївний байєсівський класифікатор, SVM,

дерево рішень. Для векторизації використовувалась техніка TF-IDF. Одними з найкращих виявились мультишарові глибинні нейронні мережі, які показали високу точність, та швидку видачу результатів. Серед недоліків варто виділити, що ці методи підходять для достатньо великих наборів даних для тренування. Також ці методи можуть видавати інші результати для інших мов.

Робота [16] є доволі об'ємною та вирішує специфічну проблему – класифікація статей в новинні газі з метою виявлення новин про дорожньо-транспортні пригоди задля сповіщення населення в цьому регіоні. Як і серед попередніх робіт, автори використовували багато різних моделей задля вибору найкращих. Такими виявились – SVM та випадковий ліс. Крім того, було зазначено, що набір даних є незбалансованим, тож додатково застосовували декілька алгоритмів вибірки даних. Комбінування SVM/випадкового лісу та ROS/RUS (random oversampling/undersampling) виявились найефективнішими. Автори обіцяють пояснити таку поведінку в майбутній роботі, що залишається відкритим питанням. Окрім цього, серед недоліків роботи варто зазначити, що тут розглядається бінарна класифікація.

Автор [33] посилається на швидке збільшення обсягу інформації та новин загалом у світі, через що виникає потреба в автоматичні класифікації цих новин. Дослідження проводиться на двох методах – класифікації за допомогою наївного байєсівського класифікатора та TF-IDF. Автор зміг швидко досягнути результатів, однак часто показник точності доволі низький, що мотивує подальше дослідження.

У роботі [23] автори провели суттєву роботу зі збору інформації з існуючих робіт у сфері класифікації новин, а також дослідили можливості класифікації новин за допомогою 4 методів: KNN, SVM, наївного байєсівського класифікатора та логістичної регресії. Серед позитивних сторін роботи варто виділити набір даних на якому проводились дослідження. Це приблизно 75 тисяч новин з семи різних ресурсів. Також до нього описуються етапи попередньої обробки. Як і в декількох попередніх роботах, найкращим результатом володіє наївний байєсівський класифікатор. Однак, виникають декілька проблем з використанням методів машинного навчання для класифікації новин, які самі автори зазначають.

Наприклад, виникнення нового, невідомого для моделі, типу новин.

Дослідження [15] присвячено класифікації новин та виділення настроїв користувачів за допомогою нового методу класифікації на основі м'яких обчислень з використанням теорії близьких множин. Алгоритм (TSC) показав себе краще класичних методів (як-от SVM чи випадковий ліс) на 7 з 10 досліджуваних наборів даних. Основна перевага запропонованого алгоритму – вища точність для коротких текстів. Однак, зі збільшенням довжини потенційної новини/публікації ефективність та доцільність використання цього методу буде зменшуватись.

Робота [9] ставить за мету вирішення задачі класифікації новин мовою тигринья – ефіосемітська мова, що знайшла поширення в Еритреї. Варто зазначити, що щодо цієї мови є мало безкоштовних та вагомих наборів даних. Саме тому автори самостійно сформували набір з 30000 новин на основі публікацій місцевих газет. Як результат, дослідники отримали порівняння існуючих класичних методів з CNN+word2vec+FastText. Моделі на основі згорткових нейронних мереж показали більшу точність класифікації, що доводить доцільність цього підходу для випадку з тигринью. Автори припускають, що метод може бути так само ефективним і для інших мов, однак він вимагає великої кількості даних для тренування.

У [7] автори акцентують увагу на важливості вибору гіперпараметрів моделей. Так, оптимізована модель SVM показала на 20% більшу точність, ніж звичайні моделі неоптимізована модель. Також була спроба покращити інші класичні підходи (SGD, випадковий ліс, логістична регресія, KNN), але оптимізована модель SVM працювала краще, ніж інші моделі. Тоді як без оптимізації її продуктивність була гіршою. Робота залишає відкритим питання підходу до вибору гіперпараметрів, тобто немає алгоритму вибору найбільш оптимізованих гіперпараметрів, а більше схиляється до емпіричного підходу. Саме тому вона може використовуватись як хороший приклад обґрунтування важливості гіперпараметрів, але не вирішення як саме їх обирати.

Дослідження [27] присвячено розробленню додатка під назвою TwitterBulletin. Це програма для збору, класифікації та відображення новин

турецькою мовою з цільовою аудиторію – журналістами. Автори зазначають, що використали турецьку бібліотеку для обробки мови Zemberek. Крім того, використали різні методи векторизації (TF-IDF, N-Gram, word2vec) та алгоритми класифікації (CNN, SVM, KNN, логістична регресія). Додаток показав хороші показники точності та безумовно має прикладне застосування. Але він заточений на використанні бібліотеки Zemberek, що унеможливлює його використання для інших мов.

Остання публікація [4] ставить за мету вирішити дві задачі – вилучення важливого вмісту з веб-ресурсів (новин з новинних сайтів) та резюмування тексту. Запропонований підхід містить виділення ключових фраз на основі вибору з фраз-кандидатів за допомогою функції ваги, яка в свою чергу містить TF-IDF, бере до уваги позицію фрази та побудову семантичних зв'язків між словами за допомогою WordNet. Даний метод є перспективним, однак він також потребує покращення, оскільки є випадки коли при стисненні втрачається важлива інформація, як-от дата, час або ж місце публікації. Крім того, автори статті більше акцентують на виявленні важливого контенту, тим самим упускаючи деталі імплементації резюмування цього контенту.

Вище описані наукові статті також було систематизовано у вигляді таблиці 1.1.

Таблиця 1.1. Огляд існуючих робіт

№	Назва	Інструментарій	Задача	Рік	Посилання
1	A Survey on Text Classification Algorithms: From Text to Predictions	word2vec, FastText, GloVe, KNN, дерево рішень, логістична регресія, CNN	Класифікація тексту	2022	[10]
2	A Text Classification Model via Multi-	TextCNN, BiLSTM, Word2Vec	Класифікація тексту	2022	[13]

№	Назва	Інструментарій	Задача	Рік	Посилання
	Level Semantic Features				
3	Applying Automatic Text Summarization for Fake News Detection	CMTR-BERT, BERT	Резюмування тексту для виявлення фейкових новин	2022	[29]
4	Automatic Semantic Categorization of News Headlines using Ensemble Machine Learning: A Comparative Study	KNN, наївний байєсівський класифікатор, SVM).	Класифікація новин	2019	[6]
5	Classification of Scientific Documents in the Kazakh Language Using Deep Neural Networks and a Fusion of Images and Text	DNN, CNN, наївний байєсівський класифікатор	Класифікація наукових робіт	2022	[5]
6	Indonesian news classification using convolutional neural network	CNN	Класифікація новин	2020	[24]
7	Multi-category Bangla News	KNN, SVM, DNN, наївний	Класифікація новин	2021	[21]

№	Назва	Інструментарій	Задача	Рік	Посилання
	Classification using Machine Learning Classifiers and Multi-layer Dense Neural Network	байєсівський класифікатор			
8	News Classification for Identifying Traffic Incident Points in a Spanish-Speaking Country: A Real-World Case Study of Class Imbalance Learning	SVM, випадковий ліс, KNN, data sampling techniques	Класифікація новин	2020	[16]
9	News Classification Using Machine Learning	Наївний байєсівський класифікатор, TF-IDF	Класифікація новин	2021	[33]
10	ONLINE NEWS CLASSIFICATIO N USING MACHINE LEARNING TECHNIQUES	SVM, KNN, наївний байєсівський класифікатор, логістична регресія	Класифікація новин	2021	[23]
11	Short Text Classification with Tolerance-Based Soft Computing Method	TSC (Tolerance Sentiment Classifier), SVM, випадковий ліс	Класифікація тексту	2022	[15]

№	Назва	Інструментарій	Задача	Рік	Посилання
12	Text Classification Based on Convolutional Neural Networks and Word Embedding for Low-Resource Languages: Tigrinya	CNN, word2vec, FastText	Класифікація тексту	2021	[9]
13	Topic Classification of Online News Articles Using Optimized Machine Learning Models	SVM, SGD, випадковий ліс, KNN, логістична регресія	Класифікація новин	2023	[7]
14	TwitterBulletin: An Intelligent and Real-Time Automated News Categorization Tool for Twitter	Zemberek, CNN, SVM, KNN, логістична регресія, TF-IDF, N-Gram, word2vec	Класифікація новин	2022	[27]
15	Web-Based News Straining and Summarization Using Machine Learning Enabled Communication Techniques for	WordNet, TF-IDF	Виявлення та резюмування новин	2022	[4]

№	Назва	Інструментарій	Задача	Рік	Посилання
	Large-Scale 5G Networks				

У процесі пошуку наукових джерел в наукометричних базах, можна помітити зростання інтересу до проблеми класифікації та/або резюмування новин за останні роки (рис. 1.3). Це підтверджує актуальність наукової спільноти до цього напрямку дослідження.

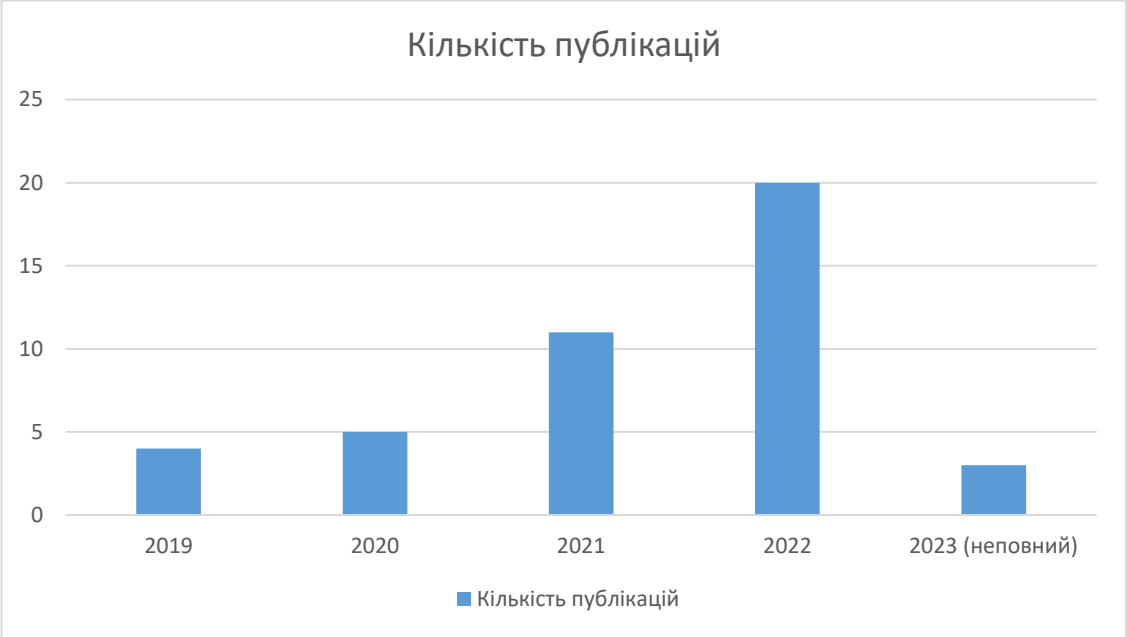

Рис. 1.3. Кількість публікацій за обраною тематикою за останні 5 років, опублікованих в наукометричній базі Scopus

Для категоризації та резюмування новинних статей існує ряд проблем, які в деяких попередньо описаних рішеннях вирішуються лише частково:

1. Недостатня кількість даних для дослідження з різними мовами.
2. Широка варіація методів для вирішення категоризації, які можуть мати різні результати в різних ситуаціях/мовах. Тобто, це потребує дослідження задля ідентифікації найбільш ефективних та точних для конкретної проблеми.
3. Проблема маркування новин з новими категоріями.
4. Ризик втрати важливої інформації при стисненні.

5. Низька кількість наукових робіт щодо резюмування новин.
6. Відсутність інструментів, які б вирішували ці дві задачі (категоризація та резюмування) водночас.

1.3. Постановка задачі.

Мета роботи полягає в дослідженні та можливості покращення ефективності методів класифікації та резюмування новин. Результатом роботи, який планується досягнути є порівняльна характеристика методів класифікації та резюмування новин на основі низки досліджень, а також концепція програмного продукту задля автоматичної обробки свіжих новинних статей.

Для досягнення мети необхідно вирішити ряд задач, а саме:

7. Дослідити існуючі методи категоризації тексту.
8. Дослідити існуючі методи резюмування тексту.
9. Здійснити огляд наборів даних, які можуть бути використані при виконанні цієї роботи. За потреби, побудувати новий набір даних.
10. Програмно реалізувати декілька методів категоризації та резюмування.
11. Провести тестування побудованих моделей та аналіз отриманих результатів.

1.4. Висновки

В даному розділі було розглянуто сферу використання засобів обробки природньої мови. Також наведено важливість в сучасному світі онлайн-ресурсів в сучасному житті людей. Був проведений аналіз літератури на основі схеми PRISMA. Результат цього показав зацікавленість наукової спільноти в області класифікації та резюмування текстів.

Крім того, було визначено мету роботи та перелік завдань, вирішення яких обов’язкове для досягнення мети.

2. ДОСЛІДНИЦЬКИЙ РОЗДІЛ

2.1. Аналіз предметної області.

Класифікація тексту – це процес визначення категорії тексту на основі його змісту, наповнення. Це здобуло поширення в багатьох сферах та вирішують низку прикладних задач (визначення спаму, визначення мови тексту, присвоєння тегів статтям, тощо).

Існує декілька підходів до побудови систем, що вирішують проблему класифікації (рис. 2.1).

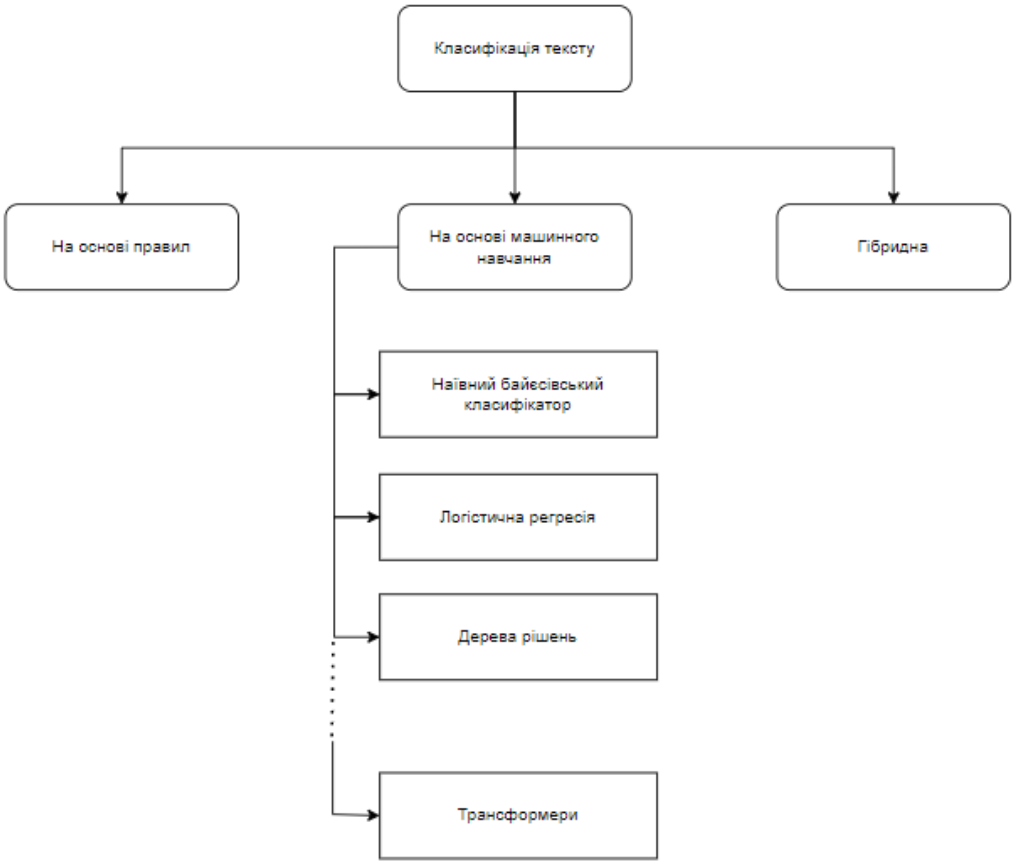

Рис. 2.1. Методи класифікації тексту

Розглянемо детальніше кожен з підходів:

- Системи на основі правил. У даному випадку віднесення тексту до конкретної категорії відбувається за наперед визначеними правилами [19]. Зазвичай це відбувається за наявністю специфічної лексики для

різних сфер. Тоді відбувається побудова словника, де до кожного класу (категорії) присвоюється список слів. Наприклад, для новин до теми “Злочини”, може бути створений словник з такими словами: кримінальний кодекс, крадіжка, корупція. Процес присвоєння категорії буде супроводжуватиметься підрахунок входження слів до нового тексту і присвоєння категорії, кількість слів з якої є найбільше в тексті. Даний підхід є простим в реалізації, але може бути доволі неточним та дорогим, адже побудова лінгвістичних правил для словників потребує глибокого розуміння класу чи категорії. Серед переваг методу – відсутня потреба в навчальних даних і легкість в розумінні.

- Системи на основі машинного навчання. Цей спосіб ґрунтується на використанні методів та моделей машинного навчання. Замість того аби самостійно вручну будувати правила віднесення тексту до категорії, розробники перекладають відповідальність на алгоритми, що можуть знайти певні зв’язки між текстом та класом. Даний підхід вимагає попередньої обробки тексту та репрезентації тексту у зрозумілому для алгоритму представленні – зазвичай векторі чисел.
- Гібридні системи поєднують використання обох попередніх підходів: на основі правил та машинного навчання. Така система додає більшої гнучкості та можливість вносити зміни. Наприклад, побудована модель машинного навчання може неправильно передбачати якийсь з класів, тоді додавання набору лінгвістичних правил допоможе обробити цей виняток. Але відразу стає очевидно, що таку систему важче підтримувати і розробляти.

Наївний байєсівський класифікатор – класифікатор на основі ймовірностей, який використовує теорему Байєса для визначення ймовірності приналежності до певного класу. При використанні цього методу вхідний текст представлений у вигляді торби слів [2], де слова представленні без врахування позиції чи граматичної форми, але враховуючи частоту входження в текст. Результат приналежності тексту до одного з класів описується формулою 2.1.

$$\hat{c} = arg\max_{c \in C} P(c|d) \quad (2.1)$$

де $\hat{c}$ це клас з найбільшою апостеріорною ймовірністю, тобто умовною ймовірністю приналежності до класу c за умови вхідного тексту d.

Логістична регресія – статистичний метод класифікації, який може бути адаптований для бінарної чи множинної класифікації (мультиноміальна логістична регресія) [31]. Даний ґрунтується на побудові лінії регресії для значень, що можна представити у двійковому вигляді [3]. У якості параметрів функції регресії можуть виступати певні мовні правила: кількість входжень слів певного лексикону, бінарне значення чи є знак оклику тощо. Серед недоліків метода варто виділити невисоку якість класифікації та високі обчислювальні витрати, однак метод є простим в реалізації.

Дерева рішень – метод машинного навчання, який використовує деревоподібну структуру, де листки репрезентують класи, а внутрішні вузли певні правила, які визначають рух вниз.

Автори [31] звертають увагу на те, як людина вивчає слова протягом життя, зокрема дитинства. Враховуючи, що словниковий запас дорослої людини коливається від 30000 до 100000, можна припустити, що діти повинні вивчати в середньому по 7 слів щодня до досягнення 20-річного віку. Більшість з них дізнаються як побічний продукт читання. Іншими словами – процес тренування, який співставний з процесом тренування моделей машинного навчання. Таким чином – з’явилась ідея трансформерів – найпоширеніших мовних попередньо навчених архітектур. Трансформер пропонує механізми самоуваги та позиційного кодування, що може бути використано для вирішення різних завдань NLP. У тому числі й категоризації та резюмування тексту (новин). Саме різні види трансформерів буде досліджено в цій роботі, а архітектуру та головні ідеї деталізовано в розділі 2.3.

Однак, варто згадати історію і причини появи трансформерів. У 2017 дослідники з Google опублікували роботу під назвою “Attention is All You Need” [39]. У цьому дослідженні було кинути виклик найпопулярнішим моделям для

вирішення задач NLP того часу – рекурентним нейронним мережам (RNN). Поява архітектури трансформерів вплинула на розвиток сфери обробки природньої мови надзвичайно сильно, і вже сьогодні кожен чув про ChatGPT (GPT – Generative Pretrained Transformer) чи BERT (Bidirectional Encoder Representations from Transformers).

У основі RNN були декілька концепцій, які були перевикористані та покращені в трансформерах. Серед них варто виділити 2 основні: модель кодера-декодера та механізм уваги.

Візьмемо приклад вирішення завдання перекладу – на вході і виході мережі повинні бути послідовності довільної довжини. Тоді застосовують архітектуру кодера-декодера. Кодер відповідає за кодування (або ж перевід) інформації в числову форму (або ж прихований стан). Натомість декодер отримує цей стан і генерує нову послідовність (рис. 2.2).

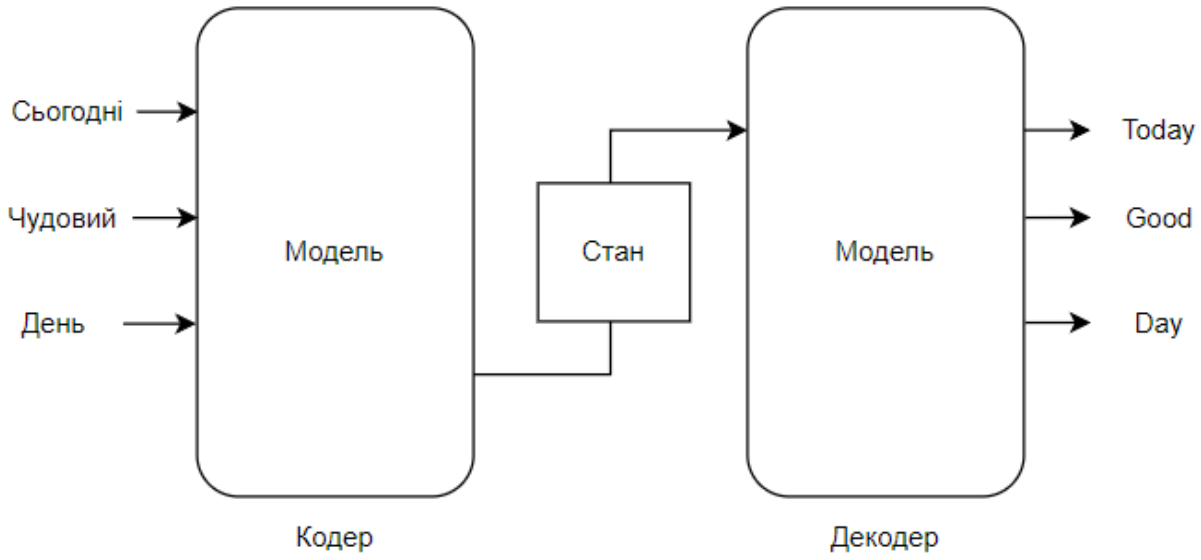

Рис. 2.2. Приклад роботи кодера-декодера

Однак, переводячи довгу послідовність слів в один остаточний стан, може трапитись, що інформація буде втрачена в процесі стиснення до фіксованого значення. Вирішення цієї проблеми існує – вже раніше згаданий механізм уваги. Основна ідея полягає в виводі прихованого стану на кожному кроці обробки

вхідної послідовності. Наприклад, для наведеного прикладу на рис. 2.10 буде 3 різні стани, що рівне кількості вхідних слів. Натомість декодер міститиме окремий блок уваги, який зможе визначати пріоритети на кожному кроці декодування, або ж які стани краще використовувати [20]. Цей механізм дозволив знаходити нетривіальні зв'язки між словами, що значно покращило якість машинного перекладу. Але разом з тим, виникла й нова проблема – обчислення були послідовні і не могли підлягати розпаралеленню. Через це навчання таких моделей могло забирати багато часу. Цей недолік було вирішено разом з появою трансформерів, що буде більш детально розкрито в підрозділі 2.3.

Згадуючи сумаризацію, варто згадати, що це процес зменшення обсягу великого тексту, зберігаючи головні тези/ідеї. Загалом алгоритми сумаризації бувають двох видів:

- Екстрактивна сумаризація. Це процес виділення вже існуючих речень з тексту. Тобто алгоритм повинен оцінити речення на основі певної метрики, відсортувати за цією оцінкою та повернути деяку кількість найважливіших речень. Алгоритм не може згенерувати нове слово чи речення, яке не було згадане в вхідному тексті.
- Абстрактна сумаризація. Даний метод є більш складним у реалізації та відрізняється можливістю генерувати вихідний результат на основі фактів в тексті та змістовного значення.

2.2. Аналіз та побудова набору даних.

Існує велика кількість наборів даних, які містять інформацію про класифікацію чи сумаризацію текстів. Це може бути як категоризація за змістом, так і за тональністю. Крім того, існують набори даних вузько-спеціалізованої тематики. Натомість саме новини можуть містити чи не найрізноманітнішу лексику з різних сфер. Для вирішення та можливості тестування необхідно аби набір даних містив інформацію про категорію новини, її заголовок або короткий зміст та повний текст статті.

У [37] зібрано одну з найчисельніших колекцій новин опублікованих на

ресурсі HuffPost [11] протягом 2012-2022 років. Набір даних містить 210 тисяч записів які промарковані на 42 категорії (наприклад, політика чи подорожі). Крім того, набір містить посилання на кожну онлайн-новину.

Набір [12] містить 3 мільйони записів датованих від 2001 до 2022 року. Однак, містить вкрай мало інформації щодо походження статті чи повного тексту. Також, варто відмітити, що деяка частина рядків містить невідому категорію.

BBC News Summary [25] – набір даних призначений для дослідження сумаризації тексту. Він містить 2225 документів поділених на 5 категорій. З недоліків варто відмітити саме низьку кількість записів та охоплення різних категорій.

Решта досліджених наборів даних містили інформацію щодо походження новин (реальна чи фейк) або ж містили недостатню інформацію щодо локальних новинних ресурсів.

Отже, найоптимальнішим варіантом є вибір першого набору даних. Велика кількість записів дозволяє проводити детальне дослідження, а наявність посилання на першоджерело надає можливість застосування засобів видобування даних (web mining). Таким чином, у цій роботі буде розширено власноруч існуючий набір даних.

Нижче наведено вигляд News Category Dataset (рис. 2.3).

	link	headline	category	short_description	authors	date
0	https://www.huffpost.com/entry/covid-boosters-...	Over 4 Million Americans Roll Up Sleeves For O...	U.S. NEWS	Health experts said it is too early to predict...	Carla K. Johnson, AP	2022-09-23
1	https://www.huffpost.com/entry/american-airlin...	American Airlines Flyer Charged, Banned For Li...	U.S. NEWS	He was subdued by passengers and crew when he ...	Mary Papenfuss	2022-09-23
2	https://www.huffpost.com/entry/funniest-tweets...	23 Of The Funniest Tweets About Cats And Dogs ...	COMEDY	"Until you have a dog you don't understand wha...	Elyse Wanshel	2022-09-23
3	https://www.huffpost.com/entry/funniest-parent...	The Funniest Tweets From Parents This Week (Se...	PARENTING	"Accidentally put grown-up toothpaste on my to...	Caroline Bologna	2022-09-23
4	https://www.huffpost.com/entry/amy-cooper-lose...	Woman Who Called Cops On Black Bird-Watcher Lo...	U.S. NEWS	Amy Cooper accused investment firm Franklin Te...	Nina Golgowski	2022-09-22

Рис. 2.3. Записи в News Category Dataset

2.3. Функціональна та математична модель.

Класична архітектора трансформера складається з двох основних компонентів: кодера та декодера, які можуть складатись у окремі блоки, тобто

повторюватись [20](рис. 2.4).

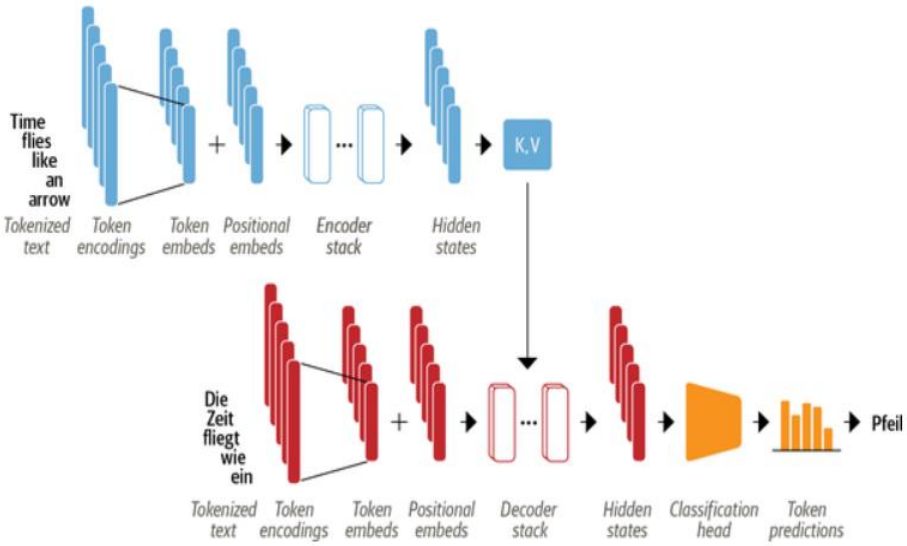

Рис. 2.4. Архітектура кодера-декодера в трансформерах

Додамо декілька слів щодо зображеної схеми:

- Вхідний текст проходить етап токенізація – розбиття послідовності на токени – менші за обсягом, зазвичай атомарні елементи. Після цього відбувається побудова вкладень слів (embeddings) – відображення слів або фраз з словника у вектори дійсних чисел [1]. Але при роботі з текстами важливо розуміти, що позиція кожного слова/токену несе додатковий зміст і впливає на його значення, або ж контекст. Це компенсується побудовою позиційних вкладень слів. Вони містять інформацію про розташування кожного токену в початковій послідовності.
- Кодер та енкодер зазвичай репрезентований не в одному екземплярі – він об’єднаний в декілька повторюваних шарів (блоків). Їхня кількість не є чимось загальноприйнята та зустрічається різна кількість блоків (6 в RoBERTa, 12 в AlBERT, 24 – оригінальний BERT).
- Декодер на кожному кроці декодування отримує результат з блоків кодера, а також свої попередні виходи. Наприклад, він виконує завдання перекладу – після отримання двох перших слів, декодер визначає найбільш ймовірне наступне слово на основі двох попередніх

слів, а також результату роботи кодера, таким чином не втрачаю контекст.

У класичній початкій версії трансформерів було представлено модель з наявністю обох елементів – кодера та декода. Це було зручно для вирішення завдань, де на вході й виході є текстові послідовності – машинний переклад. Однак, наразі трансформери вирішують набагато більше різноманітніших завдань і можуть бути трьох типів:

- Тільки кодер.
- Тільки декодер.
- Кодер та енкодер.

Моделі з наявними лише блоками кодера використовуються для вирішення завдань класифікації, і які якраз будуть досліджені в цій роботі. Натомість моделі з лише декодерами використовують в задачах передбачення слів. Наприклад, відповідей на питання чи продовження речень. Також варто зазначити, що моделі з наявністю кодером та декодером також використовуються задля сумаризації, або ж резюмування тексту. Використання в межах вибраного набору даних повинно виглядати наступним чином (табл. 2.1).

Таблиця 2.1. Приклад роботи класифікатора та сумаризатора

Оригінальний текст новини, перші декілька речень	Короткий опис (резюмування)	Категорія
U.S. health officials say 4.4 million Americans have rolled up their sleeves for the updated COVID-19 booster shot. The Centers for Disease Control and Prevention posted the count Thursday as public health experts bemoaned President Joe Biden’s recent remark that “the	Health experts said it is too early to predict whether demand would match up with the 171 million doses of the new boosters the U.S. ordered for the fall.	U.S. NEWS

Оригінальний текст новини, перші декілька речень	Короткий опис (резюмування)	Категорія
pandemic is over.”The White House said more than 5 million people received the new boosters by its own estimate that accounts for reporting lags in states.Health experts said it is too early to predict whether demand would match up with the 171 million doses of the new boosters the U.S. ordered for the fall….		

Загалом блоки кодера та декодера доволі схожі, оскільки містять наступні елементи (рис. 2.5) [31]:

- Шар самоуваги.
- Шар передавання (поширення).
- Нормалізація.

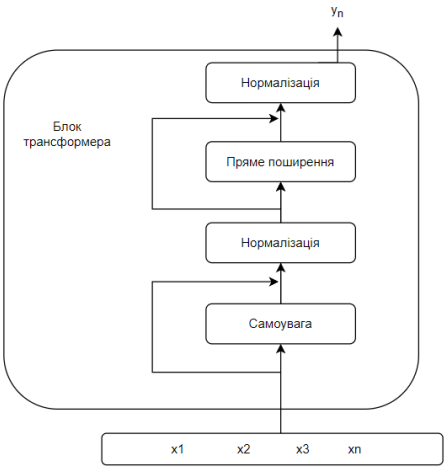

Рис. 2.5. Приклад будови блоку трансформера

Самоувага – головне нововведення архітектури трансформера. Як вже згадувалось раніше, механізм уваги це присвоєння кожному елементу певного

коефіцієнту, певної оцінки. Самоувага ж означає, що коефіцієнти будуть обчислюватись для кожного прихованого стану кодера чи декодера. Це дозволяє мережі окремо обчислювати контекст з великого обсягу інформації без проміжних рекурентних зв'язків.

Шар самоуваги отримує на вхід вектор значень $(x_1, x_2 \dots x_n)$, довжини n і видає на вихід вектор $(y_1, y_2 \dots y_n)$ такої ж довжини. При обчисленні кожного значення y_i, функція має доступ до всього вектора x та не залежить від обчислень на тому самому рівні (тобто y_j). Ця особливість дозволяє застосовувати засоби паралельних обчислень до навчання та використання безпосередньо натренованої моделі.

Самоувага базується на можливості порівняння елемента для якого обчислюється нове вихідне значення з іншими елементами текстової послідовності, таким чином зберігши контекст. Наприклад, якщо у нас послідовність з трьох елементів (x_1, x_2, x_3), то аби знайти y_3, потрібно порівняти x_3 з кожними елементом вхідної послідовності [20]. Метод порівняння – скалярний добуток:

$$score(x_i, x_j) = x_i \cdot x_j \tag{2.2}$$

Результат обчислень може набути різних значень, але задля ефективного використання оцінок потрібно нормалізувати значення, що надасть коефіцієнти пропорційного відношення поточного елемента до іншого, або ж контекст. Можливий варіант нормалізації – функція Softmax [17]:

$$\alpha_{ij} = softmax\left(score(x_i, x_j)\right) = \frac{\exp(score(x_i, x_j))}{\sum_{k=1}^{i} \exp(score(x_i, x_k))} \tag{2.3}$$

Отримана формула обчислення y_j, враховуючи 2.2 та 2.3:

$$y_i = \sum_{j \le i} \alpha_{ij} \cdot x_j \tag{2.4}$$

Однак, трансформери дозволяють обчислювати самоувагу трішки складнішим розширенішим способом. Для цього формують три матриці, які відображають різне призначення: запит, ключ і значення (query, key, value).

Аби це реалізувати формуються матриці ваг для кожного призначення – W^Q, W^K, W^V. Користуючись ними можна дістати представлення кожного значення x_i:

$$q_i = W^Q x_i,\ k_i = W^K x_i,\ v_i = W^V x_i \tag{2.5}$$

Таким чином, наступним кроком потрібно оновити формулу 2.2 якщо припустити, що подібність між двома елементами та це наскільки запит *i* підходить до ключа *j*:

$$score(x_i, x_j) = q_i \cdot k_j \tag{2.6}$$

Також варто згадати, що добуток може бути як завгодно великим чи малим, що може призвести до складнощів в подальшому обрахунку експоненти (формула 2.2.) і втрат градієнтів при навчанні. Аби уникнути цього, застосовують ділення на квадратний корінь із розмірності векторів запиту і ключа [31]:

$$score(x_i, x_j) = \frac{q_i \cdot k_j}{\sqrt{d_k}} \tag{2.7}$$

Отже, беручи до уваги 2.3, а також 2.1 з урахуванням, що x_j буде представлений у вигляді вектора значення v_j, отримуємо кінцеву формулу самоуваги:

$$SelfAttention(Q, K, V) = softmax\left(\frac{Q \cdot K^T}{\sqrt{d_k}}\right)V \tag{2.8}$$

Варто зазначити, що слова в реченні можуть співвідноситись багатьма способами. Це може бути зв’язок дієслів з об’єктами чи суб’єктами, різні семантичні та синтаксичні зв’язки тощо. Маючи лише один набір матриць (Q, K, V) неможливо врахувати всі ці зв’язки. Це вирішується реалізацією багатологоловних шарів самоуваги, де головою називають набір з трьох матриць. Ці набори розташовані на одній глибині і є незалежними один від одного. Даний підхід перегукується з набором фільтрів в згорткових нейронних мережах у вирішенні задач комп’ютерного зору, де кожен фільтр звертає увагу на свою ознаку (розмір, форма, колір тощо).

Створюючи вкладення слів і виконуючи паралельні обчислення для кожного з них, досі не згадувалась урахування розташування у вхідному тексті. Архітектура трансформерів пропонує застосування позиційний вкладень, адже це надасть додатковий контекст про порядок слів, що є важливим при роботі з текстами чи часовими рядами. Це реалізується додаванням до вхідних векторів позиційних вкладень, які обчислюються за допомогою тригонометричних функцій, тим самим створюючи унікальне значення для кожної з позицій. Але це не приєднання до кінця вхідного вектору, а звичайне поелементне додавання – на виході повинен бути вектор такого самого розміру. Розрізняють абсолютне та відносне позиційне кодування. З назви стає зрозуміло, що перше враховує позицію в тексті загалом і є продуктивним при роботі з невеликими об'ємами даних, натомість відносне – враховує елементи поруч, що може бути значно важливішим при роботі з більшими текстами.

На рисунку 2.5 також зображено нормалізацію та пряме поширення. Пряме поширення призначене для додаткової обробки виходів і виглядає як два повнозв'язні шари (кожен нейрон першого з'єднаний з кожним нейроном другого шару), у якому сигнал поширюється в одному напрямку. Важлива деталь – цей шар не обробляє вектор як одне ціле, а обробляє кожен елемент незалежно, що також дає можливість виконувати обчислення паралельно. Натомість нормалізація потрібна для приведення результатів до нормованого розміру (зазвичай в діапазоні від 0 до 1). Це допоможе краще навчатись моделям, уникнути вже згаданих проблем з градієнтом, прискорити збіжність навчання тощо. У архітектурах розрізняють два види нормалізації в залежності від її місця застосування – перед чи після певного шару. Тобто нормалізація входу чи виходу.

Рисунок 2.5 містить загальну архітектуру кодера-декодера, але декодер має дещо складніший механізм уваги, який представлений двома підшарами:

- Маскувальний багатоголовий шар самоуваги – гарантує, що згенеровані токени на кожному моменті часу побудовані на основі попередніх даних і поточному передбаченому токену. Маскування гарантує, що задача буде нетривіальною і декодер не буде просто

копіювати цільовий результат [20].

- Кодер-декодер шар уваги – на цьому етапі декодер має доступ до вихідного результат з попереднього шару, а також до проміжних результатів набору кодерів – значень (value), при цьому результат роботи декодера є запитом (query). Таким чином формуються зв'язки між різними наборами слів – наприклад, у задач машинного перекладу.

Важливим етапом при побудові та аналізі моделі машинного навчання є оцінка ефективності, або ж точності. Для класифікації метрики оцінювання базуються на декількох поняття:

- TP (true positive) – кількість правильно віднесених об'єктів до класу. Тобто модель правильно передбачила приналежність до класу.
- FP (false positive) – кількість приналежностей об'єкту до класу, яким він насправді є.
- TN (true negative) – кількість правильних передбачень неприналежності до класу.
- FN (false negative) – кількість неправильних передбачень неприналежності до класу.

На основі цих 4-х значень будуються також інші метрики:

Точність (precision) – кількість правильно передбачених об'єктів цього класу до загальної кількості об'єктів віднесених до цієї групи (2.9).

$$Presicion = \frac{TP}{TP+FP} \tag{2.9}$$

Повнота (recall) – кількість правильно передбачених об'єктів цього класу до загальної кількості справжніх об'єктів цього класу (2.10).

$$Recall = \frac{TP}{TP+FN} \tag{2.10}$$

Точність передбачення (accuracy) – кількість правильно приналежностей та неприналожностей до класу до загальних результатів (2.11).

$$Accuracy = \frac{TP+FP}{TP+FP+FN+TN} \quad (2.11)$$

F1 оцінка – метрика, що поєднує значення повноти та точності, тобто відображає поєднання точної і повної класифікації.

$$F1\ Score = 2 * \frac{Precision*Recall}{Precision+Recall} \quad (2.12)$$

Можна помітити, що ці метрики підходять до оцінювання класифікації щодо одного класу. Існує декілька способів агрегувати результати для багатокласового випадку:

- Macro average precision/recall/accuracy/f1 – підхід, при якому вираховується середнє значення для кожного класу. Наприклад у випадку розрахунку точності:

$$Macro - AveragedPrecision = \frac{1}{n}\sum_{i=1}^{n} P_i, \quad (2.13)$$

де *n* відповідно кількість класів, а P_i – оцінка точності для класу з індексом *i*.

- Micro average precision/recall/accuracy/f1 – підхід при якому вираховується кожна складова метрики загалом для всіх класів. Наприклад для точності:

$$MicroAveragedPrecision = \frac{\sum_{i=1}^{n} TP_i}{\sum_{i=1}^{n}(TP_i+FP_i)} \quad (2.14)$$

Задача резюмування вимагає також певного способу оцінки. Зазвичай, в такому випадку звертаються до Rouge (Recall-Oriented Understudy for Gisting Evaluation).

Rouge оцінює якість згенерованої сумаризації тексту порівнюючи його з попередньо написаним текстом людиною [32]. У якості одиниць для підрахунку виступають n-грами – послідовності з n елементів, які несуть певний мовний зв’язок між собою (наприклад, приклад біграми – “роботизована рука”, “дати жару”). У залежності чи ми будуємо Rouge для уніграм чи біграм є метрика Rouge1 та Rouge2. Відповідно, обрахунки проводяться наступним чином:

$$Rouge1\ Presicion = \frac{\text{к–сть слів,що входять в обидві послідовності}}{\text{к–сть слів,що є в згенерованому тексті}} \quad (2.15)$$

$$Rouge1\ Recall = \frac{\text{к–сть слів,що входять в обидві послідовності}}{\text{к–сть слів,що є в оригінальному тексті}} \quad (2.16)$$

Можна помітити, що жодна з цих метрик не може чітко оцінити якість сумаризації. Адже може статись так, що згенерований текст міститиме всі слова з оригінального, але буде мати значно більше сторонніх слів і навпаки – всі слова згенерованого тексту входитимуть в оригінальну послідовність, але вона буде значно більша за обсягом. Саме тому коли наводять метрику Rouge, то зазвичай йде мова про F1 оцінку, яка обраховується за формулою 2.12.

Також існує RougeL – у якості знаменника в точності та повноті виступає параметр найдовша спільна підпослідовність (longest common subsequence – LCS). Формула повноти набуде вигляду:

$$RougeL\ Presicion = \frac{LCS(s,e)}{\text{к–сть уніграм}}, \quad (2.17)$$

де s та e – згенерована та оригінальна послідовності.

2.5. Висновки

У даному розділі було проведено огляд існуючих підходів до вирішення поставленої задачі. Серед цього було описано ряд базових алгоритмів, а також детально розглянуто сучасне архітектурне рішення – трансформери. У ході роботи було наведено основні елементи трансформерів, історію та причини їхньої появи, а також нововведення, які вони принесли в сферу NLP. Крім того, було описано ряд метрик – показників, які будуть використовуватись в наступному розділі, аби оцінити результат роботи різних моделей та порівняти їх.

Також був проведений огляд існуючих наборів даних та їхню пристосованість до вирішення поставленого завдання. Було обрано набір даних на основі його характеристик, який буде використовуватись для тренування та тестування моделей в наступному розділі.

3. РОЗДІЛ АПРОБАЦІЙ ТА РЕЗУЛЬТАТІВ

3.1. Засоби реалізації.

Для розв'язання поставленої задачі, а саме класифікації та резюмування текстових новин була обрала мова програмування Python. Python є однією з найпопулярніших мов програмування для реалізацій завдань машинного навчання, зокрема й обробки природньої мови. Серед переваг цього вибору можна виділити наступне:

- Існує велика кількість бібліотек які спрощують розробку програмного коду, візуалізацію результатів тощо;
- Популярність мови також породжує велику кількість обговорень, досліджень та статей.

Крім того, задля проведення дослідів була обрана платформа Jupyter Notebook, яка дозволяє запускати шматки коду як логічні незалежні блоки, що робить результати більш зрозумілими та інтерактивними. Також досліди спочатку проводились на особистому ноутбуці, однак через комплексні і складні обчислення, цих потужностей було замало, було зроблено вибір в сторону іншого середовища виконання – Google Colaboratory [26]. Це безкоштовний ресурс, який надає доступ до різних обчислювальних ресурсів:

- CPU (Central Processing Unit) – центральний процесор
- GPU (Graphics Processing Unit) – графічний процесор
- TPU (Tensor Processing Unit) – тензорний процесор.

Особливо потрібно звернути увагу на останній – TPU. Тензорний процесор був розроблений компанією Google для оптимізації та прискорення виконання операцій пов'язаних з штучним інтелектом. Цей процесор заточений на роботі з тензорами – спеціальних структур даних, які використовуються при навчання нейронних мереж. Дослідження [18] показують, що TPU є в 15-30 разів швидшим ніж GPU чи CPU, та в декілька десятків разів енергоефективнішим.

Також при виконанні цієї роботи було використано ряд сторонніх бібліотек, які надають різний допоміжних функціонал:

- Pandas – бібліотека для роботи з наборами даних. Надає функціонал візуалізації та препроцесингу даних (фільтрування, видалення дублікатів, сортування тощо);
- Requests – бібліотека для виконання HTTP-запитів в середовищі мови Python;
- BeatifulSoup – інструмент для парсингу HTTP чи XML об’єктів;
- Numpy – бібліотека для підтримки великих масивів даних та операцій високого рівня над ними;
- Tensorflow – зручний інструмент для роботи з нейронними мережами на всіх етапах: створення, тренування, тестування;
- Nltk – NLP бібліотека, яка надає широкий функціонал для препроцесингу даних;
- Matplotlib – потужна бібліотека для візуалізації даних у вигляді графіків та діаграм;
- Sklearn – бібліотека для статичного аналізу даних;
- Transformers – бібліотека для роботи моделями обробки природньої мови, зокрема на основі трансформері.

3.2. Вимоги до технічного та програмного забезпечення.

Для виконання поставленого початково був використаний персональний ноутбук з наступними характеристиками:

- Центральний процесор: AMD Razen 9 4900H
- Об’єм оперативної пам’яті: 16 ГБ.
- Графічний процесор: NVIDIA GeForce RTX 2060 6 ГБ
- Операційна система: Windows 10.

Операції видобування даних та вебскрейпінгу (перетворення в структуровані табличні дані інформації з веб-сторінок) проводились цими потужностями. Однак, тренування моделей вимагало більших потужностей, тому подальше тренування та тестування моделей проводилось в середовищі Google Colab з підпискою PRO, яка надає можливість використання збільшеного об’єму оперативної пам’яті та

потужніших процесорів. Google Colab це веб-застосунок, тому головні вимоги до програмного забезпечення – доступ до мережі Інтернет та наявність браузера (Chrome, Edge, Safari).

3.3. Опис реалізації.

3.3.1. Розширення та аналіз набору даних.

Як вже зазначалось вище, обраний набір даних містить недостатню інформацію і потребує додаткової обробки задля отримання нових даних. News Category Dataset містить посилання на повнотекстову новину (рис. 3.1), яке можна використати для видобування даних з веб-сторінки.

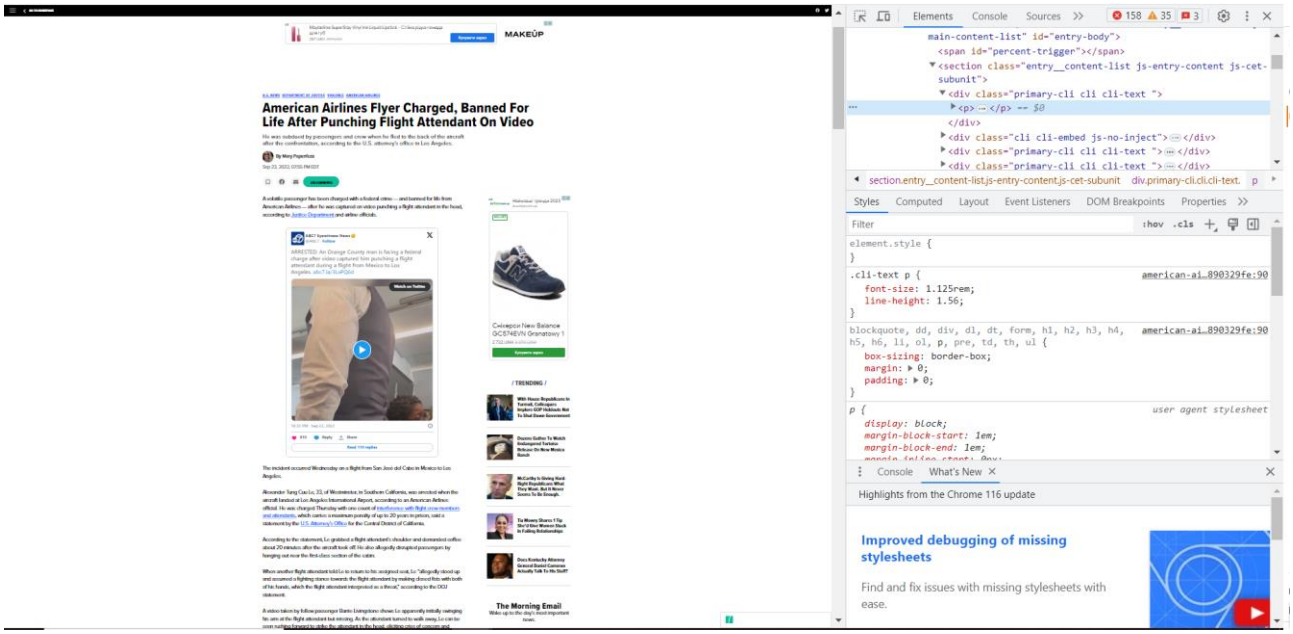

Рис. 3.1. Приклад новини на ресурсі HuffPost

Можна помітити, що основний текст статті розташований в html елементах div, які промарковані класом “primary-cli”. Також всередині може бути додатково посилання на сторонні ресурси чи інші обгорткові теги (div чи p). Засобами бібліотеки requests отримуємо html-код сторінки, а за допомогою класів BeatifulSup отримуємо текст абзаців за вищезгаданими критеріями. Результат роботи наведений на рис. 3.2.

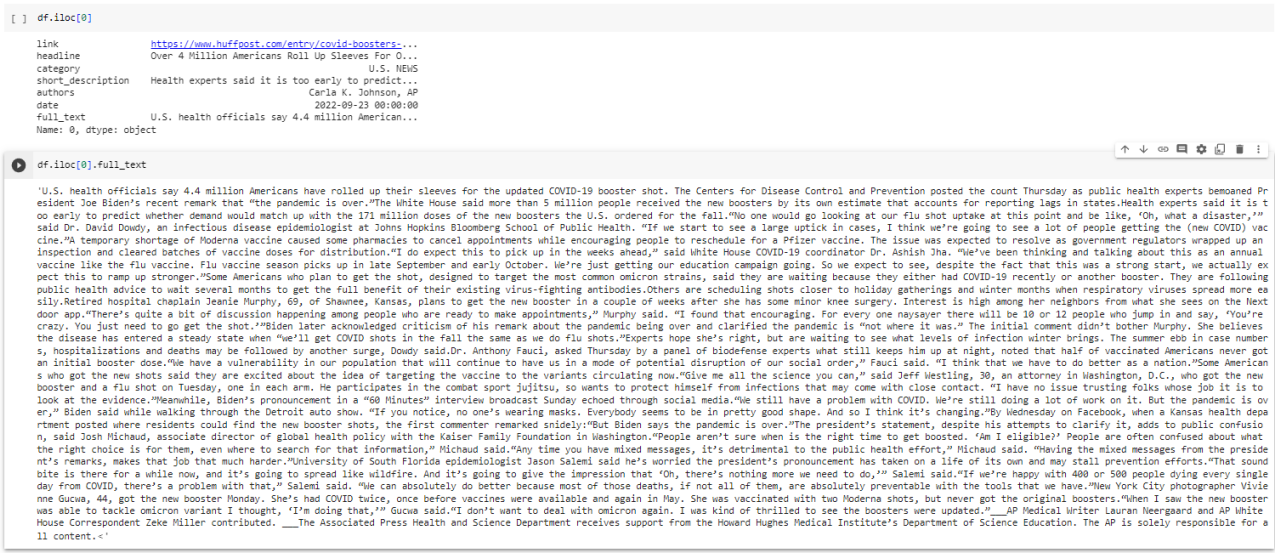

Рис. 3.2. Отриманий повнотекстовий варіант однієї з новин

Важливим етапом в машинному навчанні є препроцесинг даних. У випадку з вирішенням задачі класифікації текст повинен пройти ряд змін перед готовністю до використання:

- Токенізація: процес розділення тексту на токени (речення, слова). При цьому один великий текст стає набором лексем, кожна з яких може нести окреме інформаційне значення.
- Нормалізація: переведення форм слова до одного формату. Існує декілька факторів які потрібно врахувати: приведення слів до одного регістру (нижнього), лематизація – приведення різних морфологічних форм слова до початкової форми.
- Видалення стоп-слів: фільтрування невеликих часток мови, що не несуть значущості для вирішення задачі.
- Видалення спеціальних пунктуаційних знаків: фільтрування ком, крапок, двокрапок тощо.

Таким чином, процес препроцесингу тексту можна зобразити наступною діаграмою на рис.3.3.

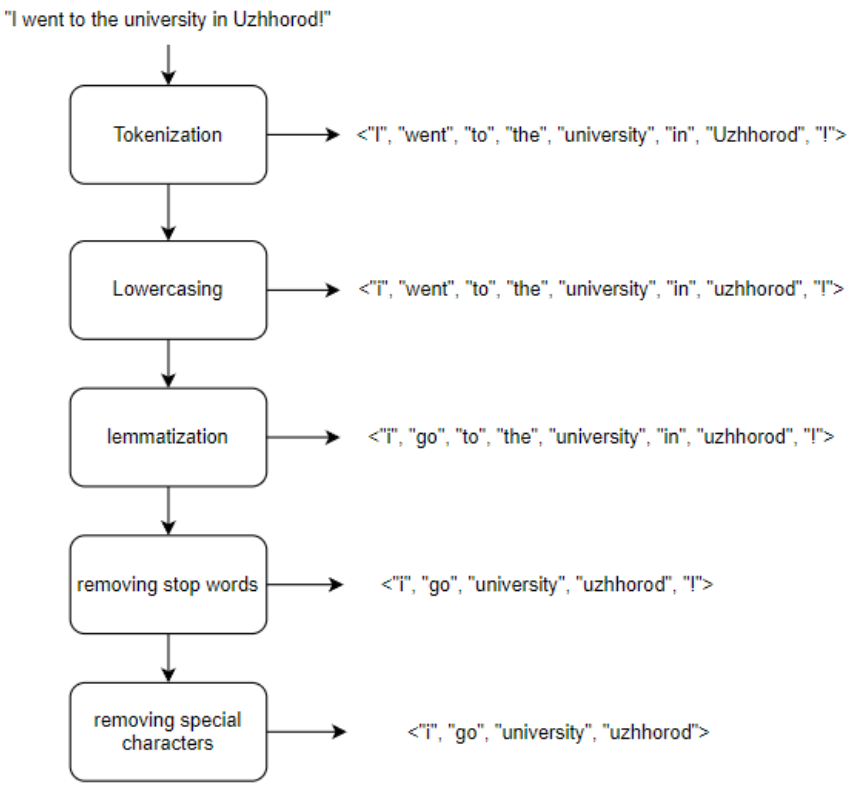

Рис. 3.3. Препроцесинг даних

Ще один етап – видалення дублікатів. Виявилось, що в датасеті присутні 13 однакових записів.

Проведемо короткий аналіз даних в наборі. Як вже зазначалось вище, в наборі присутні 42 категорії, а саме: 'U.S. NEWS', 'COMEDY', 'PARENTING', 'WORLD NEWS', 'CULTURE & ARTS', 'TECH', 'SPORTS', 'ENTERTAINMENT', 'POLITICS', 'WEIRD NEWS', 'ENVIRONMENT', 'EDUCATION', 'CRIME', 'SCIENCE', 'WELLNESS', 'BUSINESS', 'STYLE & BEAUTY', 'FOOD & DRINK', 'MEDIA', 'QUEER VOICES', 'HOME & LIVING', 'WOMEN', 'BLACK VOICES', 'TRAVEL', 'MONEY', 'RELIGION', 'LATINO VOICES', 'IMPACT', 'WEDDINGS', 'COLLEGE', 'PARENTS', 'ARTS & CULTURE', 'STYLE', 'GREEN', 'TASTE', 'HEALTHY LIVING', 'THE WORLDPOST', 'GOOD NEWS', 'WORLDPOST', 'FIFTY', 'ARTS', 'DIVORCE'. Деякі з цих категорій можна об’єднати в одну, як-от Black Voices та Latino Voices в групу під назвою Group Voices. Внаслідок таких перетворень зменшуємо кількість категорій до 27. Нижче зображено розподіл новин за різними категоріями

у вигляді секторної діаграми (рис. 3.4).

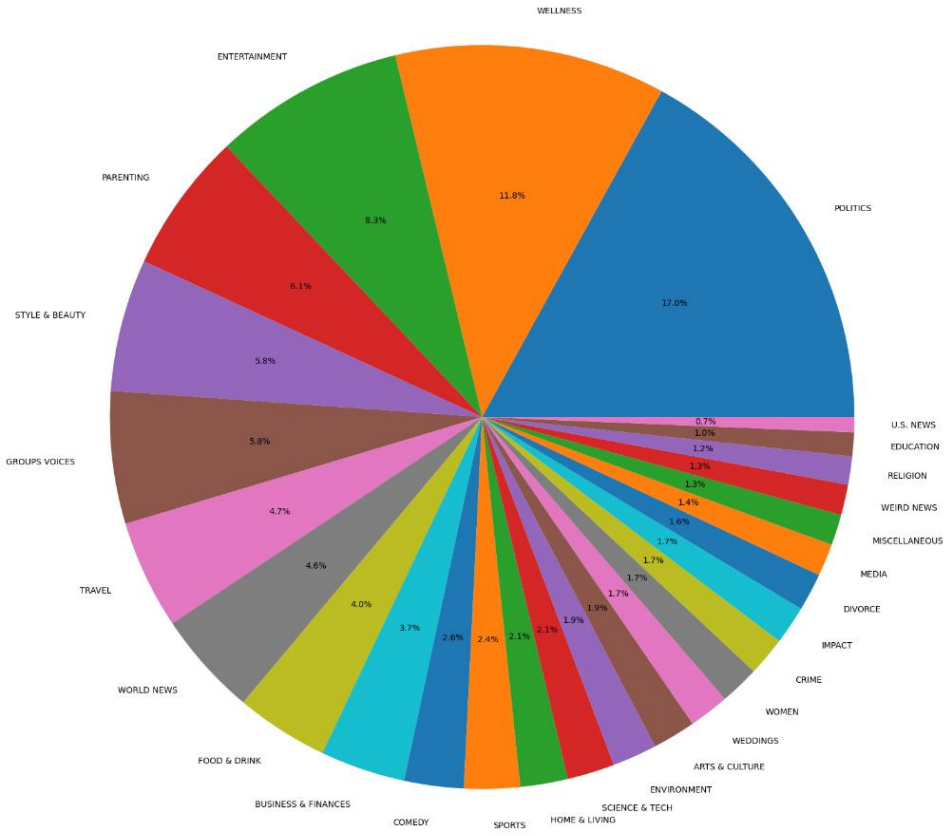

Рис. 3.4. Секторна діаграма розподілу новин за категоріями

Можна помітити, що набір даних не є збалансованим. Наприклад, записи з категорією “Politics” репрезентують 17% всього набору даних. Натомість, десяток категорій мають не більше 2%. Це доводить, що більше варто звертати увагу на метрику F1, ніж на звичайну точність, адже це компроміс між точністю і повнотою.

Також побудуємо хмару слів, яка відображає найбільш характерні слова в певних категоріях (рис. 3.5). Хмара слів зображується у вигляді малюнку з словами розташованими горизонтально та перелельно, а розмір шрифту відображає більшу сутність в межах контексту. Наприклад, у випадку з політикою очікувано виділяються слова “президент”, “трамп” (набір даних у тому числі формувався під час активної передвиборчої кампанії в США). Знаходження таких характерних слів у тексті новини може підвищити її віднесення до однієї з категорій.

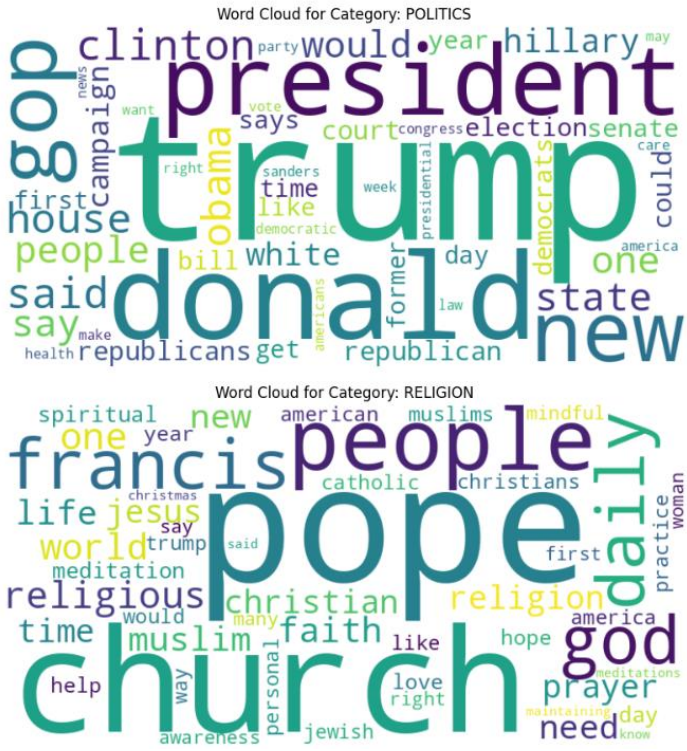

Рис. 3.5. Приклад хмари слів для двох категорій

3.3.2. Реалізація класифікації та резюмування даних.

Для проведення експериментів для класифікації використовуватиметься моделі на основі різних трансформерів, а саме DistilBert, RoBERTa, AlBERT. Всі вони є модифікацією оригінальної моделі Bert представленої в науковій роботі в 2018 році [8]. У свій час появи він перевищив найпопулярніші і сучасніші моделі NLP у вирішенні декількох завдань різної складності.

BERT започаткував ідею двонаправлених трансформерів-кодерів, натренованих за допомогою моделювання замаскованої мови (MLM – Masked Language Modeling). MLM – тренування мережі, маскуючи деяку кількість слів у початковій послідовності спеціальним токеном ([MASK]), таким чином змушуючи модель передбачити оригінальний текст. Двонаправленість у цьому випадку – можливість бачити контекст в обох напрямках: ліворуч та праворуч тексту. Відповідно, формула обрахунку вихідного вектору (2.4) змінилась:

$$y_i = \sum_{j=1}^{n} \alpha_{ij} \cdot v_j \tag{3.1}$$

Натомість все решта відповідає класичній архітектурі трансформерів:

вхідний текст підлягає токенізації, враховується позиційне кодування, присутні шари самоуваги тощо.

Таким чином основні характеристики моделі BERT:

- Розмір словника складає 30000 різних токенів.
- 12 блоків трансформера кожен з яких містить 12 голів самоуваги.
- Більше 100 мільйонів параметрів присутні в моделі.
- Обмеження на вхід – 512 токенів.

DistilBert – позиціонує себе як легший, швидший та менш вимогливий до обчислювальних потужностей послідовник Bert [38]. Розробники заявляють, що модель стала на 40% меншою і зберегла 97% всіх можливостей Bert, будучи при цьому на 60% швидшою. Якщо робити дослівний переклад – дистильований Bert.

Відмінні характеристики DistilBert:

- 6 блоків трансформера, кожен з яких містить 3 голови самоуваги.
- Близько 66 мільйонів параметрів.

RoBERTa – виникла як результат дослідження покращення продуктивності Bert шляхом зміни процесу попереднього навчання. Так, воно стало вимагати довший час, але продуктивність покращилась.

Відмінні характеристики RoBERTa це запровадження динамічного маскування. У випадку з Bert маскування проводилось на етапі підготовки даних, пакетами. Натомість RoBERTa пропонує виконувати це на етапі навчання динамічно. Крім того, автори акцентують увагу на важливості виборі гіперпараметрів, що суттєво покращує продуктивність та ефективність.

AlBERT – також є одним з послідовників архітектури BERT. Він приніс з собою інше нововведення – шарову розрідженість [36]. Це спосіб зменшити кількість параметрів мережі шляхом об’єднання параметрів декількох шарів в одне ціле. Як результат – всього 11 мільйонів параметрів.

Основа реалізації класифікації новин полягає в побудові та тренуванні нейронної мережі з використанням претренованих трансформерів, згаданих в раніше. Натомість в глибокій нейронній мережі буде декілька інших шарів (рис. 3.6).

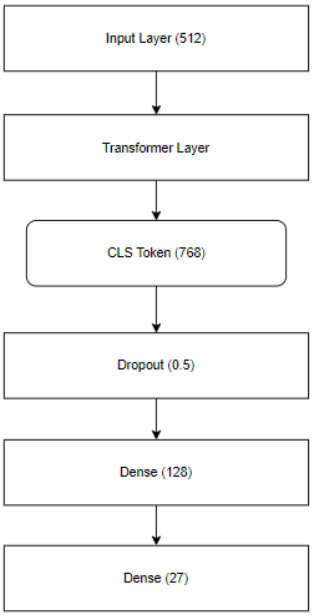

Рис. 3.6. Архітектура нейронної мережі з використанням трансформера

Перший шар приймає на вхід вкладання слів довжиною 512, які будуються токенізаторами, які відрізняються для кожного виду трансформерів. Тобто, відбувається перевід вектору текстових даних у вектор числових значень, потрібних трансформеру. Потрібний результат роботи трансформера представлений у вигляді CLS токена, який репрезентує весь текст отриманий на вході і може бути використаний для задач класифікації.

Dropout шар – метод регуляризації, який сприяє уникненню перенавчання шляхом виключення нейронів з деякою наперед вказаною ймовірністю.

Dense шар – повнозв’язний шар, тобто такий, що містить певну кількість нейронів кожен з яких є з’єднаним з кожним нейроном попереднього шару. Перший шар призначений для зменшення розмірності результатів отриманих від трансформера, а також може служити як додатковий засів регуляризації. Останній шар – вихідний шар з 27 нейронами та функцією активації softmax. Кількість нейронів зумовлена кількістю класів, тобто категорій новин, а функція softmax нормалізує отримані дані, що дає змогу репрезентувати це як ймовірності віднесення до певного класу.

У якості оптимізатора використовується оптимізатор Adam – метод стохастичної оптимізації, заснований на оцінці похідних першого та другого

порядку.

Метрика, що використовується для оцінки багатокласової класифікації – F1.

Нижче (рис.3.7-8) наведена історія тренування нейронних мереж з використанням різних трансформерів (DistilBert, RoBERTa, Albert).

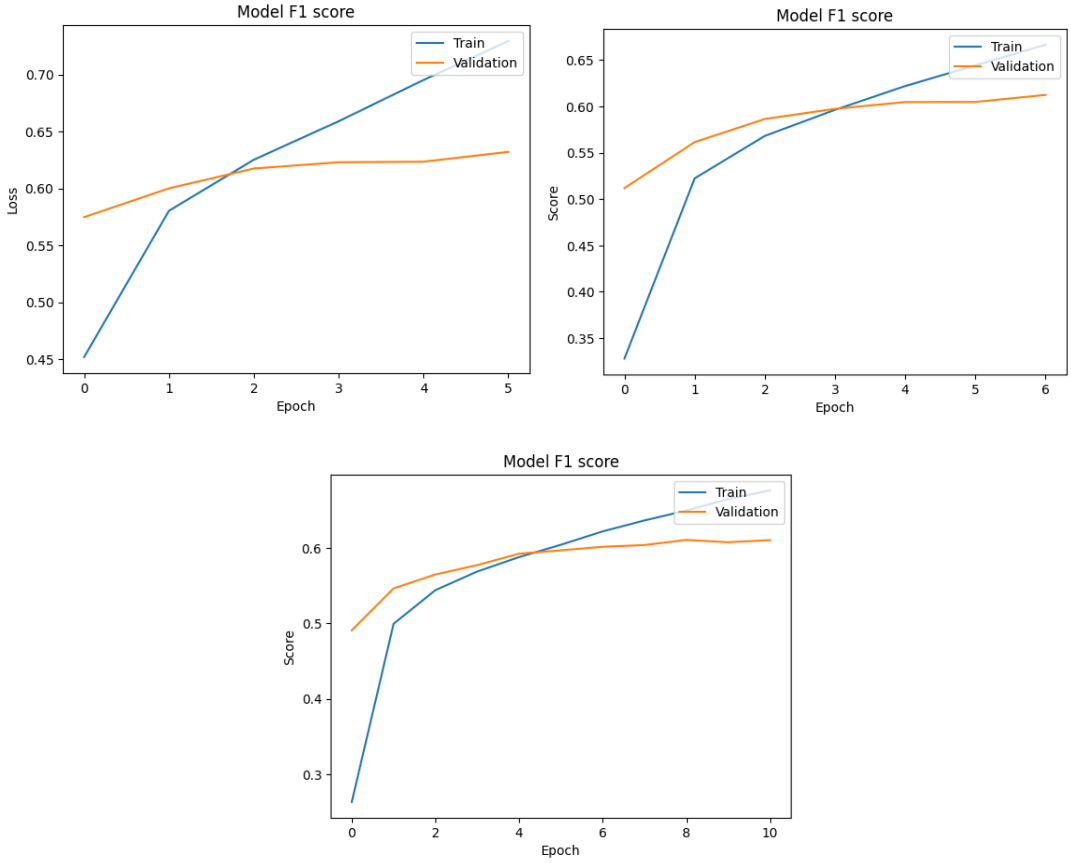

Рис. 3.7. Графік залежності метрики F1 від епохи тренування для методів DistilBert, RoBERTa та AlBERT

Усі три моделі показали швидку збіжність на перших епохах та поступове підвищення метрики точності для валідаційної вибірки. Однак, у різних ситуаціях по різному спрацював механізм ранньої зупинки. DistilBERT та RoBERTa перестали суттєво покращувати результати на валідаційній вибірці, натомість точність на тренувальній вибірці почало стрімко зростати, що могло викликати перенавчання. AlBERT теж мав схожий сценарій, але на пізніших епохах (10 проти 5). Можна припустити, що перші дві моделі навчаються швидше.

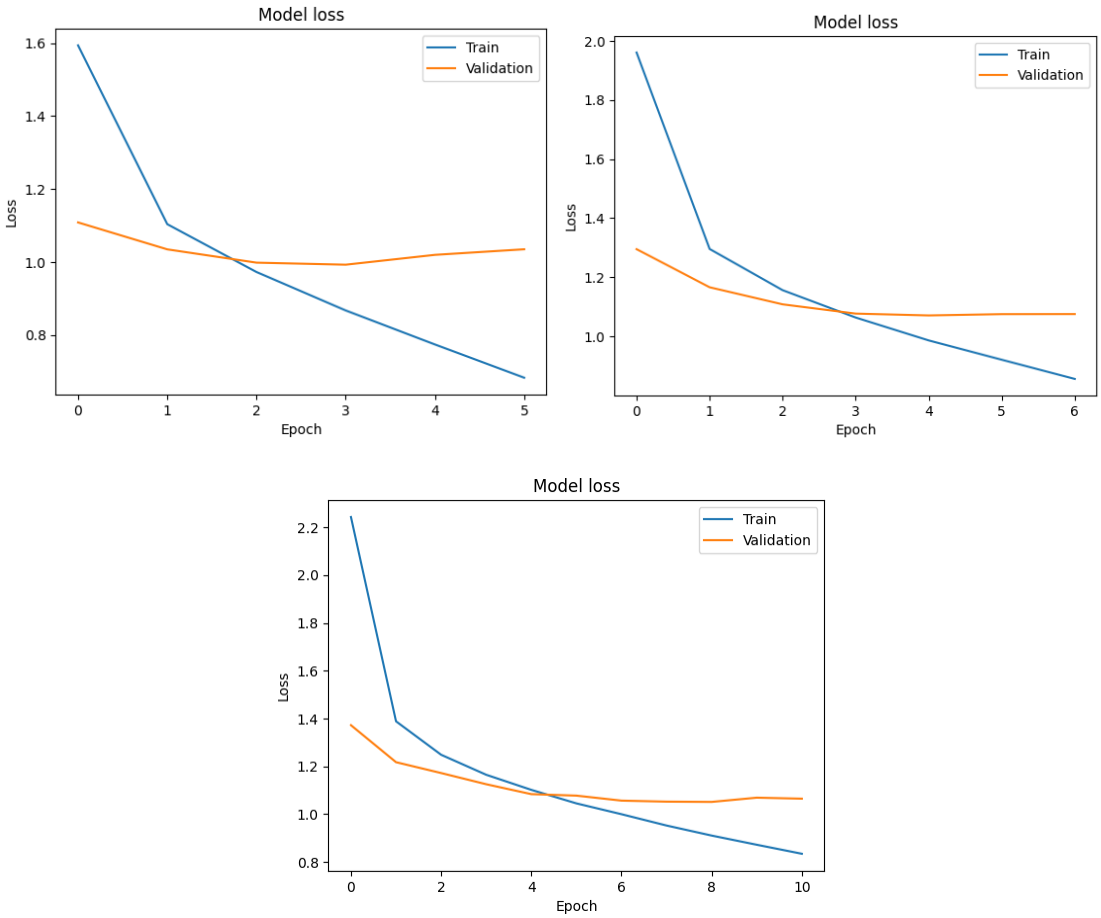

Рис.3.8. Графік залежності функції втрат від епохи тренування для методів DistilBert, RoBERTa та AlBERT

У випадку з аналізом графіку залежності функції втрат ситуація є схожою. AlBERT знову пізніше почав збігатись, а DistilBERT і RoBERTa проявили схожу поведінку. Варто зазначити, що на графіках теж помітна небезпека подальшого перенавчання. Найкраще для цього підходить DistilBERT – протягом останніх двох епох функція втрат для валідаційної вибірки почала зростати, натомість для тренувальної вибірки навпаки – продовжувати різко зменшуватись.

Для реалізації сумаризації було використано дві бібліотеки – Summarizer і SimpleT5. Перша надає доступ до низки моделей (BERT, XLNet) для сумаризації на основі претренованих трансформерів. Натомість, SimpleT5 надає функціонал для підточення моделі T5 на основі вхідних даних.

Код програми знаходиться за посиланням в репозиторії [30].

3.4. Аналіз отриманих результатів.

Систематизуємо отримані дані внаслідок тренування класифікаційних моделей в таблицю 3.1.

Таблиця. 3.1. Результати тренування та тестування класифікаційних моделей

Показник\Модель	MultinomialNB	DistilBert	RoBERTa	Albert
Accuracy	0.617	0.711	0.694	0.700
Precision	0.699	0.637	0.610	0.635
Recall	0.400	0.617	0.596	0.596
F1 score	0.432	0.620	0.597	0.609
Час тренування, хвилин	-	26,50	68,43	95,86
Час тестування, хвилин	-	43,15	88,70	90,23
Кількість параметрів трансформера	-	66362880	124645632	11683584
Категорія з найбільшою кількістю помилок	GROUPS VOICES	Politics	Politics	Politics

На основі отриманих результатів, можна відмітити домінацію моделі DistilBert над іншими. Це стосується майже всіх наведених показників. Частка правильних прогнозів є дещо вищою (0.711 проти 0.694 та 0.700), однак на кожній з моделі частка справжніх позитивних передбачень серед усіх позитивних результатів є дещо низькою. Звідси можна зробити висновок, що моделі потребують подальшого дослідження з метою покращення показників.

Також час тренування DistilBert є меншим ніж у Albert, попри те, що кількість параметрів є в декілька разів вищою. Це наочно відображає, що час тренування та роботи моделі залежить від багатьох факторів (особливостей архітектури, використання різних оптимізаторів тощо), не тільки кількості параметрів. Варто зазначити, що моделі на основі трансформерів показали набагато кращі результати ніж наївний байєсівський класифікатор, метрики якого є значно нижчими та не

можуть бути задовільними для використання в прикладних задачах класифікації тексту.

Для задачі резюмування досліджувались моделі на основі трьох трансформерів: BERT, XLNet та T5.

Модель BERT багато обговорювалась в цій роботі – це трансформер-кодер, який частіше всього використовують для задач класифікації. Але його також можна адаптувати для вирішення задач резюмування. Для цього трансформер-кодер використовують задля векторизації тексту, де кожне речення репрезентовано у вигляді вектору. Після цього застосовують кластеризацію. Отримані речення з більш схожими векторами будуть віднесені до кластеру схожого за змістом. Після чого буде можливість обрати одне речення, яке нестиме зміст кластера (екстрактивна сумаризація). Саме такий підхід реалізований в бібліотеці “bert-extractive-summarizer”.

XLNet – була створена аби покращити те, чого досягла модель BERT. Розробники дещо змінили механізм маскування: в BERT випадковим чином маскувались токени і модель повинна була передбачити, натомість XLNet переставляє всі токени випадковим чином і прагне передбачити наступний токен на основі всього контексту. Але загалом використання для вирішення задачі сумаризації схоже з BERT.

T5 – модель, яка найбільше відрізняється від попередніх двох, адже це представник моделей трансформерів кодер-декодер архітектури. Іншими словами, модель приймає на вхід текст і, як результат, видає на вихід також текст. Здатна вирішувати різні види завдань, одне з яких сумаризація, або ж резюмування.

Проведені дослідження з різними моделями сумаризаторів також систематизовані у вигляді таблиці 3.2. У якості характеристик порівняння було обранок тип сумаризації (узагальнення), час тренування та генерерації відповіді, а також набір метрик Rouge задля якісної оцінки отриманих результатів.

Таблиця 3.2. Результати тренування та тестування моделей резюмування

Показник\Модель	Bert	XLNet	T5
Узагальнення	Екстрактивне	Екстрактивне	Абстрактне
Час тренування, хвилин	-	-	78
Час генерації однієї сумаризації, секунд	1.51	1.43	0.73
Rouge1	0.187	0.285	0.310
Rouge2	0.066	0.273	0.182
RougeL	0.139	0.285	0.280
RougeLsum	0.140	0.285	0.279

Перш ніж аналізувати певні числові результати, варто згадати ключову відмінність між Bert/XLNet та T5. Перші дві моделі використовують екстрактивне узагальнення, тобто таке, що виділяє вже існуючі фрагменти тексту на вході. Натомість T5 є представником абстрактного узагальнення – генерації нового тексту, який може не бути фрагментом вхідного.

Час генерації відповіді є найменшою для T5, що робить його привабливим для великих та навантажених додатків. Також метрики Rouge для оцінки якості сумаризації вказують на домінацію T5, що я пов’язую з використанням абстрактного узагальнення.

3.5. Висновки

У цьому розділі було наведено опис середовища розробки та його переваги над іншими можливими засобами. Також був здійснений огляд додаткових бібліотек для мови Python, які були використані для реалізації та проведення досліджень.

Було наведено метод та інструменти розширення набору даних, який використовувався в подальшому дослідженні. Крім того, покроково наведено елементи препроцесингу даних, який потрібен для отримання кращих результатів. Після цього описано спосіб реалізації класифікації та резюмування новин. У якості

досліджень було обрано по 3 моделі-трансформера з якими проведено низка дослідів для отримання результатів, що були проаналізовані. Обрані моделі трансформерів для дослідження були описані, а їхні характеристики порівняні. Найкраще показали себе моделі DistilBert для категоризації та T5 для сумаризації новин. Даний висновок зроблений шляхом аналізу метрик точності для класифікації та резюмування.

ВИСНОВКИ

У ході виконання даної магістерської кваліфікаційної роботи було завершено ряд різних етапів. Серед них – пошук та аналіз різних наукових статей/робіт на основі схема PRISMA. Для пошуку наукових джерел було обрано наукометричні бази Scopus та Google Scholar. Вибір був аргументований авторитетністю та масштабністю першого та відкритості до різних джерел другого. Було відібрано 15 статей для якісного аналізу.

Наступний зроблений етап – аналіз предметної області, а також математичної моделі. Було наведено опис існуючих підходів і методів до вирішення завдання класифікації та резюмування текстів. Особливу увагу приділено сучасним архітектурам для вирішення завдань NLP – трансформерам. Протягом останніх років вони здобули шалену популярність і не дарма – їхні показники результативності та гнучкості у вирішенні різних завдань справді небезпідставні. Було наведено специфіку роботи трансформерів, їхніх новаторських рішень та архітектури.

Опісля був сформований набір даних на основі існуючої збірки новин з новинного порталу HuffPost. Вибір був аргументований розмір набору, а також можливістю до його розширення, що й було зроблено методами видобування даних з веб-сторінок новин. У ході дослідження було побудовано модель з використанням різних трансформерів задля вирішення задачі класифікації, а також наведення використання трансформерів для резюмування. Ключовою метою було дослідити різницю в роботі різних архітектур та можливості застосування в області новин. Моделі показали доволі високу точність, а такі характеристики як час навчання і роботи, дозволяють додатково порівняти їх.

Отримані результати довели можливість використання трансформерів задля вирішення задачі класифікації та їхню домінацію над примітивними алгоритмами. Так, метрика точності F1 для байєсівського класифікатора склала 0.432, натомість для архітектури з використанням трансформера DistilBERT – 0.620. Варто зазначити, що більшість трансформерів використовувались вже навчені, що дозволило використати велику кількість вже докладених зусиль науковою

спільнотою.

Тема магістерської роботи є актуальною, адже з кожним днем на людей здійснюється все більший інформаційний вплив і орієнтуватись в океані інформації стає важче. Отримані результати дозволяють впевнитись в можливості застосуванні методів машинного навчання, зокрема трансформерів, у області новин (новинні сайти, телеграм-канали, тощо).

Крім того, методів класифікації та резюмування існує набагато більше, ніж представлені в роботі, тож їхнє дослідження може продовжуватись, що дозволяє продовжувати дослідницьку діяльність наведену тут. Також планується розробка програмного продукту – веб-ресурсу, який використовуватиме трансформери задля автомаркування новин характерними темами, а також формування узагальнень новинних статей задля формування щоденних дайджестів для користувачів. Дані технології також дозволяють розробити систему рекомендацій, адже зазвичай користувачі цікавляться найбільше декількома сферами – автомаркування допоможе у цьому.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Вкладання слів. *Вікіпедія*. 15.08.2022. URL: https://uk.wikipedia.org/w/index.php?title=%D0%92%D0%BA%D0%BB%D0%B0%D0%B4%D0%B0%D0%BD%D0%BD%D1%8F_%D1%81%D0%BB%D1%96%D0%B2&oldid=36855980 (дата звернення: 12.11.2023).
2. Модель «торба слів». *Вікіпедія*. 11.08.2023. URL: https://uk.wikipedia.org/w/index.php?title=%D0%9C%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C_%C2%AB%D1%82%D0%BE%D1%80%D0%B1%D0%B0_%D1%81%D0%BB%D1%96%D0%B2%C2%BB&oldid=40142028 (дата звернення: 25.09.2023).
3. Сперкач М. О., Юзьвак Д. Ю. РОЗВ'ЯЗАННЯ ЗАДАЧІ КЛАСИФІКАЦІЇ ТЕКСТІВ МЕТОДАМИ ОБРОБКИ ПРИРОДНЬОЇ МОВИ ТА МАШИННОГО НАВЧАННЯ. *Міжнародний науковий журнал Науковий огляд*. Вип. 4, № 57. С. 62–71. Також доступний за посиланням: https://naukajournal.org/index.php/naukajournal/article/view/1817 (дата звернення: 25.09.2023).
4. Arora A., Gupta A., Siwach M. та ін. Web-Based News Straining and Summarization Using Machine Learning Enabled Communication Techniques for Large-Scale 5G Networks. *Wireless Communications and Mobile Computing*. ред. Mohammad Farukh Hashmi. Вип. 2022, 23.06.2022. С. 1–15. DOI:10.1155/2022/3792816.
5. Bogdanchikov A., Ayazbayev D., Varlamis I. Classification of Scientific Documents in the Kazakh Language Using Deep Neural Networks and a Fusion of Images and Text. *Big Data and Cognitive Computing*. Вип. 6, № 4. С. 123. DOI:10.3390/bdcc6040123.
6. Bogery R., Al N., Aslam N. та ін. Automatic Semantic Categorization of News Headlines using Ensemble Machine Learning: A Comparative Study. *International Journal of Advanced Computer Science and Applications*. Вип. 10, № 11. DOI:10.14569/IJACSA.2019.0101190.
7. Daud S., Ullah M., Rehman A. та ін. Topic Classification of Online News Articles

Using Optimized Machine Learning Models. *Computers*. Вип. 12, № 1. С. 16. DOI:10.3390/computers12010016.

8. Devlin J., Chang M.-W., Lee K. та ін. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv.org*. 11.10.2018. URL: https://arxiv.org/abs/1810.04805v2 (дата звернення: 08.10.2023).
9. Fesseha A., Xiong S., Emiru E. D. та ін. Text Classification Based on Convolutional Neural Networks and Word Embedding for Low-Resource Languages: Tigrinya. *Information*. Вип. 12, № 2. С. 52. DOI:10.3390/info12020052.
10. Gasparetto A., Marcuzzo M., Zangari A. та ін. A Survey on Text Classification Algorithms: From Text to Predictions. *Information*. Вип. 13, № 2. С. 83. DOI:10.3390/info13020083.
11. HuffPost - Breaking News, U.S. and World News. *HuffPost*. URL: https://www.huffpost.com/ (дата звернення: 25.09.2023).
12. India News Headlines Dataset. URL: https://www.kaggle.com/datasets/therohk/india-headlines-news-dataset (дата звернення: 25.09.2023).
13. Mao K., Xu J., Yao X. та ін. A Text Classification Model via Multi-Level Semantic Features. *Symmetry*. Вип. 14, № 9. С. 1938. DOI:10.3390/sym14091938.
14. NW 1615 L. St, Washington S. 800, Inquiries D. 20036 U.-419-4300 | M.-857-8562 | F.-419-4372 | M. Newspapers Fact Sheet. *Pew Research Center's Journalism Project*. URL: https://www.pewresearch.org/journalism/fact-sheet/newspapers/ (дата звернення: 25.09.2023).
15. Patel V., Ramanna S., Kotecha K. та ін. Short Text Classification with Tolerance-Based Soft Computing Method. *Algorithms*. Вип. 15, № 8. С. 267. DOI:10.3390/a15080267.
16. Rivera G., Florencia R., García V. та ін. News Classification for Identifying Traffic Incident Points in a Spanish-Speaking Country: A Real-World Case Study of Class Imbalance Learning. *Applied Sciences*. Вип. 10, № 18. С. 6253. DOI:10.3390/app10186253.

17. Softmax. *Вікіпедія*. 17.01.2023. URL: https://uk.wikipedia.org/w/index.php?title=Softmax&oldid=38068760 (дата звернення: 12.11.2023).
18. TPU vs GPU: What is better? [Performance & Speed Comparison]. *Windows Report - Error-free Tech Life*. 26.05.2022. URL: https://windowsreport.com/tpu-vs-gpu/ (дата звернення: 25.09.2023).
19. Transform your business with automated text classification. URL: https://blog.pangeanic.com/rule-based-methods-for-text-classification-in-nlp (дата звернення: 25.09.2023).
20. Tunstall L., Werra L. von, Wolf T. Natural Language Processing with Transformers: Building Language Applications with Hugging Face. 1st edition. O'Reilly Media, 2022. 406 с. ISBN 978-93-5542-032-9.
21. Yeasmin S., Kuri R., Rana A. R. M. M. H. та ін. Multi-category Bangla News Classification using Machine Learning Classifiers and Multi-layer Dense Neural Network. *International Journal of Advanced Computer Science and Applications*. Вип. 12, № 5. DOI:10.14569/IJACSA.2021.0120588.
22. Інформаційні технології і автоматизація – 2023. *Матеріали XVI міжнародної науково-практичної конференції.Інформаційні технології і автоматизація – 2023*. (Одеса, 19.10.2023). Одеса : Видавництво ОНТУ, 2023. P. 451.
23. Ahmed J., Ahmed M. ONLINE NEWS CLASSIFICATION USING MACHINE LEARNING TECHNIQUES. *IIUM Engineering Journal*. Vol. 22, Issue 2. P. 210–225. DOI:10.31436/iiumej.v22i2.1662.
24. Ali Ramdhani M., Maylawati D. S., Mantoro T. Indonesian news classification using convolutional neural network. *Indonesian Journal of Electrical Engineering and Computer Science*. Vol. 19, Issue 2. P. 1000. DOI:10.11591/ijeecs.v19.i2.pp1000-1009.
25. BBC News Summary | Kaggle. URL: https://www.kaggle.com/datasets/pariza/bbc-news-summary (accessed 25/09/2023).
26. colab.google. URL: https://colab.google/ (accessed 25/09/2023).

27. Demirci S., Sagiroglu S. TwitterBulletin: An Intelligent and Real-Time Automated News Categorization Tool for Twitter. *JUCS - Journal of Universal Computer Science*. Vol. 28, Issue 4. P. 345–377. DOI:10.3897/jucs.69377.
28. Google Scholar. URL: https://scholar.google.com/ (accessed 10/04/2023).
29. Hartl P., Kruschwitz U. Applying Automatic Text Summarization for Fake News Detection. 2022. DOI:10.48550/ARXIV.2204.01841.
30. Hirnyk Y. Розроблений програмний код. (14.11.2023). URL: https://github.com/MedioNine/MasterDiploma (accessed 28/11/2023).2023.
31. Jurafsky D., Martin J. H. Speech and language processing: an introduction to natural language processing, computational linguistics, and speech recognition. 2nd ed. Upper Saddle River, N.J : Pearson Prentice Hall, 2009. 988 p. [P98 .J87 2009]. ("Prentice Hall series in artificial intelligence" Series). ISBN 978-0-13-187321-6.
32. Lin C.-Y. ROUGE: A Package for Automatic Evaluation of Summaries. *Text Summarization Branches Out*(Barcelona, Spain, 07.2004). Barcelona, Spain : Association for Computational Linguistics, 2004. Also available online, URL: https://aclanthology.org/W04-1013 (accessed 26/09/2023).P. 74–81.
33. Mahajan S. News Classification Using Machine Learning. *International Journal on Recent and Innovation Trends in Computing and Communication*. Vol. 9, Issue 5. P. 23–27. DOI:10.17762/ijritcc.v9i5.5464.
34. Scopus. URL: https://www.scopus.com/search/form.uri?display=advanced (accessed 10/04/2023).
35. Writing Tone Detector and Tone Suggestions | Grammarly. URL: https://www.grammarly.com/tone (accessed 25/09/2023).
36. Lan Z., Chen M., Goodman S. et al. ALBERT: A Lite BERT for Self-supervised Learning of Language Representations. arXiv, 2020. DOI:10.48550/arXiv.1909.11942.
37. Misra R. News Category Dataset. arXiv, 2022. URL: http://arxiv.org/abs/2209.11429 (accessed 25/09/2023).
38. Sanh V., Debut L., Chaumond J. et al. DistilBERT, a distilled version of BERT:

smaller, faster, cheaper and lighter. arXiv, 2020. DOI:10.48550/arXiv.1910.01108.
39. Vaswani A., Shazeer N., Parmar N. et al. Attention Is All You Need. arXiv, 2017. DOI:10.48550/arXiv.1706.03762.